(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
26 December 2024 (26.12.2024)

WIPO | PCT

(10) International Publication Number
WO 2024/263733 A2

(51) International Patent Classification:
*B25J 9/00* (2006.01)

(21) International Application Number:
PCT/US2024/034758

(22) International Filing Date:
20 June 2024 (20.06.2024)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
63/509,151 20 June 2023 (20.06.2023) US

(71) Applicant: **MITEK HOLDINGS, INC.** [US/US]; 802 West Street, Wilmington, Delaware 19801 (US).

(72) Inventors: **BLANCHETTE, Stephane**; 2735, 1142 Rue, Saint-Georges, Québec G6A 1G7 (CA). **DESROCHES, Nicolas**; 952, rue Nicolas Comire, Sainte-Marie, Québec G6E 3X1 (CA). **NADEAU, Jerome**; 404 rue de la Sapiniere, Saint-Georges, Québec G5Y 0N2 (CA).

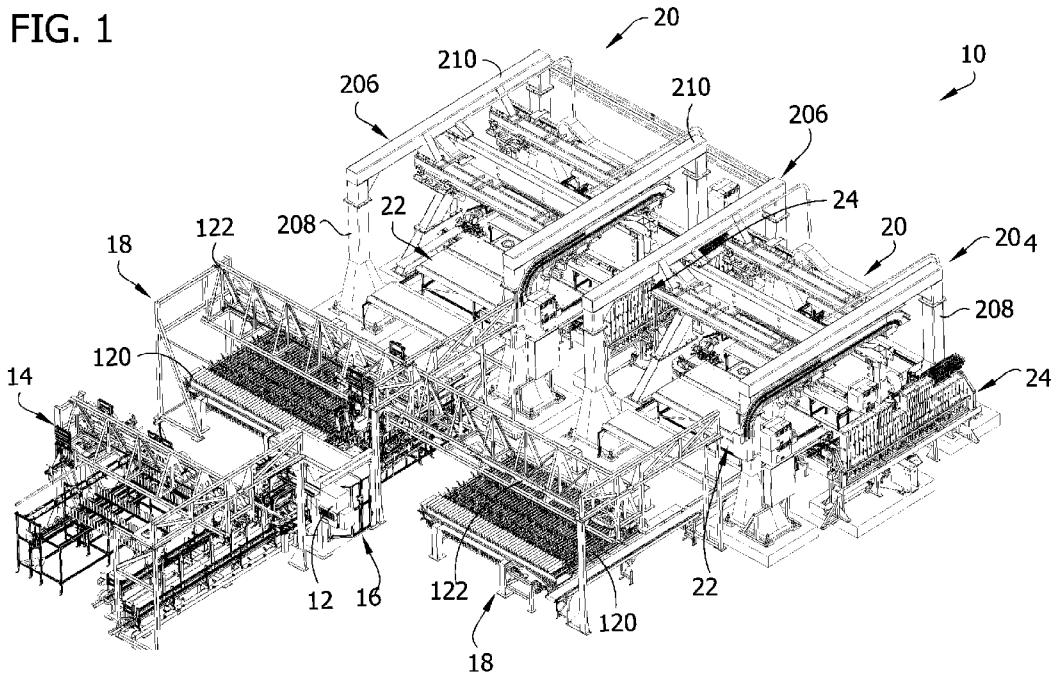
(74) Agent: **JAMES, Kurt F.** et al.; STINSON LLP, 7700 Forsyth Blvd, Suite 1100, St. Louis, Missouri 63105 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CV, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, MG, MK, MN, MU, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, CV, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SC, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, ME, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

(54) Title: AUTOMATED TRUSS MANUFACTURING AND ASSEMBLY SYSTEM

FIG. 1




(57) Abstract: An automated computer-implemented and robotic-implemented assembly method for manufacturing and assembling a wooden structure. The method includes receiving, at a truss manufacture computing device, designs for a plurality of wooden structures and assembling the wooden structures in a joint-by-joint extrusion sequence.

WO 2024/263733 A2

*[Continued on next page]*

**Declarations under Rule 4.17:**

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*

— *as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii))*

**Published:**

— *without international search report and to be republished upon receipt of that report (Rule 48.2(g))*

# AUTOMATED TRUSS MANUFACTURING AND ASSEMBLY SYSTEM

## FIELD

**[0001]** The present disclosure generally relates to an automated lumber manufacture and assembly system and method of automated manufacture and assembly of a wooden structure. More particularly, the disclosure related to an automated truss manufacture and assembly system and automated method of truss assembly.

## BACKGROUND

**[0002]** Over the past several decades, building construction has increasingly employed the fabrication and assembly of building components in off-site facilities. After such fabrication, such components are typically conveyed to and installed at the construction site. A notable example of a building component that may be manufactured and used in this manner is a wooden roof truss. Wooden roof trusses have largely displaced previous approaches to roof manufacture such as rafters.

**[0003]** Rising labor costs and demands for more time- and cost-efficient construction made it desirable to construct such structural building components and modules off site at specialized fabrication facilities that are sometimes referred to as "component manufacturers". In wood frame structures, especially in wood frame residential structures, there are great economies to be realized by using these structural building components, especially wood roof trusses.

**[0004]** Despite the tremendous efficiencies of making building components in component manufacturing facilities, manufacture and assembly of building components continues to be complex and time intensive, despite being far simpler than previous approaches. Therefore, methods of automating the component manufacturing are desirable.

**[0005]** Roof trusses and other building components manufactured by component manufacturers may frequently include multiple pieces of lumber that must be arranged and joined together with specialized connectors such as connector plates. Some exemplary roof trusses may include two top chords, a bottom chord, several webs and many also include overhangs. However, many different variations of roof truss configurations are possible and used. Truss or nailing plates with teeth are typically utilized to securely connect the pieces of lumber together to form the truss. Once assembled, the trusses are typically transported to the construction site and installed.

**[0006]** Where significant quantities of a particular structural component (or element), such as roof trusses, are needed, use of automated systems could further decrease manufacture and assembly (or fabrication) time and lower costs, especially labor costs, attendant to such manufacture and assembly. As a result, automation of such manufacture is desirable, particularly for custom structural designs where the variance in design may make non-automated manufacture or fabrication particularly time consuming due to the analysis, adjustments, and coordination necessary to create building components with varying designs. For example, truss manufacture in a component manufacturing facility requires steps including lumber selection, precutting and marking lumber, measuring, sawing, assembling, and plating the lumber using specialized connector plates. Each of these steps requires specialized knowledge and therefore specialized labor on site.

**[0007]** Through effective automation, faster construction as well as minimized cost could be achieved, and on-site construction errors could also be minimized. However, the complexity of the process and the wide variety of truss configurations (particularly for roof trusses) has made it difficult to create systems or methods to automate all or even a significant range of such configurations and rendered it even more difficult to automate with predictable efficiency. Each step of the process of manufacture has numerous dependencies and complexities. Further, due to construction specifications and norms of construction, structural components (such as roof trusses) must be manufactured and assembled with little tolerance for imprecision. For example, a wood truss typically has low levels of acceptable tolerance for each joint. Thus, an automated system would need to ensure that lumber is obtained, processed, cut, and assembled in a manner that meets these requirements for joint precision.

**[0008]** It would be desirable to utilize automated systems and methods to assemble and manufacture structural components such as wood roof trusses, using a design-based system to automate such manufacture.

## SUMMARY

**[0009]** In one aspect, an automated computer-implemented and robotic-implemented assembly method for manufacturing and assembling a wooden structure is provided. The method includes receiving, at a truss manufacture computing device, designs for a plurality of wooden structures.

**[0010]** The method further includes processing, at the truss manufacture computing device, the designs for the plurality of wooden structures to identify a recipe for manufacturing and assembling each of the plurality of wooden structures defined in each corresponding design.

**[0011]** The recipe for each design includes using the truss manufacture computing device to identify an ordered sequence of lumber components to be used to manufacture the corresponding wooden structure.

**[0012]** The method further includes processing the plurality of recipes, at the truss manufacture computing device, to identify a requested lumber input of stock lumber.

**[0013]** The method further includes obtaining the requested lumber input. In the example embodiment, the truss manufacture computing device prompts a user to identify and place a piece of stock lumber corresponding to the requested lumber input and the user places the stock lumber at an in-feed station.

**[0014]** The method further includes the in-feed station guiding the stock lumber to create the identified lumber pieces through the use of a multi-line saw (located in a cutting station) that cuts the stock lumber according to each recipe. Specifically, the truss manufacture computing device instructs the multi-line saw to make precise cuts along three axes in order to obtain specified lumber pieces of each recipe from the stock lumber.

**[0015]** The method also includes identifying methods of cutting each stock lumber to create the identified lumber pieces according to each recipe, pre-staging the stock lumber into a saw assembly, printing fiducials onto each stock lumber, and cutting the stock lumber according to the identified methods of cutting set forth in each recipe. Each of these steps is defined by a corresponding recipe and/or lumber instructions created based on the recipes, and instructed by the truss manufacture computing device to the corresponding machinery associated with each step. In the example embodiment, the truss manufacture computing device instructs the multi-line saw (and associated machinery in the saw assembly to manipulate the stock lumber) to cut the stock lumber according to the recipe and, more specifically, according to the corresponding design and lumber cutting data. Notably, after the multi-line saw cuts the stock lumber, the truss manufacture computing device causes the resulting lumber pieces to be routed and staged to an appropriate assembly section. In the example embodiment, at least two assembly sections are used, and each lumber piece is routed to the assembly section that corresponds to the design associated with that lumber piece. Accordingly, the multi-line saw receives stock lumber, and generates multiple cut

lumber pieces according to recipes specific to multiple designs, after which each lumber piece is sent to the assembly sections where a wooden structure is assembled that utilizes that lumber piece.

**[0016]** In each assembly station, the wooden structures are assembled based upon the corresponding recipe (generated based upon the corresponding design) in a joint-by-joint extrusion sequence (determined based upon the recipe and the ordered sequence), wherein each joint of wood is assembled by robotic apparatus (described in detail below) and connected by appropriate connectors as set forth by the corresponding recipe. The process continues in each assembly station until the entire wooden structure is assembled. Each joint of the wooden structure is formed by arranging lumber pieces relative to each other at a joint forming station and securing the lumber pieces together using connectors (e.g. nailing plates) to form the joints at the joint forming station.

**[0017]** As described in greater detail below, the disclosure further includes systems, mechanism, assemblies, computing devices, and software to achieve the methods described above and herein.

**[0018]** Specifically, the disclosure includes a manufacturing and assembly system designed to carry out the methods described herein driven, at least partially, by the truss manufacture computing devices and robotic systems described.

**[0019]** The manufacturing and assembly system further includes component systems including at least some of: (a) at least one in-feed station, (b) at least one in-feed manipulator assembly having an in-feed manipulator, (c) in-feed conveyors; (d) in-feed lines; (e) fiducial printers, (f) sensors, (g) cutting stations, (h) buffer stations, and (i) assembly modules. Details of each of these components and systems are described below.

**[0020]** Other objects and features of the present disclosure will be in part apparent and in part pointed out hereinafter.

## BRIEF DESCRIPTION OF THE DRAWINGS

**[0021]** FIG. 1 is a perspective of one embodiment of a manufacturing and assembly system according to the present disclosure;

**[0022]** FIG. 2 is a top view of the manufacturing and assembly system;

**[0023]** FIG. 3 is a perspective of an in-feed station of the manufacturing and assembly

system;

**[0024]** FIG. 4 is a top view of the in-feed station;

**[0025]** FIG. 5 is a perspective of an in-feed manipulator assembly;

**[0026]** FIG. 6 is a perspective an in-feed manipulator of the in-feed manipulator assembly;

**[0027]** FIG. 7 is an enlarged fragmentary perspective of the in-feed manipulator;

**[0028]** FIG. 8A is a perspective of the in-feed station showing in-feed lines of the in-feed station;

**[0029]** FIG. 8B is an enlarged fragmentary perspective of the in-feed station in FIG. 8A showing first sensors;

**[0030]** FIG. 8C is an enlarged fragmentary perspective of the in-feed station in FIG. 8A showing second sensors;

**[0031]** FIG. 8D is an enlarged fragmentary perspective of the in-feed station in FIG. 8A showing third sensors;

**[0032]** FIG. 8E is an enlarged fragmentary perspective of the in-feed station in FIG. 8A showing fourth sensors;

**[0033]** FIG. 9 is a perspective of a piece of stock lumber having fiducials printed thereon;

**[0034]** FIG. 10 is a perspective of a cutting station with doors of a cabinet removed to show an interior of the cabinet;

**[0035]** FIG. 11 is an enlarged fragmentary perspective of the cutting station showing a clamp holding a lumber piece;

**[0036]** FIG. 12 is a top view of the cutting station showing a pair of clamps each holding a lumber piece and a saw disposed adjacent a first lumber piece;

**[0037]** FIG. 13 is a top view of the cutting station showing a pair of clamps each holding a lumber piece and a saw disposed adjacent a second lumber piece with portions broken away to show underlying detail;

**[0038]** FIG. 14 is a perspective of a discharge area of the cutting station;

**[0039]** FIG. 15 is a perspective of a buffer station of the manufacturing and assembly system;

**[0040]** FIG. 16 is an enlarged fragmentary perspective of a buffer table at the buffer

station showing slots on the buffer table;

**[0041]** FIG. 17 is an enlarged fragmentary perspective of the buffer table showing an outlet end of the buffer table and pushers on the table in a retracted position;

**[0042]** FIG. 18 is an enlarged fragmentary perspective of the buffer table showing pushers on the table in an extended position;

**[0043]** FIG. 19 is an enlarged fragmentary perspective of the buffers station showing a manipulator assembly at the buffer station;

**[0044]** FIG. 20 is a perspective of a manipulator of the manipulator assembly in FIG. 19;

**[0045]** FIG. 21 is an enlarged fragmentary perspective of the manipulator in FIG. 20;

**[0046]** FIG. 22 is a perspective of an assembly module of an assembly station of the manufacturing and assembly system;

**[0047]** FIG. 23 is a side view of the assembly module;

**[0048]** FIG. 24 is a perspective of a lower platen assembly at the assembly station;

**[0049]** FIG. 25 is an enlarged fragmentary perspective of the lower platen assembly in FIG. 24 with a cover plate removed;

**[0050]** FIG. 26 is a perspective of a first robot of an assembly module;

**[0051]** FIG. 27 is a bottom perspective of a tool of the first robot;

**[0052]** FIG. 28 is a bottom view of the tool;

**[0053]** FIG. 29 is an illustration of a focal zone of a vision system viewing a fiducial on a lumber piece;

**[0054]** FIG. 30 is a front perspective of a plate distribution assembly at the assembly station;

**[0055]** FIG. 31 is an enlarged fragmentary perspective of the plate distribution assembly in FIG. 30;

**[0056]** FIG. 32 is a rear perspective of the plate distribution assembly;

**[0057]** FIG. 33 is a rear view of the plate distribution assembly;

**[0058]** FIG. 34 is an enlarged fragmentary perspective of the plate distribution assembly in FIG. 32 showing a first plate handling assembly;

**[0059]** FIG. 35 is an enlarged fragmentary perspective of the plate distribution assembly in FIG. 32 showing the first plate handling assembly with a separator removed;

**[0060]** FIG. 36 is a perspective of the separator;

**[0061]** FIG. 37 is an enlarged fragmentary perspective of the plate distribution assembly in FIG. 32 showing a specialty plate conveyor;

**[0062]** FIG. 38 is an enlarged fragmentary perspective of the plate distribution assembly in FIG. 32 showing a second plate handling assembly;

**[0063]** FIG. 39 is an illustration of a selector placing a nailing plate on a platen assembly;

**[0064]** FIG. 40 is an enlarged fragmentary perspective of the assembly station showing a lumber piece being delivered to the assembly station by an assembly conveyor;

**[0065]** FIG. 41 is an illustration of a first robot placing a first lumber piece at a reference location on the assembly table;

**[0066]** FIG. 42 is an illustration of a second robot holding the first lumber piece at the reference location with the first robot moved back to the assembly conveyor to retrieve a second lumber piece;

**[0067]** FIG. 43 is an illustration of the first and second robots holding the first and second lumber pieces on the assembly table at the reference location;

**[0068]** FIG. 44 is an illustration of the first and second lumber pieces positioned on the assembly able at the reference location and the first robot holding a third lumber piece for placement at the reference location;

**[0069]** FIG. 45 is an illustration of a fully assembled wooden structure on the assembly table;

**[0070]** FIGS. 46-53 show a schematic illustration of lumber pieces being attached in the joint-by-joint extrusion sequence to assemble a wooden structure;

**[0071]** FIG. 54 is a flow chart of a recipe routine performable by a controller of the manufacturing and assembly system;

**[0072]** FIG. 55 is a flow chart of a robot calibration routine performable by the controller of the manufacturing and assembly system;

**[0073]** FIG. 56 is a flow chart of a tool calibration routine performable by the controller of the manufacturing and assembly system;

**[0074]** FIG. 57 is a flow chart of a placement calibration routine performable by the controller of the manufacturing and assembly system; and

**[0075]** FIG. 58 is schematic illustration of the placement calibration routine;

**[0076]** FIG. 59 is a functional block diagram of an example truss manufacture computing device that may be used to control the operation of the manufacturing and assembly system for fabricating wooden trusses, or components thereof, as described.

**[0077]** FIG. 60 is a flow diagram representing an example method for controlling the operation of the manufacturing and assembly system for fabricating wooden trusses using the truss manufacture computing devices shown in FIG. 59;

**[0078]** FIG. 61 is a diagram of elements of one or more example truss manufacture computing devices that may be used in the system shown in FIGS. 59 and 60;

**[0079]** FIGS. 62-64 are flowcharts of at least one embodiment of a method that may be performed by the system of FIG. 1 for preprocessing input data to preemptively correct errors that may result in the production of a wooden structure;

**[0080]** FIGS. 65-69 are diagrams of user interfaces that may be produced by the system of FIG. 1;

**[0081]** FIGS. 70-111 are flowcharts of at least one embodiment of a method and sub-methods that may performed by the system of FIG. 1 to generate a recipe to coordinate operations of components of the system to produce one or more wooden structures;

**[0082]** FIGS. 112-124 are flowcharts of at least one embodiment of a method and sub-methods that may be performed by the system of FIG. 1 to efficiently select materials for the production of one or more wooden structures;

**[0083]** FIG. 125 is a diagram of an embodiment of a microservices architecture that may be utilized by the system of FIG. 1;

**[0084]** FIGS. 126-127 are flowcharts of at least one embodiment of a method that may be performed by the system of FIG. 1 to produce one or more wooden structures using a microservices architecture; and

**[0085]** FIGS. 128-130 are flowcharts of at least one embodiment of a method that may be performed by the system of FIG. 1 to print and utilize fiducials on one or more boards in the production of one or more wooden structures.

**[0086]** Corresponding parts are indicated by corresponding reference characters throughout the several views of the drawings.

## DETAILED DESCRIPTION

**[0087]** As described above, and herein, the present disclosure provides an automated manufacturing and assembly system (hereinafter referred to as "an automated manufacturing system") that is computer-implemented and supported through automated solutions and components including in-feeds, conveyors, sensors, multi-line saws, and robots for placing and connecting elements of wooden structures such as wooden trusses. Further, although each of the disclosed mechanical, automation, computing, and robotic elements can be used separately for specific functions, it is contemplated that they may be used in conjunction to create wooden structures. In an example embodiment, the automated manufacturing system is configured to receive, at a truss manufacture computing device, design files describing the structure, geometric design, and connectivity of wood trusses including details regarding lumber elements and connector plates. The truss manufacture computing device simulates multiple approaches to manufacture and assemble wooden structures (e.g., wood trusses) for each design and selects one having preferred simulation characteristics. The truss manufacture computing device further creates "recipes" associated with the manufacture of each wooden structure created by the manufacturing system. As described herein, a "recipe" includes the organized elements and steps used by the automated manufacturing system to create a particular wooden structure corresponding to the inputted design according to a preferred manner of manufacture and assembly. In the example embodiment, the automated manufacturing system applies the recipes to create each wooden structure with a joint-by-joint extrusion model of manufacturing and assembly. In an example embodiment, the design files are generated by a structural design software such as Structure from MiTek Sapphire®. The design files may be uploaded to the truss manufacture computing device in any suitable manner.

**[0088]** Further, in the example embodiment, the automated manufacturing system is configured to create multiple wooden structures in parallel fashions. As described herein, the automated manufacturing system is specifically configured to manufacture and assemble two wooden trusses in a simultaneous manner, each according to distinct recipes, in two distinct assembly stations. In order to accomplish this parallel processing, the automated manufacturing system obtains and creates the elements for each recipe to each distinct assembly station.

**[0089]** To effect this approach, the truss manufacture computing device processes each recipe to create lumber instructions to obtain the wood elements needed for each assembly from

commonly used stock lumber. Thus, as described below, the truss manufacture computing device analyzes each design to obtain recipes, and then analyzes each recipe to identify which wooden elements are needed to assemble each wood truss in the sequence set forth in the corresponding recipe. The truss manufacture computing device further identifies an appropriate piece of stock lumber from a listing of available lumber, such that each identified piece of stock lumber can be efficiently cut to obtain the identified wooden elements for assembly according to each recipe. In many examples, the identified piece of stock lumber is selected as one optimized to provide elements for manufacturing each truss in sequence, with minimal or no unused lumber.

**[0090]** Accordingly, the automated manufacturing system simultaneously processes multiple recipes to drive the use of the systems and methods below. It should be understood that the manufacturing system is in communication with, and controlled by, the truss manufacture computing device to ensure that the use of such components corresponds to the recipe for each wooden structure.

**[0091]** Referring to FIGS. 1 and 2, an automated wooden structure manufacturing system constructed according to the principles of the present invention is generally indicated at 10. The manufacturing system 10 is an automated manufacturing system that is configured to stage, cut, and assemble pieces of lumber LP (Fig. 44) into finished wooden structures WS (Fig. 45) including multiple lumber pieces attached together. In one embodiment, the manufacturing system 10 is configured to assemble trusses (e.g., roof trusses). Thus, the following description will describe the manufacturing system 10 within the context of truss assembly. However, the manufacturing system 10 and any methods of assembly have application to the production of other items. For example, and without limitation, the manufacturing system 10 could be used in the manufacture of wall frames and floor trusses, as well as other articles of manufacture such as furniture components. Therefore, the manufacturing system 10 may be used to cut and assemble pieces of lumber to create any wooden structure, particularly (but not exclusively) those which are joined together using nailing plates P. "Boards", "lumber", "lumber members" “lumber pieces” and "pieces of lumber" are intended to be interchangeable herein unless the context clearly indicates the contrary.

**[0092]** Broadly, the lumber pieces may be manipulated by an automated manufacturing system to form a construct of distinct pieces of lumber connected together that has utility as a

finished product or as a part of a larger construction. Although a fully automated manufacturing system 10 is disclosed, a semi-automated system could be used without departing from the scope of the disclosure. Thus, at least some aspects of the wooden structure assembly may be performed manually using components of the manufacturing system 10 or components separate from the system. Additionally, the manufacturing system 10 may include a truss manufacture computing device 12 configured to control operation of the manufacturing system (e.g., the operation of each station thereof). In the illustrated embodiment, a single truss manufacture computing device 12 is identified. However, it will be understood that multiple computing devices or control systems operatively connected to each other and/or a central computer can be utilized to control operation of the system 10 without departing from the scope of the disclosure. In one embodiment, the system 10 includes five (5) computing device operatively connected to each other. The features of the truss manufacture computing device 12 are described in detail in FIGS. 58-60.

**[0093]** Still referring to FIGS. 1 and 2, the manufacturing system 10 comprises an in-feed station 14 configured for receiving stock lumber SL (Fig. 3) and staging the stock lumber prior to being cut into lumber pieces LP for use in the assembly of the wooden structure WS (Fig. 45). A cutting station 16 is in communication with an outlet of the in-feed station 14 and is configured to cut the stock lumber SL into the lumber pieces LP. A pair of buffer stations 18 are disposed at an outlet of the cutting station 16 and are configured to place and arrange the cut piece of lumber LP for being delivered in a predetermined order to dedicated assembly stations 20. Each assembly station 20 comprises an automated assembly module 22 for positioning, arranging, and securing the lumber pieces LP together to assemble the wooden structure WS, and a plate distribution assembly 24 for supplying the fasteners (i.e., nailing plates P) for securing the lumber pieces together. The truss manufacture computing device 12 is operatively connected to each station within the manufacturing system 10 and controls the operation of the individual stations. For instance, the truss manufacture computing device 12 is configured to compile the type (e.g., size and end cuts) and order of lumber pieces LP to be cut, and the sequence in which to deliver the cut lumber pieces LP to the assembly module 22 for being assembled into the wooden structure WS. The pair of buffer stations 18 and pair of assembly stations 20 also equips the system 10 with the capability of simultaneously assembling two separate jobs (i.e., sets of trusses). Thus, the instructions for a first set of trusses can be carried out at a first assembly

station 20 and the instructions for a second set of trusses can be carried out a second assembly station 20 at the same time. Additional details of the operational control capabilities for the truss manufacture computing device 12 are provided below.

**[0094]** Referring to FIGS. 3-7, the in-feed station 14 comprises an in-feed conveyor 26 (broadly, a stock lumber receiver) configured for receiving stock lumber SL from an inventory of lumber. In one embodiment, the stock lumber SL is placed on the in-feed conveyor 26 in a "vertical" orientation whereby opposite major side surfaces are oriented such that the major surfaces may extend generally along vertical planes with the major surfaces facing in opposite horizontal directions, and opposite minor side surfaces are oriented such that the minor surfaces may extend generally along horizontal planes and face the upward and downward directions, respectively. A lift 28 may be disposed adjacent the in-feed conveyor 26 and be configured to raise the stock lumber SL to an elevated height above the in-feed conveyor. The lift 28 is configured to locate the stock lumber SL at a height for being transported along the in-feed station 14 as will be explained in greater detail below. It will be understood that the in-feed conveyor 26 and/or lift 28 may have other constructions or be omitted without departing from the scope of the disclosure. The in-feed conveyor 26 may also be configured to receive the stock lumber SL in other orientations without departing from the scope of the disclosure.

**[0095]** The truss manufacture computing device 12 may provide instructions to an operator as to the order of stock lumber SL to be loaded onto the in-feed conveyor 26. In one embodiment, a monitor 30 at the in-feed station 14 has a display for providing a visual prompt of the stock lumber SL to be loaded onto the in-feed conveyor 26. Thus, the monitor 30 may show the next stock lumber SL or a series of stock lumber to be loaded onto the in-feed conveyor 26 as a picture of the stock lumber with accompanying text to further identify the stock lumber. For instance, the display may show text and an image of a 2x4x16 MSR 1650 board which prompts the operator to select the identified board from an inventory of lumber. It will be understood that the monitor 30 may provide other prompts for identifying the stock lumber SL without departing from the scope of the disclosure. In the example embodiment, monitor 30 provides the visual prompt to request lumber according to the recipes, provided by truss manufacture computing device 12, for each of the designs, such that the requested lumber is selected as lumber that may be processed to provide lumber pieces to manufacture wooden structures for each design.

**[0096]** Sensors 32 may be disposed along the length of the in-feed conveyor 26 to

measure one or more characteristics of the stock lumber SL to verify that the proper stock lumber has been loaded onto the conveyor. For instance, first sensors 32A may measure a length of the stock lumber, and second sensors 32B may measure a width of the stock lumber SL. If either of the measured length or width dimensions fall outside of a tolerance range for the particular stock lumber SL identified by the truss manufacture computing device 12 for loading onto the in-feed conveyor 26, the sensors 32A, 32B may indicate to the controller that an incorrect stock lumber has been loaded. In this instance, the stock lumber SL is dropped down from the in-feed conveyor 26 to a return or reject line 34 for delivering the stock lumber back to the operator. The operator may then attempt to retrieve the correct stock lumber SL from the inventory. If the length and width dimensions are measured to be within the correct ranges, then the lift 28 is operated to raise the stock lumber SL up for being transported along the in-feed station 14. In one embodiment, the sensors 32A, 32B comprise distance detectors. The in-feed station 14 may also be provided with a lumber detection sensor 32C for detecting when the stock lumber SL has reached the position for being elevated by the lift 28.

**[0097]** A buffer table 36 is located at the in-feed station 14 and comprises support bars 38 extending along a length of the buffer table, and a plurality of fences 40 extending upward from the support bars. The fences 40 are spaced apart from each other along a width of the buffer table 36. Each fence 40 comprises a plurality of fence posts 42 spaced apart from each other along a length of the fence and defining openings between adjacent fence posts. The fence posts 42, and thus the openings between the fence posts, of each fence 40 are aligned along the length of the support surface 38 such that slots 44 (FIG. 4) extending along the width of the support surface are formed between the fence posts. The slots 44 are sized to receive the stock lumber SL loaded onto the in-feed conveyor 26 as will be explained in greater detail below. In the illustrated embodiment, there are twenty-five (25) slots 44 provided by the fences 40. However, a lesser or greater number of slots 44 may be provided. The slots 44 may be divided into groups to control the flow of stock lumber SL through the manufacturing system 10. In one embodiment, the slots 44 are divided into three groups. For example, a first group of slots 44 may be assigned to a first saw line, and a second group of slots may be assigned to second saw line. A third group of slots 44 may be assigned to an auxiliary line. The group assignments for an individual slot 44 may change throughout the course of an assembly process. However, throughout the assembly process, each of the three groups may continue to have a predetermined

number of slots associated with its group. For example, for the twenty-five (25) slots 44 in the illustrated embodiment, ten (10) slots may be provided for the first saw line, ten (10) slots may be provided for the second saw line, and five (5) slots may be provided for the auxiliary line. It will be understood that the slots 44 could be subdivided into a different number of groups assigned for different locations without departing from the scope of the disclosure.

**[0098]** Referring to FIGS. 3 and 5-7, a manipulator assembly 46 is generally disposed above the in-feed conveyor 26 and buffer table 36. The manipulator assembly 46 is configured to transport the stock lumber SL across the in-feed station 14 to the saw lines. The manipulator assembly 46 comprises a generally triangular support frame 48 and rails 50 on the support frame. A manipulator 52 is mounted to the support frame 48 and is movable along the rails 50 to translate the manipulator along the length of the in-feed station 14. The manipulator 52 comprises a carriage 54 moveably attached to the rails 50 to translate the manipulator across the in-feed station 14. The carriage 54 has a prime mover 56 and rollers 58 operatively connected to the prime mover for engaging the rails 50 to move the manipulator 52 along the support frame 48. A support 60 of the manipulator 52 may be moveably attached to the carriage 54. In the illustrated embodiment, the support 60 is moveable in a vertical direction to raise and lower the support above the in-feed conveyor 26 and buffer table 36. Thus, in the raised position, the manipulator 52 is configured to translate across the in-feed station 14, and in the lowered position, the manipulator is positioned to engage the stock lumber SL. To that effect, a plurality of grip fingers 62 are mounted to the support 60 and are moveable between open and closed positions for grasping the stock lumber SL. The grip fingers 62 are configured to retain the stock lumber SL to the manipulator 52 to carry the stock lumber across the in-feed station 14.

**[0099]** Sensors 64 may be provided on the manipulator 52 (e.g., support 60) to monitor the position of the manipulator. In one embodiment, one or more first sensors 64A detect a vertical position of the support 60 to determine if the manipulator 52 is in a raised or lowered position. In one embodiment, the first sensors 64A comprise proximity sensors. This information is used by the system 10 to verify the position of the manipulator 52 before a manipulator action, such as translating along the rails 50 or engaging the stock lumber SL. In the instance where the sensors 64A detect that the support 60 is in the lowered position, the grip fingers 62 can be actuated from an open position to a closed position to grasp stock lumber SL below the manipulator. One or more second sensors 64B may detect a position of the grip

fingers 62 to confirm whether the grip fingers are in the open or closed position. In one embodiment, the second sensors 64B comprise proximity sensors. Thus, with the manipulator 52 detected to be in the lowered position with the grip fingers 62 in the closed position, the system 10 will register that the manipulator has grasped the stock lumber SL. Additionally, third sensors (not shown) may be provided on the ends of the grip fingers 62 to detect the presence of the stock lumber SL once the stock lumber has been grasped. The slots 44 may also provide a verification of the orientation and shape of the stock lumber SL by requiring the stock lumber to fit within the slot in which the manipulator 52 carries the stock lumber. In this case, if there is any issue with the way in which the stock lumber SL has been handled or with the shape of the stock lumber, such as being significantly bowed out of plane, the inability of the manipulator 52 to deliver the stock lumber to the desired slot 44 will signal to the system 10 and the operator that there is an issue with the stock lumber. This may be indicated by the first sensors 64A not detecting that the support 60 has reached the lowered position after having grasped the stock lumber SL. Still other methods for verifying the location of the stock lumber SL are envisioned without departing from the scope of the disclosure.

**[00100]** Referring to FIGS. 6 and 7, the grip fingers 62 are arranged in groups of three (3) that are spaced apart along a length of the support 60. A spacing of the grip finger groups is such that the grip fingers 62 are located at positions along the width of the buffer table 36 which position the grip fingers between the fences 40. Thus, the grip fingers 62 will not interfere with or contact the fences 40 when the manipulator 52 is moved to its lowered position. Each grip finger 62 comprises an elongate blunt blade member. The grip fingers 62 are pivotally connected to a lower part of the support 60. When the grip fingers 62 are actuated by a pneumatic cylinder, two of the grip finger pivot conjointly toward the other grip finger to move to the closed position to grip the stock lumber SL, and pivot away from each other to move to the open position to release the stock lumber. The spacing between the grip fingers 62 in each group is such that the grip fingers are configured to engage side surfaces of the stock lumber SL. Thus, for stock lumber SL having a pair of opposing major side surfaces and a pair of opposing minor side surfaces, the grip fingers 62 are configured to engage only one of said major or minor side surfaces to grasp the stock lumber SL. In the embodiment where the stock lumber SL is loaded onto the in-feed conveyor 26 in the "vertical" orientation, the grip fingers 62 will be configured to engage the major side surfaces. Additionally, the grip fingers 62 are configured such that they

do not pierce or puncture the stock lumber SL when engaged with the stock lumber. Spaced apart ribs at the ends of the fingers 62 augment gripping of the stock lumber SL without penetrating the lumber. Thus, the stock lumber SL is not damaged or otherwise compromised during the handling by the manipulator 52. A distance between outermost pairs of grip fingers 62 may be about 10 ft. A spacing between adjacent grip finger groups may be between about 26 inches and 36 inches. Thus, the manipulator 52 is configured to handle stock lumber SL ranging from a size of about 4 ft. long to about 16 ft. long. In the illustrated embodiment there are five (5) grip finger groups spaced along a length of the support 60. However, it will be understood that a different number of grip finger groups may be utilized. Additionally, ridges or other frictional members may be provided on the grip fingers 62 to facilitate grasping the stock lumber SL with the grip fingers with increased grip strength. In other examples, shock absorbers may be used on the manipulator 52 to reduce dampen and/or control the rate of travel by the manipulator 52 and otherwise limit collision with other members of the system.

**[00101]** Referring to FIGS. 3-5, the manipulator 52 is movable along the rails 50 to position the manipulator above the in-feed conveyor 26 to pick up stock lumber SL on the in-feed conveyor. The manipulator 52 is then moveable to place the stock lumber SL in one of the slots 44 on the buffer table 36. When it is time to deliver the stock lumber SL to the cutting station 16 (FIG. 1), the manipulator 52 is movable along the rails 50 to pick up the stock lumber in the slots 44 and position the manipulator above in-feed lines 66A, 66B (FIG. 4) located at an opposite end of the buffer table 36 from the in-feed conveyor 26 to deliver the stock lumber SL to one of the in-feed lines. In the illustrated embodiment, there are a pair of in-feed lines 66A, 66B. A first in-feed line 66A is disposed proximate to the buffer table 36, and a second in-feed line 66B is laterally spaced from the first in-feed line away from the buffer table. The in-feed lines 66A, 66B extend along respective in-feed axes IA. The two in-feed axes IA are parallel to each other and extend along the width of the buffer table 36. Thus, the manipulator 52 is able to deliver the stock lumber SL to one of the in-feed lines 66A, 66B while maintaining the stock lumber SL in the same orientation in which it was held on the buffer table 36. As will be explained in greater detail below, the in-feed lines 66A, 66B extend to the cutting station 16 for delivering the stock lumber SL to the cutting station. Therefore, the in-feed station 14 is equipped with two separate lines for delivering stock lumber SL to the cutting station 16. This allows for two separate lines of stock lumber SL to be cut at the cutting station 16, and for two

pieces of stock lumber to be cut at the same time. As a result, a higher throughput is achievable with this configuration of the in-feed station 14.

**[00102]** Referring to FIG. 8A-8E, each in-feed line 66A, 66B comprises an alignment wall 68 and a fence 70 disposed opposite the alignment wall. A channel 72 is formed between the alignment wall 68 and fence 70 and is configured to initially receive a piece of stock lumber SL dropped by the manipulator 52. Stops 74 are movable into and out of the channel 72. When the stops 74 are disposed in the channel 72, the stops prevent the stock lumber SL from falling through the channel 72 when dropped by the manipulator 52. In the illustrated embodiment, the stops 74 comprises actuatable pins movable between retracted and extended positions. However, the stops 74 could have other configurations without departing from the scope of the disclosure. When the stock lumber SL is held in the channels 72 by the stops 74 the stock lumber may be said to be in a holding portion of the in-feed line 66. When the stops 74 are moved out of the channel 72, the stock lumber SL is free to drop down into a transport portion of the in-feed line 66. The transport portion of the in-feed line 66 stops the vertical movement of the stock lumber SL. Thus, the transport portion of the in-feed line 66 is located below the holding portion of the in-feed line. At the transport portion, a plurality of clamps 76 are actuatable to clamp the stock lumber SL against the alignment wall 68. The alignment wall runs parallel to the in-feed axis IA, thus the clamps 76 are configured to straighten the stock lumber SL such that a longitudinal axis of the stock lumber is generally coincident with the in-feed axis. While the stock lumber SL is clamped against the alignment wall 68, carriers 78 engage the stock lumber SL. The carriers 78 are located below the channel 72 at the transport portion of the in-feed line 66 and are movable with a belt 80 to transport the stock lumber SL along the in-feed axis IA. The carriers 78 may comprise clamps/pushers for securing and pushing the stock lumber SL. Thus, the carriers 78 may hold the stock lumber SL in the straightened configuration produced by the clamps 76. With the stock lumber SL secured in the carriers 78, the stock lumber is then transported along the in-feed axis IA to a predetermined "home" position. The "home" position is a known distance from the cutting station 16 (i.e., saw) which allows the system 10 to deliver the stock lumber SL to precise locations for being cut in the cutting station, as will be explained in greater detail below.

**[00103]** Sensors 82 may be provided along each in-feed line 66A, 66B (Fig. 4). One or more first sensors 82A may be provided on the stops 74 to detect when the stops are extended or

retracted (FIG. 8B). One or more second sensors 82B (FIG. 8E) may be provided to detect the presence of the stock lumber SL once it has been dropped down onto the extended stops 74 thereby instructing the system 10 on whether a piece of stock lumber SL is being held by the stops 74. In one embodiment, the second sensors 82B comprise photoelectric sensors that look for the physical presence of the stock lumber SL. When the sensors 82A detect that the stops 74 are retracted, the system will interpret the signal from the sensors that the stock lumber SL has been dropped to the transport portion. Additionally, one or more third sensors 82C may be provided on the clamps 76 to detect the movement of the clamps from the open to closed positions (FIG. 8C). The system 10 checks to see if the third sensors 82C indicate that the clamps 76 are open. If so, the stops 74 will be retracted to drop the stock lumber SL into the transport portion of the in-feed line 66A, 66B. The clamps 76 may then be actuated to hold the stock lumber SL in place positioning the stock lumber for retrieval by the carriers 78. One or more fourth sensors 82D (FIG. 8D) may be provided on the carriers 78 to detect the presence of the stock lumber SL on the carrier once the carrier has been moved to grab the stock lumber. The system 10 communicates with the carriers 78 so that the carriers know the size of the stock lumber SL so that the carriers move to the appropriate location to grab the stock lumber. Detection of the stock lumber SL by sensors 82D instructs the carriers 78 to close to grab the stock lumber. Once the stock lumber SL has been grasped by the carriers 78, the carriers transport the stock lumber SL along the in-feed line 66A, 66B.

**[00104]** One or more fifth sensors 82E (FIG. 8C) may be disposed near an outlet side of the in-feed line 66A, 66B to detect when the carriers 78 have delivered the stock lumber SL to the “home” or "zero" position (FIG. 8E). The fifth sensors 82E instruct the printer 85 and the saw 94 of the location of stock lumber SL as will be discussed in greater detail below. In one embodiment, the fifth sensors 82E comprise photoelectric sensors for detecting the presence of the stock lumber SL. The series of sensors 32, 64, 82 throughout the in-feed station 14 equips the system 10 to precisely track the position and orientation of the stock lumber SL being transported through the in-feed station. As a result, the system 10 is able to repeatedly deliver a straight piece of stock lumber SL to a precise location relative to the cutting station 16 to help ensure that precise cuts are made at the cutting station. This ultimately facilitates the proper assembly of the wooden structures WS.

**[00105]** Referring to FIGS. 4, 8, and 9, a fiducial printer 85 is disposed along the in-feed

lines 66A, 66B and is configured to print fiducials 86 (i.e., indicia) on the stock lumber SL carried along the two in-feed lines as it is being delivered to the cutting station 16. The printing instructions for the fiducial printer 85 that describe where to print on the stock lumber SL are determined based upon the recipe associated with each wooden structure and its associated design such that the fiducials allow for robotic assembly according to the recipe, using the assembly modules described below. As such, the fiducial printer 85 is in communication with truss manufacture computing device 12 to receive such instruction. Thus, the printer 85 is configured to print the fiducials 86 onto the stock lumber SL prior to the stock lumber being cut at the cutting station 16. First fiducials 86A are configured to be printed on a major side surface of the stock lumber SL, and second fiducials 86B are configured to be printed on a minor side surface of the stock lumber. As will be discussed in greater detail below, the first fiducials 86A provide a reference point for the assembly module 22 to indicate the position of the cut piece of lumber LP at a particular joint of the wooden structure WS. Additionally, the second fiducials 86B may indicate to an operator an order and/or leading end direction of the stock lumber SL such that the cut lumber pieces LP produced from the stock lumber SL can be tracked by the operator to confirm that the proper lumber pieces are being delivered to the assembly station 20 at the proper time and in the proper orientation. In one embodiment, the printer 85 may have multiple printer heads that are adjustable vertically to accommodate the size of the stock lumber SL.

**[00106]** Referring to FIGS. 10-13, the cutting station 16 is disposed at an end of the in-feed lines 66A, 66B of the in-feed station 14. The cutting station 16 comprises a cabinet 88 (broadly, a saw compartment) at least partially housing a saw assembly 90 configured to cut the stock lumber SL traveling along both in-feed lines 66A, 66B. The saw assembly 90 comprises a robot or robotic arm 92, and a saw 94 mounted on the robotic arm. The robotic arm 92 comprises multiple arm members 96 moveably connected to each other providing the robotic arm with six degrees of freedom of movement. The saw 94 is mounted on an end member 96 of the robot arm 92 and is rotatable about the end member. Therefore, the saw 94 can be moved to either side of the stock lumber SL on both in-feed lines 66A, 66B to make the necessary cuts to the stock lumber. Thus, the robotic arm 92 is configured to move the saw 94 along planes that intersects the in-feed lines 66A, 66A to make a cut through the width and thickness of the stock lumber SL. The saw 94 can make more than one cut in the same general location on the stock

lumber SL to shape an end of a lumber piece LP as desired. In the illustrated embodiment, the saw 94 includes a circular saw blade having teeth arranged about a circumference of the saw. One example of such a saw is the MiTek Blade Linear Saw. It is understood that the saw assembly 90 and, more specifically robotic arm 92 and saw 94 are configured to receive instructions from truss manufacture computing device 12.

**[00107]** A pair of board holders 97 are moveably disposed in the cabinet 88 and configured to receive the stock lumber SL delivered along the respective in-feed lines 66A, 66B. Thus, the carriers 78 transport the stock lumber SL into the cabinet 88 where the holders 97 receive the stock lumber for cutting. In the illustrated embodiment, the holders 97 comprise clamps for locking the stock lumber SL in position relative to the holders. Both holders 97 are translatable within the cabinet 88 along a track 98 that extends parallel to and coincident with the respective in-feed lines 66A, 66B. This equips the holders 97 to pull the stock lumber SL through the cabinet 88 so that the saw 94 can make the necessary cuts. Each holder 97 may comprise a bottom wall 100 and a pair of side walls 102 extending upward from the bottom wall. At least one of the side walls 102 may be moveable relative to the other side wall in order to open and close a space between the side walls. This configures the holders 97 to clamp the stock lumber SL between the side walls 102 so that the stock lumber is retained in the holders during the cutting process. In a preferred embodiment, both side walls 102 are movable. In the illustrated embodiment, a single holder 97 is shown for each saw line. However, an additional fixed holder (not shown) may be added between the carriers 78 and the holder 97 to hold the stock lumber SL in place to allow the holder 97 to move to the desired position along the stock lumber. For instance, the holder 97 may be moved to grab the stock lumber SL at a location that facilitates cutting the stock lumber as close as possible to the holder. The additional fixed holder can also facilitate handling of smaller pieces of stock lumber. In such an embodiment, each saw line will have fixed holder and a moveable holder 97. Still more holders 97 may be utilized for each line 66 without departing from the scope of the disclosure.

**[00108]** Referring to FIG. 13, a first sensor 104 may be provided on each holder 97 to detect a position of the stock lumber SL in the holder. In one embodiment, the first sensor 104 is disposed on the bottom wall 100 of the holder 97. The first sensor 104 is directed upward from the bottom wall 102 of the holder 97. Thus, the sensor first 104 is configured to measure a distance from the bottom wall 102 of the holder 97 of a bottom surface (e.g., bottom minor

surface) of the stock lumber SL when the stock lumber is secured in the holder. The system 10 starts with the assumption that the stock lumber SL will contact the bottom wall 100 of the holder 97. In this instance, the first sensor 104 will register a zero or negligible distance of the bottom of the stock lumber SL from the sensor. The saw 94 will then be operated based on the stock lumber SL being in the assumed zero position. Thus, a height of the cut made by the saw 94 will be based on the stock lumber SL being in the assumed zero position. However, if the shape or position of the stock lumber SL in the holder 97 is such that the bottom of the stock lumber is spaced away from the bottom wall 102 of the holder, the first sensor 104 will measure this distance and the system 10 will account for this distance when instructing the saw 94 to cut the stock lumber. Therefore, a height of the cut made by the saw 94 will be adjusted by an amount related to the distance of the bottom of the stock lumber SL from the first sensor 104. For example, the truss manufacture computing device 12 may extrapolate from the distance detected by the first sensor 104 to determine the actual location of the portion of the stock lumber that is to be cut, and adjust the movement of the saw 94 accordingly. In the illustrated embodiment, the first sensor 104 is located on the bottom wall 100 of the holder 97. However, the sensor 104 could be located at other locations on the holder 97 or at other locations relative to the holder without departing from the scope of the disclosure. Additionally, a second sensor 105 may be provided on one of the side walls 102. In one embodiment, the second sensor 105 comprises a sensor to detect if the holder 97 is open or closed.

**[00109]** The saw 94 is operable to produce one or more lumber pieces LP from a single piece of stock lumber SL. This is accomplished by moving the saw 94 to make a first cut, or plurality of cuts, at a first location on the stock lumber SL, and moving the stock lumber with the holders 97 to position the saw for making a second cut, or plurality of cuts, at a second location on the stock lumber spaced a distance along the length of the stock lumber to form the cut lumber piece LP. This process is continued along a single piece of stock lumber SL until all the planned lumber pieces LP are made. In one embodiment, at least one lumber piece LP is formed from a single piece of stock lumber SL. In one embodiment, at least two lumber pieces LP are formed from a single piece of stock lumber SL. In one embodiment, three lumber pieces LP are formed from a single piece of stock lumber SL. In one embodiment, four lumber pieces LP are formed form a single piece of stock lumber SL.

**[00110]** The number of lumber pieces LP and the type (i.e., cuts) of lumber pieces is

determined by the truss manufacture computing device 12 to make the most efficient use of the stock lumber SL in the assembly of the planned wooden structures WS. In particular, the truss manufacture computing device 12 may optimize the pieces of lumber LP over a given number of lumber pieces to produce the most efficient output of lumber pieces for the planned wooden structure(s) WS. For example, a rolling optimization of lumber pieces LP to be cut may be determined by the truss manufacture computing device 12. In one embodiment, the truss manufacture computing device 12 may optimize the lumber pieces LP based on a forty-three (43) lumber piece count. Thus, the truss manufacture computing device 12 will plan out the next 43 lumber pieces to be cut and delivered to the buffer stations 18 for the planned wooden structure(s) on a rolling basis. This optimization of the cut lumber pieces LP cuts down on the waste and maximizes the number of lumber pieces that can be produced from the available stock lumber SL. For example, the system 10 is configured to determine the most efficient manner to cut the stock lumber SL to produce one or more wooden structures WS. For example, a single piece of stock lumber SL can be cut to produce lumber pieces LP for different wooden structures WS. Thus, the cut lumber pieces LP from a single piece of stock lumber SL may be delivered to one of the buffers stations 18 for the for assembly in two or more different wooden structures. Similarly, the cut lumber pieces LP could be delivered to different buffer stations 18 for the assembly of different wooden structures WS by the assembly stations 20. Additionally, the optimization of the cut lumber pieces LP may include a bias for the grade of the stock lumber SL used to produce the lumber pieces. As it is understood by those skilled it the art, higher grades of lumber are preferred. Therefore, the truss manufacture computing device 12 may assess the available grades of lumber within the inventory of stock lumber SL and optimize the use of the stock lumber having the higher grades. Optimization of the cut lumber pieces LP may also be done by selecting the shortest stock lumber SL possible for each group of lumber pieces in order to reduce waste. In one embodiment, the system 10 is able to produce less than 3% of waste from a given inventory of stock lumber SL. By comparison, conventional systems that are not configured to optimize lumber production produce nearly 10% of waste. Optimization may also be performed for the lumber pieces LP delivered to the auxiliary conveyor 112. In this instance, there is an unlimited piece number for optimizing the lumber sent to the auxiliary conveyor 112. A recipe routine for determining the order of assembly of the wooden structure is described below. It will be understood that in one embodiment, the optimization determination

is made after the recipe routine is completed.

**[00111]** Referring to FIG. 14, after the saw 94 has been operated to cut the stock lumber SL into the determined lumber pieces LP, the holders 97 are further configured to carry the lumber pieces out of the cabinet 88 for being transported downstream from the cabinet to a discharge area or station 106 of the cutting station 16. In one embodiment, the holders 97 deliver the lumber pieces LP to drive chains 107 which eject the lumber pieces from the cutting station and position the lumber pieces in registration with diverters 108. In the illustrated embodiment, each saw line 66A, 66B has a dedicated diverter 108. Each diverter 108 is moveable to engage a side surface (e.g., major side surface) of the lumber piece LP to push the lumber piece LP onto a series of rollers 110 which carry the lumber piece to an auxiliary conveyor 112. There are rollers 110 dedicated to each saw line. Thus, regardless of the in-feed line 66A, 66B on which the lumber pieces LP were produced, the respective rollers 110 will carry the lumber pieces to the auxiliary conveyor 112. In one embodiment, the rollers 110 dedicated to the first saw line 66A are configured to extend over the rollers dedicated to the second saw line 66B so that the lumber pieces LP from the first saw line are not obstructed from delivery to the auxiliary conveyor 112 by the lumber pieces from the second saw line. However, the rollers 110 could be arranged differently without departing form the scope of the disclosure.

**[00112]** The lumber pieces LP that are diverted onto the auxiliary conveyor 112 can be used for wooden structure assembly separate from the automated system 10 (e.g., manual wooden structure assembly), or those lumber pieces can be reintroduced into the stock lumber supply for later use in the automated assembly. An assembly plan constructed by the truss manufacture computing device 12 for the assembly of one or more wooden structures WS may instruct the operator to collect the lumber pieces LP on the auxiliary conveyor 112 and assign the lumber pieces to their intended destination. However, the lumber pieces LP which are planned for use in the automated assembly of the wooden structures are then picked up from the waiting area 106 by one of the buffer station manipulators 116 of the buffer station 18 and carried to the buffer station 18. One or more sensors 118 are located at the waiting area 106 and are configured to detect the passing of the lumber pieces LP out of the cutting station 16. The passing of a lumber pieces LP across the sensors 118 indicates to the system 10 that the lumber piece is located in the waiting area 106. The system 10 can then signal to the buffer station manipulator 116 to come and pick up the lumber piece and carry it to the buffer station 18. In

one embodiment, the sensors 118 are photo laser sensors.

**[00113]** Referring to FIGS. 15-21, the buffer station 18 comprises a buffer table 120 for receiving the cut lumber pieces LP and a manipulator assembly 122 for delivering the cut lumber pieces to the buffer table. The buffer station 18 configures the system 10 to arrange the cut lumber pieces LP into the proper order for being sequentially delivered to the assembly station 20. This is necessary because, as will be explained in greater detail below, the cutting station 16 optimizes the cuts made to the stock lumber SL to achieve the most efficient production of cut lumber pieces LP. However, these lumber pieces may not be cut in the order in which they will be assembled to make the final wooden structure(s) WS. Thus, the buffer station 18 takes the cut lumber pieces LP and properly arranges them for delivery to the assembly station 20 in the order in which they will be used in the assembly process. In the illustrated embodiment, the buffer station 18 includes two buffer tables 120 and two manipulator assemblies 122 (see, FIG. 1). Therefore, the cut lumber pieces LP can be picked up one of the buffer station manipulators 116 and delivered to either buffer table 120 for subsequent assembly at an assembly station 20. For simplicity, a single buffer table 120 and associated manipulator assembly 122 for delivering the lumber pieces LP to one of the assembly stations 20 will be described. It will be understood that the other buffer table and manipulator assembly 122 function similarly. Further, it will be understood that only a single buffer table 120 and manipulator assembly 122 may be included in the system 10. Additionally, more than two buffer tables 120 and manipulator assemblies 122 may be provided without departing from the scope of the disclosure.

**[00114]** Referring to FIGS. 15-18, the buffer table 120 comprises an index conveyor 124 having a plurality of support beams 126 extending along a width of the buffer table. The index conveyor 124 is actuatable to move the support beams 126 in a circuitous path. A drive 125 is operatively connected to the index conveyor 124 for advancing the conveyor in an incremental fashion. In the illustrated embodiment, the index conveyor 124 moves the support beams 126 in a clock-wise direction. A plurality of fences 128 extend along a length of the buffer table 120 and upward from the support beams 126. The fences 128 are spaced apart from each other along the width of the buffer table 120. Each fence 128 comprises a plurality of fence posts 130 (broadly, first partitions) spaced apart along a length of the fence. The fence posts 130 of each fence 128 are aligned along the length of the support surface 126. Slots 132 extending along the width of the support surface are formed between adjacent fences 128. The slots 132 are sized to

receive the cut lumber pieces LP delivered to the buffer table 120 as will be explained in greater detail below. In the illustrated embodiment, there are forty-three (43) slots 132 provided by the fences 128. However, a lesser or greater number of slots 132 may be provided. For example, in one embodiment, at least thirty-five (35) slots 132 are provided. It will be understood, that a sufficient number of slots 132 are provided on the buffer table 120 to accommodate the lumber pieces LP that have been produced at the cutting station 16 for use in the assembly of the planned wooden structure(s).

**[00115]** Conveyor panels 134 (broadly, second partitions) are also mounted on the support beams 126 of the index conveyor 124 and are disposed generally at a front side of the buffer table 120. The conveyor panels 134 extend along the width of the buffer table 120 and are spaced apart from each other along the length of the buffer table. Each conveyor panel 134 is aligned with a row of fence posts 130 extending along the width of the buffer table 120. Thus, the conveyor panels 134 also define a portion of the slots 132 formed by the fence posts 130 of adjacent fences 128. Accordingly, a cut piece of lumber LP having a certain length may be sized to extend between the conveyor panels 134 and the fence posts 130. However, as will be explained in greater detail below, the conveyor panels 134 are particularly configured to receive smaller lumber pieces LP. Rotation of the support beams 126 by the index conveyor 124 will in turn cause rotation the fences 128 and conveyor panels 134 causing the lumber pieces LP on the buffer table 120 to travel in an index fashion along the length of the buffer table.

**[00116]** Referring to FIG. 18, a plurality of bars 136 are disposed at a longitudinal end of the index conveyor 124 and are mounted within a perimeter of the index conveyor. The bars 136 are actuatable to move lengthwise through spaces between adjacent fence posts 130 of the index conveyor 124 and extend past the perimeter of the index conveyor. Actuation of the bars 136 is configured to push the lumber pieces LP off of the index conveyor 124 onto an assembly conveyor 140 disposed near the longitudinal end of the buffer table 120. The bars 136 are generally disposed below the fences 128 on the support surface 126 so that the bars are configured to engage the longer pieces of lumber LP. Similarly, a plurality of fingers 142 are also disposed at the same longitudinal end as the bars 136 and mounted within the perimeter of the index conveyor 124. The fingers 142 are actuatable to move through openings 144 in the index conveyor 124 to extend past the perimeter of the index conveyor. Actuation of the fingers 142 is also configured to push the lumber pieces LP off of the index conveyor 124 onto the

assembly conveyor 140. The fingers 142 are generally disposed below the conveyor panels 134 on the buffer table 120 so that the fingers are configured to engage the shorter pieces of lumber LP. The fingers 142 may also engage the longer piece of lumber LP when the lumber pieces extend into the spaces between the conveyor panels 134. The buffer table 120 may utilize other means for dispensing the lumber pieces LP from the buffer table. As such, the bars 136 and/or fingers 142 may be omitted without departing from the scope of the disclosure.

**[00117]** Referring to FIGS. 15 and 19-21, the manipulator assembly 122 is generally disposed above the buffer table 120. The manipulator assembly 122 is configured to transport the cut lumber pieces LP from the waiting area 106 to the buffer table 120. The manipulator assembly 122 comprises a support frame 146 and rails 148 on the support frame. A manipulator 116 is mounted to the support frame 146 and is movable along the rails 148 to translate the manipulator along the length of the buffer table 120. The manipulator 116 comprises a carriage 150 moveably attached to the rails 148 to translate the manipulator across the buffer station 18 and waiting area 106. The carriage 150 has a prime mover 152 and rollers 154 operatively connected to the prime mover for engaging the rails 148 to move the manipulator 116 along the support frame 146. A support 156 is moveably attached to the carriage 150. In the illustrated embodiment, the support 156 is moveable in a vertical direction with respect to the carriage 150 to raise and lower the support above the buffer table 120. Thus, in the raised position, the manipulator 116 is configured to translate across the buffer station 18, and in the lowered position, the manipulator is positioned to retrieve the lumber pieces LP from the waiting area 106 and place the lumber pieces on the buffer table 120. To that effect, a plurality of grip fingers 158 are attached to the support 156 and are moveable between open and closed positions for grasping the lumber pieces LP. Thus, the grip fingers 158 retain the lumber pieces LP to the manipulator 116 to carry the lumber pieces across the buffer station 18.

**[00118]** The grip fingers 158 on the manipulator 116 are arranged in groups that are spaced apart along a length of the support 156. A spacing of the grip finger groups is such that the grip fingers 158 are located at positions along the width of the buffer table 120 which position the grip fingers between the fences 128 and conveyor panels 134. Thus, the grip fingers 158 will not interfere with or contact the fences 128 or conveyor panels 134 when the manipulator 116 is moved to its lowered position. Each grip finger 158 comprises an elongate blunt blade member. The grip fingers 158 are pivotally attached to fins mounted on the support

156 such that when the grip fingers are actuated (as by extension or retraction of a cylinder), the grip fingers pivot towards each other to move to the closed position to grip the lumber pieces LP, and pivot away from each other to move to the open position to release the lumber pieces. The spacing between the grip fingers 158 in each group is such that the grip fingers are configured to engage side surfaces of the lumber pieces LP. Thus, the grip fingers 158 are configured to engage only the major side surfaces of the lumber pieces LP to grasp the lumber pieces. Additionally, the grip fingers 158 are configured such that they do not pierce or puncture the lumber pieces LP when engaged with the lumber pieces. Thus, the lumber pieces LP are not damaged or otherwise compromised during the handling by the manipulator 116.

**[00119]** Further, the grip finger 158 are divided into first grip fingers 158A and second grip fingers 158B (FIG. 21). The first grip fingers 158A are disposed along a majority of the length of the support 156. Thus, the first grip members 158A span a greater distance than the second grip members 158B. In the illustrated embodiment, the first grip fingers 158A extend along about 20% of the length of the support 156 and the second grip fingers 158B extend along about 80% of the length of the support. In one embodiment a length of the support is about 13 feet. However, it will be understood that a different ratio of extension of the grip fingers 158 may be utilized. Additionally, the second grip fingers 158B may have a different configuration than the first grip fingers 158A. For example, the second grip fingers 158B may be sized smaller than the first grip fingers 158A, and are located closer together than the first grip fingers. The second grip fingers 158 may also be mounted on a body 159 that is vertically movable with respect to the support 156. In a first, elevated position, bottoms of the second grip fingers 158B may be disposed above bottoms of the first grip fingers 158A. Thus, the second grip fingers 158B will be moved out of position for engaging a lumber piece LP in the waiting area 106. However, the body 159 is movable to a second, lower position such that the bottoms of the second grip fingers 158B are generally aligned with the bottoms of the first grip fingers 158A. This positions the second grip fingers 158B for grasping the lumber pieces LP. Therefore, when it is necessary, the second grip fingers 158 are movable to the second position for grasping a shorter lumber piece LP, or for grasping a longer lumber piece LP along with the first grip fingers 158A.

**[00120]** Sensors 160 may be provided on the manipulator 116 (e.g., support 156) to monitor the position of the manipulator (Fig. 20). In one embodiment, one or more first sensors

160A detect a vertical position of the support 156 to determine if the manipulator 116 is in a raised or lowered position. In the instance where the sensors 160A detect that the support 156 is in the lowered position, the grip fingers 158 can be actuated from an open position to a closed position to grasp lumber piece LP below the manipulator 116. One or more second sensors 160B may detect a position of the grip fingers 158 to confirm whether the grip fingers are in the open or closed position. Thus, when the manipulator 116 is detected to be in the lowered position with the grip fingers 158 in the closed position, the system 10 will register that the manipulator has grasped the lumber piece LP. For example, when the manipulator 116 is lowered to pick a lumber piece LP off the waiting area 106. Subsequently, when the manipulator 116 is detected to be in the lowered position by the first sensors 160A and the grip fingers 158 are detected to be in the open position by the second sensors 160B, the system 10 will register that the manipulator has dropped the lumber piece LP into the assigned slot 132 on the buffer table 120. Additionally, third sensors (not shown) may be provided on the ends of the grip fingers 158 to detect the presence of the lumber pieces LP once they have been grasped. One or more fourth sensors 160D may also be provided to detect a vertical position of the body 159 with respect to the support 156 to determine if the body holding the second grip fingers 158B is in a raised or lowered position. The buffer station 18 may verify the position of the lumber pieces LP by other means without departing from the scope of the disclosure.

**[00121]** Sensors 162 may be provided on the bars 136 and fingers 142 to detect the movement of the bars/fingers between the retracted and extended positions. The system 10 may use the signals form the sensor 162 indicating that the bars 136 and or fingers 142 have been extended to instruct the system that a lumber piece LP has been expelled onto the assembly conveyor 140. Thus, the sensors 162 provide a confirmation to the system 10 that the buffer table 120 has expelled the lumber piece LP at the end of the table permitting the buffer table to move the next lumber piece into position for being expelled from the table. Additionally, sensors 164 on the assembly conveyor 140 may be disposed at the opposite end of the assembly conveyor, adjacent the assembly station 20. First sensors 164A may be disposed along the conveyor 140 and configured to detect the passing of a lumber piece LP traveling along the assembly conveyor 140. Thus, the first sensors 164A may signal to the buffer station 18 to push the next lumber piece LP onto the assembly conveyor 140, and indicate to the assembly module 22 at the assembly station 20 that a lumber piece LP is being delivered for assembly. One or

more second sensors 164B may signal to the assembly module 22 that a lumber piece LP has been delivered. The second sensors 164B nay also be configured to detect warped lumber pieces LP.

**[00122]** Coordination of which slot 132 on the buffer table 120 to place a lumber piece LP is instructed by a recipe routine prepared by the truss manufacture computing device 12. The recipe provides a sequence of joint formation for the assembly of the wooden structure(s). As such an order of lumber pieces LP to be delivered to the assembly station 20 will be determined. Therefore, the lumber pieces LP will be placed in the slots 132 such that the buffer table 120 delivers the lumber pieces in the order prescribed by the recipe routine. The details of the recipe routine will be discussed in greater detail below.

**[00123]** Referring to FIGS. 1 and 22-28, the assembly station 20 is located adjacent an opposite end of the assembly conveyor 140 from the buffer station 18 and is configured to perform an assembly process for assembling one or more wooden structures WS planned by the truss manufacture computing device 12. The assembly station 20 comprises the assembly module 22 for arranging the lumber pieces LP into the planned wooden structures WS, and the plate distribution assembly 24 for supplying the fasteners (i.e., nailing plates P) to the assembly module for securing the lumber pieces together. In the illustrated embodiment, there are a pair of assembly stations 20, one for each buffer table 120. However, for simplicity, operation at only one of the assembly stations 20 will be discussed. Moreover, it will be understood that the system 10 may include only one assembly station. Alternatively, the system 10 may include more than two assembly stations 20 without departing from the scope of the disclosure.

**[00124]** The assembly station 20 is uniquely configured to assemble the wooden structures WS in a joint-by-joint extrusion sequence from start to finish until the entire wooden structure is formed. More particularly, this extrusion method requires that, during the assembly process, each joint of the wooden structure is completely formed prior to positioning and attaching two or more lumber pieces together at another joint. Thus, the joint-by-joint extrusion method is different from conventional lumber-by-lumber assembly methods which, for example, are concerned with assembling the structure, such as a truss, by forming an outer cord first and then assembling each lumber member within the web of the truss in a right-to-left or left-to-right fashion. Therefore, in these conventional manufacturing systems, the focus is not on sequential joint formation. However, in the current system 10, the joint-by-joint extrusion sequence

provides an assembly method whereby each joint is fully assembled prior to starting any other joint. As a result, the formation of a first joint is not affected by the prior formation or partial formation of a second joint that is started but not finished after initiating the formation of the first joint. However, it is also possible for the assembly station 20 to operate by partially forming one joint and then proceeding to another joint before completing formation of the one joint.

**[00125]** In conventional systems, assembling the wooden structures in the lumber-by-lumber method can lead to fit problems when returning back to a prior joint to complete the prior joint after having started or completed one or more other subsequent joints. Rather, in the current system 10, each joint is fully formed prior to attaching members of another joint together so that subsequent joint formation does not affect the fit and placement of a previously started joint. This results in a completed wooden structure WS where each joint has been able to be constructed by precisely positioning the lumber pieces LP at the joint for a secure joint connection without having the joint be altered by other partially formed joints. This allows for each joint to be precisely constructed ultimately creating a completed wooden structure WS where each joint has each lumber piece LP precisely placed contributing to a more structurally sound wooden structure WS over those assembled by conventional automated methods. It will be understood that a fully or completely formed joint is defined as a joint of the wooden structure WS that has every lumber piece LP associated with that joint positioned at the joint and secured together with a nailing plate pair. Thus, any joint which does not have all associated lumber pieces and both nailing plates attached cannot be said to be a fully or completely formed joint.

**[00126]** Referring to FIGS. 22 and 23, the assembly module 22 comprises a pair of lumber assembly tables 166 spaced apart from each other by a gap 168. In the illustrated embodiment, a first lumber assembly table 166A is positioned adjacent to the end of the assembly conveyor 140 where the cut pieces of lumber LP are delivered to the assembly module 22. A second assembly table 166B is disposed on an opposite side of the gap 168 from the first assembly table 166A. The first and second assembly tables 166A, 166B may have substantially the same configuration. In particular, each lumber assembly table 166 generally includes a large, flat support surface defined by one or more table panels. The lumber assembly tables 166 may be of generally any length and width to accommodate wooden structures (e.g., trusses) of generally any size. In one embodiment, the first and second assembly tables 166A, 166B together comprise an assembly table. A bar 169 (Fig. 23) is moveable into the gap 168 between

the assembly tables 166A, 166B to support lumber pieces LP that have a significant portion of their length disposed over the gap (e.g., extend generally parallel to the y-axis of the table) when the lumber piece is positioned on the assembly tables. In one embodiment, an assembly axis AA (e.g., x-axis) of the assembly module 22 extends along a length L of the assembly tables 166, a y-axis of the assembly module extends along a height H of the assembly tables (perpendicular to the x-axis), and a z-axis of the assembly module extends in a vertical direction from the assembly tables (perpendicular to both the x and y-axis).

**[00127]** Lower rails 170 are disposed in the gap 168 between the assembly tables 166, and upper rails 172 are disposed above the assembly tables. The upper rails 172 extend parallel to the lower rails 170 and are generally aligned with the lower rails. Thus, both rails 170, 172 extend orthogonal to the assembly axis AA (i.e., parallel to the y-axis). A lower platen assembly 174 is mounted on the lower rails 170, and an upper platen assembly 176 is mounted on the upper rails 172 (FIG. 22). The upper rail 172 is disposed on a bottom of a partition 177 supported by a support frame 179. The lower rail 170 is support at a bottom of the support frame 179. The lower platen assembly 174 is configured to receive a nailing plate P and apply the nailing plate to downwardly facing surfaces of the lumber pieces LP at joints of a wooden structure WS. The upper platen assembly 176 is similarly configured to receive a nailing plate P, but apply the nailing plate to upwardly facing surfaces of the lumber pieces LP at the joints of the wooden structure. Thus, the platen assemblies 174, 176 are movable along the rails 170, 172 to align themselves with each other such that the platen assemblies can apply a nailing plate pair to a joint of the wooden structure WS at opposite sides of the lumber pieces LP to attach two or more lumber pieces together. In the illustrated embodiment, the platen assemblies 174, 176 function only to apply the nailing plates P to the lumber pieces LP and do not perform any lumber handling functions. In one embodiment, the platen assemblies 174, 176 are programmed to press to a predetermined distance to ensure proper application of the nailing plates P. For example, the platen assemblies 174, 176 may press to a distance of about 1 ½ inches.

**[00128]** Referring to FIGS. 23-25, each platen assembly 174, 176 includes a carriage 178 (broadly, a base) movably mounted on the respective upper and lower rails 172, 170. A prime mover may be operatively connected to the carriage 178 to move the carriage along a respective one of the rails 170, 172. A linear actuator or press 180 is configured to move a platen 182 of the platen assemblies 174, 176, respectively, upwards or downwards relative to the

carriage 178 to press the nailing plate P into the lumber pieces LP. In one embodiment, the press 180 temporarily locks into place after being actuated for more effectively driving the nailing plates P into the lumber pieces LP. Each platen 182 is configured to grip and hold a single nailing plate P at a time. In the illustrated embodiment, the platens 182 are magnetized to grip the nailing plate P. In particular, each platen 182 includes a cover plate 184 defining an attachment surface 186 for holding a nailing plate P, and a holder 188 disposed below the cover plate. The holder 188 defines a plurality of receptacles 190 spread or dispersed across the holder. Each receptacle 190 is configured to receive a magnet 192 such that when magnets are received in the receptacles, the magnets together apply a uniformly dispersed magnetic field across the attachment surface 186 of the cover plate 184 to hold the nailing plate P on the platen 182. This arrangement provides benefits over platen assemblies that incorporate only a single magnet or magnets that are not uniformly disposed with respect to the attachment surface. In particular, incorporating a single magnet may cause the nailing plate to shift when it is being applied to the platen causing the nailing plate to be located slightly off-center on the platen. This will result in the nailing plate being applied off-set from a center of the joint in the wooden structure. However, in the current disclosure, the multiple magnets 192 that are uniformly dispersed to provide a consistent pulling force across the attachment surface 186 which will not tend to skew the nailing plate off-center when the nailing plate is applied to the attachment surface. In this way, the assembly module 22 can ensure that the nailing plates P are being applied in the center of the joint to ensure proper connection of all the lumber pieces LP at the joint. Other configurations of the platen assemblies 174, 176 are within the scope of the present disclosure.

**[00129]** Referring to FIGS. 1, 22 and 26-28, the assembly module 22 further comprises a robotic placement assembly 202 including a support frame 204 disposed over the assembly tables 166. The support frame 204 comprises a pair of supports 206 disposed on opposite sides of the assembly tables 166, respectively. Each support 206 comprises a pair of uprights 208 and a beam 210 extending between the uprights 208. A first rail 211 extends over the first assembly table 166A, and a second rail 212 extends over the second assembly table 166B. The first and second rails 211, 212 are supported by and extend between the beams 210 of the supports 206. A first robot 214 is moveably mounted on the first rail 211, and a second robot 216 is moveably mounted on the second rail 212. The robots 214, 216 function to perform all lumber handling

functions during the assembly process. Thus, the robots 214, 216 are configured to pick, place, and advance the lumber pieces LP in the assembly of the wooden structure WS. To this end, the assembly tables 166A, 166B do not incorporate any pucks or stops to position the lumber pieces LP. Rather, the positioning function is performed exclusively by the robots 214, 216. In one embodiment, the assembly module 22 includes no more than the two robots 214, 216 for positioning the lumber pieces LP.

**[00130]** Referring to FIG. 26, each robot 214, 216 includes a carriage 218 (broadly, a support) that is moveably mounted on the respective rail 210, 212. In the illustrated embodiment, each carriage 218 moves linearly along the rails 211, 212 in a direction generally parallel to the y-axis (FIG. 22). One or more prime movers 219, are operatively connected to the carriages 218 to move the carriages along the rails 210, 212. Each carriage 218 supports a robotic arm 220 moveably mounted on to the respective carriage 218. Each robotic arm 220 comprises a base 222 attached to the carriage 218. More specifically, the base 222 is rotatably mounted on the carriage 218. A plurality of arm members are moveably attached to the base 222. In the illustrated embodiment, a first arm member 224 is pivotably attached to the base 222 at a proximal end of the first arm member, and a second arm member 226 is pivotably attached to a distal end of the first arm member. Thus, each arm member 224, 226 is connected to an adjacent component of the robotic arm 220 by a joint. A tool 228 is mounted on a distal end the second arm member 226. The tool 228 may be rotatably mounted to the second arm member 226. Therefore, by moving the carriages 218 along the rails 210, 212, and articulating the robotic arm 220 by moving the base 222 and arm members 224, 226 at the joints, the robotic arms can be moved in three dimensions to locate the tool 228 at an array of positions above the assembly tables 166A, 166B, as well as to and from the conveyor assembly 140, as will be discussed in greater detail below.

**[00131]** Referring to FIGS. 26-28, the tool 228 is a multi-functional tool for locating, positioning, and holding the lumber pieces LP on the assembly tables 166A, 166B. The tool 228 comprises an elongate body 230 having opposite first and second end margins. A gripper 232 is located at the first end margin and is configured to grasp the lumber pieces LP. In one embodiment, the gripper 232 comprises clamps 234 that are movable relative to each other between open and closed positions. In the open position, the gripper 232 is able to be positioned over a lumber piece LP, and in the closed position, the gripper is configured to engage side

surfaces (e.g., minor side surfaces) of the lumber pieces to secure the lumber piece to the tool 228. A suction device 236 is located at the second end margin of the elongate body 230 and is also configured to secure the lumber pieces LP to the tool 228. The suction device 236 comprises a negative pressure source (not shown) and suction pad 238 in communication with the negative pressure source. The suction pad 238 is engageable with a surface (e.g., major side surface) of the lumber piece LP, and the negative pressure source is actuatable to create suction at the suction pad to hold the lumber piece to the suction pad. In one embodiment, the gripper 232 and suction pad 238 are movably mounted on the elongate body 230 such that the gripper and pad may float (i.e., move toward and away from) the body. For example, air cushions 231 and bearings 233 may provide a suspension for a mount 235 on which the gripper 232 and pad 238 are disposed. Therefore, when the tool 228 comes into contact with a lumber piece LP any slight misalignment can be accounted for by the flexibility in movement of the mount 235, and thus the gripper 232 and pad 238 on the mount, relative to the elongate body 230.

**[00132]** The gripper 232 may have particular application in grasping a majority of the lumber pieces LP used in the construction of the wooden structures WS. The determination for which implement, the gripper 232 or suction device 236, that is used to grasp the lumber piece LP is based on the amount of room provided by the previously placed lumber pieces at the joint. As such, in most cases the gripper 232 may be utilized as the area around the lumber piece LP to be located at a joint is sufficient to provide clearance for the gripper 232 to open and close. For example, an area of at least 2 inch on either side of the lumber piece LP is required to pick and place the lumber piece with the gripper. However, if the area around the lumber piece LP to be located at the joint is not sufficient to provide clearance for the gripper 232, then the suction device 236 may be utilized when handling the lumber piece. For example, the area on either side of the lumber piece LP to be placed at the joint is less than 2 inches, then the suction device 236 may be utilized to handle the lumber piece. As explained above, the tool 228 is rotatably mounted on the second arm member 226 of the robotic arm 220. Thus, the tool 228 can be rotated to locate either the gripper 232 or suction device 236 in registration with the lumber piece LP to orient the tool for handling the lumber piece. Additionally, the tool 228 is configured to grasp lumber pieces LP of varying sizes. In particular, the tool 228 can grasp lumber pieces LP having a length of less than about 2 feet. In one embodiment, the tool 228 can grasp lumber pieces LP having a length of less than about 7 inches. Therefore, the robots 214, 216 are able to

handle lumber pieces LP of any size used in the assembly of a wooden structure WS. This eliminates the need for operator involvement on specialty-type wooden structures. It will be understood that the tool 228 could have other means for gasping and holding the lumber pieces LP without departing from the scope of the disclosure. For example, one or both of the gripper 232 and suction device 236 can be removed and/or replaced. In the illustrated embodiment, the tool 228 is free of any attachment devices for securing two or more lumber pieces LP together. Thus, in this embodiment, the tool 228 cannot be operated to attach the lumber pieces LP together. Rather, the tool 228 functions only to handle the lumber pieces LP.

**[00133]** Referring to FIGS. 27-29, a vision system 240 may also be mounted on the tool 228 and operable to assist the robot 214, 216 in locating a lumber pieces LP when the tool is positioned over the lumber piece. The vision system 240 may comprise a camera 242 configured to acquire images within a focal zone 243 of the camera, and an illumination source (e.g., light) 244 configured to illuminate the focal zone of the camera (Fig. 29). In one embodiment, the illumination source 244 comprise a red light. However, other illumination sources may be used without departing from the scope of the disclosure. As will be described in greater detail below, the vision system 240 is configured to view the fiducials 86A on the lumber pieces LP and communicate the viewed fiducials to the truss manufacture computing device 12 to instruct the robot 214, 216 on where to position the lumber piece at a particular joint of the wooden structure WS.

**[00134]** Referring to FIGS. 30-38, the plate distribution assembly 24 is configured for supplying and distributing nailing plates P to the assembly module 22. The plate distribution assembly 24 comprises a plate distributor unit 246 for storing and preparing the nailing plates P, and a plate selector 248 for picking the nailing plates and placing them on one of the platen assemblies 174, 176 of the assembly module 22. The plate distribution assembly 24 is configured to handle both nailing plate pairs PP and individual nailing plates P as will be discussed below. As a result, the plate distribution assembly 24 is able to automatically retrieve and apply a complete inventory of the potential nailing plates P needed to construct any wooden structure WS planned by the truss manufacture computing device 12. Additionally, the plate distribution assembly 24 provides for enhanced handling of the nailing plates P to ensure that the nailing plates can be repeatedly and accurately supplied to the assembly module 22. In particular, the plate distribution unit 246 is configured to separate nailing plate pairs PP, and the

plate selector 248 is configured to grasp the separated nailing plate pairs by the teeth, without damaging or altering the teeth, so that the flat back surfaces of the nailing plates P can be easily applied to the platen assemblies 174, 176, as will be discussed in greater detail below.

**[00135]** The plate distributor unit 246 comprises a magazine rack 250 including a plurality of magazine slots 252 for receiving stacks of nailing plate pairs PP. The magazine rack 250 may be comprised of multiple removable cassettes 254. As will be understood, the nailing plates are loaded into each cassette 254 in pairs, with the teeth sides facing and overlapping one another. Each cassette 254 may define a plurality of magazine slots 252 for receiving dedicated nailing plate pairs PP (i.e., designated sizes of nailing plate pairs). A platform 256 adjacent the magazine rack 250 supports the magazine rack and may be configured to allow an operator to load nailing plates onto the rack while the assembly module 22 is operating. Thus, an inventory of nailing plates can be dynamically selected and adjusted based on the cassettes 254 used in the magazine rack 250. Additionally, by using multiple cassettes 254, a portion of the available cassettes can be changed without having to take the entire magazine rack 250 offline. Therefore, the assembly process need not be halted during the changing of the nailing plates. In one embodiment, each cassette 254 is configured to hold a single type/size of nailing plate pair PP. Each nailing plate pair PP may be configured such that the teeth of the nailing plates are oriented face-to-face and engaged with each other configuring the nailing plates in a teeth-to-teeth orientation. In one embodiment, each cassette 254 may be configured to hold different types/sizes of nailing plate pairs. It is understood that several different sizes of nailing plates P can be used to join the pieces of lumber LP together to form the wooden structure(s) WS. In the illustrated embodiment, each cassette 254 includes at least four (4) magazine slots 252. In the illustrated embodiment, the magazine rack 250 includes eight (8) cassettes 254. However, the magazine rack 250 may include another number of cassettes or may not include any removable cassettes without departing from the scope of the disclosure. For example, each cassette 254 could include only a single magazine slot 252.

**[00136]** Referring to FIGS. 32-38, a first plate handling assembly 258 is disposed adjacent a bottom of the magazine rack 250 generally at a back side of the magazine rack. The first plate handing assembly 258 is configured to retrieve the nailing plate pairs PP from the magazine rack 250 and separate and position the nailing plate pairs for selection by the plate selector 248. The first plate handling assembly 258 comprises a retriever 260 for retrieving

nailing plate pairs PP from the magazine rack 250, and a separator 262 for separating the retrieved nailing plate pairs. The retriever 260 comprises a shuttle 264 movable along a track 266 below the magazine rack 250. A grabber 265 is movably mounted on the shuttle 264 and movable to a position generally under the magazine rack 250 to grab a nailing plate pair PP from one of the slots 252. After the grabber 265 retrieves the nailing plate pair PP from one of the magazine slots 252 (e.g., at the bottom thereof), the grabber performs an initial separation of the nailing plates to disengage the teeth of the nailing plates from each other. In particular, the grabber 265 delivers the nailing plate pair PP in registration with a pair of first magnets 267 such that the magnets are disposed above and below the nailing plate pair PP. The first magnets 267 apply a magnetic force on the nailing plate pair PP sufficient to disengage the teeth of the nailing plates from each other to separate the nailing plates and retain the nailing plates to the respective magnets. In one embodiment, the grabber 265 separates the nailing plate PP by about 1 inch. The grabber then transports the pair of nailing plates to the separator 262. Thus, the retriever 260 is configured to grab, separate, and transport every size/type of nailing plate pair PP contained within the magazine rack 250.

**[00137]** Referring to FIG. 36, the separator 262 comprises a pair of second magnets 268 moveable vertically relative to each other and configured to further separate the nailing plates P, a slider 270 moveable horizontally relative to the magnets, and a pusher 272 mounted on the slider and configured to push the nailing plates along slots 271 to position the nailing plates for being separated by the second magnets. The grabber 265 carries the nailing plate pair PP to a location between the magnets 268 so that activation of a magnetic field across the magnets will cause the final separation of the nailing plate pair to locate the nailing plates P for being retrieved by the plate selector 248. In particular, the grabber 265 locates the previously separated nailing plates P in respective slots 271. The slider 270 is actuatable to engage the pusher 272 with the nailing plates P to move the nailing plates in the slots 271 to position the nailing plates against a backstop 273. Thus, the pusher 272 is able to horizontally align the separated nailing plates P so that they are generally centered on a common vertical axis and in registration with the second magnets 268. Sensors 274 may be provided to verify that the pusher 272 has been actuated. In particular, the sensors 274 may comprise proximity sensors configured to monitor the movement of the pusher 272. Thus, actuation of the pusher 272 indicates to the system 10 that the nailing plates have been aligned. Once the separated nailing plates P are aligned, the magnets 268 are

operable to further separate the nailing plates (FIG. 36). Thus, the first magnets 267 are configured to perform a first separation to separate the nailing plate pair PP a first distance, and the second magnets 268 are configured to perform a second separation to separate the nailing plates P a second distance that is greater than the first distance. This further separation locates the nailing plates P at known vertical positions for reference by the selector 248 and provides sufficient clearance for the selector to grab the nailing plates P and take them to the platen assemblies 174, 176. Additionally, the rate the second magnets 268 are separated can be adjusted to account for the size and weight of the nailing plates P.

**[00138]** Referring to FIGS. 37 and 38, a plate conveyor 276 may be located adjacent the magazine rack 250 and is configured to receive individual nailing plates P. In the illustrated embodiment, the plate conveyor 276 comprises an indexing conveyor including a plurality of parallel rows of blades 278 movable with chains 279 of the index conveyor in a circuit. The blades 278 in each row are aligned with the blades in the other rows such that the chains 279 move the blades around the index conveyor 276 in unison. The individual nailing plates P are received within the spaces between the blades 278 such that each space between adjacent blades is configured to receive a single nailing plate P. A second plate handling assembly 280 is disposed adjacent an end of the plate conveyor 276. The second plate handing assembly 280 is configured to receive the individual nailing plates P from the plate conveyor and position the nailing plates for selection by the plate selector 248. The second plate handling assembly 280 comprises a receiver 282 for receiving a nailing plate P from the plate conveyor 276, and a plurality of sensors 284 for measuring the received nailing plate P to verify that the correct nailing plate has been dispensed. In particular, the sensors 284 are configured to detect a length and width of the nailing plates P to cross-reference the measure dimensions with the intended nailing plate. In the illustrated embodiment, the receiver 282 comprises a picker 283 for picking the nailing plates P off the plate conveyor 276, and a chute 285 located at an end of the plate conveyor configured to receive the nailing plates from the picker as they are dispensed from the plate conveyor. The chute 285 carries the nailing plates P from the plate conveyor 276 to a stop position where they are measured by the sensor 284. In the illustrated embodiment, the chute 285 carries the nailing plates P vertically downward. The sensors 284 are configured to measure the length and width of nailing plates P as they travel down the chute 285. The length and width dimensions are compared to the dimensions of the planed nailing plate P to confirm that the

correct nailing plate has been dispensed Once the plate measurement is verified, the selector 248 is permitted to retrieve the nailing plate P from the chute 285.

**[00139]** Referring to FIGS. 32, 33, and 38, the selector 248 comprises a robotic arm. The robotic arm 248 includes a base 286 (broadly, a support) moveably mounted to a support 288. For example, the base 286 may be rotatably mounted to the support 288. A plurality of arm members are moveably attached to the base 286. In the illustrated embodiment, a first arm member 290 is pivotably attached to the base 286 at a proximal end of the first arm member, and a second arm member 292 is pivotably attached to a distal end of the first arm member. Thus, each arm member 290, 292 is connected to an adjacent component of the robotic arm 248 by a joint. A tool 294 is mounted on a distal end the second arm member 292. The tool 294 may be rotatably mounted to the second arm member 292. Therefore, by rotating the robotic arm 248 about the support 288, articulating the robotic arm through movement of the arm members 290,292 at the joints and, rotating the tool 294 about the second arm member, the robotic arm 248 can be operated to transfer the nailing plates P from the distributor assembly 24 to the platen assemblies 174, 176 of the assembly module 22.

**[00140]** The tool 294 includes a gripper 296 operable to grip the nailing plates P by grabbing the teeth of the nailing plates (FIGS. 38 and 39). The selector 248 is moveable to position the tool 294 for grabbing the nailing plates P on the first plate handling assembly 258 and the nailing plates on the second plate handling assembly 280. Once the tool 294 has been operated to grasp a nailing plate P, the robotic arm 248 can be rotated about the support 288 to locate the tool over/under one of the platen assemblies 174, 176 to place the nailing plate on the platen assembly (FIG. 39). The robotic arm 248 is operatively connected to the platen assemblies 174, 176, such that the robotic arm is configured to place the nailing plates P at a center of the platens 182 of the platen assemblies. Additionally, the robotic arm 248 is configured to place the nailing plates P at a desired rotational orientation at the center of the platens 182. Thus, the robotic arm 248 may place the nailing plate P on the platen 182 in the orientation in which it will be driven into the lumber pieces LP. So the platens 182 do not have to rotate or change their orientation to properly locate the nailing plate on the lumber pieces LP. Instead the robotic arm 248 performs the angle placement allowing the functionalities of the platen assemblies 174, 176 to be simplified. Moreover, as previously mentioned, the platen assemblies 174, 176 are configured to apply a uniformly dispersed magnetic field across the

attachment surface 186 of the platen 182 to hold the nailing plate P on the attachment surface. Therefore, the nailing plates P can be repeatedly and precisely delivered to a center location of the platens 182 so that the assembly module 22 can reliably apply the nailing plates at a desired location (i.e., at a center of a joint) on the wooden structure WS. Additionally, a pair of holding plates 298 are provided for holding nailing plates P that have been retrieved by one of the plate handling assemblies 258, 280 but are not ready to be delivered to the plate assemblies 174, 176. In one embodiment, the system 10 places the nailing plates P on the holding plates 298 in the rare instances where the assembly of a particular joint is started (i.e., bottom plate is attached but not top plate) but not finished before another joint is started.

**[00141]** In operation, the assembly station 20 is capable of automatically assembling a wooden structure WS including a plurality of lumber pieces LP connected together at joints. As previously mentioned, the assembly module 22 connects the lumber pieces LP together at a plurality of joints using the nailing plates P to construct the wooden structures WS. The assembly process at the assembly station 20 begins when the first lumber piece LP is delivered to the assembly station 20 by the assembly conveyor 140 (FIG. 40). The first robot 214 is then moved over to the lumber piece LP to position the vision system 240 on the tool 228 over a fiducial 86A on the lumber piece. The system 10 will move to locate the camera 242 directly over the fiducial 86A such that the fiducial is in the center of the focal zone of the camera. The vision system 240 will then take a picture of the fiducial 86A using the camera 242 and communicate the fiducial to the truss manufacture computing device 12 (FIG. 29). Due to innate system tolerances, the camera 242 may not be located directly over the fiducial 86A. Therefore, a deviation between the actual position of the camera 242 (i.e., center of focal zone) and the location of the fiducial 86A will be recorded. This deviation will be used to instruct the first robot 214 on where it is actually located in order to properly locate the first lumber piece LP. In particular, the system 10 will determine a reference point RP (i.e., center of platens 182) to serve as the joint location and the positioning of the lumber pieces LP will be based off the reference point (FIG. 22). In one embodiment, the fiducial 86A is printed at a known distance from a joint end of the lumber piece LP. Thus, the first robot 214 uses the position of the camera 242, as determined by the deviation between the center of the focal zone and the fiducial 86A, as a second reference to determine the end of the lumber piece so that the end of the lumber piece can be placed at the reference point RP corresponding to the joint location.

**[00142]** Therefore, the position of the fiducial 86A within the focal zone 243 of the camera 242 is accounted for to ensure that the first robot 214 accurately places the lumber piece LP at the joint (FIG. 29). For instance, the system 10 is calibrated such that the robot 214 is moved to center the fiducial 86A within the focal zone 243 of the camera 242. In this ideal position, the first robot 214 will position the lumber piece LP at the joint (i.e., reference point RP) without any adjustments. However, if the tool 228 is positioned over the fiducial 86A such that the fiducial is off-center in the focal zone 243, the system 10 will account for the distance that the fiducial is off-set from the center of the focal zone so that the lumber piece LP can still be accurately placed at the joint. The first robot 214 will then position the lumber piece LP at the joint at a predetermined position using the fiducial 86A (FIG. 41) to compensate for the offset of where the tool 228 actually grabs the lumber piece LP from an ideal or "zero" location. As will be discussed in greater detail below, the system 10 will have a predetermined sequence of lumber pieces LP to be positioned to form a predetermined sequence of joints. The first and second robots 214, 216 and upper and lower platen assemblies 174, 176 will work together to complete the wooden structure assembly. Therefore, the assembly module 22 is configured to performed a completely automated assembly of the wooden structure WS.

**[00143]** If when positioning the tool 228 over the fiducial 86A, the camera 242 does not see the fiducial within the focal zone 243, the first robot 214 will reposition the tool 228. If the camera 242 does not see the fiducial 86A after two iterations of the first robot 214 repositioning the tool 228, then the system 10 will signal an error halting the assembly process and prompting the operator to come and address the issue. For instances where the suction pad 238 is used to handle a shorter lumber piece LP, the first robot 214 will take a picture of the fiducial 86A with the camera 242 and then the tool 228 will rotate 180 degrees to position the suction pad 238 for securing the lumber piece to the suction pad. If the fiducial 86A was seen in the center of the focal zone 243 of the camera 242, then the lumber piece LP will be centered on the suction pad 238 once the tool 228 is rotated. Thus, the first robot 214 can be assured that the lumber piece LP is properly positioned on the suction bad 238.

**[00144]** Continuing on with the assembly process, once the first robot 214 has placed the first lumber piece LP at the first joint (i.e., reference point RP), the lower platen assembly 174 is moved along the lower rails 170 to position the lower platen assembly at the first joint below the first lumber piece LP (FIG. 41). The selector 248 will have already placed the appropriate

nailing plate P on the lower platen assembly 174. Thus, the lower platen assembly 174 will have previously moved within the gap 168 along the lower rails 170 to a position at the bottom of the assembly tables 166 adjacent the plate distribution assembly 24 so that the selector 248 can place the nailing plate P on the lower platen assembly 174. The lower platen assembly 174 will then be actuated to drive the nailing plate P into the lower major surface of the first lumber piece LP. The lower platen assembly 174 is centered on the reference point RP to ensure that the nailing plate P is applied at a center of the joint.

**[00145]** Once the nailing plate P has been driven into the first lumber piece LP, the first robot 214 will release its grasp of the first lumber piece and move back to the assembly conveyor 140 to retrieve a second lumber piece LP (FIG. 42). The second robot 216 will then move over to the first lumber piece LP and hold the first lumber piece at its predetermined position at the first joint (FIG. 43). The second robot 216 may perform a similar routine using the vision system 240 on the second robot and the fiducial on the first lumber piece LP so that the second robot knows precisely where it has grasped the first lumber piece LP. While holding the first lumber piece LP, the second robot 216 may pick up the lumber piece and replace the lumber piece at the joint confirming that the first lumber piece is properly positioned at the joint. The first robot 214 will then pick up and carry the second lumber piece LP over to the first joint. The first robot 214 will again utilize the vision system 240 to determine the placement of the second lumber piece LP at the first joint. The first robot 214 will then hold the second lumber piece LP at the first joint. After the first robot 214 has placed the second lumber piece LP at the first joint, the upper platen assembly 176 is moved along the upper rails 172 to position the upper platen assembly at the joint between the first and second lumber pieces LP. The upper platen assembly 176 will then be actuated to drive the nailing plate P into the lower major surface of the second lumber piece LP connecting the first lumber piece to the second lumber piece.

**[00146]** If the first and second lumber pieces LP are the only two lumber pieces included in the first joint, then the upper platen assembly 176 will also be operated to drive a nailing plate P into the upper major side surfaces of the lumber pieces LP to complete the joint. The selector 248 will have already placed a nailing plate P on the upper platen assembly 176. Thus, the upper platen assembly 176 will have previously moved along the upper rails 172 to a position at a bottom of the assembly tables 166 adjacent the plate distribution assembly 24 so that the selector 248 can place the nailing plate P on the upper platen assembly 176. The upper platen assembly

176 will then be actuated to drive the nailing plate P into the upper major surfaces of the lumber pieces LP along with also driving the bottom nailing plate into the lower major side surface of the second lumber piece. The upper platen assembly 176 will be centered on the reference point RP to ensure that the nailing plate P is applied at a center of the joint.

**[00147]** However, if the first and second lumber pieces LP are not the only two lumber pieces in the first joint, then the first robot 214 will move back over to the assembly conveyor 140 to retrieve a third lumber piece for placement at the joint (FIG. 44). The third lumber piece LP will be picked up, placed, and held at the first joint as was previously described for the second lumber piece. The second robot 216 may release its grip on the first lumber piece and reposition itself to hold the second lumber piece to clear room for the first robot 214 to position the third lumber piece at the first joint. If the third piece of lumber LP is the final lumber piece at the first joint, then the upper platen assembly 176 is moved over into alignment, or will remain in alignment, with the joint (i.e., at the reference point RP) and the upper platen assembly is actuated to both drive a nailing plate on the upper platen assembly into the upper major side surfaces of the lumber pieces and drive the bottom nailing plate into the bottom surface of the third lumber piece to complete the joint. If the third lumber piece LP is not the final lumber piece at the first joint, then the upper platen assembly 176 is actuated without a nailing plate P thereon so that the actuation of the upper platen assembly functions only to drive the bottom nailing plate into the lower major side surface of the third lumber piece. It will be understood that the robots 214, 216 and platen assembly 174, 176 may continue in this fashion until the first joint is completed.

**[00148]** Once the first joint is completed, the second robot 216 will then grasp one of the lumber pieces LP to pull the partially formed wooden structure WS along the assembly tables 166 to position the structure for the continued joint-by-joint assembly of the wooden structure until the entire wooden structure is complete. In order to ensure that the wooden structure WS is moved to the desired location, the second robot 216 may be operated to locate the tool 228 over a fiducial 82A of one of the lumber pieces LP to instruct the robot of its location on the lumber piece. In this instance, the tool 228 can be operate the camera 242 as previously described to determine the actual location of the second robot 216. With this information, the second robot 216 can be ensured to accurately advance the wooden structure WS along the assembly tables 166. The next joints of the wooden structure will be formed and completed in sequence using

the unattached ends and/or sides of the attached lumber pieces LP until the full wooden structure is assembled. The completed wooden structure WS can be further pulled or advanced by the second robot 216 to transport the wooden structure away from the assembly tables 166 to clear space for the assembly of the next wooden structure (FIG. 45). The second robot 216 may carry the now fully assembled wooden structure WS to a location where the structure can be picked up and loaded onto a vehicle, such as a truck, and transported to the construction site.

**[00149]** Figures 46-53 show a schematic illustration of lumber pieces being attached in the joint-by-joint extrusion sequence of the present disclosure. The joints are numbered from the first joint formed, J1, to the last joint formed J8. As previously described, initiation of joint formation occurs when at least one nailing plate is driven into a lumber piece at a joint, and the completion of the joint occurs when the nailing plate pairs are driven into the upper and lower major surfaces of the lumber pieces. As can be seen, once a joint has been initiated, the joint is completed prior to initiating the next joint. It will be understood that the order of the joint formation for the wooden structure WS could different from the illustrated order. In particular, the following disclosure provides a framework by which the system 10 may determine the joint order and lumber sequence for forming the wooden structure WS.

**[00150]** Referring to FIG. 54, a recipe routine may be programmed into the truss manufacture computing device 12 to produce a complete sequence of actions for producing a planned wooden structure WS. Initially, to determine the recipe for assembling a particular wooden structure WS, a diagram of the wooden structure may be uploaded into the system 10 at 300. The diagram can provide the system 10 with a complete picture of the geometric construction of the wooden structure WS (e.g., truss) including all of the lumber pieces LP and nailing plates P used to produce the wooden structure. This includes the overall size, shape, and construction of the wooden structure WS, the length and configuration (i.e., end cuts) of each lumber piece LP, and the size and type of nailing plates P used for each joint. At 302 the system 10 will produce a list of the joints within the wooden structure WS along with the lumber pieces LP and nailing plates associated with each joint. Armed with the joint list, the routine then determines the most optimal sequence to produce the wooden structure WS at 304. The sequence will include the order of joints to complete, the sequence for placing the lumber pieces LP at the joints, the sequence for moving the robots 214, 216 to place the lumber pieces at the joints, the movement of the robot 248 for placing the nailing plates P on the platen assemblies

174, 176, and the movement of the platen assemblies to locate the nailing plates at the joints. In one embodiment, the recipe first determines the order of the joints to be assembled at 306, next the placement of the nailing plates P is determined at 308, and then finally the recipe determines the movements of the robots 214, 216, 248 and platen assemblies 174, 176 to complete the assembly at 310.

**[00151]** More particularly, when the recipe is determining the order of the joints at 306, the system 10 looks at the first few feet (e.g., 2-3 feet) of the wooden structure WS, either from a left side or a right side, and determines which joint in the first few feet to start the assembly of the wooden structure. In one embodiment, the recipe looks for the joint that includes the longest lumber pieces LP to start the assembly of the wooden structure WS. The following joint order may be determined based on a number of factors. In one embodiment, the system 10 selects the next joint based on the number of lumber pieces LP that will already be in place from the assembly of the previous joint(s). Thus, the joint with the most number of lumber pieces LP will be selected for the next joint to be completed. This analysis may continue until the entire joint order is determined.

**[00152]** Efficiency of robot movement is considered during the recipe routine at 310. Thus, the system 10 may also determine the order of the joints and/or the order of lumber piece placement within a joint based on the movement paths necessary to complete the joints. Therefore, in one embodiment, the joint order that facilitates the most efficient movement path for the robots 214, 216, 248 and platen assemblies 174, 176 will be prioritized. An efficient movement path may be defined by cooperative movements between the robots 214, 216 and platens where movement of one robot 214, 216 does not interfere or cross the path of the movement of another robot. Similarly, an efficient movement path may be defined by movement of the platen assemblies 174, 176 that coordinates with the movement of the robots and limits the distance the platens travel through the course of the joint formations. The detailed movements of the robots 214, 216, 248 and platen assemblies 174, 176 are also considered when determining the sequence of joint formation. For example, longer lumber pieces LP may require one of the robots to grab and move the lumber piece multiple times to position the wooden structure in alignment with the platen assemblies 174, 176 for completing the next joint. Additionally, larger nailing plates P may require multiple pressing actions by a single platen assembly 174, 176. These special requirements may be considered and prioritized in reference to other movements

of the robots 214, 216 and platen assemblies 174, 176 to determine the most efficient sequence of events. Over time specific assembly recipes can be established for specific wooden structures so that the system 10 can store the assembly recipes for use in future assembly processes. Moreover, as efficiencies improve, the recipes can be updated to ensure that the most optimal sequences are being used for assembly.

**[00153]** During the determination of the optimal assembly sequence at 304 and the joint order at 306, the recipe routine may also implement a point system to the various movements of the robots 214, 216 and platen assemblies 174, 176. The truss manufacture computing device 12 may operate the system 10 whereby the sequence that produces the highest point total, or a higher point total in comparison with another sequence, determines the order of the component movements. For example, various conditions/actions may be given a number from zero (0) to four (4) to assign a priority weight to the condition/action. The truss manufacture computing device 12 will then run a number of joint and movement sequence tests, and the sequence which produces the highest point total may be selected for the assembly of the wooden structure WS. The conditions/actions for which numbers may be associated are directed to the construction of the joints and the lumber pieces LP in the joints. In one embodiment, consideration is given to whether the wooden structure WS will need to be moved backwards (i.e., opposite an assembly direction) by greater than a threshold amount. This action may be assigned a value of zero (0) points as this is a less efficient operation. Therefore, sequences with this operation may be deprioritized. In one embodiment, consideration is given to joints that contain only two (2) lumber pieces extending along the same axis. In other words, lumber piece splices will be considered in the order of movements. This action may be assigned a value of 0.25 points. In one embodiment, consideration is given to joints with the most lumber pieces LP already in place from previously formed joints. In other words, consideration will be given to joints that can be formed from the unattached sides and free ends of the lumber pieces that were previously placed during the completion of a previous joint. This condition may be assigned a value of one (1) point. In one embodiment, consideration is given to the position of the joint in reference to the leading end of the wooden structure WS. Therefore, priority may be given to the joints that are closest to the starting point of the wooden structure WS. This will tend to result in a sequence where the joints are completed in a substantially left to right or right to left fashion. This condition may be assigned a value of two (2) points. In one embodiment, consideration is given

to the amount of lumber pieces in a joint. Thus, the next joint with the most lumber pieces LP may be given priority. This particular condition may be assigned a value of three (3) points. Finally, consideration may be given for the length of the lumber pieces LP. Thus, longer lumber pieces LP may be given priority over shorter lumber pieces. So a joint including one or more longer lumber pieces LP may be given priority over a joint using one or more shorter lumber pieces. The longer lumber piece joints may be assigned a value of four (4) points. It will be understood that these conditions are not exhaustive of every possible condition by which the system 10 may determine the order of joint assembly. However, in view of this point system, the truss manufacture computing device 12 can compute multiple different sequences totaling the points assigned to the conditions/actions of a particular sequence. In one embodiment, the sequence which produces the highest point total will ultimately be the sequence that is selected to assemble the wooden structure WS.

**[00154]** The truss manufacture computing device 12 may also be programmed to perform a calibration routine to calibrate the assembly module 22 to provide confirmation that the robots 214, 216 are being moved along their intended paths to their intended locations. The precise movements of the robots 214, 216 allows for the precise positioning of the lumber pieces LP in the assembly of the wooden structures WS. However, because the robots 214, 216 do not operate from fixed locations, but instead are movable along the rails 210, 212 of the support frame 204, there is more opportunity for the movement of the robots to deviate from their intended path. Therefore, calibrating the assembly module 22 equips the system 10 with the ability to precisely assemble the wooden structures WS using movable robots 214, 216. This distinguishes from conventional manufacturing systems where any calibration that is performed must be done on stationary robots.

**[00155]** Referring to FIG. 55, a robot calibration routine is configured to calibrate the movement of the robots 214, 216 in space. To begin the calibration routine, the first robot 214 may be calibrated at 400. First, an angle of the rail 210 on which the first robot 214 moves is determined at 402. Rather than assume the rail 210 extends perfectly horizontal, the actual axis of the rail is determined. The angle of the rail 210 is determined by moving the carriage 218 of the first robot 214 along the rail and measuring the position of the carriage (i.e., height above the assembly table 166) at least at two separate locations. With the two measured locations, an axis of the rail 210 can be determined. This allows for the calibration routine to accurately locate the

base of first robot 214 as it moves along the rail. In particular, the height of carriage 218 of the robot 214 above the assembly table 166 (z-axis) and position of the carriage along a transverse dimension of the assembly table (y-axis, dimension extending between top and bottom of table) can be mapped using the axis of the rail 210. Next, the first robot 214 is oriented such that the tool 228 is disposed directly underneath (i.e., vertically aligned) with the carriage 218 at 404. The tool 228 is also located a predetermined distance above the assembly table 166. In one embodiment, the tool is located at about 1.5 inches above the assembly table 166. Then at 406 the arm 220 is instructed to extend along a horizontal axis a predetermined distance. In one embodiment, the arm 220 is instructed to extend along a full (maximum) horizontal range of motion of the robot 214. The actual location of the tool 228 is then measured at 408. In particular, the position of the tool 228 along the height of the assembly table 166 (y-axis), and the position of the tool above the assembly table (z-axis) are recorded. Because the robot 214 is instructed to extend horizontally, the position of the tool 228 along the length of the assembly table 166 (x-axis) is assumed to remain constant. A deviation amount is then determined by comparing the actual location of the tool 228 to the intended location of the tool. A calibration coefficient is then calculated based of the deviation amount at 410. The calibration coefficient can then be used to account for any deviations in movement by the robot 214. The same routine can then be performed on the second robot 216 at 412 to calibrate the second robot.

**[00156]** Referring to FIG. 56, a calibration routine for the tool 228 may also be conducted. During this calibration routine, the tool 228 is first oriented in a horizontal position at 500. Coordinates of an end of the tool 228 are determined at 502. The robot 214, 216 is then operated to rotate the tool 180 degrees at 504. The new coordinates for the end of the tool 228 are then recorded at 506. A diameter of the tool 228 can then be calculated using the two recorded coordinates for the end of the tool at 508. Finally, at 510, the center of the tool 228 can be determined by dividing the diameter calculation in half. The calibration of the tool 228 equips the system 10 with the specific dimensions of the tool so that the system can accurately determine the location of the tool for movement of the tool during assembly of the wooden structures WS.

**[00157]** Referring to FIGS. 57 and 58, a placement calibration routine may be performed to verify the ability of the robots 214, 216 to accurately locate an object within the system 10. To begin the calibration routine, the platen assemblies 174, 176 are each assigned a zero position

at 600. The zero positions for each platen assembly 174, 176 are the centers of the attachment surfaces 186 of the platens 182. The zero position for the lower platen assembly 174 may provide the reference for the entire system 10 during the calibration routine. At 602, each of the robots 214, 216 grabs a respective calibration bar 603. In one embodiment, the calibration bars 603 are precision cut elongate metal bars having a precise known length. The calibration bars 603 may have a generally rectangular shape with a pointed end. Thus, when the robots 214, 216 grab the calibration bars 603, the locations of the pointed ends are known. At 604, the robots 214, 216 are instructed to located the calibration bars (i.e., pointed ends) at the zero position of the lower platen assembly 174. The actual positions of the calibration bars 603 are then inspected by an operator at 606. If the actual positions of the calibration bars 603 are located at the positions the robots 214, 216 were instructed to move the calibration bars to, then a first calibration step is completed at 608. The robots 214, 216 are then moved to new locations and the process can be repeated with the relocated robots at 610. However, if the actual positions of the calibration bars 603 are not located at the positions the robots 214, 216 were instructed to move the bars to, then, at 612 the placement calibration is repeated until the robots successfully locate the calibration bars at the zero position. Additionally, the placement calibration routine may continue at 614 by moving the lower platen assembly 174 to another location and repeating the placement calibration steps at the relocated lower platen assembly. As will be understood, the various calibration routines may be performed prior to using the system 10 for assembling any wooden structures WS to help ensure the system is properly calibrated to perform as needed.

**[00158]** The operations of the components (e.g., in-feed station 14, cutting station 16, buffer station 18, assembly station 20) of the system 10 are controlled by a truss manufacture computing device 12. The control system (broadly, a computer) includes a CPU or processor (e.g., a control system processor) and RAM or memory (broadly, non-transitory computer-readable storage medium). The truss manufacture computing device 12 controls and operates the various components of the manufacturing system 10. Broadly, the memory includes (e.g., stores) processor-executable instructions for controlling the operation of the manufacturing system 10 and the components thereof. The instructions embody one or more of the functional aspects of the manufacturing system 10 and the components thereof, as described herein, with the processor executing the instructions to perform said one or more functional aspects. The components of the manufacturing system 10 may be in wired or wireless communication with the control

system. Other configurations of the control system are within the scope of the present disclosure.

**[00159]** FIG. 59 is a functional block diagram of example truss manufacture computing device 12 that may be used to control the operation of the manufacturing system for fabricating wooden trusses, or components thereof, as described. Specifically, truss manufacture computing device 12 illustrates an example configuration of a computing device for the systems shown herein. Truss manufacture computing device 12 illustrates an example configuration of a computing device operated by a user 595 in accordance with one embodiment of the present invention. Truss manufacture computing device 12 may include, but is not limited to computing devices associated with user interfaces to control the manufacturing system 10, the in-feed station and user interfaces, sensors, conveyors, fiducial printers, multi-line saws, and robotic assemblies. Truss manufacture computing device 12 may also include servers, desktops, laptops, mobile computing devices, stationary computing devices, computing peripheral devices, smart phones, wearable computing devices, and vehicular computing devices. In some variations, truss manufacture computing device 12 may be any computing device capable of the described method for manufacturing and assembling wood structures and, specifically, wood trusses. In some variations, the characteristics of the described components may be more or less advanced, primitive, or non-functional.

**[00160]** In an example embodiment, truss manufacture computing device 12 includes a processor 591 and a memory 592. The processor 591 may be embodied as any type of circuity or device capable of performing the functions described herein. For example, the processor 591 may be embodied as a single or multi-core processor(s), a microcontroller, or other processor or processing/controlling circuit. In some embodiments, the processor 591 may be embodied as, include, or be coupled to a field programmable gate array (FPGA), an application specific integrated circuit (ASIC), reconfigurable hardware or hardware circuitry, or other specialized hardware to facilitate performance of the functions described herein.

**[00161]** In embodiments, the processor 591 is capable of receiving, e.g., from the memory 592 or via an I/O subsystem, a set of instructions which when executed by the processor 591 cause the truss manufacture computing device 12 to perform one or more operations described herein. In embodiments, the processor 591 is further capable of receiving, e.g., from the memory 592 or via the I/O subsystem, one or more signals from external sources, e.g., from

the peripheral devices and/or via a communications interface 594 from an external computing device, external source, or external network. As one will appreciate, a signal may contain encoded instructions and/or information. In embodiments, once received, such a signal may first be stored, e.g., in the memory 592, thereby allowing for a time delay in the receipt by the processor 591 before the processor 591 operates on a received signal. Likewise, the processor 591 may generate one or more output signals, which may be transmitted to an external device, e.g., an external memory or an external compute engine via the communications interface 594 or, e.g., to one or more display devices (e.g., input/output components). In some embodiments, a signal may be subjected to a time shift in order to delay the signal. For example, a signal may be stored in the memory 592 to allow for a time shift prior to transmitting the signal to an external device. One will appreciate that the form of a particular signal will be determined by the particular encoding a signal is subject to at any point in its transmission (e.g., a signal stored will have a different encoding that a signal in transit, or, e.g., an analog signal will differ in form from a digital version of the signal prior to an analog-to-digital (A/D) conversion).

**[00162]** The memory 592 may be embodied as any type of volatile (e.g., dynamic random access memory (DRAM), etc.) and/or non-volatile memory or data storage (e.g., hard disk drive(s), solid-state drive(s), or other data storage device(s)) capable of performing the functions described herein. Volatile memory may be a storage medium that requires power to maintain the state of data stored by the medium. In some embodiments, all or a portion of the memory 592 may be integrated into the processor 591.

**[00163]** Truss manufacture computing device 12 also includes at least one input/output component 593 for receiving information from and providing information to user 595. In some examples, input/output component 593 may be of limited functionality or non-functional as in the case of some wearable computing devices. In other examples, input/output component 593 is any component capable of conveying information to or receiving information from user 595. More specifically, input/output component 593 is configured to provide inputs and outputs for controlling the manufacturing system 10. Thus, input/output component 593 is configured to include inputs for receiving truss designs which are then processed into recipes, providing status information regarding the manufacture of wood trusses (or other components), providing instructions for loading or requesting components such as lumber or connector plates, and providing any diagnostic or alert information as necessary.

**[00164]** In some embodiments, input/output component 593 includes an output adapter such as a video adapter and/or an audio adapter. Input/output component 593 may alternatively include an output device such as a display device, a liquid crystal display (LCD), organic light emitting diode (OLED) display, or "electronic ink" display, or an audio output device, a speaker or headphones. Input/output component 593 may also include any devices, modules, or structures for receiving input from user 595. Input/output component 593 may therefore include, for example, a keyboard, a pointing device, a mouse, a stylus, a touch sensitive panel, a touch pad, a touch screen, a gyroscope, an accelerometer, a position detector, or an audio input device. A single component such as a touch screen may function as both an output and input device of input/output component 593. Input/output component 593 may further include multiple sub-components for carrying out input and output functions.

**[00165]** Truss manufacture computing device 12 may also include a communications interface 594, which may be communicatively couplable to a remote device such as a remote computing device, a remote server, or any other suitable system. Communication interface 594 may include, for example, a wired or wireless network adapter or a wireless data transceiver for use with a mobile phone network, Global System for Mobile communications (GSM), 3G, 4G, 5G or other mobile data network or Worldwide Interoperability for Microwave Access (WIMAX). Communications interface 594 is configured to allow truss manufacture computing device 12 to interface with any other computing device or network using an appropriate wireless or wired communications protocol such as, without limitation, BLUETOOTH®, Ethernet, or IEE 802.11. Communications interface 594 allows truss manufacture computing device 12 to communicate with any other computing devices with which it is in communication or connection.

**[00166]** FIG. 60 is a flow diagram 6000 representing an example method for controlling the operation of the manufacturing system for fabricating wooden trusses using the truss manufacture computing device 12 (shown in FIG. 59). The method includes receiving 6010, at truss manufacture computing device 12, designs for a plurality of wooden structures. In the example embodiment, the designs for the plurality of wooden structures represent designs for a plurality of wood roof trusses. In some embodiments, each design further includes design data. In some embodiments, each design data includes element placement data, element geometric orientation data, lumber definition data, lumber cutting data, connector definition data, and

connection data defining connections between elements. The element placement data represents the geometric placement of each element of corresponding design, relative to other elements and to the corresponding design as a whole. Each element may represent a component portion of the wooden structure including lumber components, connector plates, and any other appropriate structural element. The element geometric orientation data represents the geometric orientation of each element of corresponding design, relative to other elements and to the corresponding design as a whole. The lumber definition data includes information to identify appropriate properties of lumber defined as elements in the design, including tree type, grade, density requirements, and other quality requirements. The lumber cutting data includes the required shape of the cut lumber for use in manufacturing the wood roof truss. The connector definition data includes the grade, size, and structural definitions of any connector defined as an element in the design. The connection data includes the spatial geometric relationship between two or more elements to create connections or joints within each design.

**[00167]** The method further includes processing 6020, at truss manufacture computing device 12, the designs for the plurality of wooden structures to identify a recipe for manufacturing and assembling each of the plurality of wooden structures defined in each corresponding design. (In an example embodiment, truss manufacture computing device 12 includes a design processing module configured to perform processing 6020 and the steps described herein.) The recipe for each design includes using the truss manufacture computing device 12 to identify an ordered sequence of lumber components to be used to manufacture the corresponding wooden structure. In the example embodiment, the ordered sequence is determined by the truss manufacture computing device 12 based upon simulations of varying sequences for assembly of the wooden structure. Each of the simulated sequences is assessed based on simulation characteristics including the expected time to manufacture, expected joint quality, and expected travel time for robots used in manufacture. In the example embodiment, the simulation characteristics are assessed based on a simulation of manufacture of an associated wooden structure according to each simulated sequence and each associated recipe. Accordingly, expected time to manufacture represents the simulated time to manufacture according to each simulated sequence and each associated recipe. Similarly, expected joint quality represents the expected quality of each joint of elements (based on predicted "fit" or "tightness" between elements, particularly lumber pieces and connector plates) with the preferred

expected joint quality being a "tight" joint with minimal deviation from the associated design. Further, expected travel time for robots represents the amount of time that the simulation expects assembly robots to move to manufacture each associated wood structure for each simulated sequence and associated recipe. The ordered sequence is determined as the one having the preferred simulation characteristics. In the example embodiment, the preferred simulation characteristics are the lowest expected time to manufacture, the highest expected joint quality, and the lowest expected travel time for robots used in manufacture. The preferred simulation characteristics may be set by a user or the system and may include further simulation characteristics.

**[00168]** In one example, the simulation of manufacture may use the following approach. As each design provides details on the locations and geometric alignment of wood elements and metal connectors, a simulation module identifies possible "start points" for beginning the manufacture of the wood truss. Possible start points are typically selected from the exterior of the truss, typically at or near a horizontal end of the wood truss. The simulation module then simulates a manufacture of trusses based on each start point with each simulated truss manufactured in a joint-by-joint basis, where each new joint is connected to the truss as it is manufactured. Because the extrusion model is presumed for operational efficiency, the simulator does not typically evaluate or consider the manufacture of non-connected joints for a given simulated truss. By way of analogy, the simulation module essentially creates a tree structure in each simulation, identifying possible simulations based on all connected branches created from connecting start points to joints and to subsequent joints, until the truss is assembled. Typically, due to the norms of truss design, the simulation module proceeds in a generally horizontal fashion from one horizontal end of a truss to the other. Generally, the simulation proceeds along each available "joint path" that completes the simulation. Further, the start point is not necessarily the horizontal endpoint of a given design because in many examples manufacture from such an endpoint may not be preferred. For example, many designs for wood trusses involve small wooden elements or narrow joints that are less desirable locations on which to begin manufacture. For each simulation, the qualities of expected time to manufacture, expected joint quality (on a joint-by-joint basis), and expected travel time for robots used in manufacture are evaluated and a preferred path is identified to create the recipe. Because of the prioritization of both time and quality, the simulation is crucial to filter out fast manufacturing joint paths that

have lower expected joint quality as well as complex routes (involving moving back over the same region repeatedly, for example) that may otherwise have higher joint values but slower manufacturing.

**[00169]** Regarding joint quality, it is assessed based on the ability to form and hold a joint that maintains its alignment (as set forth in the design) and tightness. The ability to form and hold a joint is directly related to the ability of the robots used in assembly to physically maneuver into position for assembly. In many examples, due to piece size, joint angle, or the size or weight of wood elements, a particular joint may be more or less difficult to create for a given design depending on when it is constructed, thus explaining the purpose of this criteria.

**[00170]** The method further includes processing 6030 the plurality of recipes, at the truss manufacture computing device 12, to identify a requested lumber input of stock lumber. More specifically, truss manufacture computing device 12 processes each recipe to create lumber instructions to obtain the wood elements needed for each assembly from commonly used stock lumber. Thus, as described below, truss manufacture computing device 12 analyzes each design to obtain recipes, and then analyzes each recipe to identify which wooden elements are needed to assemble each wood truss in the sequence set forth in the corresponding recipe, and how to obtain such wooden elements from inputted stock lumber. The truss manufacture computing device 12 further identifies an appropriate piece of stock lumber from a listing of available lumber, such that each identified piece of stock lumber can be efficiently cut to obtain the identified wooden elements for assembly according to each recipe. In many examples, the identified piece of stock lumber is selected as one optimized to provide elements for manufacturing each truss in sequence, with minimal or no unused lumber. In some examples, the truss manufacture computing device 12 is configured to track available lumber through a lumber inventory module. (In an example embodiment, truss manufacture computing device 12 includes a recipe processing module configured to perform processing 6030 and the steps described herein).

**[00171]** In the example embodiment, truss manufacture computing device 12 determines the requested lumber input as a lumber input with the greatest amount of throughput for manufacturing the wooden trusses according to each recipe and associated ordered sequence. More specifically, the requested lumber input is selected as the lumber with characteristics that allow for providing lumber components (and cut pieces therefrom) for each recipe. For example,

the requested lumber input is typically selected as a piece of lumber that meets lumber definition data requirements for multiple designs (and recipes), and also is sized to be cut into pieces of lumber for the ordered sequence of each recipe. In the example embodiment, the truss manufacture computing device 12 prompts a user to identify and place a piece of stock lumber corresponding to the requested lumber input and the user places the stock lumber at an in-feed station.

**[00172]** The method further includes truss manufacture computing device 12 detecting input of the requested lumber input based on a sensor at the in-feed station. Truss manufacture computing device 12 further is configured to pre-stage 6040 through the automated manufacturing system to the inputted lumber to saw assembly 90. Pre-staging 6040 includes identifying methods of pre-staging the stock lumber into a saw assembly from lumber instructions obtained in processing 6030 and printing fiducials onto each stock lumber based on printing instructions also determined in lumber instructions.

**[00173]** Each of these steps is defined by the lumber instructions (and therefore by the recipes). Truss manufacture computing device 12 instructs the corresponding machinery associated with each step based on the lumber instructions. For example, truss manufacture computing device 12 instructs the conveyance and staging of lumber as defined in lumber instructions, further including printing instructions for printing fiducials on the stock lumber using the fiducial printer.

**[00174]** The method further includes truss manufacture computing device 12 instructing 6050 saw assembly 90 (and robotic arm 92 to saw 94) to cut the stock lumber according to the lumber instructions obtained from processing 6030. Specifically, truss manufacture computing device 12 instructs saw assembly 90 (and robotic arm 92 to saw 94) to make precise cuts along three axes in order to obtain specified lumber pieces of each recipe from the stock lumber. Due to the parallel processing of designs and recipes, the lumber instructions necessarily consider saw cuts that create appropriate lumber pieces for the manufacture of each structure. Notably, after the multi-line saw cuts the stock lumber, the truss manufacture computing device 12 causes the cut lumber pieces to be routed and staged to an appropriate assembly section.

**[00175]** The method further includes truss manufacture computing device 12 instructing 6060 a plurality of robots to assemble a first wooden truss according to a specified recipe. In an example embodiment, two assembly sections are used and the wooden structures are assembled

based upon the corresponding recipe (generated based upon the corresponding design) in a joint-by-joint extrusion sequence (determined based upon the recipe and the ordered sequence) performed by the robots for the corresponding assembly station. Therefore, each joint of wood is assembled by robotic apparatus (described in detail below) and connected by appropriate connectors as set forth by the corresponding recipe. Further, truss manufacture computing device 12 is configured to instruct the plate distribution assembly to obtain each connector plate for each joint as set forth in the recipe, and to instruct the platen assembly to connect each joint with an appropriate plate. The process continues in each assembly station until the entire wooden structure is assembled. Each joint of the wooden structure is formed by arranging lumber pieces relative to each other at a joint forming station and securing the lumber pieces together using connectors (e.g. nailing plates) to form the joints at the joint forming station.

**[00176]** FIG. 61 is a diagram of elements of one or more example truss manufacture computing devices that may be used in the system shown in FIGS. 59 and 60. Specifically, truss manufacture computing device 12 includes a simulation module 6102 that is configured to simulate the manufacture of truss designs. Truss manufacture computing device 12 also includes a simulation evaluation module 6104 that is configured to assess and evaluate each simulation to identify the simulation with preferred simulation characteristics. Truss manufacture computing device 12 also includes a recipe generation module 6106 which creates the recipes described herein for manufacturing and assembling each corresponding truss. Truss manufacture computing device 12 also includes lumber and printing module 6108 which identifies lumber and printing instructions used to identify appropriate stock lumber for performing recipes in parallel, staging the stock lumber through the automated manufacturing system, and printing on the stock lumber with the fiducial printer. Truss manufacture computing device 12 also includes a robotic instruction module 6110 which instructs the robots in each assembly station to perform the joint-by-joint assembly, plating (and conveyance of plates), and truss manufacture. Truss manufacture computing device 12 also includes a user interface module 6112 that is used to present and process the user interfaces that allow users to interact with truss manufacture computing device 12. The modules 6102, 6104, 6106, 6108, 6110, 6112 may be embodied as hardware (e.g., circuitry), virtualized hardware (e.g., portions or combinations of resources exposed through virtualized environments, such as virtual machines or containers, that share host infrastructure), software, or a combination thereof.

**[00177]** Referring now to FIG. 62, the truss manufacture computing device 12, in operation, may perform a method 6200 of preprocessing input data to preemptively correct one or more errors that may otherwise result during the production (e.g., manufacture, assembly, etc.) of wooden structure(s) defined in the input data. In the illustrative embodiment, the method 6200 begins with block 6202 in which the truss manufacture computing device 12 obtains (e.g., from a file or set of files, such as file(s) encoded in an Extensible Markup Language (XML) or other format), input data indicative of one or more wooden structures to be produced by an automated system (e.g., the automated system 10). In doing so, and as indicated in block 6204, the truss manufacture computing device 12 may obtain input data indicative of one or more trusses (the wooden structures) to be produced. As indicated in block 6206, the truss manufacture computing device 12 may obtain input data indicative of a set of multiple wooden structures to be produced (e.g., rather than a single wooden structure). The truss manufacture computing device 12 may obtain input data indicative of a target shape for each wooden structure, a set of parts within the wooden structure, and materials (e.g., nailing plates, grade of lumber) to produce the wooden structure, as indicated in block 6208. In some embodiments, the truss manufacture computing device 12 may obtain input data indicative of a quantity of each wooden structure to produce, as indicated in block 6210. The wooden structures, quantities of the wooden structures, parts, and materials may be defined as or be associated with a job in the input data. Referring briefly to FIG. 65, a user interface 6500 illustrates an example set of shapes 6510, 6512, 6514, 6516 (e.g., trusses) and corresponding nailing plates 6520 and parts 6530 associated with a given shape to be produced. As described herein, the automated system 10 produces the corresponding shape according to a recipe that defines the operations to be performed by the machines (e.g., assembly robots 214, 216, saw assembly 90, etc.) associated with the automated system 10.

**[00178]** Continuing the method 6200, the truss manufacture computing device 12, in the illustrative embodiment, performs, as a function of the obtained input data (e.g., from block 6202) and present properties (e.g., status, available materials, etc.) of the automated system 10, validation operations to determine whether an error will be encountered during production of the wooden structure(s). In doing so, and as indicated in block 6214, the truss manufacture computing device 12 determines whether materials specified in the obtained input data are available to the automated system 10 (e.g., based on status data received by the truss manufacture computing device 12 from machine controllers and/or other sources). In doing so,

the truss manufacture computing device 12 may determine whether nailing plates specified in the obtained input data are available in a nailing plate inventory, as indicated in block 6216. For example, and as indicated in block 6218, the truss manufacture computing device 12 may determine whether the nailing plates are available in one or more nailing plate magazines (e.g., in the magazine rack 250 as may be reported by the plate distribution assembly 24). As indicated in block 6220, the truss manufacture computing device 12 may determine whether lumber specified in the obtained input data is available to the automated system 10 (e.g., based on status data indicative of the present inventory of stock lumber).

**[00179]** Referring now to FIG. 63, in performing the validation operations, the truss manufacture computing device 12 may simulate execution of a recipe for producing the one or more wooden structures (e.g., trusses) defined in the input data, as indicated in block 6222. In doing so, and as indicated in bock 6224, the truss manufacture computing device 12 may simulate execution of the recipe based on internal models (e.g., digital twins) of components (e.g., machines) of the automated system 10. The models may define the dimensions of the robots and aspects of the movements of the robots (e.g., movement speed, range of motion, applied force or pressure, and the like). The truss manufacture computing device 12 may simulate execution based on internal models of assembly robots (e.g., the robots 214, 216), as indicated in block 6226. As indicated in block 6228, the truss manufacture computing device 12 may determine whether the assembly robots will be unable to create a defined joint (e.g., due to an inability to manipulate a board to a target position or angle, due to a lack of available clearance, etc.).

**[00180]** Utilizing the models, the truss manufacture computing device 12 may determine whether the assembly robots 214, 216 will collide (e.g., with each other or with another component of the automated system 10) when performing the operations of the recipe, as indicated in block 6230. As described in more detail herein, collision detection may be performed based on a shared code and data package (e.g., in a microservices architecture) defining geometries of components and shape comparison logic. The truss manufacture computing device 12 may determine whether the assembly robots 214, 216 will be unable to pick up a board (e.g., lumber piece) or part at a defined position (e.g., at a position along the length of a board), such as due to a lack of available clearance, as indicated in block 6232. Referring briefly to FIG. 66, a user interface 6600 that may be provided by the truss manufacture

computing device 12 illustrates a simulated execution of a recipe in which the positions of the robots 214, 216 are tracked, using models 6610, 6612 (e.g., digital twins) of the robots, to determine whether the operations can be executed without error (e.g., without collisions, etc.). The presence of any of the conditions described above (e.g., lack of available nailing plates, lack of available stock lumber, or inability to perform one or more operations in the simulated execution of the recipe) constitutes a corresponding error.

**[00181]** Referring back to FIG. 63, in block 6234, the truss manufacture computing device 12 determines the subsequent course of action based on whether an error was encountered in the above operations. If not, the method 6200 advances to block 6236 in which the truss manufacture computing device 12 may produce an indication (e.g., in the memory 592, via the input/output component 593, in a user interface, etc.) that no errors were encountered. In doing so, the truss manufacture computing device 12 may produce an indication that the corresponding recipe was validated, as indicated in block 6238. Referring back to block 6234, if an error was encountered, the method 6200, in the illustrative embodiment, advances to block 6240 of FIG. 64, in which the truss manufacture computing device 12 identifies, as a function of the validation operations (e.g., from block 6212), one or more adjustments to enable production of the wooden structure(s) without the error(s).

**[00182]** Referring now to FIG. 64, the truss manufacture computing device 12 may present the determined error(s) to a user (e.g., in a user interface), as indicated in block 6242. In doing so, the truss manufacture computing device 12 may receive one or more user-defined adjustments, as indicated in block 6244. Additionally or alternatively, the truss manufacture computing device 12 may identify one or more adjustments as a function of a defined set of available adjustments (e.g., from a lookup table (e.g., in the memory 592) that associates errors with corresponding predefined adjustments to resolve the errors), as indicated in block 6246. As indicated in block 6248, the truss manufacture computing device 12 may determine an offset (e.g., an offset along the y-axis), a rotation angle for the wooden structure, and/or a reflection along one or more axes (e.g., to enable a robot 214, 216 to pick up a part or perform another assembly operation) when the operation would otherwise be impracticable. Referring briefly to FIG. 67, a user interface 6700 that may be presented by the truss manufacture computing device 12 includes an element 6710 that enables a user to select or type a degree of rotation for the wooden structure. In the illustrative embodiment, the truss manufacture computing device 12

updates a representation of the wooden structure in a corresponding window 6712 in response to a change in the rotation specified in the element 6710.

**[00183]** Referring back to FIG. 64, the truss manufacture computing device 12 may determine an adjustment as a function of available materials, as indicated in block 6250. For example, and as indicated in block 6252, the truss manufacture computing device 12 may identify a replacement (e.g., substitute) nailing plate (e.g., when a nailing plate specified in the input data is unavailable). The truss manufacture computing device 12 may identify a replacement nailing plate as a function of dimensions of the specified nailing plate (e.g., as specified in the input data) and dimensions of available nailing plates (e.g., as reported by the plate distribution assembly 24), as indicated in block 6254. In the illustrative embodiment, the truss manufacture computing device 12 may apply a rule (e.g., in identifying a replacement nailing plate) that any nailing plate having dimensions that are equal to or greater than the dimensions of the nailing plate specified in the input data is an acceptable replacement nailing plate. Referring briefly to FIG. 68, a user interface 6800 that may be provided by the truss manufacture computing device 12 includes a set 6810 of nailing plates that are specified in input data (e.g., for a truss) and that are not available to the automated system 10, and a corresponding set 6812 of nailing plates that can used as replacements (e.g., because the nailing plates in the set 6812 have dimensions that are greater than or equal to the dimensions of the nailing plates in the set 6810).

**[00184]** Similarly, the truss manufacture computing device 12 may identify replacement lumber (e.g., if the lumber specified in the input data is not available to the automated system 10), as indicated in block 6256. In doing so, the truss manufacture computing device 12 may identify the replacement lumber as a function of a specified grade for the lumber (e.g., as specified in the input data) and the grades of stock lumber available in the inventory of the automated system 10, as indicated in block 6258. For example, in some embodiments, the truss manufacture computing device 12 may determine that lumber having a grade that is equal to or greater than that specified in the input data is an acceptable replacement. Additional operations that the truss manufacture computing device 12 may perform in selecting replacement lumber are described with respect to FIGS. 110 to 122. The truss manufacture computing device 12, in the illustrative embodiment, stores data indicative of the identified adjustments (e.g., in the memory 592), as indicated in block 6260. In doing so, the truss manufacture computing device 12 may

store the adjustments as user intervention data, as indicated in block 6262. In some embodiments, the truss manufacture computing device 12 may repeatedly execute the operations of the method 6200 (e.g., identifying additional adjustments and incorporating the adjustments into further simulations of execution of the recipe) until no errors are detected (e.g., until the recipe is validated).

**[00185]** In addition to the adjustments (e.g., user intervention data) described above, parameters (e.g., dimensions, offsets, limits, etc.) associated with the components (e.g., machines) of the automated system 10 may be utilized by the truss manufacture computing device 10 in the creation or simulated execution of a recipe. The parameters may be defined or adjusted on a periodic basis pursuant to the calibration routines described above. An embodiment of a set of such parameters 6910 is shown in the user interface 6900 of FIG. 69.

**[00186]** Referring now to FIG. 70, in the illustrative embodiment, the truss manufacture computing device 12 may perform a method 7000 of creating (e.g., generating) recipe(s), each indicative of a set of operations to be performed by components (e.g., machines) of the automated system 10 to produce one or more wooden structures (e.g., trusses). In the illustrative embodiment, the method 7000 begins in block 7002 in which the truss manufacture computing device 12 obtains recipe generation input data, which may be embodied as any data indicative of parameters with which the recipe should comply in the coordination of components (e.g., machines of the in-feed station 14, buffer station 18, assembly station 20) of the automated system 10 to produce one or more wooden structures (e.g., wooden trusses). The truss manufacture computing device 12 may obtain the recipe generation input data from memory 592 (e.g., read from a file or database), from another computing device (e.g., via the communications interface 594), from the input/output interface 593, and/or other sources. In obtaining recipe generation input data, the truss manufacture computing device 12 may obtain production data indicative of one or more jobs (e.g., batches indicative of quantities of each of one or more wooden structures (e.g., trusses) to be produced), as indicated in block 7004. The truss manufacture computing device 12 may also obtain user intervention data indicative of one or more user interventions (e.g., adjustments, such as replacement materials (e.g., replacement nailing plates, replacement lumber), a rotation to be applied to one or more wooden structures, etc.). In some embodiments, at least a portion of the user interventions may be produced as a result of execution of the method 6200 of FIGS. 62-64.

**[00187]** In obtaining recipe generation input data, the truss manufacture computing device 12 may also obtain truss shape data which may be embodied as any data indicative of a target shape for a truss (e.g., the shape of the truss to be produced), as indicated in block 7008. Further, and as indicated in block 7010, the truss manufacture computing device 12 may obtain assembler parameter data indicative of parameters (e.g., offsets, dimensions, limits, etc.) for one or more devices (e.g., components, machines, etc.) of the automated system 10 used to assemble or otherwise produce the wooden structure(s) (e.g., trusses). The parameter data may include parameter data discussed above with reference to FIG. 69, which may be produced at least in part from the calibration operations described above. Continuing the method 7000, in block 7012, the truss manufacture computing device 12 may identify, as a function of the obtained recipe generation input data (e.g., from block 7002), one or more unique trusses. In doing so, the truss manufacture computing device 12 may identify the unique trusses as a function of truss identifiers specified in the input data, job names, truss labels, and/or batch names, as indicated in block 7014. As indicated in block 7016, to improve efficiency (e.g., to reduce the possibility of creating two or more recipes for the same unique truss), the truss manufacture computing device 12 filters out (e.g., excludes from a set for further analysis) non-unique trusses (e.g., any trusses not identified as unique pursuant to the above operations).

**[00188]** Subsequently, and as indicated in block 7018, the truss manufacture computing device 12 determines the subsequent course of action based on whether all recipes have been generated for all unique trusses identified from the obtained recipe generation input data. If recipes have been generated for all of the unique trusses, the truss manufacture computing device 12 outputs the recipes for the unique trusses (e.g., writing the recipes to memory 592 (e.g., as file(s), data sets in a database, etc.), sending the recipes to another computing device via the communications interface 594, etc.), as indicated in block 7020. When at least one unique truss does not have a corresponding recipe, the method 7000 instead branches to block 7022 in which the truss manufacture computing device 12 selects a unique truss for which a recipe has not yet been generated. Subsequently, in block 7024, the truss manufacture computing device 12 generates a recipe for the selected truss (e.g., to produce the truss in a joint-by-joint extrusion sequence with the automated system 10). In doing so, the truss manufacture computing device 12 determines operations of components of an automated structure manufacturing system (e.g., the automated system 10) to efficiently produce the selected truss (e.g., in a joint-by-joint extrusion

sequence), as indicated in block 7026. Afterwards, the method 7000 loops back to block 7018 to determine whether any other unique trusses do not have a corresponding recipe. Embodiments of methods associated with blocks 7024, 7026 for generating a recipe for a given (e.g., selected truss) are described in more detail herein.

**[00189]** Referring now to FIG. 71, the truss manufacture computing device 12, in the illustrative embodiment, may execute a method 7100, corresponding to blocks 7024, 7026 of the method 7000, for creating a recipe for a selected truss. In the illustrative embodiment, the method 7100 begins with block 7102 in which the truss manufacture computing device 12 produces a list of joints for the selected truss. An embodiment of a method 7300 that may be executed by the truss manufacture computing device 12 to produce the list of joints is described herein with reference to FIG. 73. Still referring to FIG. 71, in block 7104, the truss manufacture computing device 12 determines, from the produced list of joints and as a function of (e.g., based on) one or more production target(s) (e.g., to minimize wasted lumber, to minimize robotic movements, to minimize production time, to maximize tightness of fit (e.g., joint quality), etc.) an ordered set of joints. The production targets may be defined as default targets (e.g., hard coded or defined in a default configuration file), defined in a set of recipe creation input data (e.g., production data), defined through a user interface (e.g., via the input/output interface 593), or established otherwise. Regardless, in determining the ordered set of joints, the truss manufacture computing device 12 calculates (e.g., determines) the first joint for manufacture of the selected truss, as indicated in block 7106 and calculates (e.g., determines) an order for remaining joints of the selected truss, as indicated in block 7108. That is, the truss manufacture computing device 12 determines an order for the joints based on a determination that the order will satisfy the production targets as well as or better than (i.e., to a greater degree than) any other order of the joints. An embodiment of a method 7500 for calculating the first joint is described with reference to FIG. 75 herein. Similarly, an embodiment of a method 8000 that may be executed by the truss manufacture computing device 12 for calculating an order for the remaining joints of the selected truss is described with reference to FIG. 80.

**[00190]** Still referring to FIG. 71, in block 7110, the truss manufacture computing device 12 determines, as a function of the ordered set of joints, a set of assembly operations. An embodiment of a method 8300 for determining the set of assembly operations for an ordered set of joints is described with reference to FIG. 83 herein. After determining the set of assembly

operations, the truss manufacture computing device 12 determines the subsequent course of action as a function of whether remaining operations (e.g., operations that have not been further analyzed, as described herein) are present in the set of assembly operations. In response to a determination that no remaining operations are present, the method 7100 advances to block 7114 in which the truss manufacture computing device outputs a set of operations (e.g., the recipe) for the selected truss. Initially, however, all of the assembly operations in the set will be remaining and the method 7100 will instead advance to block 7116 in which the truss manufacture computing device 12 selects the next assembly operation in the set. Having selected the next assembly operation, the truss manufacture computing device 12 calculates a set of recipe operations for the selected assembly operation, as indicated in block 7118. That is, the truss manufacture computing device 12 may calculate multiple recipe operations for a given assembly operation, as described herein. A method 7200 that may be executed by the truss manufacture computing device 12 to calculate the set of recipe operations for a selected assembly operation is described with reference to FIG. 72. In the illustrative embodiment, the method 7100 repeatedly loops back to block 7112, discussed above, until no assembly operations remain in the set.

**[00191]** Referring now to FIG. 72, in operation, the truss manufacture computing device 12 may execute a method 7200 for calculating a set of recipe operations for a selected assembly operation. The method 7200 corresponds to block 7118 of FIG. 71. In the illustrative embodiment, the truss manufacture computing device 12 calculates a set of primary robot pickup options that the truss manufacture computing device 12 determines to be valid (e.g., satisfying a set of criteria), as indicated in block 7202. A method 9000 that may be executed by the truss manufacture computing device 12 to calculate primary robot pickup options is described with reference to FIG. 90. Still referring to FIG. 72, in block 7204, the truss manufacture computing device 12 calculates a set of secondary robot pickup options that the truss manufacture computing device 12 determines to be valid (e.g., satisfying a set of criteria). A method 9100 that may be executed by the truss manufacture computing device 12 to calculate the valid secondary robot pickup options is described with reference to FIG. 91. In block 7206, the truss manufacture computing device 12 determines (e.g., identifies every permutation of) pairs of pickup options from the valid primary and secondary robot pickup options (e.g., from blocks 7202 and 7204). In block 7208, the truss manufacture computing device 12 selects a pickup option pair from the set of pairs of pickup option pairs determined in block 7206. Subsequently, in block 7210, the truss

manufacture computing device 12 assigns primary and secondary robot roles to first and second robots (e.g., the robots 214, 216). A method 9200 that may be executed by the truss manufacture computing device 12 for assigning the primary and secondary roles to the robots 214, 216 is described with reference to FIG. 92.

**[00192]** After assigning the primary and secondary roles to the robots 214, 216, the truss manufacture computing device 12, in the illustrative embodiment, calculates gantry (e.g. carriage 218) and robot arm (e.g., arm 220) positions for the first and second robots 214, 216. An embodiment of a method 9300 that may be executed by the truss manufacture computing device 12 to calculate the gantry and robot arm positions for the robots 214, 216 is described with reference to FIG. 93. Continuing the method 7200, the truss manufacture computing device 12 determines whether the selected pickup option pair (e.g., selected in block 7208) is to be designated as the best pickup option pair, as indicated in block 7214. In the context of the present disclosure, the terms "better" and "best" indicate that the manufacture computing device 12 has determined, according to a set of operations or criteria, to designate an item as such. A method 9900 that may be executed by the truss manufacture computing device 12 to determine whether the selected pickup option pair is the best pickup option pair (e.g., whether the pickup option pair should be designated as the best pickup option pair) is described with reference to FIG. 99.

**[00193]** In block 7216, the truss manufacture computing device 12 determines whether any additional pairs of pickup options are available to be analyzed. If so, the method 7200 loops back to block 7208 in which the truss manufacture computing device 12 selects the next pickup option pair from the set and repeats the operations described above. Otherwise, the method 7200 advances to block 7218 in which the truss manufacture computing device 12 outputs (e.g., writes to the memory 592, outputs via the input/output interface 593, and/or the communications interface 594, etc.) the pickup option pair that the truss manufacture computing device 12 determined is the best. In block 7220, the truss manufacture computing device 12 adds recipe operations determined (e.g., by the truss manufacture computing device 12) to be necessary. An embodiment of a method 10400 that may be executed by the truss manufacture computing device 12 for adding recipe operations determined to be necessary is described with reference to FIG. 104. Further, in block 7222, the truss manufacture computing device 12 may add one or more intermediate recipe operations. An embodiment of a method 10600 for adding intermediate recipe operations that may be executed by the truss manufacture computing device 12 is

described with reference to FIG. 106.

**[00194]** Referring now to FIG. 73, the truss manufacture computing device 12 may perform a method 7300 for producing a list of joints for a selected truss. The method 7300 corresponds to block 7102 of the method 7100 described with reference to FIG. 71. In the illustrative embodiment, the method 7300 begins in block 7302 in which the truss manufacture computing device 12 converts truss shape data (e.g., obtained in block 7008 of the method 7000) to points and polygons in a two-dimensional orthogonal coordinate system (e.g., a Cartesian coordinate system with x and y axes). In doing so, in the illustrative embodiment, the truss manufacture computing device 12 defines the origin as the furthest point on the selected truss (e.g., the truss represented in the shape data), as indicated in block 7304. In block 7306, the truss manufacture computing device 12 determines a subsequent course of action based on whether the user invention data (e.g., obtained in block 7006 of the method 7000) indicates any geometric user interventions (e.g., adjustments to the geometry of the truss). If so, the method 7300 advances to block 7308 in which the truss manufacture computing device 12 applies the user intervention(s) defined in the user intervention data to the converted truss shape data (e.g., converted in block 7302). In doing so, the truss manufacture computing device 12 may apply rotations that are defined in the user intervention data, as indicated in block 7310. Additionally or alternatively, the truss manufacture computing device 12 may flip (e.g., reflect) the selected truss along an axis, as indicated in block 7312. That is, the truss manufacture computing device 12 may flip (e.g., reflect) the truss along the x-axis as indicated in block 7314 and/or the y-axis, as indicated in block 7316 in accordance with the user intervention data. As indicated in block 7318, the truss manufacture computing device 12 may apply an offset to the selected truss. For example, and as indicated in block 7320, the truss manufacture computing device 12 may apply an offset along the y-axis, as defined in the user intervention data. Afterwards, or in response to a determination in block 7306 that no geometric user interventions are present, the method 7300 advances to block 7322 of FIG. 74.

**[00195]** Referring now to FIG. 74, in at least some embodiments, in block 7322, the truss manufacture computing device 12 converts all part angles to plus or minus 90 degrees (e.g., thereby reducing the data to be sent to a corresponding machine of the automated system 10). Subsequently, the truss manufacture computing device 12 may shorten, for tolerance, a subset of the parts of the selected truss, as indicated in block 7324. In doing so, the truss manufacture

computing device 12 may identify parts that are not perimeter parts (e.g., located on an outer boundary of the truss) and that are not wedge parts (e.g., forming a wedge), as indicated in block 7326. As indicated in block 7328, of the remaining parts (e.g., the parts that are not perimeter parts and that are not wedge parts), the truss manufacture computing device 12 may shorten non-vertical parts by a greater amount than vertical parts (e.g., as vertical parts may be more critical). In doing so, the truss manufacture computing device 12 may shorten non-vertical parts by one eighth of an inch, as indicated in block 7330 and may shorten vertical parts by one sixteenth of an inch, as indicated in block 7332.

**[00196]** Subsequently, in block 7334, the truss manufacture computing device 12 may generate, for each nailing plate associated with the selected truss (e.g., as indicated in the recipe generation input data obtained in block 7002 of the method 7000), one or more joint objects (e.g., data objects representative of joints). In doing so, and as indicated in block 7336, the truss manufacture computing device 12 temporarily creates (e.g., in the memory 592) a smaller plate rectangle (e.g., a rectangle having dimensions smaller than the rectangle defined by the nailing plate). In the illustrative embodiment, the truss manufacture computing device 12 temporarily creates the smaller plate rectangle to determine which part of the truss belongs to which nailing plate, as indicated in block 7338. Further, in generating the joint objects, the truss manufacture computing device 12 illustratively generates joint objects that include plate (e.g., nailing plate) and parts information associated with the corresponding joint, as indicated in block 7340. Subsequently, in block 7342, the truss manufacture computing device 12 outputs (e.g., writes to memory 592, etc.) a list of joints for the selected truss.

**[00197]** Referring now to FIG. 75, the truss manufacture computing device 12 may execute a method 7500 for calculating (e.g., selecting) the first joint for a selected truss. The method 7500 corresponds to block 7106 of the method 7100. In the illustrative embodiment, the method 7500 begins with block 7502 in which the truss manufacture computing device 12 determines whether the user identified (e.g., in the obtained recipe generation input data from block 7002) a valid first joint for the selected truss. If so, the method 7500 advances to block 7504 in which the truss manufacture computing device 12 designates the user-identified first joint as the first joint for the selected truss. Otherwise, if the user did not identify a valid first joint, the method 7500 instead advances to block 7506, in which the truss manufacture computing device 12 initially designates the best first joint as the first joint the list of joints for

the truss (e.g., the list of joints produced in block 7102 of the method 7100). Afterwards, the method 7500 advances to block 7508, in which the truss manufacture computing device 12 determines a best first joint from the set of all joints in the list of joints. In doing so, the truss manufacture computing device 12 determines the best first part in each joint, as indicated in block 7510. An embodiment of a method 7600 that may be executed by the truss manufacture computing device 12 to determine the best first part in a selected joint is described with reference to FIG. 76. In block 7512, the truss manufacture computing device 12 determines an order for remaining parts in the joint (e.g., after the best first part for the joint has been determined). An embodiment of a method 7700 that may be executed by the truss manufacture computing device 12 for determining an order for the remaining parts in a selected joint is described with reference to FIG. 77.

**[00198]** In block 7514, the truss manufacture computing device 12 calculates a joint score for each joint in the selected truss. An embodiment of a method 7800 that may be executed by the truss manufacture computing device 12 for calculating a joint score for a given joint in a truss is described with reference to FIG. 78. In the illustrative embodiment, the truss manufacture computing device 12 executes that method 7800 for each joint in the list of joints for the selected truss to determine a corresponding joint score. Further, in block 7516, the truss manufacture computing device 12 designates the joint with the highest joint score as the best first joint. Subsequently, the truss manufacture computing device 12 designates the best first joint (e.g., determined from the operations of block 7508) as the first joint (e.g., the output of the method 7500), as indicated in block 7518. After executing block 7518 or block 7504 (e.g., if the user entered a valid first joint), the truss manufacture computing device 12 outputs (e.g., writes to memory 592, etc.) the first joint (e.g., an identifier of the first joint), as indicated in block 7520.

**[00199]** Referring now to FIG. 76, the truss manufacture computing device 12 may execute a method 7600 for determining a best first part in a selected joint. The method 7600 corresponds to block 7508 of the method 7500. In the illustrative embodiment, the method 7600 begins with block 7602 in which the truss manufacture computing device 12 selects the next part (e.g., from a list of parts) of the selected joint for analysis. After selecting the part, the method 7600 advances to block 7604 in which the truss manufacture computing device 12 determines whether the selected part is a wedge. If not, the method 7600 advances to block 7606 in which the truss manufacture computing device 12 designates the selected part as the newest best part in

the set of parts for analysis, then loops back to block 7602 to select the next part to analyze. Otherwise, if the selected part is a wedge, the method 7600 instead advances from block 7604 to block 7608, in which the truss manufacture computing device 12 determines whether the selected part is long enough for a clamp (e.g., clamps 234 of the gripper 232). That is, to increase the likelihood that sufficient space will exist around a pressing area to hold the part with a clamp (e.g., the clamps 234), the truss manufacture computing device 12 determines whether the length of the part is greater than or equal to a predefined length (e.g., in memory 592) associated with the clamp. If so, the method advances to block 7606 to identify the selected part as the newest best part 7606 and loop back to block 7602 to select the next part for analysis.

**[00200]** Otherwise, if the truss manufacture computing device 12 determines that the part is not long enough for the clamp, the method 7600 advances to block 7610, in which the truss manufacture computing device 12 determines whether the selected part has an angle of less than 140 degrees or greater than 220 degrees. The truss manufacture computing device 12 makes the determination, in some embodiments, because the robot 214 cannot reach around the press 180 towards the robot 216 past a defined extent. In response to a determination that the angle of the part satisfies the conditions of block 7610, the method 7600 advances to block 7606, in which the truss manufacture computing device 12 designates the selected part as the newest best part and loops back to block 7602 to select another part for analysis.

**[00201]** If the conditions are not satisfied in block 7610, the method 7600 instead advances to block 7612, in which the truss manufacture computing device 12 determines whether the selected part is at least 56% on the table (e.g., the assembly table 166). The truss manufacture computing device 12 makes the determination because, in at least some embodiments, vertical parts may fall in a table gap where the press (e.g., the press 180) moves in a plus or minus y direction (e.g., along a y axis). If the selected part is at least 56% on the table (e.g., the assembly table 166), the method 7600 advances to block 7606, in which the truss manufacture computing device 12 designates the selected part as the newest best part and loops back to block 7602 to select another part for analysis. Otherwise, if the selected part is not at least 56% on the table, the method 7600 advances to block 7614, in which the truss manufacture computing device 12 determines whether the selected part is a perimeter part (e.g., forms a portion of the perimeter of the truss). In the illustrative embodiment, the truss manufacture computing device 12 applies a bias (e.g., a preference) for a perimeter part as the first part, rather than a non-perimeter part. In

response to a determination that the selected part is a perimeter part, the method 7600 advances to block 7606, in which the truss manufacture computing device 12 designates the selected part as the newest best part, then loops back to block 7602 to select another part for analysis.

**[00202]** Otherwise, if the selected part is not a perimeter part, the method 7600 instead advances to block 7616, in which the truss manufacture computing device 12 determines whether a number of a zone associated with the selected part is less than the number of the zone associated with the current best part. In doing so, the truss manufacture computing device 12 may utilize a lookup table or other data structure that associates angles with zones. In the illustrative embodiment, the truss manufacture computing device 12 determines the zone based on a zero-based index of zones in 18 degree increments in the positive or negative directions, such that if the angle is between 0 and 18 degrees or 0 and -18 degrees, the zone is 0, if the angle is between 18 and 36 degrees or -18 and -36 degrees, the zone is 1, and so on. In response to a determination that the zone number for the selected part is less than the zone number for the current best part, the method 7600 advances to block 7606, in which the truss manufacture computing device 12 designates the selected part as the newest best part and loops back to block 7602 to select the next available part for analysis. Otherwise, the method 7600 advances to block 7618, in which the truss manufacture computing device 12 determines whether the selected part has a greater percentage on the table (e.g., the assembly table 166) than the current best part. If so, the method 7600 advances to block 7606, in which the truss manufacture computing device 12 designates the selected part as the newest best part and loops back to block 7602 to select the next available part for analysis. Otherwise, if the selected part is does not have a greater percentage on the table 166, the method 7600 loops back to block 7602 to select the next available part for analysis without designating the current selected part as the newest best part. The method 7600 continually loops until all of the parts have been analyzed for their potential to be designated as the best part.

**[00203]** Referring now to FIG. 77, the truss manufacture computing device 12 may execute a method 7700 for determining an order for the remaining parts in a selected joint (e.g., after the first part has been determined). The method 7700 corresponds to block 7512 of the method 7500. The method 7700, in the illustrative embodiment, begins with block 7702, in which the truss manufacture computing device 12 selects a part from the list of remaining parts for the joint (e.g., the parts other than the part that was designated as the first part). Subsequently,

in block 7704, the truss manufacture computing device 12 sets a part score for the selected part as a function of a set of part score factors. The truss manufacture computing device 12 may initially set the part score for the part to zero, then adjust the part score as a function of the determinations made within block 7704. In doing so, the truss manufacture computing device 12 may increase the part score as a function of whether the selected part touches other parts that are already in the selected joint, as indicated in block 7706. The truss manufacture computing device 12 may determine whether the part touches other parts by comparing coordinates associated with the geometries of the parts. As described in more detail herein, in at least some embodiments, the truss manufacture computing device 12 utilizes a shared package in a microservices architecture with executable instructions for performing geometric comparisons (e.g., to detect overlaps, collisions, contact, relative sizes, etc.). In some embodiments, the truss manufacture computing device 12 may increase the part score of the selected part based on the number of parts that are touched by the selected part (e.g., increasing the part score by one for each part that is touched by the selected part), as indicated in block 7708. As such, the truss manufacture computing device 12, in operation, applies a bias (e.g., a preference) to place a part that already touches one or more other parts. Additionally or alternatively, the truss manufacture computing device 12 may increase the part score if the select part is a perimeter part, as indicated in block 7710. The truss manufacture computing device 12 may also increase the part score if the selected part is not a wedge, as indicated in block 7712. As such, the truss manufacture computing device 12, in at least some embodiments, applies a bias or preference for perimeter parts and non-wedge parts.

**[00204]** Subsequently, in block 7714, the truss manufacture computing device 12 determines whether remaining parts are available in the joint (e.g., that have not yet been assigned a part score). If so, the method loops back to block 7702 to select the next part. Otherwise, the method 7700 advances to block 7716, in which the truss manufacture computing device 12 adds the highest scoring part (e.g., the part with the highest part score) to a list of parts for the selected joint. The method 7700 advances to block 7718 in which the truss manufacture computing device 12 determines whether parts remain to be analyzed. If so (e.g., if not all of the parts have been added to the list referenced in block 7716), the method 7700 loops back to block 7702 to select the next remaining part for analysis (e.g., assignment of a part score). Otherwise (e.g., if no parts remain), the method 7700 advances to block 7720, in which the truss manufacture computing device 12 outputs (e.g., writes to memory 592) the order of parts (e.g.,

the list of parts from block 7716) for the selected joint.

**[00205]** Referring now to FIG. 78, the truss manufacture computing device 12 may execute a method 7800 to calculate a joint score. The method 7800 corresponds to block 7514 of the method 7500. In the illustrative embodiment, the method 7800 begins with block 7802, in which the truss manufacture computing device 12 sets the joint score for the selected joint to zero. In block 7804, the truss manufacture computing device 12, in the illustrative embodiment, may increase the joint score as a function of one or more joint score factors. In doing so, the truss manufacture computing device 12 may increase the joint score by one if the first part of the joint is not a wedge, as indicated in block 7806. Similarly, the truss manufacture computing device 12 may increase the joint score by one if the second part of the joint is also not a wedge, as indicated in block 7808. As such, the truss manufacture computing device 12 implements a preference or bias against utilizing wedge parts as the first two parts of a joint. As indicated in block 7810, the truss manufacture computing device 12 may increase the joint score by one if the first part is a perimeter part. Likewise, if the second part is a perimeter part, the truss manufacture computing device 12 may increase the joint score by one, as indicated in block 7812. Accordingly, in operation, the truss manufacture computing device 12 applies a bias or preference for joints in which the first two parts are perimeter parts. The truss manufacture computing device 12 may increase the joint score by one if the first part is long enough for a clamp (e.g., satisfies a defined length associated with a clamp, such as the clamps 234 of the gripper 232), as indicated in block 7814. Similarly, the truss manufacture computing device 12 may increase the joint score by one if the second part is long enough for the clamp, as indicated in block 7816.

**[00206]** Further, the truss manufacture computing device 12 may increase the joint score based on a percentage of the first part on a table (e.g., the assembly table 166), as indicated in block 7818. In doing so, the truss manufacture computing device 12 may increase the joint score by one if the first part is more than 75% on the table (e.g., the assembly table 166), as indicated in block 7820. Otherwise, if the first part is not more than 75% on the table, the truss manufacture computing device 12, in the illustrative embodiment, increases the joint score by the percentage of the first part on the table, as indicated in block 7822. Referring now to FIG. 79, the truss manufacture computing device 12 increases the joint score based on the percentage of the second part on the table (e.g., the assembly table 166), as indicated in block 7824. In doing so,

the truss manufacture computing device 12 increases the joint score by one if the second part is more than 75% on the table, as indicated in block 7826. Otherwise, the truss manufacture computing device 12 increases the joint score by the percentage of the second part on the table, as indicated in block 7828. Accordingly, through the operations of blocks 7818, 7820, 7822, 7824, 7826, 7828, the truss manufacture computing device 12 applies a preference or bias for joints in which the first two parts are unlikely to fall in a gap in the table (e.g., the assembly table 166) where the press 180 moves.

**[00207]** Still referring to FIG. 79, the truss manufacture computing device 12 may increase the joint score based on an angle between the first and second parts, as indicated in block 7830. In doing so, the truss manufacture computing device 12 may increase the joint score by one if the angle is greater than 45 degrees, as indicated in block 7832. Otherwise, the truss manufacture computing device 12, in the illustrative embodiment, increases the joint score by the number of degrees of the angle between the parts (e.g., divided by 100 and expressed as a decimal), as indicated in block 7834. Accordingly, the truss manufacture computing device 12 implements a preference for joints in which the first two parts form a large (e.g., greater than 45 degrees) angle, as larger angles may increase the ability of the automated system 10 to utilize both robots 214, 216 rather than a single robot on the parts. In block 7836, the truss manufacture computing device 12 outputs (e.g., writes to memory 592) the joint score for the selected joint.

**[00208]** Referring now to FIG. 80, the truss manufacture computing device 12 may execute a method 8000 for calculating an order for remaining joints of a selected truss (e.g., after the first joint has been determined). The method 8000 corresponds to block 7108 of the method 7100. In the illustrative embodiment, the method 8000 begins with block 8002, in which the truss manufacture computing device 12 determines whether unadded joints remain (e.g., joints of the truss that have not been added to an ordered set of joints). In response to a determination that no unadded joints remain, the method 8000 advances to block 8004 in which the truss manufacture computing device 12 outputs (e.g., writes to memory 592) the order for the remaining joints (e.g., other than the first joint) of the selected truss. Otherwise, if unadded joints do remain, the method 8000 instead advances to block 8006 in which the truss manufacture computing device 12 determines the next best joint. An embodiment of a method 8100 that may be executed by the truss manufacture computing device 12 for determining the next best joint is described with reference to FIG. 81. After determining the next best joint, the truss manufacture computing

device 12, in the illustrative embodiment, determines an order of parts for the determined next best joint, as indicated in block 8008. An embodiment of a method 8200 that may be executed by the truss manufacture computing device 12 for determining the order of parts for the determined next best joint (e.g., from block 8006) is described with reference to FIG. 82.

**[00209]** Referring now to FIG. 81, the truss manufacture computing device 12 may execute a method 8100 for determining the next best joint in a truss. The method 8100 corresponds to block 8006 of the method 8000. In the illustrative embodiment, the method 8100 begins with block 8102 in which the truss manufacture computing device 12 determines whether any unadded joints remain (e.g., joints in the truss that have not been added to an ordered set of joints). If not, the method 8100 advances to block 8104, in which the truss manufacture computing device 12 outputs (e.g., writes to memory 592) the joint determined to be the next best joint. However, if unadded joints still remain, the method 8100 instead advances to block 8106, in which the truss manufacture computing device 12 selects a remaining joint of the selected truss. Further, in block 8108, the truss manufacture computing device 12 determines whether the selected joint (e.g., from block 8106) is the next best joint. In doing so, the truss manufacture computing device 12 determines that the selected truss is the next best joint if the selected joint will cause a truss collision if the joint is picked (e.g., as the next best trust) later, as indicated in block 8110. That is, the truss manufacture computing device 12 may determine that if the selected joint is not designated as the next best truss in the current iteration of the method 8100, then the automated system 10 may be later required to move the truss beyond a predefined distance from a starting position, which may cause a collision (e.g., with a wall, such as in a –x direction). As indicated in block 8112, the truss manufacture computing device 12 may determine that the selected join is the next best joint if all parts of the joint are already placed with one or more previous joints. As such, the truss manufacture computing device 12 may determine to finish a joint if all other parts of the joint are already in place.

**[00210]** In some embodiments, the truss manufacture computing device 12 determines that the selected joint is the next best joint if the selected joint is a splice, as indicated in block 8114. A splice may be embodied as two parts (e.g., parallel parts) that are connected at corresponding ends. As indicated in block 8114, the truss manufacture computing device 12 may determine that the selected joint is the next best joint if the selected joint has more parts than the current joint designated as the next best joint. That is, the truss manufacture computing device 12

implements a preference or bias for joints with more parts that are already in place. If the conditions of block 8116 are not satisfied, the truss manufacture computing device 12 may still designate the selected part as the next best part if the selected part has more perimeter parts within a defined reach distance (e.g., a distance defined in memory 592, which in at least some embodiments may be set in the assembler parameter data from block 7010 or otherwise associated with an internal model (e.g., digital twin) of the corresponding component (e.g., robot 214, 216) of the automated system 10), as indicated in block 8118. In the illustrative embodiment, the truss manufacture computing device 12 implements a preference to clamp the perimeter part with the robot 216, and utilizes a reach distance of 37 inches. In block 8120, the truss manufacture computing device 12 may determine that the selected joint is the next best joint if the selected joint is closer to the start point of the truss than the current next best joint. That is, the truss manufacture computing device 12 may implement a bias or preference for joints that are closer to the start point and increase the likelihood that the assembly robots 214, 216 will operate on the truss from one end to the other (e.g., minimizing movement), rather than repeatedly moving back and forth along the length of the truss.

**[00211]** As indicated in block 8122, the truss manufacture computing device 12 may determine that the selected joint is the next best joint if the selected joint has more total parts than the current next best joint. In doing so, the truss manufacture computing device 12 applies a preference for joints that have more total parts in a final joint (e.g., not merely parts that are already placed). The truss manufacture computing device 12 may determine that the selected joint is the next best joint if the select joint has a higher (e.g., greater) total length of parts than the current next best joint, as indicated in block 8124. That is, the truss manufacture computing device 12 may implement a preference for joints with higher (e.g. greater) total lengths of all parts in the joint. After block 8108, the method 8100 loops back to block 8102 to determine whether additional (e.g., unadded) joints remain, and, if so, executes the operations discussed above for another joint to determine if that joint should be deemed the next best joint.

**[00212]** Referring now to FIG. 82, the truss manufacture computing device 12 may execute a method 8200 for determining the order of parts for the determined next best joint. The method 8200 corresponds to block 8008 of the method 8000. The method 8200, in the illustrative embodiment, begins with block 8202, in which the truss manufacture computing device 12 selects a part from the list of remaining parts for the selected joint (e.g., the determined next best

joint). Subsequently, in block 8204, the truss manufacture computing device 12 sets a part score for the selected part as a function of a set of part score factors. The truss manufacture computing device 12 may initially set the part score for the part to zero, then adjust the part score as a function of the determinations made within block 8204. In doing so, the truss manufacture computing device 12 may increase the part score as a function of whether the selected part touches other parts that are already in the selected joint, as indicated in block 8206. The truss manufacture computing device 12 may determine whether the part touches other parts by comparing coordinates associated with the geometries of the parts. In some embodiments, the truss manufacture computing device 12 may increase the part score of the selected part based on the number of parts that are touched by the selected part (e.g., increasing the part score by one for each part that is touched by the selected part). As such, the truss manufacture computing device 12, in operation, applies a bias to place a part that already touches one or more other parts. Additionally or alternatively, the truss manufacture computing device 12 may increase the part score if the select part is a perimeter part, as indicated in block 8210. The truss manufacture computing device 12 may also increase the part score if the selected part is not a wedge, as indicated in block 8212. As such, the truss manufacture computing device 12, in at least some embodiments, applies a bias or preference for perimeter parts and non-wedge parts.

**[00213]** Subsequently, in block 8214, the truss manufacture computing device 12 determines whether remaining parts are available in the joint (e.g., that have not yet been assigned a part score). If so, the method loops back to block 8202 to select the next part. Otherwise, the method 8200 advances to block 8216, in which the truss manufacture computing device 12 adds the highest scoring part (e.g., the part with the highest part score) to a list of parts for the selected joint (e.g., the next best joint). The method 8200 advances to block 8218 in which the truss manufacture computing device 12 determines whether parts remain to be analyzed. If so (e.g., if not all of the parts have been added to the list referenced in block 8216), the method 8200 loops back to block 8202 to select the next remaining part for analysis (e.g., assignment of a part score). Otherwise (e.g., if no parts remain), the method 8200 advances to block 8220, in which the truss manufacture computing device 12 outputs (e.g., writes to memory 592) the order of parts (e.g., the list of parts associated with block 8008) for the selected joint (e.g., the next best joint).

**[00214]** Referring now to FIG. 83, the truss manufacture computing device 12 may

execute a method 8300 for determining the set of assembly operations for an ordered set of joints. The method 8300 corresponds to block 7110 of the method 7100 described above. The method 8300, in the illustrative embodiment, begins with block 8302 in which the truss manufacture computing device 12 selects the next joint from an ordered set of joints (e.g., the ordered set of joints determined in block 7104 of the method 7100) in the truss. Subsequently, in block 8304, the truss manufacture computing device 12 determines assembly operations for the selected joint as a function of whether the joint has two parts in it. Other joints may have already placed parts for the selected joint. If the joint has two parts, the truss manufacture computing device 12 determines a set of assembly operations as a function of whether the joint has a bottom plate, as indicated in block 8306. If the joint does have a bottom plate and the joint has all parts, the truss manufacture computing device 12 determines to add a top plate for the joint as a new assembly operation, as indicated in block 8308. If the joint does not have a bottom plate, the truss manufacture computing device 12, in the illustrative embodiment, determines a set of corresponding assembly operations, as indicated in block 8310. In doing so, in response to a determination that the selected joint does not have all parts, the truss manufacture computing device 12 determines to add a bottom plate for the selected joint as a new assembly operation, as indicated in block 8312. If the joint does have all parts, the truss manufacture computing device 12 determines to add a top plate and a bottom plate for the joint as a new assembly operation, as indicated in block 8314.

**[00215]** Subsequently, in block 8316, for each part in the joint, the truss manufacture computing device 12 adds one or more assembly operations for a new part. An embodiment of a method 8400 that may be executed by the truss manufacture computing device 12 to add assembly operations for new parts is described with reference to FIG. 84. In block 8318, the truss manufacture computing device 12 determines whether any remaining joints are present in the ordered set, for which assembly operations have not been determined. If so, the method 8300 loops back to block 8302 to select another joint from the ordered set and determine assembly operations for that selected joint. Otherwise, the method 8300 advances to block 8320 in which the truss manufacture computing device 12 outputs (e.g., writes to memory 592) the list of assembly operations for all joints in the ordered set of joints.

**[00216]** Referring now to FIG. 84, the truss manufacture computing device 12 may execute a method 8400 for adding assembly operations for new parts in a joint. The method 8400

corresponds to block 8316 of the method 8300. In the illustrative embodiment, the method 8400 begins with block 8402 in which the truss manufacture computing device 12 selects a part for the selected joint. In block 8404, the truss manufacture computing device 12 selects a next other joint (e.g., a joint of the truss other than the selected joint). In block 8406, the truss manufacture computing device 12 determines a set of one or more assembly operations as a function of whether the selected part (e.g., from block 8402) is the third part of any other joint (e.g., the other joint selected in block 8404). In operation, the truss manufacture computing device 12 implements a bias towards starting the other joint if possible, as doing makes the partially completed truss more rigid. In response to a determination that the selected part is a third part in another joint, the trust manufacture computing device 12 determines a set of assembly operations as a function of whether the other joint has a bottom plate, as indicated in block 8408. In doing so, in response to a determination that the other joint has a bottom plate and the other joint has all parts, the truss manufacture computing device 12 may add a new assembly operation to add the top plate for the other joint, as indicated in block 8410. The truss manufacture computing device 12, in response to a determination that the other joint does not have a bottom plate, may determine a set of assembly operations as a function of whether the other joint has all parts, as indicated in block 8412. In doing so, the truss manufacture computing device 12, in response to a determination that the other joint does not have all parts, may add an assembly operation to add the bottom plate for the other joint, as indicated in block 8414. Conversely, in response to a determination that the other joint does have all parts, the truss manufacture computing device 12 may determine to add an assembly operation to add a top plate and a bottom plate to the other joint, as indicated in block 8416. In block 8418, the truss manufacture computing device 12 determines whether there are any remaining other joints. If so, the method 8400 loops back to block 8404 to select one of the remaining other joints and repeat the operations described above for determining corresponding assembly operations. Otherwise, the method 8400 advances to block 8420 of FIG. 85, in which the truss manufacture computing device 12 determines assembly operations as a function of whether the selected joint has a bottom plate.

**[00217]** Referring now to FIG. 85, in determining the assembly operations, the truss manufacture computing device 12 may, in response to a determination that the selected joint does not have a bottom plate, add a new assembly operation to add the bottom plate and the selected part for the selected joint, as indicated in block 8422. In response to a determination that

the selected joint does have a bottom plate, the truss manufacture computing device 12 may determine a set of assembly operations as a function of whether the selected part is the last part in the selected joint, as indicated in block 8424. In doing so, and as indicated in block 8426, in response to a determination that the selected part is the last part in the selected joint, the truss manufacture computing device 12 may add an assembly operation to add the selected part and top plate for the selected joint, as indicated in block 8426. Conversely, in response to a determination that the selected part is not the last part in the selected joint, the truss manufacture computing device 12 may add an assembly operation to add the selected part to the selected joint (e.g., without adding the top plate), as indicated in block 8428. In block 8430, the truss manufacture computing device 12 determines the subsequent course of action based on whether remaining parts exist. If so, the method 8400 loops back to block 8402, in which the truss manufacture computing device 12 selects another part from the set of remaining parts and repeats the operations of the method 8400 described above with respect to the newly selected part. Otherwise, the method 8400 advances to block 8432, in which the trust manufacture computing device 12 outputs (e.g., writes to the memory 592) the set (e.g., list) of assembly operations for the selected joint.

**[00218]** Referring now to FIG. 86, the truss manufacture computing device 12 may execute a method 8600 for calculating pickup points (e.g., for a selected part). The method 8600, in the illustrative embodiment, begins with block 8602, in which the truss manufacture computing device 12 calculates a closest pickup point that does not interfere with the press (e.g., the press 180). In doing so, the truss manufacture computing device 12 may identify an area around the press 180, including a trajectory of the press 180, that may not be entered by a robot 214, 216 and exclude from a set of available pickup points, any pickup points within that identified area. In block 8604, the truss manufacture computing device 12 determines, for each direction from the center of the selected part, a set of valid pickup points. In doing so, and as indicated in block 8606, the truss manufacture computing device 12 obtains (e.g., reads from the memory 592) a pickup options list. In doing so, and as indicated in block 8608, the truss manufacture computing device 12 may obtain a pick options list in which a first option is to utilize a regular clamp (e.g., one or more clamps 234 of the gripper 232) to pick up the selected part. As indicated in block 8610, the truss manufacture computing device 12 may obtain a pickup options list in which second, third, and fourth options are only calculated if valid solutions (e.g.,

for picking up the selected part) are not found for both a standard direction and an inverse direction for the regular clamp (e.g., one or more of the clamps 234 of the gripper 232).

**[00219]** The truss manufacture computing device 12 may obtain a pickup options list in which the second option utilizes a regular vacuum (e.g., the suction pad 238) with no other movement required (e.g., to pick up the selected part), as indicated in block 8612. As indicated in block 8614, the truss manufacture computing device 12 may obtain a pickup options list in which the third and fourth options are only calculated for operations in which a new part is being added. In some embodiments, the truss manufacture computing device 12 may obtain a pickup options list in which the third option is to pick up the selected part from the center utilizing the vacuum (e.g., the suction pad 238), move the press 180 out of the way, place the part with the vacuum (e.g., the suction pad 238), and move the press 180 back into place, as indicated in block 8616. The truss manufacture computing device 12 may obtain a pickup options list in which the fourth option is similar to the third option described above, except the fourth option involves picking up the selected part from an edge of the part, rather than the center of the part, as indicated in block 8618. Further, and as indicated in block 8618, the truss manufacture computing device 12 performs a set of determinations for each pickup option to produce valid solutions (e.g., for picking up) for half of the selected part. In doing so, the truss manufacture computing device 12 may perform determinations for each half inch increment from the center of the selected part to an edge of the part, to produce valid solution(s) for a current pickup option, as indicated in block 8622. An embodiment of a method 8700 for performing the determinations of block 8620 is described with reference to FIG. 87. An embodiment of a method 8800 for performing the determinations of block 8622 is described with reference to FIG. 88. In block 8624, the truss manufacture computing device 12 outputs (e.g., writes to the memory 592) valid solutions (e.g., for picking up) for the full part.

**[00220]** Referring now to FIG. 87, the truss manufacture computing device 12 may execute a method 8700 for performing determinations for each pickup option (e.g., from block 8606 of the method 8600) to produce valid solutions for half of a selected part. The method 8700 corresponds to block 8620 of the method 8600. In the illustrative embodiment, the method 8700 begins with block 8702 in which the truss manufacture computing device 12 selects the next available pickup option from the pickup options list (e.g., from block 8606 of the method 8600). Subsequently, in block 8704, the truss manufacture computing device 12 calculates a current tool

interference polygon. That is, the truss manufacture computing device 12 calculates a polygon for the gripper 232 (which includes one or more clamps 234) or the vacuum (e.g., suction pad 238) based on the corresponding pickup option (e.g., whether the pickup option involves the gripper 232 or the vacuum (e.g., suction pad 238)). In the illustrative embodiment, the truss manufacture computing device 12 calculates the polygons for both the standard and inverse directions and the polygon, in the illustrative embodiment, is a rectangle. In performing geometric operations (e.g., calculations of polygons, interference, etc.), the truss manufacture computing device 12 may utilize a shared package of executable instructions (e.g., in a microservices architecture) defining functions for performing the geometric operations.

**[00221]** Continuing the method 8700, the truss manufacture computing device 12 may calculate a tool interference polygon for the other tool (e.g., the tool other than the one associated with block 8704). For example, if the truss manufacture computing device 12 calculated a polygon for the gripper 232 (e.g., which includes one or more clamps 234) in block 8704, the truss manufacture computing device 12 may calculate the interference polygon for the vacuum (e.g., suction pad 238) in block 8706. Similar to block 8704, in block 8706, the truss manufacture computing device 12 calculates the interference polygon for both directions (e.g., standard and inverse) and utilizes a rectangle shape for the polygon. In block 8708, the truss manufacture computing device 12 calculates a tool center polygon (e.g., the center of the tool 228). In doing so, the truss manufacture computing device 12, in the illustrative embodiment, calculates the polygon as an octagonal shape. In block 8710, the truss manufacture computing device 12 calculates a "sausage" interference polygon. In doing so, in the illustrative embodiment, the truss manufacture computing device 12 combines the polygons from blocks 8704, 8706, 8708 together. The resulting polygon, in the illustrative embodiment, may be shaped as a cylinder with hemispherical ends.

**[00222]** In block 8712, the truss manufacture computing device 12 may calculate an effective tool pickup area. In doing so, the truss manufacture computing device 12 may determine an area of the gripper 232 (which includes clamps 234) or vacuum (e.g., suction pad 238) used to ensure sufficient interference with a part (e.g., a defined amount of interference) to enable a valid (e.g., successful) pickup of the part. In the illustrative embodiment, the truss manufacture computing device 12 calculates the effective tool pickup area as a rectangle. In block 8714, the truss manufacture computing device 12 calculates a polygon representative of

interference of the current tool with the part. In doing so, the truss manufacture computing device 12 may calculate the interference as a rectangular shape. As indicated in block 8716, the truss manufacture computing device 12, in the illustrative embodiment, performs determinations for each half inch increment from the center to edge(s) of the part to produce valid solutions for half of the part for the selected pickup options. A detailed description of a method 8800 corresponding to block 8716 is provided in connection with FIG. 88. In block 8718, the truss manufacture computing device 12 determines the subsequent course of action based on whether remaining options (e.g., pickup options) exist. If so, the method 8700 loops back to block 8702, in which the truss manufacture computing device 12 selects the next available pickup option and repeats the remaining operations of the method 8700 for that selected pickup option. Otherwise, the method 8700 advances to block 8720, in which the truss manufacture computing device 12 outputs (e.g., writes to the memory 592) the valid solutions for half of the selected part.

**[00223]** Referring now to FIG. 88, in operation, the truss manufacture computing device 12 may execute a method 8800 for performing determinations for each half inch increment from the center to one or more edges of a selected part to produce valid pickup solutions for half of the part for a selected pickup option. The method 8800 corresponds to block 8622 of the method 8600 and block 8716 of the method 8700. In the illustrative embodiment, the method 8800 begins with block 8802, in which the truss manufacture computing device 12 selects the next available increment of the part. In doing so, the truss manufacture computing device 12, in the illustrative embodiment, selects the next available half inch increment along a length from the center of the part to an edge of the part, as indicated in block 8804. In block 8806, the truss manufacture computing device 12 calculates final tool position interference. That is, the truss manufacture computing device 12 calculates the final position of the tool 228 and whether and to what extent that final tool position interferes with a final position of the press 180 and/or the selected part (e.g., utilizing a shared package of executable instructions for geometric operations and shape data for the components). Subsequently, in block 8808, the truss manufacture computing device 12 calculates tool trajectory interference. In doing so, the truss manufacture computing device 12 may determine the trajectory of the tool 228 from initial to final position interfering with the final position of the press 180 and/or part location(s). In block 8810, the truss manufacture computing device 12 calculates press trajectory interference. That is, the truss manufacture computing device 12 may determine the trajectory of the press 180 from initial to

final position interfering with the final position of the tool 228 and/or the trajectory of the tool 228 and/or part location(s).

**[00224]** In block 8812, the truss manufacture computing device 12 determines subsequent operations as a function of the calculated interferences (e.g., from blocks 8806, 8808, 8810). In doing so, and as indicated in block 8814, the truss manufacture computing device 12 may determine whether utilizing the standard direction of movement (e.g., of the tool 228) will avoid interferences. In response to a determination that utilizing the standard direction of movement will avoid interferences, the truss manufacture computing device 12 adds the standard direction as a valid solution for the selected increment (e.g. half inch increment) along the part, as indicated in block 8816. As indicated in block 8818, the truss manufacture computing device 12 may determine whether the inverse direction of movement (e.g., of the tool 228) will avoid interferences. In response to a determination that utilizing the inverse direction of movement will avoid interferences, the truss manufacture computing device 12 may add the inverse direction as a valid solution for the selected increment, as indicated in block 8820. The operations associated with blocks 8814, 8816, 8818, 8820, in at least some embodiments, may eliminate complexities in the coordination of movements of the components of the automated system 10, by reducing the likelihood that the tool 228 will interfere with the press 180. If neither direction of movements (e.g., standard or inverse) will eliminate interferences, the truss manufacture computing device 12 may determine to move the press 180 out of the way of the tool 228 to enable the pickup operation at the selected increment. In block 8822, the truss manufacture computing device 12 may determine whether interferences will be present for more than one increment (e.g., more than one half inch increment) along the part. In doing so, the truss manufacture computing device 12 may set the next available increment as multiple increments from the selected increment (e.g., if interferences will exist over more than one increment), as indicated in block 8824. By setting the next available increment as a multiple of increments from the presently selected increment, the truss manufacture computing device 12 reduces the compute resources (e.g., time, energy, processor cycles) that would otherwise be consumed by the truss manufacture computing device 12 in performing interference calculation operations for each of the intervening increments (e.g., the increments to be skipped over).

**[00225]** To further enhance the efficiency of the operations, and as indicated in block 8826, the truss manufacture computing device 12 may determine not to check additional

increments (e.g., for interference) if one or more valid solutions exist for standard and inverse directions (e.g., of the tool 228) and there is no determined interference with the press 180 or the trajectory of the press 180. By analyzing the increments from the center of the part outwards in increments and by ending the analysis when a valid solution has been identified (e.g., rather than performing the analysis for further increments), the truss manufacture computing device 12 implements a preference for pickup points that are closer to a joint rather than further away (e.g., thereby reducing movements of the components of the automated system 10 and the time required for production of the truss). Referring briefly to FIG. 89, the method 8800 continues to block 8828 in which the truss manufacture computing device 12 determines the subsequent course of action based on whether remaining increments of the part are available to be analyzed (e.g., for pickup points). In response to a determination that additional increments are available to be analyzed, the method 8800 loops back to block 8802 of FIG. 88, in which the truss manufacture computing device 12 selects the next available increment and repeats the remaining operations of the method 8800 described above. Otherwise, the method 8800 advances to block 8830, in which the truss manufacture computing device 12 outputs (e.g., writes to memory 592) the valid solution(s) for half of the part for the selected pickup option.

**[00226]** Referring now to FIG. 90, the truss manufacture computing device 12 may perform a method 9000 for calculating a set of valid primary robot pickup options. The method 9000 corresponds to block 7202 of the method 7200. In the illustrative embodiment, the method 9000 begins with block 9002, in which the truss manufacture computing device 12 calculates pickup points for the selected part. In doing so, the truss manufacture computing device 12, in the illustrative embodiment, executes the method 8600 described with reference to FIG. 86 above. Afterwards, the method 9000 advances to block 9004, in which the truss manufacture computing device 12 determines the subsequent course of action based on whether a robot 214, 216 (e.g., the robot assigned to the primary robot role) is picking up a new part for the wooden structure (e.g., truss). If not, the method 9000 advances to block 9006, in which the truss manufacture computing device 12 selects an available other joint from a set of one or more other joints. In block 9008, the truss manufacture computing device 12 determines whether the selected other joint is within a defined distance (e.g., 48 inches) of the present joint and is connect to the part to be picked up. If so, the truss manufacture computing device 12 calculates, for each other part in the selected other joint, a set of pickup points (e.g., by executing the

method 8600 for each of the other parts in the selected other joint). In doing so, and as indicated in block 9010, the truss manufacture computing device 12, in the illustrative embodiment, adds the calculated pickup points to a set of valid primary pickup points for the assembly operation. The added pickup points represent alternate pickup point operations in case the earlier-calculated pickup options lead to interference (e.g., between the primary robot and the press 180).

**[00227]** As indicated in block 9012, in the illustrative embodiment, the method 9000 loops back to block 9006 to select the next available other part and perform the operations in blocks 9008, 9010 for each of the available other joints. After the truss manufacture computing device 12 has performed the operations of blocks 9006, 9008, 9010 for all of the available other joints or if the truss manufacture computing device 12 determined in block 9004 that a robot 214, 216 (e.g., the robot assigned to the primary robot role) is not picking up a new part, the method 9000 advances to block 9014. In block 9014, the truss manufacture computing device 12 outputs (e.g., writes to the memory 592) the set of all valid primary robot pickup options for the selected assembly operation.

**[00228]** Referring now to FIG. 91, the truss manufacture computing device 12 may execute a method 9100 for calculating valid secondary robot pickup options. The method 9100 corresponds with block 7204 of the method 7200, described above with reference to FIG. 72. The method 9100, in the illustrative embodiment, begins with block 9102, in which the truss manufacture computing device 12 analyzes the set of primary robot pickup options (e.g., from block 7202 of the method 7200, and as determined through execution of the method 9000 described above with reference to FIG. 90) to determine whether one or more primary robot pickup options remain (e.g., to be utilized in the method 9100). In block 9104, the truss manufacture computing device 12 selects the next primary robot pickup option from the set. In block 9106, the truss manufacture computing device 12 determines whether any other parts already in the selected joint remain (e.g., that have not been analyzed by the truss manufacture computing device 12 in the execution of the method 9100 in connection with the selected primary robot pickup option). In response to a determination that at least one other part already in the selected joint remains to be analyzed, the truss manufacture computing device 12 selects one of those other parts in block 9108. Subsequently, in block 9110, the truss manufacture computing device 12 determines a set of secondary robot pickup options for the selected other part that is already in the selected joint. In doing so, and as indicated in block 9112, the truss

manufacture computing device 12 calculates pickup points for the selected other part (e.g., selected in block 9108), such as by executing the method 8600, described above.

**[00229]** In block 9114, the truss manufacture computing device 12 adds, in response to a determination that a new part is being picked up, pickup points to a set of valid secondary pickup points for the selected other part. In the illustrative embodiment, the truss manufacture computing device 12 performs the operations of blocks 9108, 9110, 9112, 9114 for each other part that is already in the joint. Afterwards, the method 9100 advances to block 9116, in which the truss manufacture computing device 12 adds the valid secondary pickup points (e.g., determined through the above operations) to the primary pickup point (e.g., primary pickup option) that was selected in block 9104. In the illustrative embodiment, the truss manufacture computing device 12 performs the above operations of the method 9100 for each primary robot pickup option (e.g., primary robot pickup point) in the set. Afterwards, the method 9100 advances to block 9118, in which the truss manufacture computing device 12 outputs (e.g., writes to the memory 592) all of the valid primary and secondary robot pickup options (e.g., pickup points) for the present assembly operation (e.g., the assembly operation for which recipe operations are determined in the method 7200).

**[00230]** Referring now to FIG. 92, the truss manufacture computing device 12 may execute a method 9200 for assigning primary and secondary roles to the robots 214, 216. The method 9200 corresponds with block 7210 of the method 7200. In block 9202, the truss manufacture computing device 12 assigns roles as a function of a set of role assignment factors. In doing so, and as indicated in block 9204, the truss manufacture computing device 12 may assign roles as a function of whether a new part is being added in the current (e.g., selected) assembly operation (e.g., the current assembly operation associated with the method 7200). In doing so, and as indicated in block 9206, the truss manufacture computing device 12 may assign, in response to a determination that a new part is being added, the primary role to the first robot (e.g., the robot 214) and the secondary role to the second robot (e.g., the robot 216). As indicated in block 9208, the truss manufacture computing device 12 may assign roles as a function of whether only one robot is required for the current assembly operation. In doing so, and as indicated in block 9210, the truss manufacture computing device 12 may assign, in response to a determination that only one robot is required for the assembly operation, the primary role to the second robot (e.g., the robot 216) and not assign the secondary role to any robot (e.g., as no

secondary robot role is required).

**[00231]** In block 9212, the truss manufacture computing device 12 may assign roles as a function of robot tool 228 locations. In doing so, and as indicated in block 9214, the truss manufacture computing device 12 may assign the primary role to the first robot (e.g., the robot 214) and the secondary role to the second robot (e.g., the robot 216) in response to a determination that, for the current assembly operation, the x coordinate of the center of the tool 228 for the primary robot is less than the x coordinate of the center of the tool 228 for the secondary robot. Otherwise, the truss manufacture computing device 12, in the illustrative embodiment reverses the roles, assigning the primary role to the second robot (e.g., the robot 216) and assigning the secondary role to the first robot (e.g., the robot 214). Subsequently, in block 9218, the truss manufacture computing device 12 outputs (e.g., writes to the memory 592) data indicative of the roles assigned to the robots 214, 216.

**[00232]** Referring now to FIG. 93, in operation, the truss manufacture computing device 12 may execute a method 9300 for calculating first robot (e.g., robot 214) and second robot (e.g., robot 216) gantry (e.g., carriage 218) and arm (e.g., arm 220) positions. The method 9300 corresponds with block 7212 of the method 7200 described above with reference to FIG. 72. In the illustrative embodiment, the method 9300 begins with block 9302 in which the truss manufacture computing device 12 calculates a polygon for the arm (e.g., arm 220) of the second robot (e.g., the robot 216). In doing so, the truss manufacture computing device 12 may utilize a defined value indicative of the distance from an edge of a servo motor of the second robot 216 to the tool (e.g., tool 228) of the second robot 216, as indicated in block 9304. In some embodiments, the truss manufacture computing device 12 utilizes a defined distance of 39 inches, as indicated in block 9306. As indicated in block 9308, the truss manufacture computing device 12 may utilize a lower threshold distance (e.g., a lower limit) indicative of a distance from the gantry to the servo motor section. In doing so, the truss manufacture computing device 12 may utilize a lower threshold distance (e.g., lower limit) of 19 inches, as indicated in block 9310. In the illustrative embodiment, the truss manufacture computing device 12 calculates the arm polygon for the second robot 216 as only a temporary final arm polygon, to determine the final location for the first robot 214. Further, in the illustrative embodiment, the second robot 216 moves into position before the first robot 214 moves into position for an assembly operation.

**[00233]** In block 9312, the truss manufacture computing device 12, in the illustrative

embodiment, calculates press displacement (e.g., displacement of the press 180). An embodiment of a method 9400 that the truss manufacture computing device 12 may execute to calculate press displacement is described with reference to FIG. 94. Further, in block 9314, the truss manufacture computing device 12 calculates a first robot trajectory (e.g., a trajectory of the first robot 214) and a second robot trajectory (e.g., a trajectory of the second robot 216). An embodiment of a method 9600 that may be executed by the truss manufacture computing device 12 for calculating the trajectories of the first robot 214 and the second robot 216 is described with reference to FIG. 96. Still referring to FIG. 93, in block 9316, the truss manufacture computing device 12 may calculate the first robot gantry position and the second robot gantry position. An embodiment of a method 9700 that the truss manufacture computing device 12 may execute to calculate the gantry positions for the first robot 214 and the second robot 216 is described with reference to FIG. 97. In block 9318, the truss manufacture computing device outputs (e.g., writes to the memory 592) the first robot gantry (e.g., carriage 218) and second robot gantry (e.g., carriage 218) y positions. In block 9320, the truss manufacture computing device 12 may calculate, as a function of the gantry y position and final tool position data, an orientation for the arm 220 of each robot 214, 216. Afterwards, the truss manufacture computing device 12 outputs (e.g., writes to the memory 592) the first robot 214 and second robot 216 gantry (e.g., carriage 218) and arm (e.g., arm 220) positions.

**[00234]** Referring now to FIG. 94, in operation, the truss manufacture computing device 12 may execute a method 9400 for calculating press displacement. The method 9400 corresponds with block 9312 of the method 9300 described above. In the illustrative embodiment, the method 9400 begins with block 9402, in which the truss manufacture computing device 12 determines whether the vacuum (e.g., suction pad 238) will be used to add a new part. In the illustrative embodiment, the press 180 is only moved (e.g., displaced) in connection with an assembly operation when the gripper 232 (e.g., rather than the vacuum (e.g., suction pad 238)) is used by a robot 214, 216 to hold a part. Accordingly, in response to a determination that the vacuum (e.g., suction pad 238) will be used, the method 9400 branches to block 9404, in which the truss manufacture computing device 12 determines that the press 180 will not be moved (e.g., displaced). Otherwise, the method 9400 advances to block 9406 in which the truss manufacture computing device 12 determines whether the final press position (e.g., final position of the press 180) will interfere with the final location of the second robot 216 (e.g., using polygons

representative of the press 180 and second robot 216, and geometric operations to test for interference of the polygons). In block 9408, the truss manufacture computing device 12 determines the subsequent course of action based on whether the final press position will interfere with the final location of the second robot 216. If not, the method 9400 advances to block 9410, in which the truss manufacture computing device 12 determines whether the final press 180 position is in the positive y direction from the current press 180 position. In block 9412, the truss manufacture computing device 12 determines the subsequent course of action based on whether the final position of the press 180 will be in the positive y direction compared to the current position of the press 180. If not, the method 9400 advances to block 9414 in which the truss manufacture computing device 12 moves the press 180 a defined distance in the –y direction. In doing so, in the illustrative embodiment, the truss manufacture computing device 12 may move the press 180 a distance of 1550 millimeters in the –y direction, as indicated in block 9416. If, on the other hand, the final position of the press 180 will be in the +y direction from the current position of the press 180, the method 9400 instead advances to block 9418, in which the truss manufacture computing device 12 moves the press 180 a defined distance in the +y direction. In doing so, and as indicated in block 9420, the truss manufacture computing device 12 may move the press 180 a distance of 1550 millimeters in the +y direction.

**[00235]** Referring back to block 9408, if the truss manufacture computing device 12 instead determines that interference will occur, the method 9400 advances to block 9428 of FIG. 95 in which the truss manufacture computing device 12 determines whether moving the press 180 an extra distance in the positive y direction will reduce interference with the second robot 216. In block 9430, the truss manufacture computing device 12 determines the subsequent course of action based on whether moving the press 180 in the positive y direction will reduce interference with the second robot 216. If not, the method 9400 advances to block 9432, in which the truss manufacture computing device 12 moves the press 180 a defined distance in the negative y direction. In doing so, the truss manufacture computing device 12 may move the press 180 a distance of 1000 millimeters in the negative y direction, as indicated in block 9434. Otherwise, if moving the press an extra distance in the positive y direction will not reduce interference, the method 9400 instead branches to block 9436, in which the truss manufacture computing device 12 moves the press 180 a defined distance in the positive y direction. In doing so, the truss manufacture computing device 12 may move the press 180 a distance of 1000

millimeters in the positive y direction, as indicated in block 9438.

**[00236]** Referring back to FIG. 94, after potentially moving the press 180 in accordance with the determinations of block 9408, 9412, and/or 9430, the method 9400 advances to block 9422 in which the truss manufacture computing device 12 may move the press 180 as a function of whether the press 180 is outside of a set of limits associated with the press 180. In doing so, and as indicated in block 9424, the truss manufacture computing device 12 may selectively move the press 180 to an upper or lower bound (e.g., if the press 180 is outside of those bounds). That is, the truss manufacture computing device 12 may move the press 180 to the bound (e.g., upper or lower bound) that is the closest to the current position of the press 180. In doing so, the truss manufacture computing device 12 may utilize an upper bound of 4550 millimeters in the positive y direction and utilize a lower bound of -1260 in the negative y direction, as indicated in block 9426.

**[00237]** Referring now to FIG. 96, the truss manufacture computing device 12 may execute a method 9600 for calculating trajectories for the first robot 214 and the second robot 216. The method 9600 corresponds with block 9314 of the method 9300 described with reference to FIG. 93. The truss manufacture computing device 12 may execute the method separately for each robot 214, 216. In the illustrative embodiment, the method 9600 begins with block 9602 in which the truss manufacture computing device 12 determines whether the robot 214, 216 can approach a final position without interference (e.g., without colliding with another object). In making the determination in block 9602, the truss manufacture computing device 12 may check for interference between the robot 214, 216 and the press 180 (e.g., based on the final position and trajectory). Further, if the robot in question is the first robot 214, the truss manufacture computing device 12 may also check for interference with the second robot 216 (e.g., based on final position and trajectory). Additionally, if the robot in question is the first robot 214, the truss manufacture computing device 12 may check for potential interference with another object using the part being carried by the robot 214 (e.g., adding the dimensions of the part to a polygon representing the robot 214 for use in interference detection). In block 9604, the truss manufacture computing device 12 determines the subsequent course of action based on whether interference is predicted in block 9602.

**[00238]** If the truss manufacture computing device 12 determined that interference will not occur, the method 9600 advances to block 9606 in which the truss manufacture computing

device 12 determines to move the robot 214, 216 in a straight line to the final position. Otherwise, in response to a determination that the robot 214, 216 cannot approach the final position without interference, the method 9600 advances to block 9608, in which the truss manufacture computing device 12 determines whether the current position of the robot 214, 216 is higher than the final position for the robot 214, 216 in the y direction. In block 9610, the truss manufacture computing device 12 determines the subsequent course of action based on whether the current position of the robot 214, 216 is higher than the final position of the robot 214, 216. If so, the method 9600 advances to block 9612, in which the truss manufacture computing device 12 determines to move the robot 214, 216 in a straight line in the x direction to position the robot 214, 216 directly above the final position in the y direction. As indicated in block 9614, the truss manufacture computing device 12, in the illustrative embodiment, additionally determines to move the robot 214, 216 in the y direction down to the final position for the robot 214, 216. Referring back to block 9610, in response to a determination that the current position of the robot 214, 216 is not higher than the final position of the robot 214, 216, the method 9600 instead branches to block 9616, in which the truss manufacture computing device 12 determines to move the robot 214, 216 in a straight line in the x direction to position the robot directly below the final position in the y direction. Further, in the illustrative embodiment, the truss manufacture computing device 12 determines to move the robot 214, 216 in the y direction up to the final position for the robot 214, 216. If the robot in question is the first robot 214, the truss manufacture computing device 12, in the illustrative embodiment, may determine whether the first robot 214 can fit between the press 180 and the second robot 216 (e.g., based on whether polygons representing the components would interfere). If not, the truss manufacture computing device 12 may determine to move the first robot 214 around the second robot 216 (e.g., to avoid collision(s)).

**[00239]** Referring now to FIG. 97, the truss manufacture computing device 12 may execute a method 9700 for calculating gantry (e.g., carriage 218) positions for the first robot 214 and the second robot 216. The method 9700 corresponds with block 9316 of the method 9300 described above with reference to FIG. 93. The truss manufacture computing device 12 may execute the method 9700 for each robot 214, 216 separately. In the illustrative embodiment, the method 9700 begins with block 9702, in which the truss manufacture computing device 12 calculates gantry y position zones. In doing so, and as indicated in block 9704, the truss

manufacture computing device 12 may utilize three possible gantry y position zones, including near, far, and dead. Further, and as indicated in block 9706, the truss manufacture computing device 12 applies a bias (e.g., a preference) against utilization of the dead zone (e.g., a zone designated as such due to potential difficulty in positioning robotic components there) to maintain efficient maneuverability of the corresponding gantry (e.g., carriage 218) and/or components attached thereto.

**[00240]** Subsequently, in block 9708, the truss manufacture computing device 12, in the illustrative embodiment, calculates, for each zone, a set of possible y positions. In doing so, and as indicated in block 9710, the truss manufacture computing device 12 may calculate, for each of multiple incremental y values in the zone, a set of possible gantry y positions. As indicated in block 9712, in response to a determination that the robot arm 220 will not interfere with anything at the gantry position for the incremental y value, the truss manufacture computing device 12, in the illustrative embodiment, adds that y value to a set (e.g., a list) of valid position for the zone. In block 9714, the truss manufacture computing device 12 outputs (e.g., writes to the memory 592) the valid y positions for the zone. In block 9716, the truss manufacture computing device 12 outputs (e.g., writes to the memory 592) a set (e.g., a list) of valid y position for all zones (e.g., as a result of performing the operations of blocks 9710, 9712, 9714 for all of the zones). In block 9718, the truss manufacture computing device 12 determines whether any points (e.g., valid y positions) are present in the near zone or the far zone. If so, the truss manufacture computing device 12 removes, from the set of valid y positions, any positions in the dead zone, as indicated in block 9720 (e.g., thereby implementing a preference for y positions outside of the dead zone, in order to maintain efficient maneuverability for robotic components). In the illustrative embodiment, the method 9700 advances to block 9722 of FIG. 98, in which the truss manufacture computing device 12 identifies a best y position (e.g., from the set of remaining valid y positions).

**[00241]** Referring now to FIG. 98, in the illustrative embodiment, the truss manufacture computing device 12 applies preferences (e.g., through exclusion of y values that would result in violation of the preferences). In doing so, the truss manufacture computing device 12 may apply a preference for a gantry (e.g., carriage 218) y value that is greater than the y value of the tool 228 of the corresponding robot 214, 216, as indicated in block 9724. As indicated in block 9726, the truss manufacture computing device 12 prevents gantry rotation from crossing over a +x axis

for either robot 214, 216 (e.g., the truss manufacture computing device 12 excludes from the set of possibilities for best y position, any y positions that would result in gantry rotation crossing over the +x axis for either robot 214, 216). Additionally, as indicated in block 9728, the truss manufacture computing device 12 applies a preference to maintain the robot arm 220 in the same direction as the trajectory of the robot 214, 216. Further, and as indicated in block 9730, the truss manufacture computing device 12 applies a preference to maintaining a wait position (e.g., standby position) in the same zone as the current position of the corresponding robot 214, 216.

**[00242]** Referring now to FIG. 99, the truss manufacture computing device 12 may execute a method 9900 for determining whether a selected pickup option pair is the best pickup option pair. The method 9900 corresponds with block 7214 of the method 7200 described with reference to FIG. 72. In the illustrative embodiment, the method 9900 begins with block 9902, in which the truss manufacture computing device 12 determines, as a function of evaluation factors, whether the selected pickup option pair (e.g., from block 7214 of the method 7200) is the best pickup option pair. In doing so, the truss manufacture computing device 12 identifies, as problems (e.g., conditions that, if caused by a given pickup option pair, detract from the pickup option pair being designated as the best pickup option pair), interference between components and/or items out of reach, as indicated in block 9904.

**[00243]** As indicated in block 9906, the truss manufacture computing device 12 may determine, as a function of robot and gantry evaluation factors, whether the selected pickup option pair is the best pickup option pair. In doing so, the truss manufacture computing device 12 may determine that the selected pickup option pair is the best pickup option pair if the selected pickup option pair has less significant robot or gantry problem(s) than the current best pickup option pair, as indicated in block 9908. In making that determination, the truss manufacture computing device 12 may establish a preference order of no problem (e.g., most preferable), a problem with the first robot 214 (e.g., less preferable), and a problem with the second robot 216 (e.g., least preferable), as indicated in block 9910. If the truss manufacture computing device 12 determines that the selected pickup option pair is the best pickup option pair, the method 9900 ends at that point. Otherwise, the method 9900 continues, with the truss manufacture computing device 12 performing one or more additional operations to determine whether the selected pickup option pair is the best pickup option pair, as described herein.

**[00244]** In continuing the method 9900, the truss manufacture computing device 12 may

determine, as a function of robot arm evaluation factors, whether the selected pickup option is the best pickup option, as indicated in block 9912. In doing so, the truss manufacture computing device 12 may determine that the selected pickup option pair is the best pickup option pair if the selected pickup option pair has similar but less significant robot arm problem(s) than the current best pickup option pair, as indicated in block 9914. As indicated in block 9916, the truss manufacture computing device 12 may establish a preference order of no problem (e.g., most preferable), a problem with the trajectory of the first robot 214 (e.g., less preferable), and a problem with the arm 220 of the second robot 216 (e.g., least preferable). In the illustrative embodiment, the method 9900 continues in block 9918 of FIG. 100, in which the truss manufacture computing device 12 may determine, as a function of vacuum utilization evaluation factors, whether the selected pickup option pair is the best pickup option.

**[00245]** Referring now to FIG. 100, the truss manufacture computing device 12 may determine that the selected pickup option pair is the best pickup option pair if the selected pickup option pair does not utilize a vacuum (e.g., the suction pad 238), as indicated in block 9920. The truss manufacture computing device 12 may establish a preference order of no vacuum utilization (e.g., most preferable), utilization of the vacuum (e.g., suction pad 238) by the first robot 214 (e.g., less preferable), and utilization of the vacuum (e.g., suction pad 238) by the second robot 216 (e.g., least preferable), as indicated in block 9922. The truss manufacture computing device 12 may determine, as a function of gantry dead zone utilization factors, whether the selected pickup option is the best pickup option pair, as indicated in block 9924. In doing so, and as indicated in block 9926, the truss manufacture computing device 12 may determine that the selected pickup option pair is the best pickup option pair if the selected pickup option pair does not utilize a gantry dead zone. Similarly, the truss manufacture computing device 12 may establish a preference order of no dead zone utilization (e.g., preferable) and dead zone utilization (e.g., less preferable), as indicated in block 9928. As indicated in block 9930, the truss manufacture computing device 12 may determine whether the selected pickup option pair is the best pickup option pair as a function of press movement evaluation factors. In doing so, the truss manufacture computing device 12 may determine that the selected pickup option pair is the best pickup option pair if the selected pickup option pair does not involve movement of the press 180, as indicated in block 9932. The truss manufacture computing device 12, in some embodiments, may establish a preference order of no movement of the press 180 (e.g., most

preferable, to minimize movements, conserve energy, reduce time to manufacture, etc.), movement of the press 180 for the first robot 214 (e.g., less preferable), and movement of the press 180 for the second robot 216 (e.g., least preferable), as indicated in block 9934.

**[00246]** In the illustrative embodiment, the method 9900 advances to block 9936 of FIG. 101, in which the truss manufacture computing device 12 determines whether the selected pickup option pair is the best pickup option pair as a function of neighbor joint pickup evaluation factors. In doing so, and as indicated in block 9938, the truss manufacture computing device 12 may determine that the selected pickup option pair is the best pickup option pair if the selected pickup option pair does not involve picking up a neighbor joint (e.g., a joint within a predefined distance of the selected joint). In at least some embodiments, the truss manufacture computing device 12 may establish a preference order of picking up the current joint (e.g., the selected joint) as the most preferable and picking up a neighbor joint as less preferable, as indicated in block 9940. As indicated in block 9942, the truss manufacture computing device 12 may determine whether the selected pickup option pair is the best pickup option pair as a function of neighbor joint proximity factors. In doing so, the truss manufacture computing device 12 may determine that the selected pickup option pair is the best pickup option pair if the selected pickup option pair involves picking up a neighbor joint that is within a predefined distance, as indicated in block 9944. In making that determination, the truss manufacture computing device 12 may utilize a predefined distance of 20 inches, as indicated in block 9946. As indicated in block 9948, the truss manufacture computing device 12 may determine whether the selected pickup option pair is the best pickup option pair as a function of intermediate operation evaluation factors. In doing so, the truss manufacture computing device 12 may determine that the selected pickup option pair is the best pickup option pair if the selected pickup option pair does not involve an intermediate operation, as indicated in block 9950. In doing so, the truss manufacture computing device 12 may utilize a definition of intermediate operation as the second robot 216 moving a partially completed truss between regular operations (e.g., assembly operations), as indicated in block 9952.

**[00247]** Referring now to FIG. 102, the truss manufacture computing device 12 may determine, as a function of perimeter part evaluation factors, whether the selected pickup option pair is the best pickup option pair, as indicated in block 9954. In doing so, and as indicated in block 9956, the truss manufacture computing device 12 may determine that the selected pickup

option pair is the best pickup option pair if the selected pickup option pair involves a perimeter part. As indicated in block 9958, the truss manufacture computing device 12 may determine whether the selected pickup option pair is the best pickup option pair if the selected pickup option pair has a higher combined score than the current best pickup option pair, as indicated in block 9960. The truss manufacture computing device 12 may determine the combined score by combining results of the previous checks, as indicated in block 9962 (e.g., by assigning corresponding numeric values to conditions ranked in order of preference in the above determinations and combining the resulting numeric value). As indicated in block 9964, the truss manufacture computing device 12 may determine whether the selected pickup option pair is the best pickup option pair as a function of part length evaluation factors. In doing so, the truss manufacture computing device 12 may determine that the selected pickup option pair is the best pickup option pair if the selected pickup option pair involves picking up a longer part than the current best pickup option pair, as indicated in block 9966. The truss manufacture computing device 12 may determine whether the selected pickup option pair is the best pickup option pair as a function of center distance evaluation factors, as indicated in block 9968. In doing so, the truss manufacture computing device 12 may determine that the selected pickup option pair is the best pickup option pair if the selected pickup option pair involves a greater distance between the center of the tool 228 and the center of the press 180 than the current best pickup option pair, as indicated in block 9970. In making the determination of block 9970, the truss manufacture computing device 12 may define the center of the 228 as the center of the end effector (e.g., the center of the elongate body 230 of the tool 228), as indicated in block 9972.

**[00248]** Referring now to FIG. 103, in the illustrative embodiment, the truss manufacture computing device 12 may determine whether the selected pickup option pair is the best pickup option pair as a function of overlap evaluation factors, as indicated in block 9974. In doing so, and as indicated in block 9976, the truss manufacture computing device 12 may determine that the selected pickup option pair is the best pickup option pair if the selected pickup option pair involves less overlap area than the current best pickup option pair. The truss manufacture computing device 12 may utilize a definition of overlap as an amount each robot with reach into an area associated with another robot (e.g., as determined utilizing corresponding polygons representative of each robot 214, 216), as indicated in block 9978. As indicated in block 9980, the truss manufacture computing device 12 may determine whether the selected

pickup option pair is the best pickup option pair as a function of clamp to press distance evaluation factors. In doing so, and as indicated in block 9982, the truss manufacture computing device 12 may determine that the selected pickup option pair is the best pickup option pair if the selected pickup option pair involves a shorter distance between the center of the gripper 232 (which includes clamps 234) and the center of the press 180 than the current best pickup option pair.

**[00249]** The truss manufacture computing device 12 may determine whether the selected pickup option pair is the best pickup option pair as a function of robot distance evaluation factors, as indicated in block 9984. In doing so, and as indicated in block 9986, the truss manufacture computing device 12 may determine that the selected pickup option pair is the best pickup option pair if the selected pickup option pair involves a greater distance between the robots 214, 216 than the current best pickup option pair (e.g., to allow more freedom of movement of the robots 214, 216 and reduce a likelihood of collision. The truss manufacture computing device 12 may determine whether the selected pickup option pair is the best pickup option pair as a function of tool y position evaluation factors, as indicated in block 9988. In doing so, the truss manufacture computing device 12 may determine that the selected truss pickup option pair is the best pickup option pair if the selected pickup option pair involves a greater y position of the center of the tool(s) 228 than the current best pickup option pair, as indicated in block 9990. In the illustrative embodiment, the determinations in block 9902 are made sequentially and only on the condition that the preceding determination did not result in the truss manufacture computing device 12 identifying the selected pickup option pair as the best pickup option pair (e.g., thereby potentially replacing a previously identified best pickup option pair). That is, the determinations may be treated as fallback determinations in descending order of preference, whereby a pickup option pair designated as the best pickup option pair earlier in the sequence of determinations would be preferable over a pickup option pair determined to be the best pickup option pair later in the sequence of determinations.

**[00250]** Referring now to FIG. 104, the truss manufacture computing device 12 may execute a method 10400 for adding recipe operations determined (e.g., by the truss manufacture computing device 12) to be necessary. The method 10400 corresponds to block 7220 of the method 7200 of FIG. 72. In the illustrative embodiment, the method 10400 begins with block 10402 in which the truss manufacture computing device 12 determines whether a new pair is

being added in the current assembly operation. If so, the method 10400 advances to block 10404, in which the truss manufacture computing device 12 determines whether the new part exceeds a defined length limit or weight limit. In doing so, and as indicated in block 10406, the truss manufacture computing device 12 determines, in the illustrative embodiment, whether the new part exceeds a defined length or weight limit to pick up the part and move it from one end (e.g., of the part). In other words, the truss manufacture computing device 12 performs a check to determine whether an extra operation should be added to avoid a collision (e.g., due to the part being too long). In block 10408, the truss manufacture computing device 12 determines the subsequent course of action based on whether the limit (e.g., associated with block 10404) is exceeded. If not, the method 10400 advances to block 10410, in which the truss manufacture computing device 12 determines whether rotation of the part will result in interference (e.g., with a component of the automated system 10, based on polygons representative of the components of the automated system 10 and the part). In block 10412, the truss manufacture computing device 12 determines the subsequent course of action based on whether interference will result from rotation of the part. If so, or if the limit associated with block 10404 was exceeded, the method 10400 advances to block 10414, in which the truss manufacture computing device 12 adds a recipe operation to pick up the part from the middle (e.g., rather than an end) and move the part into a position (e.g., a position designated as the correct position). Subsequently, in block 10416, the truss manufacture computing device 12 recalculates pickup options for the first robot 214 and the second robot 216 (e.g., by executing the methods 9000, 9100 described above). Subsequently, or if no new part is being added, or if no interference would result from rotation of the new part, the method 10400 advances to block 10418 of FIG. 105, in which the truss manufacture computing device 12 adds a recipe operation based on the first robot 214 and second robot 216 pickup options.

**[00251]** In block 10420 of FIG. 105, the truss manufacture computing device 12 determines whether a single press is sufficient for the current (e.g., selected) joint. In doing so, the truss manufacture computing device 12 determines whether the nailing plate for the joint is too large (e.g., exceeds a predefined size) for a single press to cover the entire nailing plate, as indicated in block 10422. In block 10424, the truss manufacture computing device 12, in the illustrative embodiment, determines, in response to a determination that a single press is not sufficient for the current joint and as a function of whether the plate requires multiple rows of

presses, one or more additional recipe operations. In block 10426, if the truss manufacture computing device 12 determined that multiple rows are required, the truss manufacture computing device 12 adds one or more additional recipe operations for each column for multiple rows of presses. In doing so, the truss manufacture computing device 12 applies (e.g., does not exceed) and upper limit of two rows (e.g., with no limit on the number of columns, and in which the second row is a copy of the first row), as indicated in block 10428. In block 10430, in response to a determination that multiple rows are not required, the truss manufacture computing device 12 adds one or more additional recipe operations to cover the entire plate.

**[00252]** Referring now to FIG. 106, the truss manufacture computing device 12 may execute a method 10600 for adding intermediate recipe operations. The method 10600 corresponds with block 7222 of the method 7200 described above with reference to FIG. 72. In the illustrative embodiment, the method 10600 begins with block 10602, in which the truss manufacture computing device 12 determines whether a distance to the selected joint exceeds a defined distance threshold (e.g., defined in the memory 592). In block 10604, the truss manufacture computing device 12 determines the subsequent course of action based on whether the distance threshold (e.g., from block 10602) is exceeded. If so, the method 10600 advances to block 10606, in which the truss manufacture computing device 12 ends the method 10600 without creating intermediate recipe operations. Otherwise, if the distance threshold is not exceeded, the method 10600 advances to block 10608, in which the truss manufacture computing device 12 adds one or more intermediate operations to move the truss towards a target position. In doing so, and as indicated in block 10610, the truss manufacture computing device 12 may set an intermediate movement distance to a total distance from the current truss position to the target truss position.

**[00253]** In block 10612, while a valid intermediate movement distance is not found, the truss manufacture computing device 12 may perform operations to find (e.g., determine) a maximum valid intermediate movement distance. A method 10700 that may be executed by the truss manufacture computing device 12 to determine a maximum valid intermediate movement distance is described with reference to FIG. 107. In block 10614, the truss manufacture computing device 12, in the illustrative embodiment, calculates the best pickup point for each part that has already been placed for the truss (e.g., by executing the logic associated with the method 9900). As indicated in block 10616, the truss manufacture computing device 12

determines the best part and pickup point for the intermediate operation. A method 10800 that may be executed by the truss manufacture computing device 12 for determining the best part and pickup point for the intermediate operation is described with reference to FIG. 108. In block 10618, the truss manufacture computing device 12 adds the intermediate recipe operation using the determined best part and pickup point (e.g., determined in block 10616) to move the truss the maximum valid movement distance (e.g., determined in block 10612). In block 10620, the truss manufacture computing device 12 determines the subsequent course of action based on whether the truss is in the target position. If not, the method 10600 loops back to block 10608 to add another intermediate operation to move the truss closer to the target position. Otherwise, the method 10600 advances to block 10622, in which the truss manufacture computing device 12 outputs (e.g., writes to the memory 592) a list of intermediate operation(s) (e.g., determined in one or more iterations of the block 10608) for the current assembly operation.

**[00254]** Referring now to FIG. 107, the truss manufacture computing device 12 may execute a method 10700 for finding a maximum valid intermediate movement distance. The method 10700 corresponds with block 10612 of the method 10600 described above with reference to FIG. 106. In the illustrative embodiment, the method 10700 begins with block 10702, in which the truss manufacture computing device 12 determines whether the second robot 216 and moving truss will interfere with one or more objects (e.g., based on polygons representative of the second robot 216, truss, and other objects in the automated system 10). In doing so, and as indicated in block 10704, the truss manufacture computing device 12 may test for interference with an exit wall, one or more pillars, and/or the press 180. In block 10706, the truss manufacture computing device 12 determines the subsequent course of action based on the determination from block 10702. In response to a determination that the second robot 216 will not interfere with another object, the method 10700 advances to block 10708 in which the truss manufacture computing device 12 determines whether the second robot 216 will move beyond a defined set of maximum or minimum gantry limits.

**[00255]** In block 10710, the truss manufacture computing device 12 determines the subsequent course of action based on whether the second robot 216 will move beyond the limits associated with block 10708. If not, the method 10700 advances to block 10712, in which the truss manufacture computing device 12 determines whether the second robot 216 will move past a defined distance across a center line. In doing so, the truss manufacture computing device 12

may utilize a defined distance of 26 inches (e.g., to avoid interference between the second robot 216 and the carriage 178 associated with the press 180). In block 10716, the truss manufacture computing device 12 determines the subsequent course of action based on whether the second robot 216 will move past the defined distance from block 10712. In response to a determination that the second robot 216 will not exceed (e.g., move past) that defined distance, the truss manufacture computing device 12 determines, in block 10718, whether tooling wires (e.g., of the robotic arm 220) will stretch beyond a defined distance (e.g., due to rotation past a defined angle). In block 10720, the truss manufacture computing device 12 determines the subsequent course of action based on whether the distance from block 10718 will be exceeded. If so, or if any of the conditions from block 10706, 10710, or 10716 are true, the method 10700 advances to block 10722, in which the truss manufacture computing device 12 divides the intermediate movement distance in half and tests the new distance (e.g., half of the earlier intermediate movement distance), such as by executing the above determinations from the method 10700 using the newly determined intermediate distance. Otherwise, the method 10700 instead advances from block 10720 to block 10724, in which the truss manufacture computing device 12 determines that the present intermediate distance is the maximum valid distance.

**[00256]** Referring now to FIG. 108, the truss manufacture computing device 12 may execute a method 10800 for determining a best part and best pickup point for an intermediate operation. The method 10800 corresponds with block 10616 of the method 10600 described above with reference to FIG. 106. In the illustrative embodiment, the method 10600 begins with block 10802, in which the truss manufacture computing device 12 determines whether the current part and pickup point has a less significant problem (e.g., based on orders of preference discussed above) with clamping, robot arm trajectory, or the robot gantry (e.g., the carriage 218) than the current best part and pickup point. In doing so, the truss manufacture computing device 12 utilizes a definition of gantry problem as no solution or utilization of the dead zone described above, as indicated in block 10804. If the problems are not less significant for the current part and pickup point, the method 10800 advances from block 10806 to block 10808, in which the truss manufacture computing device determines whether the current part and pickup point requires another displacement. In block 10810, the truss manufacture computing device 12 determines the subsequent course of action based on whether another displacement is required. If not, the method advances to block 10812 in which the truss manufacture computing device 12

determines whether the current displacement distance is within 50% of the best displacement distance.

**[00257]** In block 10814, the truss manufacture computing device 12 determines the subsequent course of action based on whether the displacement distance is within 50% of the best displacement distance. If so or if another displacement is not needed (e.g., from block 10810), the method advances to block 10816, in which the truss manufacture computing device 12 determines whether the current part and pickup point requires vacuum (e.g., use of the suction pad 238). In block 10818, the truss manufacture computing device 12 determines the subsequent course of action based on whether vacuum (e.g., the suction pad 238) is needed for the current part and pickup point. If not, or if the problems analyzed in block 10806 are less significant, or if the displacement distance is within 50% of the best displacement distance (e.g., from block 10814), the method 10800 advances to block 10820, in which the truss manufacture computing device 12 sets the current part and pickup point as the best part and pickup point. Otherwise, the method advances from block 10818 to block 10822 of FIG. 109, in which the truss manufacture computing device 12 determines whether the current part and pickup point requires movement of the press 180.

**[00258]** In block 10824, the truss manufacture computing device 12 determines the next course of action based on whether movement of the press 180 is required. If not, the method 10800 branches back to block 10820 in which the truss manufacture computing device 12 designates the current part and pickup point as the best. Otherwise, the method 10800 advances to block 10826, in which the truss manufacture computing device 12 determines whether the current part and pickup point has a higher combined score than the current best part and pickup point. In doing so, the truss manufacture computing device 12 may calculate the combined score based on a combination of scores from previous checks (e.g., the preceding determinations in the method 10800), as indicated in block 10828. In block 10830, the truss manufacture computing device 12 determines the subsequent operation based on the result from block 10826. If the current part and pickup point has a higher combined score than the current best part and pickup point, then the method 10800 branches back to block 10820 described above. Otherwise, the method 10800 advances to block 10832, in which the truss manufacture computing device 12 determines whether the current part and pickup point will cause overlap in robot areas (e.g., overlap in polygons associated with the robots 214, 216). In doing so, in some embodiments, the

truss manufacture computing device 12 defines overlap as the amount that each robot 214, 216 will reach into an area of the other robot 214, 216, as indicated in block 10834. In block 10836, the truss manufacture computing device 12 applies a preference order of no overlap (e.g., most preferable) and less overlap (e.g., more preferable than relatively more overlap). In block 10838, the truss manufacture computing device 12 determines the subsequent course of action based on whether overlap will occur. If not, the method 10800 branches back to block 10820 in which the truss manufacture computing device 12 designates the current part and pickup point as the best. Otherwise, the method 10800 advances to block 10840, in which the truss manufacture computing device 12 determines whether the current part and pickup point causes less overlap that the current part and pickup point (e.g., through comparison of corresponding polygons and overlap between them).

**[00259]** The method advances to block 10842 of FIG. 110, in which the truss manufacture computing device 12 determines the subsequent course of action based on whether the current part and pickup point causes less overlap than the best part and pickup point. If so, the method 10800 branches to block 10820 in which the truss manufacture computing device 12 designates the current part and pickup point as the best. Otherwise the method 10800 advances to block 10844, in which the truss manufacture computing device 12 determines whether the current part and pickup point has a smaller distance to (e.g., is closer to) a preferred location (e.g., a location defined as preferred location) the current best part and pickup point. In doing so, and as indicated in block 10846, the truss manufacture computing device 12 may define the preferred location as at least 24 inches from the center line, if the truss is moving in the positive x direction. As indicated in block 10848, the truss manufacture computing device 12 may define the preferred location as less than or equal to 87 inches from the center line if the truss is moving in the negative x direction. In block 10850, the truss manufacture computing device 12 determines the subsequent course of action based on whether the current part and pickup point is closer to the preferred location than the current best part and pickup point. If so, the method 10800 branches to block 10820, in which the truss manufacture computing device 12 designates the current part and pickup point as the best. Otherwise, the method 10800 advances to block 10852, in which the truss manufacture computing device 12 determines whether the current part and pickup point involves a longer part than the current part and pickup point (e.g., by comparing the lengths of the parts). In block 10854, the truss manufacture computing device 12

determines the subsequent course of action based on whether the current part is longer than that part associated with the best part and pickup point. If the part associated with the current part and pickup point is longer, the method 10800 branches to block 10820 of FIG. 108, in which the truss manufacture computing device 12 designates the current part and pickup point as the best. Otherwise, the method 10800 advances to block 10856, in which the truss manufacture computing device 12 determines whether the current part and pickup point involves a longer distance between the press 180 and center of the tool 228 than the current best part and pickup point.

**[00260]** Referring now to block 10858 of FIG. 111, if the current part and pickup point involves a longer distance between the press 180 and the center of the tool 228, the method 10800 advances to block 10820 of FIG. 108, in which the truss manufacture computing device 12 designates the current part and pickup point as the best. Otherwise, the method 10800 advances to block 10860, in which the truss manufacture computing device 12 determines whether the current part and pickup point provides a longer distance between the gripper 232 (which includes one or more clamps 234) and the press 180 than the current best part and pickup point. In block 10862, in response to a determination that the current part and pickup point provides a longer distance, the method 10800 advances to block 10820 in which the truss manufacture computing device 12 determines that the current part and pickup option is the best. Otherwise, the method 10800 advances to block 10864, in which the truss manufacture computing device 12 determines not to set (e.g., designate) the current part and pickup point as the best part and pickup point. In the illustrative embodiment, the truss manufacture computing device 12 performs the method 10800 for each of multiple parts and pickup points to determine which is the best part and pickup point from the group.

**[00261]** Referring now to FIG. 112, the truss manufacture computing device 12 may perform a method 11200 for performing efficient selection of materials (e.g., lumber) to produce a wooden structure (e.g., a truss). That is, through execution of the method 11200, the truss manufacture computing device 12 operates to satisfy an efficiency target (e.g., a goal, such as to minimize the amount of wasted (e.g., unused) lumber and/or to minimize total cost of materials while utilizing lumber satisfying a required grade for the wooden structure). In the illustrative embodiment, the method 11200 begins with block 11202, in which the truss manufacture computing device 12 obtains lumber selection data, which may be embodied as any data

indicative of parameters to be utilized by the truss manufacture computing device 12 in the selection of lumber. The truss manufacture computing device 12 may obtain the lumber selection data from memory 592 (e.g., read from a file or database), from another computing device (e.g., via the communications interface 594), from the input/output interface 593, and/or other sources. In obtaining lumber selection data, the truss manufacture computing device may obtain assembly recipe data (e.g., produced through execution of the method 7000 and methods called therefrom (e.g., executed in the performance of the method 7000)), as indicated in block 11204. The truss manufacture computing device may obtain production data indicative of one or more jobs (e.g., batches or sets of each of one or more wooden structures (e.g., trusses) to be produced), as indicated in block 11206. The truss manufacture computing device 12 may obtain line data indicative of the status of one or more in-feed lines 66 associated with lumber and/or a status of the cutting station 16, as indicated in block 11208. Additionally, in the illustrative embodiment, the truss manufacture computing device 12 obtains inventory data indicative of lumber inventory (e.g., the lumber pieces available to the automated system 10, including characteristics of the lumber pieces, such as their grade, price (e.g., per unit of length), dimensions, quantity, etc.), as indicated in block 11210.

**[00262]** As indicated in block 11212, the truss manufacture computing device 12 may obtain saw parameter data, which may be embodied as any data indicative of parameters (e.g., offsets, dimensions, limits, etc.) associated with the saw assembly 90 and components thereof (e.g., the saw 94). The saw parameter data may include parameter data discussed with reference to FIG. 69, which may be produced at least in part from calibration operations. The truss manufacture computing device 12 may also obtain standard lengths configuration data, which may be embodied as any data indicative of one or more default, expected, or target lengths for pieces of lumber (e.g., stock lumber), as indicated in block 11214. In block 11216, the truss manufacture computing device 12 determines the parts in a batch (e.g., a set of trusses to be produced, as may be defined in the production data). In doing so, and as indicated in block 11218, the truss manufacture computing device 12, in the illustrative embodiment, determines all parts for all trusses in the batch. A method 11400 that may be executed by the truss manufacture computing device 12 for determining the parts in a batch is described with reference to FIG. 114.

**[00263]** Referring now to FIG. 113, in block 11220, the truss manufacture computing device 12 begins a loop of operations for each part in the batch. In the illustrative embodiment,

the truss manufacture computing device 12 determines whether one or more unanalyzed parts remain in the batch. In response to a determination that one or more unanalyzed parts do remain in the batch, the method 11200 proceeds to block 11222 in which the truss manufacture computing device 12 selects the next (e.g., the first part, in the initial iteration of the operations) part in the batch. Afterwards, in block 11224, the truss manufacture computing device 12 determines a set of boards (e.g., lumber pieces) for potential use in the batch (e.g., in the production of parts for one or more trusses associated with the batch). A method 11600 that may be executed by the truss manufacture computing device 12 to determine a set of boards for potential use in a batch is described with reference to FIG. 116. Further, in block 11226, the truss manufacture computing device 12 determines a set of potential parts for (e.g., to be produced using) each board (e.g., lumber piece). A method 11800 for determining a set of potential parts for each board that may be executed by the truss manufacture computing device 12 is described with reference to FIG. 118. In the illustrative embodiment, the truss manufacture computing device 12 initiates (e.g., concurrently) an analysis to determine the best board with a better grade (e.g., than a grade specified in the data (e.g., production data) defining the truss(es) to be produced), as indicated in block 11228. Additionally, the truss manufacture computing device 12 initiates (e.g., concurrently) an analysis to determine the best board with the same grade (e.g., as a grade specified in the data (e.g., production data) defining the truss(es) to be produced), as indicated in block 11230. A method 11900 for determining a best board (e.g., in association with blocks 11228, 11230) that may be executed by the truss manufacture computing device 12 is described with reference to FIG. 119.

**[00264]** The method 11200 continues in block 11232, in which the truss manufacture computing device 12 outputs (e.g., writes to the memory 592) the best board for all grades (e.g., the best board determined from the operations associated with blocks 11228, 11230). In block 11234, the truss manufacture computing device 12 removes one or more parts created by the determined best board (e.g., from block 11232) from the list for processing (e.g., for subsequent iterations of the loop). In block 11236, the truss manufacture computing device 12 adds one or more parts (e.g., to be created from the board identified as the best board) from a defined set of parts designated as standard parts (e.g., lumber pieces to be placed in a lumber yard, and having lengths defined in the standard length configuration data from block 11214). A method 12400 that may be executed by the truss manufacture computing device 12 for adding standard parts

(e.g., to be created from a given board) is described with reference to FIG. 124. Subsequently, the method loops back to block 11220 to determine whether there are remaining parts to be analyzed for the batch. If so, the truss manufacture computing device 12 selects the next part and performs the operations described above for that selected part. Otherwise, the method 11200 branches to block 11238, in which the truss manufacture computing device 12 outputs (e.g., writes to the memory 592) the set of all boards (e.g., all boards to be used in the production of the truss(es) associated with the batch) with created parts associated with the boards (e.g., identifiers of the parts to be created from each board in the set of boards).

**[00265]** Referring now to FIG. 114, the truss manufacture computing device 12 may execute a method 11400 for determining parts in a batch. The method 11400 corresponds with block 11218 of the method 12000 described above. In the illustrative embodiment, the method 11400, begins with block 11402 in which the truss manufacture computing device 12 determines whether the batch is queued for the first line (e.g., in-feed line 66A) or the second line (e.g., in-feed line 66B). In block 11404, the truss manufacture computing device 12 determines the subsequent course of action as a function of whether the batch (e.g., job) is queued for a line 66A, 66B. If so, the method 11400 advances to block 11406 in which the truss manufacture computing device 12 enters a loop to be executed for every truss in the assembly recipe (e.g., obtained in block 11204) for the current job (e.g., batch). In block 11408, the truss manufacture computing device 12 selects the next truss in the recipe. In block 11410, the truss manufacture computing device 12 enters a loop for every part in the selected truss. In doing so, the truss manufacture computing device 12 determines whether any unanalyzed parts remain in the selected truss. If not, the method 11400 branches to block 11412 in which the truss manufacture computing device 12 outputs (e.g., writes to the memory 592) all parts (e.g., identifiers of all parts) for the selected truss. Otherwise, if unanalyzed parts do remain for the selected truss, the method 11400 advances to block 11414 in which the truss manufacture computing device 12 selects the next part for the selected truss. Additionally, in block 11416, the truss manufacture computing device 12 calculates a pickup location. The pickup location is, in the illustrative embodiment, embodied as a length along the part (e.g., the selected part from block 11414) where the first robot 214 will pick up the part. In block 11418, the truss manufacture computing device 12 determines whether the pickup location (e.g., calculated in block 11416) satisfies a predefined length threshold. In doing so, in the illustrative embodiment, the truss manufacture

computing device 12 determines whether the pickup location is greater than 2500 millimeters (e.g., along the length of the part), as indicated in block 11420.

**[00266]** Referring now to block 11422 of FIG. 115, the truss manufacture computing device 12 determines the subsequent course of action based on whether the predefined length threshold (e.g., from block 11418) is satisfied. If so, the method 11400 advances to block 11430, in which the truss manufacture computing device 12 determines to rotate the part 180 degrees. If the length threshold is not satisfied, the method advances to block 11424, in which the truss manufacture computing device 12 determines whether the pickup location satisfies a predefined threshold associated with a percentage of the total length of the part. In doing so, the truss manufacture computing device 12 may determine whether the pickup location is greater than 70% of the length of the part, as indicated in block 11426. In block 11428, the truss manufacture computing device 12 determines the subsequent course of action based on whether the threshold from block 11424 is satisfied. If so, the method 11400 branches to block 11430, in which the truss manufacture computing device 12 determines to rotate the part 180 degrees. Otherwise, the truss manufacture computing device 12 advances to block 11432, in which the truss manufacture computing device 12 determines not to rotate the part (e.g., the pickup location is not too far along the length of the part). In either case, after making the determination to rotate the part or not, the method 11400 loops back to block 11410 to determine whether any additional parts remain in the selected truss. If not, the truss manufacture computing device 12 outputs all parts for the selected truss in block 11412 (e.g., as described above) and loops back to block 11406, in which the truss manufacture computing device 12 determines whether any additional trusses remain in the recipe. If not, the method 11400 branches to block 11434 of FIG. 115, in which the truss manufacture computing device 12 outputs (e.g., writes to the memory 592) all parts for all trusses in the batch.

**[00267]** Referring now to FIG. 116, the truss manufacture computing device 12, in the illustrative embodiment, may perform a method 11600 for determining a set of boards for potential use in a batch (e.g., associated with a job). The method 11600 corresponds with block 11224 of the method 11200, described above. In the illustrative embodiment, the method 11600 begins with block 11602, in which the truss manufacture computing device 12 determines whether remaining boards are present in the inventory (e.g., from block 11210 of the method 11200) that have not been analyzed in the method 11600. If not, the method 11600 advances to

block 11604, in which the truss manufacture computing device 12 outputs (e.g., writes to the memory 592) the set of boards for potential use in the batch. Otherwise, the method 11600 instead advances to block 11606, in which the truss manufacture computing device 12 selects the next board from the lumber inventory, as indicated in block 11606. In block 11608, the truss manufacture computing device 12 determines whether the width of the selected board is equal to the width of the selected part associated with the batch (e.g., the part selected in block 11222 of the method 11200). In block 11610, the truss manufacture computing device 12 determines the subsequent course of action based on whether the width of the currently selected board is equal to the part width. If so, the method advances to block 11612, in which the truss manufacture computing device 12 determines whether the length of the selected board satisfies a length threshold. In doing so, and as indicated in block 11614, the truss manufacture computing device 12 may determine whether the length of the selected board is greater than or equal to the length of the selected part. In block 11616, the truss manufacture computing device 12 determines the subsequent course of action based on the determination from block 11612.

**[00268]** In the illustrative embodiment, if the length of the board does satisfy the length threshold, the method 11600 advances to block 11618, in which the truss manufacture computing device 12 determines whether the grade of the selected board is equal to a lumber grade defined in association with the part (e.g., in the memory 592, as defined in the lumber selection input data from block 11202, as defined in a table of parts and associated materials and lumber pieces, etc.). In block 11620, the truss manufacture computing device 12 determines the subsequent course of action based on whether the grade of the selected board is equal to the grade defined for the selected part. If not, the method 11600 advances to block 11622, in which the truss manufacture computing device 12 determines whether the grade of the selected board is greater than the grade defined in association with the selected part.

**[00269]** Referring now to FIG. 117, in block 11624, the truss manufacture computing device 12 determines the subsequent course of action based on whether the grade of the selected board is greater than the grade defined in association with the selected part. If so, the method 11600 advances to block 11626, in which the truss manufacture computing device 12 determines whether the grade of the selected board is less than the grade of the board from a previous iteration of the method 11600 (e.g., starting at block 11606). In block 11628, the truss manufacture computing device 12 determines the subsequent operations based on whether the

grade of the selected board is less than the grade of the board from an earlier iteration of the method 11600 (e.g., a previously considered board). If so, the method 11600 advances to block 11630, in which the truss manufacture computing device 12 determines whether the length of the selected board is less than the previously considered board (e.g., a board from a previous iteration of the method 11600) having a grade that is greater than the grade defined in association with the selected part.

**[00270]** In block 11632, the truss manufacture computing device 12 determines the subsequent course of action based on whether the length of the currently selected board is less than the length of a previously considered board that had a higher grade than the grade associated with the selected part. If not or if the determinations from any of blocks 11610, 11616, 11624 are negative, the method 11600 advances to block 11634, in which the truss manufacture computing device 12 excludes, from a set of boards for potential use, the selected board. Otherwise, if the selected board is shorter (e.g., from block 11632), or if the determination in block 11620 is positive, the method 11600 advances to block 11636, in which the truss manufacture computing device 12 adds the selected board to the set of boards for potential use. After performing block 11634 (e.g., excluding the select board from the set for potential use) or block 11636 (e.g., adding the selected board to the set of boards for potential use), the method 11600 loops back to block 11602 to determine whether additional boards are present for analysis. If so, the method 11600 loops through the above-discussed operations after selecting the next board in block 11606. Otherwise, and as described above, the method 11600 advances to block 11604, in which the truss manufacture computing device 12 outputs (e.g., writes to the memory 592) the set of boards for potential use.

**[00271]** Referring now to FIG. 118, the truss manufacture computing device 12 may perform a method 11800 for determining a set of potential parts for each board (e.g., each board in the set of potential boards, determined via execution of the method 11600). The method 11800 corresponds with block 11226 of the method 11200. In the illustrative embodiment, the method 11800 begins with block 11802, in which the truss manufacture computing device 12 determines whether additional boards (e.g., from the set of potential boards, determined from execution of the method 11600, which is called from block 11224 of the method 11200) remain for analysis. In response to a determination that no additional boards are present in the set, the method 11800 advances to block 11804, in which the truss manufacture computing device 12 outputs (e.g.,

writes to the memory 592) all possible parts for all board in the set of boards for potential use. However, in response to a determination that additional boards exist for analysis, the method 11800 instead advances to block 11806, in which the truss manufacture computing device 12 selects a board from the set of boards for potential use. In block 11808, the truss manufacture computing device 12 determines a subsequent course of action based on whether additional parts remain for analysis for the current batch (e.g., set of trusses to be produced in association with a job). If not, the method 11800 advances to block 11810, in which the truss manufacture computing device 12 outputs (e.g., writes to the memory 592) a list of total possible parts for (e.g., that may be produced using) the selected board, then loops back to block 11802 to potentially select another board for analysis.

**[00272]** Otherwise, the method 11800 advances to block 11812, in which the truss manufacture computing device 12 selects a part associated with the batch. In block 11814, the truss manufacture computing device 12 determines whether the grade of the selected board (e.g., selected in block 11806) is equal to the grade defined in association with the selected part. In block 11816, the truss manufacture computing device 12 determines the subsequent course of action based on the result of the determination from block 11814. If the grade is not equal, the method 11800 loops back to block 11808, in which the truss manufacture computing device 12 potentially selects another part associated with the batch (e.g., if available). Otherwise, if the grades are equal, the method 11800 advances to block 11818, in which the truss manufacture computing device 12 determines whether the width of the selected board is equal to the width of the selected part. In block 11820, the truss manufacture computing device 12 determines whether to proceed with the selected part or not based on the determination from block 11818. If the widths are not equal, the method 11800 loops back to block 11808 to potentially select another part. Otherwise (e.g., if the widths are equal), the method 11800 advances to block 11822, in which the truss manufacture computing device 12 determines whether the selected part can fit on the selected board (e.g., whether the dimensions of the part fit within the dimensions of the board, after other parts have potentially consumed a portion of the length of the selected board). In block 11824, the truss manufacture computing device 12 determines whether to continue with the selected part based on whether the selected part will fit within the selected board. If not, the method 11800 loops back to block 11808, in which the truss manufacture computing device 12 potentially selects another part for analysis relative to the selected board. Otherwise, if the part

does fit within the selected board, the method 11800 advances to block 11826, in which the truss manufacture computing device 12 adds the selected part to the list of possible parts to be created from the selected board. Accordingly, through the above operations, the truss manufacture computing device 12 reduces wasted materials (e.g., lumber) by utilizing the remainder of the board to produce as many parts as possible. Afterwards, the method 11800 loops back to block 11802 to potentially select another board from the set of boards for potential use.

**[00273]** Referring now to FIG. 119, the truss manufacture computing device 12 may perform a method 11900 for determining a best board (e.g., from a set of potential boards for use). The method 11900 corresponds with blocks 11228, 11230 of the method 11200, described above. In the illustrative embodiment, the method 11900 begins with block 11902, in which the truss manufacture computing device 12 determines whether boards remain for analysis (e.g., from the set of boards determined in the method 11600, called by block 11224 of the method 12000). In response to a determination that no additional boards remain in the set, the method 11900 branches to block 11904, in which the truss manufacture computing device 12 outputs (e.g., writes to the memory 592) the best board (e.g., an identifier of the board determined to be the best) for the selected grade (e.g., a grade passed in as a parameter from either of blocks 11228, 11230 of the method 11200). Otherwise, if one or more boards do remain in the set, the method 11900 instead advances to block 11906 in which the truss manufacture computing device 12 selects a board from the set. In block 11908, the truss manufacture computing device 12 determines every possible ordering of parts for the selected board. In doing so, and as indicated in block 11910, the truss manufacture computing device 12 determines, for a given number of parts (e.g., "n") associated with the selected board (e.g., as determined by the method 11800, which is called from block 11226 of the method 11200) n factorial (e.g., n!) possible permutations (e.g., orderings of parts).

**[00274]** In block 11912, the truss manufacture computing device 12 determines all possible groups of possible orderings of any number of parts for the selected board. That is, the truss manufacture computing device 12 may produce groups of one part, two parts, and so on, up to n parts in a group, and then determines every possible ordering of the parts in each of those groups (e.g., group 1: 1! possibility, group 2: 2! possibilities, . . .group n-1: (n-1)! possibilities, group n: n! possibilities). In block 11914, the truss manufacture computing device 12 enters a loop of operations to be performed for each group created above. If one or more groups that have

not yet been analyzed are present, the method 11900 advances to block 11916, in which the truss manufacture computing device 12 selects another group for analysis. In block 11918, the truss manufacture computing device 12 enters a nested loop in which the truss manufacture computing device 12 performs operations for each possible ordering within the selected group. If one or more unanalyzed orderings remain, the method 11900 advances to block 11920, in which the truss manufacture computing device 12 selects a possible ordering of parts for the group. In block 11922, the truss manufacture computing device 12 enters a further nested loop for operations on each remaining part within the selected possible ordering for parts in the selected group. If unanalyzed parts remain for the selected possible ordering for the selected group, the method 11900 advances to block 11924 of FIG. 120, in which the truss manufacture computing device 12 selects the next part in the selected possible ordering of parts for the selected group.

**[00275]** Referring now to FIG. 120, the truss manufacture computing device 12, in the illustrative embodiment, determines whether the selected part (e.g., selected in block 11924) is the first part in the selected possible ordering for the selected board, as indicated in block 11926. Subsequently, in block 11926, the truss manufacture computing device 12 determines the subsequent operations based on the determination from block 11924. If the part is the first part in the selected possible ordering for the selected board, the method 11900 advances to block 11930 in which the truss manufacture computing device 12 may selectively (e.g., conditionally) rotate the board. In doing so, the truss manufacture computing device 12 may rotate the board to orient a straight edge of the selected part with a straight edge of the board, if possible (e.g., if the rotation will enable the part to fit within the dimensions of the board), as indicated in block 11932. Subsequently, the method 11900 loops back to block 11922 to determine whether additional parts remain for the selected ordering of parts in the selected group.

**[00276]** Referring back to block 11928, if the selected part is not the first part, the method 11900 instead advances to block 11934, in which the truss manufacture computing device 12 determines whether the selected part can be rotated. In doing so, and as indicated in block 11936, the truss manufacture computing device 12 determines whether rotation (e.g., of the selected part) would satisfy one or more size parameters. As indicated in block 11938, the truss manufacture computing device 12 determines whether rotation of the part would satisfy a size parameter for a saw clamp (e.g., whether the rotations would be too small (e.g., not satisfy a predefined value) for the saw clamp (e.g., board holders 97)). In block 11940, the truss

manufacture computing device 12 may determine whether rotation would satisfy a length parameter for one or more of the assembly robots 214, 216. That is, in at least some embodiments, the truss manufacture computing device 12 may determine whether rotation would result in a length that is too long (e.g., greater than a defined length value) associated with the robots 214, 216 (e.g., making manipulation of the board by the robots 214, 216 impossible). In block 11942 (e.g., if the selected part can be rotated based on the determination from block 11934), the truss manufacture computing device 12, in the illustrative embodiment, identifies a rotation option that would cause the part to use less of the board than other available rotation options.

**[00277]** Referring now to FIG. 121, the truss manufacture computing device 12 may determine whether an angle of the selected part can nest within the angle of the previous part (e.g., the previous part in the selected ordering of parts for the selected group). In doing so, and as indicated in block 11946, the truss manufacture computing device 12 may determine whether both parts (e.g., the selected part and the previous part in the ordering) can cross over a centerline between them if angled cuts are utilized. In some embodiments, the truss manufacture computing device 12 may selectively add one or more additional cuts (e.g., to be performed by the saw 94) depending on (e.g., as a function of) the angles. In block 11948, the truss manufacture computing device 12 may determine whether the selected part fits better (e.g., has a fit that satisfies conditions for being deemed "better") after the previous part or at the beginning of the board. In doing so, the truss manufacture computing device 12 determines whether the last piece of the board will be large enough (e.g., satisfying predefined dimension(s)) to enable clamps (e.g., board holders 97, clamps 234 of the gripper 232, etc.) to manipulate the board, as indicated in block 11950. The truss manufacture computing device 12 may also determine whether the part will fit at the beginning of the selected board if the part is too small (e.g., not satisfying predefined dimension(s)) for the end of the board, as indicated in block 11952. Subsequently, the method 11900 loops back to block 11922.

**[00278]** In block 11922, in response to a determination the no more parts remain for analysis for the selected possible order of parts for the selected group of parts, the method 11900 advances to block 11954 of FIG. 121, in which the truss manufacture computing device 12 outputs (e.g., writes to the memory 592) part location configuration data (e.g., data indicative of a determined set of locations for the parts) for the selected possible ordering of parts within the

selected group. Afterwards, the method 11900 advances to block 11956, in which the truss manufacture computing device 12 determines whether the selected possible order of parts is better (e.g., satisfies defined criteria for being designated as better, such as making the more use of the lumber (e.g., reducing waste), enabling more efficient robotic manipulation and cutting of the board, etc.) than the previous best possible ordering of parts for the selected group. In doing so, the truss manufacture computing device 12 makes the determination as a function of an amount of the board utilized the parts, as indicated in block 11958. As such, with a target goal of fitting as many parts as possible within a board (e.g., to minimize waste), the truss manufacture computing device 12 may determine that one part ordering (e.g., the selected part ordering) that results in the parts utilizing less of the board than another part ordering (e.g., thereby leaving a portion of the board for additional parts) is better than the other part ordering. In block 11960, the truss manufacture computing device 12 determines the subsequent course of action based on whether the selected part ordering is better than the previous best ordering of parts. If so, the method 11900 advances to block 11962, in which the truss manufacture computing device 12 identifies (e.g., designates) the selected possible part ordering as the best possible part ordering for the selected board. Subsequently, or if the result in block 11960 was negative, the method 11900 loops back to block 11918 of FIG. 119. If no possible orderings remain for the selected group, the method 11900 branches from block 11918 to block 11964 of FIG. 122, in which the truss manufacture computing device 12 outputs (e.g., writes to the memory 592) the identified best possible ordering of parts for the selected group of parts.

**[00279]** Referring now to FIG. 122, the method 11900, in the illustrative embodiment, advances to block 11966, in which the truss manufacture computing device 12 determines whether any other parts may fit on the selected board (e.g., by comparing dimensions of the parts to dimensions of a remaining portion of the board). In block 11968, the truss manufacture computing device 12 determines a subsequent set of operations based on whether one or more parts can fit on the selected board. If so, the method advances to block 11970, in which the truss manufacture computing device 12 determines to continue with checking (e.g., analyzing) a next group of parts. In doing so, and as indicated in block 11972, the truss manufacture computing device 12 determines that the size of the next group of parts will be the size of the selected group of parts, plus one more part (e.g., from a presently selected group of three parts to a group of four parts). Subsequently, the method 11900 loops back to block 11914, in which the truss

manufacture computing device 12 determines whether a remaining group of possible orderings is present in the determined set of groups. If so, the method 11900 advances to block 11916 to select the next group of parts and analyze the group according to the operations described above.

**[00280]** Referring back to block 11968, if the truss manufacture computing device 12 determined that no additional parts can fit within the remaining portion of the selected board, the method 11900 instead advances to block 11974 of FIG. 122, in which the truss manufacture computing device 12 identifies the selected group as the best group. Subsequently, or in response to a determination, in block 11914, that no other groups remain to be analyzed, the method 11900 advances to block 11976, in which the truss manufacture computing device 12 outputs (e.g., writes to the memory 592) the best (e.g., as determined by the truss manufacture computing device 12 according to the above operations of the method 11900) possible ordering of parts for all groups. In block 11978, the truss manufacture computing device 12 determines whether the selected board is better than the previous best board. In doing so, the truss manufacture computing device 12 may make the determination as a function of price per inch, as indicated in block 11980. In doing so, the truss manufacture computing device 12 may apply a preference to a relatively lower price per inch (e.g., to minimize cost to produce the corresponding truss(es)), as indicated in block 11982. That is, between two otherwise equal boards (e.g., having the same grade), the truss manufacture computing device 12, in the illustrative embodiment designates the board having a lower price per inch as the better board. Afterwards, the method 11900 advances to block 11984, in which the truss manufacture computing device 12 determines the subsequent course of action based on whether the selected board has been determined to be better than the previous best board. If so, the truss manufacture computing device 12 identifies the selected board as the best board for the selected grade (e.g., a grade defined as a parameter passed to the method 11900 from one of the blocks 11228, 11230). Subsequently, or if the selected board is not determined to be better than the previously determined best board, the method 11900 loops back to block 11902 of FIG. 119 to potentially select another board from the set of potential boards. If no more boards are present in the set for analysis, the method 11900 advances to block 11904 in which the truss manufacture computing device 12 outputs (e.g., writes to memory 592) the best board (e.g., as determined by the truss manufacture computing device 12) for the selected grade (e.g., the grade passed in as a parameter from one of blocks 11228, 11230).

**[00281]** Referring now to FIG. 124, the truss manufacture computing device 12 may

execute a method 12400 for adding (to a set of parts to be created from a board) parts from a defined set of standard parts (e.g., from the standard length configuration data obtained in block 11214 of the method 12000). The method 12400 corresponds to block 11236 of the method 11200, described above. In the illustrative embodiment, the method 12400 begins with block 12402, in which the truss manufacture computing device 12 determines whether the selected board has at least four feet remaining (e.g., after other parts associated with preceding operations have been produced from the board). In block 12404, the truss manufacture computing device 12 determines the subsequent course of action based on the determination from block 12402. If at least four feet remain in the board, the method 12400 advances to block 12406 in which the truss manufacture computing device 12 adds a standard part for the remaining length of the selected board, rounded down to the nearest foot (e.g., based on standard part lengths defined in the standard length configuration data obtained in block 11214). In the illustrative embodiment, the standard part(s) may be placed back in a lumber yard for later use in the production of wooden structure(s).

**[00282]** Subsequently, or if the determination in block 12404 is negative (e.g., that at least four feet does not remain in the board), the method 12400 advances to block 12408, in which the truss manufacture computing device 12 determines whether additional standard parts can be made from the selected board. In doing so, and as indicated in block 12410, the truss manufacture computing device 12 may make the determination as a function of the standard length configuration data (e.g., the standard length configuration data obtained in block 11214 of the method 12000). That is, the truss manufacture computing device 12 determines whether the remaining length of the board is sufficient to accommodate the lengths of any one or more defined lengths in the standard length configuration data. In block 12412, the truss manufacture computing device 12 determines the subsequent course of action based on the determination from block 12408. If additional standard parts can be created from the selected board (e.g., the selected board is long enough to accommodate one or more additional standard parts), the method 12400 advances to block 12414, in which the truss manufacture computing device 12 adds as many standard parts as possible (e.g., as many as will fit) for the selected board. In doing so, the truss manufacture computing device 12 adds, to the set of parts to be created, each part as a function of the remaining size (e.g., length) of the selected board and the sizes (e.g., lengths) of the standard parts as defined in the standard length configuration data from block 11214 of the

method 11200. Referring back to block 12412, in response to a determination that additional standard parts will not fit within the remaining length of the selected board, the method 12400 instead advances to block 12418, in which the truss manufacture computing device 12 determines not to add additional standard parts (e.g., to be created from the selected board).

**[00283]** Referring now to FIG. 125, the automated system 10 may utilize a microservices architecture 12500 in which sets of operations or functions are executed by a set of services (e.g., processes executed by one or more computing devices, including the truss manufacture computing device 12) that communicate using lightweight communication protocols (e.g., network communication protocols, such as hypertext transfer protocol (HTTP), hypertext transfer protocol secure (HTTPS), and/or other protocols that may be mapped onto or used in place of HTTP/HTTPS). The microservices architecture 12500 enables control operations to be loosely coupled and to be updated or modified on an ongoing basis without requiring all processes to be stopped and restarted (e.g., which may result in extended downtime for the automated system 10). Further, by enabling loose coupling of the operations (e.g., through communication using network communication protocols), the architecture is more robust and more readily scaled and distributed across multiple computing devices (e.g., thereby avoiding a single point of failure) as compared to alternative architectures in which all operations are executed within a single physical computing device. In the illustrative embodiment, the microservices 12512 may utilize a set of shared packages which may include executable instructions and/or data (e.g., libraries) for use in functions that are common across multiple microservices 12512. Those shared packages 12510, in the illustrative embodiment, include a geometry package 12520, an entities package 12522, a modules package 12524, a data storage package 12526, and a machine communicator package 12528. The microservices 12512 include a jobs microservice 12530, an assembly recipe generator microservice 12532, a lumber optimizer microservice 12534, an assembler microservice 12536 to communicate with an assembler machine controller 12542, a plate microservice 12538 to communicate with a plate machine controller 12544, and a saw microservice 12540 to communicate with a saw machine controller 12546. The architecture 12500, in the illustrative embodiment, also includes a set of additional microservices 12514, which include a shell microservice 12550, a log aggregator microservice 12552, and a machine diagnostics microservice 12554.

**[00284]** The geometry package 12520, in the illustrative embodiment, includes functions

(e.g., executable instructions that define functions) for creating and manipulating geometric shapes, including creating points, circles, and polygons, and comparing their relative positions (e.g., distances from each other, intersections, overlaps, collisions), sizes, and angles of rotation. Additionally, the entities package 12522, in the illustrative embodiment, includes entities for (e.g., objects representative of) the components of the automated system 10 (e.g., components of one or more of the assembly station 20, the in-feed station 14, the cutting station 16, the buffer station 18) that are utilized by more than one microservice 12512. In the illustrative embodiment, the entities package relies on (e.g., utilizes functions and data defined in) the geometry package 12520. The modules package 12524, in the illustrative embodiment, includes functions and data for the assembly recipe generator microservice 12532, the lumber optimizer microservice 12534, and, in some embodiments, a saw recipe calculator microservice that are used by more than one service (e.g., microservice or solution (e.g., application built to utilize the architecture 12500)). The modules package 12524 may depend on (e.g., utilize functions and data defined in) the geometry package 12520 and the entities package 12522.

**[00285]** The data storage package 12526, in the illustrative embodiment, includes functions and data for saving data in file storage (e.g., locally, through volumes utilized in a containerization system, or otherwise), providing get and set objects that are saved in JavaScript Object Notation representations (or other file or data interchange format), nested folder support, and data storage in other data storage formats (e.g., databases, blobs, etc.). The machine communicator package 12528, in the illustrative embodiment, includes functions and data for enabling reading and writing data and/or files to and from software and machine controllers (e.g., the machine controllers 12542, 12544, 12546). Additionally, the machine communicator package 12528 provides a polling loop for reading machine registers (e.g., data values in the memory 592 of machines of the automated system 10), such as over internet protocol (IP), providing a data bridge (e.g., for read/write operations), and storage of machine connection configuration data. The machine communicator package 12528, in some embodiments, depends on functions and/or data defined in the data storage package 12526.

**[00286]** Referring to the microservices 12512, each microservice 12512 may communicate through one or more interfaces (e.g., application programming interfaces, user interfaces, etc.) that are accessible via one or more port numbers (e.g., using HTTP/HTTPS). The jobs microservice 12530, in the illustrative embodiment, executes operations to manage data

related to jobs performed by the automated system 10. In doing so, the jobs microservice 12530 may receive (e.g., in block 6202 of the method 6200) a file or other data set (e.g., a Job XML file) defining one or more jobs to be executed, including the quantity of each wooden structure (e.g., truss) to be produced by the automated system 10. The jobs microservice, in the illustrative embodiment, additionally manages user interfaces, such as user interfaces to create, read, update, and delete (CRUD) operations for jobs, corresponding recipes, parameters for assembly robots 214, 216, the saw assembly 90, and/or other machines of the automated system 10, lumber inventory, and/or other components and/or operations of the automated system 10. In the illustrative embodiment, the jobs microservice 12530 utilizes data storage for job queries (e.g., with respect to the multiple lines (e.g., lines 66A, 66B, etc.) of the automated system 10), recipe results (e.g., recipes produced by the assembly recipe generator microservice 12532), and optimization results (e.g., selection of boards for use in trusses, as determined by the lumber optimizer microservice 12534).

**[00287]** The assembly recipe generator microservice 12532, in the illustrative embodiment, generates an assembly recipe for a given job with truss shapes and production data. That is the assembly recipe generator microservices 12532 may execute the method 7000 and the methods called therefrom (e.g., in response to a corresponding request from the jobs microservice 12530). The assembly recipe generator microservice 12532 may additionally perform operations for creating, reading, updating, and deleting assembler parameters (e.g., parameters associated with the assembly robots 214, 216) such as those defined in association with calibration operations described herein. The assembly recipe generator microservice 12532, in the illustrative embodiment, utilizes data storage for assembler parameters and uses the geometry package 12520, the entities package 12522, the modules package 12524, and the data storage package 12526.

**[00288]** The lumber optimizer microservice 12534, in the illustrative embodiment, generates a set of optimized lumber (e.g., to satisfy a set of one or more target parameters, such as to minimize cost, minimize wasted lumber, etc.) for a given job with lumber demand using lumber inventory data. That is, in the illustrative embodiment, the lumber optimizer microservice 12534 may execute (e.g., in response to a request from the jobs microservice 12530) the method 11200 described above to select lumber pieces (e.g., boards) to enable efficient use of the lumber (e.g., to avoid wasted lumber, to minimize the cost of materials, etc.). The lumber optimizer

microservice 12534 may perform operations to create, read, update, and delete lumber inventory data (e.g., associated with block 11210 of the method 11200), saw parameter data (e.g., associated with block 11212 of the method 11200), and standard lengths configuration data (e.g., associated with block 11214 of the method 11200). The lumber optimizer microservice 12534, in the illustrative embodiment, utilizes data storage for saw parameter data, lumber inventory data, and standard lengths configuration data and relies on the geometry package 12520, the entities package 12522, the modules packages 12524, and the data storage package 12526.

**[00289]** The assembler microservice 12536, the plate microservice 12538, and the saw microservice 12540, in the illustrative embodiment, each write data to and read data from a corresponding machine controller (e.g., the assembler machine controller 12542 for controlling the assembly robots 214, 216 and/or other components of the assembly station 20, the plate machine controller 12544 for controlling the press 180 and/or associated components, and the saw machine controller 12546 for controlling components of the cutting station 16, such as the saw assembly 90). In operation, the microservices 12536, 12538, 12540 provide interfaces for the corresponding machine controllers, enabling the setting and reading of register values, issuing commands, reading machine files, setting configuration data, monitoring status data, including inventory (e.g., of nailing plates), errors, alerts, alarms, and the like. In the illustrative embodiment, the microservices 12536, 12538, 12540 may convert operations defined in a recipe to corresponding commands or register values that are usable by the corresponding machine controller 12542, 12544, 12546. In operation, the microservices 12536, 12538, 12540 utilize data storage (e.g., for corresponding recipes, configuration data, and/or other data utilized in the performance of the associated operations of each microservice 12536, 12538, 12540 described above). The microservices 12536, 12538, 12540, in the illustrative embodiment, rely on the geometry package 12520, the entities package 12522, the modules packages 12524, the data storage package 12526, and the machine communicator package 12528.

**[00290]** The shell microservice 12550, in the illustrative embodiment, provides a user interface that acts as a shell (e.g., a framework) for all other user interfaces. In the illustrative embodiment, the user interface(s) may utilize JavaScript or TypeScript functions combined with hypertext markup language (HTML), cascading style sheets (CSS), and image files (e.g., portable network graphics (PNG) files, or the like). In some embodiments, the shell may be implemented with the Angular framework. The log aggregator microservice 12552, in the

illustrative embodiment, stores a log of events that occur in the automated system 10. In doing so, the log aggregator microservice 12552 may store machine-level logging events (e.g., in response to machine controller data changes) and/or production-level logging events (e.g., in response to user interactions with one or more components of the automated system 10, such as via a user interface provided by the shell microservice 12550). The log aggregator microservice 12552 may also recycle log files (e.g., on a periodic basis, in response to the log files reaching a predefined size limit, and/or based on other factors). The machine diagnostics microservice 12554, in the illustrative embodiment, analyzes an operational status of the machines (e.g., the assembly robots 214, 216, the saw assembly 90, the press 180) of the automated system 10, such as based on status data reported by the corresponding microservices 12536, 12538, 12540 and may provide a corresponding update indicative of the status to a user interface (e.g., utilizing the shell microservice 12550) and/or to a log (e.g., utilizing the log aggregator microservice 12552).

**[00291]** Referring now to FIG. 126, the truss manufacture computing device 12, which may be distributed across multiple physical computing devices (e.g., across various computing devices of the automated system 10, across multiple racks in a data center, etc.), in operation, may execute a method 12600 for utilizing a microservices architecture (e.g., the microservices architecture 12500) to produce one or more wooden structures with components (e.g., machines) of the automated system 10. In the illustrative embodiment, the method 12600 begins with block 12602, in which the truss manufacture computing device 12 obtains data indicative of a set of one or more jobs (e.g., one or more Job XML files, as described above) for the production of one or more wooden structures (e.g., one or more trusses) by an automated system 10. In block 12604, the truss manufacture computing device 12 utilizes a microservices architecture (e.g., the microservices architecture 12500) to produce the wooden structure(s) associated with the job(s). In doing so, and as indicated in block 12606, the truss manufacture computing device 12 may communicate data (e.g., requests to generate one or more recipes, recipe results, requests to optimize lumber selection, lumber optimization results, requests to add jobs to queues) to and from the microservices 12530, 12532, 12534, 12536, 12538, 12540, 12550, 12552, 12554 using a network communication protocol. In doing so, and as indicated in block 12608, the truss manufacture computing device 12 may communicate among the microservices 12530, 12532, 12534, 12536, 12538, 12540, 12550, 12552, 12554 using hypertext transfer protocol (HTTP) and/or hypertext transfer protocol secure (HTTPS), as indicated in block 12608. In the

illustrative embodiment, data communications that would otherwise be communicated within the same circuit board (e.g., via shared memory) or inter-integrated circuit (e.g., I2C) communication, are mapped to network communication protocols (e.g., application programming interface (API) calls mapped to HTTP communications (e.g., using REST (representation state transfer) API calls). Those communications may include communication of values in memory (e.g., to and from registers). The mapping of the communications to network communication protocols enables loose coupling between processes and the ability to halt, update, and restart processes with minimal impact on the operations of the automated system 10.

**[00292]** In the illustrative embodiment, the truss manufacture computing device 12 utilizes a jobs microservice (e.g., the jobs microservice 12530) to manage routing of data associated with the jobs (e.g., associated with block 12602). In doing so, the truss manufacture computing device 12 may utilize the jobs microservice 12530 to perform create, read, update, and delete operations (CRUD operations) on jobs, corresponding recipes, parameters (e.g., for assembly robots 214, 216, the saw assembly 90, and/or other machines of the automated system 10), lumber inventory, and/or other components and/or operations of the automated system 10, as indicated in block 12612. The truss manufacture computing device 12 may utilize the jobs microservice 12530 to provide a user interface to other microservices (e.g., the assembly recipe generator microservice 12532, the lumber optimizer microservice 12534). As indicated in block 12614, the truss manufacture computing device 12 may utilize an assembly recipe generator microservice (e.g., the assembly recipe generator microservice 12532) to produce a recipe indicative of a sequence of operations to produce the wooden structure(s) (e.g., through execution of the method 7000 in response to a corresponding request from the jobs microservice 12530). As indicated in block 12616, the truss manufacture computing device 12 may utilize a lumber optimizer microservice (e.g., the lumber optimizer microservice 12534) to select lumber to satisfy a set of target parameter(s) (e.g., to minimize cost, to minimize wasted lumber, etc.), such as by executing the method 11200 (e.g., in response to a corresponding request from the jobs microservice 12530). The truss manufacture computing device 12 may utilize microservices to control machines of the automated system to produce the wooden structure(s), as indicated in block 12618. In doing so, the truss manufacture computing device 12 may utilize microservices (e.g., the microservices 12536, 12538, 12540) to read and write machine register values (e.g., associated with the machine controllers 12542, 12544, 12546) using network communication

protocols (e.g., inter- or intra- board serial communications mapped to network communication protocol(s), such as HTTP or HTTPS), as indicated in block 12620.

**[00293]** Referring now to FIG. 127, continuing the method 12600, the truss manufacture computing device 12 may utilize an assembler microservice (e.g., the assembler microservice 12536) to control one or more assembler machines (e.g., the robots 214, 216), as indicated in block 12622. In doing so, and as indicated in block 12624, the truss manufacture computing device 12, in the illustrative embodiment, utilizes the assembler microservice 12536 to communicate with one or more assembler machine controller(s) 12542. For example, and as indicated in block 12626, the truss manufacture computing device 12 may utilize the assembler microservice 12536 to communicate with machine controller(s) 12542 associated with the assembly robots 214, 216 (e.g., the assembler machine controller 12542 may include one or more assembly robot controllers). As indicated in block 12628, the truss manufacture computing device 12 may utilize a plate microservice (e.g., the plate microservice 12538) to control one or more plate machines (e.g., the press 180). In doing so, and as indicated in block 12630, the truss manufacture computing device 12 may utilize the plate microservice 12538 to communicate with one or more plate machine controllers (e.g., the plate machine controller 12544). Similarly, and as indicated in block 12632, the truss manufacture computing device 12 may utilize a saw microservice (e.g., the saw microservice 12540) to control one or more saw machines (e.g., the saw assembly 90). In doing so, and as indicated in block 12634, the truss manufacture computing device 12 may utilize the saw microservice 12540 to communicate with one or more saw machine controllers (e.g., the saw machine controller 12546).

**[00294]** The truss manufacture computing device 12 may utilize a shell microservice (e.g., the shell microservice 12550, described with reference to FIG. 125) to provide one or more user interfaces, as indicated in block 12636. Further, the truss manufacture computing device 12, in executing the method 12600, may utilize a log aggregator microservice (e.g., the log aggregator microservice 12552, also described above with reference to FIG. 126) to manage logs produced by the automated system 10. In some embodiments, the truss manufacture computing device 12 may utilize a machine diagnostics microservice (e.g., the machine diagnostics microservice 12554) to analyze an operational status of one or more machines (e.g., the assembly robots 214, 216, the saw assembly 90, etc.) of the automated system 10, as indicated in block 12640. In performing the method 12600, the truss manufacture computing device 12 may utilize

microservices that are based on one or more shared packages of executable instructions and/or data, as indicated in block 12642. For example, and as described above with reference to FIG. 125, the microservices 12530, 12532, 12534, 12536, 12538, 12540, 12550, 12552, 12554, in the illustrative embodiment, utilize the shared packages 12520, 12522, 12524, 12526, 12528 to determine relative positions, orientations, and sizes of shapes (e.g., polygons) representative of machines (e.g., the robots 214, 216, the press 180, etc.) and/or lumber pieces (e.g., boards) to avoid collisions, ensure successful engagement between a tool (e.g., tool 228) used to pick up or otherwise manipulate a part, perform data storage and retrieval operations (e.g., write and read recipes, lumber selection data, shape data, job data, configuration parameters, etc.), and communicate with machine controllers 12542, 12544, 12546.

**[00295]** As indicated in block 12644, the truss manufacture computing device 12 may interrupt execution of one of the microservices 12530, 12532, 12534, 12536, 12538, 12540, 12550, 12552, 12554 (e.g., shutdown, update, and restart) without interrupting execution of the other microservices 12530, 12532, 12534, 12536, 12538, 12540, 12550, 12552, 12554 in the architecture 12500 (e.g., as the microservices 12530, 12532, 12534, 12536, 12538, 12540, 12550, 12552, 12554 may be executed in separate processes or physical computers). In doing so, any data sent to a microservice 12530, 12532, 12534, 12536, 12538, 12540, 12550, 12552, 12554 that is temporarily interrupted may be stored in a queue and re-sent to be received and processed when execution of the microservice 12530, 12532, 12534, 12536, 12538, 12540, 12550, 12552, 12554 is resumed.

**[00296]** Referring now to FIG. 128, the automated system 10 (e.g., as controlled by the truss manufacture computing device 12), may execute a method 12800 for printing and utilizing fiducials on boards (e.g., lumber pieces) in the production of wooden structure(s) (e.g., one or more trusses). In the illustrative embodiment, the method 12800 begins with block 12802, in which the automated system 10 obtains a board. In doing so, and as indicated in block 12804, the automated system 10 obtains, at an in-feed station (e.g., the in-feed station 14), a piece of stock lumber (SL) (e.g., a piece of stock lumber selected from a lumber inventory, such as a lumber yard). As indicated in block 12806, the automated system 10 may detect, utilizing one or more sensors (e.g., the sensors 82B), that the board has entered an in-feed line 66 of the in-feed station 14. That is, the automated system 10 (e.g., the truss manufacture computing device 12) may obtain data (e.g., a Boolean value or other data transmitted via a network communication

protocol from a corresponding machine controller) indicating that the sensors 82B detected the presence of the stock lumber SL once it has been dropped down onto the extended stops 74 of the in-feed station 14 and that the stops 74 were subsequently retracted to release the board onto a transport section of the in-feed line 66.

**[00297]** In block 12808, the automated system 10 (e.g., as controlled by the truss manufacture computing device 12) moves the board to a defined fiducial printing position (e.g., a position designated for printing of fiducials). In doing so, and as indicated in block 12810, the automated system 10, in the illustrative embodiment, moves the board along an in-feed axis (IA) (e.g., such that the longitudinal axis of the board is coincident with the longitudinal axis of the in-feed line 66) to the defined fiducial printing position. As indicated in block 12812, the automated system 10, in the illustrative embodiment, moves the board along the in-feed axis using carriers 78 driven by a belt 80. Further, and as indicated in block 12814, the automated system 10 detects, with a corresponding sensor, the presence of an end of the board. That is, the automated system 10 may detect the presence of the board with a photoelectric sensor (e.g., the sensor 82E, which may be embodied as a photoelectric sensor), as indicated in block 12816. As indicated in block 12818, the automated system 10 may advance, in response to detection of the presence of the end of the board (e.g., in response to receiving corresponding data from the sensor 82E indicating detection of the end of the board), the board a predefined length along the in-feed axis of the in-feed line 66. In doing so, the automated system 10 may utilize a decoder to detect the length that the carrier 78 has advanced the board along the in-feed axis, as indicated in block 12820.

**[00298]** As indicated in block 12822, the automated system 10 may print, in response to a determination that the board has been moved to the defined fiducial printing position, one or more fiducials on one or more sides of the board. In block 12824, the automated system 10 may print the one or more fiducials on a major side of the board. As indicated in block 12826, the automated system 10 may also print one or more fiducials on a minor side of the board, as described with reference to FIG. 9 above. Referring now to FIG. 129, in printing the one or more fiducials, the automated system 10 may print one or more predefined symbols, as indicated in block 12828. In doing so, and as indicated in block 12830, the automated system 10 may print predefined symbols that are mirrored in opposite directions (e.g., rotated 180 degrees from each other, thereby facilitating recognition of the symbols from multiple angles). The automated

system 10 may print one or more predefined symbols indicative of information about the board, as indicated in block 12832. In doing so, the automated system 10 may print one or more predefined symbols indicative of an identifier (e.g., a serial number) of the board, as indicated in block 12834. The automated system 10, in some embodiments, may print one or more predefined symbols that are indicative of an index value in a sequence of boards to be used in a production process of the automated system 10 (e.g., to be used in the production of a truss), as indicated in block 12836. In some embodiments, the automated system 10 may print one or more predefined symbols indicative of an identifier of a leading end or trailing end of the board (e.g., indicating which end of the board is presently being printed on), as indicated in block 12838. The automated system 10 may, in some embodiments, print one or more symbols indicative of the grade of the board, as indicated in block 12840. In some embodiments, the automated system 10 may print the fiducial(s) using a printer (e.g., the fiducial printer 85) with multiple print heads that are arranged along a width of the board (e.g., to enable printing at any of multiple locations along width of the board), as indicated in block 12842.

**[00299]** In block 12844, the automated system 10 determines whether to print on the other end of the board. In making the determination, the automated system 10 may determine whether the automated system 10 has already printed fiducials on the other end of the board or whether a configuration setting (e.g., in the memory 592) indicates to print on only one end of the board. In response to a determination to print on the other end of the board, the method 12800 advances to block 12846, in which the automated system 10 moves the board to a second defined fiducial printing position associated with a second end of the board. Referring now to FIG. 130, in block 12848, the automated system 10 advances the board along the in-feed axis (e.g., using the carriers 78) until the second end of the board is detected. In doing so, in at least some embodiments, the automated system 10 may advance the board until the sensor 82E detects that the board is no longer present (e.g. no longer within a line of sight of the sensor 82E), as indicated in block 12850. Further, and as indicated in block 12852, the automated system 10 may reverse, in response to detection of the second end of the board, a direction of movement of the board along the in-feed axis (e.g., of the in-feed line 66). In doing so, the automated system 10 may advance the board in the opposite direction by a predefined length (e.g., monitored by a decoder), as indicated in block 12854. Subsequently, the method 12800 loops back to block 12822 of FIG. 128, in which the automated system 12 prints the one or more fiducials on the

present (e.g., second) end of the board, then advances to block 12844 of FIG. 129. In response to a determination not to print on the other end (e.g., because the automated system 10 has already printed on both ends of the board), the method 12800 advances to block 12856 of FIG. 130.

**[00300]** Referring to FIG. 130, in block 12856, the automated system 10 utilizes a vision system (e.g., the vision system 240) to identify one or more fiducials on boards in the production of wooden structure(s) (e.g., wooden truss(es)). In doing so, and as indicated in block 12858, the automated system 10 acquires one or more images of the fiducial(s) with one or more cameras (e.g., the camera 242) of components (e.g., the assembly robots 214, 216) of the automated system 10. As indicated in block 12860, the automated system 10 may determine the position of the board relative to components of the automated system 10 based on the acquired images of the fiducials. For example, given that the fiducial is at a defined position on the board (e.g., the defined fiducial printing position, which is a predefined length from a corresponding end of the board), as discussed above, a component contacting the board on the fiducial is contacting the board at the defined position on the board (e.g., the predefined length from the end of the board). As indicated in block 12862, the automated system 10 may position the camera 242 of the tool 228 of a robot 214, 216 over the fiducials printed at the fiducial printing position, and determine the location of the tool 228 (e.g., the center of the tool 228) relative to the board based on the predefined position of the fiducials on the board and a defined offset of the location of the camera 242 from the center of the tool 228 (e.g., by determining that the position of the center of the tool 228 is the position of the fiducials, plus the offset of the camera 242). In utilizing the vision system 240, the automated system 10 may determine information indicated by the fiducials, as indicated in block 12864. In doing so, the automated system 10 may determine an identifier of the board, the index value of the board in a sequence, a grade of the board, and/or which end of the board is represented in the acquired images, as indicated in block 12866. In determining the information, the truss manufacture computing device 12 may apply object recognition operations, such as a Hough transform and/or object character recognition, to identify the symbols in the fiducials and, in some embodiments, compare the identified symbols to a lookup table of corresponding data (e.g., data, in the memory 592, such as strings or other data structures that are indexed by the symbols in the fiducials). In other embodiments, rather than utilizing a lookup table to determine data based on a symbol, the automated system 10 (e.g., the truss manufacture computing device 12) may interpret the symbols directly (e.g., interpret the

symbols as the data, rather than as a reference to a data set in the memory 592).

**[00301]** The problems of automatically creating wood trusses with varying designs with high quality joints and overall manufacture are known. The systems and methods described herein solve these known technical problems in truss manufacturing, and in the field of construction engineering, using a computer and robot implemented method that creates wood trusses with a novel manufacturing approach. Because of the interrelationship between the software of this system, and the simulation and recipe driven methods that control the robotic and machinery components, the inventions reflected in the computer-implemented method are necessarily rooted in computing technology.

**[00302]** Although described in connection with an exemplary computing system environment, embodiments of the aspects of the disclosure are operational with numerous other general purpose or special purpose computing system environments or configurations. The computing system environment is not intended to suggest any limitation as to the scope of use or functionality of any aspect of the disclosure. Moreover, the computing system environment should not be interpreted as having any dependency or requirement relating to any one or combination of components illustrated in the exemplary operating environment. Examples of well-known computing systems, environments, and/or configurations that may be suitable for use with aspects of the disclosure include, but are not limited to, personal computers, server computers, hand-held or laptop devices, multiprocessor systems, microprocessor-based systems, set top boxes, programmable consumer electronics, mobile telephones, network PCs, minicomputers, mainframe computers, distributed computing environments that include any of the above systems or devices, and the like.

**[00303]** Embodiments of the aspects of the disclosure may be described in the general context of data and/or processor-executable instructions, such as program modules, stored one or more tangible, non-transitory storage media and executed by one or more processors or other devices. Generally, program modules include, but are not limited to, routines, programs, objects, components, and data structures that perform particular tasks or implement particular abstract data types. Aspects of the disclosure may also be practiced in distributed computing environments where tasks are performed by remote processing devices that are linked through a communications network. In a distributed computing environment, program modules may be located in both local and remote storage media including memory storage devices.

**[00304]** In operation, processors, computers and/or servers may execute the processor-executable instructions (e.g., software, firmware, and/or hardware) such as those illustrated herein to implement aspects of the disclosure.

**[00305]** Embodiments of the aspects of the disclosure may be implemented with processor-executable instructions. The processor-executable instructions may be organized into one or more processor-executable components or modules on a tangible processor readable storage medium. Aspects of the disclosure may be implemented with any number and organization of such components or modules. For example, aspects of the disclosure are not limited to the specific processor-executable instructions or the specific components or modules illustrated in the figures and described herein. Other embodiments of the aspects of the disclosure may include different processor-executable instructions or components having more or less functionality than illustrated and described herein.

**[00306]** The order of execution or performance of the operations in embodiments of the aspects of the disclosure illustrated and described herein is not essential, unless otherwise specified. That is, the operations may be performed in any order, unless otherwise specified, and embodiments of the aspects of the disclosure may include additional or fewer operations than those disclosed herein. For example, it is contemplated that executing or performing a particular operation before, contemporaneously with, or after another operation is within the scope of aspects of the disclosure.

**[00307]** Having described the invention in detail, it will be apparent that modifications and variations are possible without departing from the scope of the invention defined in the appended claims.

**[00308]** When introducing elements of the present invention or the preferred embodiment(s) thereof, the articles "a", "an", "the" and "said" are intended to mean that there are one or more of the elements. The terms "comprising", "including" and "having" are intended to be inclusive and mean that there may be additional elements other than the listed elements.

**[00309]** In view of the above, it will be seen that the several objects of the invention are achieved and other advantageous results attained.

**[00310]** As various changes could be made in the above products without departing from the scope of the invention, it is intended that all matter contained in the above description and shown in the accompanying drawings shall be interpreted as illustrative and not in a limiting

sense.

OTHER STATEMENTS OF THE INVENTION

**[00311]** A1. A method of assembling a wooden structure including a plurality of lumber pieces connected together at a plurality of joints, the method comprising:

arranging lumber pieces relative to each other at a first joint location of the wooden structure;

attaching the lumber pieces positioned at the first joint location to each other using a pair of nailing plates secured to top and bottom surfaces of the lumber pieces to form a first joint of the wooden structure;

arranging one or more lumber pieces relative other lumber pieces at a second joint location of the wooden structure after forming the first joint;

attaching each lumber piece at the second joint location using a pair of nailing plates secured to top and bottom surfaces of the lumber pieces at the second joint location to form a second joint of the wooden structure;

arranging one or more lumber pieces relative to other lumber pieces at a third joint location of the wooden structure after forming the second joint;

attaching each lumber piece at the third joint location using a pair of nailing plates secured to top and bottom surfaces of the lumber pieces at the third joint location to form a third joint of the wooden structure; and

continuing to assemble the wooden structure in a joint-by-joint sequence until the entire wooden structure is assembled such that each joint of the wooden structure is completely formed prior to positioning and attaching two or more lumber pieces at another joint, wherein a bottom nailing plate is attached to each lumber piece after arranging the lumber piece at a joint prior to subsequently placing another lumber piece at a joint.

A2. The method of claim A1, wherein the wooden structure is assembled using an automated manufacturing and assembly system.

A3. The method of claims A1 or A2, wherein positioning one or more pieces at the second joint location comprises picking up at least one of the plurality of lumber pieces from a location remote from the second joint location and from the first joint and carrying said at least one of the plurality of lumber pieces to the second joint location.

A4. The method of any one of claims 1-3 wherein the joints of the wooden structure are assembled at a single joint forming station.

A5. The method of any one of claims A1-A4 wherein positioning one or more lumber pieces at the second joint location comprises moving a partially assembled wooden structure including at least the first joint over an assembly table in a first direction to position at least one of the lumber pieces at the second joint location.

A6. The method of claim A5, wherein positioning one or more lumber pieces at the third location comprises moving a partially assembled wooden structure including at least the first and second joints over an assembly table in a second direction to position at least one of the lumber pieces at the third joint location, the second direction including at least a component of movement that is opposite to the first direction.

A7. The method of any one of claims A1-A6, wherein forming the first joint and forming the second joint are carried out one immediately after another.

A8. The method of any one of claims A1-A7, wherein each of said attaching steps comprises fully attaching the bottom nailing plate at the joint prior to attaching the top nailing plate.

A9. The method of claim A8, further comprising using an upper platen to attach the bottom nailing plate at the joint.

A10. The method of claim A1, wherein an order of the joints assembled in the joint-by-joint sequence is determined based on at least one of a location of the joint, a number of lumber pieces at the joint, and a size of the lumber pieces at the joint.

A11. The method of claim A10, wherein the first joint location includes the joint with the longest lumber piece.

A12. The method of claim A10, wherein the second joint location includes the joint having the most number of lumber pieces already in place after forming the first joint.

A13. The method of claim A10, wherein a point system is assigned to various assembly actions and conditions, and a sequence having a highest point total is selected to for the joint-by-joint sequence.

A14. The method of any one of claims A1-A13, wherein the wooden structure is a truss.

B1. An automated assembly method for assembling a wooden structure including a plurality of lumber pieces connected together at a plurality of joints, the method comprising

assembling the wooden structure in a joint-by-joint extrusion sequence until the entire wooden structure is assembled, whereby each joint of the wooden structure is formed by arranging lumber pieces relative to each other at a joint forming station and securing the lumber pieces together using top and bottom nailing plates to form the joints at the joint forming station, the bottom nailing plate being attached to each lumber piece immediately after arranging the lumber piece at a joint prior to subsequently placing another lumber piece at a joint.

B2. The method of claim B1, further comprising assembling the wooden structure using robotic arms to place the lumber pieces in position on an assembly table.

B3. The method of claim B2, wherein no more than two robotic arms at the joint forming station are used to assemble the wooden structure.

B4. The method of claim B2, further comprising attaching the lumber pieces together using upper and lower attachment devices to drive nailing plates into upper and lower surfaces of the lumber pieces.

B5. The method of claims B2 or B3, wherein the robotic arms do not perform any attachment functions for attaching the lumber pieces together.

B6. The method of claims B2 or B3, further comprising transporting the lumber pieces along the assembly table using at least one of the robot arms.

B7. The method of claim B1, wherein the nailing plates are not attached to the lumber pieces at a location other than the joint forming station.

B8. The method of claim B1, wherein the wooden structure is a truss.

C1. An automated assembly method for assembling a wooden structure including a plurality of lumber pieces connected together at a plurality of joints, the method comprising:

using a first robot to position a first lumber piece on an assembly table at a joint location of the wooden structure;

using a second robot to position a second lumber piece on the assembly table at the joint location of the wooden structure;

attaching the first lumber piece to the second lumber piece on the assembly table using an attachment device separate from the first and second robots; and

transporting the wooden structure along the assembly table using one of the first and second robots.

C2. The method of claim C1, wherein the first and second robots hold the first and

second lumber pieces as the first lumber piece is attached to the second lumber piece.

C3. The method of claims C1 or C2, wherein the transporting occurs after the first and second lumber pieces are attached.

C4. The method of claim C1, wherein the first robot retrieves the lumber pieces from a delivery location and the second robot transports the wooden structure along the assembly table.

C5. The method of claim C1, wherein the robots do not perform any attachment functions for attaching the lumber pieces together.

C6. The method of claim C1 wherein transporting the wooden structure along the assembly table comprises moving the wood structure in opposite directions.

C7. The method of claim C1, further comprising:

grabbing the first lumber piece with a tool of the first robot;

viewing a fiducial on the first lumber piece with a camera of the tool; and

determining a position of the tool on the first lumber piece based on a relative position between the tool and the fiducial.

C8. The method of claim C7, further comprising positioning the first lumber piece at the joint location of the wooden structure based on the position of the tool.

D1. An automated assembly method for assembling a wooden structure including a plurality of lumber pieces connected together at a plurality of joints, the method comprising:

using a first robot to position a first lumber piece on an assembly table at a joint location of the wooden structure;

using a second robot to position a second lumber piece on the assembly table at the joint location of the wooden structure;

using one of the first and second robots to hold the first lumber piece at the joint location of the wooden structure;

using the other of the first and second robots to hold the second lumber piece at the joint location of the wooden structure; and

attaching the first lumber piece to the second lumber piece on the assembly table using an attachment device separate from the first and second robots, wherein no other robots are used to position lumber pieces on the assembly table during the assembly of the wooden structure.

D2. The method of claim D1, wherein the first and second robots hold the first and second lumber pieces at the joint location of the wooden structure while the lumber pieces are

being attached.

D3. The method of claim D1, wherein attaching the first lumber piece to the second lumber piece comprises driving a nailing plate into the lumber pieces using a movable platen.

D4. The method of claim D3, wherein attaching the first lumber piece to the second lumber piece comprises driving a nailing plate pair into upper and lower surfaces of the lumber pieces, respectively, using upper and lower platen assemblies.

D5. The method of claim D1, wherein the robots do not perform any attachment functions for attaching the lumber pieces together.

E1. A wooden structure assembly station for assembling a wooden structure, the station comprising:

an assembly table having a length and a width;

a first robot configured to position a first lumber piece on the assembly table at a joint location of the wooden structure, the first robot including a first articulated robot arm;

a second robot, one of the first and second robots being configured to hold the first lumber piece at the joint location of the wooden structure, and the other of the first and second robots being configured to hold a second lumber piece at the joint location of the wooden structure, the second robot including a second articulated robot arm; and

an attachment assembly configured to attach the first lumber piece to the second lumber piece on the assembly table, the attachment assembly being located along the length of the assembly table and extending along the width of the table, the first robot being located along the length of the assembly table on one side of the attachment assembly, and the second robot arm being located along the length of the table on the other side of the attachment assembly.

E2. The assembly station of claim E1, wherein the attachment assembly comprises a movable platen configured to drive a nailing plate into the first and second lumber pieces.

E3. The assembly station of claim E2, wherein the attachment assembly comprises a second movable platen configured to drive a second nailing plate into the first and second lumber pieces.

F1. An automated truss assembly robot comprising:

a support;

a plurality of robot arm members operatively connected to the support, each robot arm member being connected to an adjacent robot arm member by a joint; and

a tool mounted on a distal end of a distal most robot arm member, the tool including a gripper and a suction pad each being configured to retain a lumber piece to the robot for assembling a wooden structure using the robot.

F2. The robot of claim F1, wherein the tool comprises a vision system for capturing an image of indicia on the lumber piece.

G1. A method of positioning a lumber piece having positioning indicia formed thereon on a wooden structure assembly table comprising:

moving a robot arm end effector of a robot to the lumber piece;

acquiring an image of the indicia on the lumber piece using a camera on the robot using the camera;

determining a placement location for the lumber piece on the assembly table based on the acquired image;

securing the lumber piece to the end effector; and

moving the lumber piece to the determined placement location using the robot such that an end of the lumber piece is disposed at the joint location.

G2. The method of claim G1, wherein the lumber piece comprises a first lumber piece and the robot comprises a first robot, the method further comprising releasing the first lumber piece from the first robot and securing the first lumber piece at the determined placement location with a second robot.

G3. The method of claim G2, further comprising:

acquiring an image of the indicia on the lumber piece using a camera on the second robot; and

moving the first lumber piece to the determined location using the second robot.

G4. The method of claim G3, further comprising positioning a second lumber piece at the joint location with the first robot while the second robot holds the first lumber piece at the joint location.

H1. A plate dispensing assembly comprising:

a magazine rack comprising a plurality of magazine slots for receiving stacks of nailing plate pairs;

a plate conveyor comprising a plurality of plate slots for receiving individual nailing plates;

a first plate handling assembly for retrieving nailing plate pairs from the magazine rack and separating the nailing plate pairs; and

a second plate handling assembly for retrieving the individual nailing plates from the plate conveyor.

H2. The assembly of claim H1, wherein the magazine rack comprises a plurality of removable cassettes defining the magazine slots.

H3. The assembly of claim H1, wherein the first plate handling assembly comprises a retriever for retrieving nailing plate pairs from the magazine rack, and a separator for separating the retrieved nailing plate pairs.

H4. The assembly of claim H1, wherein the second plate handling assembly comprises a dispenser in communication with an outlet of the conveyor for dispensing the individual nailing plates.

I1. A method of dispensing nailing plate pair for use in the assembly of a wooden structure including a plurality of lumber pieces, the method comprising:

retrieving a nailing plate pair from a magazine rack using an automated selector of a plate dispensing assembly, the nailing plate pair being arranged teeth-to-teeth;

separating the nailing plate pair a first distance using an automated separator of the plate dispensing assembly;

aligning the separated nailing plate pair; and

separating the aligned nailing plate pair a second distance that is greater than the first distance using the automated separator to position the nailing plates for being removed from the plate dispensing assembly.

I2. The method of claim I1, wherein the separated nailing plate pair is aligned using an automated pusher.

J1. A method of applying nailing plates to an attachment device comprising:

operating a robotic arm to retrieve a nailing plate from a plate dispensing assembly;

grasping teeth of the nailing plate with the robotic arm; and

placing the nailing plate on the attachment device with the robotic arm.

J2. The method of claim J1, further comprising applying a uniformly dispersed magnetic field across a surface of the attachment device to hold the nailing plate on the surface of the attachment device.

K1. An attachment device for use in attaching nailing plates to lumber pieces in the formation of a wooden structure, the device comprising:

a base; and

a platen movably attached to the base, the platen comprising an attachment surface for holding a nailing plate and a magnet holder defining a plurality of receptacles dispersed across the magnet holder, each receptacle receiving a magnet such that the magnets together apply a uniformly dispersed magnetic field across the attachment surface of the platen to hold the nailing plate on the attachment surface of the platen.

K2. The attachment device of claim K1, further comprising a cover plate disposed over the magnet holder and defining the attachment surface.

K3. The attachment device of claim K1, wherein the platen includes at least 5 magnets.

K4. The attachment device of claim K1, wherein the magnets are circular.

L1. An in-feed station for staging lumber prior to being fed to a saw comprising:

an in-feed conveyor for receiving stock lumber, each piece of stock lumber having opposite major surfaces and opposite minor surfaces;

an in-feed buffer table disposed adjacent the in-feed conveyor, the in-feed buffer table including a plurality of holding slots; and

a manipulator moveable over the in-feed conveyor to retrieve the stock lumber from the in-feed conveyor, and moveable over the in-feed buffer table to place the stock lumber in one of the holding slots, the manipulator including a gripper for grasping the stock lumber along a major side surface of the stock lumber such that the stock lumber is transported with its minor surfaces facing upward and downward, and its major surfaces facing horizontally from the in-feed conveyor to the in-feed buffer table.

L2. The in-feed station of claim L1, further comprising a saw in-feed comprising separate first and second delivering lines for delivering two separate lines of stock lumber to a saw.

L3. The in-feed station of claim L1, further comprising sensors for detecting a length and width of the stock lumber.

L4. The in-feed station of claim L1, further comprising a reject conveyor for receiving rejected stock lumber fed into the in-feed conveyor.

L5. The in-feed station of claim L1, further comprising a saw in-feed disposed adjacent the in-feed buffer table, the saw in-feed including a first saw in-feed line and a second saw in-

feed line, the manipulator being configured to deliver stock lumber to both the first and second saw in-feed lines.

L6. The in-feed station of claim L5, wherein the first and second in-feed lines extend parallel to each other.

M1. A manipulator assembly for use at an in-feed station for retrieving and delivering stock lumber to a saw line, the manipulator comprising:

a gantry including a support rail; and

a manipulator attached to the gantry, the manipulator including a carriage moveably attached to the support rail to translate the manipulator across the in-feed station, a base moveably attached to the carriage in a vertical direction to raise and lower the base, and a plurality of grip fingers mounted to the base and moveable between open and closed positions for grasping the stock lumber such that the manipulator is configured to transport the stock lumber across the in-feed station.

M2. The assembly of claim M1, wherein the assembly is free of a piercing device for piercing the stock lumber.

M3. The assembly of claim M1, further comprising sensors for detecting when the base is in a raised or lowered position.

M4. The assembly of claim M1, further comprising sensors for detecting when the grip fingers are in the open and closed positions.

M5. The assembly of claim M1, wherein the grip fingers comprise a first set of grip fingers and a second set of grip fingers having a different configuration than the first set of grip fingers.

N1. A method of handling lumber for being delivered to a saw, the method comprising:

delivering stock lumber to an in-feed station;

staging the stock lumber on a buffer table at the in-feed station;

delivering the staged stock lumber to a saw in-feed including a first saw in-feed line and a second saw in-feed line;

transporting a first stock lumber piece to a saw along the first saw in-feed line; and

transporting a second stock lumber piece to the saw along the second saw in-feed line.

N2. The method of claim N1, wherein the first and second in-feed lines extend parallel to each other.

N3. The method of claim N1, wherein delivering the staged stock lumber to the saw in-feed further comprises first delivering one of the first and second stock lumber pieces to a holding position, then retrieving said stock lumber piece from the holding position and dropping said stock lumber piece into one of the first and second saw in-feed lines.

N4. The method of claim N1, further comprising clamping one of the first and second staged stock lumber piece to straighten said stock lumber piece prior to transporting said stock lumber piece along one of the first and second saw in-feed lines.

N5. The method of claim N1, further comprising transporting one of the first and second stock lumber pieces to a zero location having a predetermined distance from the saw and holding said stock lumber piece at the zero location.

N6. The method of claim N1, further comprising printing fiducials on one of the first and second stock lumber pieces while said stock lumber piece is being transported on one of the first and second saw in-feed lines.

O1. A method of cutting stock lumber pieces for use in assembling a wooden structure, the method comprising:

delivering a first stock lumber piece to a saw along a first saw in-feed line;

cutting the first stock lumber piece with the saw;

delivering a second stock lumber piece to the saw along a second saw in-feed line; and

cutting the second stock lumber piece with the saw.

O2. The method of claim O1, wherein the first and second saw in-feed lines extend parallel to each other.

O3. The method of claim O1, further comprising detecting a position of the first and second stock lumber pieces along the first and second saw in-feed lines.

O4. The method of claim O1, wherein delivering the second stock lumber piece to the saw occurs during the cutting of the first stock lumber piece with the saw.

P1. A method of cutting stock lumber pieces for use in assembling a wooden structure, the method comprising:

determining a distance of a stock lumber piece on a saw in-feed line from a saw;

delivering the stock lumber piece to a first location relative to the saw;

securing the stock lumber piece with a clamp;

measuring a position of the stock lumber piece relative to a reference point on the clamp;

accounting for a difference between the position of the stock lumber piece and the reference point; and

cutting the stock lumber piece based on the difference between the position of the stock lumber piece and the reference point.

P2. The method of claim P1, wherein measuring the position of the stock lumber piece relative to the reference point on the clamp comprises measuring a distance between the reference point and a bottom of the stock lumber piece.

P3. The method of claim P1, further comprising adjusting a cut height based on the difference between the position of the stock lumber piece and the reference point.

Q1. A saw assembly for cutting a stock lumber piece comprising:

a robotic arm;

a saw mounted on the robotic arm such that the saw is configured to cut a stock lumber piece along a plane that intersects a saw in-feed axis; and

a clamp moveable along to the saw in-feed axis and configured to clamp the stock lumber piece in place along the in-feed axis, the clamp including a sensor for measuring a distance between the stock lumber piece and the clamp when the stock lumber piece is clamped by the clamp.

Q2. The saw assembly of claim Q1, wherein the sensor is disposed at a bottom of the clamp and is configured to measure a distance from the sensor to a bottom of the stock lumber piece.

R1. A saw assembly for cutting a stock lumber piece comprising:

a robotic arm;

a saw mounted on the robotic arm such that the saw is configured to cut a stock lumber piece along a plane that intersects a saw in-feed axis; and

a lumber in-feed along which the stock lumber piece moves through the saw;

wherein the robotic are is configured to move on laterally opposite sides of the stock lumber piece for making cuts starting on either side of the stock lumber piece.

S1. A saw assembly for cutting a stock lumber piece comprising:

a saw compartment;

a robotic arm in the saw compartment;

a saw mounted on the robotic arm; and

first and second, spaced apart lumber in-feed lines configured to move the stock lumber piece through the saw compartment;

wherein the robotic are is configured to cut the stock lumber on either the first lumber in-feed line or the second lumber in-feed line.

T1. A buffer table for use in an automated wooden structure assembly comprising:

a platform;

an index conveyor belt movably mounted to the platform generally along a length of the buffer table;

a plurality of first partitions disposed on the index conveyor belt, the first partitions having a first dimension extending along a width of the buffer table and defining a plurality of first slots configured for receiving cut pieces of lumber; and

a plurality of second partitions disposed on the index conveyor belt, the second partitions having a second dimension extending along the width of the buffer table and defining a plurality of second slots configured for receiving cut pieces of lumber, the first dimension being greater than the second dimension such that the first slots are configured to receive longer pieces of lumber than the second slots.

T2. The buffer table of claim T1, further comprising openings in the index conveyor belt.

T3. The buffer table of claim T2, further comprising pushers movable through the openings in the index conveyor belt to push the cut pieces of lumber off the index conveyor belt.

T4. The buffer table of or claim T3 in combination with an assembly table conveyor, the pushers being configured to push the cut pieces of lumber onto the assembly table conveyor.

T5. The buffer table of claim T1, wherein the first partitions comprise fences, and the second partitions comprise plate members.

T6. The buffer table of claim T5, wherein the first partitions are aligned with the second partitions.

T7. The buffer table of claim T1, wherein the partitions form at least 35 separate slots.

T8. The buffer table of claim T7, wherein the partition form 43 separate slots.

T9. The buffer table of claim T1, further comprising a manipulator for placing cut pieces of lumber in the slots and a controller for positioning the cut pieces of lumber in the slots in an order that the cut pieces of lumber will be used to assemble the wooden structure.

U1. A method of calibrating a wooden structure manufacturing and assembly system

comprising:

instructing a first robot to position a tip of a first calibration member at a reference point; and

instructing a second robot to position a tip of a second calibration member at the reference point, calibration of the wooden structure manufacturing and assembly system being indicated when the tip of the first calibration member touches the tip of the second calibration member.

U2. The method of claim U1, further comprising relocating the reference point and calibrating the first and second robots to position the tips of the first and second calibration members and the relocated reference point.

U3. The method of claim U1, further comprising relocating the first and second robots and calibrating the first and second robots to position the tips of the first and second calibration members and the reference point.

V1. An automated wooden structure manufacturing and assembly system controller comprising one or more processors and computer executable instructions embodied on a computer readable storage medium, the computer executable instructions including instructions for controlling the assembly of a wooden structure, the instructions including:

determining an order of lumber pieces to be used during the assembly of the wooden structure;

after determining the order of lumber pieces, determining a placement of nailing plates on the lumber pieces to form joints between the lumber pieces; and

after determining the placement of the nailing plates, determining a sequence of movements of robots to assemble the wooden structure using the lumber pieces.

V2. The controller of claim V1, wherein the movement of the robots is based on a point system assigned to various conditions of the joints of the wooden structure.

V3. The controller of claims V1 or V2, further comprising instructions for coordinating movements of an attachment assembly for attaching the nailing plates to the lumber pieces with the movement of the robots.

V4. The controller of any one of claim V2, wherein a plurality of movement sequences for the robots are analyzed and the movement sequence with the highest point total is selected.

V5. The controller of claim V2, further comprising instructions for determining an order

of the joints to be completed during the assembly of the wooden structure.

V6. The controller of claim V5, wherein the order of the joints to be completed is determined based on at least one of a location of the joint, a number of lumber pieces at the joint, and a size of the lumber pieces at the joint.

V7. The controller of claim V6, wherein a point system is assigned to various assembly actions and conditions, and the order of the joints to be completed having a highest point total is selected to for the assembly of the wooden structure.

W1. A method of selecting stock lumber for use in automated assembly of a wooden structure including a plurality of lumber pieces, the method comprising:

determining an inventory of stock lumber;

selecting the stock lumber needed to produce a first set of lumber piece of the wooden structure based on the inventory; and

determining which lumber pieces in the first set of lumber pieces to produce from each selected stock lumber to maximize the number of lumber pieces produced from the stock lumber.

W2. The method of claim W1, further comprising:

selecting the stock lumber needed to produce a second set of lumber pieces of the wooden structure based on the inventory; and

determining which lumber pieces in the second set of lumber pieces to produce from each selected stock lumber.

W3. The method of claim W1, further comprising determining a grade of the stock lumber to be used to produce the wooden structure.

W4. The method of claim W1, further comprising producing at least three pieces of lumber from a single piece of stock lumber.

X1. A method of selecting stock lumber for use in automated assembly of a wooden structure including a plurality of lumber pieces, the method comprising:

determining an inventory of stock lumber;

selecting the stock lumber needed to produce a first set of lumber piece of the wooden structure based on the inventory by choosing the least expensive lumber required to form a wooden structure having the mechanical properties required of the wooden structure.

Y1. A computer-implemented method of manufacturing and assembling wooden trusses, the method comprising:

receiving, at a truss manufacture computing device, designs for a plurality of wooden structures;

processing, at the truss manufacture computing device, the designs to identify a recipe for manufacturing and assembling each of the plurality of wooden structures defined in each corresponding design;

processing the plurality of recipes, at the truss manufacture computing device, to identify a requested lumber input of stock lumber;

instructing an automated truss manufacturing system to pre-stage the requested lumber input;

instructing a saw assembly to cut the lumber input based, in part, on the identified recipes; and

instructing a plurality of robots to assemble a first wooden truss according to a first recipe of the identified recipe.

Y2. The computer-implemented method of claim Y1, further comprising:

identifying, at the truss manufacture computing device, a plurality of possible sequences to manufacture a first design of the designs;

simulating, at the truss manufacture computing device, manufacture of the first design according to each of the plurality of possible sequences, wherein each simulated manufacture includes simulation characteristics; and

identifying the recipe by identifying the possible sequence having the preferred simulation characteristics.

Y3. The computer-implemented method of claim Y2, wherein the simulation characteristics include expected time to manufacture, expected joint quality, and expected travel time for robots used in manufacture.

Y4. The computer-implemented method of claim Y1, further comprising:

processing the plurality of recipes, at the truss manufacture computing device, to identify lumber instructions; and

identifying the requested lumber input of stock lumber based on the lumber instructions.

Y5. The computer-implemented method of claim Y1, further comprising:

processing the plurality of recipes, at the truss manufacture computing device, to identify lumber instructions; and

instructing a saw assembly to cut the lumber according to the lumber instructions.

Y6. The computer-implemented method of claim Y4, further comprising:

presenting a prompt, to a user at a user interface in communication with the truss manufacture computing device, to feed stock lumber based on the lumber instructions.

Y7. The computer-implemented method of claim Y1, wherein instructing the plurality of robots further comprises:

identifying a joint for assembly from a first recipe of the recipes; and

instructing a pair of assembly robots to position a pair of lumber pieces defining a joint, according to the recipe.

Y8. The computer-implemented method of claim Y7, further comprising:

identifying a connector plate associated with the joint, according to the recipe;

instructing a plate distribution assembly to obtain the connector plate associated with the joint;

instructing a platen assembly to connect the joint using the connector plate.

Z1. An automated truss manufacturing system for manufacturing and assembling wooden trusses, said system comprising:

an in-feed station;

a saw station;

an assembly station having a plurality of robots; and

a truss manufacture computing device having a processor and a memory, said processor in communication with said in-feed station, said saw station, and said assembly station, said processor configured to:

receive designs for a plurality of wooden structures;

process the designs to identify a recipe for manufacturing and assembling each of the plurality of wooden structures defined in each corresponding design;

process the plurality of recipes, at the truss manufacture computing device, to identify a requested lumber input of stock lumber;

instruct said in-feed station to pre-stage the requested lumber input;

instruct said saw station to cut the lumber input based, in part, on the identified recipes; and

instruct the plurality of robots to assemble a first wooden truss according to a first recipe of the identified recipe.

Z2. The automated truss manufacturing system of claim Z1, wherein said processor is configured to:

identify a plurality of possible sequences to manufacture a first design of the designs;

simulate manufacture of the first design according to each of the plurality of possible sequences, wherein each simulated manufacture includes simulation characteristics; and

identify the recipe by identifying the possible sequence having the preferred simulation characteristics.

Z3. The automated truss manufacturing system of claim Z2, wherein the simulation characteristics include expected time to manufacture, expected joint quality, and expected travel time for robots used in manufacture.

Z4. The automated truss manufacturing system of claim Z1, wherein said processor is configured to:

process the plurality of recipes to identify lumber instructions; and

identify the requested lumber input of stock lumber based on the lumber instructions.

Z5. The automated truss manufacturing system of claim Z1, wherein said processor is configured to:

process the plurality of recipes to identify lumber instructions; and

instruct said cutting station to cut the lumber according to the lumber instructions.

Z6. The automated truss manufacturing system of claim Z4, wherein said processor is configured to:

present a prompt, to a user at a user interface in communication with the truss manufacture computing device, to feed stock lumber based on the lumber instructions.

Z7. The automated truss manufacturing system of claim Z1, wherein said processor is configured to:

identify a joint for assembly from a first recipe of the recipes; and

instruct the robots at the assembly station to position a pair of lumber pieces defining a joint, according to the recipe.

Z8. The automated truss manufacturing system of claim Z7, wherein said processor is configured to:

identify a connector plate associated with the joint, according to the recipe;

instruct a plate distribution assembly at the assembly station to obtain the connector plate associated with the joint;

instruct a platen assembly at the assembly station to connect the joint using the connector plate.

AA1. An automated wooden structure manufacturing system for manufacturing and assembling wooden structures, the system comprising:

an in-feed station comprising a first buffer table configured for receiving and staging stock lumber;

a cutting station comprising a saw assembly configured for cutting the stock lumber into multiple lumber pieces;

a buffer station comprising a second buffer table configured for holding a first set of the lumber pieces and a third buffer table configured for holding a second set of the lumber pieces; and

an assembly station comprising an assembly module configured for attaching the lumber pieces together to assemble the wooden structure.

AA2. The system of AA1, wherein the buffer station comprises a

wherein the buffer station comprises a first mechanical manipulator configured to place a first set of the lumber pieces on the second buffer table and a second mechanical manipulator configured to place a second set of the lumber pieces on a third buffer table.

AB1. A computing device comprising:

circuitry configured to:

obtain input data indicative of a wooden structure to be produced by an automated system;

perform, as a function of the obtained input data and present properties of the automated system, validation operations to determine whether an error will be encountered during production of the wooden structure; and

identify, in response to a determination that an error will be encountered and as a function of the validation operations, one or more adjustments to enable production of the wooden structure without the error.

AB2. The computing device of claim AB1, wherein to obtain input data indicative of a wooden structure comprises to obtain input data indicative of a truss to be produced.

AB3. The computing device of claim AB1, wherein to obtain input data indicative of a wooden structure comprises to obtain input data indicative of a set of multiple wooden structures to be produced.

AB4. The computing device of claim AB1, wherein to obtain input data comprises to obtain input data indicative of a target shape, parts, and materials to produce the wooden structure.

AB5. The computing device of claim AB1, wherein to obtain input data comprises to obtain input data indicative of a quantity of the wooden structure to produce.

AB6. The computing device of claim AB1, wherein to perform validation operations comprises to determine whether materials specified in the obtained input data are available to the automated system.

AB7. The computing device of claim AB6, wherein to determine whether materials specified in the obtained input data are available comprises to determine whether nailing plates specified in the obtained input data are available in a nailing plate inventory of the automated system.

AB8. The computing device of claim AB7, wherein to determine whether the nailing plates are available in a nailing plate inventory comprises to determine whether the nailing plates are available in one or more nailing plate magazines of the automated system.

AB9. The computing device of claim AB6, wherein to determine whether materials specified in the obtained input data are available comprises to determine whether lumber specified in the obtained input data is available to the automated system.

AB10. The computing device of claim AB1, wherein to perform validation operations comprises to simulate execution of a recipe indicative of a series of operations of components of the automated system for producing the wooden structure.

AB11. The computing device of claim AB10, wherein to simulate execution of the recipe comprises to simulate execution based on models of components of the automated system.

AB12. The computing device of claim AB11, wherein to simulate execution comprises to simulate execution based on models of assembly robots of the automated system.

AB13. The computing device of claim AB12, wherein the circuitry is further configured to determine, based on the simulated execution, whether one or more of the assembly robots will be unable to create a defined joint of the wooden structure.

AB14. The computing device of claim AB12, wherein the circuitry is further configured to determine, based on the simulated execution, whether one or more of the assembly robots will collide with a component of the automated system.

AB15. The computing device of claim AB12, wherein the circuitry is further configured to determine whether one or more of the assembly robots will be unable to pick up a board or part of the wooden structure at a defined position.

AB16. The computing device of claim AB1, wherein to identify one or more adjustments comprises to:

present the determined error in a user interface; and

receive a user-defined adjustment.

AB17. The computing device of claim AB1, wherein to identify one or more adjustments comprises to identify the adjustment as a function of a predefined set of adjustments mapped to predefined errors.

AB18. The computing device of claim AB1, wherein to identify one or more adjustments comprises to determine an offset, a rotation angle, or a reflection along one or more axes for the wooden structure.

AB19. The computing device of claim AB1, wherein to identify one or more adjustments comprises to determine an adjustment as a function of a set of materials available to the automated system.

AB20. The computing device of claim AB19, wherein to determine an adjustment as a function of a set of materials available to the automated system comprises to identify a replacement nailing plate as a function of dimensions of a nailing plate specified in the input data and dimensions of one or more nailing plates available to the automated system.

AB21. The computing device of claim AB20, wherein to identify a replacement nailing plate as a function of dimensions of the nailing plate specified in the input data comprises to identify, as a replacement nailing plate, a nailing plate that is available to the automated system and that has dimensions that are greater than or equal to the dimensions of the nailing plate specified in the input data.

AB22. The computing device of claim AB1, wherein to determine an adjustment as a function of a set of materials available to the automated system comprises to identify replacement lumber as a function of a grade of lumber specified in the input data and one or more grades of lumber available to the automated system.

AB23. The computing device of claim AB22, wherein to identify replacement lumber as a function of a grade of lumber specified in the input data comprises to identify, as replacement lumber, lumber that is available to the automated system and that has a grade that is greater than or equal to the grade of the lumber specified in the input data.

AB24. The computing device of claim AB1, wherein the circuitry is further configured to:

store the one or more adjustments; and

simulate execution of a recipe for producing the wooden structure with the one or more adjustments applied to operations of components of the automated system defined in the recipe.

AC1. A method comprising:

obtaining, by a computing device, input data indicative of a wooden structure to be produced by an automated system;

performing, by the computing device and as a function of the obtained input data and present properties of the automated system, validation operations to determine whether an error will be encountered during production of the wooden structure; and

identifying, by the computing device and in response to a determination that an error will be encountered and as a function of the validation operations, one or more adjustments to enable production of the wooden structure without the error.

AC2. The method of claim AC1, wherein obtaining input data indicative of a wooden structure comprises obtaining input data indicative of a truss to be produced.

AC3. The method of claim AC1, wherein obtaining input data indicative of a wooden structure comprises obtaining input data indicative of a set of multiple wooden structures to be produced.

AC4. The method of claim AC1, wherein obtaining input data comprises obtaining input data indicative of a target shape, parts, and materials to produce the wooden structure.

AC5. The method of claim AC1, wherein obtaining input data comprises obtaining input data indicative of a quantity of the wooden structure to produce.

AC6. The method of claim AC1, wherein performing validation operations comprises determining whether materials specified in the obtained input data are available to the automated system.

AC7. The method of claim AC6, wherein determining whether materials specified in the obtained input data are available comprises determining whether nailing plates specified in the obtained input data are available in a nailing plate inventory of the automated system.

AC8. The method of claim AC7, wherein determining whether the nailing plates are available in a nailing plate inventory comprises determining whether the nailing plates are available in one or more nailing plate magazines of the automated system.

AC9. The method of claim AC6, wherein determining whether materials specified in the obtained input data are available comprises determining whether lumber specified in the obtained input data is available to the automated system.

AC10. The method of claim AC1, wherein performing validation operations comprises simulating execution of a recipe indicative of a series of operations of components of the automated system for producing the wooden structure.

AC11. The method of claim AC10, wherein simulating execution of the recipe comprises simulating execution based on models of components of the automated system.

AC12. The method of claim AC11, wherein simulating execution comprises simulating execution based on models of assembly robots of the automated system.

AC13. The method of claim AC12, further comprising determining, by the computing device and based on the simulated execution, whether one or more of the assembly robots will be unable to create a defined joint of the wooden structure.

AC14. The method of claim AC12, further comprising determining, by the computing device and based on the simulated execution, whether one or more of the assembly robots will collide with a component of the automated system.

AC15. The method of claim AC12, further comprising determining, by the computing device, whether one or more of the assembly robots will be unable to pick up a board or part of the wooden structure at a defined position.

AC16. The method of claim AC1, wherein identifying one or more adjustments comprises:

presenting, by the computing device, the determined error in a user interface; and

receiving, by the computing device, a user-defined adjustment.

AC17. The method of claim AC1, wherein identifying one or more adjustments comprises identifying the adjustment as a function of a predefined set of adjustments mapped to predefined errors.

AC18. The method of claim AC1, wherein identifying one or more adjustments comprises determining an offset, a rotation angle, or a reflection along one or more axes for the wooden structure.

AC19. The method of claim AC1, wherein identifying one or more adjustments comprises determining an adjustment as a function of a set of materials available to the automated system.

AC20. The method of claim AC19, wherein determining an adjustment as a function of a set of materials available to the automated system comprises identifying a replacement nailing plate as a function of dimensions of a nailing plate specified in the input data and dimensions of one or more nailing plates available to the automated system.

AC21. The method of claim AC20, wherein identifying a replacement nailing plate as a function of dimensions of the nailing plate specified in the input data comprises identifying, as a replacement nailing plate, a nailing plate that is available to the automated system and that has dimensions that are greater than or equal to the dimensions of the nailing plate specified in the input data.

AC22. The method of claim AC1, wherein determining an adjustment as a function of a set of materials available to the automated system comprises identifying replacement lumber as a

function of a grade of lumber specified in the input data and one or more grades of lumber available to the automated system.

AC23. The method of claim AC22, wherein identifying replacement lumber as a function of a grade of lumber specified in the input data comprises identifying, as replacement lumber, lumber that is available to the automated system and that has a grade that is greater than or equal to the grade of the lumber specified in the input data.

AC24. The method of claim AC1, further comprising:

storing, by the computing device, the one or more adjustments; and

simulating, by the computing device, execution of a recipe for producing the wooden structure with the one or more adjustments applied to operations of components of the automated system defined in the recipe.

AD1. One or more machine-readable storage media comprising a plurality of instructions stored thereon that, in response to being executed, cause a computing device to perform the method of any of claims AC1-AC24.

AE1. A computing device comprising:

circuitry configured to:

obtain recipe generation input data indicative of one or more parameters to be satisfied in coordinating operations of components of an automated system to produce one or more wooden trusses;

identify, as a function of the obtained recipe generation input data, one or more unique wooden trusses to be produced; and

generate, for each unique wooden truss, a recipe indicative of operations to be performed by the components of the automated system to produce the corresponding wooden truss in a joint-by-joint extrusion sequence.

AE2. The computing device of claim AE1, wherein to obtain recipe generation input data comprises to obtain shape data indicative of a target shape for a wooden truss.

AE3. The computing device of claim AE2, wherein to obtain recipe generation input data comprises to additionally obtain one or more of production data indicative of quantities of the unique wooden trusses to produce, intervention data indicative of substitute materials to be used or a rotation to be applied to one or more wooden truss parts, or assembler parameter data indicative of one or more offsets, dimensions, or limits for a component of the automated system.

AE4. The computing device of claim AE1, wherein to identify one or more unique trusses comprises to identify the one or more unique trusses as a function of a truss identifier, a job name, a truss label, or a batch name.

AE5. The computing device of claim AE1, wherein to identify one or more unique trusses comprises to filter out non-unique trusses from the obtained recipe generation input data to reduce an amount of time for recipe generation.

AE6. The computing device of claim AE1, wherein to generate a recipe indicative of operations to be performed comprises to:

produce a list of joints for a selected wooden truss;

determine, from the produced list of joints, an ordered set of joints;

determine, as a function of the ordered set of joints, a set of assembly operations; and

calculate, for each assembly operation in the set of assembly operations, a set of recipe operations.

AE7. The computing device of claim AE6, wherein to calculate recipe operations comprises to:

calculate a set of valid primary robot pickup options indicative of locations or orientations that would enable a primary robot of the automated system to pick up a part of the wooden truss for assembly;

calculate a set of valid secondary robot pickup options indicative of locations or orientations that would enable a secondary robot of the automated system to pick up a part of the wooden truss for assembly;

determine pairs of pickup options from the valid primary robot pickup options and valid secondary robot pickup options;

determine, as a function of target criteria, a pickup option pair from the set to be designated as a best pickup option pair.

AE8. The computing device of claim AE7, wherein the circuitry is further configured to:

determine, after the determination of the best pickup option pair, a set of recipe operations deemed to be necessary; and

determine a set of intermediate recipe operations.

AE9. The computing device of claim AE6, wherein to produce a list of joints for the selected wooden truss comprises to:

convert truss shape data associated with the selected wooden truss from the obtained recipe generation input data to points and polygons in a two-dimensional orthogonal coordinate system;

apply one or more interventions defined in intervention data in the obtained recipe generation input data to the converted truss shape data to adjust an orientation associated with the selected wooden truss;

convert part angles defined in the converted truss shape data to a predefined range;

shorten lengths defined in the converted truss shape data for a subset of parts of the selected wooden truss for production tolerance; and

generate, for each of multiple nailing plates associated with the selected wooden truss, a set of one or more joint objects indicative of joints of the selected wooden truss.

AE10. The computing device of claim AE9, wherein to apply one or more interventions comprises to apply one or more rotations, flip the selected wooden truss along an axis or the two-dimensional coordinate system, or apply an offset to the selected wooden truss.

AE11. The computing device of claim AE9, wherein to shorten lengths for a subset of the parts comprises to:

identify a set of parts that are not perimeter parts and not wedge parts; and

shorten non-vertical parts in the set by a greater amount than vertical parts.

AE12. The computing device of claim AE11, wherein to shorten non-vertical parts by a greater amount than vertical parts comprises to:

shorten non-vertical parts by one eighth of an inch; and

shorten vertical parts by one sixteenth of an inch.

AE13. The computing device of claim AE9, wherein to generate, for each of multiple nailing plates associated with the selected wooden truss, a set of one or more joint objects, comprises to:

temporarily create, in a memory of the computing device and for each nailing plate, a rectangle having dimensions that are smaller than the corresponding nailing plate to determine an association between the nailing plates and parts of the wooden truss; and

generate joint objects that include data indicative of one or more nailing plates and parts of the wooden truss associated with each corresponding joint.

AE14. The computing device of claim AE6, wherein to produce a list of joints for the selected wooden truss comprises to:

calculate a first joint for the selected wooden truss; and

calculate an order for remaining joints of the selected wooden truss.

AE15. The computing device of claim AE14, wherein to calculate a first joint for the selected wooden truss comprises to:

determine, as a function of a set of criteria, a best first part in each joint;

determine an order for remaining parts in each joint;

calculate a joint score for each joint; and

designate the first joint as the joint with the highest joint score.

AE16. The computing device of claim AE15, wherein to determine the best first part in a joint comprises to iterate through each of the parts associated with the joint and determine whether a part associated with a present iteration is the best first part as a function of whether the part is a wedge, whether the part satisfies a predefined length associated with a clamp of the automated system, whether the part has an angle that satisfies a predefined range, whether an overlap between the part and an assembly table of the automated system satisfies a predefined percentage, whether

the part defines a perimeter of the selected wooden truss, and whether the part is within a predefined zone in the automated system.

AE17. The computing device of claim AE15, wherein to determine an order for remaining parts in each joint comprises, for a selected joint, to:

determine a part score for each remaining part associated with the selected joint; and

order the remaining parts as a function of the determined part score for each part.

AE18. The computing device of claim AE17, wherein to determine a part score comprises to:

adjust the part score as a function of whether the selected part touches one or more other parts positioned in the selected joint;

adjust the part score as a function of whether the selected part defines a perimeter of the selected wooden truss; and

adjust the part score as a function of whether the selected part defines a wedge of the selected wooden truss.

AE19. The computing device of claim AE18, wherein to adjust the part score as a function of whether the selected part touches one or more other parts positioned in the selected joint comprises to increase the part score based on a number of parts in the selected joint that are touched by the selected part.

AE20. The computing device of claim AE15, wherein to calculate the joint score for a selected joint comprises to:

initially set the joint score for the selected joint to zero;

increase the joint score as a function of joint score factors, wherein the joint score factors include one or more of whether a first part of the joint defines a wedge, whether a second part of the joint defines a wedge, whether the first part of the joint defines a perimeter of the selected wooden truss, whether the second part of the joint defines a perimeter of the selected wooden truss, whether the first part of the joint satisfies a predefined length to be held by a clamp of the automated system, whether the second part of the joint satisfied the predefined length to be

held by the clamp, a percentage of the first part of the joint that overlaps a surface of an assembly table of the automated system, a percentage of the second part of the joint that overlaps the surface of the assembly table of the automated system, or an angle between the first part and the second part of the joint.

AE21. The computing device of claim AE14, wherein to calculate an order for remaining joints of the selected wooden truss comprises, for each joint in a set of remaining joints associated with the selected wooden truss, to:

select the joint for analysis;

determine whether the selected joint is a next best joint for the order of remaining joints as a function of one or more of whether the selected joint will cause a truss collision if picked later, whether all parts of the joint are already positions with one or more previous joints in the joint order, whether the selected joint defines a splice, whether the selected joint has more parts than another joint determined to be the next best joint, whether the selected joint is closer to a defined starting point associated with the selected wooden truss, or whether the selected joint has a higher total length of parts than another joint determined to be the next best joint.

AE22. The computing device of claim AE21, wherein to determine whether the selected joint is the next best joint as function of whether the selected joint has more parts than another joint determined to be the next best joint comprises to determine that the selected joint is the next best join in response to a determination that the selected joint has more parts that define a perimeter of the selected wooden truss within a predefined reach distance of a robot of the automated system than the other joint determined to be the next best joint.

AE23. The computing device of claim AE6, wherein to determine, as a function of the ordered set of joints, a set of assembly operations, comprises, for each joint, to:

select the joint for analysis;

determine the assembly operations for the selected joint as a function of whether the selected joint has two parts; and

add, for each part in the joint, one or more assembly operations for a new part.

AE24. The computing device of claim AE23, wherein to determine the assembly operations for the selected joint as a function of whether the selected joint has two parts comprises to determine, in response to a determination that the joint has two parts, assembly operations as a function of whether the joint has a bottom plate.

AE25. The computing device of claim AE24, wherein to determine the assembly operations as a function of whether the joint has a bottom plate comprises to add, in response to a determination that the joint has a bottom plate and that the joint has all parts, a top plate for the joint as a new assembly operation.

AE26. The computing device of claim AE24, wherein to determine the assembly operations as a function of whether the joint has a bottom plate comprises to determine, in response to a determination that the joint does not have a bottom plate, a set of corresponding assembly operations.

AE27. The computing device of claim AE26, wherein to determine a set of corresponding assembly operations comprises to:

add, in response to a determination that the joint does not have all parts, a bottom plate for the joint as a new assembly operation; or

add, in response to a determination that the joint does have all parts, a top plate and a bottom plate to the joint as new assembly operations.

AE28. The computing device of claim AE23, wherein to add, for each part in the joint, one or more assembly operations for a new part comprises to:

select a part for analysis,

for each remaining joint of the selected wooden truss, select a next other joint; and

determine one or more assembly operations as a function of whether the selected part is a third part of another joint.

AE29. The computing device of claim AE28, wherein to determine one or more assembly operations as a function of whether the selected part is a third part comprises to determine, in

response to a determination that the selected part is a third part, the one or more assembly operations as a function of whether the other joint has a bottom plate.

AE30. The computing device of claim AE29, wherein to determine the one or more assembly operations as a function of whether the other joint has a bottom plate comprises to:

add, in response to a determination that the other joint has a bottom plate and the other joint has all parts, a new assembly operation to add the top plate for the other joint; or

determine, in response to a determination that the other joint does not have a bottom plate, one or more assembly operations as a function of whether the other joint has all parts.

AE31. The computing device of claim AE30, wherein to determine one or more assembly operations as a function of whether the other joint has all parts comprises to:

add, in response to a determination that the other joint does not have all parts, an assembly operation to add the bottom plate for the other joint; or

add, in response to a determination that the other joint does have all parts, an assembly operation to add a top plate and the bottom plate to the other joint.

AE32. The computing device of claim AE28, wherein to add, for each part in the joint, one or more assembly operations further comprises to determine assembly operations as a function of whether the selected joint has a bottom plate.

AE33. The computing device of claim AE7, wherein to calculate a set of valid primary robot pickup options comprises to:

calculate a set of pickup points for the part; and

calculate, in response to a determination that the robot is to pick up a new part, and for each of a set of available other joints in the selected wooden truss, and as a function of whether the other joint is within a defined distance of a present joint and whether the other joint is connected to the part to be picked up by the robot, a set of pickup points for each other part in the selected other joint.

AE34. The computing device of claim AE33, wherein to calculate a set of pickup points comprises to:

calculate a closest pickup point that does will not cause interference with operations of a press of the automated system; and

determine, for each of multiple directions from a center of the part, a set of valid pickup points as a function of a prioritization of pickup options for utilization of a clamp, a utilization of suction device, and movement of at least one component of the automated system to avoid a collision with the robot or the part.

AE35. The computing device of claim AE34, wherein the circuitry is further configured to calculate, for each pickup option in a set of pickup options for a part, an interference polygon for a first tool of the robot, an interference polygon for a second tool of the robot, a tool center interference polygon, a cylindrical combined interference polygon, an effective tool pickup area, and a tool interference with part polygon.

AE36. The computing device of claim AE35, wherein the circuitry is further configured, for each of a set of pickup options and for each of multiple increments of length along a part of the selected wooden truss, to:

calculate a final tool position interference;

calculate a tool trajectory interference;

calculate a press trajectory interference; and

determine a set of subsequent operations as a function of the calculated interferences.

AE37. The computing device of claim AE36, wherein to determine a set of subsequent operations comprises to determine which of a set of directions of movement of a component within the automated system will avoid a collision in picking up or moving a part of the selected wooden truss.

AE38. The computing device of claim AE1, wherein the circuitry is further configured to:

assign primary and secondary roles to each of a first robot and a second robot of the automated system as a function of a set of role assignment factors; and

calculate gantry and robot arm positions for the first robot and the second robot.

AE39. The computing device of claim AE38, wherein to assign primary and secondary roles as a function of role assignment factors comprises to assign roles as a function of whether a new part is being added in a current assembly operation, assign roles as a function of whether only one robot is needed for the current assembly operation, or assign roles a function of relative locations of tools of each of the first and second robots.

AE40. The computing device of claim AE38, wherein to calculate gantry and robot arm positions for the first robot and the second robot comprises to:

calculate a second robot arm polygon;

calculate a press displacement;

calculate a first robot trajectory and a second robot trajectory;

calculate a first robot gantry position and a second robot gantry position; and

calculate, as a function of a gantry y position and final tool position data, first robot and second robot arm orientations.

AE41. The computing device of claim AE8, wherein to determine, after the determination of the best pickup option pair, a set of recipe operations deemed to be necessary comprises to determine necessary recipe operations as a function of whether a part to be added in a current operation exceeds a defined length or weight limit, whether rotation of the part will result in interference, and a target number of presses to be applied to a nailing plate based on a relative size of a surface of a press of the automated system to the nailing plate.

AF1. A method comprising:

obtaining, by a computing device, recipe generation input data indicative of one or more parameters to be satisfied in coordinating operations of components of an automated system to produce one or more wooden trusses;

identifying, by the computing device and as a function of the obtained recipe generation input data, one or more unique wooden trusses to be produced; and

generating, by the computing device and for each unique wooden truss, a recipe indicative of operations to be performed by the components of the automated system to produce the corresponding wooden truss in a joint-by-joint extrusion sequence.

AF2. The method of claim AF1, wherein obtaining recipe generation input data comprises obtaining shape data indicative of a target shape for a wooden truss.

AF3. The method of claim AF2, wherein obtaining recipe generation input data comprises additionally obtaining one or more of production data indicative of quantities of the unique wooden trusses to produce, intervention data indicative of substitute materials to be used or a rotation to be applied to one or more wooden truss parts, or assembler parameter data indicative of one or more offsets, dimensions, or limits for a component of the automated system.

AF4. The method of claim AF1, wherein identifying one or more unique trusses comprises identifying the one or more unique trusses as a function of a truss identifier, a job name, a truss label, or a batch name.

AF5. The method of claim AF1, wherein identifying one or more unique trusses comprises filtering out non-unique trusses from the obtained recipe generation input data to reduce an amount of time for recipe generation.

AF6. The method of claim AF1, wherein generating a recipe indicative of operations to be performed comprises:

producing a list of joints for a selected wooden truss;

determining, from the produced list of joints, an ordered set of joints;

determining, as a function of the ordered set of joints, a set of assembly operations; and

calculating, for each assembly operation in the set of assembly operations, a set of recipe operations.

AF7. The method of claim AF6, wherein calculating recipe operations comprises:

calculating a set of valid primary robot pickup options indicative of locations or orientations that would enable a primary robot of the automated system to pick up a part of the wooden truss for assembly;

calculating a set of valid secondary robot pickup options indicative of locations or orientations that would enable a secondary robot of the automated system to pick up a part of the wooden truss for assembly;

determining pairs of pickup options from the valid primary robot pickup options and valid secondary robot pickup options;

determining, as a function of target criteria, a pickup option pair from the set to be designated as a best pickup option pair.

AF8. The method of claim AF7, further comprising:

determining, by the computing device and after the determination of the best pickup option pair, a set of recipe operations deemed to be necessary; and

determining, by the computing device, a set of intermediate recipe operations.

AF9. The method of claim AF6, wherein producing a list of joints for the selected wooden truss comprises:

converting truss shape data associated with the selected wooden truss from the obtained recipe generation input data to points and polygons in a two-dimensional orthogonal coordinate system;

applying one or more interventions defined in intervention data in the obtained recipe generation input data to the converted truss shape data to adjust an orientation associated with the selected wooden truss;

converting part angles defined in the converted truss shape data to a predefined range;

shortening lengths defined in the converted truss shape data for a subset of parts of the selected wooden truss for production tolerance; and

generating, for each of multiple nailing plates associated with the selected wooden truss, a set of one or more joint objects indicative of joints of the selected wooden truss.

AF10. The method of claim AF9, wherein applying one or more interventions comprises applying one or more rotations, flipping the selected wooden truss along an axis or the two-dimensional coordinate system, or applying an offset to the selected wooden truss.

AF11. The method of claim AF9, wherein shortening lengths for a subset of the parts comprises:

identifying a set of parts that are not perimeter parts and not wedge parts; and

shortening non-vertical parts in the set by a greater amount than vertical parts.

AF12. The method of claim AF11, wherein shortening non-vertical parts by a greater amount than vertical parts comprises:

shortening non-vertical parts by one eighth of an inch; and

shortening vertical parts by one sixteenth of an inch.

AF13. The method of claim AF9, wherein generating, for each of multiple nailing plates associated with the selected wooden truss, a set of one or more joint objects, comprises:

temporarily creating, in a memory of the computing device and for each nailing plate, a rectangle having dimensions that are smaller than the corresponding nailing plate to determine an association between the nailing plates and parts of the wooden truss; and

generating joint objects that include data indicative of one or more nailing plates and parts of the wooden truss associated with each corresponding joint.

AF14. The method of claim AF6, wherein to produce a list of joints for the selected wooden truss comprises:

calculating a first joint for the selected wooden truss; and

calculating an order for remaining joints of the selected wooden truss.

AF15. The method of claim AF14, wherein calculating a first joint for the selected wooden truss comprises:

determining, as a function of a set of criteria, a best first part in each joint;

determining an order for remaining parts in each joint;

calculating a joint score for each joint; and

designating the first joint as the joint with the highest joint score.

AF16. The method of claim AF15, wherein determining the best first part in a joint comprises iterating through each of the parts associated with the joint and determining whether a

part associated with a present iteration is the best first part as a function of whether the part is a wedge, whether the part satisfies a predefined length associated with a clamp of the automated system, whether the part has an angle that satisfies a predefined range, whether an overlap between the part and an assembly table of the automated system satisfies a predefined percentage, whether the part defines a perimeter of the selected wooden truss, and whether the part is within a predefined zone in the automated system.

AF17. The method of claim AF15, wherein determining an order for remaining parts in each joint comprises, for a selected joint:

determining a part score for each remaining part associated with the selected joint; and

ordering the remaining parts as a function of the determined part score for each part.

AF18. The method of claim AF17, wherein determining a part score comprises:

adjusting the part score as a function of whether the selected part touches one or more other parts positioned in the selected joint;

adjusting the part score as a function of whether the selected part defines a perimeter of the selected wooden truss; and

adjusting the part score as a function of whether the selected part defines a wedge of the selected wooden truss.

AF19. The method of claim AF18, wherein adjusting the part score as a function of whether the selected part touches one or more other parts positioned in the selected joint comprises increasing the part score based on a number of parts in the selected joint that are touched by the selected part.

AF20. The method of claim AF15, wherein calculating the joint score for a selected joint comprises:

initially setting the joint score for the selected joint to zero;

increasing the joint score as a function of joint score factors, wherein the joint score factors include one or more of whether a first part of the joint defines a wedge, whether a second

part of the joint defines a wedge, whether the first part of the joint defines a perimeter of the selected wooden truss, whether the second part of the joint defines a perimeter of the selected wooden truss, whether the first part of the joint satisfies a predefined length to be held by a clamp of the automated system, whether the second part of the joint satisfied the predefined length to be held by the clamp, a percentage of the first part of the joint that overlaps a surface of an assembly table of the automated system, a percentage of the second part of the joint that overlaps the surface of the assembly table of the automated system, or an angle between the first part and the second part of the joint.

AF21. The method of claim AF14, wherein calculating an order for remaining joints of the selected wooden truss comprises, for each joint in a set of remaining joints associated with the selected wooden truss:

selecting the joint for analysis;

determining whether the selected joint is a next best joint for the order of remaining joints as a function of one or more of whether the selected joint will cause a truss collision if picked later, whether all parts of the joint are already positioned with one or more previous joints in the joint order, whether the selected joint defines a splice, whether the selected joint has more parts than another joint determined to be the next best joint, whether the selected joint is closer to a defined starting point associated with the selected wooden truss, or whether the selected joint has a higher total length of parts than another joint determined to be the next best joint.

AF22. The method of claim AF21, wherein determining whether the selected joint is the next best joint as function of whether the selected joint has more parts than another joint determined to be the next best joint comprises determining that the selected joint is the next best join in response to a determination that the selected joint has more parts that define a perimeter of the selected wooden truss within a predefined reach distance of a robot of the automated system than the other joint determined to be the next best joint.

AF23. The method of claim AF6, wherein determining, as a function of the ordered set of joints, a set of assembly operations, comprises, for each joint:

selecting the joint for analysis;

determining the assembly operations for the selected joint as a function of whether the selected joint has two parts; and

adding, for each part in the joint, one or more assembly operations for a new part.

AF24. The method of claim AF23, wherein determining the assembly operations for the selected joint as a function of whether the selected joint has two parts comprises determining, in response to a determination that the joint has two parts, assembly operations as a function of whether the joint has a bottom plate.

AF25. The method of claim AF24, wherein determining the assembly operations as a function of whether the joint has a bottom plate comprises adding, in response to a determination that the joint has a bottom plate and that the joint has all parts, a top plate for the joint as a new assembly operation.

AF26. The method of claim AF24, wherein determining the assembly operations as a function of whether the joint has a bottom plate comprises determining, in response to a determination that the joint does not have a bottom plate, a set of corresponding assembly operations.

AF27. The method of claim AF26, wherein determining a set of corresponding assembly operations comprises:

adding, in response to a determination that the joint does not have all parts, a bottom plate for the joint as a new assembly operation; or

adding, in response to a determination that the joint does have all parts, a top plate and a bottom plate to the joint as new assembly operations.

AF28. The method of claim AF23, wherein adding, for each part in the joint, one or more assembly operations for a new part comprises:

selecting a part for analysis,

for each remaining joint of the selected wooden truss, selecting a next other joint; and

determining one or more assembly operations as a function of whether the selected part is a third part of another joint.

AF29. The method of claim AF28, wherein determining one or more assembly operations as a function of whether the selected part is a third part comprises determining, in response to a determination that the selected part is a third part, the one or more assembly operations as a function of whether the other joint has a bottom plate.

AF30. The method of claim AF29, wherein determining the one or more assembly operations as a function of whether the other joint has a bottom plate comprises:

adding, in response to a determination that the other joint has a bottom plate and the other joint has all parts, a new assembly operation to add the top plate for the other joint; or

determining, in response to a determination that the other joint does not have a bottom plate, one or more assembly operations as a function of whether the other joint has all parts.

AF31. The method of claim AF30, wherein determining one or more assembly operations as a function of whether the other joint has all parts comprises:

adding, in response to a determination that the other joint does not have all parts, an assembly operation to add the bottom plate for the other joint; or

adding, in response to a determination that the other joint does have all parts, an assembly operation to add a top plate and the bottom plate to the other joint.

AF32. The method of claim AF28, wherein adding, for each part in the joint, one or more assembly operations further comprises determining assembly operations as a function of whether the selected joint has a bottom plate.

AF33. The method of claim AF7, wherein calculating a set of valid primary robot pickup options comprises:

calculating a set of pickup points for the part; and

calculating, in response to a determination that the robot is to pick up a new part, and for each of a set of available other joints in the selected wooden truss, and as a function of whether the other joint is within a defined distance of a present joint and whether the other joint is

connected to the part to be picked up by the robot, a set of pickup points for each other part in the selected other joint.

AF34. The method of claim AF33, wherein calculating a set of pickup points comprises:

calculating a closest pickup point that does will not cause interference with operations of a press of the automated system; and

determining, for each of multiple directions from a center of the part, a set of valid pickup points as a function of a prioritization of pickup options for utilization of a clamp, a utilization of suction device, and movement of at least one component of the automated system to avoid a collision with the robot or the part.

AF35. The method of claim AF34, further comprising calculating, by the computing device and for each pickup option in a set of pickup options for a part, an interference polygon for a first tool of the robot, an interference polygon for a second tool of the robot, a tool center interference polygon, a cylindrical combined interference polygon, an effective tool pickup area, and a tool interference with part polygon.

AF36. The method of claim AF35, further comprising, for each of a set of pickup options and for each of multiple increments of length along a part of the selected wooden truss:

calculating a final tool position interference;

calculating a tool trajectory interference;

calculating a press trajectory interference; and

determining a set of subsequent operations as a function of the calculated interferences.

AF37. The method of claim AF36, wherein determining a set of subsequent operations comprises determining which of a set of directions of movement of a component within the automated system will avoid a collision in picking up or moving a part of the selected wooden truss.

AF38. The method of claim AF1, further comprising:

assigning, by the computing device, primary and secondary roles to each of a first robot and a second robot of the automated system as a function of a set of role assignment factors; and

calculating, by the computing device, gantry and robot arm positions for the first robot and the second robot.

AF39. The method of claim AF38, wherein assigning primary and secondary roles as a function of role assignment factors comprises assigning roles as a function of whether a new part is being added in a current assembly operation, assigning roles as a function of whether only one robot is needed for the current assembly operation, or assigning roles a function of relative locations of tools of each of the first and second robots.

AF40. The method of claim AF38, wherein calculating gantry and robot arm positions for the first robot and the second robot comprises:

calculating a second robot arm polygon;

calculating a press displacement;

calculating a first robot trajectory and a second robot trajectory;

calculating a first robot gantry position and a second robot gantry position; and

calculating, as a function of a gantry y position and final tool position data, first robot and second robot arm orientations.

AF41. The method of claim AF8, wherein determining, after the determination of the best pickup option pair, a set of recipe operations deemed to be necessary comprises determining necessary recipe operations as a function of whether a part to be added in a current operation exceeds a defined length or weight limit, whether rotation of the part will result in interference, and a target number of presses to be applied to a nailing plate based on a relative size of a surface of a press of the automated system to the nailing plate.

AG1. One or more machine-readable storage media comprising a plurality of instructions stored thereon that, in response to being executed, cause a computing device to perform the method of any of claims AF1-AF41.

AH1. A computing device comprising:

circuitry configured to:

obtain lumber selection input data indicative of one or more parameters to be utilized in the selection of lumber for the production of one or more wooden trusses by an automated system; and

select, as a function of the one or more parameters and characteristics of lumber available in a lumber inventory of the automated system, a set of lumber pieces from the lumber inventory to satisfy an efficiency target in the automated production of the one or more wooden trusses.

AH2. The computing device of claim AH1, wherein to obtain lumber selection input data comprises to obtain one or more of assembly recipe data indicative of operations to be performed by the automated system to produce the one or more wooden trusses, production data indicative of quantities of the wooden trusses to be produced in one or more batches, line data indicative of a status of one or more in-feed lines of the automated system, inventory data indicative of the characteristics of lumber available in the lumber inventory, saw parameter data indicative of parameters of a saw of the automated system to be used to cut the lumber, or standard length configuration data indicative of one or more lengths defined for pieces of lumber to be used by the automated system.

AH3. The computing device of claim AH1, wherein the circuitry is further to determine, as a function of the lumber selection input data, a set of parts to be used in the production of wooden trusses in a batch.

AH4. The computing device of claim AH3, wherein the circuitry is further configured to:

determine a set of boards for potential use in a batch of one or more wooden trusses to be produced;

determine a set of potential parts for each board in the set of boards, including identifying, as a function of the parameters, a best board from which to produce the potential parts from lumber in the lumber inventory having a grade equal to a grade specified in connection with a design for a wooden truss to be produced and from lumber in the lumber inventory having a grade that is greater than the grade specified in connection with the design; and

remove, from a list of remaining parts to be produced, one or more of the potential parts to be produced from the determined best board.

AH5. The computing device of claim AH4, wherein the circuitry is further configured to determine the best board to be used for each of the remaining parts to be produced.

AH6. The computing device of claim AH3, wherein to determine the set of parts to be used in the production of wooden trusses in the batch comprises to:

determine whether the batch is queued for a first line or a second line of the automated system;

determine, in response to a determination that the batch is queued for the first line or the second line, whether a wooden truss remains to be produced from an assembly recipe defined in the lumber selection input data;

determine, in response to a determination that a wooden truss remains to be produced from the assembly recipe, whether a part remains to be produced for the wooden truss; and

calculate, in response to a determination that a part remains to be produced for the wooden truss, a pickup location indicative of a length along the part where a robot of the automated system will pick up the part.

AH7. The computing device of claim AH6, wherein the circuitry is further configured to:

determine whether the pickup location satisfies a predefined length threshold; and

rotate, in response to a determination that the pickup location does not satisfy the predefined length threshold, the part by 180 degrees.

AH8. The computing device of claim AH6, wherein the circuitry is further configured to:

determine, in response to a determination that the pickup location does not satisfy the predefined length threshold, whether the pickup location satisfies a percentage of part length threshold; and

rotate, in response to a determination that the pickup location satisfies the percentage of part length threshold, the part by 180 degrees.

AH9. The computing device of claim AH8, wherein to determine whether the pickup location satisfies the percentage of part length threshold comprises to determine whether the pickup location is greater than 70% of the part length.

AH10. The computing device of claim AH8, wherein the circuitry is further configured to determine, in response to a determination that the pickup location does not satisfy the percentage of part length threshold, not to rotate the part.

AH11. The computing device of claim AH4, wherein to determine a set of boards for potential use in the batch comprises to:

select a board from the lumber inventory;

determine whether a width of the selected board satisfies a width threshold for a part in the batch to be produced from the selected board;

determine, in response to a determination that the width of the selected board satisfies the width threshold, whether a length of the selected board satisfies a length threshold for the part; and

determine, in response to a determination that the length satisfies the length threshold, a set of one or more responsive operations as a function of a comparison between a grade of the selected board and a defined grade for the part.

AH12. The computing device of claim AH11, wherein the circuitry is further configured to exclude the board from the set of boards for potential use in response to a determination that the width of the selected board does not satisfy the width threshold or that the length of the selected board does not satisfy the length threshold.

AH13. The computing device of claim AH11, wherein to determine a set of one or more responsive operations as a function of a comparison between the grade of the selected board and the defined grade for the part comprises to:

determine whether the grade of the selected board is equal to the defined grade for the part;

determine, in response to a determination that the grade of the selected board is not equal to the defined grade for the part, whether the grade of the selected board is greater than the defined grade for the part;

determine, in response to a determination that the grade of the selected board is greater than the defined grade for the part, whether the defined grade for the part is less than a grade of another board in the set of boards analyzed for inclusion in the set of boards for potential use and that has a grade that is greater than the defined grade for the part;

determine, in response to a determination that the grade of the selected board is less than the grade of the other board, whether the length of the selected board is less than a length of the other board; and

add, in response to a determination that the length of the selected board is less than the length of the other board, the selected board to the set of boards for potential use.

AH14. The computing device of claim AH4, wherein to determine a set of potential parts for each board in the set of boards comprises to:

select a board from the set of boards for potential use; and

determine, for each part in the batch, whether to add the part to the set of potential parts for the selected board, as a function of a comparison of a grade of the selected board to a defined grade for the part, a comparison of the width of the selected board to a width for the part, and a determination of whether the part will fit on the selected board.

AH15. The computing device of claim AH4, wherein to determine the best board comprises to:

select a board from the set of boards for potential use in the batch;

determine, as a function of a number of parts to be produced from the selected board, a set of permutations of possible orderings for the parts;

define a set of groups of parts from the set of permutations of possible orderings for the parts; and

determine the best board as a function of the defined set of groups of parts.

AH16. The computing device of claim AH15, wherein the circuitry is further configured to:

determine, for a part in a group in the defined set of groups of parts, whether the part is the first part in the possible ordering of parts associated with the group; and

selectively, rotate the board in response to a determination that the part is the first part.

AH17. The computing device of claim AH16, wherein to selectively rotate the board comprises to rotate the board to orient a straight edge of the part with a straight edge of the board.

AH18. The computing device of claim AH16, wherein the circuity is further configured to determine, in response to a determination that the part is not the first part, whether the selected part can be rotated.

AH19. The computing device of claim AH18, wherein to determine whether the part can be rotated comprises to determine whether rotation of the part would satisfy one or more size parameters.

AH20. The computing device of claim AH19, wherein to determine whether rotation of the part would satisfy one or more size parameters comprises to determine whether rotation of the part would satisfy a size parameter associated with a clamp of a saw of the automated system and whether rotation of the part would satisfy a length parameter associated with a robot of the automated system.

AH21. The computing device of claim AH20, wherein the circuitry is further configured to identify a rotation option that would cause the part to use less of the board than a set of other rotation options.

AH22. The computing device of claim AH21, wherein the circuitry is further configured to:

determine whether an angle of the part can nest within an angle of a previous part to be produced from the board; and

determine whether the part has a better fit between being ordered after the previous part or being located at a beginning of the board.

AH23. The computing device of claim AH22, wherein to determine whether an angle of the part can nest within the angle of the previous part comprises to determine whether both parts can cross over a centerline between the parts if angled cuts are utilized.

AH24. The computing device of claim AH22, wherein to determine whether the part has a better fit comprises to:

determine whether a last piece of the board will be large enough to enable one or more clamps of the automated system to manipulate the board; and

determine whether the part will fit at the beginning of the board if the part is not large enough to be produced from the end of the board.

AH25. The computing device of claim AH15, wherein the circuitry is further configured to:

output data indicative of a part location configuration for a possible ordering of parts; and

determine, as a function of an amount of the board utilized by the parts, whether a selected possible ordering of the parts is better than a previous best possible ordering of parts in a selected group of possible orderings.

AH26. The computing device of claim AH1, wherein the circuitry is further configured to:

determine whether a board from which a set of parts of a wooden truss is to be produced has a remaining length that satisfies a predefined length;

determine, in response to a determination that the remaining length satisfies the predefined length, whether one or more standard parts having a lengths defined in a set of standard length configuration data in the lumber selection input data can be produced from the remaining length of the board; and

add, in response to a determination that one or more standard parts can be produced from the remaining length of the board, the one or more standard parts to a set of standard parts to be produced from the board.

AI1. A method comprising:

obtaining, by a computing device, lumber selection input data indicative of one or more parameters to be utilized in the selection of lumber for the production of one or more wooden trusses by an automated system; and

selecting, by the computing device and as a function of the one or more parameters and characteristics of lumber available in a lumber inventory of the automated system, a set of lumber pieces from the lumber inventory to satisfy an efficiency target in the automated production of the one or more wooden trusses.

AI2. The method of claim AI1, wherein obtaining lumber selection input data comprises obtaining one or more of assembly recipe data indicative of operations to be performed by the automated system to produce the one or more wooden trusses, production data indicative of quantities of the wooden trusses to be produced in one or more batches, line data indicative of a status of one or more in-feed lines of the automated system, inventory data indicative of the characteristics of lumber available in the lumber inventory, saw parameter data indicative of a parameters of a saw of the automated system to be used to cut the lumber, or standard length configuration data indicative of one or more lengths defined for pieces of lumber to be used by the automated system.

AI3. The method of claim AI1, further comprising determining, by the computing device and as a function of the lumber selection input data, a set of parts to be used in the production of wooden trusses in a batch.

AI4. The method of claim AI3, further comprising:

determining, by the computing device, a set of boards for potential use in a batch of one or more wooden trusses to be produced;

determining, by the computing device, a set of potential parts for each board in the set of boards, including identifying, as a function of the parameters, a best board from which to produce the potential parts from lumber in the lumber inventory having a grade equal to a grade specified in connection with a design for a wooden truss to be produced and from lumber in the lumber inventory having a grade that is greater than the grade specified in connection with the design; and

removing, by the computing device, from a list of remaining parts to be produced, one or more of the potential parts to be produced from the determined best board.

AI5. The method of claim AI4, further comprising determining, by the computing device, the best board to be used for each of the remaining parts to be produced.

AI6. The method of claim AI3, wherein determining the set of parts to be used in the production of wooden trusses in the batch comprises:

determining, by the computing device, whether the batch is queued for a first line or a second line of the automated system;

determining, by the computing device and in response to a determination that the batch is queued for the first line or the second line, whether a wooden truss remains to be produced from an assembly recipe defined in the lumber selection input data;

determining, by the computing device and in response to a determination that a wooden truss remains to be produced from the assembly recipe, whether a part remains to be produced for the wooden truss; and

calculating, by the computing device and in response to a determination that a part remains to be produced for the wooden truss, a pickup location indicative of a length along the part where a robot of the automated system will pick up the part.

AI7. The method of claim AI6, further comprising:

determining, by the computing device, whether the pickup location satisfies a predefined length threshold; and

rotating, by the computing device and in response to a determination that the pickup location does not satisfy the predefined length threshold, the part by 180 degrees.

AI8. The method of claim AI6, further comprising:

determining, by the computing device and in response to a determination that the pickup location does not satisfy the predefined length threshold, whether the pickup location satisfies a percentage of part length threshold; and

rotating, by the computing device and in response to a determination that the pickup location satisfies the percentage of part length threshold, the part by 180 degrees.

AI9. The method of claim AI8, wherein determining whether the pickup location satisfies the percentage of part length threshold comprises determining whether the pickup location is greater than 70% of the part length.

AI10. The method of claim AI8, further comprising determining, by the computing device and in response to a determination that the pickup location does not satisfy the percentage of part length threshold, not to rotate the part.

AI11. The method of claim AI4, wherein determining a set of boards for potential use in the batch comprises:

selecting, by the computing device, a board from the lumber inventory;

determining, by the computing device, whether a width of the selected board satisfies a width threshold for a part in the batch to be produced from the selected board;

determining, by the computing device and in response to a determination that the width of the selected board satisfies the width threshold, whether a length of the selected board satisfies a length threshold for the part; and

determining, by the computing device and in response to a determination that the length satisfies the length threshold, a set of one or more responsive operations as a function of a comparison between a grade of the selected board and a defined grade for the part.

AI12. The method of claim AI11, further comprising excluding, by the computing device, the board from the set of boards for potential use in response to a determination that the width of the selected board does not satisfy the width threshold or that the length of the selected board does not satisfy the length threshold.

AI13. The method of claim AI11, wherein determining a set of one or more responsive operations as a function of a comparison between the grade of the selected board and the defined grade for the part comprises:

determining whether the grade of the selected board is equal to the defined grade for the part;

determining, in response to a determination that the grade of the selected board is not equal to the defined grade for the part, whether the grade of the selected board is greater than the defined grade for the part;

determining, in response to a determination that the grade of the selected board is greater than the defined grade for the part, whether the defined grade for the part is less than a grade of another board in the set of boards analyzed for inclusion in the set of boards for potential use and that has a grade that is greater than the defined grade for the part;

determining, in response to a determination that the grade of the selected board is less than the grade of the other board, whether the length of the selected board is less than a length of the other board; and

adding, in response to a determination that the length of the selected board is less than the length of the other board, the selected board to the set of boards for potential use.

AI14. The method of claim AI4, wherein determining a set of potential parts for each board in the set of boards comprises:

selecting a board from the set of boards for potential use; and

determining, for each part in the batch, whether to add the part to the set of potential parts for the selected board, as a function of a comparison of a grade of the selected board to a defined grade for the part, a comparison of the width of the selected board to a width for the part, and a determination of whether the part will fit on the selected board.

AI15. The method of claim AI4, wherein determining the best board comprises:

selecting a board from the set of boards for potential use in the batch;

determining, as a function of a number of parts to be produced from the selected board, a set of permutations of possible orderings for the parts;

defining a set of groups of parts from the set of permutations of possible orderings for the parts; and

determining the best board as a function of the defined set of groups of parts.

AI16. The method of claim AI15, further comprising:

determining, by the computing device and for a part in a group in the defined set of groups of parts, whether the part is the first part in the possible ordering of parts associated with the group; and

selectively rotating, by the computing device, the board in response to a determination that the part is the first part.

AI17. The method of claim AI16, wherein selectively rotating the board comprises rotating the board to orient a straight edge of the part with a straight edge of the board.

AI18. The method of claim AI16, further comprising determining, by the computing device and in response to a determination that the part is not the first part, whether the selected part can be rotated.

AI19. The method of claim AI18, wherein determining whether the part can be rotated comprises determining whether rotation of the part would satisfy one or more size parameters.

AI20. The method of claim AI19, wherein determining whether rotation of the part would satisfy one or more size parameters comprises determining whether rotation of the part would satisfy a size parameter associated with a clamp of a saw of the automated system and whether rotation of the part would satisfy a length parameter associated with a robot of the automated system.

AI21. The method of claim AI20, further comprising identifying, by the computing device, a rotation option that would cause the part to use less of the board than a set of other rotation options.

AI22. The method of claim AI21, further comprising:

determining, by the computing device, whether an angle of the part can nest within an angle of a previous part to be produced from the board; and

determining, by the computing device, whether the part has a better fit between being ordered after the previous part or being located at a beginning of the board.

AI23. The method of claim AI22, wherein determining whether an angle of the part can nest within the angle of the previous part comprises determining whether both parts can cross over a centerline between the parts if angled cuts are utilized.

AI24. The method of claim AI22, wherein determining whether the part has a better fit comprises:

determining whether a last piece of the board will be large enough to enable one or more clamps of the automated system to manipulate the board; and

determining whether the part will fit at the beginning of the board if the part is not large enough to be produced from the end of the board.

AI25. The method of claim AI15, further comprising:

outputting, by the computing device, data indicative of a part location configuration for a possible ordering of parts; and

determining, by the computing device and as a function of an amount of the board utilized by the parts, whether a selected possible ordering of the parts is better than a previous best possible ordering of parts in a selected group of possible orderings.

AI26. The method of claim AI1, further comprising:

determining, by the computing device, whether a board from which a set of parts of a wooden truss is to be produced has a remaining length that satisfies a predefined length;

determining, by the computing device and in response to a determination that the remaining length satisfies the predefined length, whether one or more standard parts having a lengths defined in a set of standard length configuration data in the lumber selection input data can be produced from the remaining length of the board; and

adding, by the computing device and in response to a determination that one or more standard parts can be produced from the remaining length of the board, the one or more standard parts to a set of standard parts to be produced from the board.

AJ1. One or more machine-readable storage media comprising a plurality of instructions stored thereon that, in response to being executed, cause a computing device to perform the method of any of claims AI1-AI26.

AK1. An automated system comprising:

a fiducial printer;

an in-feed line to receive a board;

a carrier within the in-feed line to selectively move the board; and

circuitry configured to:

move, with the carrier, a board received in the in-feed line to a defined fiducial printing position; and

print, with the fiducial printer and in response to a determination that the board has been moved to the defined fiducial printing position, a fiducial on a side of the board, wherein the fiducial represents information to be detected by a machine vision system to perform one or more operations associated with production, by the automated system, of a wooden structure using the board.

AK2. The automated system of claim AK1, wherein the circuitry is further configured to detect, utilizing one or more sensors, that the board has entered the in-feed line, and wherein to move the board comprises to move, with the carrier, the board in response to the determination that the board has entered the in-feed line.

AK3. The automated system of claim AK1, wherein to move, with the carrier, the board in the in-feed line comprises to move the board along an in-feed axis to the defined fiducial printing position.

AK4. The automated system of claim AK3, wherein the circuitry is further configured to detect, with a corresponding sensor, an end of the board in the in-feed line.

AK5. The automated system of claim AK4, wherein to detect the end of the board comprises to detect the end of the board with a photoelectric sensor.

AK6. The automated system of claim AK5, wherein the circuitry is further configured to advance, in response to detection of the end of the board, the board along an in-feed axis a predefined length to move the board to the defined fiducial printing position.

AK7. The automated system of claim AK1, wherein to print a fiducial on a side of the board comprises to print the fiducial on a major side of the board or a minor side of the board.

AK8. The automated system of claim AK7, wherein to print a fiducial comprises to print one or more predefined symbols.

AK9. The automated system of claim AK8, wherein to print one or more predefined symbols comprises to print predefined symbols that are mirrored in opposite directions.

AK10. The automated system of claim AK8, wherein to print one or more predefined symbols comprises to print one or more predefined symbols that are indicative of information about the board.

AK11. The automated system of claim AK10, wherein to print one or more predefined symbols that are indicative of information about the board comprises to print one or more predefined symbols indicative of an identifier of the board.

AK12. The automated system of claim AK10, wherein to print one or more symbols that are indicative of information about the board comprises to print one or more predefined symbols indicative of an index value in a sequence of boards to be used in the production of the wooden structure by the automated system.

AK13. The automated system of claim AK10, wherein to print one or more symbols that are indicative of information about the board comprises to print one or more predefined symbols indicative of an identifier of a leading end or trailing end of the board.

AK14. The automated system of claim AK10, wherein to print one or more symbols that are indicative of information about the board comprises to print one or more symbols indicative of a grade of the board.

AK15. The automated system of claim AK1, wherein to print the fiducial comprises to print the fiducial with multiple print heads arranged along a width of the board.

AK16. The automated system of claim AK1, wherein the defined fiducial printing position is a first defined fiducial printing position that is associated with a first end of the board and the circuity is further configured to:

move the board to a second defined fiducial printing position that is associated with a second end of the board; and

print, in response to a determination that the board is in the second defined fiducial printing position, one or more second fiducials on the board.

AK17. The automated system of claim AK16, wherein to move the board to the second defined printing position comprises to advance the board along an in-feed axis until the automated system detects the second end of the board with a sensor that detected the presence of the first end of the board.

AK18. The automated system of claim AK17, wherein to advance the board along the in-feed axis until the automated system detects the second end of the board comprises to advance the board along the in-feed axis until the sensor detects that the board is no longer present.

AK19. The automated system of claim AK17, wherein the circuitry is further configured to reverse, in response to detection of the second end of the board, a direction of movement of the board along the in-feed axis.

AK20. The automated system of claim AK19, wherein to reverse the direction of movement comprises to advance the board in the opposite direction by a predefined length.

AK21. The automated system of claim AK1, wherein the circuitry is further configured to utilize the machine vision system to identify the fiducial on the board in the production of the wooden structure.

AK22. The automated system of claim AK21, wherein to utilize the machine vision system comprises to acquire an image of the fiducial with a camera of a component of the automated system.

AK23. The automated system of claim AK22, wherein the circuitry is further configured to determine a position of the board relative to the component based on the acquired image of the fiducial.

AK24. The automated system of claim AK22, wherein the camera is associated with a tool of a robot of the automated system and the circuitry is further configured to position the camera over the fiducial and determine the location of the tool relative to the board based on a predefined position of the fiducial on the board and an offset of the location of the camera relative to a center of the tool.

AK25. The automated system of claim AK21, wherein to utilize the machine vision system comprises to determine information indicated by the fiducial.

AK26. The automated system of claim AK25, wherein to determine information indicated by the fiducial comprises to determine one more of an identifier of the board, an index value in a sequence associated with the board, a grade of the board, or which of multiple ends of the board is imaged.

AL1. A method comprising:

moving, with a carrier of an automated system, a board received in an in-feed line of the automated system to a defined fiducial printing position; and

printing, with a fiducial printer of the automated system and in response to a determination that the board has been moved to the defined fiducial printing position, a fiducial on a side of the board, wherein the fiducial represents information to be detected by a machine vision system to perform one or more operations associated with production, by the automated system, of a wooden structure using the board.

AL2. The method of claim AL1, further comprising detecting, utilizing one or more sensors, that the board has entered the in-feed line, and wherein moving the board comprises moving, with the carrier, the board in response to the determination that the board has entered the in-feed line.

AL3. The method of claim AL1, wherein moving, with the carrier, the board in the in-feed line comprises moving the board along an in-feed axis to the defined fiducial printing position.

AL4. The method of claim AL3, further comprising detecting, with a corresponding sensor, an end of the board in the in-feed line.

AL5. The method of claim AL4, wherein detecting the end of the board comprises detecting the end of the board with a photoelectric sensor.

AL6. The method of claim AL5, further comprising advancing, in response to detection of the end of the board, the board along an in-feed axis a predefined length to move the board to the defined fiducial printing position.

AL7. The method of claim AL1, wherein printing a fiducial on a side of the board comprises printing the fiducial on a major side of the board or a minor side of the board.

AL8. The method of claim AL7, wherein printing a fiducial comprises printing one or more predefined symbols.

AL9. The method of claim AL8, wherein printing one or more predefined symbols comprises printing predefined symbols that are mirrored in opposite directions.

AL10. The method of claim AL8, wherein printing one or more predefined symbols comprises printing one or more predefined symbols that are indicative of information about the board.

AL11. The method of claim AL10, wherein printing one or more predefined symbols that are indicative of information about the board comprises printing one or more predefined symbols indicative of an identifier of the board.

AL12. The method of claim AL10, wherein printing one or more symbols that are indicative of information about the board comprises printing one or more predefined symbols

indicative of an index value in a sequence of boards to be used in the production of the wooden structure by the automated system.

AL13. The method of claim AL10, wherein printing one or more symbols that are indicative of information about the board comprises printing one or more predefined symbols indicative of an identifier of a leading end or trailing end of the board.

AL14. The method of claim AL10, wherein printing one or more symbols that are indicative of information about the board comprises printing one or more symbols indicative of a grade of the board.

AL15. The method of claim AL1, wherein printing the fiducial comprises printing the fiducial with multiple print heads arranged along a width of the board.

AL16. The method of claim AL1, wherein the defined fiducial printing position is a first defined fiducial printing position that is associated with a first end of the board, the method further comprising:

moving the board to a second defined fiducial printing position that is associated with a second end of the board; and

printing, in response to a determination that the board is in the second defined fiducial printing position, one or more second fiducials on the board.

AL17. The method of claim AL16, wherein moving the board to the second defined printing position comprises advancing the board along an in-feed axis until the automated system detects the second end of the board with a sensor that detected the presence of the first end of the board.

AL18. The method of claim AL17, wherein advancing the board along the in-feed axis until the automated system detects the second end of the board comprises advancing the board along the in-feed axis until the sensor detects that the board is no longer present.

AL19. The method of claim AL17, further comprising reversing, by the carrier and in response to detection of the second end of the board, a direction of movement of the board along the in-feed axis.

AL20. The method of claim AL19, wherein reversing the direction of movement comprises advancing the board in the opposite direction by a predefined length.

AL21. The method of claim AL1, further comprising utilizing the machine vision system to identify the fiducial on the board in the production of the wooden structure.

AL22. The method of claim AL21, wherein utilizing the machine vision system comprises acquiring an image of the fiducial with a camera of a component of the automated system.

AL23. The method of claim AL22, further comprising determining, by the automated system, a position of the board relative to the component based on the acquired image of the fiducial.

AL24. The method of claim AL22, wherein the camera is associated with a tool of a robot of the automated system, the method further comprising positioning, by the automated system, the camera over the fiducial and determining the location of the tool relative to the board based on a predefined position of the fiducial on the board and an offset of the location of the camera relative to a center of the tool.

AL25. The method of claim AL21, wherein utilizing the machine vision system comprises determining information indicated by the fiducial.

AL26. The method of claim AL25, wherein determining information indicated by the fiducial comprises determining one more of an identifier of the board, an index value in a sequence associated with the board, a grade of the board, or which of multiple ends of the board is imaged.

AM1. One or more machine-readable storage media comprising a plurality of instructions stored thereon that, in response to being executed, cause a computing device to perform the method of any of claims AL1-AL26.

AN1. A computing device comprising:

circuitry configured to:

obtain data indicative of a set of one or more jobs for the production of one or more wooden structures by an automated system; and

utilize a microservices architecture to produce the one or more wooden structures associated with the one or more jobs.

AN2. The computing device of claim AN1, wherein to utilize a microservices architecture further comprises to interrupt execution of one microservice without interrupting execution of other microservices in the microservices architecture.

AN3. The computing device of claim AN1, wherein to utilize the microservices architecture comprises to communicate data between microservices using a network communication protocol.

AN4. The computing device of claim AN3, wherein to communicate data between microservices using a network communication protocol comprises to communicate using hypertext transfer protocol or hypertext transfer protocol secure.

AN5. The computing device of claim AN1, wherein to utilize a microservices architecture comprises to utilize an assembly recipe generator microservice to produce a recipe indicative of a sequence of operations of components of the automated system to produce the one or more wooden structures.

AN6. The computing device of claim AN5, wherein to utilize a microservices architecture further comprises to utilize a lumber optimizer microservice to select lumber from a lumber inventory to satisfy one or more target parameters in the automated production of the one or more wooden structures by the automated system.

AN7. The computing device of claim AN6, wherein to utilize a microservices architecture further comprises to utilize a set of microservices to control machines of the automated system to produce the one or more wooden structures.

AN8. The computing device of claim AN7, wherein to utilize a set of microservices to control machines of the automated system comprises to utilize a set of microservices to read and write machine register values using network communication protocols.

AN9. The computing device of claim AN7, wherein to utilize a set of microservices to control machines of the automated system comprises to utilize an assembler microservice to control one or more assembler machines.

AN10. The computing device of claim AN9, wherein to utilize an assembler microservice to control one or more assembler machines comprises to utilize an assembler microservice to communicate with one or more assembler machine controllers.

AN11. The computing device of claim AN10, wherein to utilize an assembler microservice to communicate with one or more assembler machine controllers comprises to utilize an assembler microservice to communicate with one or more assembly robot controllers.

AN12. The computing device of claim AN7, wherein to utilize a set of microservices to control machines of the automated system comprises to utilize a plate microservice to control a plate machine for manipulating nailing plates and a saw microservice to control a saw machine of the automated system by reading and writing machine register values of a controller of the plate machine and a controller of the saw machine using a network communication protocol.

AN13. The computing device of claim AN1, wherein to utilize a microservices architecture further comprises to utilize a shell microservice to provide a shell for one or more user interfaces.

AN14. The computing device of claim AN1, wherein to utilize a microservices architecture further comprises to utilize a log aggregator microservice to manage logs produced by the automated system.

AN15. The computing device of claim AN1, wherein to utilize a microservices architecture further comprises to utilize a machine diagnostics microservice to analyze an operational status of one or more machines of the automated system.

AN16. The computing device of claim AN1, wherein to utilize a microservices architecture comprises to utilize microservices that are based on one or more shared packages of executable instructions or data.

AO1. A method comprising:

obtaining, by a computing device, data indicative of a set of one or more jobs for the production of one or more wooden structures by an automated system; and

utilizing, by the computing device, a microservices architecture to produce the one or more wooden structures associated with the one or more jobs.

AO2. The method of claim AO1, wherein utilizing a microservices architecture further comprises interrupting execution of one microservice without interrupting execution of other microservices in the microservices architecture.

AO3. The method of claim AO1, wherein utilizing the microservices architecture comprises communicating data between microservices using a network communication protocol.

AO4. The method of claim AO3, wherein communicating data between microservices using a network communication protocol comprises communicating using hypertext transfer protocol or hypertext transfer protocol secure.

AO5. The method of claim AO1, wherein utilizing a microservices architecture comprises utilizing an assembly recipe generator microservice to produce a recipe indicative of a sequence of operations of components of the automated system to produce the one or more wooden structures.

AO6. The method of claim AO5, wherein utilizing a microservices architecture further comprises utilizing a lumber optimizer microservice to select lumber from a lumber inventory to

satisfy one or more target parameters in the automated production of the one or more wooden structures by the automated system.

AO7. The method of claim AO6, wherein utilizing a microservices architecture further comprises utilizing a set of microservices to control machines of the automated system to produce the one or more wooden structures.

AO8. The method of claim AO7, wherein utilizing a set of microservices to control machines of the automated system comprises utilizing a set of microservices to read and write machine register values using network communication protocols.

AO9. The method of claim AO7, wherein utilizing a set of microservices to control machines of the automated system comprises utilizing an assembler microservice to control one or more assembler machines.

AO10. The method of claim AO9, wherein utilizing an assembler microservice to control one or more assembler machines comprises utilizing an assembler microservice to communicate with one or more assembler machine controllers.

AO11. The method of claim AO10, wherein utilizing an assembler microservice to communicate with one or more assembler machine controllers comprises utilizing an assembler microservice to communicate with one or more assembly robot controllers.

AO12. The method of claim AO7, wherein utilizing a set of microservices to control machines of the automated system comprises utilizing a plate microservice to control a plate machine for manipulating nailing plates and a saw microservice to control a saw machine of the automated system by reading and writing machine register values of a controller of the plate machine and a controller of the saw machine using a network communication protocol.

AO13. The method of claim AO1, wherein utilizing a microservices architecture further comprises utilizing a shell microservice to provide a shell for one or more user interfaces.

AO14. The method of claim AO1, wherein utilizing a microservices architecture further comprises utilizing a log aggregator microservice to manage logs produced by the automated system.

AO15. The method of claim AO1, wherein utilizing a microservices architecture further comprises utilizing a machine diagnostics microservice to analyze an operational status of one or more machines of the automated system.

AO16. The method of claim AO1, wherein utilizing a microservices architecture comprises utilizing microservices that are based on one or more shared packages of executable instructions or data.

AP1. One or more machine-readable storage media comprising a plurality of instructions stored thereon that, in response to being executed, cause a computing device to perform the method of any of claims AO1-AO16.

**WHAT IS CLAIMED IS:**

1. A method of automated handling of lumber for assembling a wooden structure, the method comprising:

delivering stock lumber to an in-feed station;

staging the stock lumber on a first buffer table at the in-feed station;

delivering the staged stock lumber to an in-feed line;

transporting the stock lumber to a cutting station along the in-feed line;

cutting the stock lumber into multiple lumber pieces at the cutting station;

delivering the lumber pieces to a second buffer table at a buffer station;

arranging the lumber pieces on the second buffer table in an order in which the lumber pieces will be used in the assembly of the wooden structure; and

transporting the lumber pieces from the buffer station to an assembly station for assembly the wooden structure.

2. The method of claim 1, further comprising determining the order in which the lumber pieces will be used in the assembly of the wooden structure.

3. The method of claim 2, wherein determining the order in which lumber pieces will be used occurs prior to transporting the stock lumber to the cutting station.

4. The method of claim 2, wherein determining the order in which the lumber pieces will be used is based on determining the order in which joints in the assembled wooden structure are to be formed.

5. The method of claim 1, wherein delivering the stock lumber to the in-feed station comprises grasping the stock lumber along a major side surface of the stock lumber with a mechanical manipulator such that the stock lumber is transported with its minor surfaces facing upward and downward, and its major surfaces facing horizontally.

6. The method of claim 5, wherein delivering the lumber pieces to the second buffer table comprises grasping a first set of the lumber pieces with a second mechanical manipulator

disposed above the second buffer table.

7. The method of claim 6, further comprising delivering a second set of the lumber pieces to a third buffer table with a third mechanical manipulator.

8. The method of claim 1 further comprising lifting stock lumber delivered to the feed station to a position for being transported to the first buffer table.

9. The method of claim 1, wherein delivering the staged stock lumber to the in-feed line further comprises first delivering the stock lumber to a holding position, then retrieving said stock lumber from the holding position and dropping the stock lumber into the in-feed line.

10. The method of claim 1, further comprising printing fiducials on the stock lumber while the stock lumber is being transported on saw in-feed line.

11. The method of claim 1, further comprising assembling the wooden structure at the assembly station using no more than two robotic arms to place the lumber pieces in position on an assembly table at the assembly station.

12. The method of claim 11, further comprising transporting the lumber pieces along the assembly table using at least one of the robot arms.

13. An automated wooden structure manufacturing system for manufacturing and assembling wooden structures, the system comprising:

an in-feed station comprising a first buffer table configured for receiving and staging stock lumber;

a cutting station comprising a saw assembly configured for cutting the stock lumber into multiple lumber pieces;

a buffer station comprising a second buffer table configured for holding the lumber pieces and arranging the lumber pieces in an order in which the lumber pieces will be used in the assembly of a wooden structure; and

an assembly station comprising an assembly module configured for attaching the lumber pieces together to assemble the wooden structure.

14. The system of claim 13, wherein the in-feed station comprises an in-feed line connecting the first buffer table to the cutting station.

15. The system of claim 14, wherein the in-feed station comprises a mechanical manipulator configured to place the stock lumber on the first buffer table, and pick the stock lumber from the first buffer table and deliver the stock lumber to the in-feed line.

16. The system of claim 15 wherein the in-feed station includes a lift for raising the stock lumber to a position for being grasped by the mechanical manipulator.

17. The system of claim 14, further comprising a printer along the in-feed line for printing fiducials on the stock lumber.

18. The system of claim 14, wherein the in-feed line comprises a first in-feed line, the in-feed station comprising a second in-feed line.

19. The system of claim 13, wherein the buffer station comprises a mechanical manipulator configured to place the lumber pieces on the second buffer table.

20. The system of claim 19, wherein the mechanical manipulator comprises a first mechanical manipulator configured to place a first set of the lumber pieces on the second buffer table, the buffer station further comprising a second mechanical manipulator configured to place a second set of the lumber pieces on a third buffer table.

21. The system of claim 13, wherein the second buffer table includes a plurality of first and second slots configured for receiving the pieces of lumber, the first slots having a length that is greater than a length of the second slots such that the first slots are configured to receive longer pieces of lumber than the second slots.

22. The system of claim 13, wherein the assembly module comprises a first robot configured to position a first lumber piece on an assembly table at the assembly station, and a second robot configured to hold a second lumber piece on the assembly table.

23. The system of claim 22, wherein the first and second robots each include articulated robot arms.

24. The system of claim 23, wherein the assembly station includes no other robots for positioning or holding lumber pieces on the assembly table.

25. An automated wooden structure manufacturing system for manufacturing and assembling wooden structures, the system comprising:

a controller operable to demand stock lumber for constructing at least one of the wooden structures;

a monitor for displaying what pieces of the stock lumber are demanded;

an in-feed station comprising a stock lumber receiver, a first manipulator for picking up the stock lumber from the stock lumber receiver; and a first buffer table, the controller being configured to cause the manipulator to place the stock lumber on the first buffer table;

an in-feed line arranged to receive stock lumber from the first buffer table;

a cutting station comprising a saw assembly configured to receive stock lumber delivered by the in-feed line and to cut the stock lumber into multiple lumber pieces;

a discharge station positioned to receive the multiple pieces of lumber from the cutting station;

a buffer station comprising a second manipulator arranged to pick up the multiple pieces of lumber from the discharge station, a second buffer table configured for holding the lumber pieces transferred to the second buffer table by the second manipulator;

an assembly conveyor arranged to receive pieces of the stock lumber from the second buffer table; and

an assembly station arranged to receive the pieces of the stock lumber from the assembly conveyor, the assembly station comprising an assembly module configured for attaching the lumber pieces together to assemble said at least one of the wooden structures.

26. A method of assembling a wooden structure including a plurality of lumber pieces connected together at a plurality of joints, the method comprising:

arranging lumber pieces relative to each other at a first joint location of the wooden structure;

attaching the lumber pieces positioned at the first joint location to each other using a pair of nailing plates secured to top and bottom surfaces of the lumber pieces to form a first joint of the wooden structure;

arranging one or more lumber pieces relative other lumber pieces at a second joint location of the wooden structure after forming the first joint;

attaching each lumber piece at the second joint location using a pair of nailing plates secured to top and bottom surfaces of the lumber pieces at the second joint location to form a second joint of the wooden structure;

arranging one or more lumber pieces relative to other lumber pieces at a third joint location of the wooden structure after forming the second joint;

attaching each lumber piece at the third joint location using a pair of nailing plates secured to top and bottom surfaces of the lumber pieces at the third joint location to form a third joint of the wooden structure; and

continuing to assemble the wooden structure in a joint-by-joint sequence until the entire wooden structure is assembled such that each joint of the wooden structure is completely formed prior to positioning and attaching two or more lumber pieces at another joint, wherein a bottom nailing plate is attached to each lumber piece after arranging the lumber piece at a joint prior to subsequently placing another lumber piece at a joint.

27. The method of claim 26, wherein the wooden structure is assembled using an automated manufacturing and assembly system.

28. The method of claims 26 or 27, wherein positioning one or more pieces at the second joint location comprises picking up at least one of the plurality of lumber pieces from a location remote from the second joint location and from the first joint and carrying said at least one of the plurality of lumber pieces to the second joint location.

29. The method of any one of claims 26-28 wherein the joints of the wooden structure are assembled at a single joint forming station.

30. The method of any one of claims 26-29 wherein positioning one or more lumber pieces at the second joint location comprises moving a partially assembled wooden structure including at least the first joint over an assembly table in a first direction to position at least one of the lumber pieces at the second joint location.

31. The method of claim 30, wherein positioning one or more lumber pieces at the third location comprises moving a partially assembled wooden structure including at least the first and second joints over an assembly table in a second direction to position at least one of the lumber pieces at the third joint location, the second direction including at least a component of movement that is opposite to the first direction.

32. The method of any one of claims 26-31, wherein forming the first joint and forming the second joint are carried out one immediately after another.

33. The method of any one of claims 26-32, wherein each of said attaching steps comprises fully attaching the bottom nailing plate at the joint prior to attaching the top nailing plate.

34. The method of claim 33, further comprising using an upper platen to attach the bottom nailing plate at the joint.

35. The method of claim 26, wherein an order of the joints assembled in the joint-by-joint sequence is determined based on at least one of a location of the joint, a number of lumber pieces at the joint, and a size of the lumber pieces at the joint.

36. The method of claim 35, wherein the first joint location includes the joint with the longest lumber piece.

37. The method of claim 35, wherein the second joint location includes the joint having the most number of lumber pieces already in place after forming the first joint.

38. The method of claim 35, wherein a point system is assigned to various assembly actions and conditions, and a sequence having a highest point total is selected to for the joint-by-joint sequence.

39. The method of any one of claims 26-38, wherein the wooden structure is a truss.

40. The method of any one of claims 26-39 wherein the step of arranging the lumber pieces at the first joint location includes placing the lumber pieces so that they overlie the bottom nailing plate.

41. An automated assembly method for assembling a wooden structure including a plurality of lumber pieces connected together at a plurality of joints, the method comprising assembling the wooden structure in a joint-by-joint extrusion sequence until the entire wooden structure is assembled, whereby each joint of the wooden structure is formed by arranging lumber pieces relative to each other at a joint forming station and securing the lumber pieces together using top and bottom nailing plates to form the joints at the joint forming station, the bottom nailing plate being attached to each lumber piece immediately after arranging the lumber piece at a joint prior to subsequently placing another lumber piece at a joint.

42. The method of claim 41, further comprising assembling the wooden structure using robotic arms to place the lumber pieces in position on an assembly table.

43. The method of claim 42, wherein no more than two robotic arms at the joint forming station are used to assemble the wooden structure.

44. The method of claim 42, further comprising attaching the lumber pieces together using upper and lower attachment devices to drive nailing plates into upper and lower surfaces of the lumber pieces.

45. The method of claims 42 or 43, wherein the robotic arms do not perform any attachment functions for attaching the lumber pieces together.

46. The method of claims 42 or 43, further comprising transporting the lumber pieces along the assembly table using at least one of the robot arms.

47. The method of claim 41, wherein the nailing plates are not attached to the lumber pieces at a location other than the joint forming station.

48. The method of claim 41, wherein the wooden structure is a truss.

49. The method of claim 42, further comprising:

moving a robot arm end effector of a robot to a lumber piece;

acquiring an image of indicia on the lumber piece using a camera on the robot using a camera;

determining a placement location for the lumber piece on the assembly table based on the acquired image;

securing the lumber piece to the end effector; and

moving the lumber piece to the determined placement location using the robot such that an end of the lumber piece is disposed at the joint.

50. The method of claim 49, wherein the lumber piece comprises a first lumber piece and the robot comprises a first robot, the method further comprising releasing the first lumber piece from the first robot and securing the first lumber piece at the determined placement location with a second robot.

51. The method of claim 50, further comprising:

acquiring an image of the indicia on the lumber piece using a camera on the second robot; and

moving the first lumber piece to the determined location using the second robot.

52. The method of claim 51, further comprising positioning a second lumber piece at the joint with the first robot while the second robot holds the first lumber piece at the joint.

53. The method of claim 41, wherein an order of the joints assembled in the joint-by-joint sequence is determined based on at least one of a location of the joint, a number of lumber pieces at the joint, and a size of the lumber pieces at the joint, and wherein a point system is assigned to various assembly actions and conditions, and a sequence having a highest point total is selected to for the joint-by-joint sequence.

54. An automated assembly method for assembling a wooden structure including a plurality of lumber pieces connected together at a plurality of joints, the method comprising:

using a first robot to position a first lumber piece on an assembly table at a joint location of the wooden structure;

using a second robot to position a second lumber piece on the assembly table at the joint location of the wooden structure;

attaching the first lumber piece to the second lumber piece on the assembly table using an attachment device separate from the first and second robots; and

transporting the wooden structure along the assembly table using one of the first and second robots.

55. The method of claim 54, wherein the first and second robots hold the first and second lumber pieces as the first lumber piece is attached to the second lumber piece.

56. The method of claims 54 or 55, wherein the transporting occurs after the first and second lumber pieces are attached.

57. The method of claim 54, wherein the first robot retrieves the lumber pieces from a delivery location and the second robot transports the wooden structure along the assembly table.

58. The method of claim 54, wherein the robots do not perform any attachment functions for attaching the lumber pieces together.

59. The method of claim 54 wherein transporting the wooden structure along the assembly table comprises moving the wood structure in opposite directions.

60. The method of claim 54, further comprising:

grabbing the first lumber piece with a tool of the first robot;

viewing a fiducial on the first lumber piece with a camera of the tool; and

determining a position of the tool on the first lumber piece based on a relative position between the tool and the fiducial.

61. An automated assembly method for assembling a wooden structure including a plurality of lumber pieces connected together at a plurality of joints, the method comprising:

using a first robot to position a first lumber piece on an assembly table at a joint location of the wooden structure;

using a second robot to position a second lumber piece on the assembly table at the joint location of the wooden structure;

using one of the first and second robots to hold the first lumber piece at the joint location of the wooden structure;

using the other of the first and second robots to hold the second lumber piece at the joint location of the wooden structure; and

attaching the first lumber piece to the second lumber piece on the assembly table using an attachment device separate from the first and second robots, wherein no other robots are used to position lumber pieces on the assembly table during the assembly of the wooden structure.

62. The method of claim 61, wherein the first and second robots hold the first and second lumber pieces at the joint location of the wooden structure while the lumber pieces are being attached.

63. The method of claim 61, wherein attaching the first lumber piece to the second lumber piece comprises driving a nailing plate into the lumber pieces using a movable platen.

64. The method of claim 63, wherein attaching the first lumber piece to the second lumber piece comprises driving a nailing plate pair into upper and lower surfaces of the lumber pieces, respectively, using upper and lower platen assemblies.

65. The method of claim 61, wherein the robots do not perform any attachment functions for attaching the lumber pieces together.

67. A wooden structure assembly station for assembling a wooden structure, the station comprising:

an assembly table having a length and a width;

a first robot configured to position a first lumber piece on the assembly table at a joint location of the wooden structure, the first robot including a first articulated robot arm;

a second robot, one of the first and second robots being configured to hold the first lumber piece at the joint location of the wooden structure, and the other of the first and second robots being configured to hold a second lumber piece at the joint location of the wooden structure, the second robot including a second articulated robot arm; and

an attachment assembly configured to attach the first lumber piece to the second lumber piece on the assembly table, the attachment assembly being located along the length of the assembly table and extending along the width of the table, the first robot being located along the length of the assembly table on one side of the attachment assembly, and the second robot arm being located along the length of the table on the other side of the attachment assembly.

68. The assembly station of claim 67, wherein the attachment assembly comprises a movable platen configured to drive a nailing plate into the first and second lumber pieces.

69. The assembly station of claim 68, wherein the attachment assembly comprises a second movable platen configured to drive a second nailing plate into the first and second lumber pieces.

70. The assembly station of claim 67, wherein no other robots are used to position lumber pieces on the assembly table during the assembly of the wooden structure.

71. A method of cutting stock lumber pieces for use in assembling a wooden structure, the method comprising:

delivering a first stock lumber piece to a saw along a first saw in-feed line;

cutting the first stock lumber piece in the first saw in-feed line with the saw;

delivering a second stock lumber piece to the saw along a second saw in-feed line; and

cutting the second stock lumber piece in the second saw in-feed line with the saw.

72. The method of claim 71, wherein the first and second saw in-feed lines extend parallel to each other.

73. The method of claim 71, wherein delivering the first stock lumber piece comprises transferring the first stock lumber piece from a first carrier disposed outside of a saw compartment housing the saw to a first holder disposed inside the saw compartment.

74. The method of claim 73, wherein delivering the second stock lumber piece comprises transferring the second stock lumber piece from a second carrier disposed outside of the saw compartment housing the saw to a second holder disposed inside the saw compartment.

75. The method of claim 71, further comprising detecting a position of the first and second stock lumber pieces along the first and second saw in-feed lines.

76. The method of claim 71, wherein delivering the second stock lumber piece to the saw occurs during the cutting of the first stock lumber piece with the saw.

77. The method of 71, further comprising:

determining a distance of one of the first and second stock lumber pieces on the corresponding saw in-feed line from the saw;

delivering said one of the first and second stock lumber pieces to a first location relative to the saw;

securing said one of the first and second stock lumber pieces with a clamp;

measuring a position of said one of the first and second stock lumber pieces relative to a reference point on the clamp;

accounting for a difference between the position of said one of the first and second stock lumber pieces and the reference point; and

cutting said one of the first and second stock lumber pieces based on the difference between the position of said one of the first and second stock lumber pieces and the reference point.

78. The method of claim 77, wherein measuring the position of said one of the first and second stock lumber pieces relative to the reference point on the clamp comprises measuring a distance between the reference point and a bottom of said one of the first and second stock lumber pieces.

79. The method of claim 78, further comprising adjusting a cut height based on the difference between the position of said one of the first and second stock lumber pieces and the reference point.

80. A saw assembly for cutting a stock lumber piece comprising:

a saw compartment;

a robotic arm in the saw compartment;

a saw mounted on the robotic arm; and

first and second, spaced apart lumber in-feed lines configured to move the stock lumber piece through the saw compartment;

wherein the robotic arm is configured to cut the stock lumber on either the first lumber in-feed line or the second lumber in-feed line.

81. The assembly of 80, wherein the robotic arm is configured to move on laterally opposite sides of the stock lumber piece for making cuts starting on either side of the stock lumber piece.

82. The assembly of 80, wherein the saw is configured to cut the stock lumber piece along a plane that intersects the first and second lumber in-feed lines.

83. The assembly of 82, further comprising a clamp moveable along the first lumber in-feed line and configured to clamp the stock lumber piece in place along the first lumber in-feed line, the clamp including a sensor for measuring a distance between the stock lumber piece and the clamp when the stock lumber piece is clamped by the clamp.

84. The assembly of 83, wherein the sensor is disposed at a bottom of the clamp and is configured to measure a distance from the first sensor to a bottom of the stock lumber piece.

85. The assembly of 84, further comprising a second sensor configured to detect a configuration of the clamp.

86. The assembly of 83, further comprising a second clamp moveable along the second lumber in-feed line and configured to clamp the stock lumber piece in place along the second lumber in-feed line, the second clamp including a second sensor for measuring a distance between the stock lumber piece and the second clamp when the stock lumber piece is clamped by the second clamp.

87. The assembly of 80, wherein the robotic arm comprises multiple arm members moveably connected to each other providing the robotic arm with six degrees of freedom of movement.

88. The assembly of 87, wherein the saw is mounted on an end member of the robot arm and is rotatable about the end member.

89. The assembly of 80, wherein the saw comprises a circular saw blade.

90. The assembly of 80, further comprising a holder moveably disposed in the saw compartment and configured to receive the stock lumber delivered along one of the lumber in-feed lines.

91. The assembly of 90, wherein the holder comprises a clamp for locking the stock lumber in position relative to the holder.

92. The assembly of 91, wherein the holder comprises a bottom wall and a pair of side walls extending upward from the bottom wall, at least one of the side walls being moveable relative to the other side wall in order to open and close a space between the side walls to clamp the stock lumber between the side walls.

93. An automated wooden structure manufacturing and assembly system controller comprising one or more processors and computer executable instructions embodied on a computer readable storage medium, the computer executable instructions including instructions for controlling the assembly of a wooden structure, the instructions including:

determining an order of lumber pieces to be used during the assembly of the wooden structure;

after determining the order of lumber pieces, determining a placement of nailing plates on the lumber pieces to form joints between the lumber pieces; and

after determining the placement of the nailing plates, determining a sequence of movements of robots to assemble the wooden structure using the lumber pieces.

94. The controller of claim 93, wherein the movement of the robots is based on a point system assigned to various conditions of the joints of the wooden structure.

95. The controller of claims 93 or 94, further comprising instructions for coordinating movements of an attachment assembly for attaching the nailing plates to the lumber pieces with

the movement of the robots.

96. The controller of any one of claim 94, wherein a plurality of movement sequences for the robots are analyzed and the movement sequence with the highest point total is selected.

97. The controller of claim 94, further comprising instructions for determining an order of the joints to be completed during the assembly of the wooden structure.

98. The controller of claim 97, wherein the order of the joints to be completed is determined based on at least one of a location of the joint, a number of lumber pieces at the joint, and a size of the lumber pieces at the joint.

99. The controller of claim 98, wherein a point system is assigned to various assembly actions and conditions, and the order of the joints to be completed having a highest point total is selected to for the assembly of the wooden structure.

100. A method of selecting stock lumber for use in automated assembly of a wooden structure including a plurality of lumber pieces, the method comprising:

determining an inventory of stock lumber;

selecting the stock lumber needed to produce a first set of lumber piece of the wooden structure based on the inventory; and

determining which lumber pieces in the first set of lumber pieces to produce from each selected stock lumber to maximize the number of lumber pieces produced from the stock lumber.

101. The method of claim 100, further comprising:

selecting the stock lumber needed to produce a second set of lumber pieces of the wooden structure based on the inventory; and

determining which lumber pieces in the second set of lumber pieces to produce from each selected stock lumber.

102. The method of claim 100, further comprising determining a grade of the stock

lumber to be used to produce the wooden structure.

103. The method of claim 100, further comprising producing at least three pieces of lumber from a single piece of stock lumber.

104. A method of selecting stock lumber for use in automated assembly of a wooden structure including a plurality of lumber pieces, the method comprising:

determining an inventory of stock lumber;

selecting the stock lumber needed to produce a first set of lumber piece of the wooden structure based on the inventory by choosing the least expensive lumber required to form a wooden structure having the mechanical properties required of the wooden structure.

105. A computer-implemented method of manufacturing and assembling wooden trusses, the method comprising:

receiving, at a truss manufacture computing device, designs for a plurality of wooden structures;

processing, at the truss manufacture computing device, the designs to identify a recipe for manufacturing and assembling each of the plurality of wooden structures defined in each corresponding design;

processing the plurality of recipes, at the truss manufacture computing device, to identify a requested lumber input of stock lumber;

instructing an automated truss manufacturing system to pre-stage the requested lumber input;

instructing a saw assembly to cut the lumber input based, in part, on the identified recipes; and

instructing a plurality of robots to assemble a first wooden truss according to a first recipe of the identified recipe.

106. The computer-implemented method of claim 105, further comprising:

identifying, at the truss manufacture computing device, a plurality of possible sequences to manufacture a first design of the designs;

simulating, at the truss manufacture computing device, manufacture of the first design according to each of the plurality of possible sequences, wherein each simulated manufacture includes simulation characteristics; and

identifying the recipe by identifying the possible sequence having the preferred simulation characteristics.

107. The computer-implemented method of claim 106, wherein the simulation characteristics include expected time to manufacture, expected joint quality, and expected travel time for robots used in manufacture.

108. The computer-implemented method of claim 105, further comprising:

processing the plurality of recipes, at the truss manufacture computing device, to identify lumber instructions; and

identifying the requested lumber input of stock lumber based on the lumber instructions.

109. The computer-implemented method of claim 105, further comprising:

processing the plurality of recipes, at the truss manufacture computing device, to identify lumber instructions; and

instructing a saw assembly to cut the lumber according to the lumber instructions.

110. The computer-implemented method of claim 108, further comprising:

presenting a prompt, to a user at a user interface in communication with the truss manufacture computing device, to feed stock lumber based on the lumber instructions.

111. The computer-implemented method of claim 105, wherein instructing the plurality of robots further comprises:

identifying a joint for assembly from a first recipe of the recipes; and

instructing a pair of assembly robots to position a pair of lumber pieces defining a joint, according to the recipe.

112. The computer-implemented method of claim 111, further comprising:

identifying a connector plate associated with the joint, according to the recipe;

instructing a plate distribution assembly to obtain the connector plate associated with the joint;

instructing a platen assembly to connect the joint using the connector plate.

113. An automated truss manufacturing system for manufacturing and assembling wooden trusses, said system comprising:

an in-feed station;

a saw station;

an assembly station having a plurality of robots; and

a truss manufacture computing device having a processor and a memory, said processor in communication with said in-feed station, said saw station, and said assembly station, said processor configured to:

receive designs for a plurality of wooden structures;

process the designs to identify a recipe for manufacturing and assembling each of the plurality of wooden structures defined in each corresponding design;

process the plurality of recipes, at the truss manufacture computing device, to identify a requested lumber input of stock lumber;

instruct said in-feed station to pre-stage the requested lumber input;

instruct said saw station to cut the lumber input based, in part, on the identified recipes; and

instruct the plurality of robots to assemble a first wooden truss according to a first recipe of the identified recipe.

114. The automated truss manufacturing system of claim 113, wherein said processor is configured to:

identify a plurality of possible sequences to manufacture a first design of the designs;

simulate manufacture of the first design according to each of the plurality of possible sequences, wherein each simulated manufacture includes simulation characteristics; and

identify the recipe by identifying the possible sequence having the preferred simulation characteristics.

115. The automated truss manufacturing system of claim 1114, wherein the simulation characteristics include expected time to manufacture, expected joint quality, and expected travel time for robots used in manufacture.

116. The automated truss manufacturing system of claim 113, wherein said processor is configured to:

process the plurality of recipes to identify lumber instructions; and

identify the requested lumber input of stock lumber based on the lumber instructions.

117. The automated truss manufacturing system of claim 113, wherein said processor is configured to:

process the plurality of recipes to identify lumber instructions; and

instruct said cutting station to cut the lumber according to the lumber instructions.

118. The automated truss manufacturing system of claim 116, wherein said processor is configured to:

present a prompt, to a user at a user interface in communication with the truss manufacture computing device, to feed stock lumber based on the lumber instructions.

119. The automated truss manufacturing system of claim 113, wherein said processor is configured to:

identify a joint for assembly from a first recipe of the recipes; and

instruct the robots at the assembly station to position a pair of lumber pieces defining a joint, according to the recipe.

120. The automated truss manufacturing system of claim 117, wherein said processor is configured to:

identify a connector plate associated with the joint, according to the recipe;

instruct a plate distribution assembly at the assembly station to obtain the connector plate associated with the joint;

instruct a platen assembly at the assembly station to connect the joint using the connector plate.

121. A computing device comprising:

circuitry configured to:

obtain input data indicative of a wooden structure to be produced by an automated system;

perform, as a function of the obtained input data and present properties of the automated system, validation operations to determine whether an error will be encountered during production of the wooden structure; and

identify, in response to a determination that an error will be encountered and as a function of the validation operations, one or more adjustments to enable production of the wooden structure without the error.

122. The computing device of claim 121, wherein to obtain input data indicative of a wooden structure comprises to obtain input data indicative of a truss to be produced.

123. The computing device of claim 121, wherein to obtain input data indicative of a wooden structure comprises to obtain input data indicative of a set of multiple wooden structures to be produced.

124. The computing device of claim 121, wherein to obtain input data comprises to obtain input data indicative of a target shape, parts, and materials to produce the wooden structure.

125. The computing device of claim 121, wherein to obtain input data comprises to obtain input data indicative of a quantity of the wooden structure to produce.

126. The computing device of claim 121, wherein to perform validation operations comprises to determine whether materials specified in the obtained input data are available to the automated system.

127. The computing device of claim 126, wherein to determine whether materials specified in the obtained input data are available comprises to determine whether nailing plates specified in the obtained input data are available in a nailing plate inventory of the automated system.

128. The computing device of claim 127, wherein to determine whether the nailing plates are available in a nailing plate inventory comprises to determine whether the nailing plates are available in one or more nailing plate magazines of the automated system.

129. The computing device of claim 126, wherein to determine whether materials specified in the obtained input data are available comprises to determine whether lumber specified in the obtained input data is available to the automated system.

130. The computing device of claim 121, wherein to perform validation operations comprises to simulate execution of a recipe indicative of a series of operations of components of the automated system for producing the wooden structure.

131. The computing device of claim 130, wherein to simulate execution of the recipe comprises to simulate execution based on models of components of the automated system.

132. The computing device of claim 131, wherein to simulate execution comprises to simulate execution based on models of assembly robots of the automated system.

133. The computing device of claim 132, wherein the circuitry is further configured to determine, based on the simulated execution, whether one or more of the assembly robots will be unable to create a defined joint of the wooden structure.

134. The computing device of claim 132, wherein the circuitry is further configured to determine, based on the simulated execution, whether one or more of the assembly robots will collide with a component of the automated system.

135. The computing device of claim 132, wherein the circuitry is further configured to determine whether one or more of the assembly robots will be unable to pick up a board or part of the wooden structure at a defined position.

136. The computing device of claim 121, wherein to identify one or more adjustments comprises to:

present the determined error in a user interface; and

receive a user-defined adjustment.

137. The computing device of claim 121, wherein to identify one or more adjustments comprises to identify the adjustment as a function of a predefined set of adjustments mapped to predefined errors.

138. The computing device of claim 121, wherein to identify one or more adjustments comprises to determine an offset, a rotation angle, or a reflection along one or more axes for the wooden structure.

139. The computing device of claim 121, wherein to identify one or more adjustments comprises to determine an adjustment as a function of a set of materials available to the automated system.

140. The computing device of claim 139, wherein to determine an adjustment as a function of a set of materials available to the automated system comprises to identify a replacement nailing plate as a function of dimensions of a nailing plate specified in the input data and dimensions of one or more nailing plates available to the automated system.

141. The computing device of claim 140, wherein to identify a replacement nailing plate as a function of dimensions of the nailing plate specified in the input data comprises to identify,

as a replacement nailing plate, a nailing plate that is available to the automated system and that has dimensions that are greater than or equal to the dimensions of the nailing plate specified in the input data.

142. The computing device of claim 121, wherein to determine an adjustment as a function of a set of materials available to the automated system comprises to identify replacement lumber as a function of a grade of lumber specified in the input data and one or more grades of lumber available to the automated system.

143. The computing device of claim 142, wherein to identify replacement lumber as a function of a grade of lumber specified in the input data comprises to identify, as replacement lumber, lumber that is available to the automated system and that has a grade that is greater than or equal to the grade of the lumber specified in the input data.

144. The computing device of claim 121, wherein the circuitry is further configured to:

store the one or more adjustments; and

simulate execution of a recipe for producing the wooden structure with the one or more adjustments applied to operations of components of the automated system defined in the recipe.

145. A method comprising:

obtaining, by a computing device, input data indicative of a wooden structure to be produced by an automated system;

performing, by the computing device and as a function of the obtained input data and present properties of the automated system, validation operations to determine whether an error will be encountered during production of the wooden structure; and

identifying, by the computing device and in response to a determination that an error will be encountered and as a function of the validation operations, one or more adjustments to enable production of the wooden structure without the error.

146. The method of claim 145, wherein obtaining input data indicative of a wooden structure comprises obtaining input data indicative of a truss to be produced.

147. The method of claim 145, wherein obtaining input data indicative of a wooden structure comprises obtaining input data indicative of a set of multiple wooden structures to be produced.

148. The method of claim 145, wherein obtaining input data comprises obtaining input data indicative of a target shape, parts, and materials to produce the wooden structure.

149. The method of claim 145, wherein obtaining input data comprises obtaining input data indicative of a quantity of the wooden structure to produce.

150. The method of claim 145, wherein performing validation operations comprises determining whether materials specified in the obtained input data are available to the automated system.

151. The method of claim 150, wherein determining whether materials specified in the obtained input data are available comprises determining whether nailing plates specified in the obtained input data are available in a nailing plate inventory of the automated system.

152. The method of claim 151, wherein determining whether the nailing plates are available in a nailing plate inventory comprises determining whether the nailing plates are available in one or more nailing plate magazines of the automated system.

153. The method of claim 150, wherein determining whether materials specified in the obtained input data are available comprises determining whether lumber specified in the obtained input data is available to the automated system.

154. The method of claim 145, wherein performing validation operations comprises simulating execution of a recipe indicative of a series of operations of components of the automated system for producing the wooden structure.

155. The method of claim 154, wherein simulating execution of the recipe comprises simulating execution based on models of components of the automated system.

156. The method of claim 155, wherein simulating execution comprises simulating execution based on models of assembly robots of the automated system.

157. The method of claim 156, further comprising determining, by the computing device and based on the simulated execution, whether one or more of the assembly robots will be unable to create a defined joint of the wooden structure.

158. The method of claim 156, further comprising determining, by the computing device and based on the simulated execution, whether one or more of the assembly robots will collide with a component of the automated system.

159. The method of claim 156, further comprising determining, by the computing device, whether one or more of the assembly robots will be unable to pick up a board or part of the wooden structure at a defined position.

160. The method of claim 145, wherein identifying one or more adjustments comprises:
presenting, by the computing device, the determined error in a user interface; and
receiving, by the computing device, a user-defined adjustment.

161. The method of claim 145, wherein identifying one or more adjustments comprises identifying the adjustment as a function of a predefined set of adjustments mapped to predefined errors.

162. The method of claim 145, wherein identifying one or more adjustments comprises determining an offset, a rotation angle, or a reflection along one or more axes for the wooden structure.

163. The method of claim 145, wherein identifying one or more adjustments comprises determining an adjustment as a function of a set of materials available to the automated system.

164. The method of claim 163, wherein determining an adjustment as a function of a set of materials available to the automated system comprises identifying a replacement nailing plate

as a function of dimensions of a nailing plate specified in the input data and dimensions of one or more nailing plates available to the automated system.

165. The method of claim 164, wherein identifying a replacement nailing plate as a function of dimensions of the nailing plate specified in the input data comprises identifying, as a replacement nailing plate, a nailing plate that is available to the automated system and that has dimensions that are greater than or equal to the dimensions of the nailing plate specified in the input data.

166. The method of claim 145, wherein determining an adjustment as a function of a set of materials available to the automated system comprises identifying replacement lumber as a function of a grade of lumber specified in the input data and one or more grades of lumber available to the automated system.

167. The method of claim 166, wherein identifying replacement lumber as a function of a grade of lumber specified in the input data comprises identifying, as replacement lumber, lumber that is available to the automated system and that has a grade that is greater than or equal to the grade of the lumber specified in the input data.

168. The method of claim 145, further comprising:

storing, by the computing device, the one or more adjustments; and

simulating, by the computing device, execution of a recipe for producing the wooden structure with the one or more adjustments applied to operations of components of the automated system defined in the recipe.

169. One or more machine-readable storage media comprising a plurality of instructions stored thereon that, in response to being executed, cause a computing device to perform the method of any of claims 145-168.

170. A computing device comprising:

circuitry configured to:

obtain recipe generation input data indicative of one or more parameters to be satisfied in coordinating operations of components of an automated system to produce one or more wooden trusses;

identify, as a function of the obtained recipe generation input data, one or more unique wooden trusses to be produced; and

generate, for each unique wooden truss, a recipe indicative of operations to be performed by the components of the automated system to produce the corresponding wooden truss in a joint-by-joint extrusion sequence.

171. The computing device of claim 170, wherein to obtain recipe generation input data comprises to obtain shape data indicative of a target shape for a wooden truss.

172. The computing device of claim 171, wherein to obtain recipe generation input data comprises to additionally obtain one or more of production data indicative of quantities of the unique wooden trusses to produce, intervention data indicative of substitute materials to be used or a rotation to be applied to one or more wooden truss parts, or assembler parameter data indicative of one or more offsets, dimensions, or limits for a component of the automated system.

173. The computing device of claim 170, wherein to identify one or more unique trusses comprises to identify the one or more unique trusses as a function of a truss identifier, a job name, a truss label, or a batch name.

174. The computing device of claim 170, wherein to identify one or more unique trusses comprises to filter out non-unique trusses from the obtained recipe generation input data to reduce an amount of time for recipe generation.

175. The computing device of claim 170, wherein to generate a recipe indicative of operations to be performed comprises to:

produce a list of joints for a selected wooden truss;

determine, from the produced list of joints, an ordered set of joints;

determine, as a function of the ordered set of joints, a set of assembly operations; and

calculate, for each assembly operation in the set of assembly operations, a set of recipe operations.

176. The computing device of claim 175, wherein to calculate recipe operations comprises to:

calculate a set of valid primary robot pickup options indicative of locations or orientations that would enable a primary robot of the automated system to pick up a part of the wooden truss for assembly;

calculate a set of valid secondary robot pickup options indicative of locations or orientations that would enable a secondary robot of the automated system to pick up a part of the wooden truss for assembly;

determine pairs of pickup options from the valid primary robot pickup options and valid secondary robot pickup options;

determine, as a function of target criteria, a pickup option pair from the set to be designated as a best pickup option pair.

177. The computing device of claim 176, wherein the circuitry is further configured to:

determine, after the determination of the best pickup option pair, a set of recipe operations deemed to be necessary; and

determine a set of intermediate recipe operations.

178. The computing device of claim 175, wherein to produce a list of joints for the selected wooden truss comprises to:

convert truss shape data associated with the selected wooden truss from the obtained recipe generation input data to points and polygons in a two-dimensional orthogonal coordinate system;

apply one or more interventions defined in intervention data in the obtained recipe generation input data to the converted truss shape data to adjust an orientation associated with the selected wooden truss;

convert part angles defined in the converted truss shape data to a predefined range;

shorten lengths defined in the converted truss shape data for a subset of parts of the selected wooden truss for production tolerance; and

generate, for each of multiple nailing plates associated with the selected wooden truss, a set of one or more joint objects indicative of joints of the selected wooden truss.

179. The computing device of claim 178, wherein to apply one or more interventions comprises to apply one or more rotations, flip the selected wooden truss along an axis or the two-dimensional coordinate system, or apply an offset to the selected wooden truss.

180. The computing device of claim 178, wherein to shorten lengths for a subset of the parts comprises to:

identify a set of parts that are not perimeter parts and not wedge parts; and

shorten non-vertical parts in the set by a greater amount than vertical parts.

181. The computing device of claim 180, wherein to shorten non-vertical parts by a greater amount than vertical parts comprises to:

shorten non-vertical parts by one eighth of an inch; and

shorten vertical parts by one sixteenth of an inch.

182. The computing device of claim 178, wherein to generate, for each of multiple nailing plates associated with the selected wooden truss, a set of one or more joint objects, comprises to:

temporarily create, in a memory of the computing device and for each nailing plate, a rectangle having dimensions that are smaller than the corresponding nailing plate to determine an association between the nailing plates and parts of the wooden truss; and

generate joint objects that include data indicative of one or more nailing plates and parts of the wooden truss associated with each corresponding joint.

183. The computing device of claim 175, wherein to produce a list of joints for the selected wooden truss comprises to:

calculate a first joint for the selected wooden truss; and

calculate an order for remaining joints of the selected wooden truss.

184. The computing device of claim 183, wherein to calculate a first joint for the selected wooden truss comprises to:

determine, as a function of a set of criteria, a best first part in each joint;

determine an order for remaining parts in each joint;

calculate a joint score for each joint; and

designate the first joint as the joint with the highest joint score.

185. The computing device of claim 184, wherein to determine the best first part in a joint comprises to iterate through each of the parts associated with the joint and determine whether a part associated with a present iteration is the best first part as a function of whether the part is a wedge, whether the part satisfies a predefined length associated with a clamp of the automated system, whether the part has an angle that satisfies a predefined range, whether an overlap between the part and an assembly table of the automated system satisfies a predefined percentage, whether the part defines a perimeter of the selected wooden truss, and whether the part is within a predefined zone in the automated system.

186. The computing device of claim 184, wherein to determine an order for remaining parts in each joint comprises, for a selected joint, to:

determine a part score for each remaining part associated with the selected joint; and

order the remaining parts as a function of the determined part score for each part.

187. The computing device of claim 186, wherein to determine a part score comprises to:

adjust the part score as a function of whether the selected part touches one or more other parts positioned in the selected joint;

adjust the part score as a function of whether the selected part defines a perimeter of the selected wooden truss; and

adjust the part score as a function of whether the selected part defines a wedge of the selected wooden truss.

188. The computing device of claim 187, wherein to adjust the part score as a function of whether the selected part touches one or more other parts positioned in the selected joint comprises to increase the part score based on a number of parts in the selected joint that are touched by the selected part.

189. The computing device of claim 184, wherein to calculate the joint score for a selected joint comprises to:

initially set the joint score for the selected joint to zero;

increase the joint score as a function of joint score factors, wherein the joint score factors include one or more of whether a first part of the joint defines a wedge, whether a second part of the joint defines a wedge, whether the first part of the joint defines a perimeter of the selected wooden truss, whether the second part of the joint defines a perimeter of the selected wooden truss, whether the first part of the joint satisfies a predefined length to be held by a clamp of the automated system, whether the second part of the joint satisfied the predefined length to be held by the clamp, a percentage of the first part of the joint that overlaps a surface of an assembly table of the automated system, a percentage of the second part of the joint that overlaps the surface of the assembly table of the automated system, or an angle between the first part and the second part of the joint.

190. The computing device of claim 183, wherein to calculate an order for remaining joints of the selected wooden truss comprises, for each joint in a set of remaining joints associated with the selected wooden truss, to:

select the joint for analysis;

determine whether the selected joint is a next best joint for the order of remaining joints as a function of one or more of whether the selected joint will cause a truss collision if picked later, whether all parts of the joint are already positions with one or more previous joints in the joint order, whether the selected joint defines a splice, whether the selected joint has more parts than another joint determined to be the next best joint, whether the selected joint is closer to a defined starting point associated with the selected wooden truss, or whether the selected joint has a higher total length of parts than another joint determined to be the next best joint.

191. The computing device of claim 190, wherein to determine whether the selected joint is the next best joint as function of whether the selected joint has more parts than another joint determined to be the next best joint comprises to determine that the selected joint is the next best join in response to a determination that the selected joint has more parts that define a perimeter of the selected wooden truss within a predefined reach distance of a robot of the automated system than the other joint determined to be the next best joint.

192. The computing device of claim 175, wherein to determine, as a function of the ordered set of joints, a set of assembly operations, comprises, for each joint, to:

select the joint for analysis;

determine the assembly operations for the selected joint as a function of whether the selected joint has two parts; and

add, for each part in the joint, one or more assembly operations for a new part.

193. The computing device of claim 192, wherein to determine the assembly operations for the selected joint as a function of whether the selected joint has two parts comprises to determine, in response to a determination that the joint has two parts, assembly operations as a function of whether the joint has a bottom plate.

194. The computing device of claim 193, wherein to determine the assembly operations as a function of whether the joint has a bottom plate comprises to add, in response to a determination that the joint has a bottom plate and that the joint has all parts, a top plate for the joint as a new assembly operation.

195. The computing device of claim 193, wherein to determine the assembly operations as a function of whether the joint has a bottom plate comprises to determine, in response to a determination that the joint does not have a bottom plate, a set of corresponding assembly operations.

196. The computing device of claim 195, wherein to determine a set of corresponding assembly operations comprises to:

add, in response to a determination that the joint does not have all parts, a bottom plate for the joint as a new assembly operation; or

add, in response to a determination that the joint does have all parts, a top plate and a bottom plate to the joint as new assembly operations.

197. The computing device of claim 192, wherein to add, for each part in the joint, one or more assembly operations for a new part comprises to:

select a part for analysis,

for each remaining joint of the selected wooden truss, select a next other joint; and

determine one or more assembly operations as a function of whether the selected part is a third part of another joint.

198. The computing device of claim 197, wherein to determine one or more assembly operations as a function of whether the selected part is a third part comprises to determine, in response to a determination that the selected part is a third part, the one or more assembly operations as a function of whether the other joint has a bottom plate.

199. The computing device of claim 198, wherein to determine the one or more assembly operations as a function of whether the other joint has a bottom plate comprises to:

add, in response to a determination that the other joint has a bottom plate and the other joint has all parts, a new assembly operation to add the top plate for the other joint; or

determine, in response to a determination that the other joint does not have a bottom plate, one or more assembly operations as a function of whether the other joint has all parts.

200. The computing device of claim 199, wherein to determine one or more assembly operations as a function of whether the other joint has all parts comprises to:

add, in response to a determination that the other joint does not have all parts, an assembly operation to add the bottom plate for the other joint; or

add, in response to a determination that the other joint does have all parts, an assembly operation to add a top plate and the bottom plate to the other joint.

201. The computing device of claim 197, wherein to add, for each part in the joint, one or more assembly operations further comprises to determine assembly operations as a function of whether the selected joint has a bottom plate.

202. The computing device of claim 176, wherein to calculate a set of valid primary robot pickup options comprises to:

calculate a set of pickup points for the part; and

calculate, in response to a determination that the robot is to pick up a new part, and for each of a set of available other joints in the selected wooden truss, and as a function of whether the other joint is within a defined distance of a present joint and whether the other joint is connected to the part to be picked up by the robot, a set of pickup points for each other part in the selected other joint.

203. The computing device of claim 202, wherein to calculate a set of pickup points comprises to:

calculate a closest pickup point that does will not cause interference with operations of a press of the automated system; and

determine, for each of multiple directions from a center of the part, a set of valid pickup points as a function of a prioritization of pickup options for utilization of a clamp, a utilization of suction device, and movement of at least one component of the automated system to avoid a collision with the robot or the part.

204. The computing device of claim 203, wherein the circuitry is further configured to calculate, for each pickup option in a set of pickup options for a part, an interference polygon for a first tool of the robot, an interference polygon for a second tool of the robot, a tool center interference polygon, a cylindrical combined interference polygon, an effective tool pickup area, and a tool interference with part polygon.

205. The computing device of claim 204, wherein the circuitry is further configured, for each of a set of pickup options and for each of multiple increments of length along a part of the selected wooden truss, to:

calculate a final tool position interference;

calculate a tool trajectory interference;

calculate a press trajectory interference; and

determine a set of subsequent operations as a function of the calculated interferences.

206. The computing device of claim 205, wherein to determine a set of subsequent operations comprises to determine which of a set of directions of movement of a component within the automated system will avoid a collision in picking up or moving a part of the selected wooden truss.

207. The computing device of claim 170, wherein the circuitry is further configured to:

assign primary and secondary roles to each of a first robot and a second robot of the automated system as a function of a set of role assignment factors; and

calculate gantry and robot arm positions for the first robot and the second robot.

208. The computing device of claim 207, wherein to assign primary and secondary roles as a function of role assignment factors comprises to assign roles as a function of whether a new part is being added in a current assembly operation, assign roles as a function of whether only one robot is needed for the current assembly operation, or assign roles a function of relative locations of tools of each of the first and second robots.

209. The computing device of claim 207, wherein to calculate gantry and robot arm positions for the first robot and the second robot comprises to:

calculate a second robot arm polygon;

calculate a press displacement;

calculate a first robot trajectory and a second robot trajectory;

calculate a first robot gantry position and a second robot gantry position; and

calculate, as a function of a gantry y position and final tool position data, first robot and second robot arm orientations.

210. The computing device of claim 177, wherein to determine, after the determination of the best pickup option pair, a set of recipe operations deemed to be necessary comprises to

determine necessary recipe operations as a function of whether a part to be added in a current operation exceeds a defined length or weight limit, whether rotation of the part will result in interference, and a target number of presses to be applied to a nailing plate based on a relative size of a surface of a press of the automated system to the nailing plate.

211. A method comprising:

obtaining, by a computing device, recipe generation input data indicative of one or more parameters to be satisfied in coordinating operations of components of an automated system to produce one or more wooden trusses;

identifying, by the computing device and as a function of the obtained recipe generation input data, one or more unique wooden trusses to be produced; and

generating, by the computing device and for each unique wooden truss, a recipe indicative of operations to be performed by the components of the automated system to produce the corresponding wooden truss in a joint-by-joint extrusion sequence.

212. The method of claim 211, wherein obtaining recipe generation input data comprises obtaining shape data indicative of a target shape for a wooden truss.

213. The method of claim 212, wherein obtaining recipe generation input data comprises additionally obtaining one or more of production data indicative of quantities of the unique wooden trusses to produce, intervention data indicative of substitute materials to be used or a rotation to be applied to one or more wooden truss parts, or assembler parameter data indicative of one or more offsets, dimensions, or limits for a component of the automated system.

214. The method of claim 211, wherein identifying one or more unique trusses comprises identifying the one or more unique trusses as a function of a truss identifier, a job name, a truss label, or a batch name.

215. The method of claim 211, wherein identifying one or more unique trusses comprises filtering out non-unique trusses from the obtained recipe generation input data to reduce an amount of time for recipe generation.

216. The method of claim 211, wherein generating a recipe indicative of operations to be performed comprises:

producing a list of joints for a selected wooden truss;

determining, from the produced list of joints, an ordered set of joints;

determining, as a function of the ordered set of joints, a set of assembly operations; and

calculating, for each assembly operation in the set of assembly operations, a set of recipe operations.

217. The method of claim 216, wherein calculating recipe operations comprises:

calculating a set of valid primary robot pickup options indicative of locations or orientations that would enable a primary robot of the automated system to pick up a part of the wooden truss for assembly;

calculating a set of valid secondary robot pickup options indicative of locations or orientations that would enable a secondary robot of the automated system to pick up a part of the wooden truss for assembly;

determining pairs of pickup options from the valid primary robot pickup options and valid secondary robot pickup options;

determining, as a function of target criteria, a pickup option pair from the set to be designated as a best pickup option pair.

218. The method of claim 217, further comprising:

determining, by the computing device and after the determination of the best pickup option pair, a set of recipe operations deemed to be necessary; and

determining, by the computing device, a set of intermediate recipe operations.

219. The method of claim 216, wherein producing a list of joints for the selected wooden truss comprises:

converting truss shape data associated with the selected wooden truss from the obtained recipe generation input data to points and polygons in a two-dimensional orthogonal coordinate system;

applying one or more interventions defined in intervention data in the obtained recipe generation input data to the converted truss shape data to adjust an orientation associated with the selected wooden truss;

converting part angles defined in the converted truss shape data to a predefined range;

shortening lengths defined in the converted truss shape data for a subset of parts of the selected wooden truss for production tolerance; and

generating, for each of multiple nailing plates associated with the selected wooden truss, a set of one or more joint objects indicative of joints of the selected wooden truss.

220. The method of claim 219, wherein applying one or more interventions comprises applying one or more rotations, flipping the selected wooden truss along an axis or the two-dimensional coordinate system, or applying an offset to the selected wooden truss.

221. The method of claim 219, wherein shortening lengths for a subset of the parts comprises:

identifying a set of parts that are not perimeter parts and not wedge parts; and

shortening non-vertical parts in the set by a greater amount than vertical parts.

222. The method of claim 221, wherein shortening non-vertical parts by a greater amount than vertical parts comprises:

shortening non-vertical parts by one eighth of an inch; and

shortening vertical parts by one sixteenth of an inch.

223. The method of claim 219, wherein generating, for each of multiple nailing plates associated with the selected wooden truss, a set of one or more joint objects, comprises:

temporarily creating, in a memory of the computing device and for each nailing plate, a rectangle having dimensions that are smaller than the corresponding nailing plate to determine an association between the nailing plates and parts of the wooden truss; and

generating joint objects that include data indicative of one or more nailing plates and parts of the wooden truss associated with each corresponding joint.

224. The method of claim 216, wherein to produce a list of joints for the selected wooden truss comprises:

calculating a first joint for the selected wooden truss; and

calculating an order for remaining joints of the selected wooden truss.

225. The method of claim 224, wherein calculating a first joint for the selected wooden truss comprises:

determining, as a function of a set of criteria, a best first part in each joint;

determining an order for remaining parts in each joint;

calculating a joint score for each joint; and

designating the first joint as the joint with the highest joint score.

226. The method of claim 225, wherein determining the best first part in a joint comprises iterating through each of the parts associated with the joint and determining whether a part associated with a present iteration is the best first part as a function of whether the part is a wedge, whether the part satisfies a predefined length associated with a clamp of the automated system, whether the part has an angle that satisfies a predefined range, whether an overlap between the part and an assembly table of the automated system satisfies a predefined percentage, whether the part defines a perimeter of the selected wooden truss, and whether the part is within a predefined zone in the automated system.

227. The method of claim 225, wherein determining an order for remaining parts in each joint comprises, for a selected joint:

determining a part score for each remaining part associated with the selected joint; and

ordering the remaining parts as a function of the determined part score for each part.

228. The method of claim 227, wherein determining a part score comprises:

adjusting the part score as a function of whether the selected part touches one or more other parts positioned in the selected joint;

adjusting the part score as a function of whether the selected part defines a perimeter of the selected wooden truss; and

adjusting the part score as a function of whether the selected part defines a wedge of the selected wooden truss.

229. The method of claim 228, wherein adjusting the part score as a function of whether the selected part touches one or more other parts positioned in the selected joint comprises increasing the part score based on a number of parts in the selected joint that are touched by the selected part.

230. The method of claim 225, wherein calculating the joint score for a selected joint comprises:

initially setting the joint score for the selected joint to zero;

increasing the joint score as a function of joint score factors, wherein the joint score factors include one or more of whether a first part of the joint defines a wedge, whether a second part of the joint defines a wedge, whether the first part of the joint defines a perimeter of the selected wooden truss, whether the second part of the joint defines a perimeter of the selected wooden truss, whether the first part of the joint satisfies a predefined length to be held by a clamp of the automated system, whether the second part of the joint satisfied the predefined length to be held by the clamp, a percentage of the first part of the joint that overlaps a surface of an assembly table of the automated system, a percentage of the second part of the joint that overlaps the surface of the assembly table of the automated system, or an angle between the first part and the second part of the joint.

231. The method of claim 224, wherein calculating an order for remaining joints of the selected wooden truss comprises, for each joint in a set of remaining joints associated with the selected wooden truss:

selecting the joint for analysis;

determining whether the selected joint is a next best joint for the order of remaining joints as a function of one or more of whether the selected joint will cause a truss collision if picked later, whether all parts of the joint are already positioned with one or more previous joints in the joint order, whether the selected joint defines a splice, whether the selected joint has more parts

than another joint determined to be the next best joint, whether the selected joint is closer to a defined starting point associated with the selected wooden truss, or whether the selected joint has a higher total length of parts than another joint determined to be the next best joint.

232. The method of claim 231, wherein determining whether the selected joint is the next best joint as function of whether the selected joint has more parts than another joint determined to be the next best joint comprises determining that the selected joint is the next best join in response to a determination that the selected joint has more parts that define a perimeter of the selected wooden truss within a predefined reach distance of a robot of the automated system than the other joint determined to be the next best joint.

233. The method of claim 216, wherein determining, as a function of the ordered set of joints, a set of assembly operations, comprises, for each joint:

selecting the joint for analysis;

determining the assembly operations for the selected joint as a function of whether the selected joint has two parts; and

adding, for each part in the joint, one or more assembly operations for a new part.

234. The method of claim 233, wherein determining the assembly operations for the selected joint as a function of whether the selected joint has two parts comprises determining, in response to a determination that the joint has two parts, assembly operations as a function of whether the joint has a bottom plate.

235. The method of claim 234, wherein determining the assembly operations as a function of whether the joint has a bottom plate comprises adding, in response to a determination that the joint has a bottom plate and that the joint has all parts, a top plate for the joint as a new assembly operation.

236. The method of claim 234, wherein determining the assembly operations as a function of whether the joint has a bottom plate comprises determining, in response to a determination that the joint does not have a bottom plate, a set of corresponding assembly operations.

237. The method of claim 236, wherein determining a set of corresponding assembly operations comprises:

adding, in response to a determination that the joint does not have all parts, a bottom plate for the joint as a new assembly operation; or

adding, in response to a determination that the joint does have all parts, a top plate and a bottom plate to the joint as new assembly operations.

238. The method of claim 233, wherein adding, for each part in the joint, one or more assembly operations for a new part comprises:

selecting a part for analysis,

for each remaining joint of the selected wooden truss, selecting a next other joint; and

determining one or more assembly operations as a function of whether the selected part is a third part of another joint.

239. The method of claim 238, wherein determining one or more assembly operations as a function of whether the selected part is a third part comprises determining, in response to a determination that the selected part is a third part, the one or more assembly operations as a function of whether the other joint has a bottom plate.

240. The method of claim 239, wherein determining the one or more assembly operations as a function of whether the other joint has a bottom plate comprises:

adding, in response to a determination that the other joint has a bottom plate and the other joint has all parts, a new assembly operation to add the top plate for the other joint; or

determining, in response to a determination that the other joint does not have a bottom plate, one or more assembly operations as a function of whether the other joint has all parts.

241. The method of claim 240, wherein determining one or more assembly operations as a function of whether the other joint has all parts comprises:

adding, in response to a determination that the other joint does not have all parts, an assembly operation to add the bottom plate for the other joint; or

adding, in response to a determination that the other joint does have all parts, an assembly operation to add a top plate and the bottom plate to the other joint.

242. The method of claim 238, wherein adding, for each part in the joint, one or more assembly operations further comprises determining assembly operations as a function of whether the selected joint has a bottom plate.

243. The method of claim 217, wherein calculating a set of valid primary robot pickup options comprises:

calculating a set of pickup points for the part; and

calculating, in response to a determination that the robot is to pick up a new part, and for each of a set of available other joints in the selected wooden truss, and as a function of whether the other joint is within a defined distance of a present joint and whether the other joint is connected to the part to be picked up by the robot, a set of pickup points for each other part in the selected other joint.

244. The method of claim 243, wherein calculating a set of pickup points comprises:

calculating a closest pickup point that does will not cause interference with operations of a press of the automated system; and

determining, for each of multiple directions from a center of the part, a set of valid pickup points as a function of a prioritization of pickup options for utilization of a clamp, a utilization of suction device, and movement of at least one component of the automated system to avoid a collision with the robot or the part.

245. The method of claim 244, further comprising calculating, by the computing device and for each pickup option in a set of pickup options for a part, an interference polygon for a first tool of the robot, an interference polygon for a second tool of the robot, a tool center interference polygon, a cylindrical combined interference polygon, an effective tool pickup area, and a tool interference with part polygon.

246. The method of claim 245, further comprising, for each of a set of pickup options and for each of multiple increments of length along a part of the selected wooden truss:

calculating a final tool position interference;

calculating a tool trajectory interference;

calculating a press trajectory interference; and

determining a set of subsequent operations as a function of the calculated interferences.

247. The method of claim 246, wherein determining a set of subsequent operations comprises determining which of a set of directions of movement of a component within the automated system will avoid a collision in picking up or moving a part of the selected wooden truss.

248. The method of claim 211, further comprising:

assigning, by the computing device, primary and secondary roles to each of a first robot and a second robot of the automated system as a function of a set of role assignment factors; and

calculating, by the computing device, gantry and robot arm positions for the first robot and the second robot.

249. The method of claim 248, wherein assigning primary and secondary roles as a function of role assignment factors comprises assigning roles as a function of whether a new part is being added in a current assembly operation, assigning roles as a function of whether only one robot is needed for the current assembly operation, or assigning roles a function of relative locations of tools of each of the first and second robots.

250. The method of claim 248, wherein calculating gantry and robot arm positions for the first robot and the second robot comprises:

calculating a second robot arm polygon;

calculating a press displacement;

calculating a first robot trajectory and a second robot trajectory;

calculating a first robot gantry position and a second robot gantry position; and

calculating, as a function of a gantry y position and final tool position data, first robot and second robot arm orientations.

251. The method of claim 218, wherein determining, after the determination of the best pickup option pair, a set of recipe operations deemed to be necessary comprises determining necessary recipe operations as a function of whether a part to be added in a current operation exceeds a defined length or weight limit, whether rotation of the part will result in interference, and a target number of presses to be applied to a nailing plate based on a relative size of a surface of a press of the automated system to the nailing plate.

252. One or more machine-readable storage media comprising a plurality of instructions stored thereon that, in response to being executed, cause a computing device to perform the method of any of claims 211-251.

253. A computing device comprising:

circuitry configured to:

obtain lumber selection input data indicative of one or more parameters to be utilized in the selection of lumber for the production of one or more wooden trusses by an automated system; and

select, as a function of the one or more parameters and characteristics of lumber available in a lumber inventory of the automated system, a set of lumber pieces from the lumber inventory to satisfy an efficiency target in the automated production of the one or more wooden trusses.

254. The computing device of claim 253, wherein to obtain lumber selection input data comprises to obtain one or more of assembly recipe data indicative of operations to be performed by the automated system to produce the one or more wooden trusses, production data indicative of quantities of the wooden trusses to be produced in one or more batches, line data indicative of a status of one or more in-feed lines of the automated system, inventory data indicative of the characteristics of lumber available in the lumber inventory, saw parameter data indicative of parameters of a saw of the automated system to be used to cut the lumber, or standard length configuration data indicative of one or more lengths defined for pieces of lumber to be used by the automated system.

255. The computing device of claim 253, wherein the circuitry is further to determine, as a function of the lumber selection input data, a set of parts to be used in the production of wooden trusses in a batch.

256. The computing device of claim 255, wherein the circuitry is further configured to:

determine a set of boards for potential use in a batch of one or more wooden trusses to be produced;

determine a set of potential parts for each board in the set of boards, including identifying, as a function of the parameters, a best board from which to produce the potential parts from lumber in the lumber inventory having a grade equal to a grade specified in connection with a design for a wooden truss to be produced and from lumber in the lumber inventory having a grade that is greater than the grade specified in connection with the design; and

remove, from a list of remaining parts to be produced, one or more of the potential parts to be produced from the determined best board.

257. The computing device of claim 256, wherein the circuitry is further configured to determine the best board to be used for each of the remaining parts to be produced.

258. The computing device of claim 255, wherein to determine the set of parts to be used in the production of wooden trusses in the batch comprises to:

determine whether the batch is queued for a first line or a second line of the automated system;

determine, in response to a determination that the batch is queued for the first line or the second line, whether a wooden truss remains to be produced from an assembly recipe defined in the lumber selection input data;

determine, in response to a determination that a wooden truss remains to be produced from the assembly recipe, whether a part remains to be produced for the wooden truss; and

calculate, in response to a determination that a part remains to be produced for the wooden truss, a pickup location indicative of a length along the part where a robot of the automated system will pick up the part.

259. The computing device of claim 258, wherein the circuitry is further configured to:

determine whether the pickup location satisfies a predefined length threshold; and

rotate, in response to a determination that the pickup location does not satisfy the predefined length threshold, the part by 180 degrees.

260. The computing device of claim 258, wherein the circuitry is further configured to:

determine, in response to a determination that the pickup location does not satisfy the predefined length threshold, whether the pickup location satisfies a percentage of part length threshold; and

rotate, in response to a determination that the pickup location satisfies the percentage of part length threshold, the part by 180 degrees.

261. The computing device of claim 260, wherein to determine whether the pickup location satisfies the percentage of part length threshold comprises to determine whether the pickup location is greater than 70% of the part length.

262. The computing device of claim 260, wherein the circuitry is further configured to determine, in response to a determination that the pickup location does not satisfy the percentage of part length threshold, not to rotate the part.

263. The computing device of claim 256, wherein to determine a set of boards for potential use in the batch comprises to:

select a board from the lumber inventory;

determine whether a width of the selected board satisfies a width threshold for a part in the batch to be produced from the selected board;

determine, in response to a determination that the width of the selected board satisfies the width threshold, whether a length of the selected board satisfies a length threshold for the part; and

determine, in response to a determination that the length satisfies the length threshold, a set of one or more responsive operations as a function of a comparison between a grade of the selected board and a defined grade for the part.

264. The computing device of claim 263, wherein the circuitry is further configured to exclude the board from the set of boards for potential use in response to a determination that the width of the selected board does not satisfy the width threshold or that the length of the selected board does not satisfy the length threshold.

265. The computing device of claim 263, wherein to determine a set of one or more responsive operations as a function of a comparison between the grade of the selected board and the defined grade for the part comprises to:

determine whether the grade of the selected board is equal to the defined grade for the part;

determine, in response to a determination that the grade of the selected board is not equal to the defined grade for the part, whether the grade of the selected board is greater than the defined grade for the part;

determine, in response to a determination that the grade of the selected board is greater than the defined grade for the part, whether the defined grade for the part is less than a grade of another board in the set of boards analyzed for inclusion in the set of boards for potential use and that has a grade that is greater than the defined grade for the part;

determine, in response to a determination that the grade of the selected board is less than the grade of the other board, whether the length of the selected board is less than a length of the other board; and

add, in response to a determination that the length of the selected board is less than the length of the other board, the selected board to the set of boards for potential use.

266. The computing device of claim 256, wherein to determine a set of potential parts for each board in the set of boards comprises to:

select a board from the set of boards for potential use; and

determine, for each part in the batch, whether to add the part to the set of potential parts for the selected board, as a function of a comparison of a grade of the selected board to a defined grade for the part, a comparison of the width of the selected board to a width for the part, and a determination of whether the part will fit on the selected board.

267. The computing device of claim 256, wherein to determine the best board comprises to:

select a board from the set of boards for potential use in the batch;

determine, as a function of a number of parts to be produced from the selected board, a set of permutations of possible orderings for the parts;

define a set of groups of parts from the set of permutations of possible orderings for the parts; and

determine the best board as a function of the defined set of groups of parts.

268. The computing device of claim 267, wherein the circuitry is further configured to:

determine, for a part in a group in the defined set of groups of parts, whether the part is the first part in the possible ordering of parts associated with the group; and

selectively, rotate the board in response to a determination that the part is the first part.

269. The computing device of claim 268, wherein to selectively rotate the board comprises to rotate the board to orient a straight edge of the part with a straight edge of the board.

270. The computing device of claim 268, wherein the circuity is further configured to determine, in response to a determination that the part is not the first part, whether the selected part can be rotated.

271. The computing device of claim 270, wherein to determine whether the part can be rotated comprises to determine whether rotation of the part would satisfy one or more size parameters.

272. The computing device of claim 271, wherein to determine whether rotation of the part would satisfy one or more size parameters comprises to determine whether rotation of the part would satisfy a size parameter associated with a clamp of a saw of the automated system and whether rotation of the part would satisfy a length parameter associated with a robot of the automated system.

273. The computing device of claim 272, wherein the circuitry is further configured to identify a rotation option that would cause the part to use less of the board than a set of other rotation options.

274. The computing device of claim 273, wherein the circuitry is further configured to:

determine whether an angle of the part can nest within an angle of a previous part to be produced from the board; and

determine whether the part has a better fit between being ordered after the previous part or being located at a beginning of the board.

275. The computing device of claim 274, wherein to determine whether an angle of the part can nest within the angle of the previous part comprises to determine whether both parts can cross over a centerline between the parts if angled cuts are utilized.

276. The computing device of claim 274, wherein to determine whether the part has a better fit comprises to:

determine whether a last piece of the board will be large enough to enable one or more clamps of the automated system to manipulate the board; and

determine whether the part will fit at the beginning of the board if the part is not large enough to be produced from the end of the board.

277. The computing device of claim 267, wherein the circuitry is further configured to:

output data indicative of a part location configuration for a possible ordering of parts; and

determine, as a function of an amount of the board utilized by the parts, whether a selected possible ordering of the parts is better than a previous best possible ordering of parts in a selected group of possible orderings.

278. The computing device of claim 253, wherein the circuitry is further configured to:

determine whether a board from which a set of parts of a wooden truss is to be produced has a remaining length that satisfies a predefined length;

determine, in response to a determination that the remaining length satisfies the predefined length, whether one or more standard parts having a lengths defined in a set of standard length configuration data in the lumber selection input data can be produced from the remaining length of the board; and

add, in response to a determination that one or more standard parts can be produced from the remaining length of the board, the one or more standard parts to a set of standard parts to be produced from the board.

279. A method comprising:

obtaining, by a computing device, lumber selection input data indicative of one or more parameters to be utilized in the selection of lumber for the production of one or more wooden trusses by an automated system; and

selecting, by the computing device and as a function of the one or more parameters and characteristics of lumber available in a lumber inventory of the automated system, a set of lumber pieces from the lumber inventory to satisfy an efficiency target in the automated production of the one or more wooden trusses.

280. The method of claim 279, wherein obtaining lumber selection input data comprises obtaining one or more of assembly recipe data indicative of operations to be performed by the automated system to produce the one or more wooden trusses, production data indicative of quantities of the wooden trusses to be produced in one or more batches, line data indicative of a status of one or more in-feed lines of the automated system, inventory data indicative of the characteristics of lumber available in the lumber inventory, saw parameter data indicative of a parameters of a saw of the automated system to be used to cut the lumber, or standard length configuration data indicative of one or more lengths defined for pieces of lumber to be used by the automated system.

281. The method of claim 279, further comprising determining, by the computing device and as a function of the lumber selection input data, a set of parts to be used in the production of wooden trusses in a batch.

282. The method of claim 281, further comprising:

determining, by the computing device, a set of boards for potential use in a batch of one or more wooden trusses to be produced;

determining, by the computing device, a set of potential parts for each board in the set of boards, including identifying, as a function of the parameters, a best board from which to produce the potential parts from lumber in the lumber inventory having a grade equal to a grade specified in connection with a design for a wooden truss to be produced and from lumber in the lumber inventory having a grade that is greater than the grade specified in connection with the design; and

removing, by the computing device, from a list of remaining parts to be produced, one or more of the potential parts to be produced from the determined best board.

283. The method of claim 282, further comprising determining, by the computing device, the best board to be used for each of the remaining parts to be produced.

284. The method of claim 281, wherein determining the set of parts to be used in the production of wooden trusses in the batch comprises:

determining, by the computing device, whether the batch is queued for a first line or a second line of the automated system;

determining, by the computing device and in response to a determination that the batch is queued for the first line or the second line, whether a wooden truss remains to be produced from an assembly recipe defined in the lumber selection input data;

determining, by the computing device and in response to a determination that a wooden truss remains to be produced from the assembly recipe, whether a part remains to be produced for the wooden truss; and

calculating, by the computing device and in response to a determination that a part remains to be produced for the wooden truss, a pickup location indicative of a length along the part where a robot of the automated system will pick up the part.

285. The method of claim 284, further comprising:

determining, by the computing device, whether the pickup location satisfies a predefined length threshold; and

rotating, by the computing device and in response to a determination that the pickup location does not satisfy the predefined length threshold, the part by 180 degrees.

286. The method of claim 284, further comprising:

determining, by the computing device and in response to a determination that the pickup location does not satisfy the predefined length threshold, whether the pickup location satisfies a percentage of part length threshold; and

rotating, by the computing device and in response to a determination that the pickup location satisfies the percentage of part length threshold, the part by 180 degrees.

287. The method of claim 286, wherein determining whether the pickup location satisfies the percentage of part length threshold comprises determining whether the pickup location is greater than 70% of the part length.

288. The method of claim 286, further comprising determining, by the computing device and in response to a determination that the pickup location does not satisfy the percentage of part length threshold, not to rotate the part.

289. The method of claim 282, wherein determining a set of boards for potential use in the batch comprises:

selecting, by the computing device, a board from the lumber inventory;

determining, by the computing device, whether a width of the selected board satisfies a width threshold for a part in the batch to be produced from the selected board;

determining, by the computing device and in response to a determination that the width of the selected board satisfies the width threshold, whether a length of the selected board satisfies a length threshold for the part; and

determining, by the computing device and in response to a determination that the length satisfies the length threshold, a set of one or more responsive operations as a function of a comparison between a grade of the selected board and a defined grade for the part.

290. The method of claim 289, further comprising excluding, by the computing device, the board from the set of boards for potential use in response to a determination that the width of

the selected board does not satisfy the width threshold or that the length of the selected board does not satisfy the length threshold.

291. The method of claim 289, wherein determining a set of one or more responsive operations as a function of a comparison between the grade of the selected board and the defined grade for the part comprises:

determining whether the grade of the selected board is equal to the defined grade for the part;

determining, in response to a determination that the grade of the selected board is not equal to the defined grade for the part, whether the grade of the selected board is greater than the defined grade for the part;

determining, in response to a determination that the grade of the selected board is greater than the defined grade for the part, whether the defined grade for the part is less than a grade of another board in the set of boards analyzed for inclusion in the set of boards for potential use and that has a grade that is greater than the defined grade for the part;

determining, in response to a determination that the grade of the selected board is less than the grade of the other board, whether the length of the selected board is less than a length of the other board; and

adding, in response to a determination that the length of the selected board is less than the length of the other board, the selected board to the set of boards for potential use.

292. The method of claim 282, wherein determining a set of potential parts for each board in the set of boards comprises:

selecting a board from the set of boards for potential use; and

determining, for each part in the batch, whether to add the part to the set of potential parts for the selected board, as a function of a comparison of a grade of the selected board to a defined grade for the part, a comparison of the width of the selected board to a width for the part, and a determination of whether the part will fit on the selected board.

293. The method of claim 282, wherein determining the best board comprises:

selecting a board from the set of boards for potential use in the batch;

determining, as a function of a number of parts to be produced from the selected board, a set of permutations of possible orderings for the parts;

defining a set of groups of parts from the set of permutations of possible orderings for the parts; and

determining the best board as a function of the defined set of groups of parts.

294. The method of claim 293, further comprising:

determining, by the computing device and for a part in a group in the defined set of groups of parts, whether the part is the first part in the possible ordering of parts associated with the group; and

selectively rotating, by the computing device, the board in response to a determination that the part is the first part.

295. The method of claim 294, wherein selectively rotating the board comprises rotating the board to orient a straight edge of the part with a straight edge of the board.

296. The method of claim 294, further comprising determining, by the computing device and in response to a determination that the part is not the first part, whether the selected part can be rotated.

297. The method of claim 296, wherein determining whether the part can be rotated comprises determining whether rotation of the part would satisfy one or more size parameters.

298. The method of claim 297, wherein determining whether rotation of the part would satisfy one or more size parameters comprises determining whether rotation of the part would satisfy a size parameter associated with a clamp of a saw of the automated system and whether rotation of the part would satisfy a length parameter associated with a robot of the automated system.

299. The method of claim 298, further comprising identifying, by the computing device, a rotation option that would cause the part to use less of the board than a set of other rotation options.

300. The method of claim 299, further comprising:

determining, by the computing device, whether an angle of the part can nest within an angle of a previous part to be produced from the board; and

determining, by the computing device, whether the part has a better fit between being ordered after the previous part or being located at a beginning of the board.

301. The method of claim 300, wherein determining whether an angle of the part can nest within the angle of the previous part comprises determining whether both parts can cross over a centerline between the parts if angled cuts are utilized.

302. The method of claim 300, wherein determining whether the part has a better fit comprises:

determining whether a last piece of the board will be large enough to enable one or more clamps of the automated system to manipulate the board; and

determining whether the part will fit at the beginning of the board if the part is not large enough to be produced from the end of the board.

303. The method of claim 293, further comprising:

outputting, by the computing device, data indicative of a part location configuration for a possible ordering of parts; and

determining, by the computing device and as a function of an amount of the board utilized by the parts, whether a selected possible ordering of the parts is better than a previous best possible ordering of parts in a selected group of possible orderings.

304. The method of claim 279, further comprising:

determining, by the computing device, whether a board from which a set of parts of a wooden truss is to be produced has a remaining length that satisfies a predefined length;

determining, by the computing device and in response to a determination that the remaining length satisfies the predefined length, whether one or more standard parts having a lengths defined in a set of standard length configuration data in the lumber selection input data can be produced from the remaining length of the board; and

adding, by the computing device and in response to a determination that one or more standard parts can be produced from the remaining length of the board, the one or more standard parts to a set of standard parts to be produced from the board.

305. One or more machine-readable storage media comprising a plurality of instructions stored thereon that, in response to being executed, cause a computing device to perform the method of any of claims 279-304.

306. An automated system comprising:

a fiducial printer;

an in-feed line to receive a board;

a carrier within the in-feed line to selectively move the board; and

circuitry configured to:

move, with the carrier, a board received in the in-feed line to a defined fiducial printing position; and

print, with the fiducial printer and in response to a determination that the board has been moved to the defined fiducial printing position, a fiducial on a side of the board, wherein the fiducial represents information to be detected by a machine vision system to perform one or more operations associated with production, by the automated system, of a wooden structure using the board.

307. The automated system of claim 306, wherein the circuitry is further configured to detect, utilizing one or more sensors, that the board has entered the in-feed line, and wherein to move the board comprises to move, with the carrier, the board in response to the determination that the board has entered the in-feed line.

308. The automated system of claim 306, wherein to move, with the carrier, the board in the in-feed line comprises to move the board along an in-feed axis to the defined fiducial printing position.

309. The automated system of claim 308, wherein the circuitry is further configured to detect, with a corresponding sensor, an end of the board in the in-feed line.

310. The automated system of claim 309, wherein to detect the end of the board comprises to detect the end of the board with a photoelectric sensor.

311. The automated system of claim 310, wherein the circuitry is further configured to advance, in response to detection of the end of the board, the board along an in-feed axis a predefined length to move the board to the defined fiducial printing position.

312. The automated system of claim 306, wherein to print a fiducial on a side of the board comprises to print the fiducial on a major side of the board or a minor side of the board.

313. The automated system of claim 312, wherein to print a fiducial comprises to print one or more predefined symbols.

314. The automated system of claim 313, wherein to print one or more predefined symbols comprises to print predefined symbols that are mirrored in opposite directions.

315. The automated system of claim 313, wherein to print one or more predefined symbols comprises to print one or more predefined symbols that are indicative of information about the board.

316. The automated system of claim 315, wherein to print one or more predefined symbols that are indicative of information about the board comprises to print one or more predefined symbols indicative of an identifier of the board.

317. The automated system of claim 315, wherein to print one or more symbols that are indicative of information about the board comprises to print one or more predefined symbols indicative of an index value in a sequence of boards to be used in the production of the wooden structure by the automated system.

318. The automated system of claim 315, wherein to print one or more symbols that are indicative of information about the board comprises to print one or more predefined symbols indicative of an identifier of a leading end or trailing end of the board.

319. The automated system of claim 315, wherein to print one or more symbols that are indicative of information about the board comprises to print one or more symbols indicative of a grade of the board.

320. The automated system of claim 306, wherein to print the fiducial comprises to print the fiducial with multiple print heads arranged along a width of the board.

321. The automated system of claim 306, wherein the defined fiducial printing position is a first defined fiducial printing position that is associated with a first end of the board and the circuity is further configured to:

move the board to a second defined fiducial printing position that is associated with a second end of the board; and

print, in response to a determination that the board is in the second defined fiducial printing position, one or more second fiducials on the board.

322. The automated system of claim 321, wherein to move the board to the second defined printing position comprises to advance the board along an in-feed axis until the automated system detects the second end of the board with a sensor that detected the presence of the first end of the board.

323. The automated system of claim 322, wherein to advance the board along the in-feed axis until the automated system detects the second end of the board comprises to advance the board along the in-feed axis until the sensor detects that the board is no longer present.

324. The automated system of claim 322, wherein the circuitry is further configured to reverse, in response to detection of the second end of the board, a direction of movement of the board along the in-feed axis.

325. The automated system of claim 324, wherein to reverse the direction of movement comprises to advance the board in the opposite direction by a predefined length.

326. The automated system of claim 306, wherein the circuitry is further configured to utilize the machine vision system to identify the fiducial on the board in the production of the wooden structure.

327. The automated system of claim 326, wherein to utilize the machine vision system comprises to acquire an image of the fiducial with a camera of a component of the automated system.

328. The automated system of claim 327, wherein the circuitry is further configured to determine a position of the board relative to the component based on the acquired image of the fiducial.

329. The automated system of claim 327, wherein the camera is associated with a tool of a robot of the automated system and the circuitry is further configured to position the camera over the fiducial and determine the location of the tool relative to the board based on a predefined position of the fiducial on the board and an offset of the location of the camera relative to a center of the tool.

330. The automated system of claim 326, wherein to utilize the machine vision system comprises to determine information indicated by the fiducial.

331. The automated system of claim 330, wherein to determine information indicated by the fiducial comprises to determine one more of an identifier of the board, an index value in a sequence associated with the board, a grade of the board, or which of multiple ends of the board is imaged.

332. A method comprising:

moving, with a carrier of an automated system, a board received in an in-feed line of the automated system to a defined fiducial printing position; and

printing, with a fiducial printer of the automated system and in response to a determination that the board has been moved to the defined fiducial printing position, a fiducial on a side of the board, wherein the fiducial represents information to be detected by a machine vision

system to perform one or more operations associated with production, by the automated system, of a wooden structure using the board.

333. The method of claim 332, further comprising detecting, utilizing one or more sensors, that the board has entered the in-feed line, and wherein moving the board comprises moving, with the carrier, the board in response to the determination that the board has entered the in-feed line.

334. The method of claim 332, wherein moving, with the carrier, the board in the in-feed line comprises moving the board along an in-feed axis to the defined fiducial printing position.

335. The method of claim 334, further comprising detecting, with a corresponding sensor, an end of the board in the in-feed line.

336. The method of claim 335, wherein detecting the end of the board comprises detecting the end of the board with a photoelectric sensor.

337. The method of claim 336, further comprising advancing, in response to detection of the end of the board, the board along an in-feed axis a predefined length to move the board to the defined fiducial printing position.

338. The method of claim 332, wherein printing a fiducial on a side of the board comprises printing the fiducial on a major side of the board or a minor side of the board.

339. The method of claim 338, wherein printing a fiducial comprises printing one or more predefined symbols.

340. The method of claim 339, wherein printing one or more predefined symbols comprises printing predefined symbols that are mirrored in opposite directions.

341. The method of claim 339, wherein printing one or more predefined symbols comprises printing one or more predefined symbols that are indicative of information about the board.

342. The method of claim 341, wherein printing one or more predefined symbols that are indicative of information about the board comprises printing one or more predefined symbols indicative of an identifier of the board.

343. The method of claim 341, wherein printing one or more symbols that are indicative of information about the board comprises printing one or more predefined symbols indicative of an index value in a sequence of boards to be used in the production of the wooden structure by the automated system.

344. The method of claim 341, wherein printing one or more symbols that are indicative of information about the board comprises printing one or more predefined symbols indicative of an identifier of a leading end or trailing end of the board.

345. The method of claim 341, wherein printing one or more symbols that are indicative of information about the board comprises printing one or more symbols indicative of a grade of the board.

346. The method of claim 332, wherein printing the fiducial comprises printing the fiducial with multiple print heads arranged along a width of the board.

347. The method of claim 332, wherein the defined fiducial printing position is a first defined fiducial printing position that is associated with a first end of the board, the method further comprising:

moving the board to a second defined fiducial printing position that is associated with a second end of the board; and

printing, in response to a determination that the board is in the second defined fiducial printing position, one or more second fiducials on the board.

348. The method of claim 347, wherein moving the board to the second defined printing position comprises advancing the board along an in-feed axis until the automated system detects the second end of the board with a sensor that detected the presence of the first end of the board.

349. The method of claim 348, wherein advancing the board along the in-feed axis until the automated system detects the second end of the board comprises advancing the board along the in-feed axis until the sensor detects that the board is no longer present.

350. The method of claim 348, further comprising reversing, by the carrier and in response to detection of the second end of the board, a direction of movement of the board along the in-feed axis.

351. The method of claim 350, wherein reversing the direction of movement comprises advancing the board in the opposite direction by a predefined length.

352. The method of claim 332, further comprising utilizing the machine vision system to identify the fiducial on the board in the production of the wooden structure.

353. The method of claim 352, wherein utilizing the machine vision system comprises acquiring an image of the fiducial with a camera of a component of the automated system.

354. The method of claim 353, further comprising determining, by the automated system, a position of the board relative to the component based on the acquired image of the fiducial.

355. The method of claim 353, wherein the camera is associated with a tool of a robot of the automated system, the method further comprising positioning, by the automated system, the camera over the fiducial and determining the location of the tool relative to the board based on a predefined position of the fiducial on the board and an offset of the location of the camera relative to a center of the tool.

356. The method of claim 352, wherein utilizing the machine vision system comprises determining information indicated by the fiducial.

357. The method of claim 356, wherein determining information indicated by the fiducial comprises determining one more of an identifier of the board, an index value in a sequence associated with the board, a grade of the board, or which of multiple ends of the board is imaged.

358. One or more machine-readable storage media comprising a plurality of instructions stored thereon that, in response to being executed, cause a computing device to perform the method of any of claims 332-357.

359. A computing device comprising:

circuitry configured to:

obtain data indicative of a set of one or more jobs for the production of one or more wooden structures by an automated system; and

utilize a microservices architecture to produce the one or more wooden structures associated with the one or more jobs.

360. The computing device of claim 359, wherein to utilize a microservices architecture further comprises to interrupt execution of one microservice without interrupting execution of other microservices in the microservices architecture.

361. The computing device of claim 359, wherein to utilize the microservices architecture comprises to communicate data between microservices using a network communication protocol.

362. The computing device of claim 361, wherein to communicate data between microservices using a network communication protocol comprises to communicate using hypertext transfer protocol or hypertext transfer protocol secure.

363. The computing device of claim 359, wherein to utilize a microservices architecture comprises to utilize an assembly recipe generator microservice to produce a recipe indicative of a

sequence of operations of components of the automated system to produce the one or more wooden structures.

364. The computing device of claim 363, wherein to utilize a microservices architecture further comprises to utilize a lumber optimizer microservice to select lumber from a lumber inventory to satisfy one or more target parameters in the automated production of the one or more wooden structures by the automated system.

365. The computing device of claim 364, wherein to utilize a microservices architecture further comprises to utilize a set of microservices to control machines of the automated system to produce the one or more wooden structures.

366. The computing device of claim 365, wherein to utilize a set of microservices to control machines of the automated system comprises to utilize a set of microservices to read and write machine register values using network communication protocols.

367. The computing device of claim 365, wherein to utilize a set of microservices to control machines of the automated system comprises to utilize an assembler microservice to control one or more assembler machines.

368. The computing device of claim 367, wherein to utilize an assembler microservice to control one or more assembler machines comprises to utilize an assembler microservice to communicate with one or more assembler machine controllers.

369. The computing device of claim 368, wherein to utilize an assembler microservice to communicate with one or more assembler machine controllers comprises to utilize an assembler microservice to communicate with one or more assembly robot controllers.

370. The computing device of claim 365, wherein to utilize a set of microservices to control machines of the automated system comprises to utilize a plate microservice to control a plate machine for manipulating nailing plates and a saw microservice to control a saw machine of

the automated system by reading and writing machine register values of a controller of the plate machine and a controller of the saw machine using a network communication protocol.

371. The computing device of claim 359, wherein to utilize a microservices architecture further comprises to utilize a shell microservice to provide a shell for one or more user interfaces.

372. The computing device of claim 359, wherein to utilize a microservices architecture further comprises to utilize a log aggregator microservice to manage logs produced by the automated system.

373. The computing device of claim 359, wherein to utilize a microservices architecture further comprises to utilize a machine diagnostics microservice to analyze an operational status of one or more machines of the automated system.

374. The computing device of claim 359, wherein to utilize a microservices architecture comprises to utilize microservices that are based on one or more shared packages of executable instructions or data.

375. A method comprising:

obtaining, by a computing device, data indicative of a set of one or more jobs for the production of one or more wooden structures by an automated system; and

utilizing, by the computing device, a microservices architecture to produce the one or more wooden structures associated with the one or more jobs.

376. The method of claim 375, wherein utilizing a microservices architecture further comprises interrupting execution of one microservice without interrupting execution of other microservices in the microservices architecture.

377. The method of claim 375, wherein utilizing the microservices architecture comprises communicating data between microservices using a network communication protocol.

378. The method of claim 377, wherein communicating data between microservices using a network communication protocol comprises communicating using hypertext transfer protocol or hypertext transfer protocol secure.

379. The method of claim 375, wherein utilizing a microservices architecture comprises utilizing an assembly recipe generator microservice to produce a recipe indicative of a sequence of operations of components of the automated system to produce the one or more wooden structures.

380. The method of claim 379, wherein utilizing a microservices architecture further comprises utilizing a lumber optimizer microservice to select lumber from a lumber inventory to satisfy one or more target parameters in the automated production of the one or more wooden structures by the automated system.

381. The method of claim 380, wherein utilizing a microservices architecture further comprises utilizing a set of microservices to control machines of the automated system to produce the one or more wooden structures.

382. The method of claim 381, wherein utilizing a set of microservices to control machines of the automated system comprises utilizing a set of microservices to read and write machine register values using network communication protocols.

383. The method of claim 381, wherein utilizing a set of microservices to control machines of the automated system comprises utilizing an assembler microservice to control one or more assembler machines.

384. The method of claim 383, wherein utilizing an assembler microservice to control one or more assembler machines comprises utilizing an assembler microservice to communicate with one or more assembler machine controllers.

385. The method of claim 384, wherein utilizing an assembler microservice to communicate with one or more assembler machine controllers comprises utilizing an assembler microservice to communicate with one or more assembly robot controllers.

386. The method of claim 381, wherein utilizing a set of microservices to control machines of the automated system comprises utilizing a plate microservice to control a plate machine for manipulating nailing plates and a saw microservice to control a saw machine of the automated system by reading and writing machine register values of a controller of the plate machine and a controller of the saw machine using a network communication protocol.

387. The method of claim 375, wherein utilizing a microservices architecture further comprises utilizing a shell microservice to provide a shell for one or more user interfaces.

388. The method of claim 375, wherein utilizing a microservices architecture further comprises utilizing a log aggregator microservice to manage logs produced by the automated system.

389. The method of claim 375, wherein utilizing a microservices architecture further comprises utilizing a machine diagnostics microservice to analyze an operational status of one or more machines of the automated system.

390. The method of claim 375, wherein utilizing a microservices architecture comprises utilizing microservices that are based on one or more shared packages of executable instructions or data.

391. One or more machine-readable storage media comprising a plurality of instructions stored thereon that, in response to being executed, cause a computing device to perform the method of any of claims 375-390.

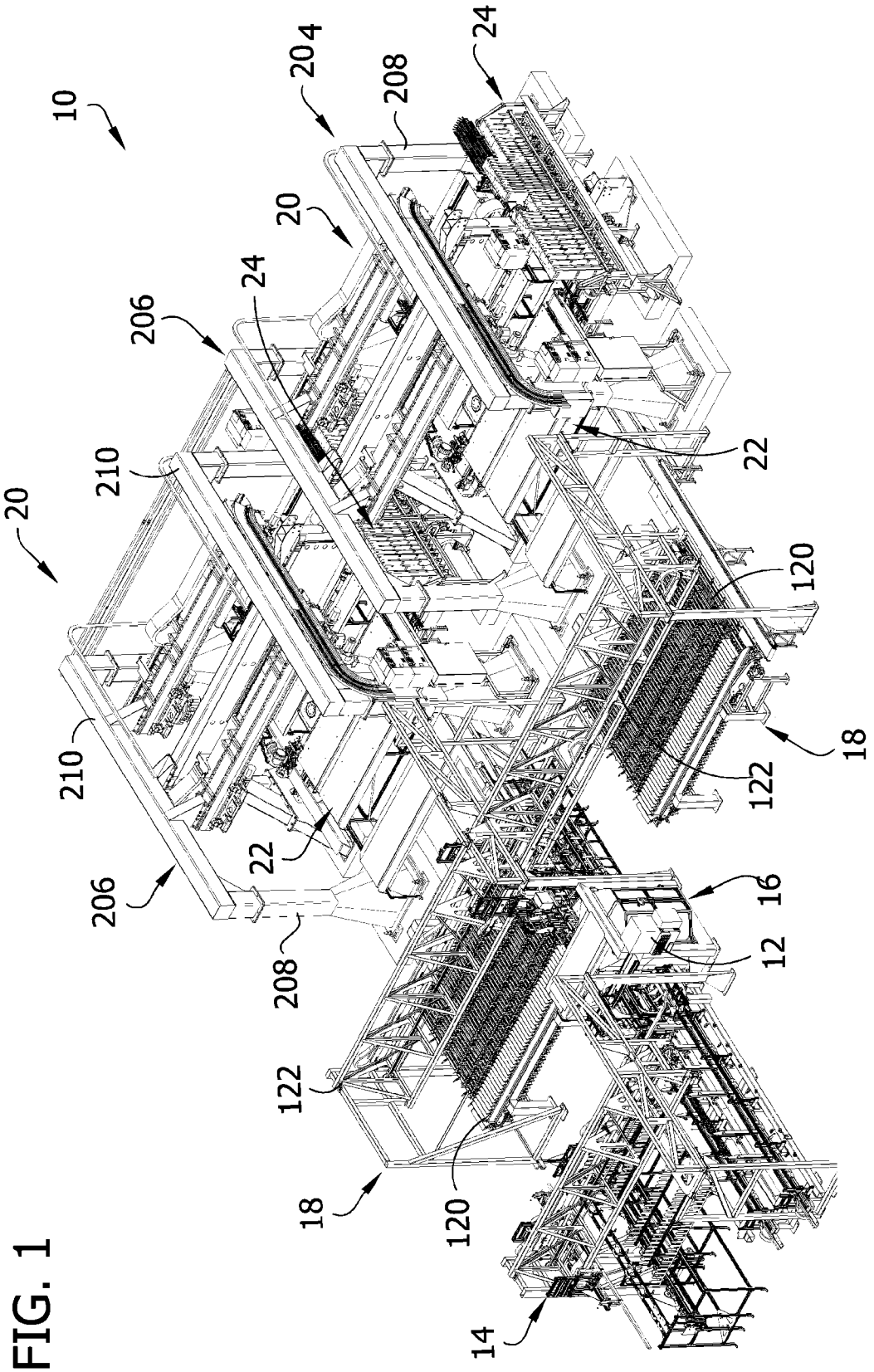

10
204
208
24
20
206
24
210
22
20
120
210
18
122
22
206
16
208
12
122
18
120
14

FIG. 1

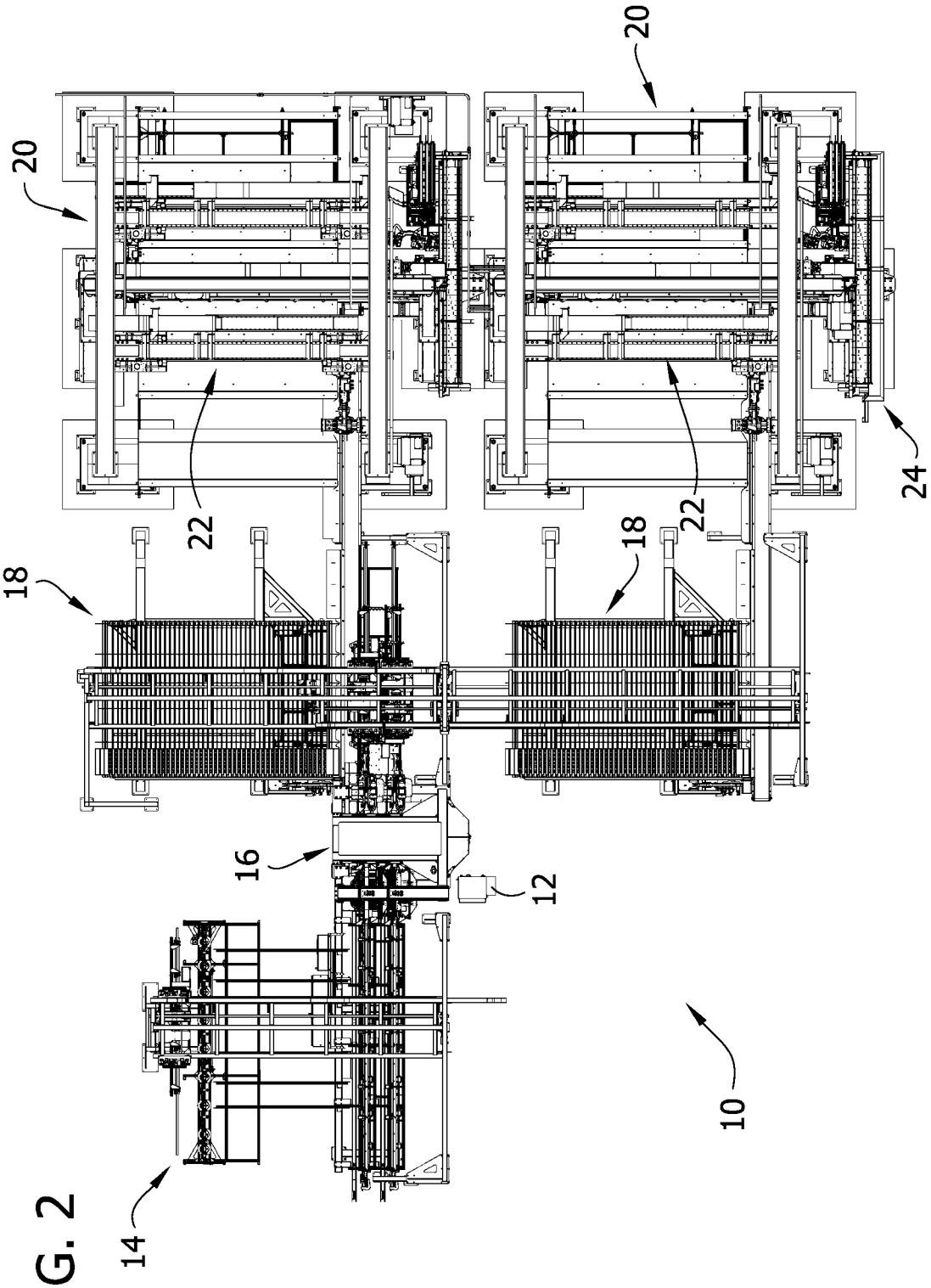



FIG. 2

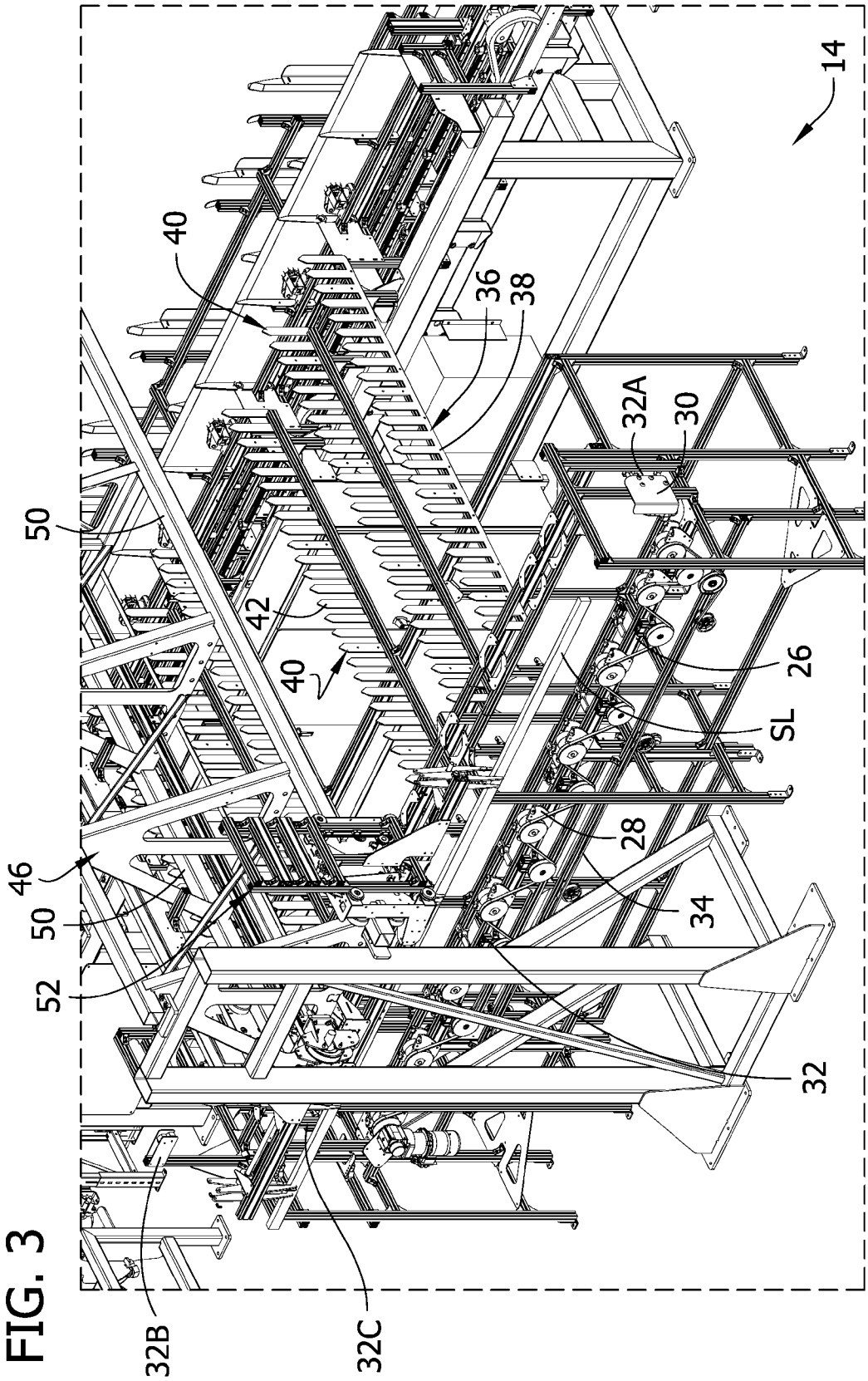



FIG. 3

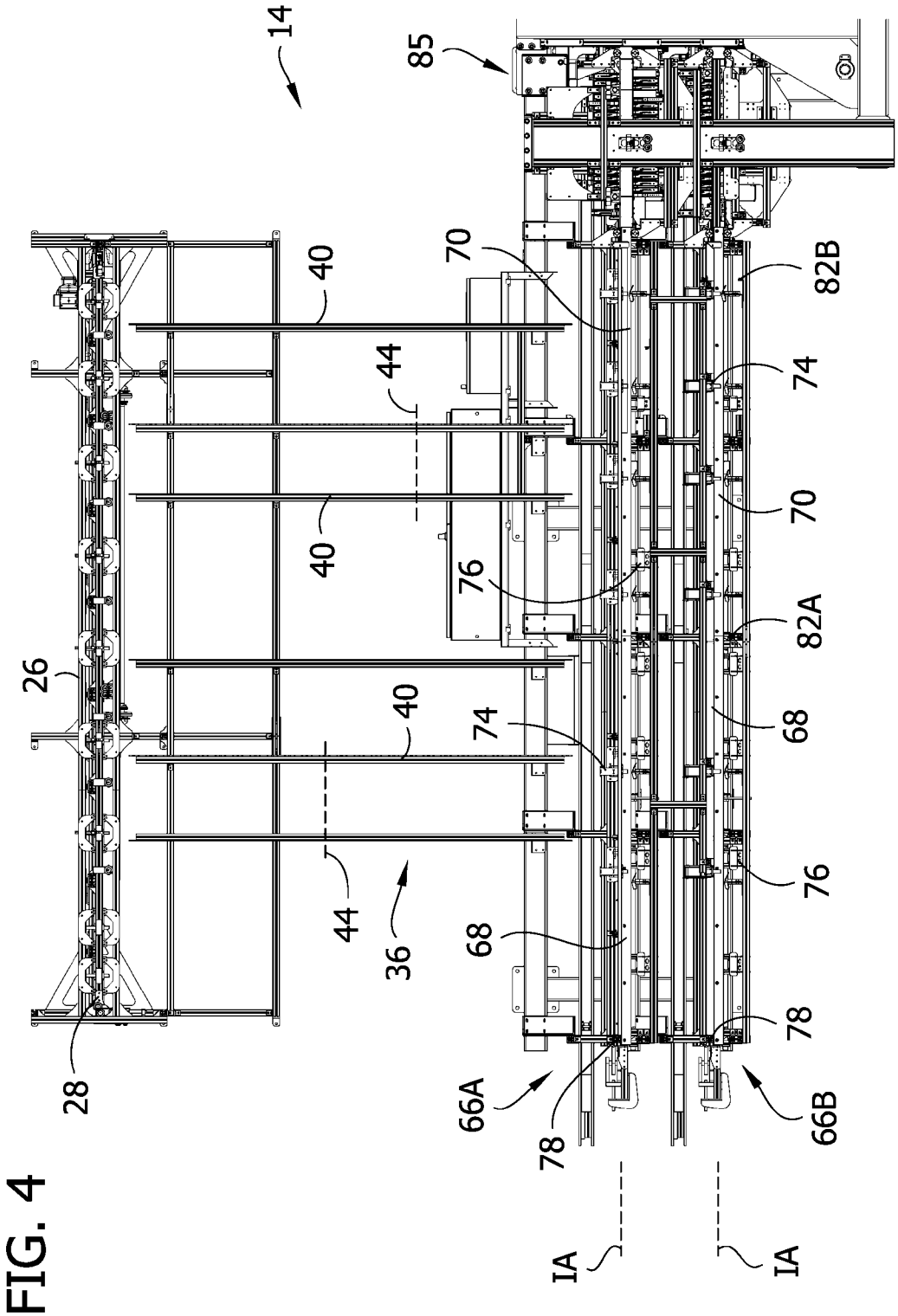

14
85
70
82B
40
44
74
40
70
76
82A
26
40
74
68
44
36
68
76
28
66A
78
78
66B
IA
IA

FIG. 4

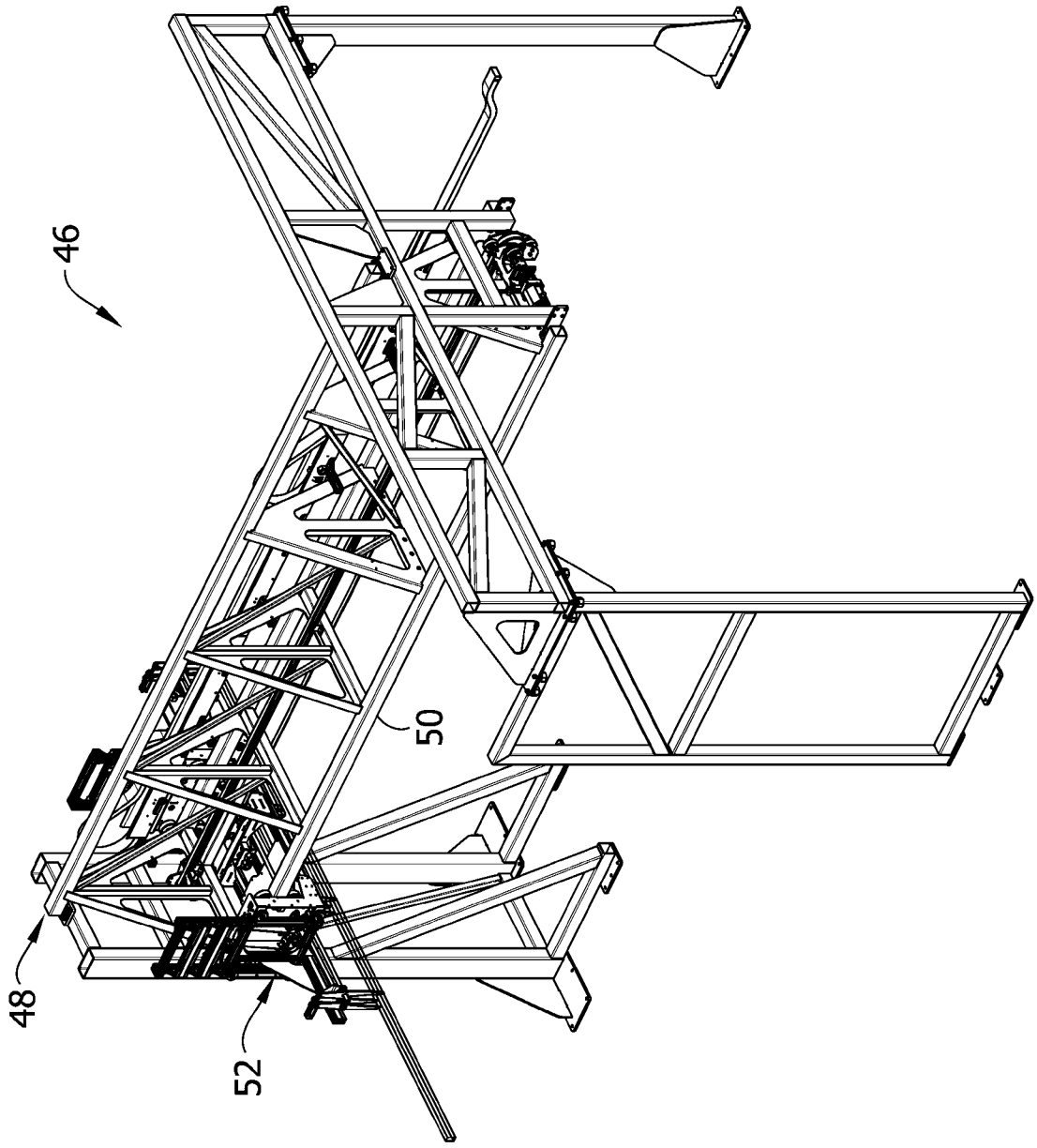



FIG. 5

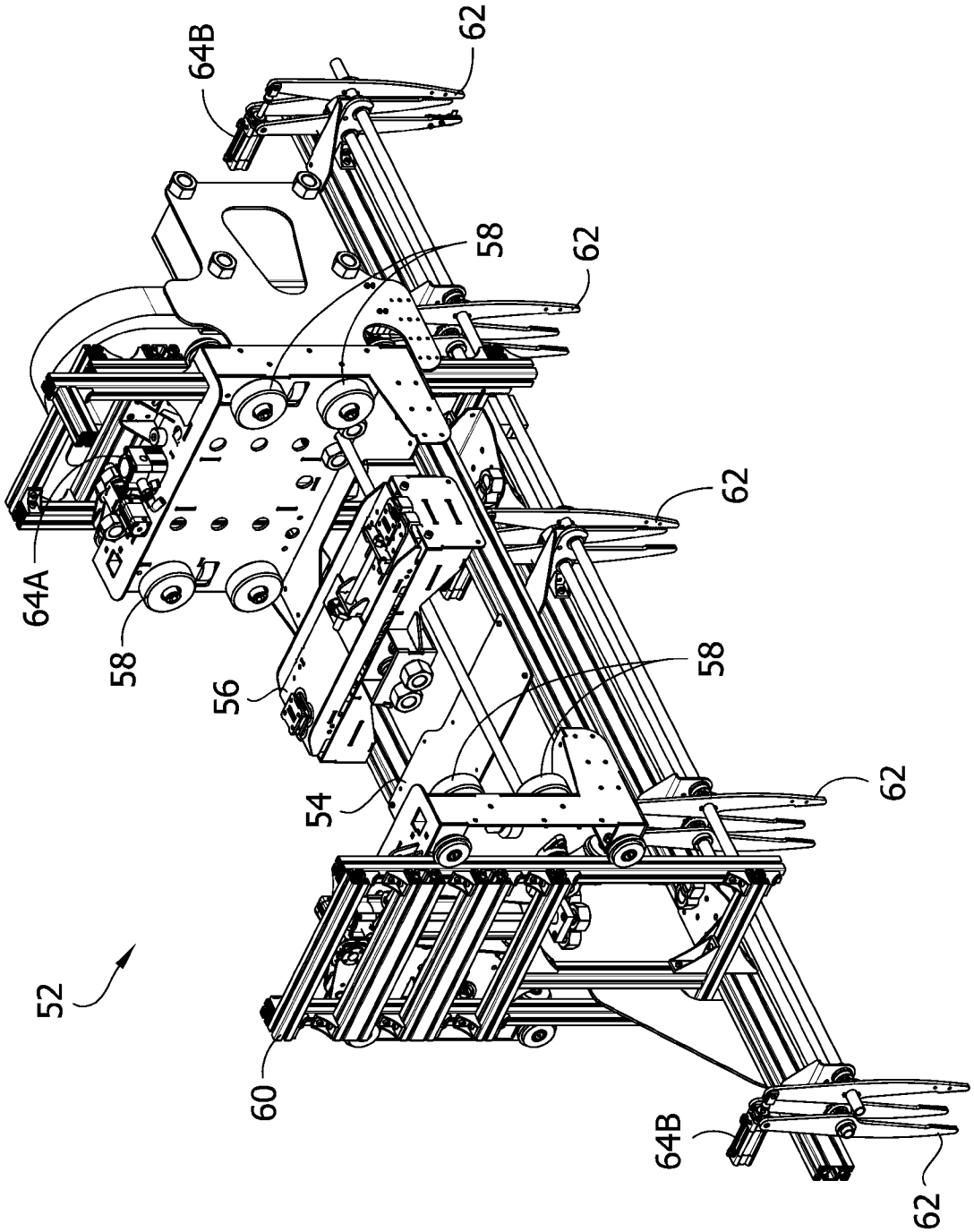
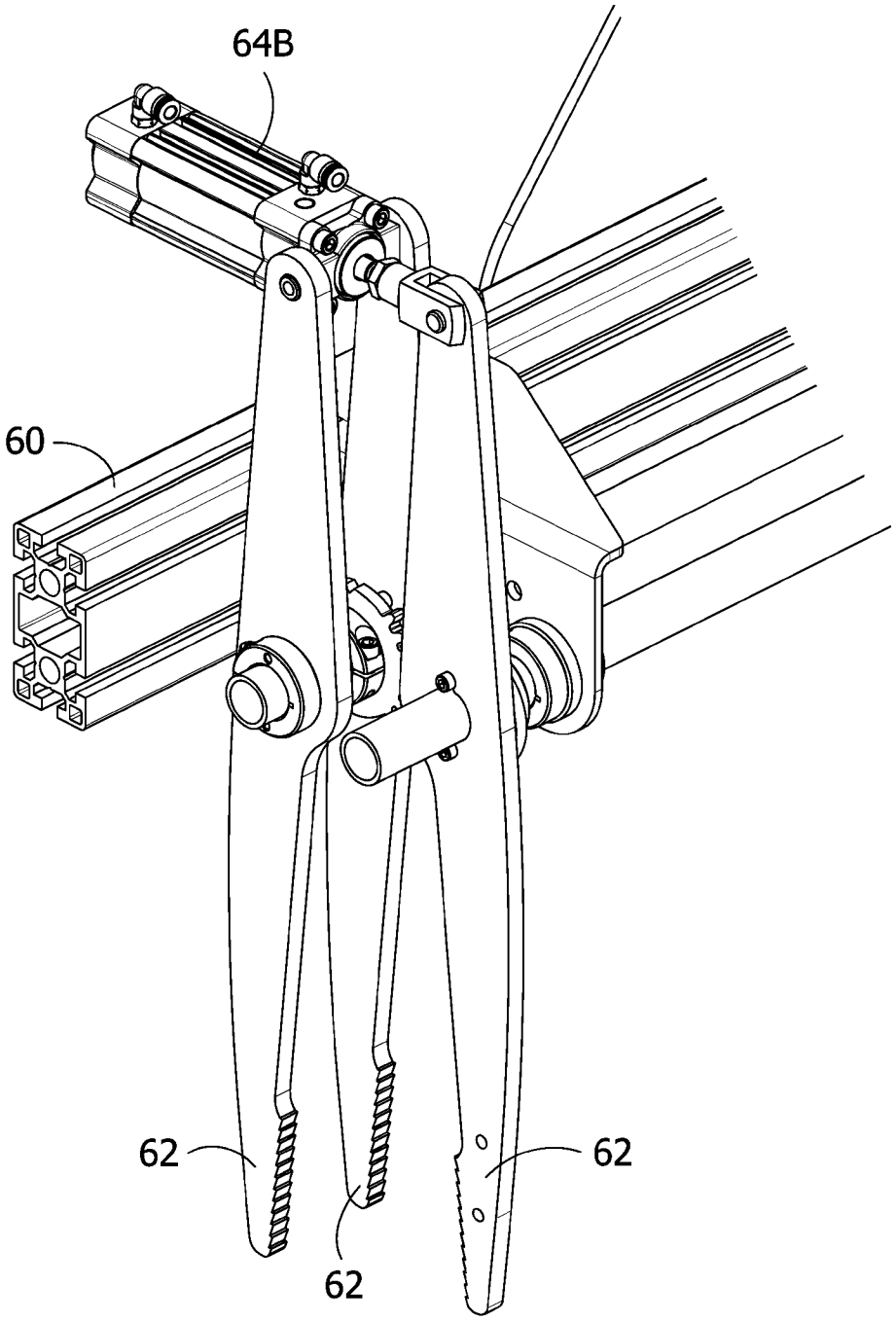



FIG. 6

FIG. 7



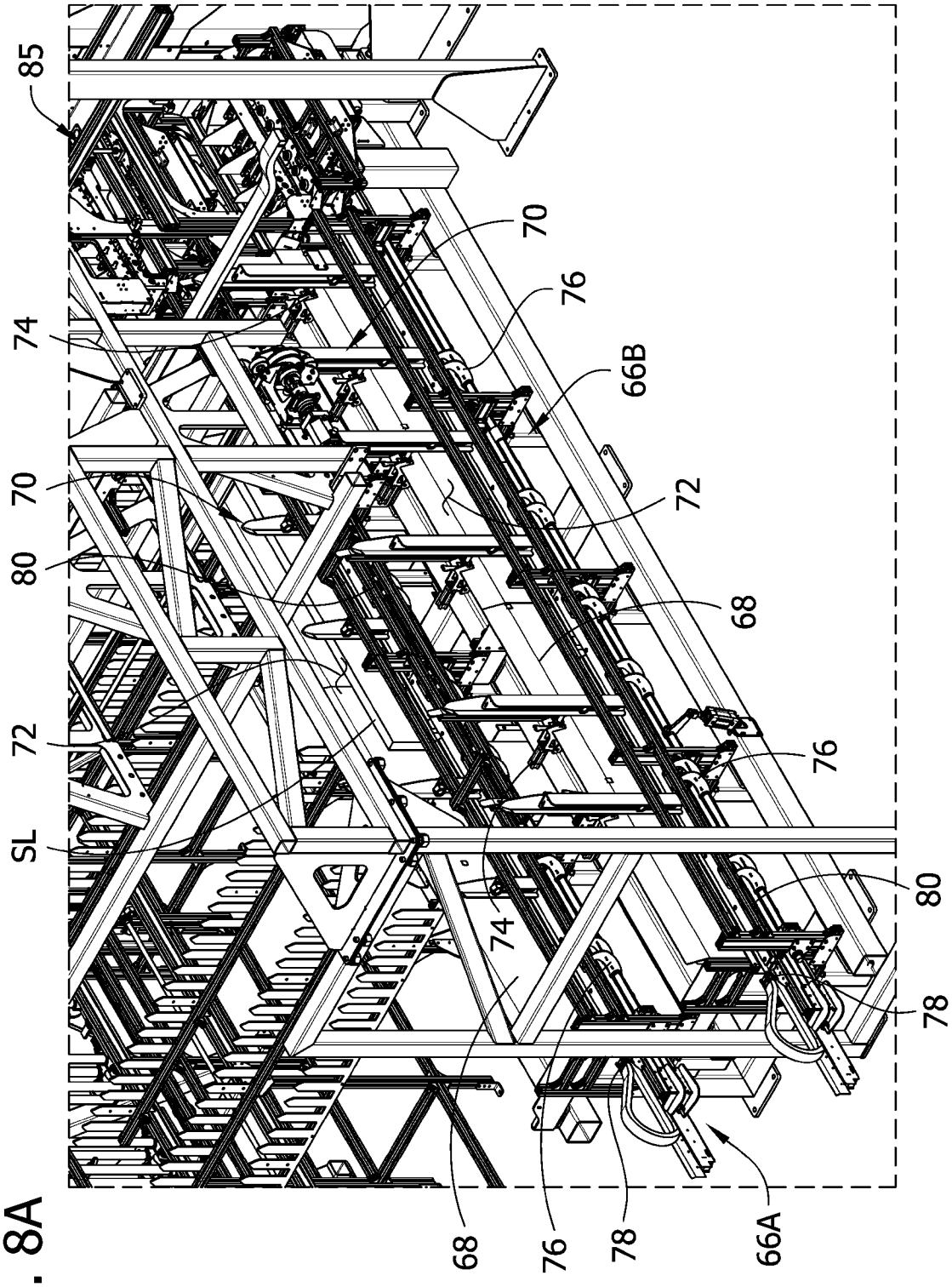



FIG. 8A

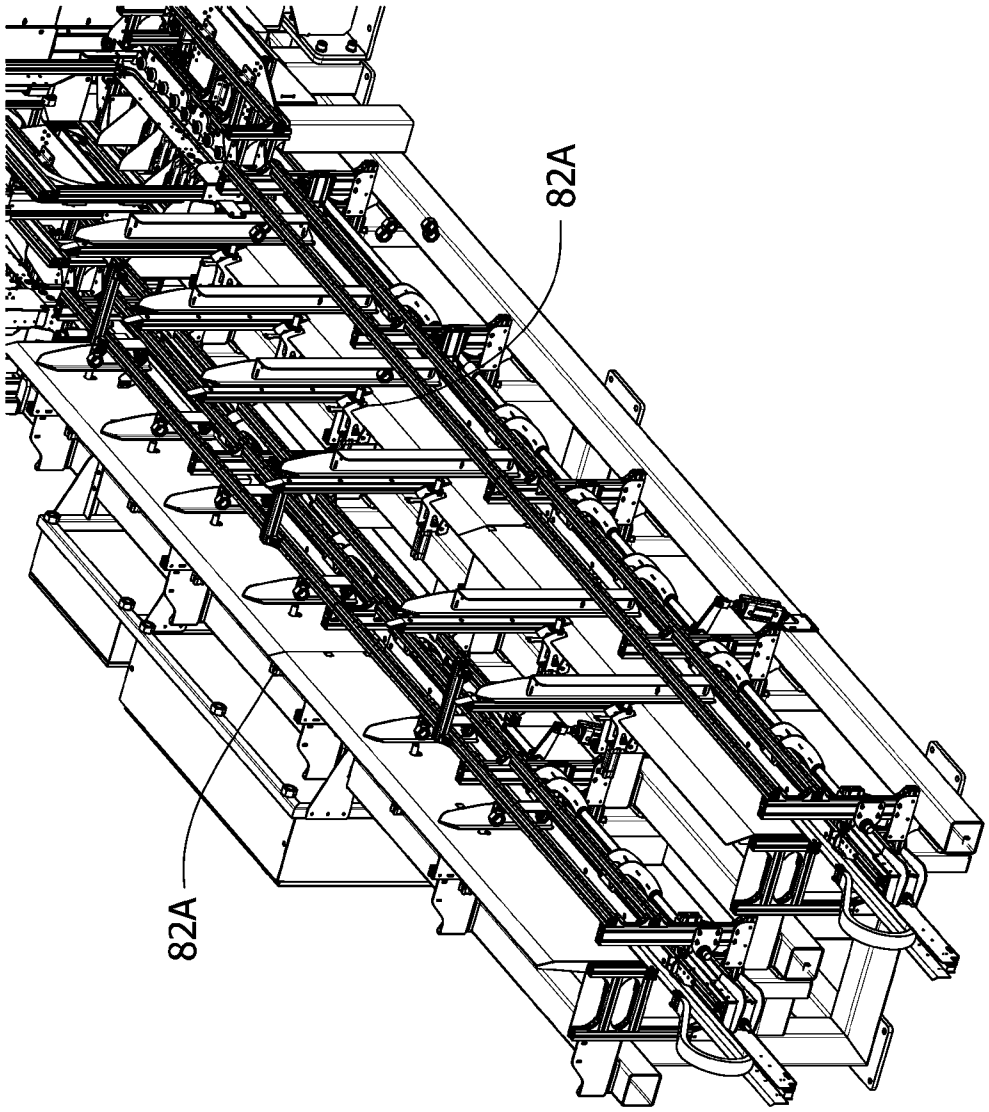



FIG. 8B

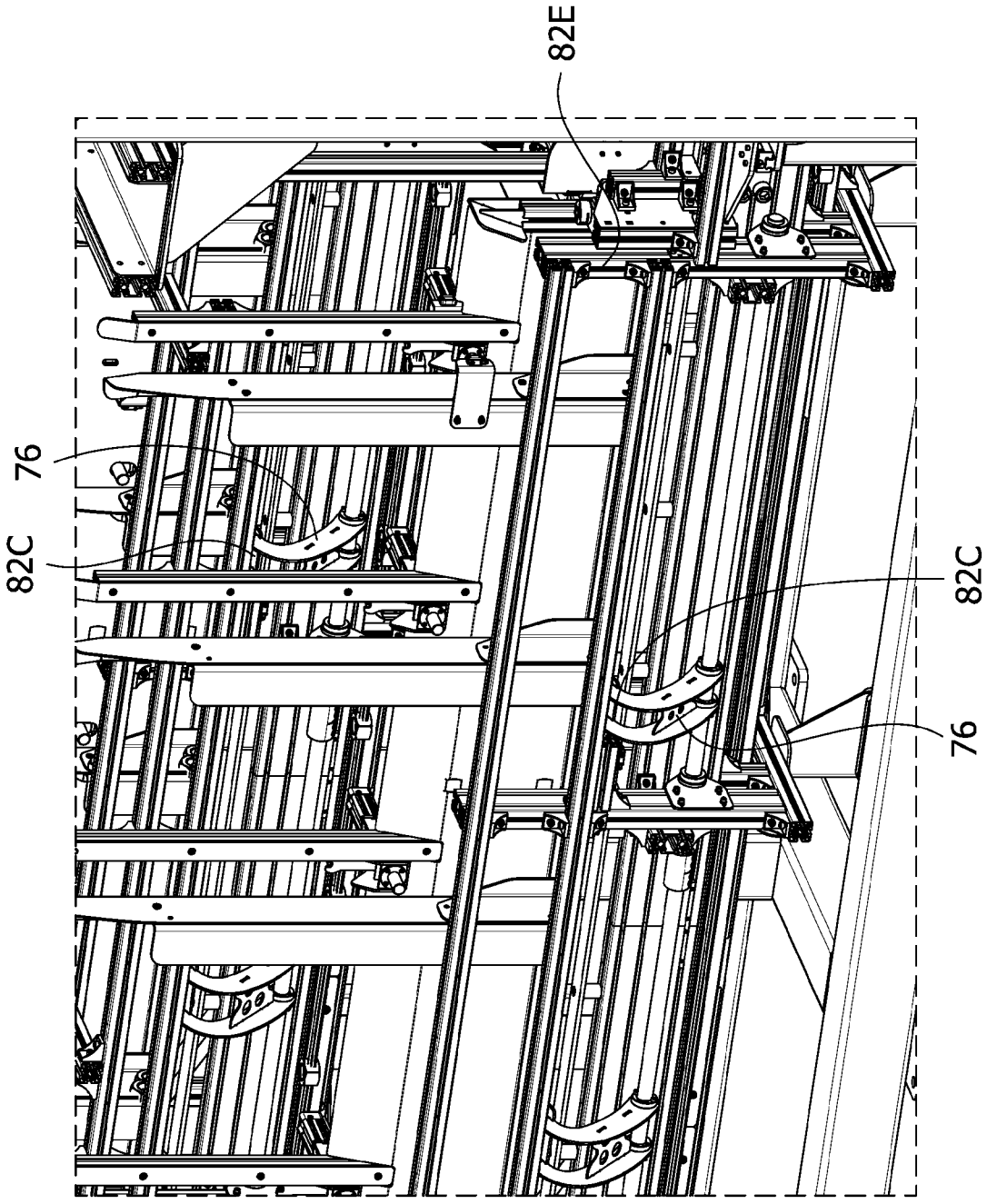



FIG. 8C

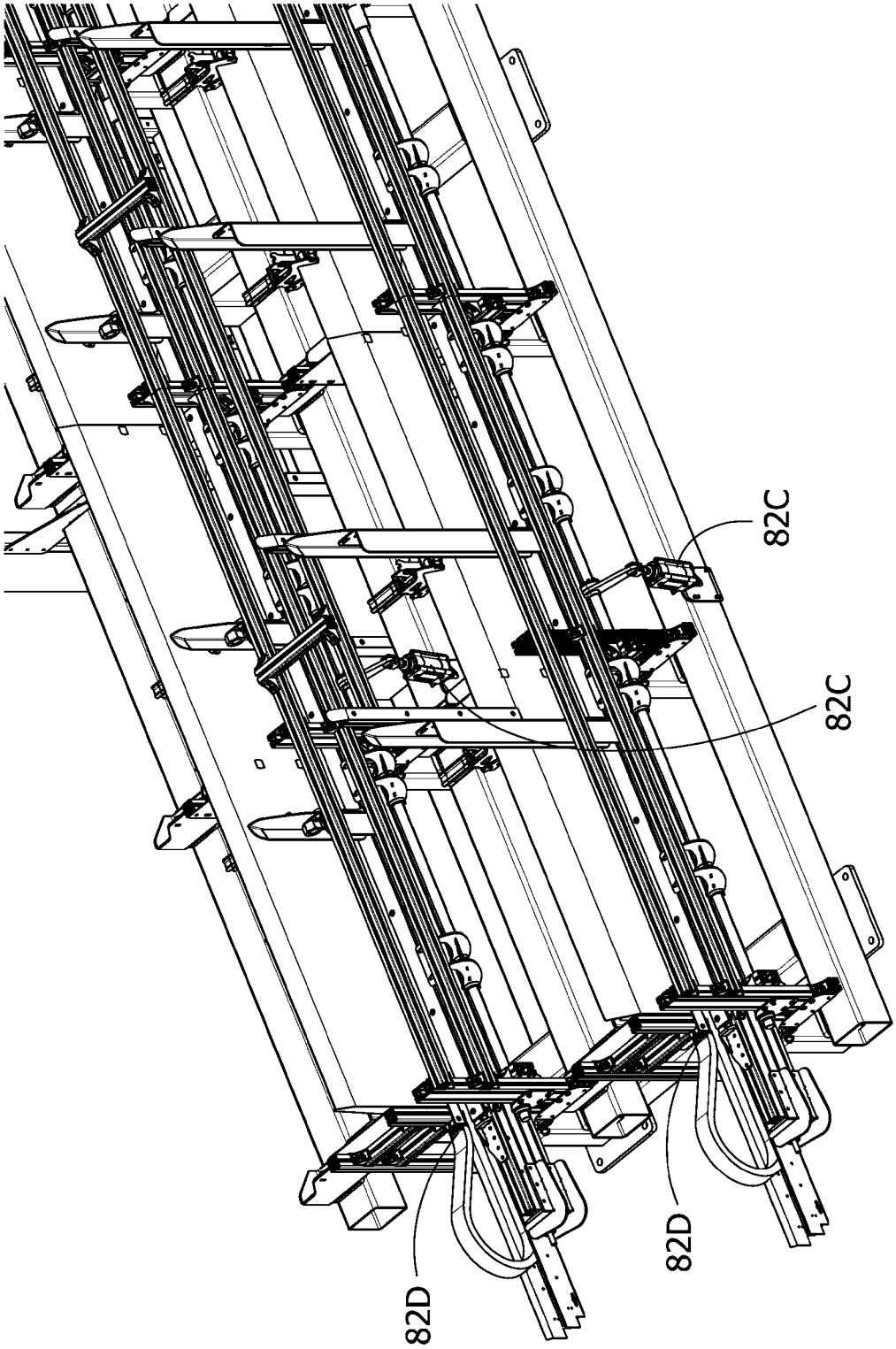
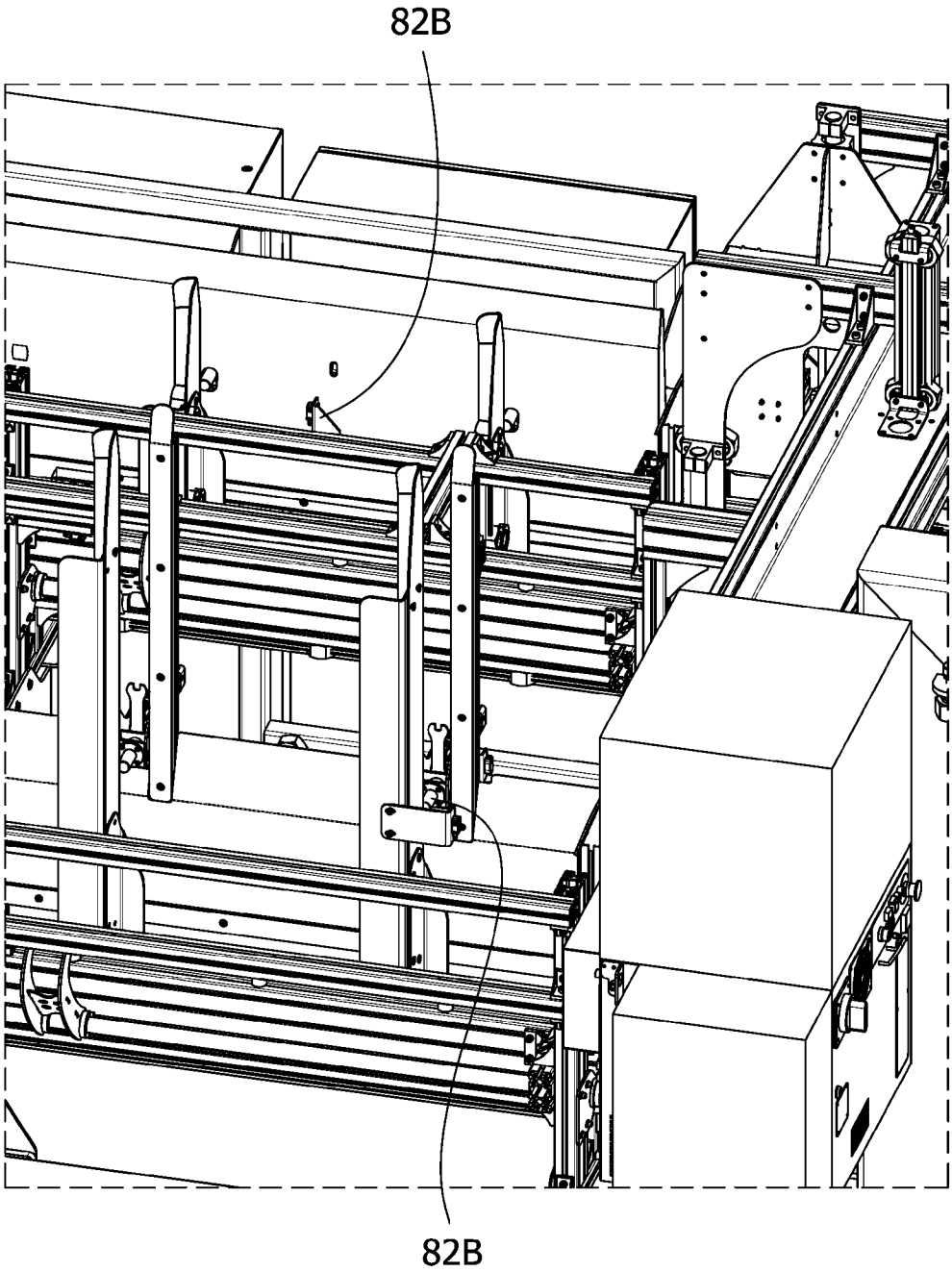



FIG. 8D

FIG. 8E



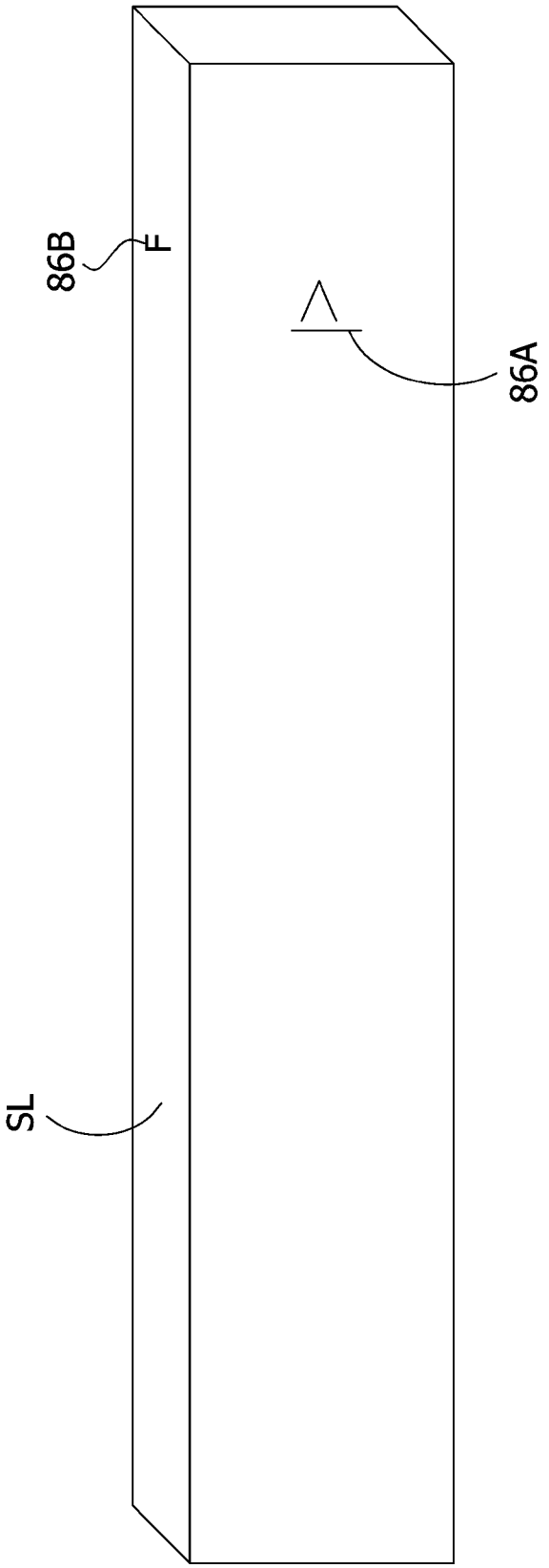
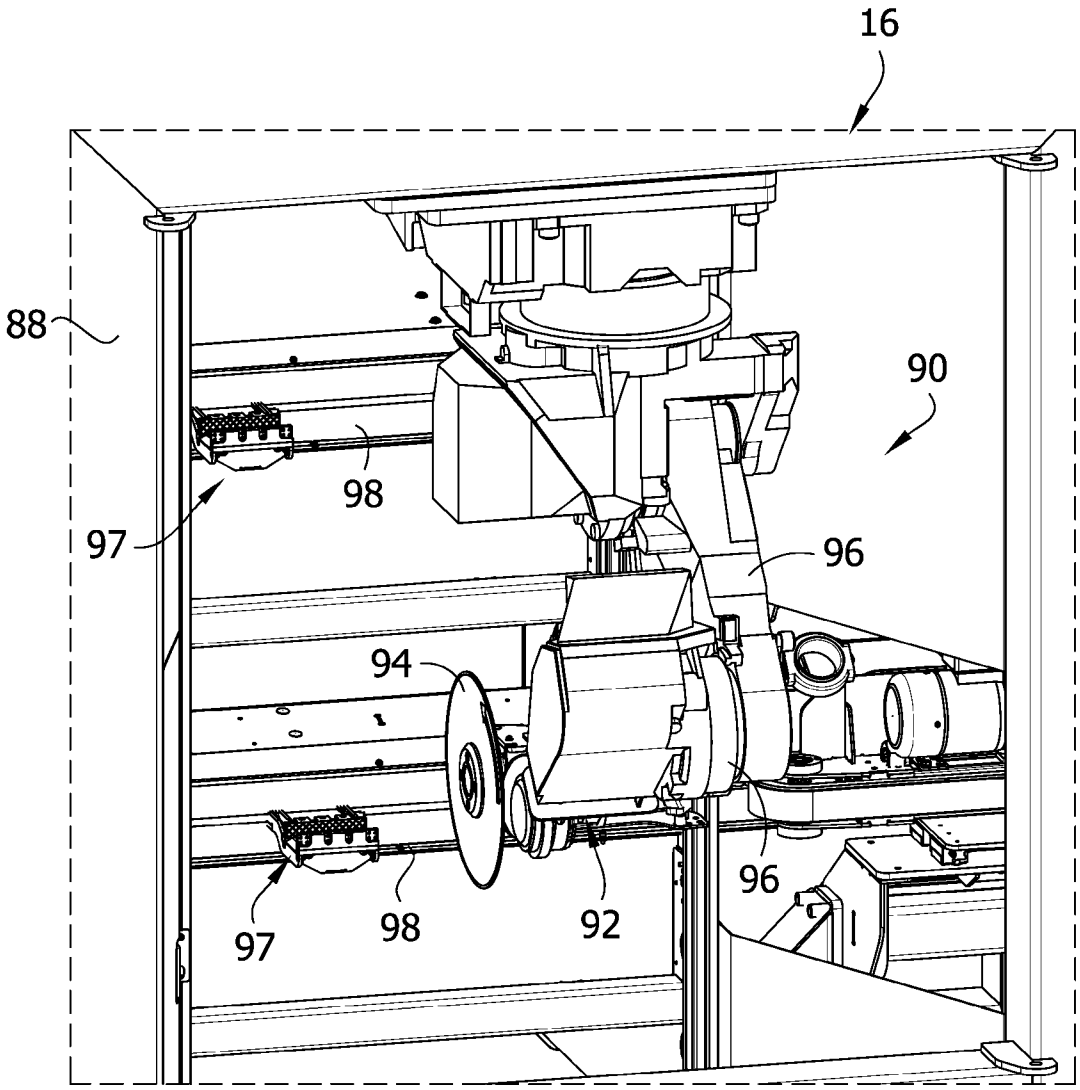



FIG. 9

FIG. 10



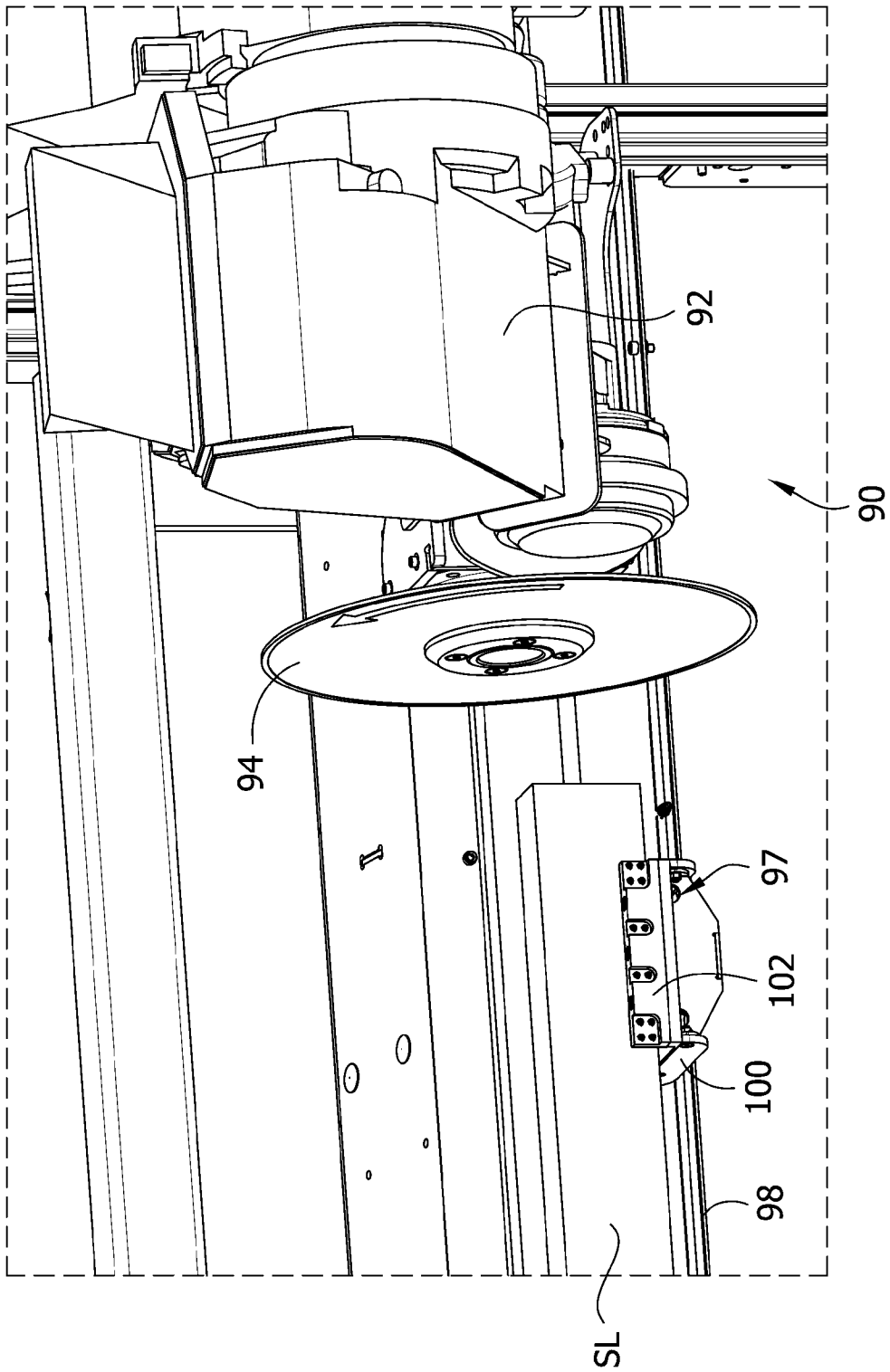



FIG. 11

FIG. 12

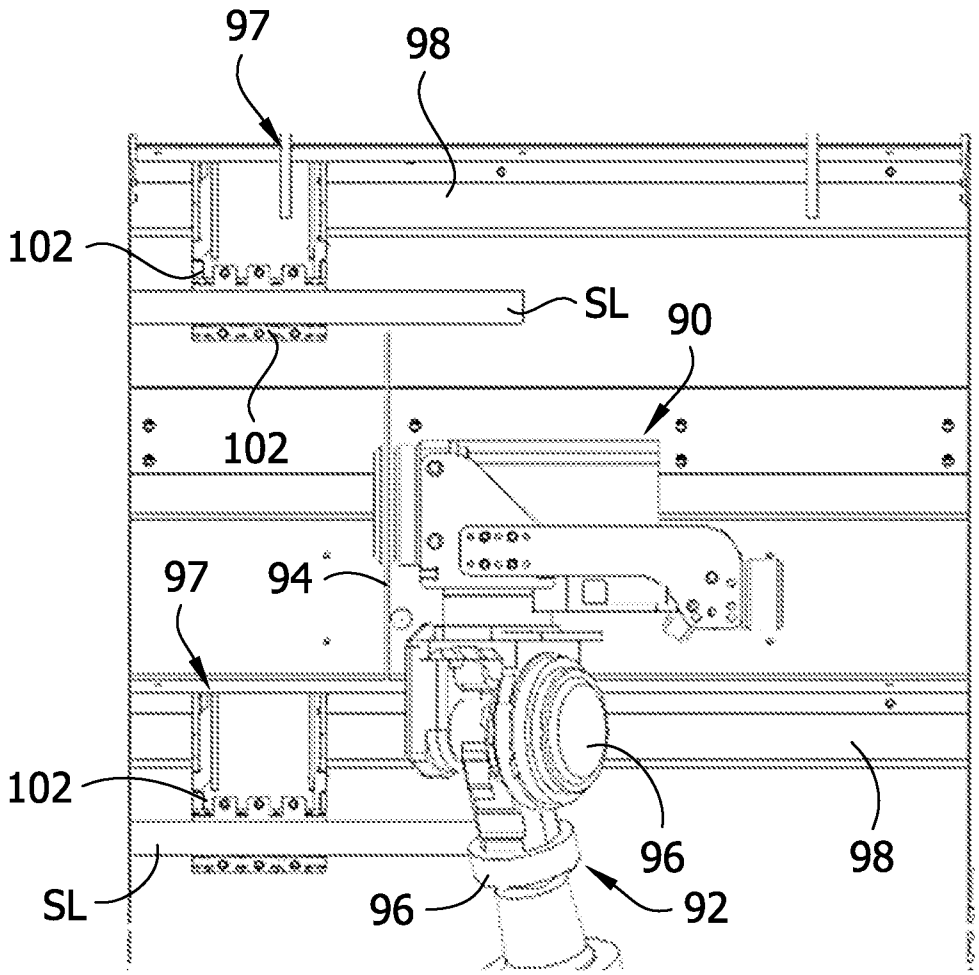
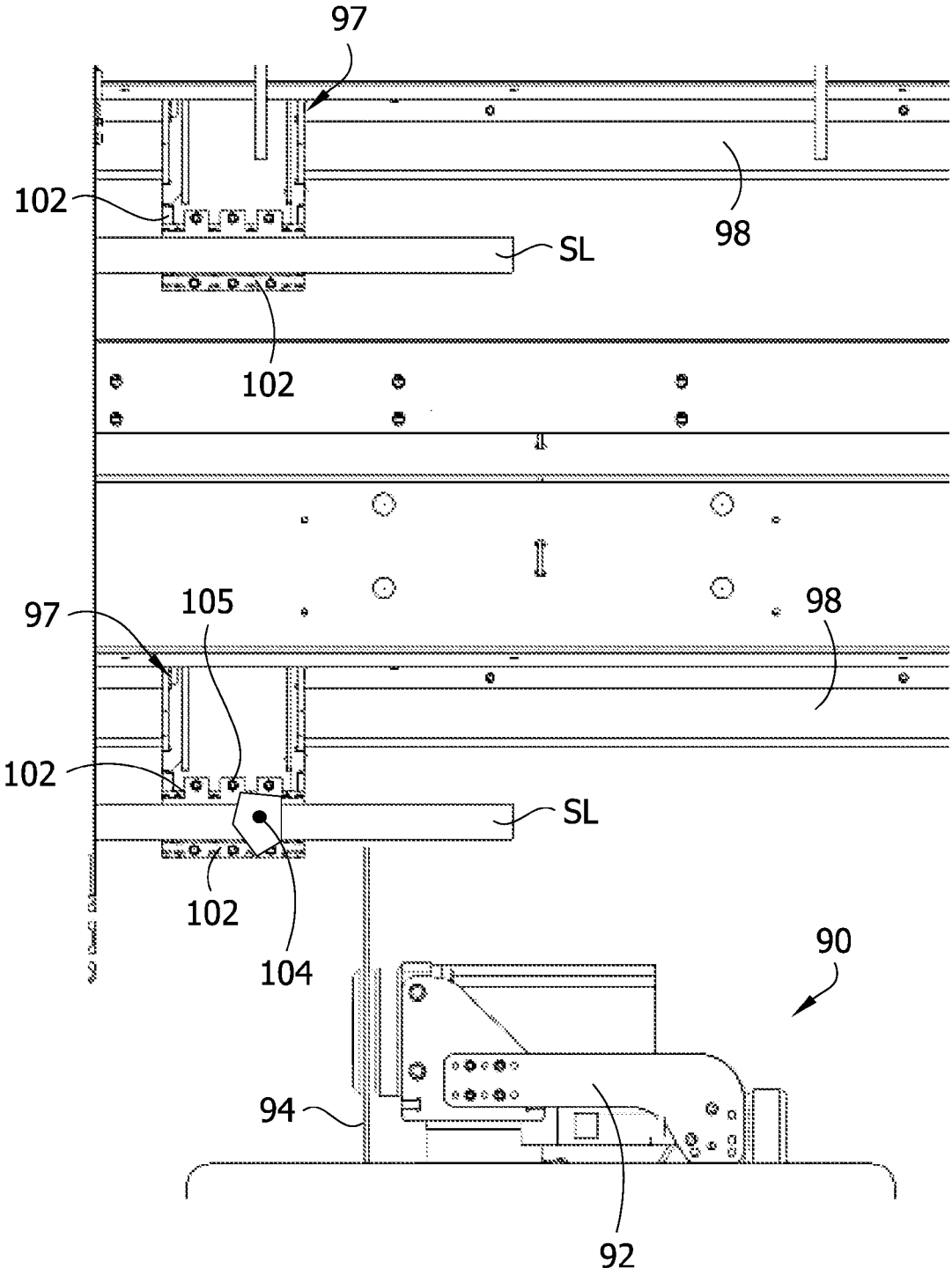


FIG. 13



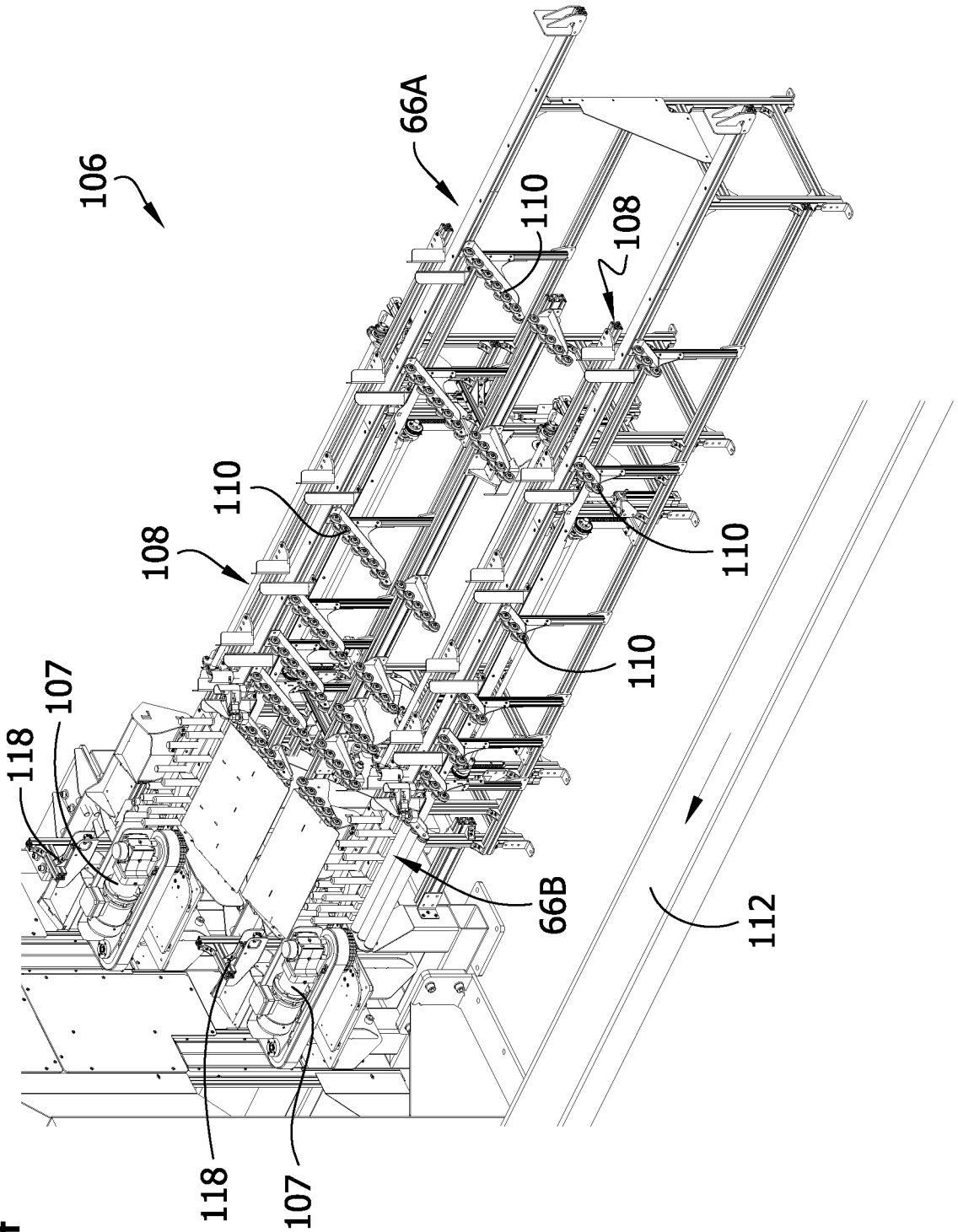



FIG. 14

FIG. 15

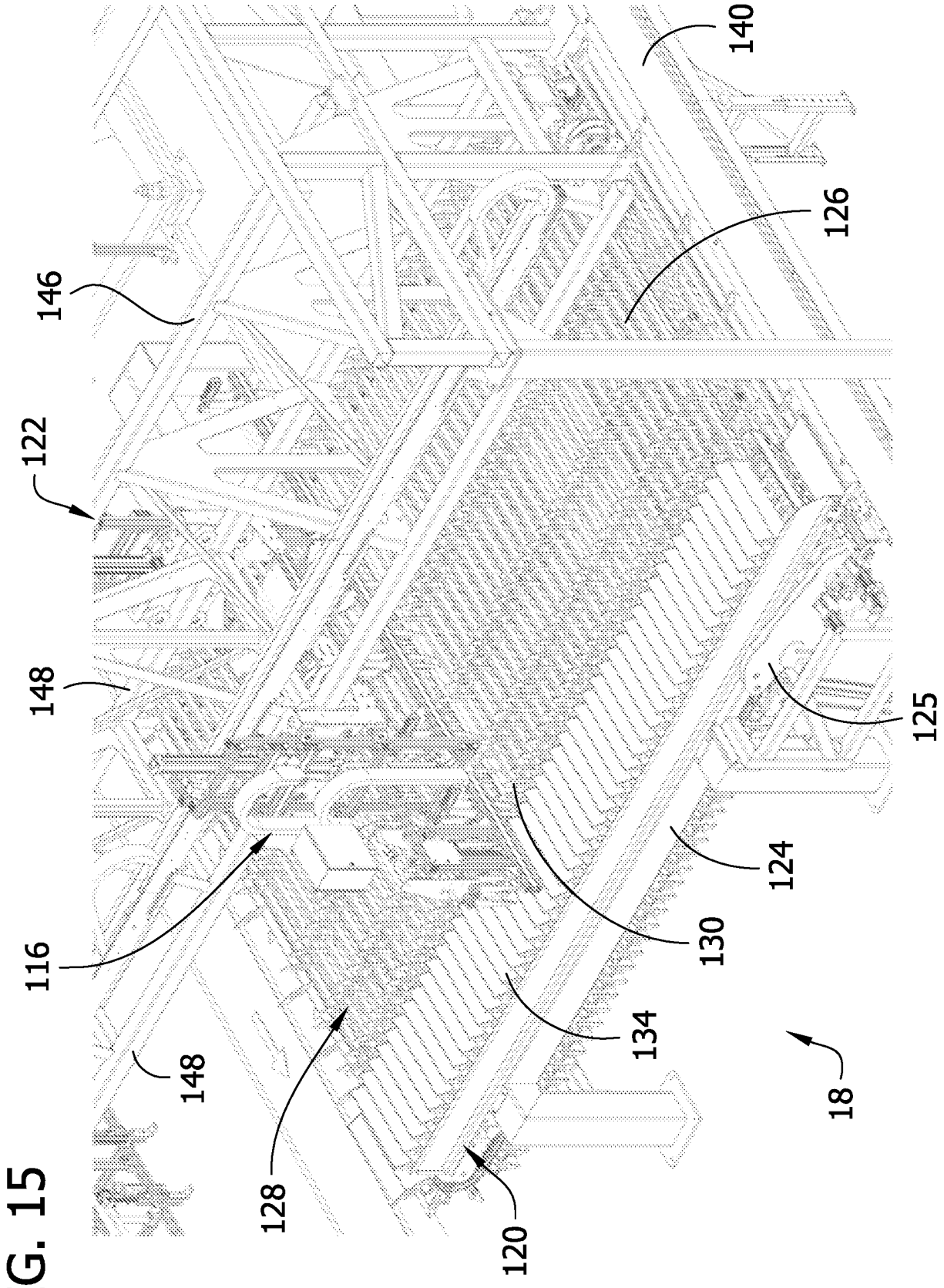
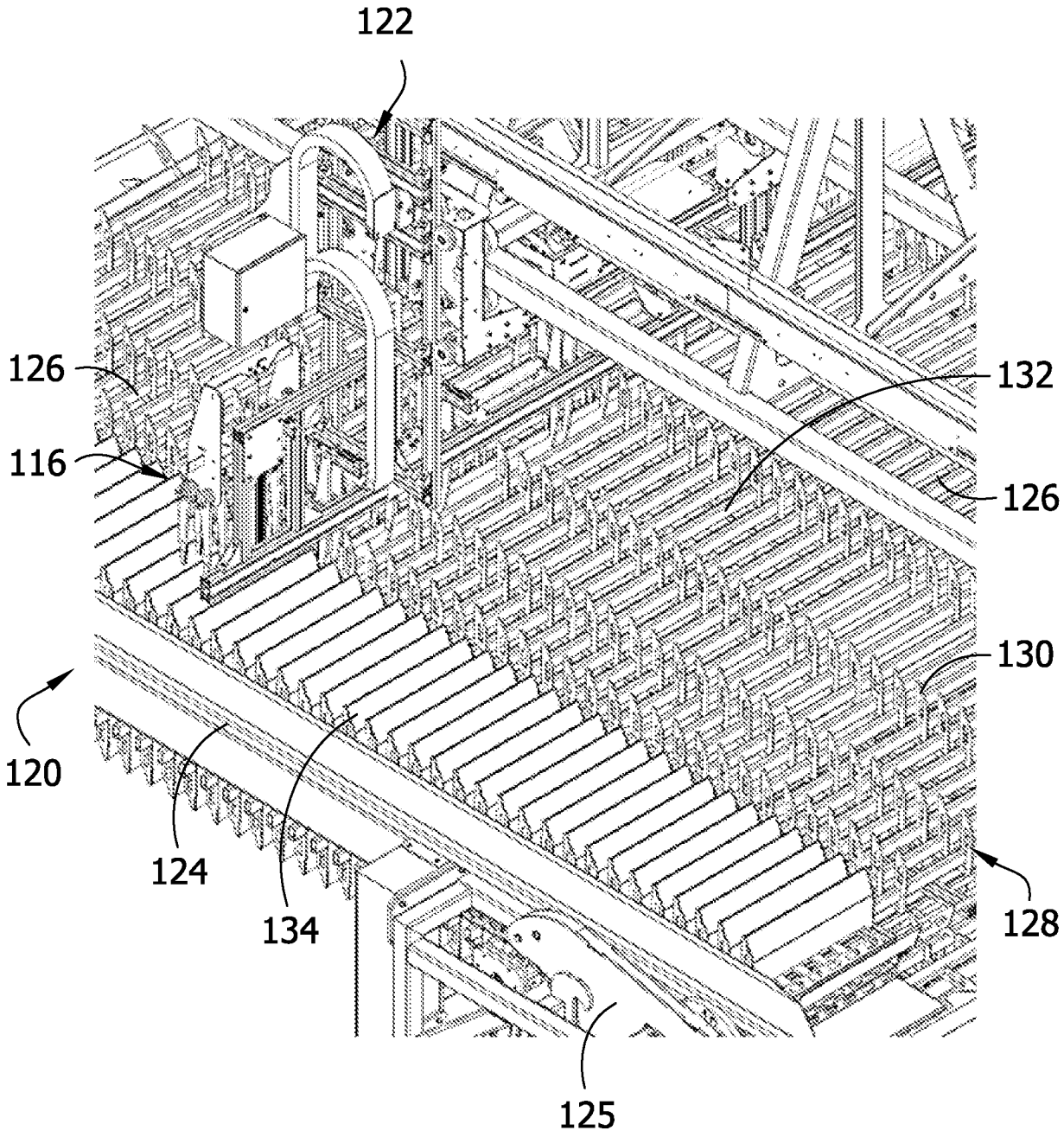



FIG. 15

FIG. 16



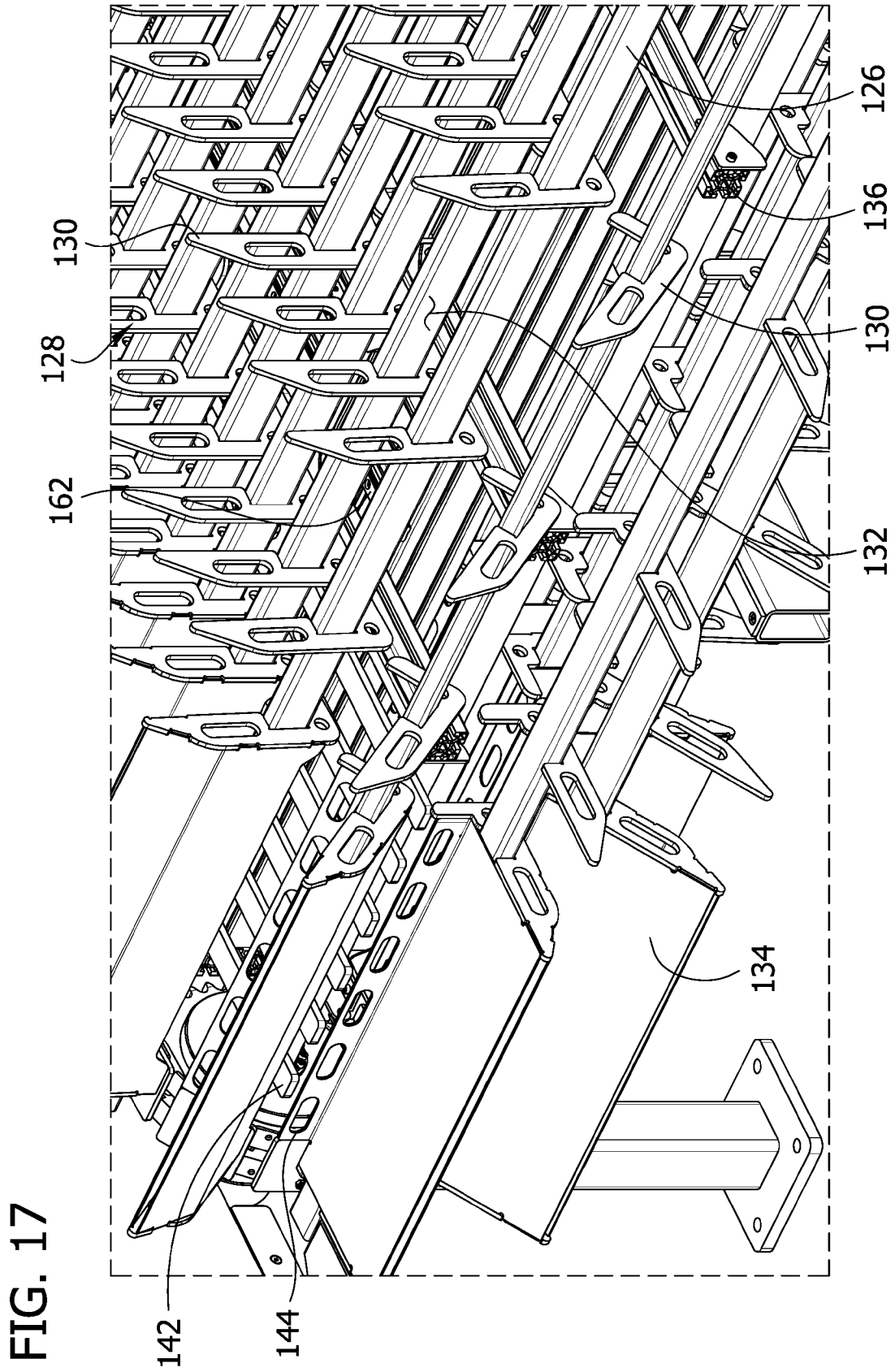



FIG. 17

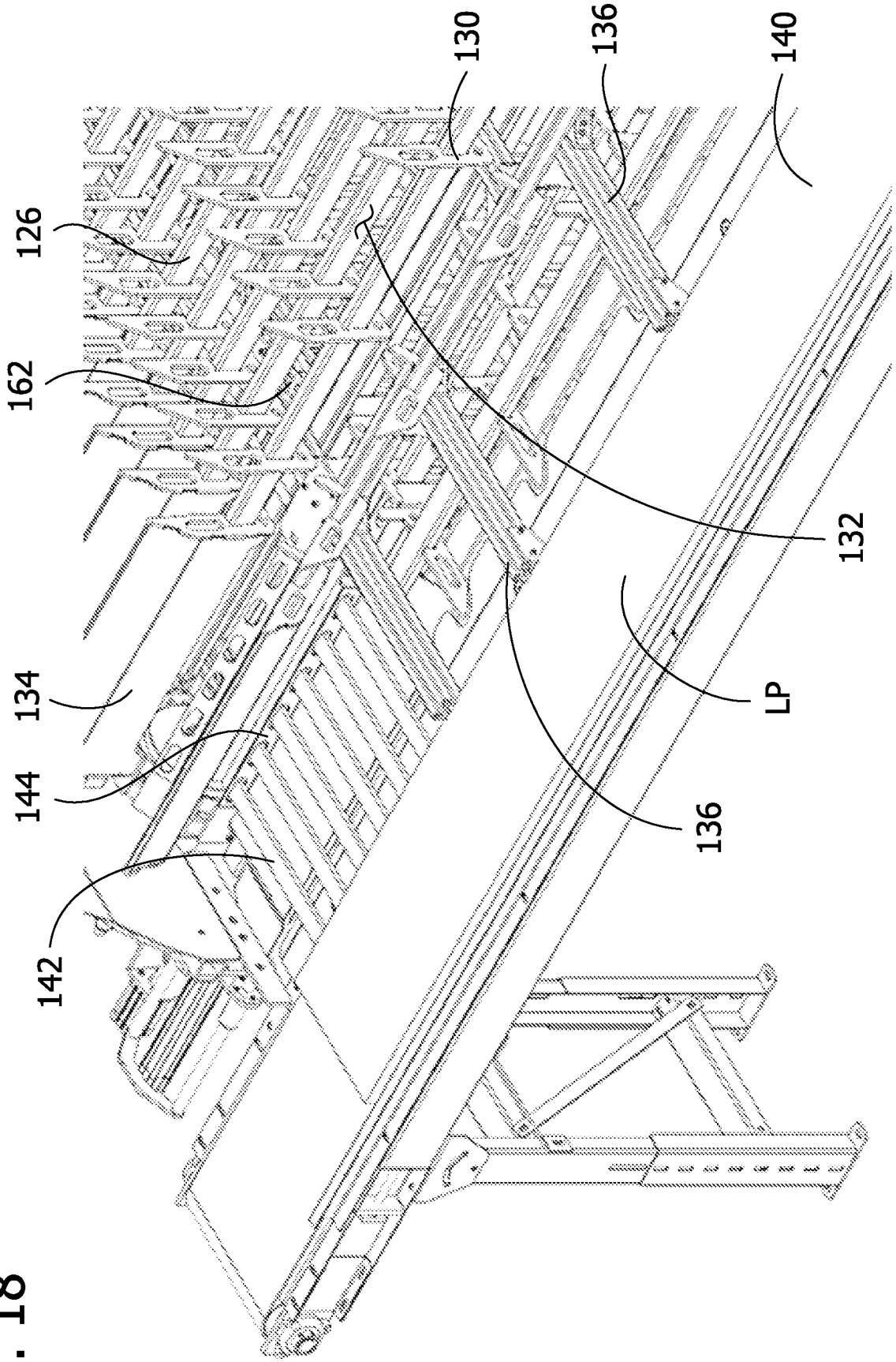



FIG. 18

FIG. 19

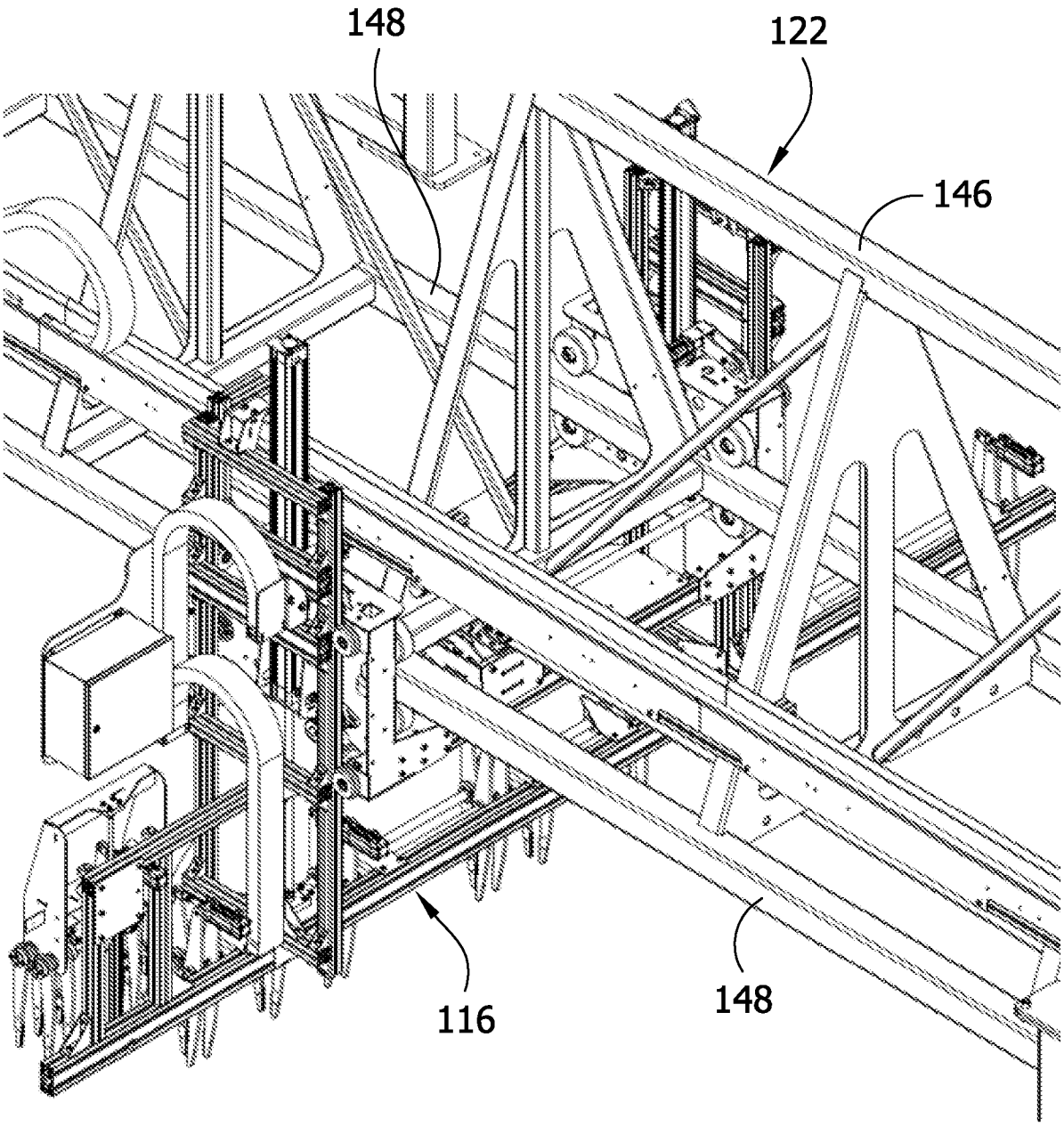


FIG. 20

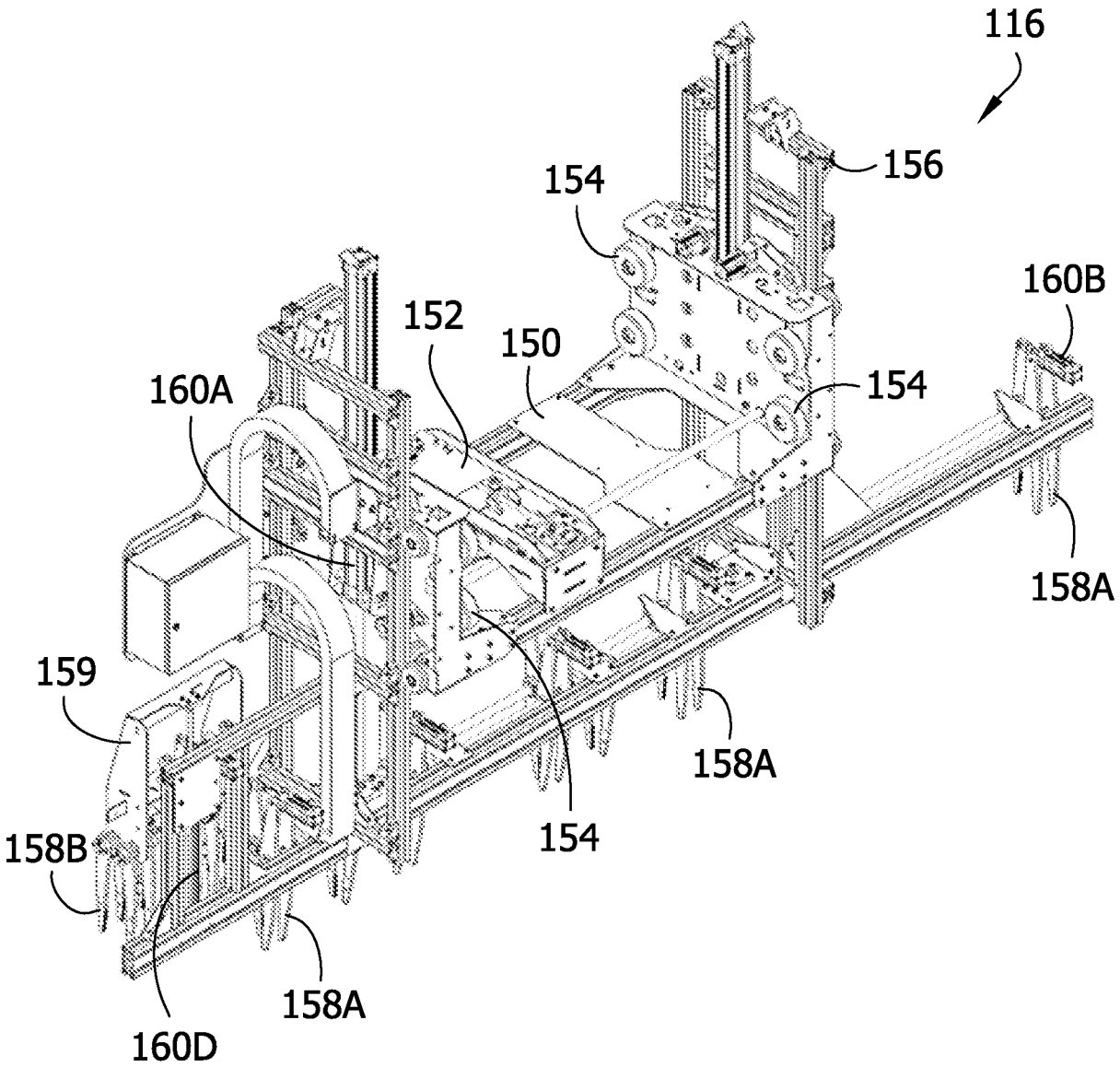
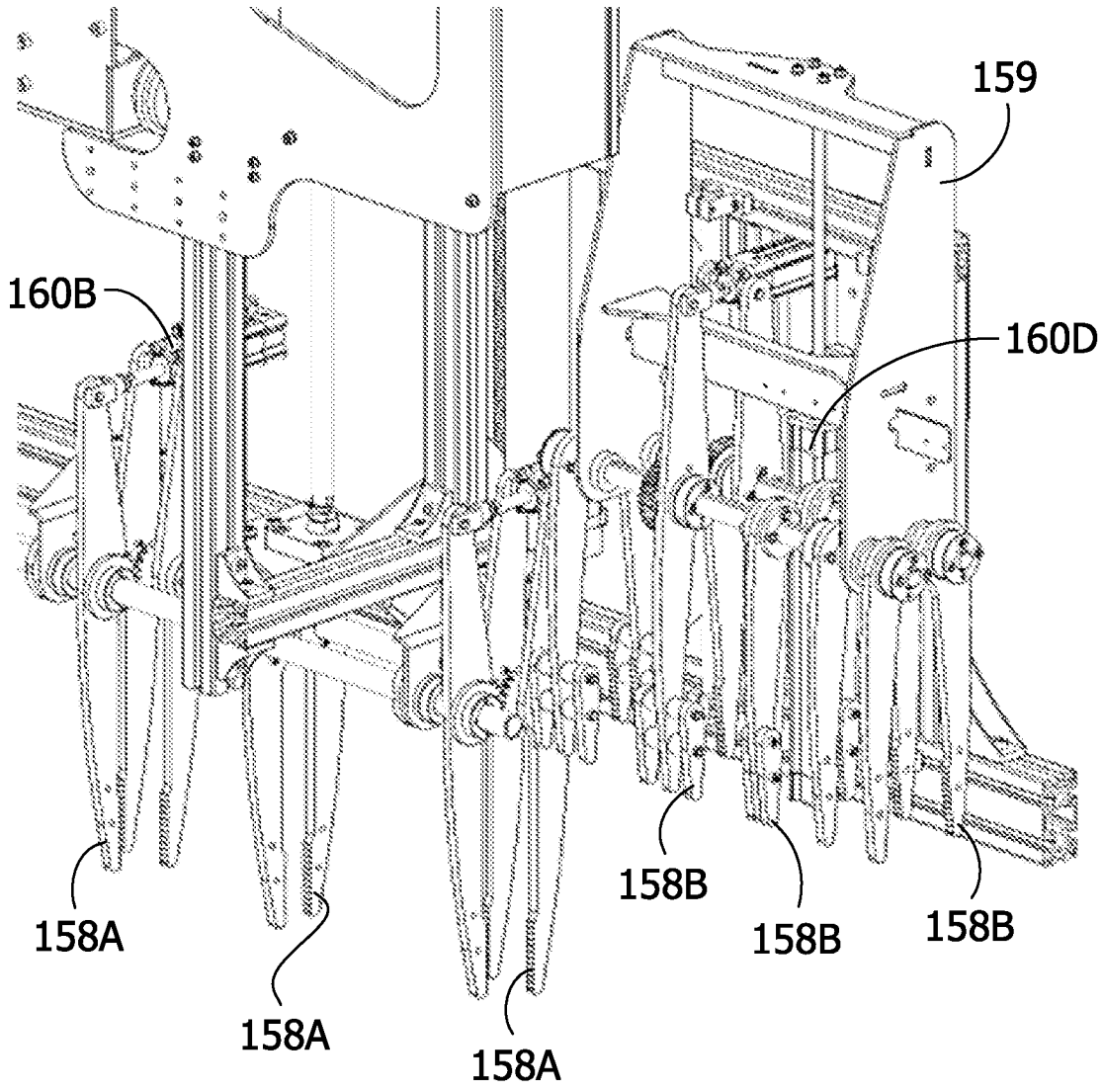

116
156
154
160B
152
150
160A
154
158A
159
158A
154
158B
158A
160D

FIG. 21



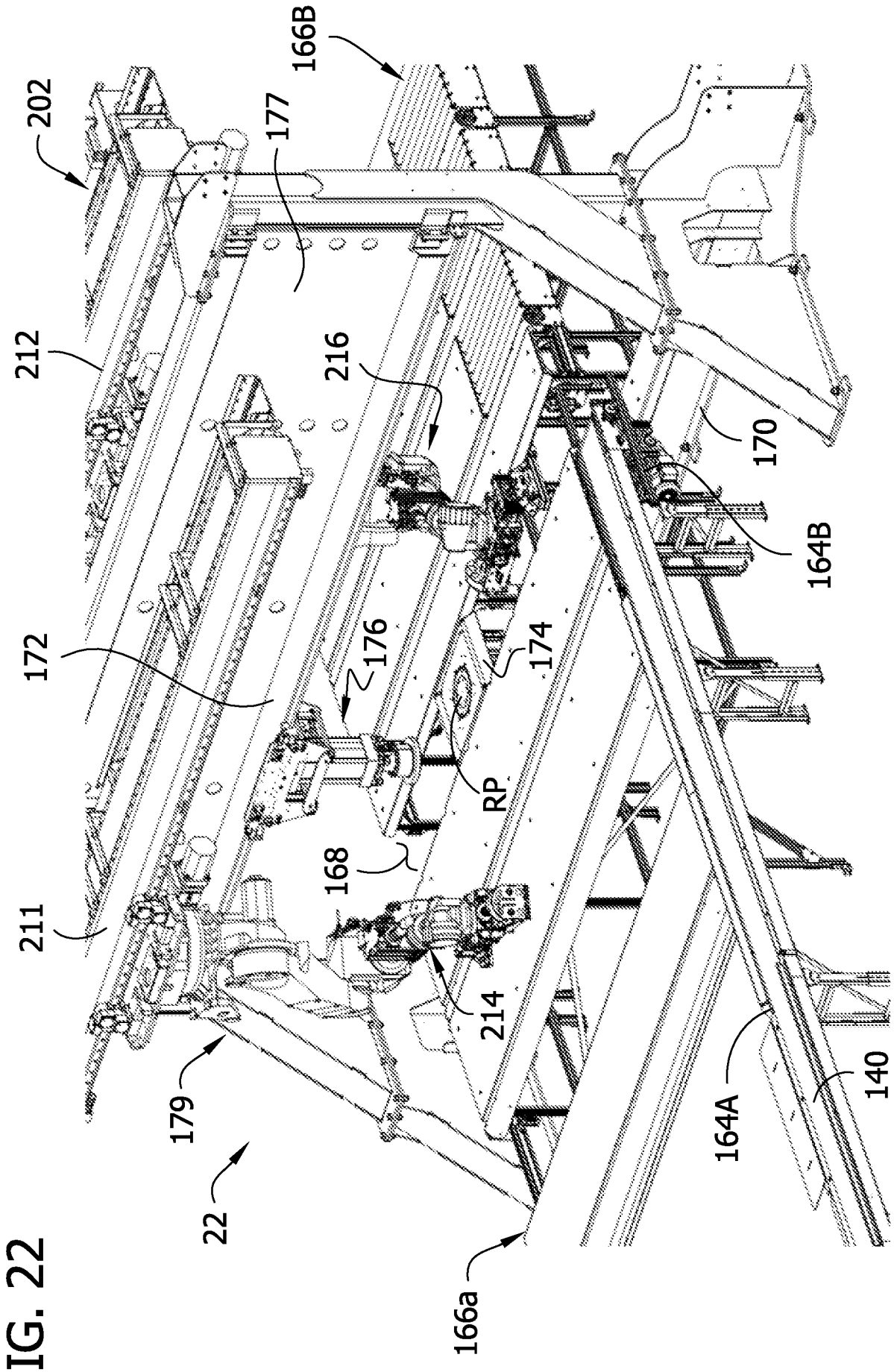

166B
202
177
212
216
170
164B
172
176
174
RP
168
211
214
179
22
164A
140
166a

FIG. 22

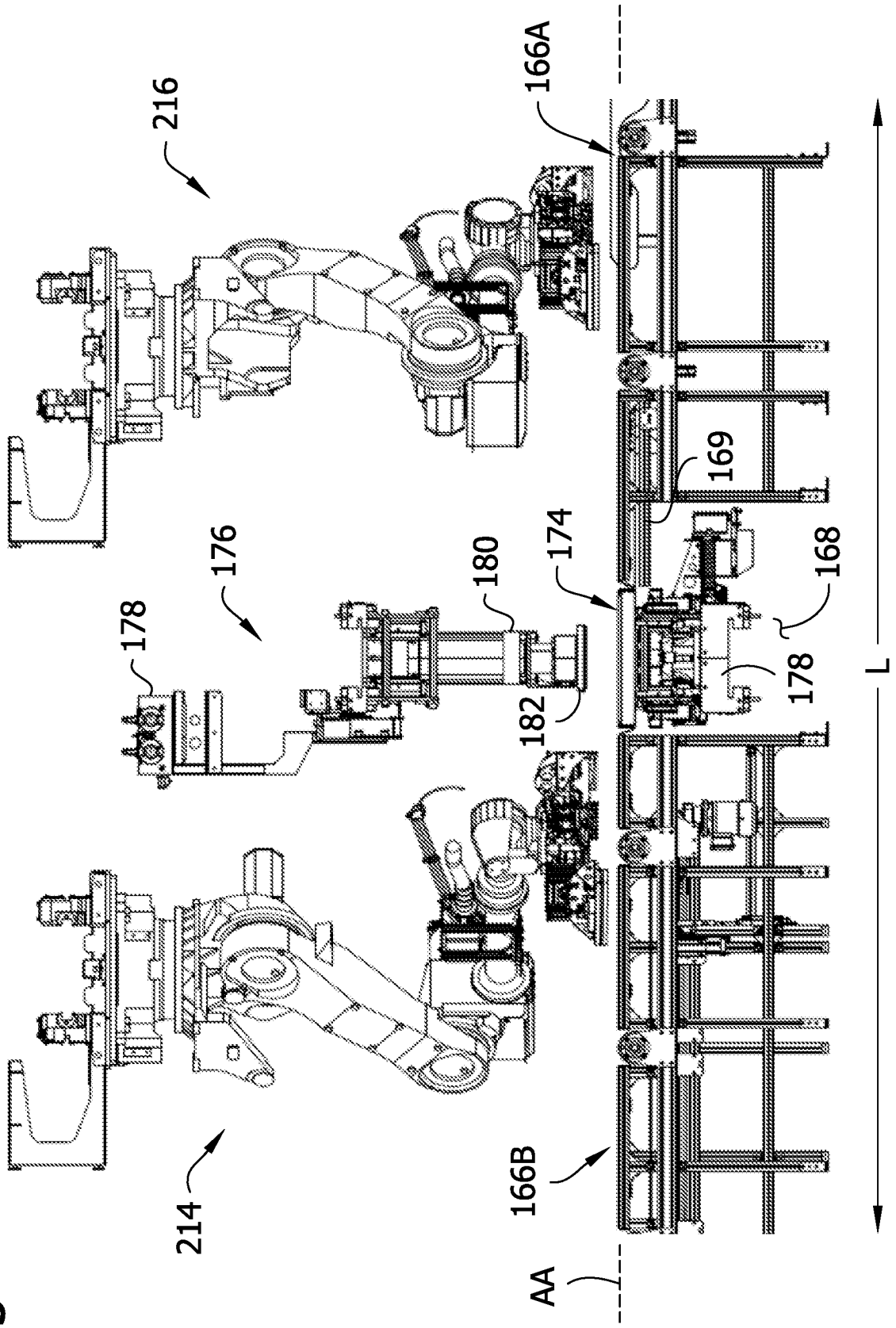



FIG. 23

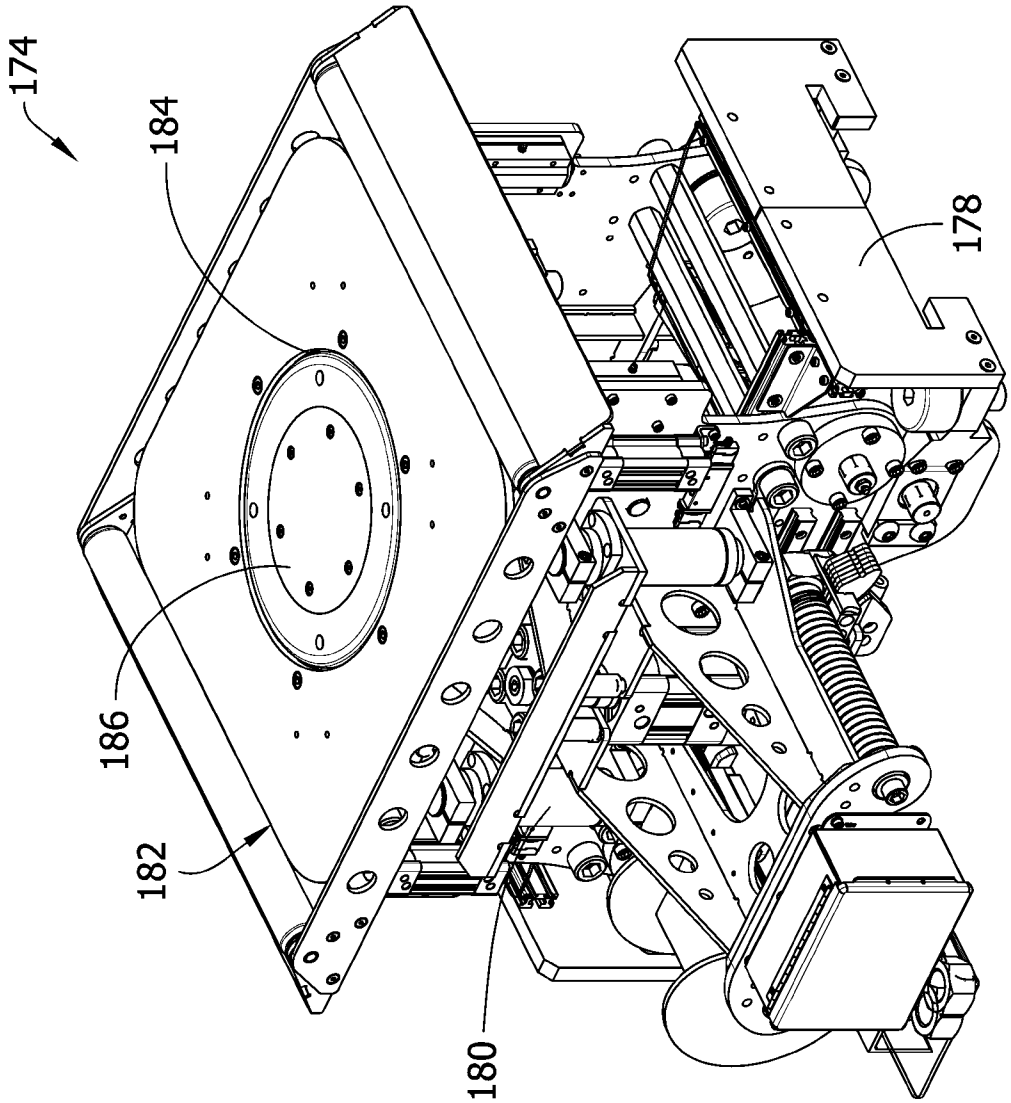



FIG. 24

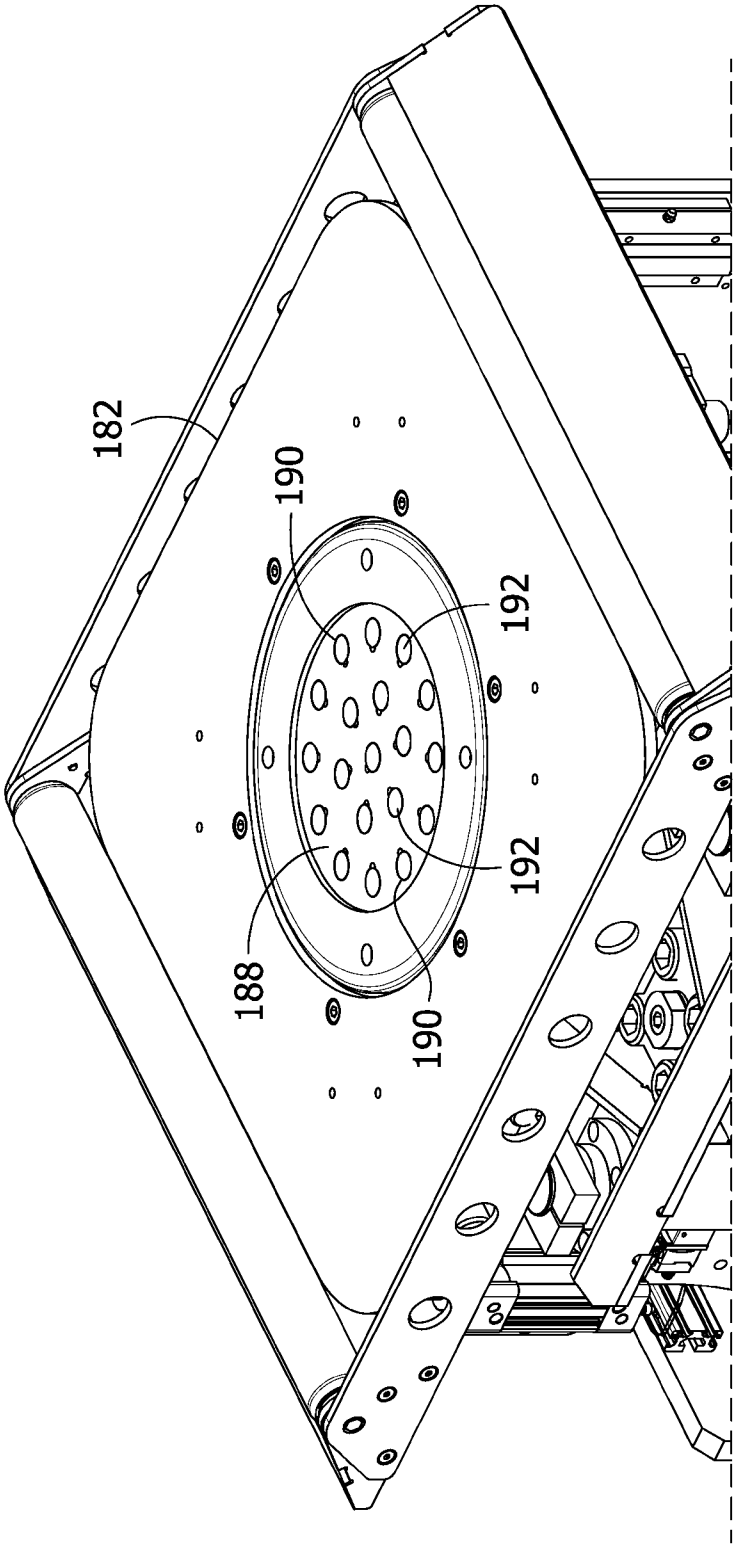
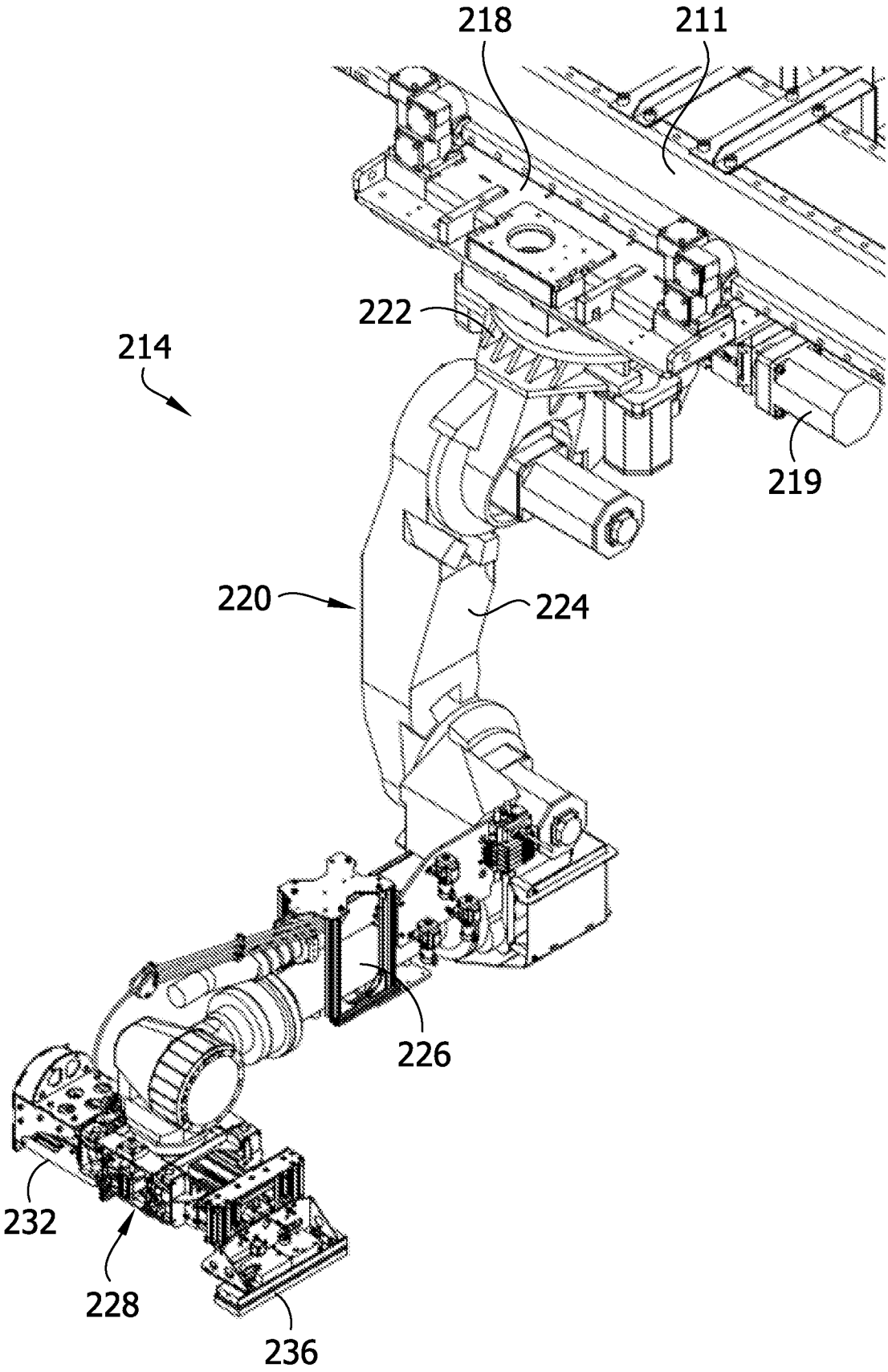



FIG. 25

FIG. 26



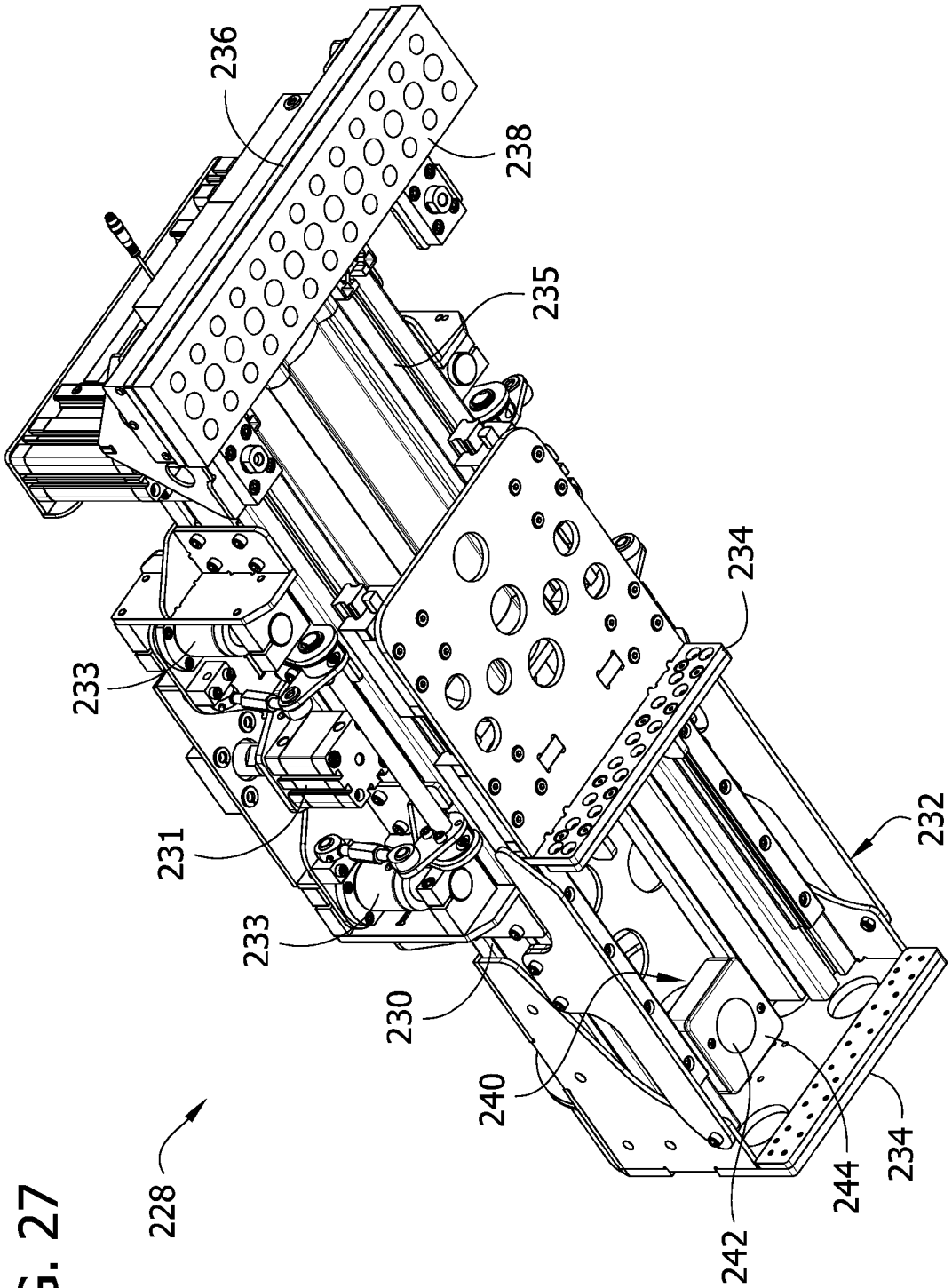
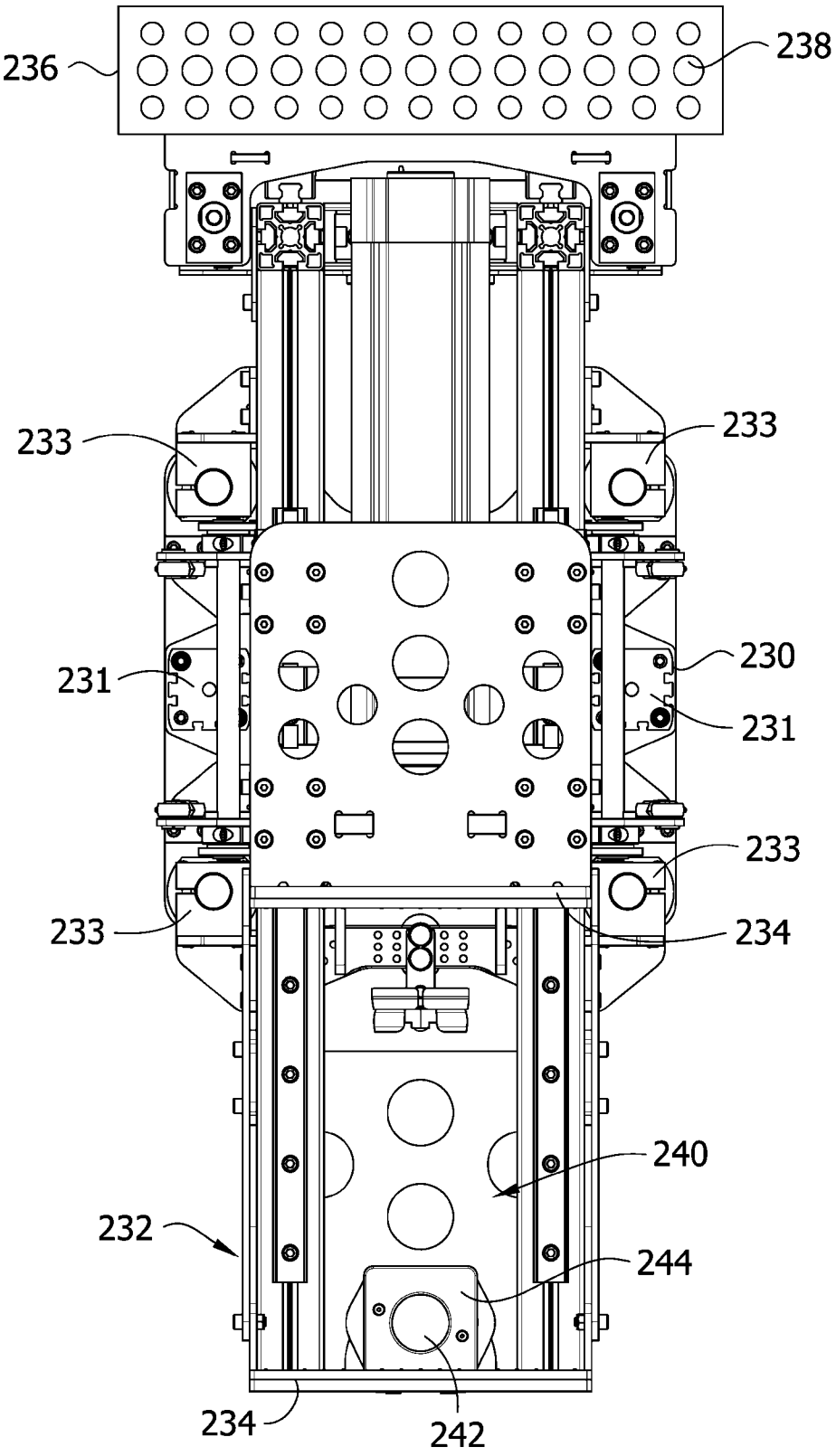



FIG. 27

FIG. 28



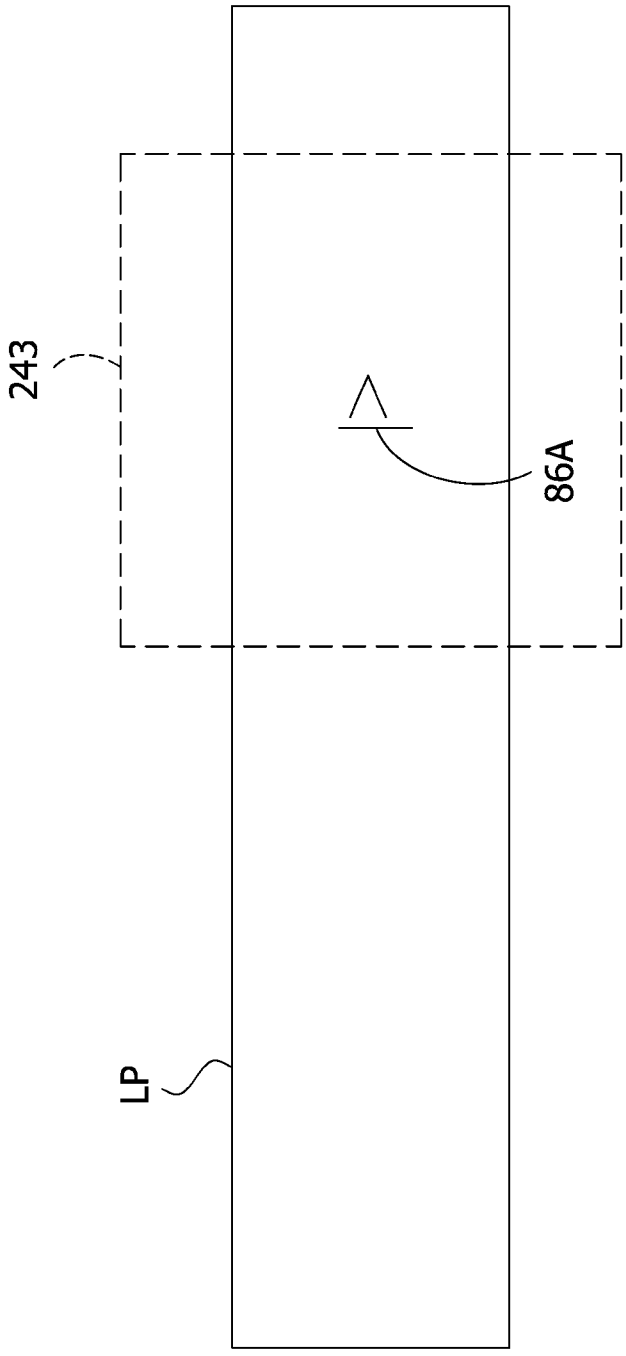



FIG. 29

FIG. 30

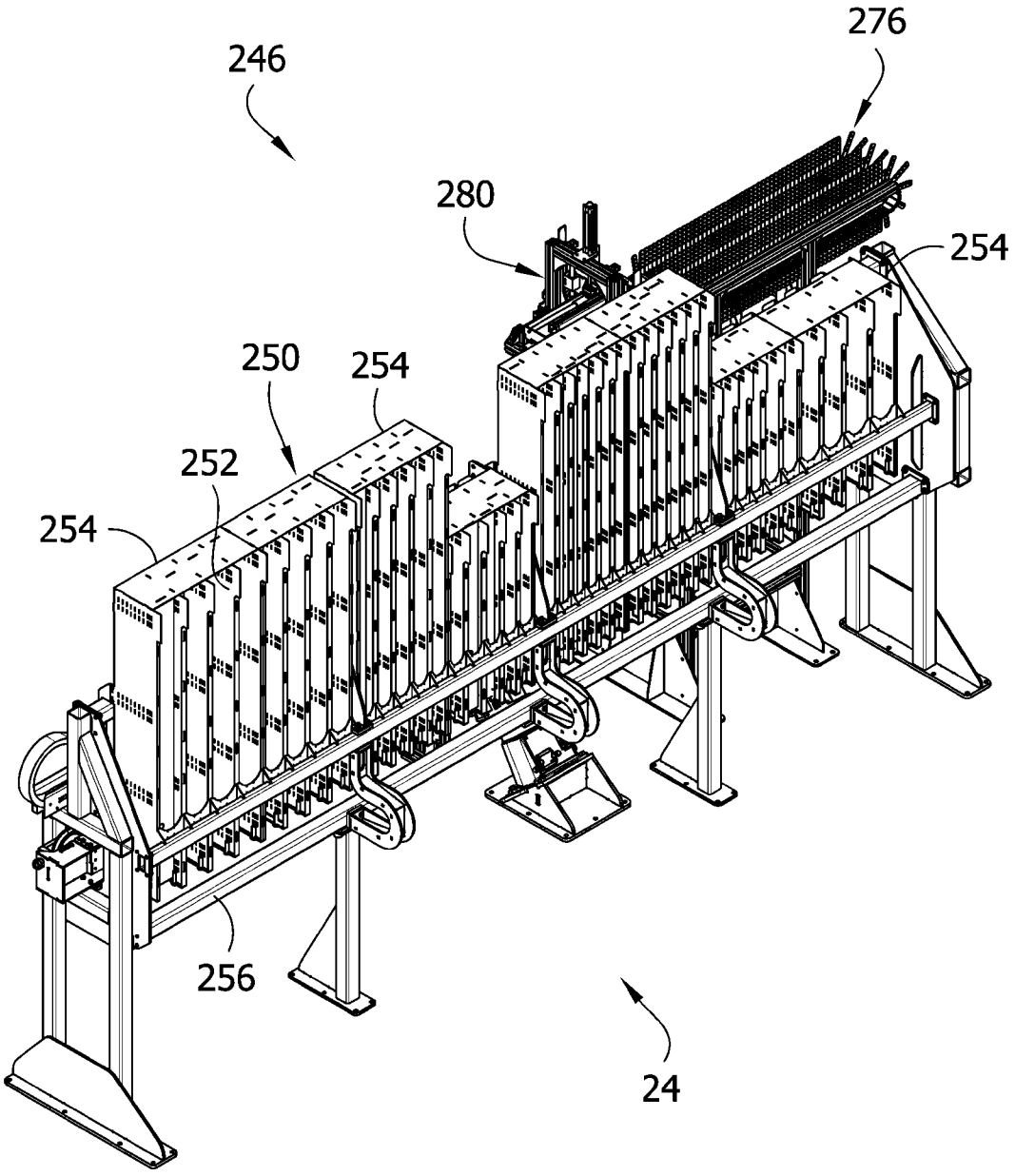
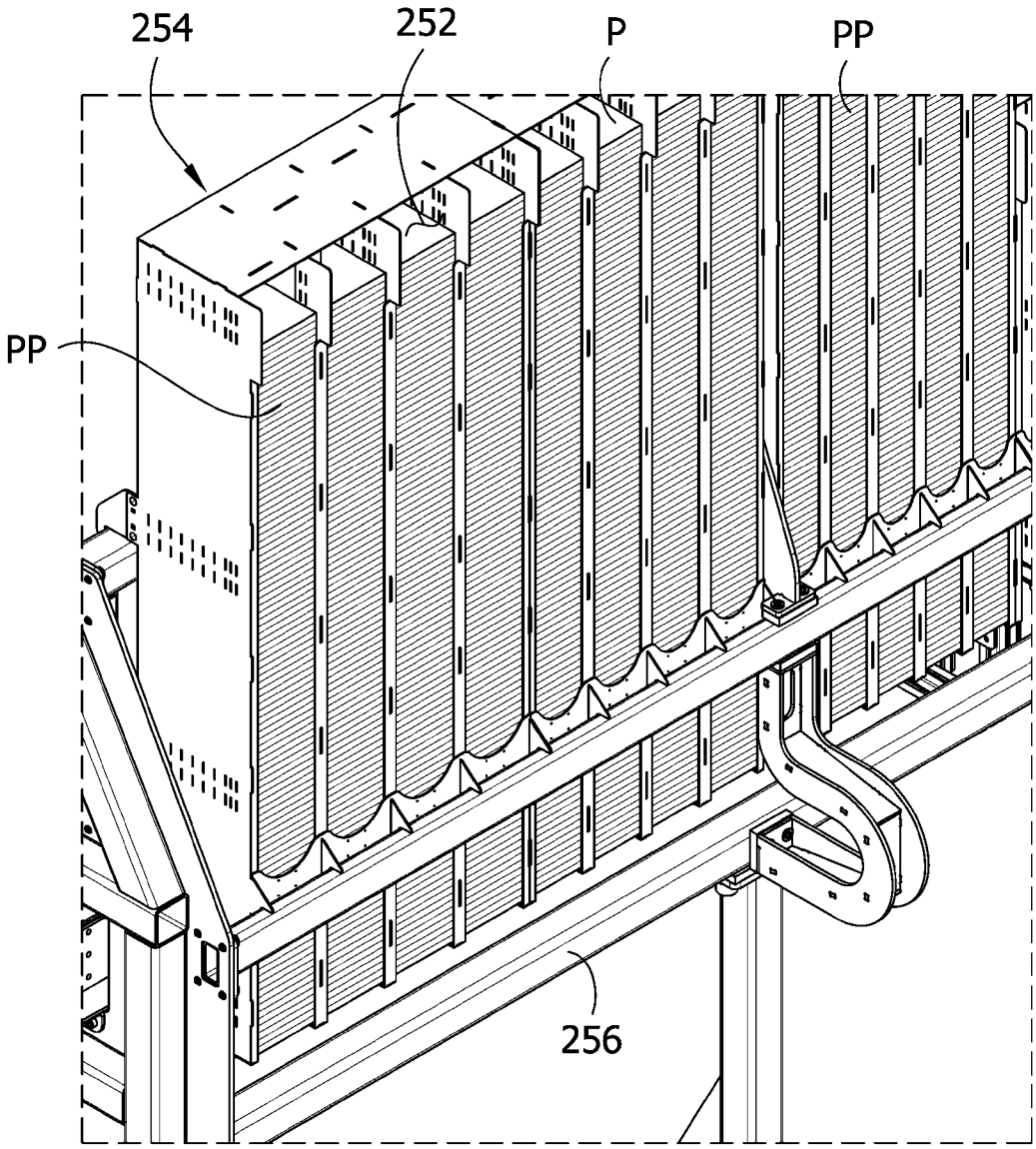


FIG. 31



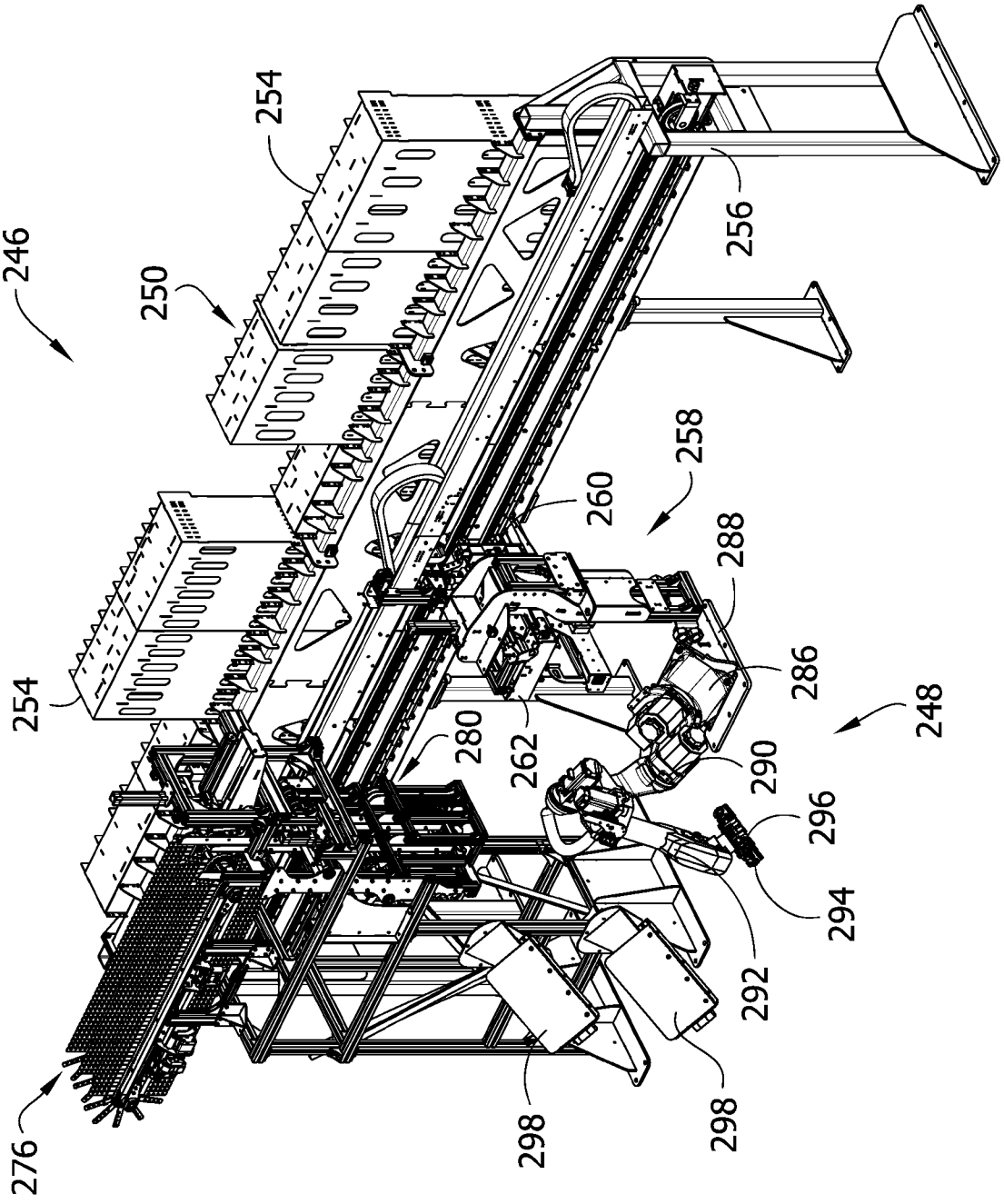

246
250
254
256
258
260
288
286
248
254
280
262
290
296
294
292
298
298
276

FIG. 32

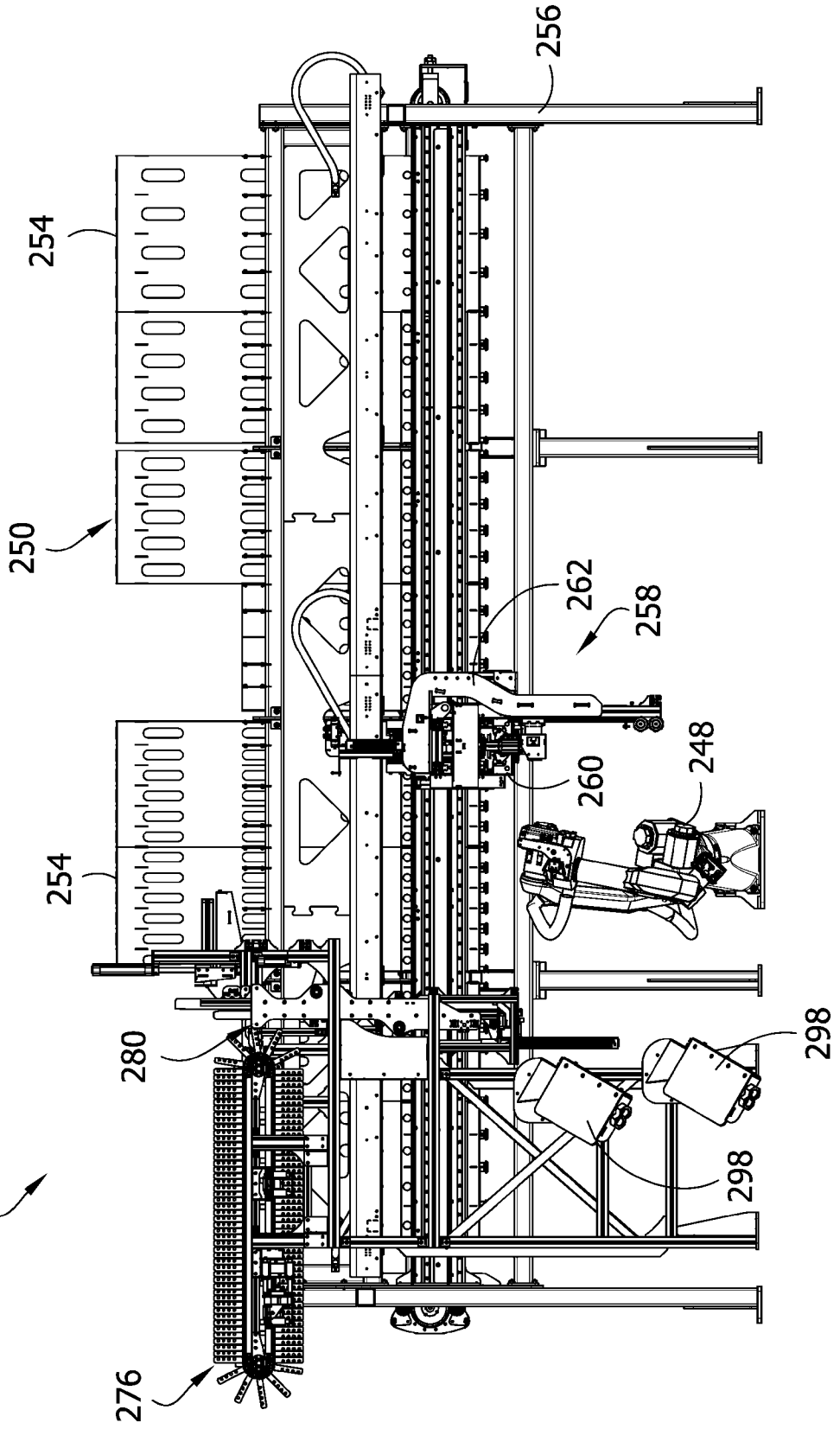



FIG. 33

FIG. 34

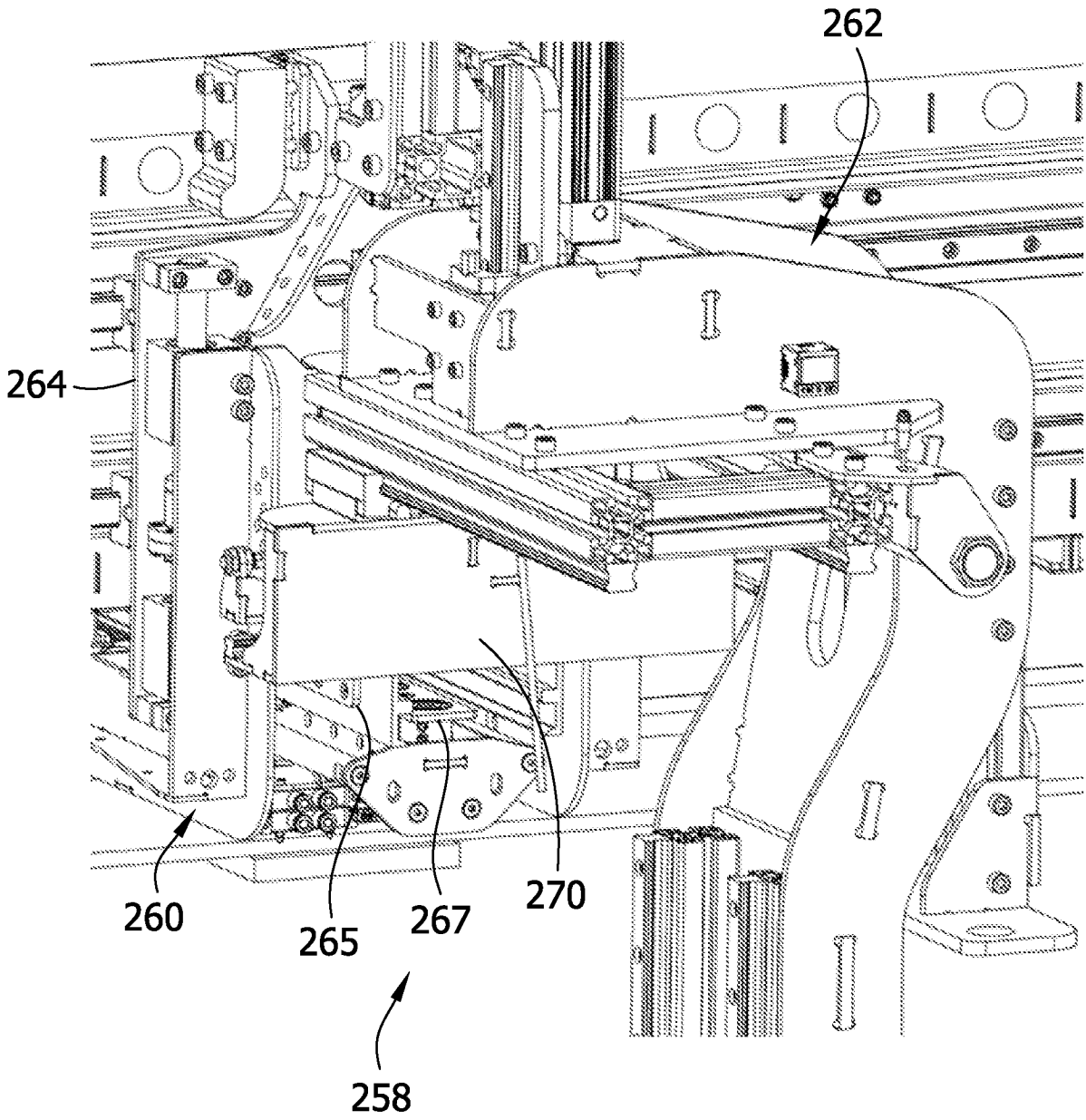


FIG. 35

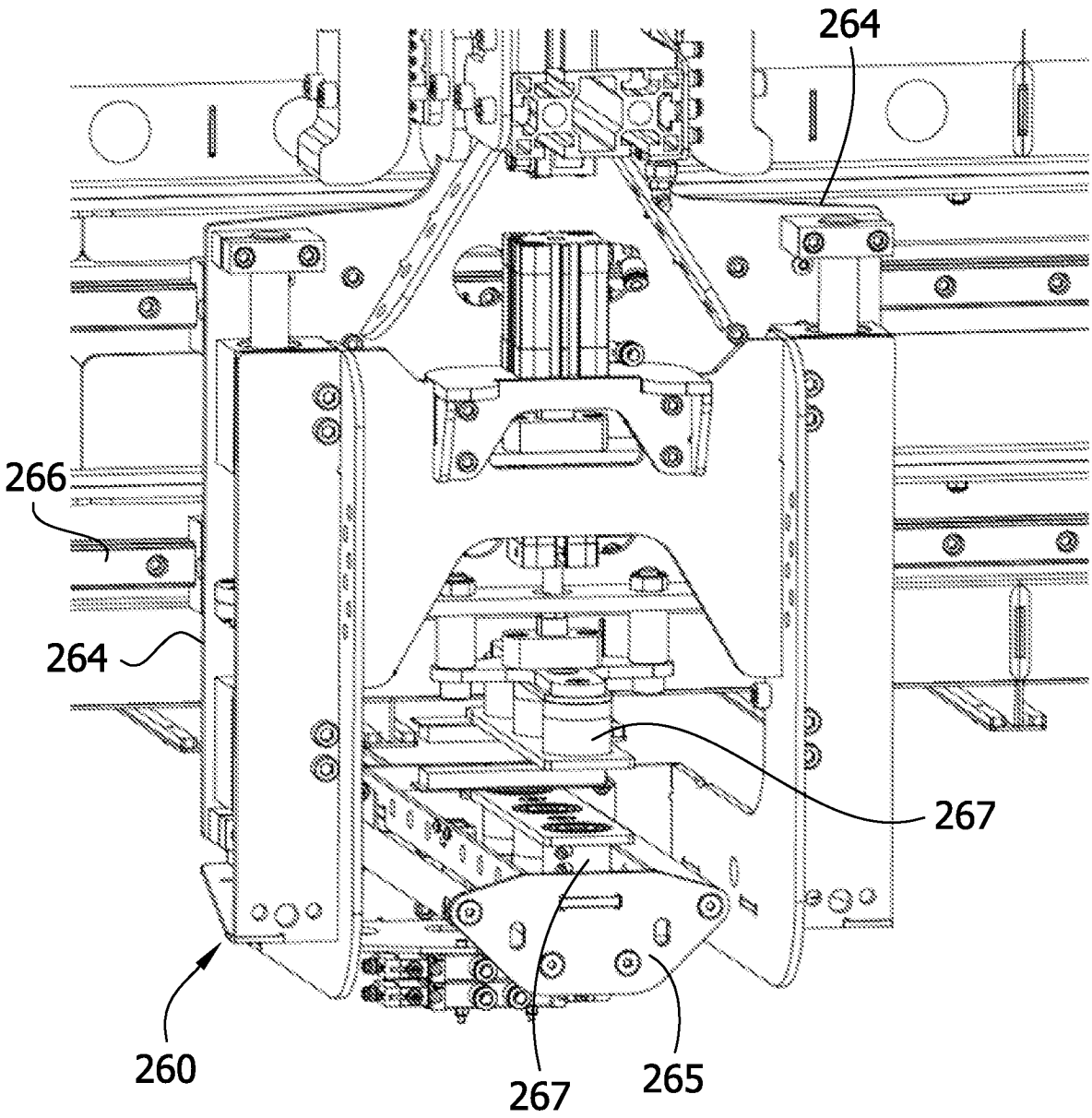


FIG. 36

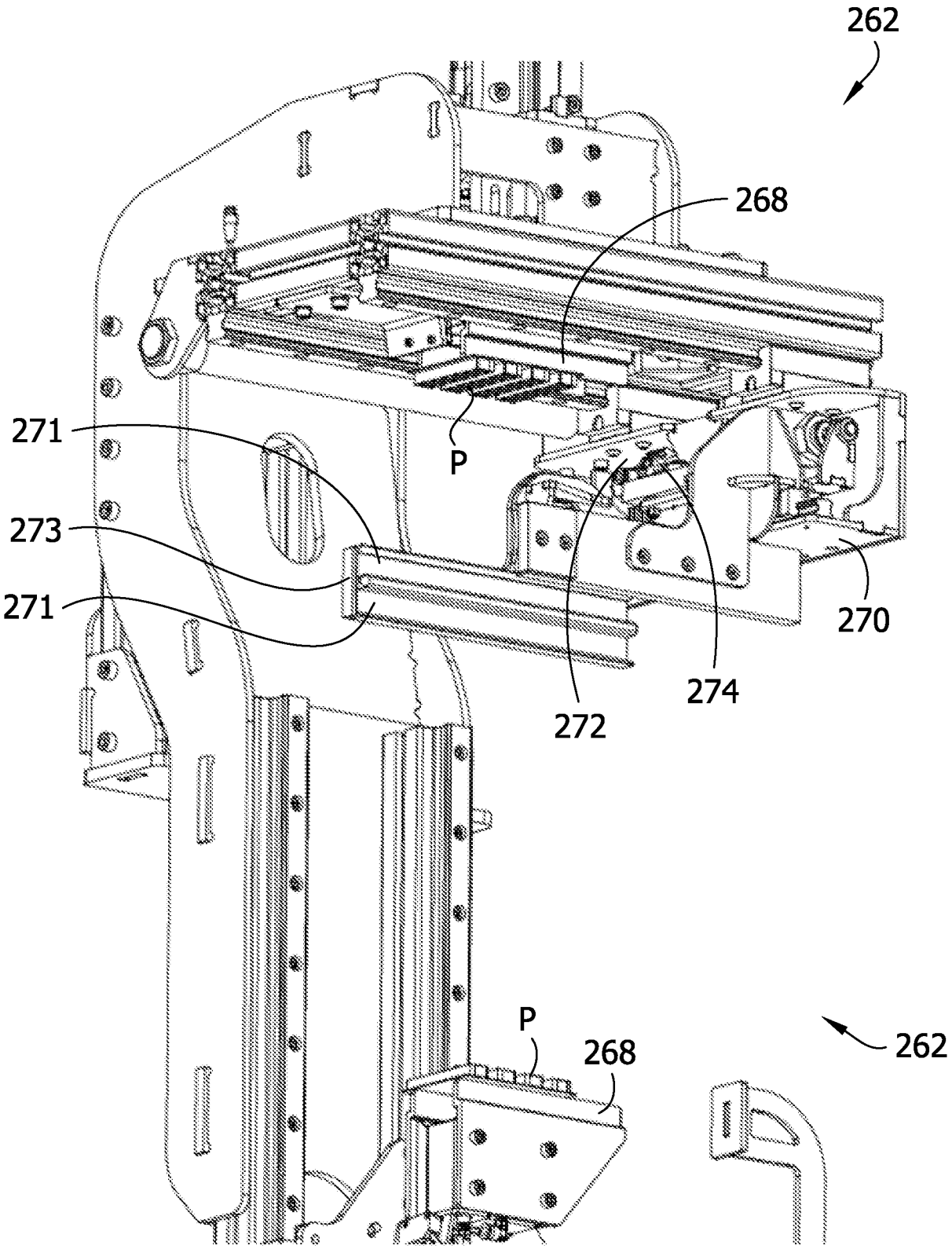


FIG. 37

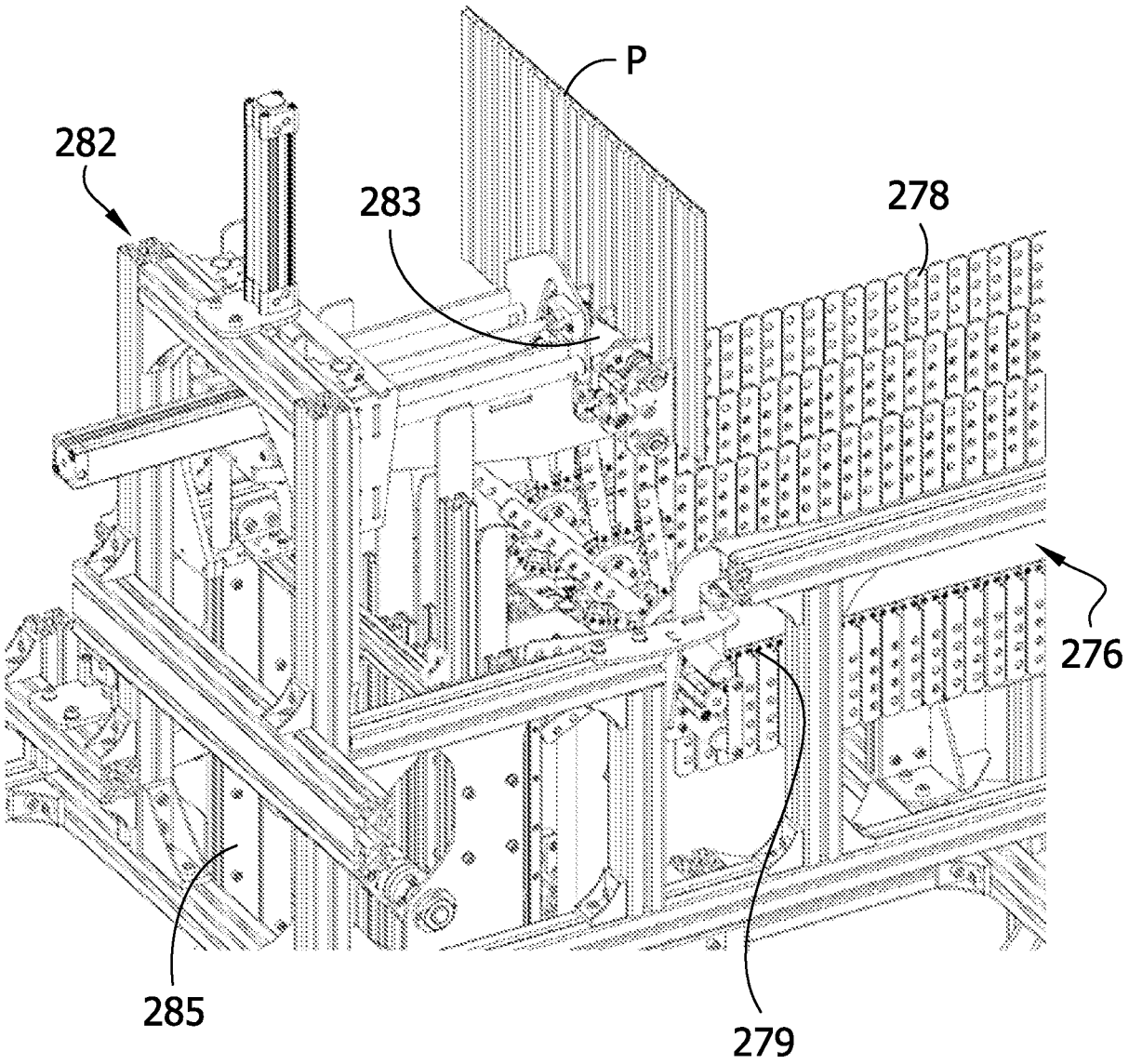


FIG. 38

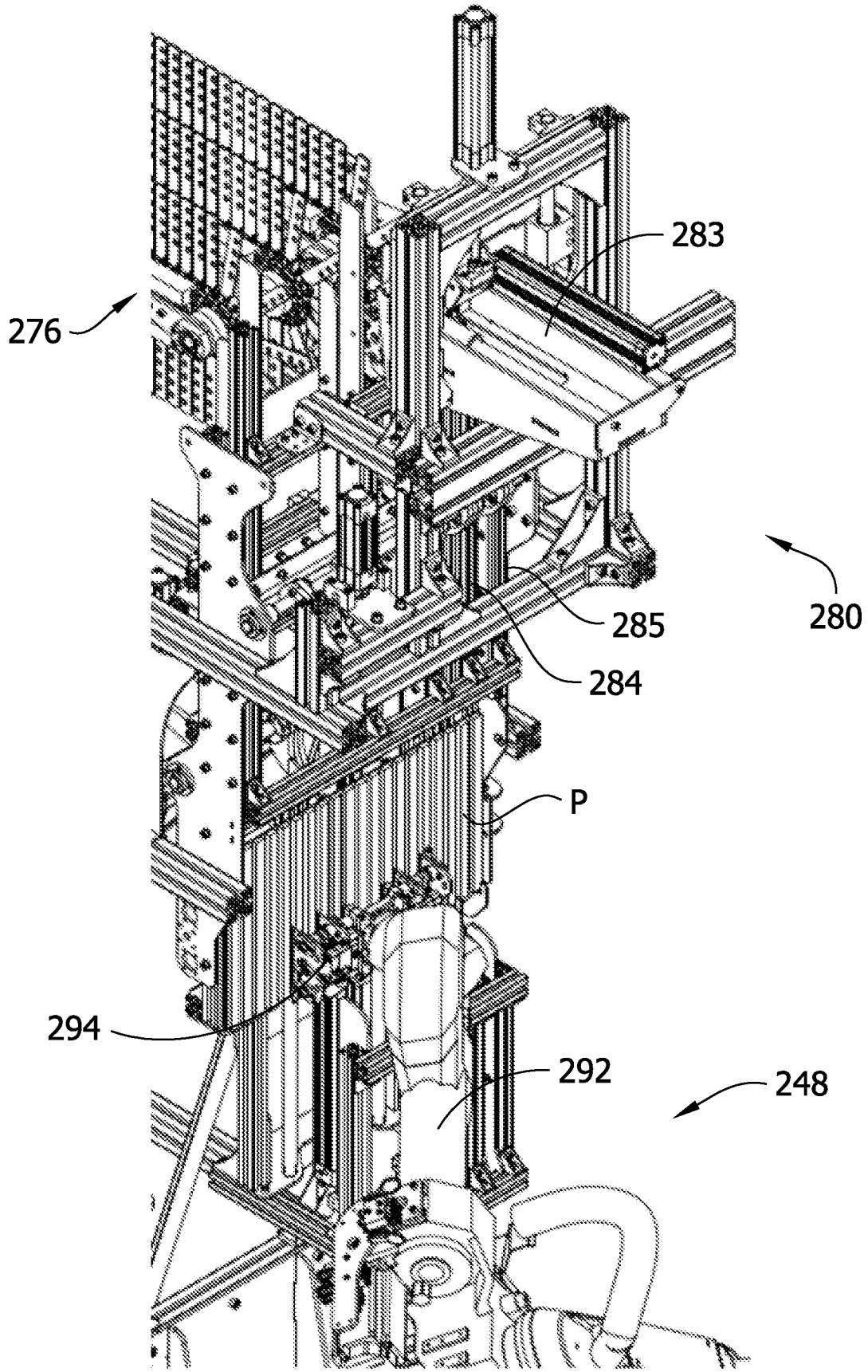


FIG. 39

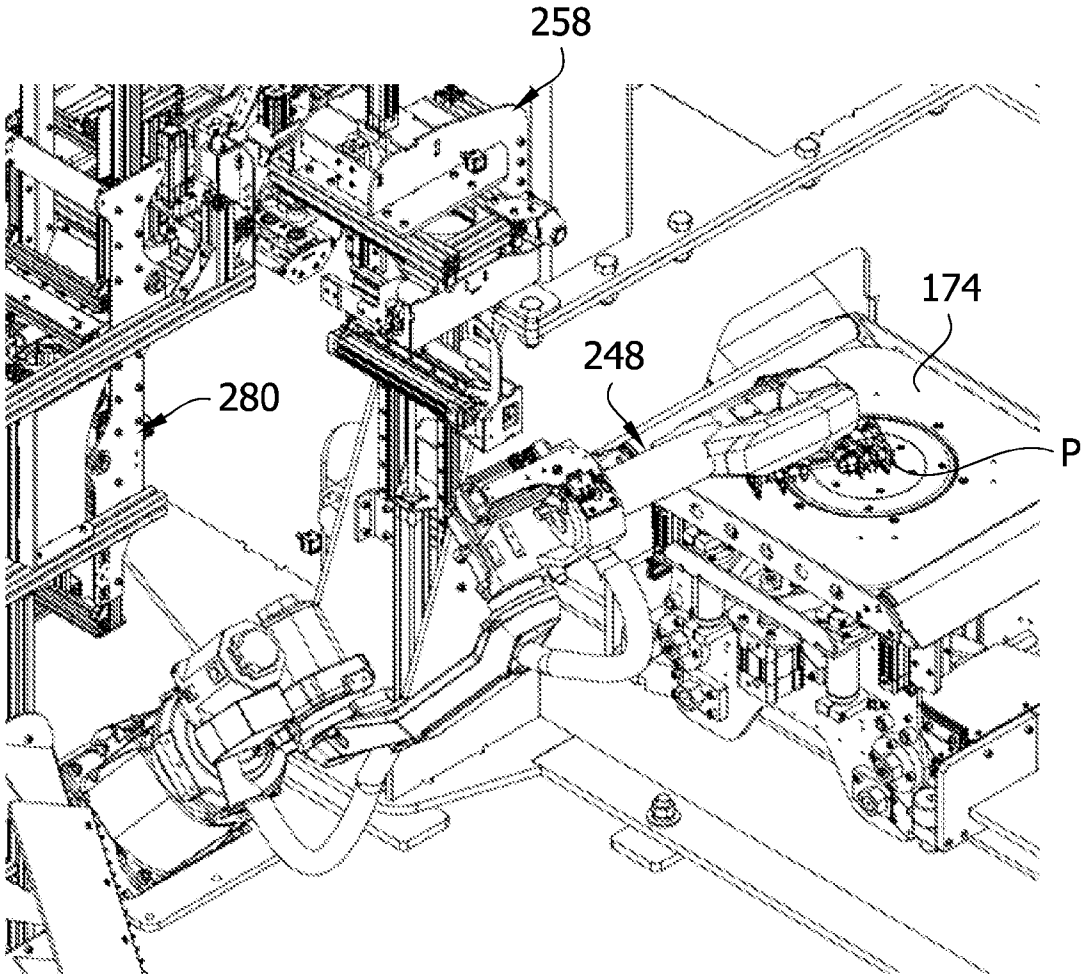


FIG. 40

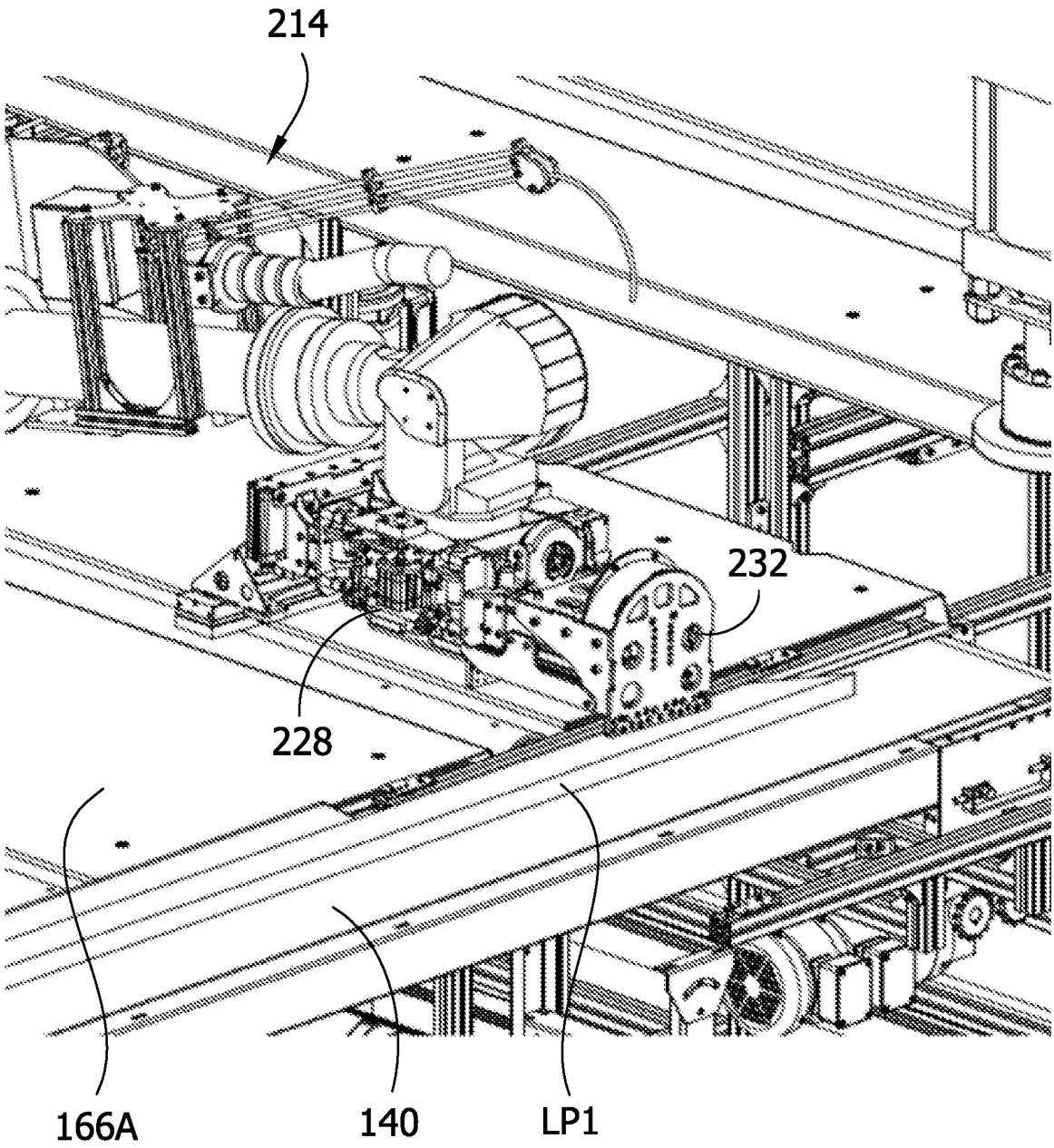


FIG. 41

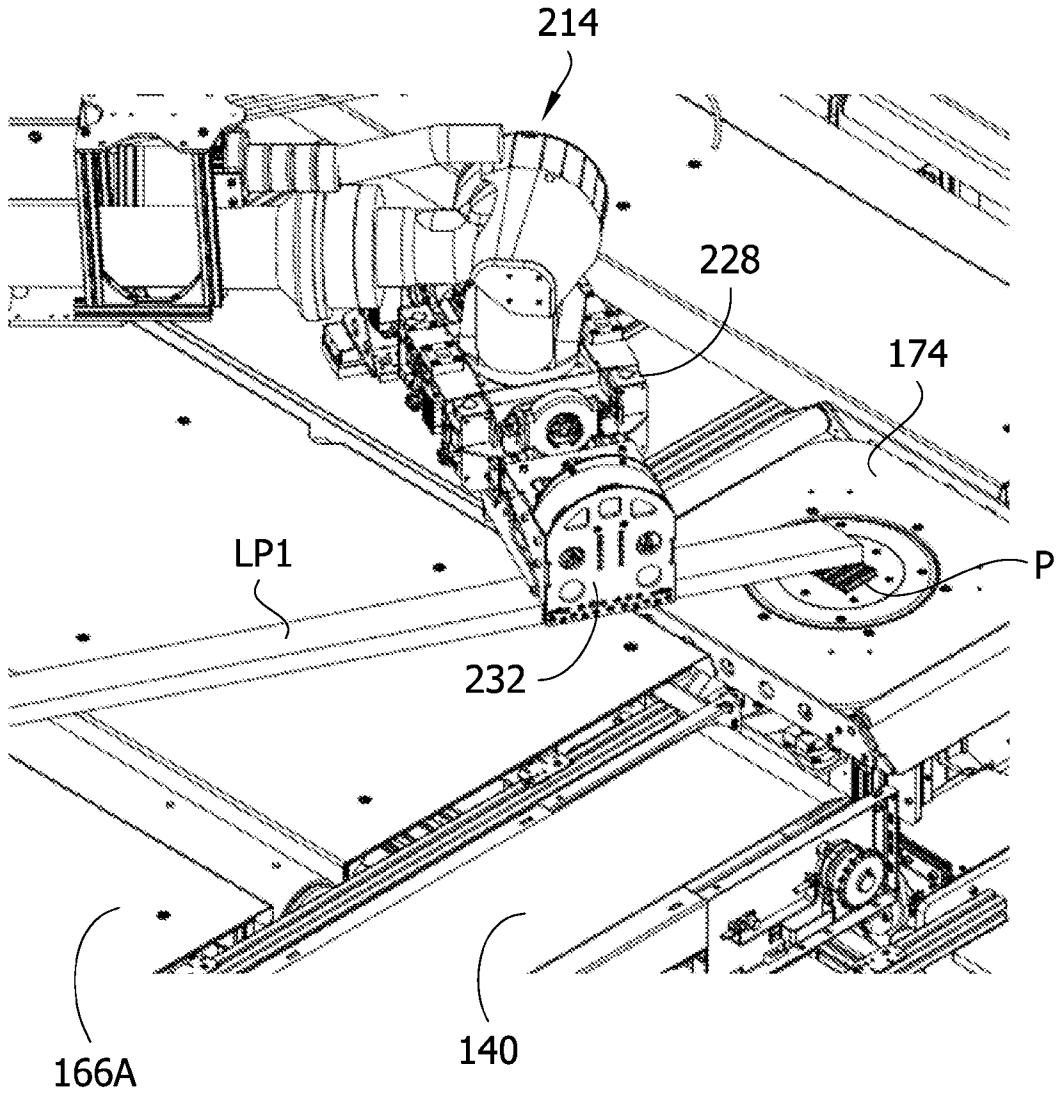


FIG. 42

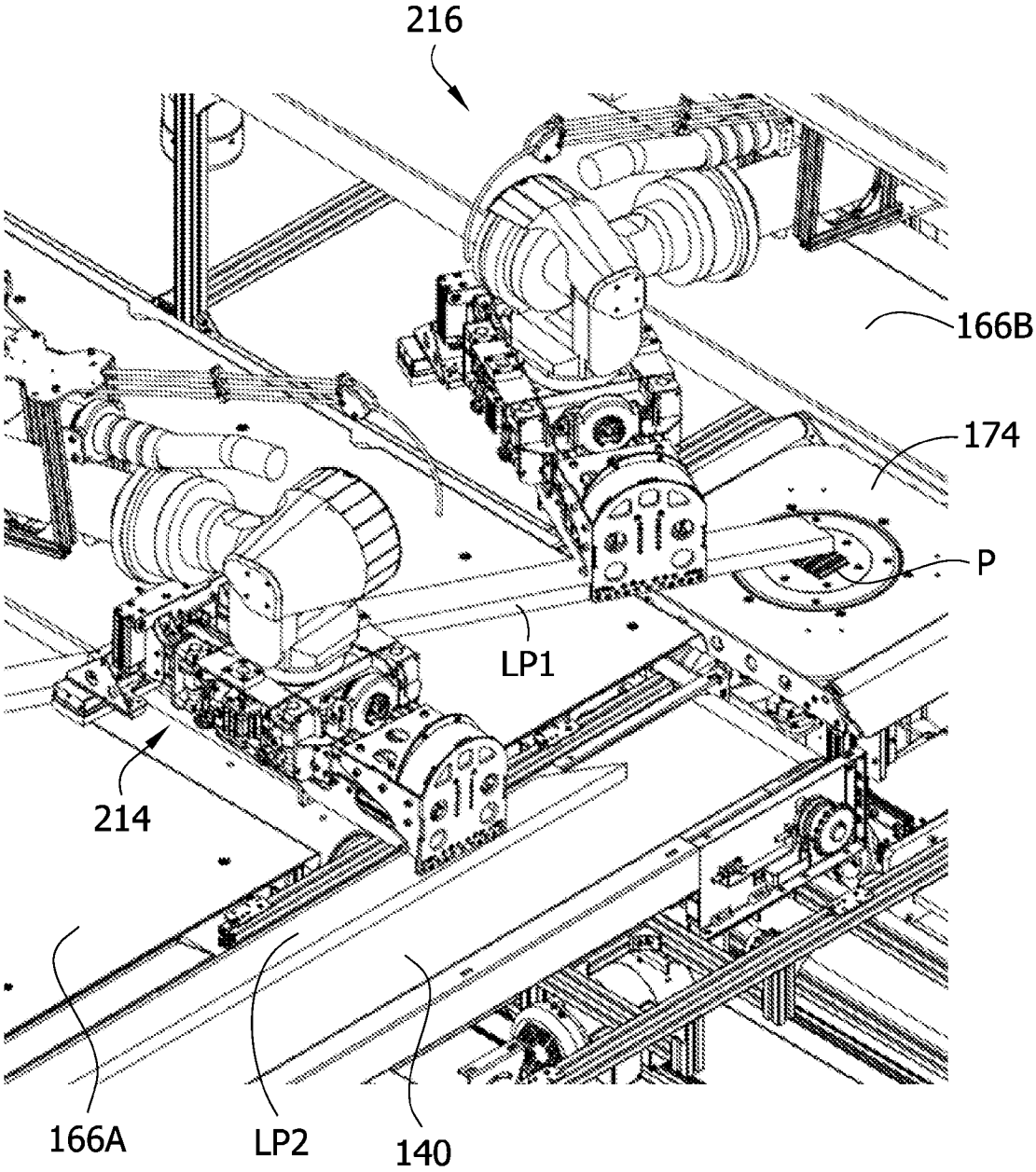


FIG. 43

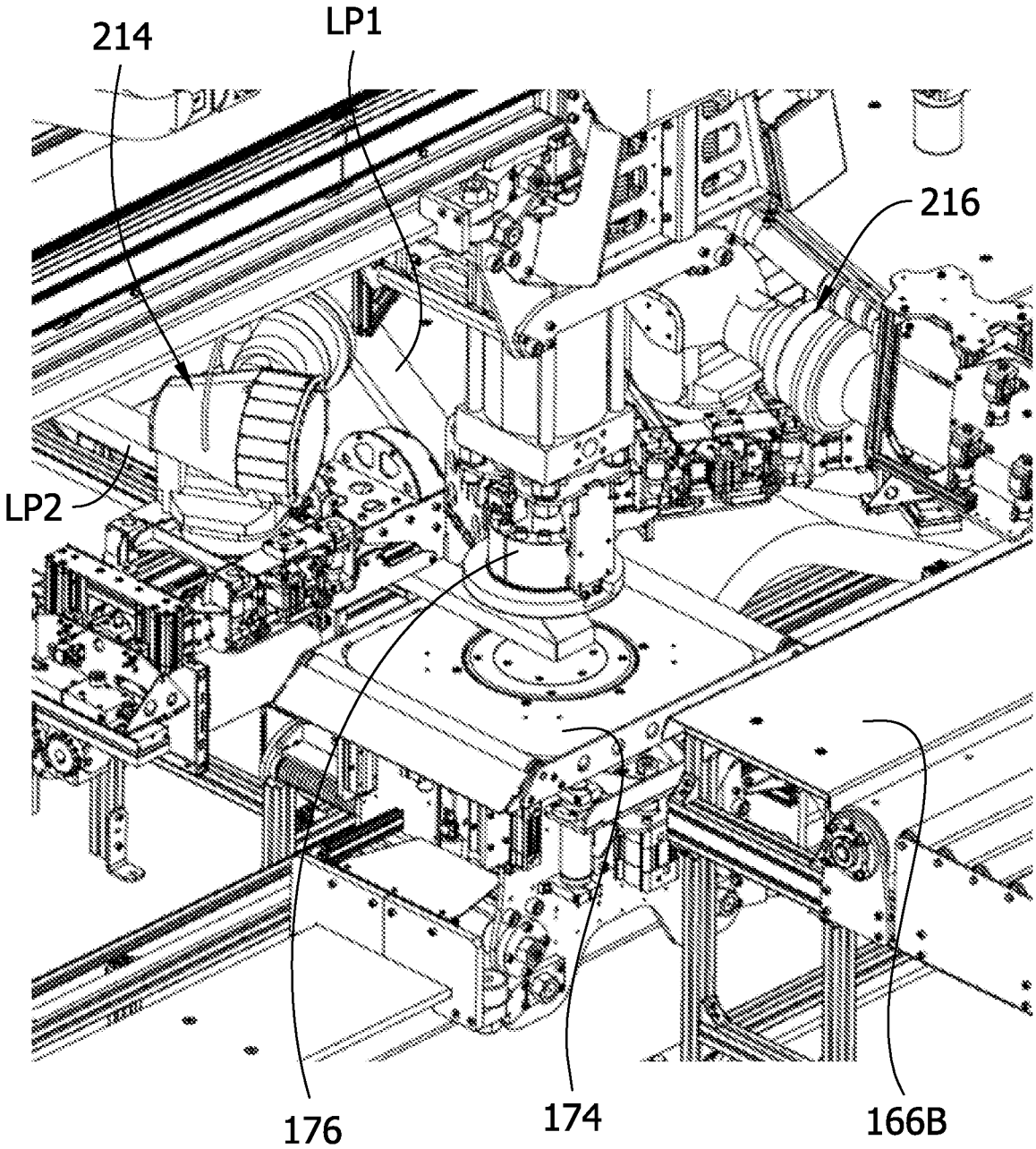


FIG. 44

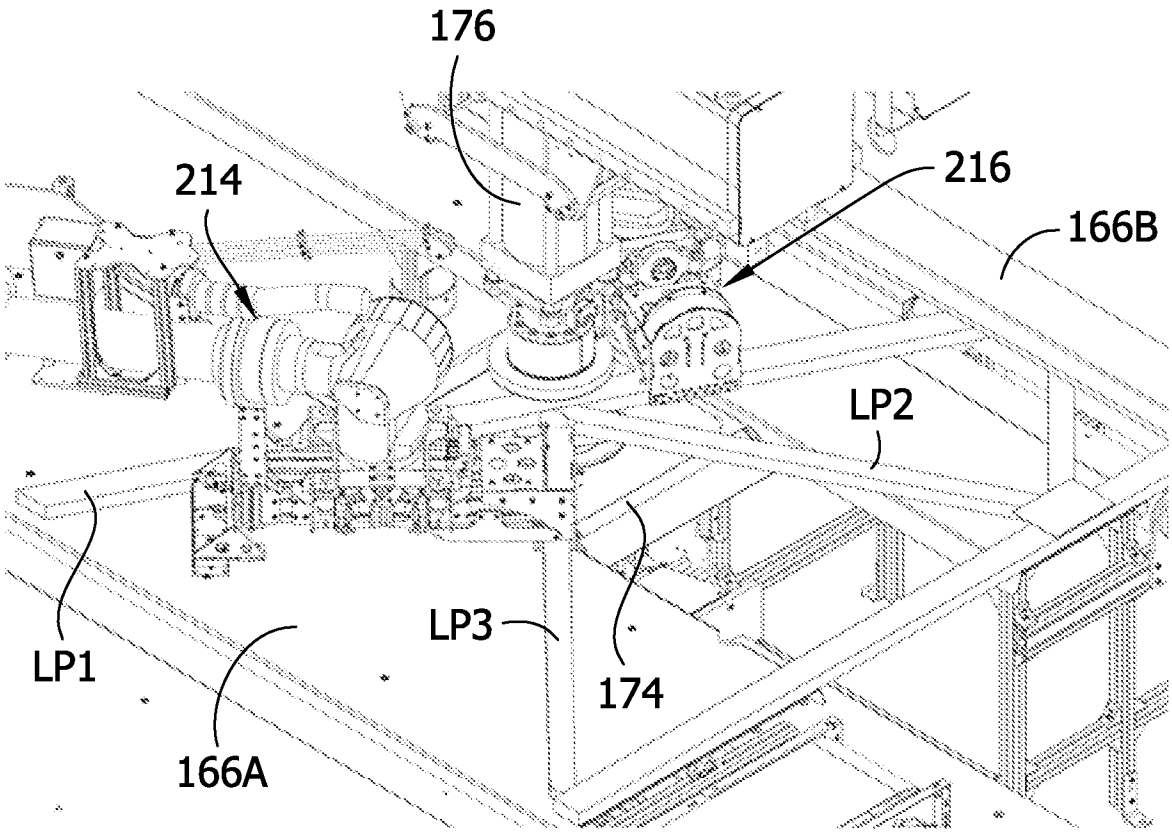


FIG. 45

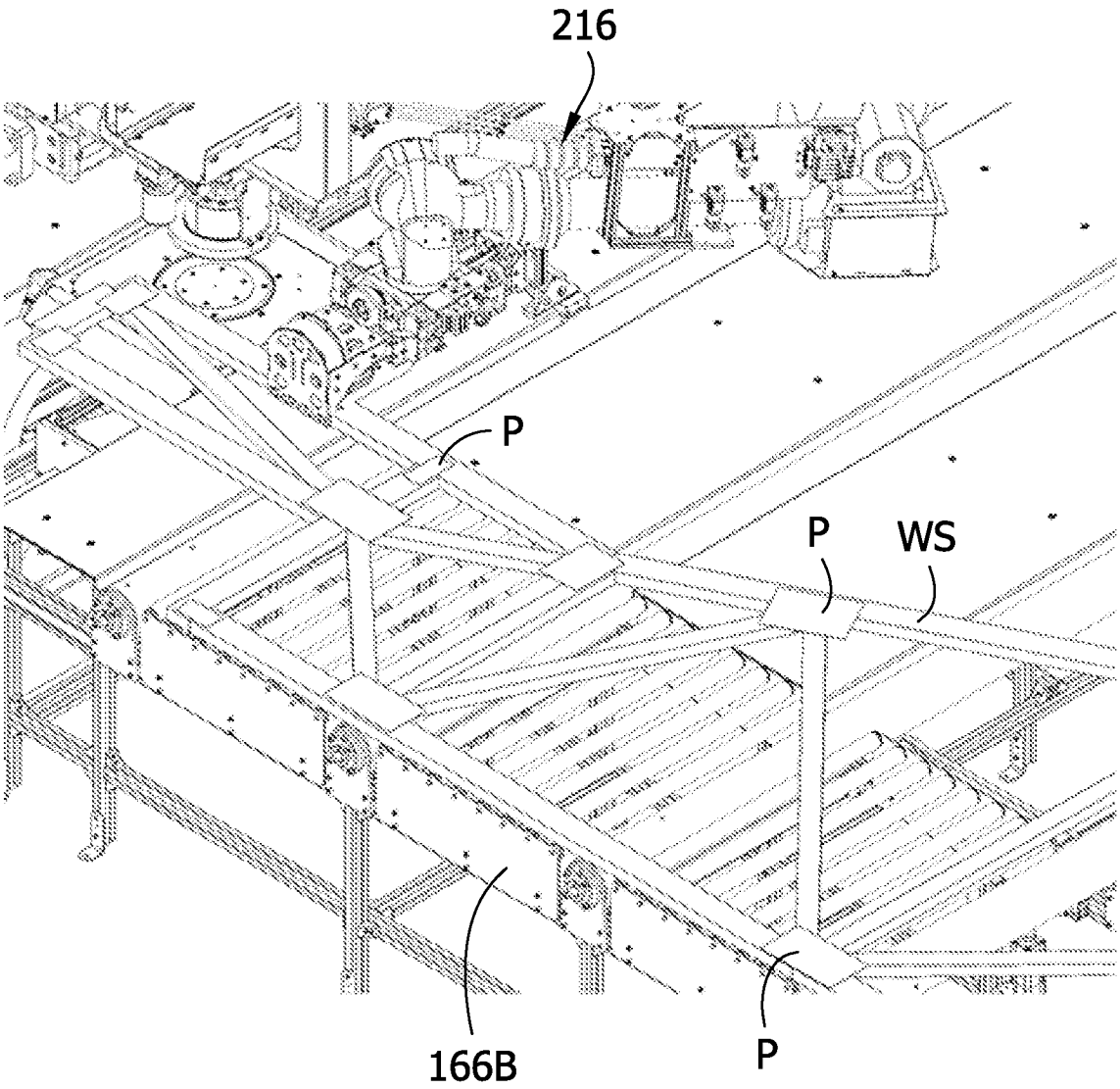


FIG. 46

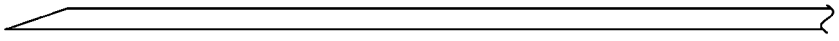


FIG. 47

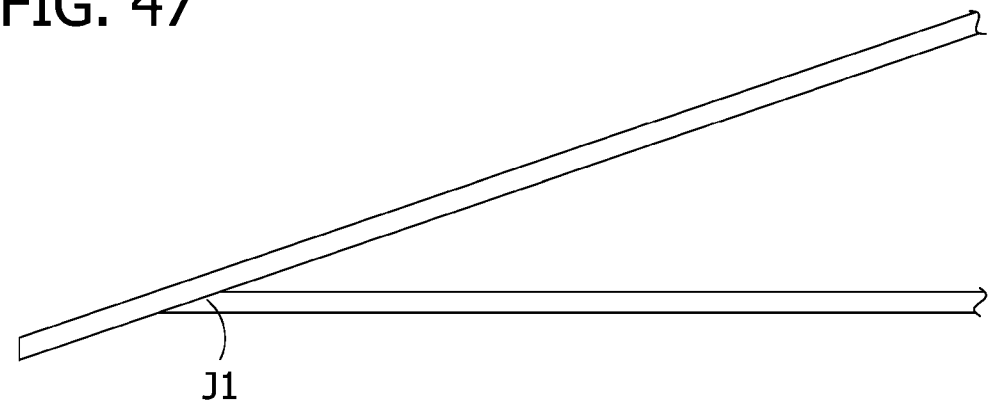



FIG. 48

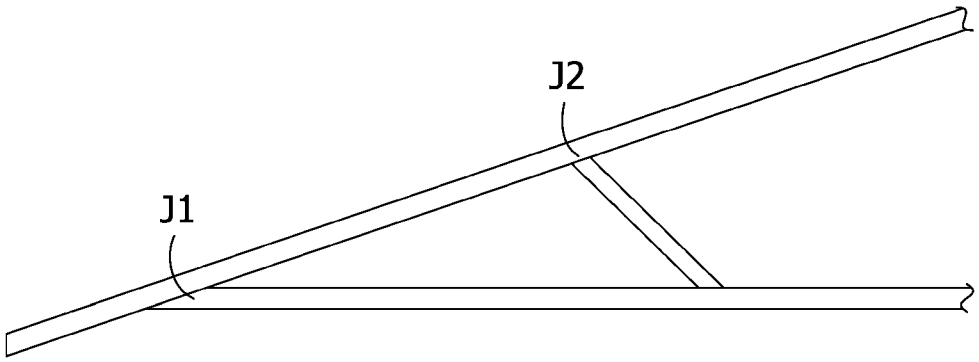


FIG. 49

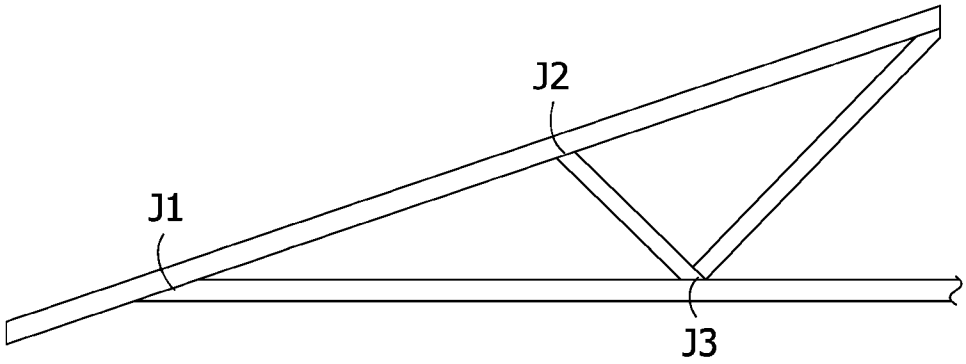



FIG. 50

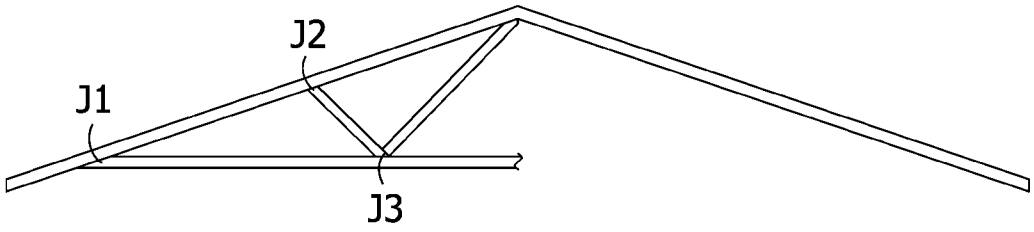



FIG. 51

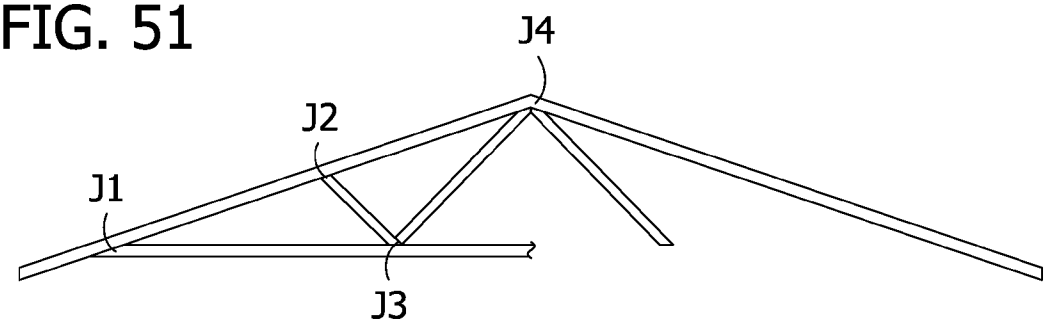


FIG. 52

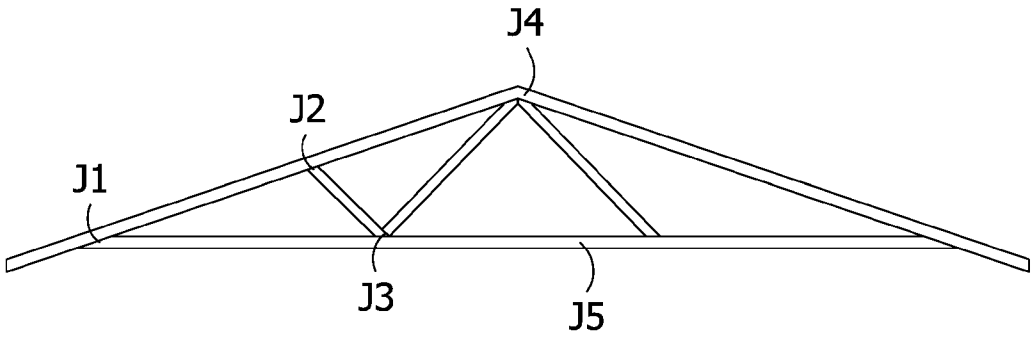



FIG. 53

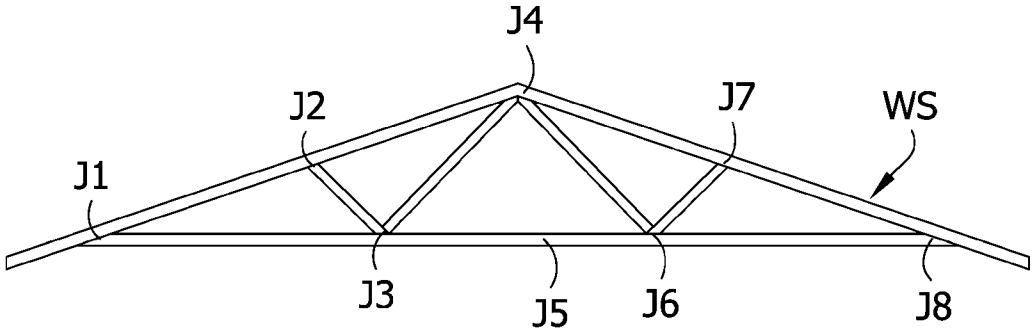


FIG. 54

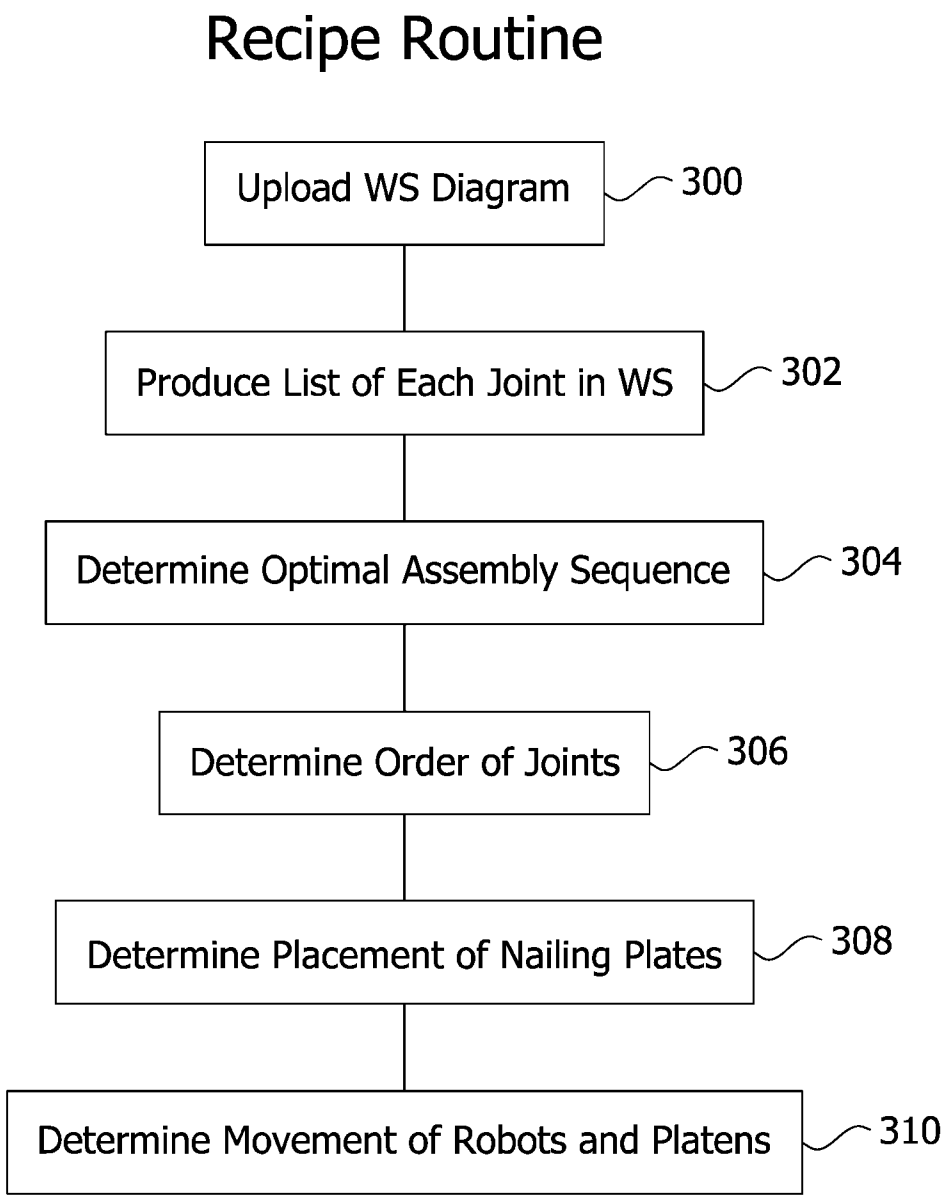

Recipe Routine
Upload WS Diagram
300
Produce List of Each Joint in WS
302
Determine Optimal Assembly Sequence
304
Determine Order of Joints
306
Determine Placement of Nailing Plates
308
Determine Movement of Robots and Platens
310

FIG. 55

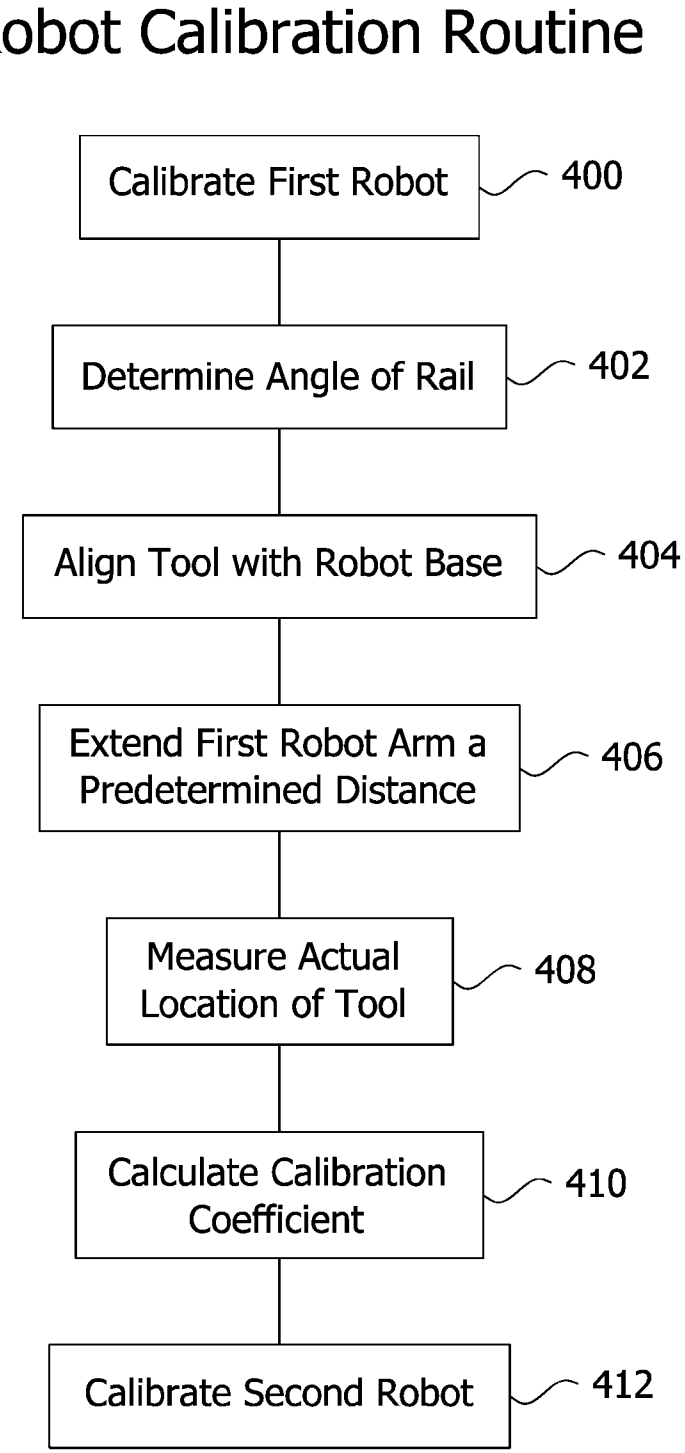

Robot Calibration Routine
Calibrate First Robot
400
Determine Angle of Rail
402
Align Tool with Robot Base
404
Extend First Robot Arm a Predetermined Distance
406
Measure Actual Location of Tool
408
Calculate Calibration Coefficient
410
Calibrate Second Robot
412

FIG. 56

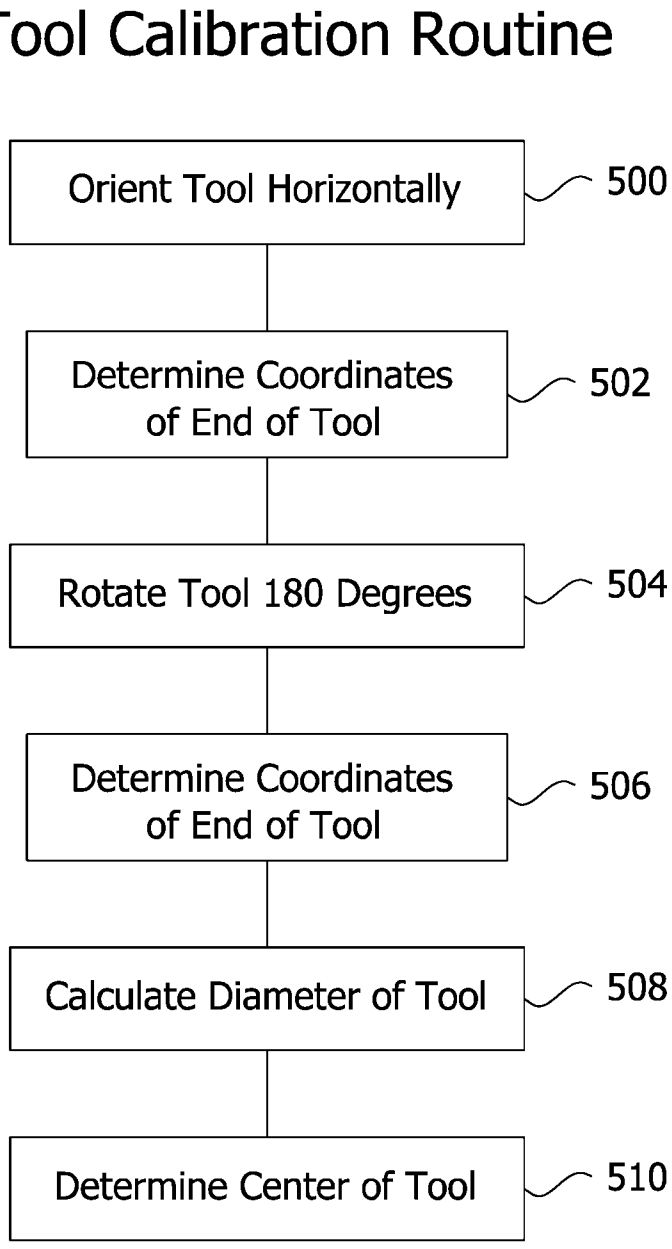
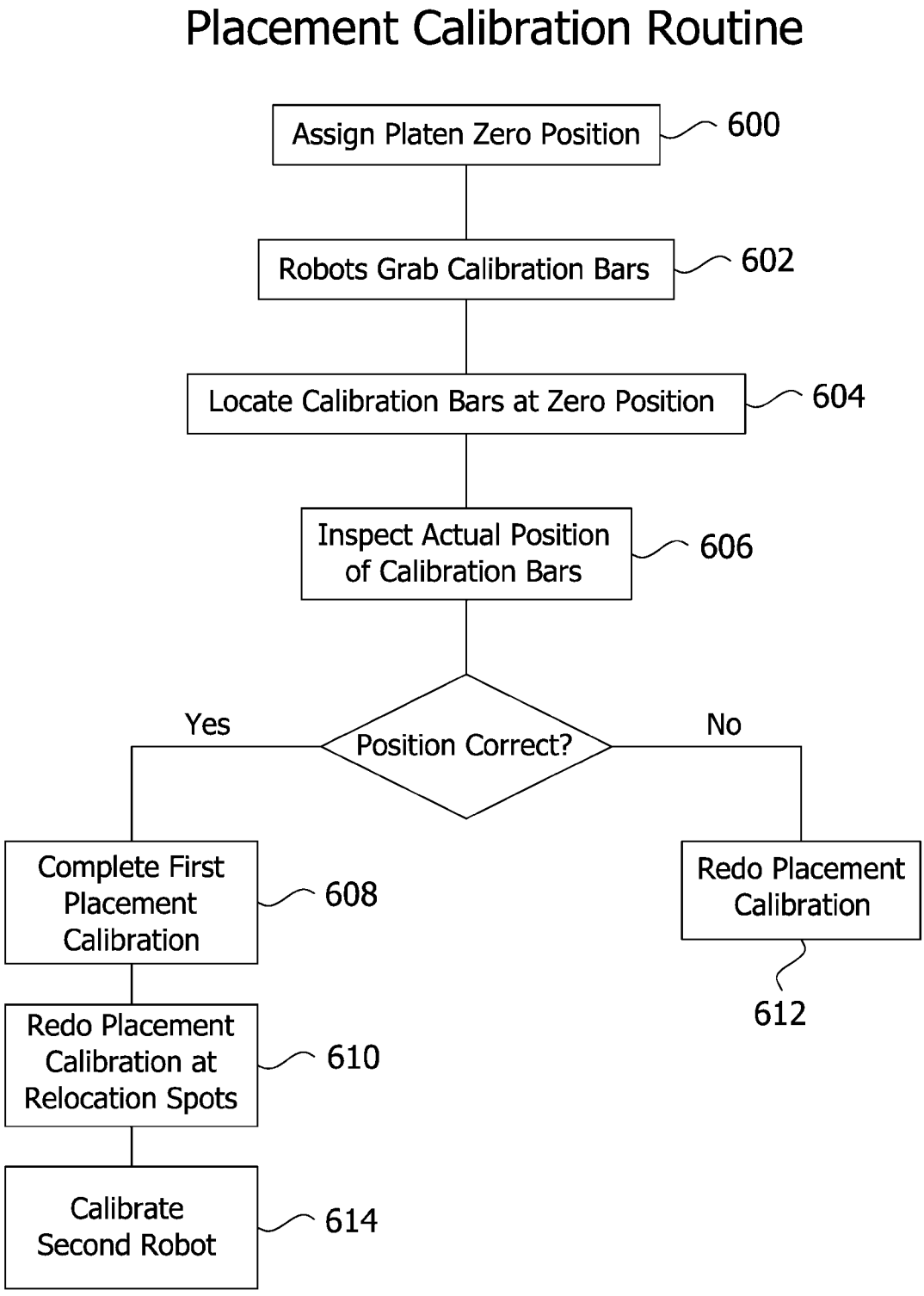

Tool Calibration Routine
Orient Tool Horizontally
500
Determine Coordinates of End of Tool
502
Rotate Tool 180 Degrees
504
Determine Coordinates of End of Tool
506
Calculate Diameter of Tool
508
Determine Center of Tool
510

FIG. 57


Placement Calibration Routine
Assign Platen Zero Position
600
Robots Grab Calibration Bars
602
Locate Calibration Bars at Zero Position
604
Inspect Actual Position of Calibration Bars
606
Position Correct?
Yes
No
Complete First Placement Calibration
608
Redo Placement Calibration at Relocation Spots
610
Calibrate Second Robot
614
Redo Placement Calibration
612

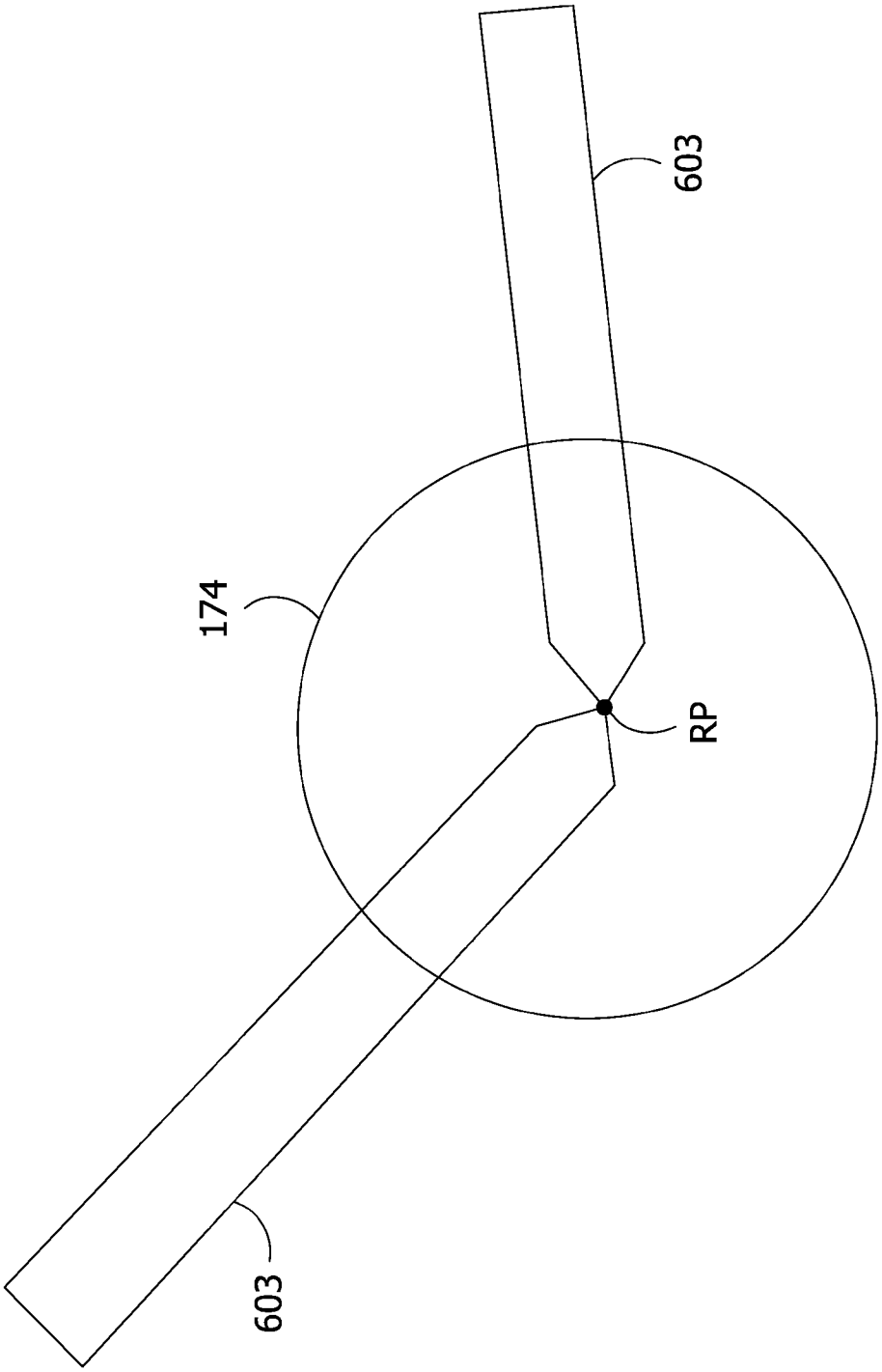



FIG. 58

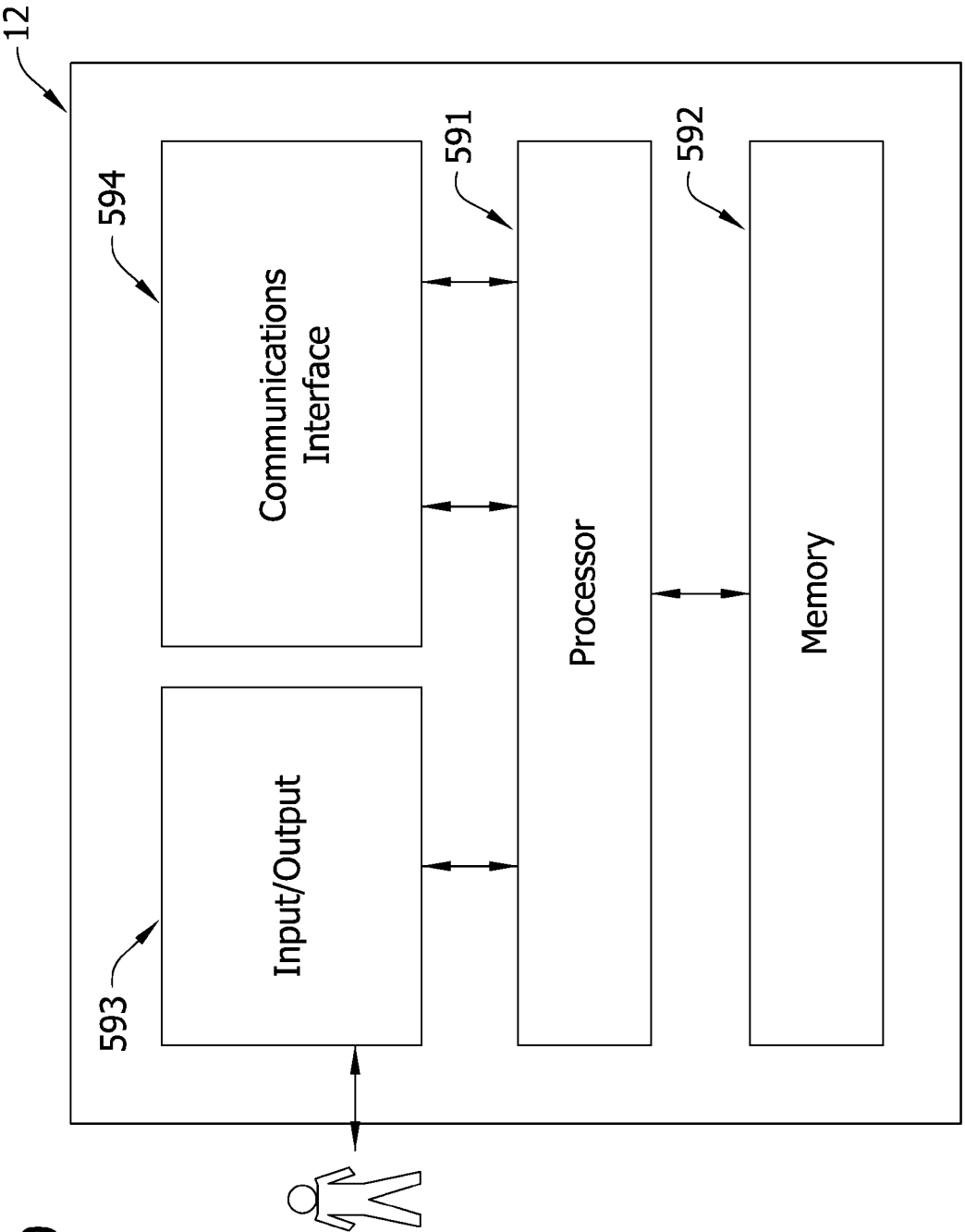

12
594
Communications Interface
591
592
Processor
Memory
Input/Output
593

FIG. 59

FIG. 60

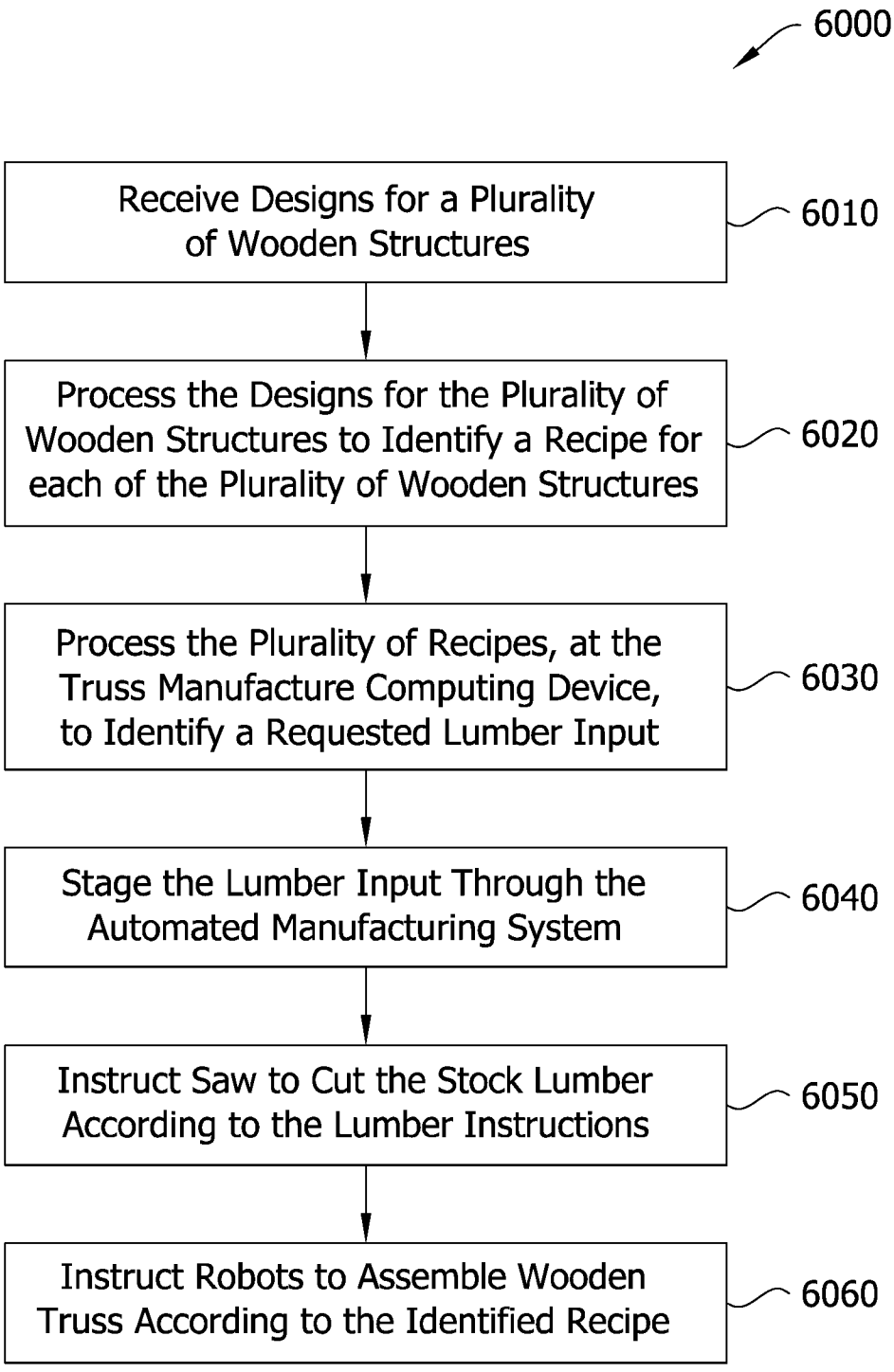
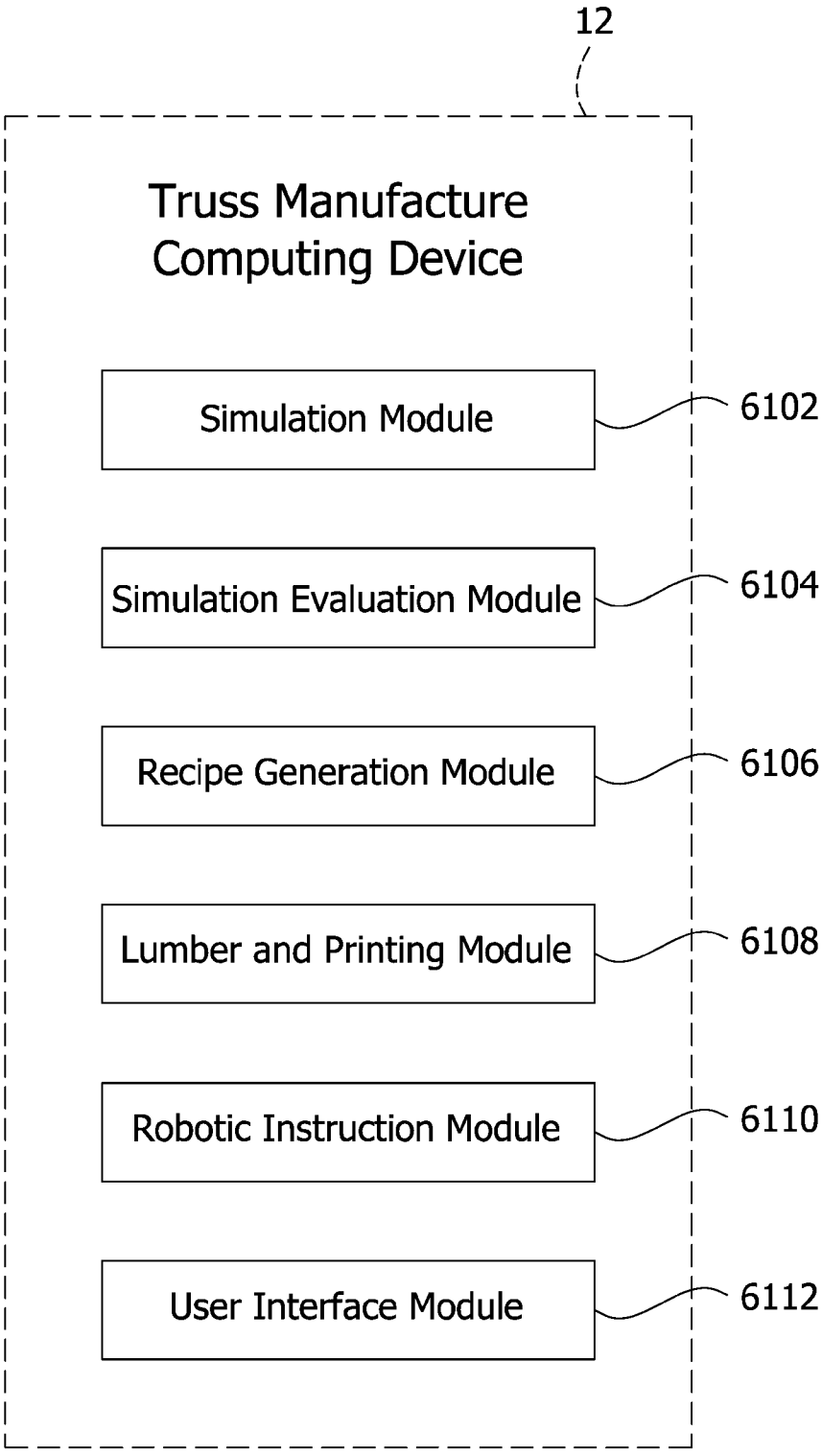

6000
Receive Designs for a Plurality of Wooden Structures
6010
Process the Designs for the Plurality of Wooden Structures to Identify a Recipe for each of the Plurality of Wooden Structures
6020
Process the Plurality of Recipes, at the Truss Manufacture Computing Device, to Identify a Requested Lumber Input
6030
Stage the Lumber Input Through the Automated Manufacturing System
6040
Instruct Saw to Cut the Stock Lumber According to the Lumber Instructions
6050
Instruct Robots to Assemble Wooden Truss According to the Identified Recipe
6060

FIG. 61


12
Truss Manufacture Computing Device
Simulation Module
6102
Simulation Evaluation Module
6104
Recipe Generation Module
6106
Lumber and Printing Module
6108
Robotic Instruction Module
6110
User Interface Module
6112

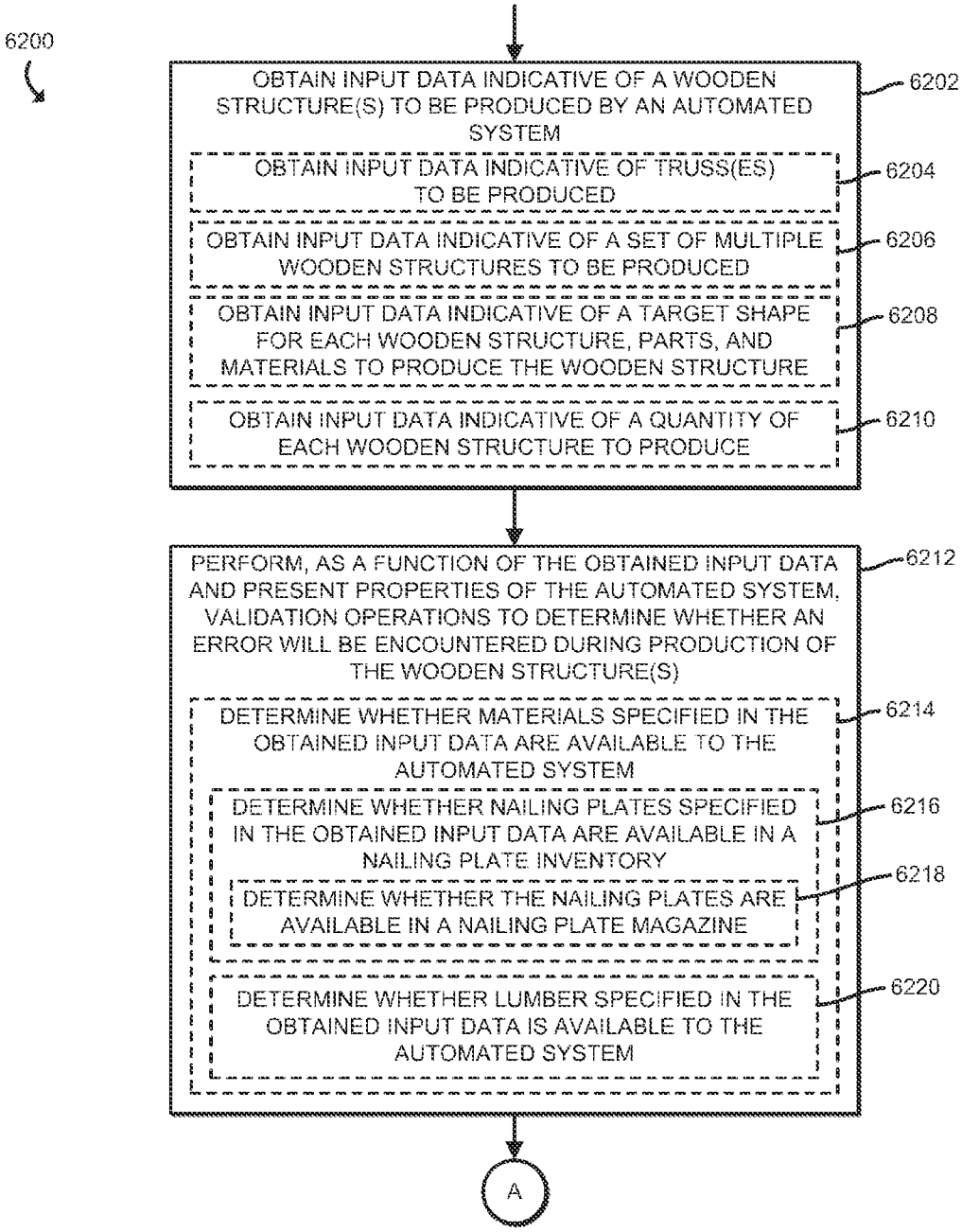

6200
OBTAIN INPUT DATA INDICATIVE OF A WOODEN STRUCTURE(S) TO BE PRODUCED BY AN AUTOMATED SYSTEM
6202
OBTAIN INPUT DATA INDICATIVE OF TRUSS(ES) TO BE PRODUCED
6204
OBTAIN INPUT DATA INDICATIVE OF A SET OF MULTIPLE WOODEN STRUCTURES TO BE PRODUCED
6206
OBTAIN INPUT DATA INDICATIVE OF A TARGET SHAPE FOR EACH WOODEN STRUCTURE, PARTS, AND MATERIALS TO PRODUCE THE WOODEN STRUCTURE
6208
OBTAIN INPUT DATA INDICATIVE OF A QUANTITY OF EACH WOODEN STRUCTURE TO PRODUCE
6210
PERFORM, AS A FUNCTION OF THE OBTAINED INPUT DATA AND PRESENT PROPERTIES OF THE AUTOMATED SYSTEM, VALIDATION OPERATIONS TO DETERMINE WHETHER AN ERROR WILL BE ENCOUNTERED DURING PRODUCTION OF THE WOODEN STRUCTURE(S)
6212
DETERMINE WHETHER MATERIALS SPECIFIED IN THE OBTAINED INPUT DATA ARE AVAILABLE TO THE AUTOMATED SYSTEM
6214
DETERMINE WHETHER NAILING PLATES SPECIFIED IN THE OBTAINED INPUT DATA ARE AVAILABLE IN A NAILING PLATE INVENTORY
6216
DETERMINE WHETHER THE NAILING PLATES ARE AVAILABLE IN A NAILING PLATE MAGAZINE
6218
DETERMINE WHETHER LUMBER SPECIFIED IN THE OBTAINED INPUT DATA IS AVAILABLE TO THE AUTOMATED SYSTEM
6220
A

FIG. 62

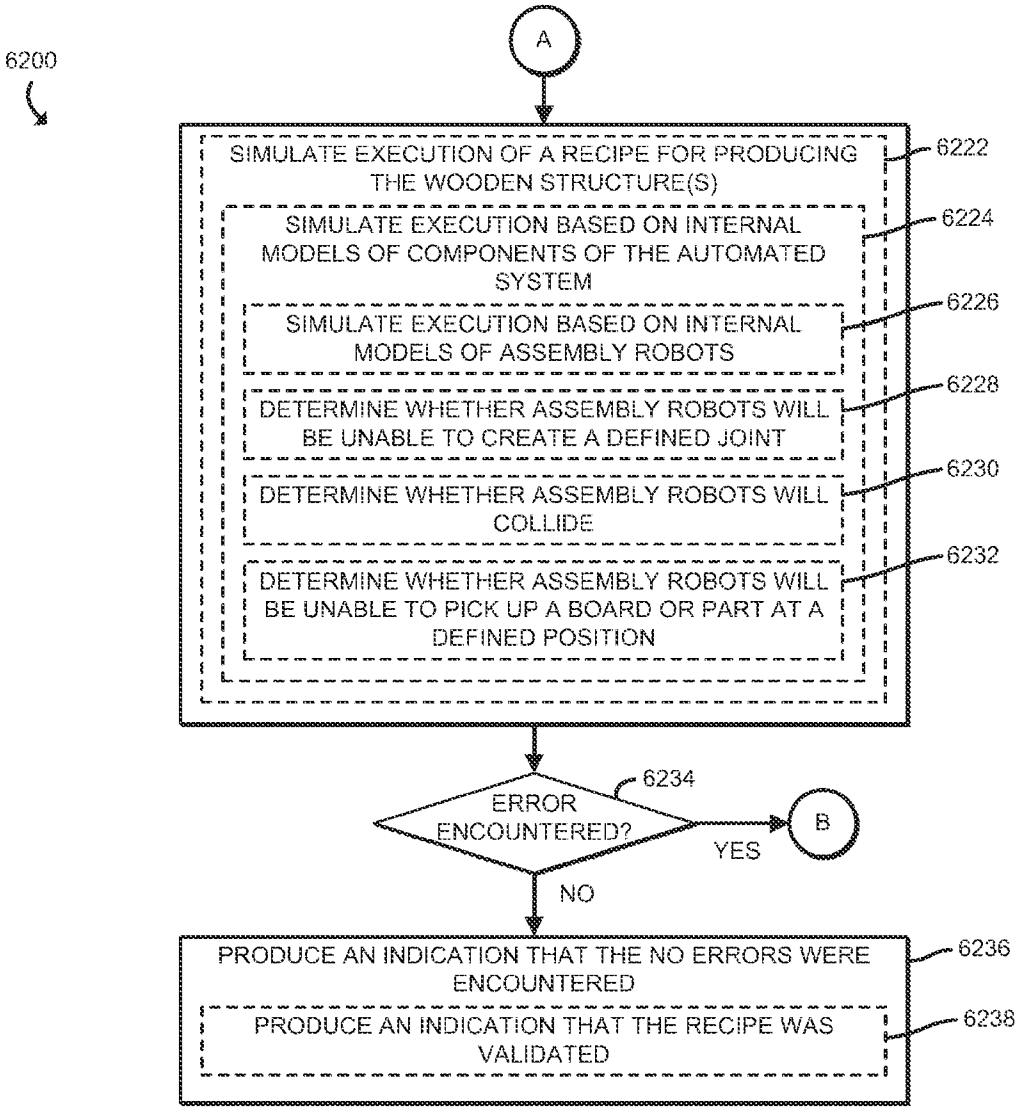

6200
A
SIMULATE EXECUTION OF A RECIPE FOR PRODUCING THE WOODEN STRUCTURE(S)
6222
SIMULATE EXECUTION BASED ON INTERNAL MODELS OF COMPONENTS OF THE AUTOMATED SYSTEM
6224
SIMULATE EXECUTION BASED ON INTERNAL MODELS OF ASSEMBLY ROBOTS
6226
DETERMINE WHETHER ASSEMBLY ROBOTS WILL BE UNABLE TO CREATE A DEFINED JOINT
6228
DETERMINE WHETHER ASSEMBLY ROBOTS WILL COLLIDE
6230
DETERMINE WHETHER ASSEMBLY ROBOTS WILL BE UNABLE TO PICK UP A BOARD OR PART AT A DEFINED POSITION
6232
ERROR ENCOUNTERED?
6234
YES
B
NO
PRODUCE AN INDICATION THAT THE NO ERRORS WERE ENCOUNTERED
6236
PRODUCE AN INDICATION THAT THE RECIPE WAS VALIDATED
6238

FIG. 63

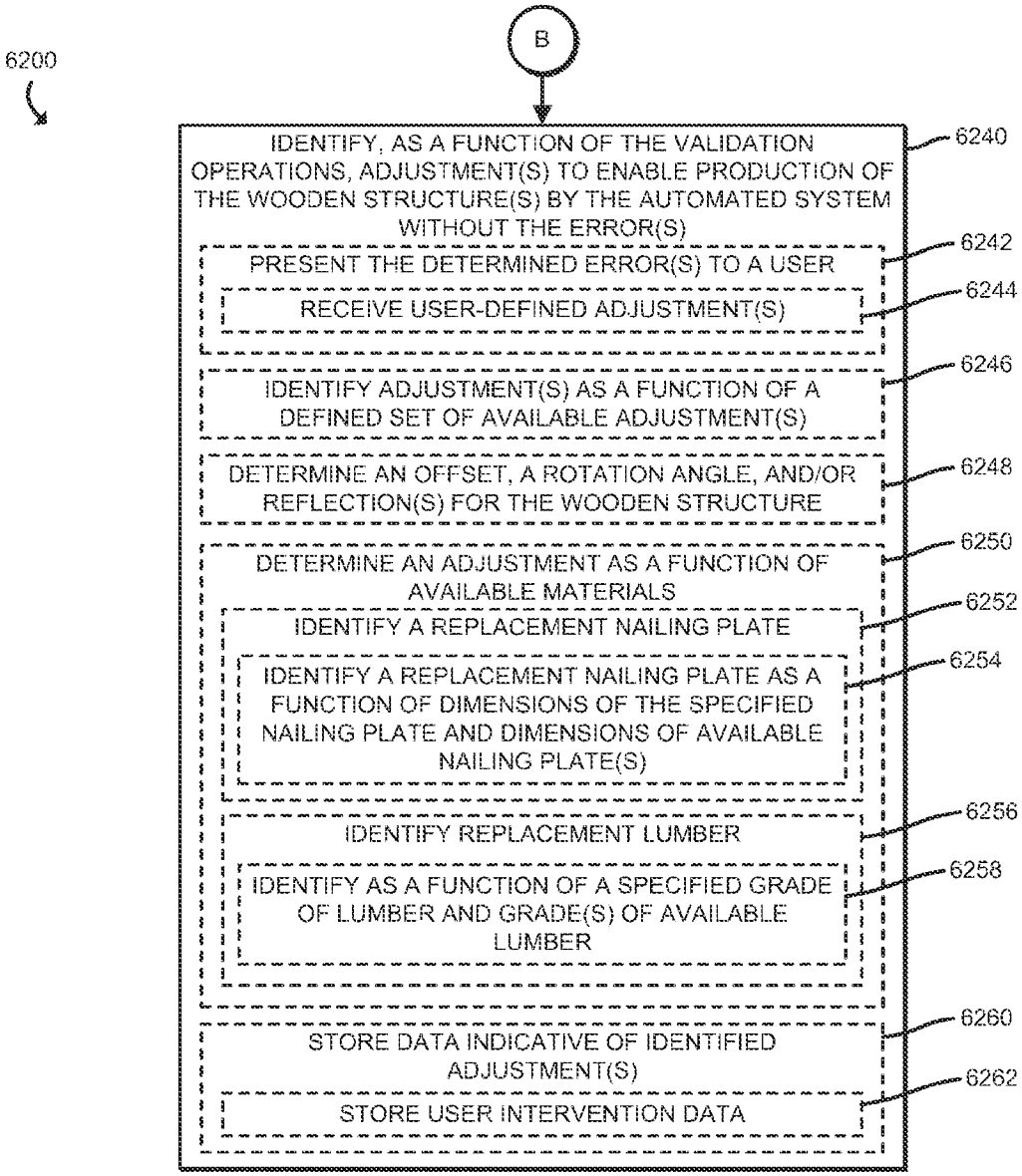

6200
B
IDENTIFY, AS A FUNCTION OF THE VALIDATION OPERATIONS, ADJUSTMENT(S) TO ENABLE PRODUCTION OF THE WOODEN STRUCTURE(S) BY THE AUTOMATED SYSTEM WITHOUT THE ERROR(S)
6240
PRESENT THE DETERMINED ERROR(S) TO A USER
6242
RECEIVE USER-DEFINED ADJUSTMENT(S)
6244
IDENTIFY ADJUSTMENT(S) AS A FUNCTION OF A DEFINED SET OF AVAILABLE ADJUSTMENT(S)
6246
DETERMINE AN OFFSET, A ROTATION ANGLE, AND/OR REFLECTION(S) FOR THE WOODEN STRUCTURE
6248
DETERMINE AN ADJUSTMENT AS A FUNCTION OF AVAILABLE MATERIALS
6250
IDENTIFY A REPLACEMENT NAILING PLATE
6252
IDENTIFY A REPLACEMENT NAILING PLATE AS A FUNCTION OF DIMENSIONS OF THE SPECIFIED NAILING PLATE AND DIMENSIONS OF AVAILABLE NAILING PLATE(S)
6254
IDENTIFY REPLACEMENT LUMBER
6256
IDENTIFY AS A FUNCTION OF A SPECIFIED GRADE OF LUMBER AND GRADE(S) OF AVAILABLE LUMBER
6258
STORE DATA INDICATIVE OF IDENTIFIED ADJUSTMENT(S)
6260
STORE USER INTERVENTION DATA
6262

FIG. 64

6500
BUILD SEQUENCES
SHAPE
LABEL
SEQUENCE
QUANTITY
USER INTERVENTIONS
NO OF PARTS
NO OF PLATES
BUNDLE
RECIPE
6510
0
0
14
12
6512
0
0
12
12
6514
0
0
12
11
6516
0
0
12
10
VIEW RECIPE
EXPORT RECIPE
PLATES
6520
UNIQUE ID
DESCRIPTION
625
MT20 3X6
626
MT20 1.5X4
627
MT20 4X5
PARTS
6530
UNIQUE ID
DESCRIPTION
STOCK WIDTH
302
B1
3.5
303
T1
3.5
304
T2
3.5

FIG. 65

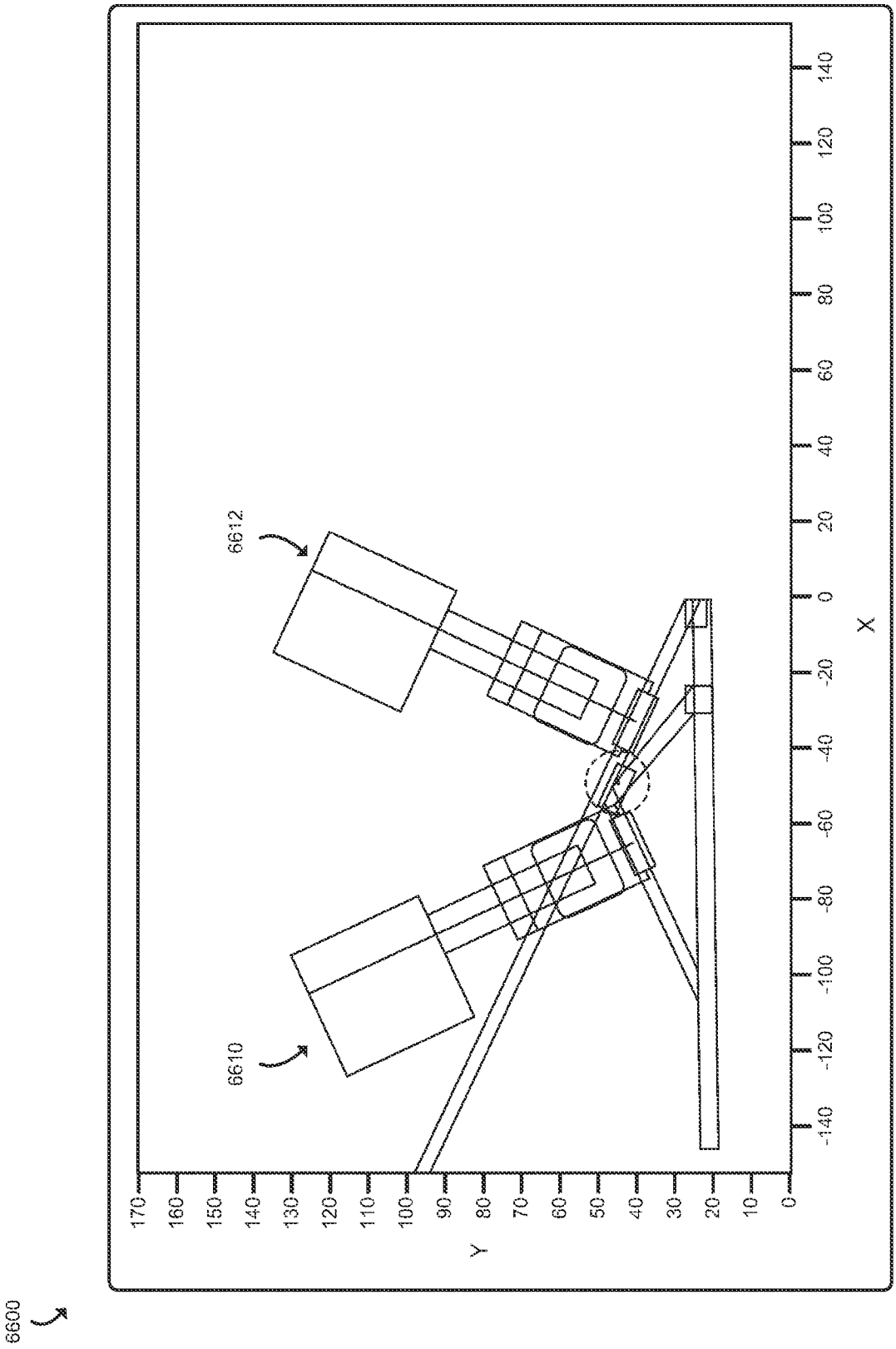

6600
6610
6612
Y
X
170
160
150
140
130
120
110
100
90
80
70
60
50
40
30
20
10
0
-140
-120
-100
-80
-60
-40
-20
0
20
40
60
80
100
120
140

FIG. 66

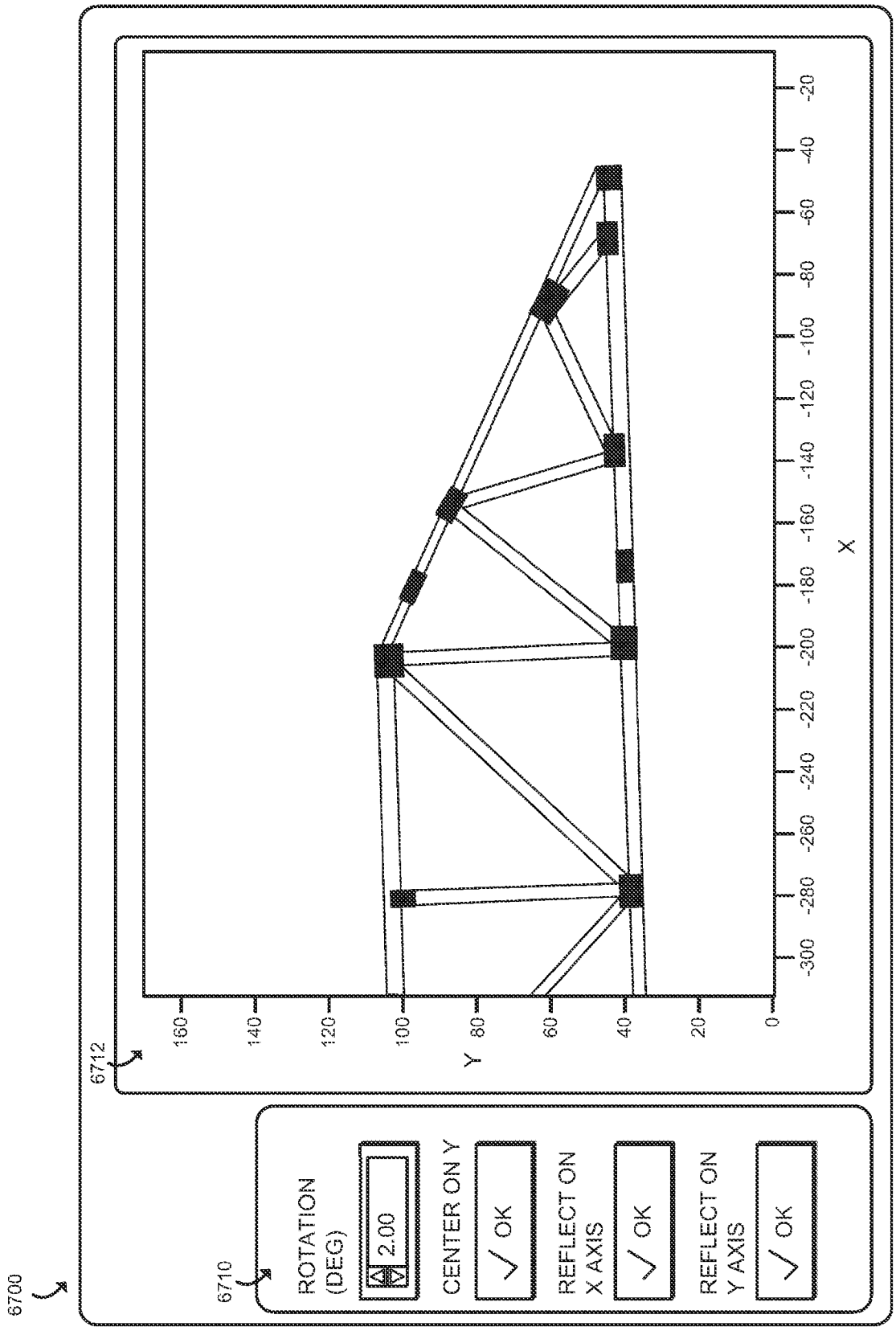

6700
6710
ROTATION (DEG)
2.00
CENTER ON Y
OK
REFLECT ON X AXIS
OK
REFLECT ON Y AXIS
OK
6712
Y
160
140
120
100
80
60
40
20
0
X
-20
-40
-60
-80
-100
-120
-140
-160
-180
-200
-220
-240
-260
-280
-300

FIG. 67

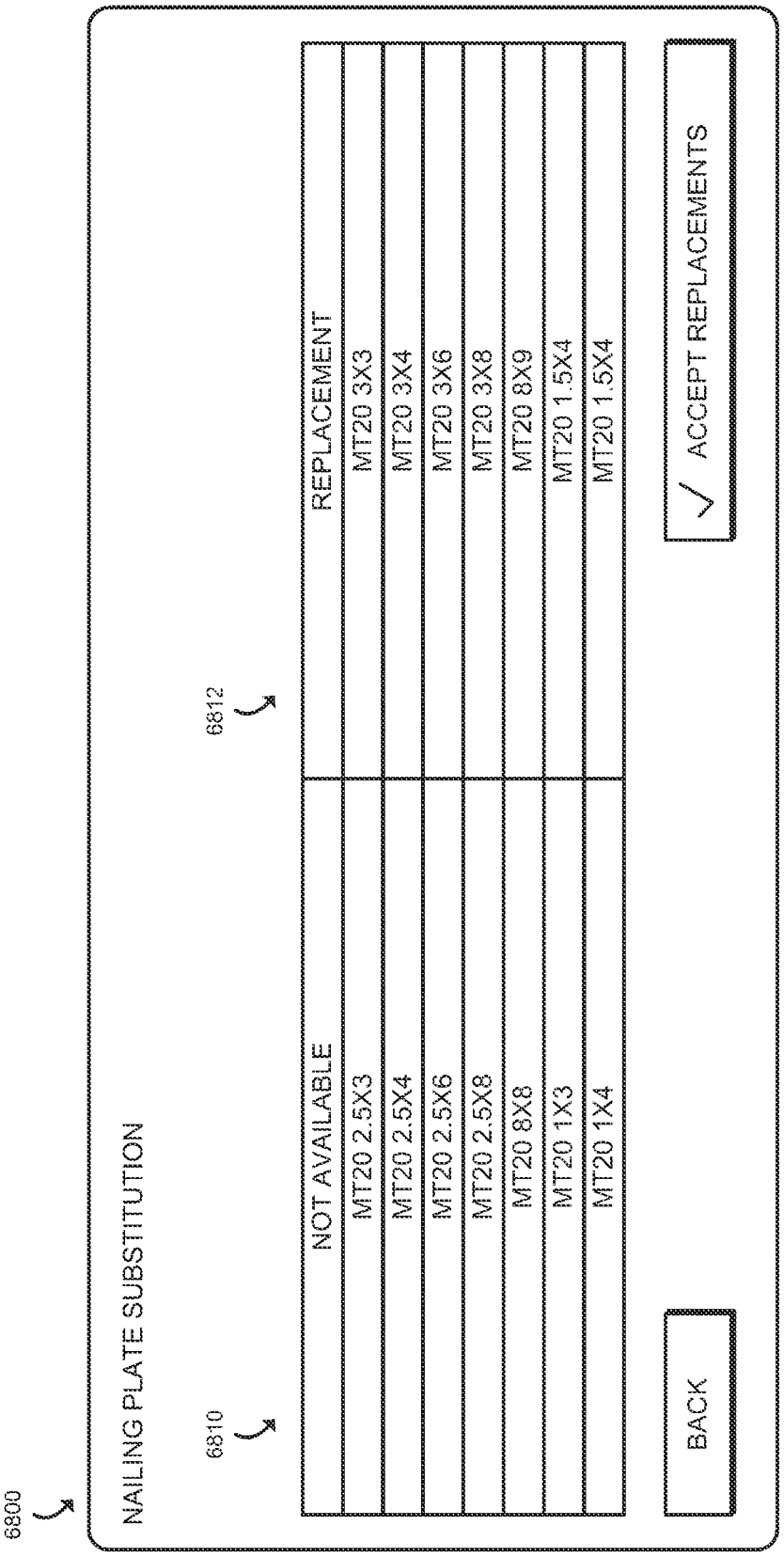

6800
NAILING PLATE SUBSTITUTION
6810
6812
NOT AVAILABLE
REPLACEMENT
MT20 2.5X3
MT20 3X3
MT20 2.5X4
MT20 3X4
MT20 2.5X6
MT20 3X6
MT20 2.5X8
MT20 3X8
MT20 8X8
MT20 8X9
MT20 1X3
MT20 1.5X4
MT20 1X4
MT20 1.5X4
BACK
✓ ACCEPT REPLACEMENTS

FIG. 68

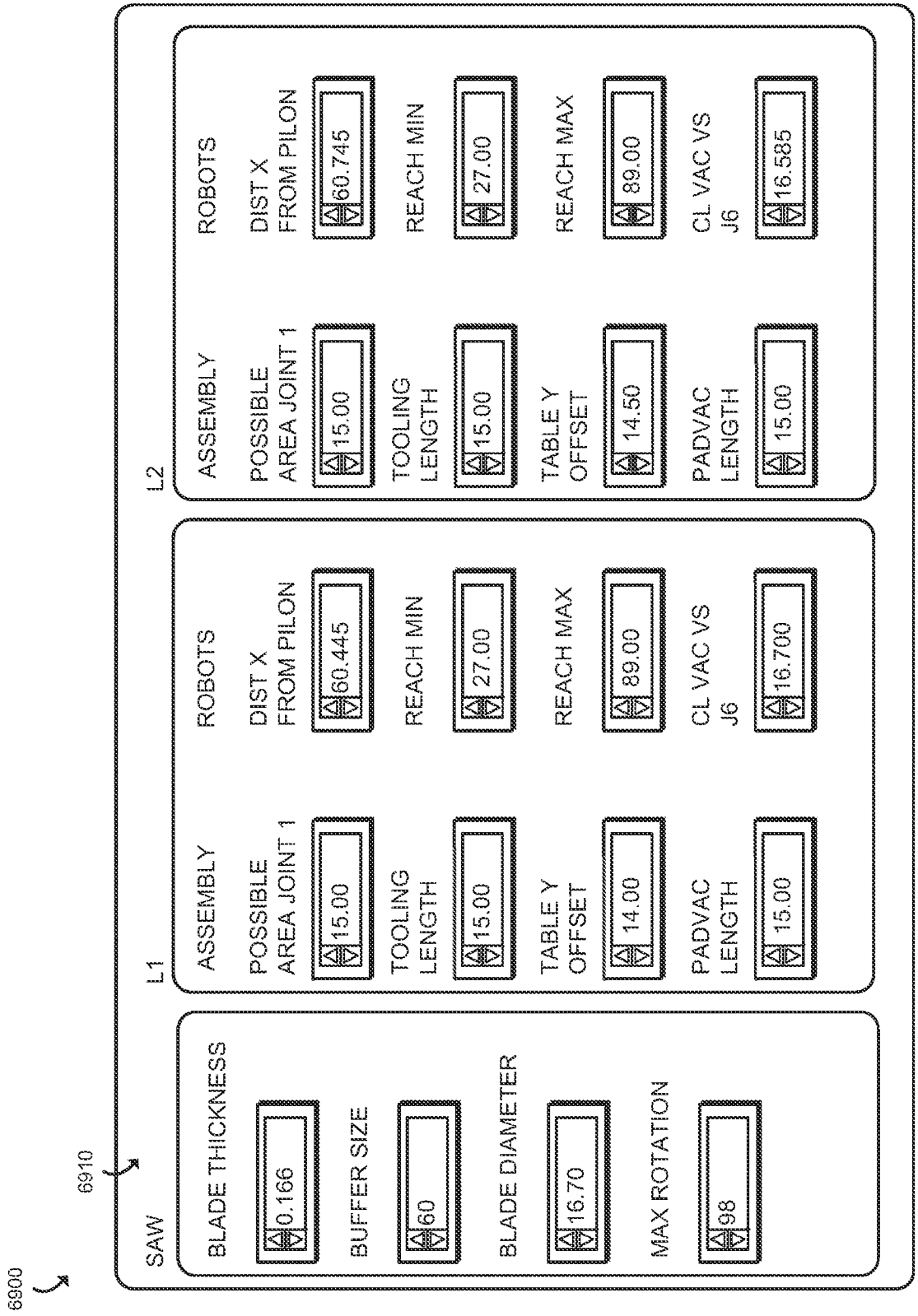

6900
6910
SAW
BLADE THICKNESS
0.166
BUFFER SIZE
60
BLADE DIAMETER
16.70
MAX ROTATION
98
L1
ASSEMBLY
POSSIBLE AREA JOINT 1
15.00
TOOLING LENGTH
15.00
TABLE Y OFFSET
14.00
PADVAC LENGTH
15.00
ROBOTS
DIST X FROM PILON
60.445
REACH MIN
27.00
REACH MAX
89.00
CL VAC VS J6
16.700
L2
ASSEMBLY
POSSIBLE AREA JOINT 1
15.00
TOOLING LENGTH
15.00
TABLE Y OFFSET
14.50
PADVAC LENGTH
15.00
ROBOTS
DIST X FROM PILON
60.745
REACH MIN
27.00
REACH MAX
89.00
CL VAC VS J6
16.585

FIG. 69

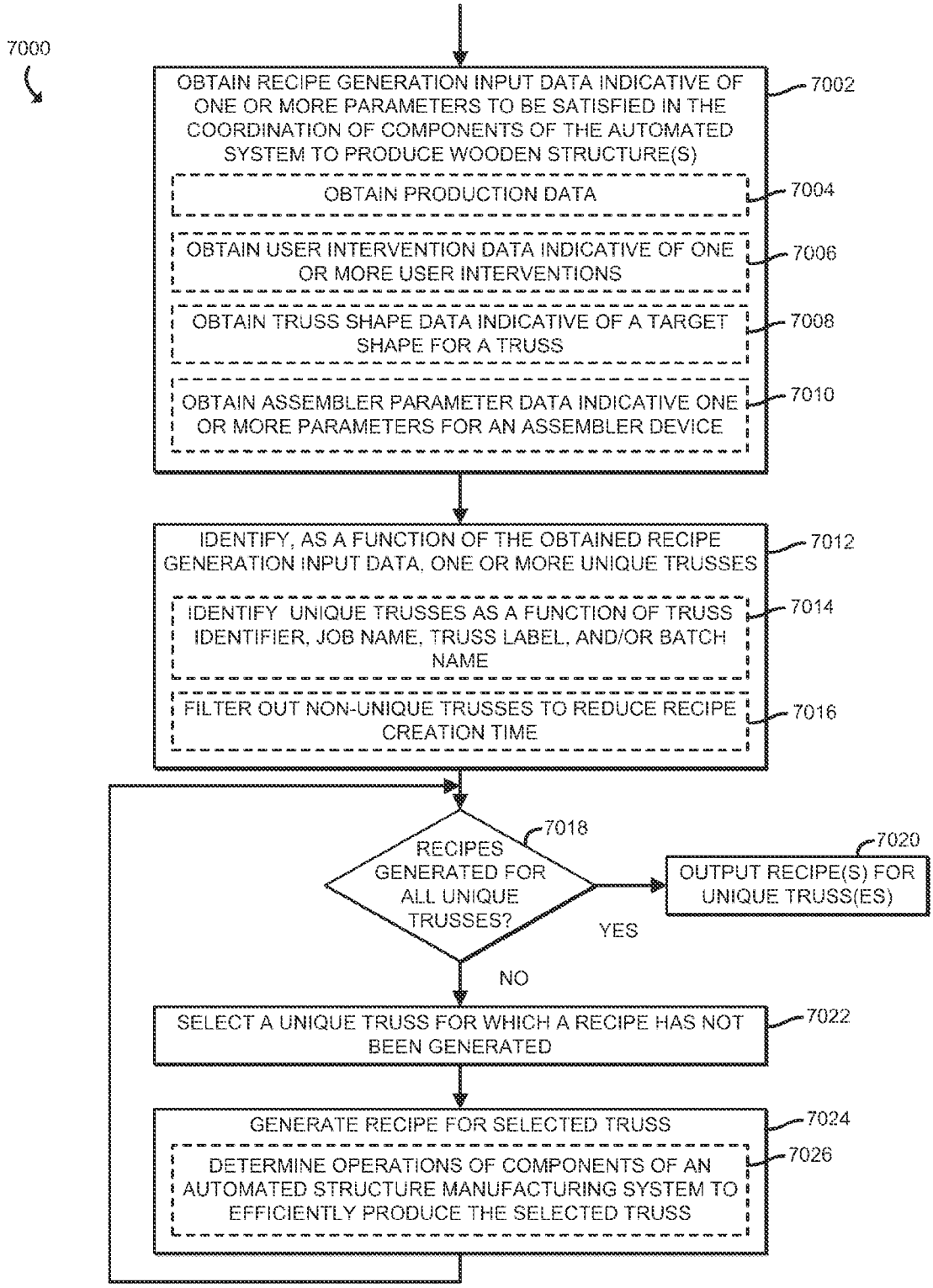

7000
OBTAIN RECIPE GENERATION INPUT DATA INDICATIVE OF ONE OR MORE PARAMETERS TO BE SATISFIED IN THE COORDINATION OF COMPONENTS OF THE AUTOMATED SYSTEM TO PRODUCE WOODEN STRUCTURE(S)
7002
OBTAIN PRODUCTION DATA
7004
OBTAIN USER INTERVENTION DATA INDICATIVE OF ONE OR MORE USER INTERVENTIONS
7006
OBTAIN TRUSS SHAPE DATA INDICATIVE OF A TARGET SHAPE FOR A TRUSS
7008
OBTAIN ASSEMBLER PARAMETER DATA INDICATIVE ONE OR MORE PARAMETERS FOR AN ASSEMBLER DEVICE
7010
IDENTIFY, AS A FUNCTION OF THE OBTAINED RECIPE GENERATION INPUT DATA, ONE OR MORE UNIQUE TRUSSES
7012
IDENTIFY UNIQUE TRUSSES AS A FUNCTION OF TRUSS IDENTIFIER, JOB NAME, TRUSS LABEL, AND/OR BATCH NAME
7014
FILTER OUT NON-UNIQUE TRUSSES TO REDUCE RECIPE CREATION TIME
7016
RECIPES GENERATED FOR ALL UNIQUE TRUSSES?
7018
YES
OUTPUT RECIPE(S) FOR UNIQUE TRUSS(ES)
7020
NO
SELECT A UNIQUE TRUSS FOR WHICH A RECIPE HAS NOT BEEN GENERATED
7022
GENERATE RECIPE FOR SELECTED TRUSS
7024
DETERMINE OPERATIONS OF COMPONENTS OF AN AUTOMATED STRUCTURE MANUFACTURING SYSTEM TO EFFICIENTLY PRODUCE THE SELECTED TRUSS
7026

FIG. 70

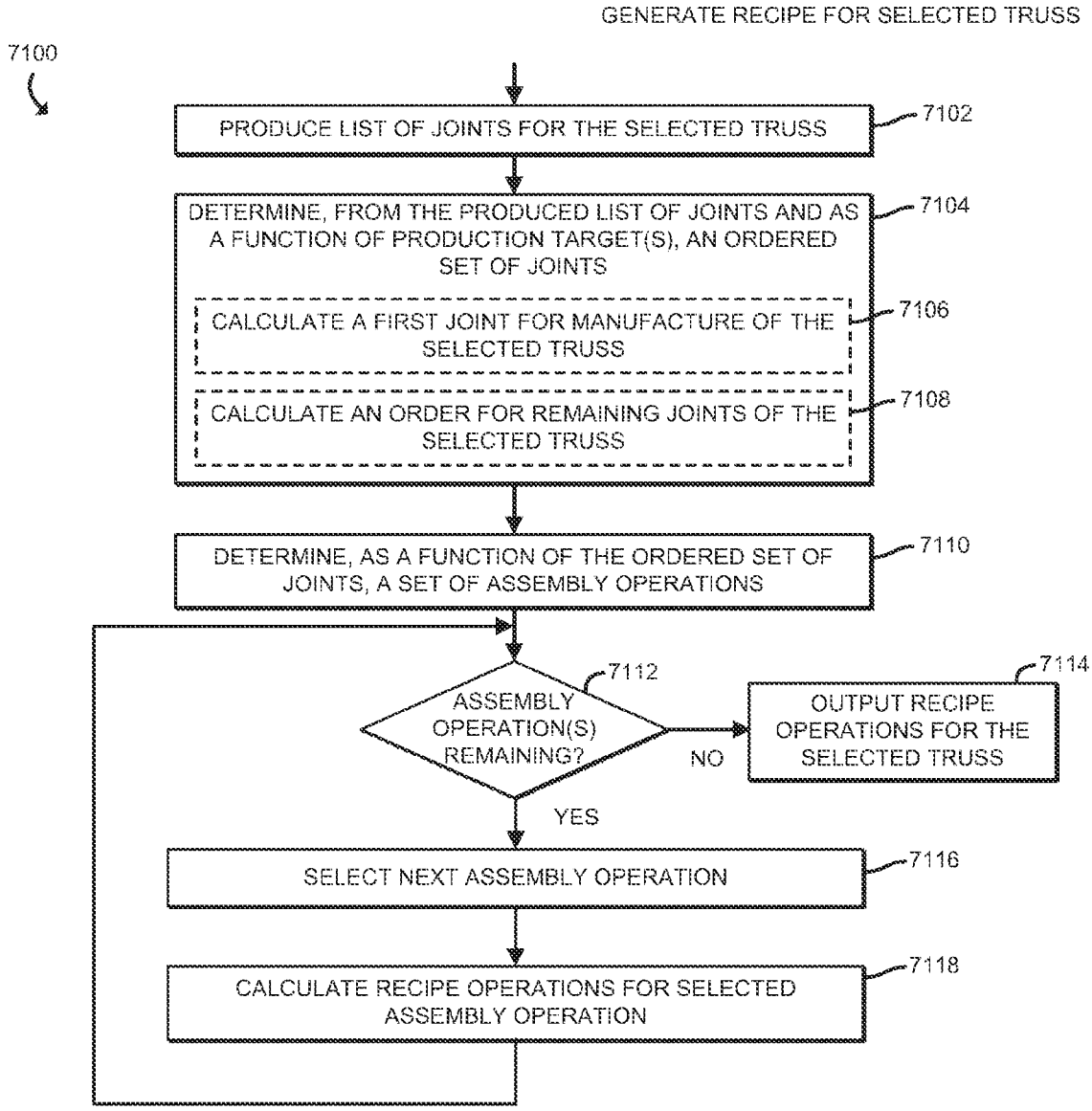

GENERATE RECIPE FOR SELECTED TRUSS
7100
PRODUCE LIST OF JOINTS FOR THE SELECTED TRUSS
7102
DETERMINE, FROM THE PRODUCED LIST OF JOINTS AND AS A FUNCTION OF PRODUCTION TARGET(S), AN ORDERED SET OF JOINTS
7104
CALCULATE A FIRST JOINT FOR MANUFACTURE OF THE SELECTED TRUSS
7106
CALCULATE AN ORDER FOR REMAINING JOINTS OF THE SELECTED TRUSS
7108
DETERMINE, AS A FUNCTION OF THE ORDERED SET OF JOINTS, A SET OF ASSEMBLY OPERATIONS
7110
ASSEMBLY OPERATION(S) REMAINING?
7112
NO
OUTPUT RECIPE OPERATIONS FOR THE SELECTED TRUSS
7114
YES
SELECT NEXT ASSEMBLY OPERATION
7116
CALCULATE RECIPE OPERATIONS FOR SELECTED ASSEMBLY OPERATION
7118

FIG. 71

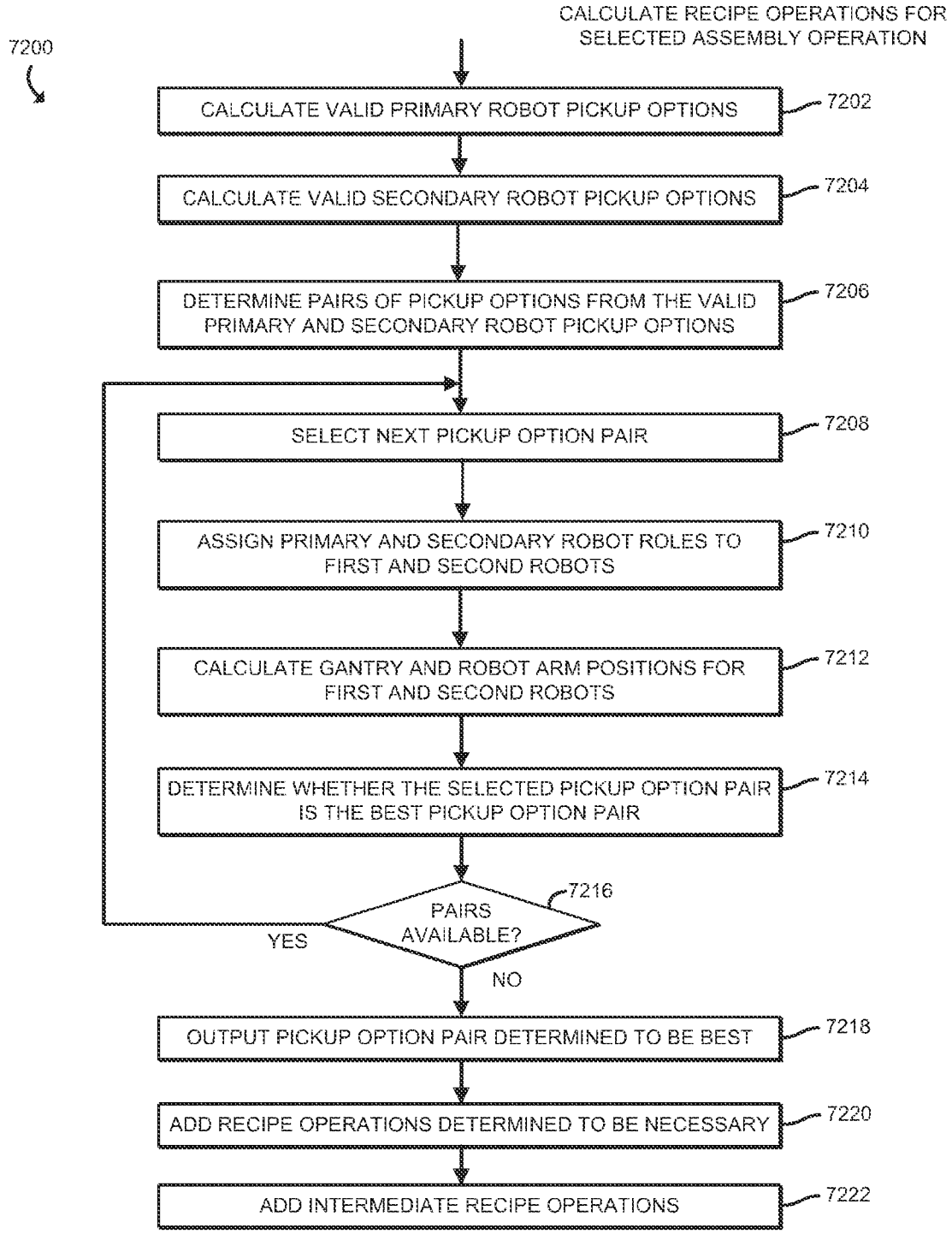

CALCULATE RECIPE OPERATIONS FOR SELECTED ASSEMBLY OPERATION
7200
CALCULATE VALID PRIMARY ROBOT PICKUP OPTIONS
7202
CALCULATE VALID SECONDARY ROBOT PICKUP OPTIONS
7204
DETERMINE PAIRS OF PICKUP OPTIONS FROM THE VALID PRIMARY AND SECONDARY ROBOT PICKUP OPTIONS
7206
SELECT NEXT PICKUP OPTION PAIR
7208
ASSIGN PRIMARY AND SECONDARY ROBOT ROLES TO FIRST AND SECOND ROBOTS
7210
CALCULATE GANTRY AND ROBOT ARM POSITIONS FOR FIRST AND SECOND ROBOTS
7212
DETERMINE WHETHER THE SELECTED PICKUP OPTION PAIR IS THE BEST PICKUP OPTION PAIR
7214
PAIRS AVAILABLE?
7216
YES
NO
OUTPUT PICKUP OPTION PAIR DETERMINED TO BE BEST
7218
ADD RECIPE OPERATIONS DETERMINED TO BE NECESSARY
7220
ADD INTERMEDIATE RECIPE OPERATIONS
7222

FIG. 72

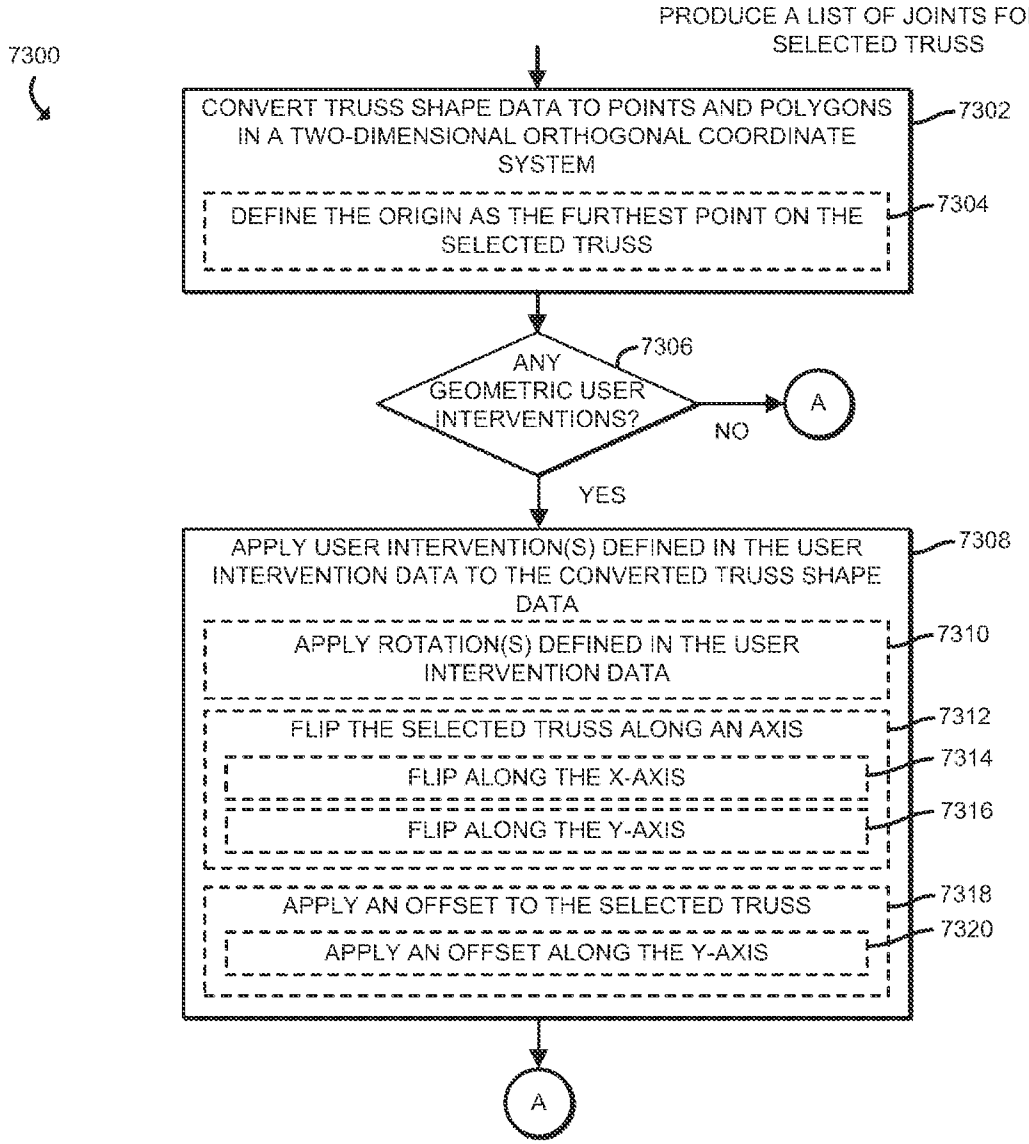

7300
PRODUCE A LIST OF JOINTS FOR THE SELECTED TRUSS
CONVERT TRUSS SHAPE DATA TO POINTS AND POLYGONS IN A TWO-DIMENSIONAL ORTHOGONAL COORDINATE SYSTEM
7302
DEFINE THE ORIGIN AS THE FURTHEST POINT ON THE SELECTED TRUSS
7304
ANY GEOMETRIC USER INTERVENTIONS?
7306
NO
A
YES
APPLY USER INTERVENTION(S) DEFINED IN THE USER INTERVENTION DATA TO THE CONVERTED TRUSS SHAPE DATA
7308
APPLY ROTATION(S) DEFINED IN THE USER INTERVENTION DATA
7310
FLIP THE SELECTED TRUSS ALONG AN AXIS
7312
FLIP ALONG THE X-AXIS
7314
FLIP ALONG THE Y-AXIS
7316
APPLY AN OFFSET TO THE SELECTED TRUSS
7318
APPLY AN OFFSET ALONG THE Y-AXIS
7320
A

FIG. 73

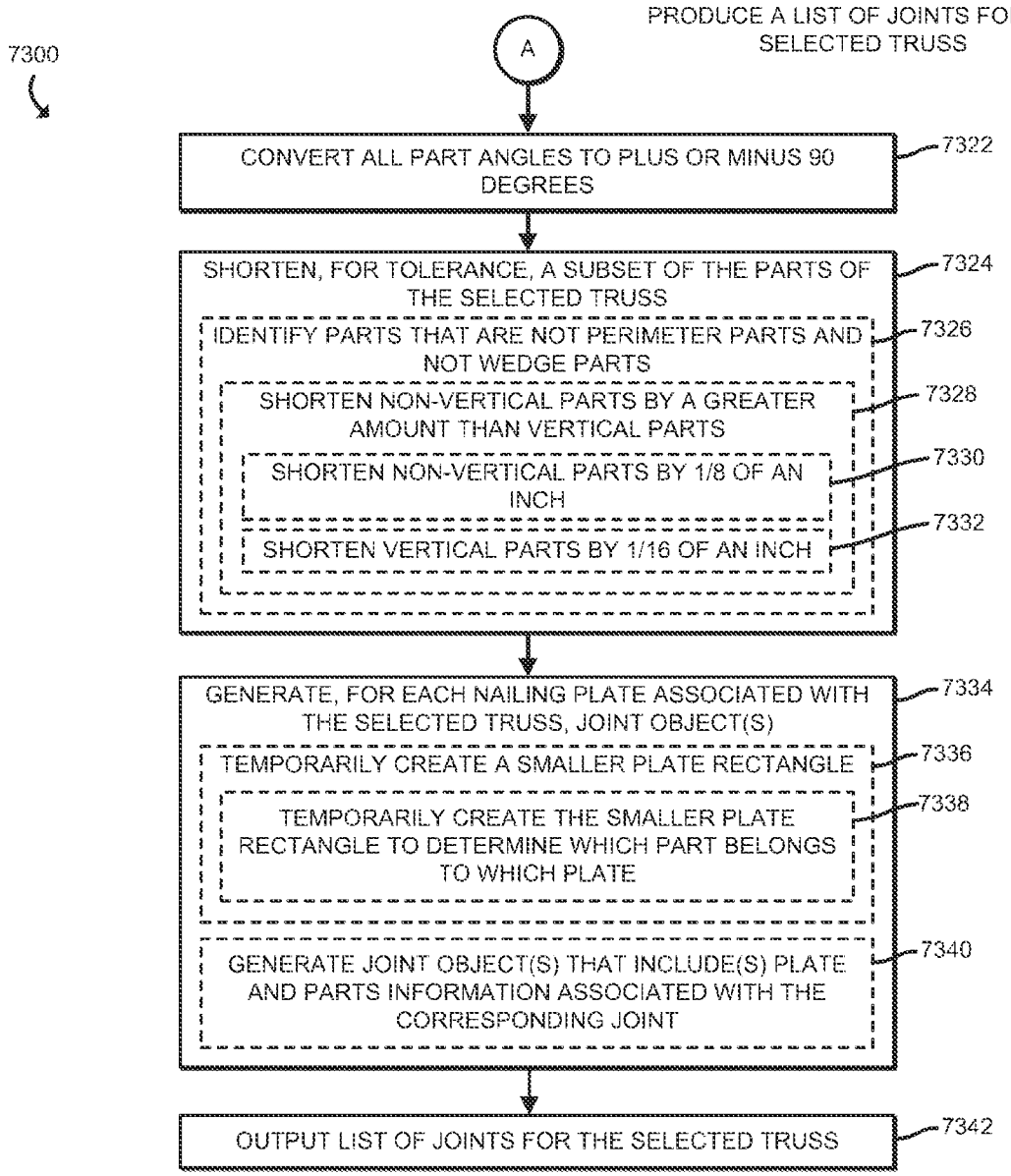

7300
A
PRODUCE A LIST OF JOINTS FOR THE SELECTED TRUSS
CONVERT ALL PART ANGLES TO PLUS OR MINUS 90 DEGREES
7322
SHORTEN, FOR TOLERANCE, A SUBSET OF THE PARTS OF THE SELECTED TRUSS
7324
IDENTIFY PARTS THAT ARE NOT PERIMETER PARTS AND NOT WEDGE PARTS
7326
SHORTEN NON-VERTICAL PARTS BY A GREATER AMOUNT THAN VERTICAL PARTS
7328
SHORTEN NON-VERTICAL PARTS BY 1/8 OF AN INCH
7330
SHORTEN VERTICAL PARTS BY 1/16 OF AN INCH
7332
GENERATE, FOR EACH NAILING PLATE ASSOCIATED WITH THE SELECTED TRUSS, JOINT OBJECT(S)
7334
TEMPORARILY CREATE A SMALLER PLATE RECTANGLE
7336
TEMPORARILY CREATE THE SMALLER PLATE RECTANGLE TO DETERMINE WHICH PART BELONGS TO WHICH PLATE
7338
GENERATE JOINT OBJECT(S) THAT INCLUDE(S) PLATE AND PARTS INFORMATION ASSOCIATED WITH THE CORRESPONDING JOINT
7340
OUTPUT LIST OF JOINTS FOR THE SELECTED TRUSS
7342

FIG. 74

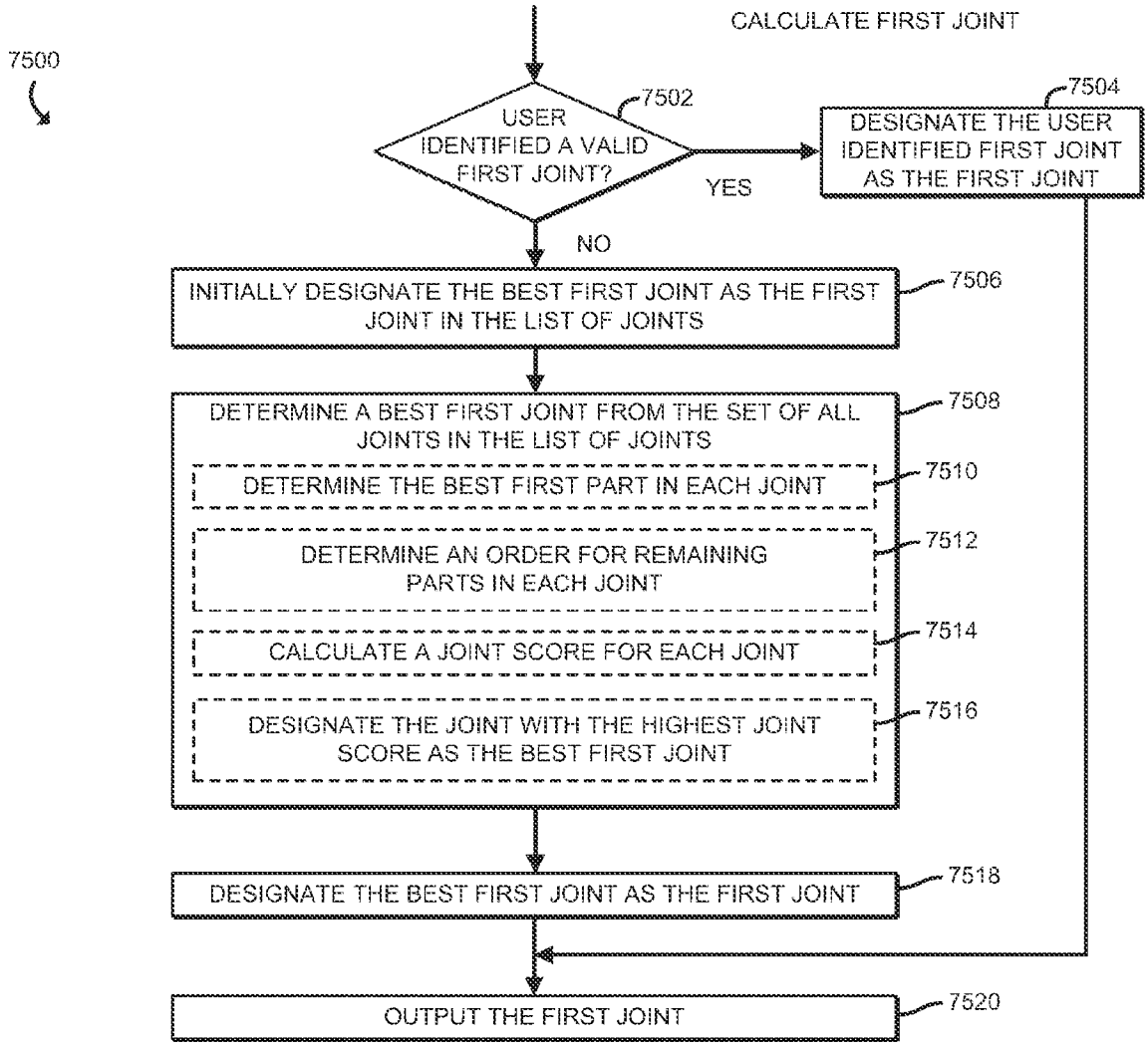

CALCULATE FIRST JOINT
7500
USER IDENTIFIED A VALID FIRST JOINT?
7502
YES
DESIGNATE THE USER IDENTIFIED FIRST JOINT AS THE FIRST JOINT
7504
NO
INITIALLY DESIGNATE THE BEST FIRST JOINT AS THE FIRST JOINT IN THE LIST OF JOINTS
7506
DETERMINE A BEST FIRST JOINT FROM THE SET OF ALL JOINTS IN THE LIST OF JOINTS
7508
DETERMINE THE BEST FIRST PART IN EACH JOINT
7510
DETERMINE AN ORDER FOR REMAINING PARTS IN EACH JOINT
7512
CALCULATE A JOINT SCORE FOR EACH JOINT
7514
DESIGNATE THE JOINT WITH THE HIGHEST JOINT SCORE AS THE BEST FIRST JOINT
7516
DESIGNATE THE BEST FIRST JOINT AS THE FIRST JOINT
7518
OUTPUT THE FIRST JOINT
7520

FIG. 75

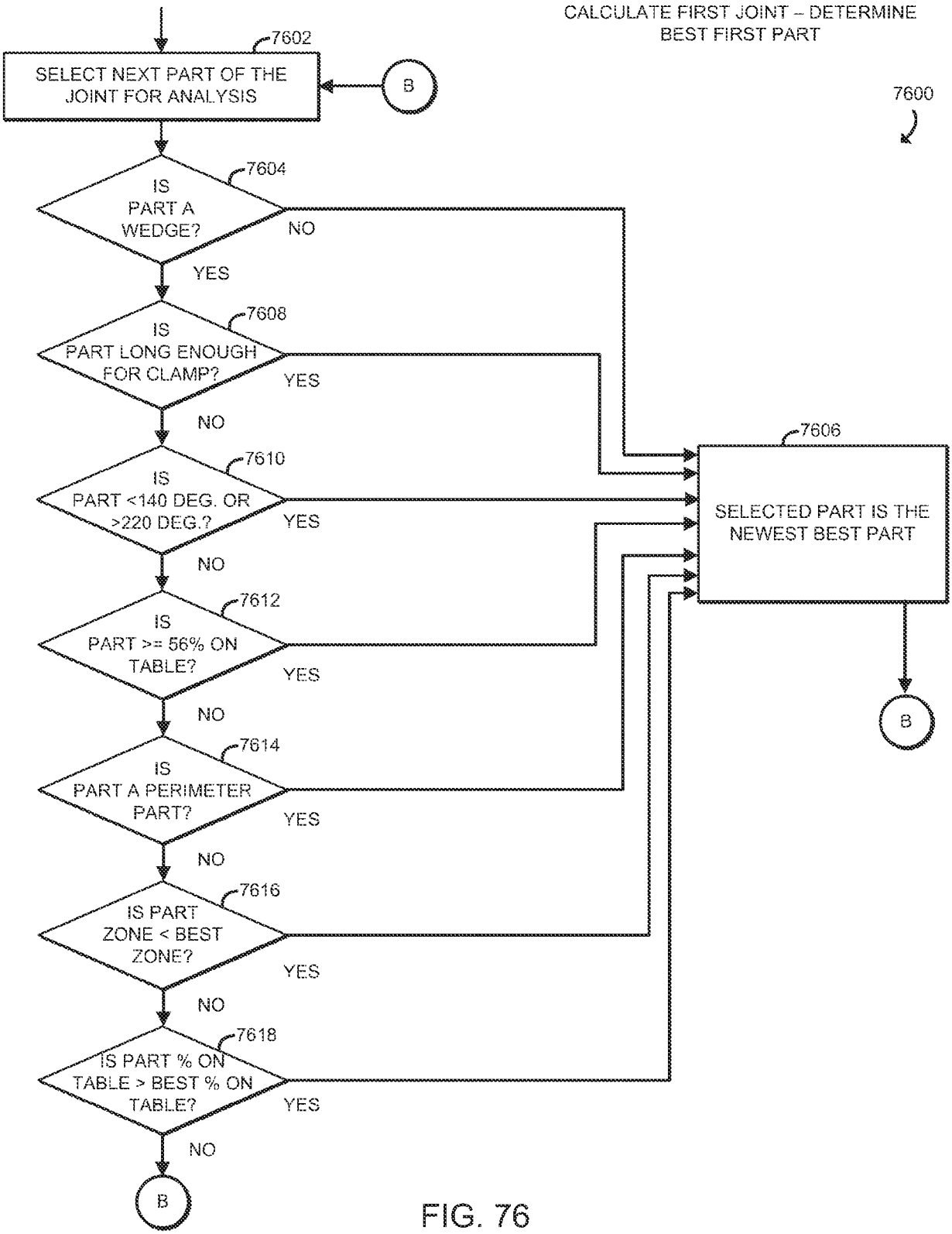

CALCULATE FIRST JOINT – DETERMINE BEST FIRST PART
7600
7602
SELECT NEXT PART OF THE JOINT FOR ANALYSIS
B
7604
IS PART A WEDGE?
NO
YES
7608
IS PART LONG ENOUGH FOR CLAMP?
YES
NO
7610
IS PART <140 DEG. OR >220 DEG.?
YES
NO
7612
IS PART >= 56% ON TABLE?
YES
NO
7614
IS PART A PERIMETER PART?
YES
NO
7616
IS PART ZONE < BEST ZONE?
YES
NO
7618
IS PART % ON TABLE > BEST % ON TABLE?
YES
NO
B
7606
SELECTED PART IS THE NEWEST BEST PART
B

FIG. 76

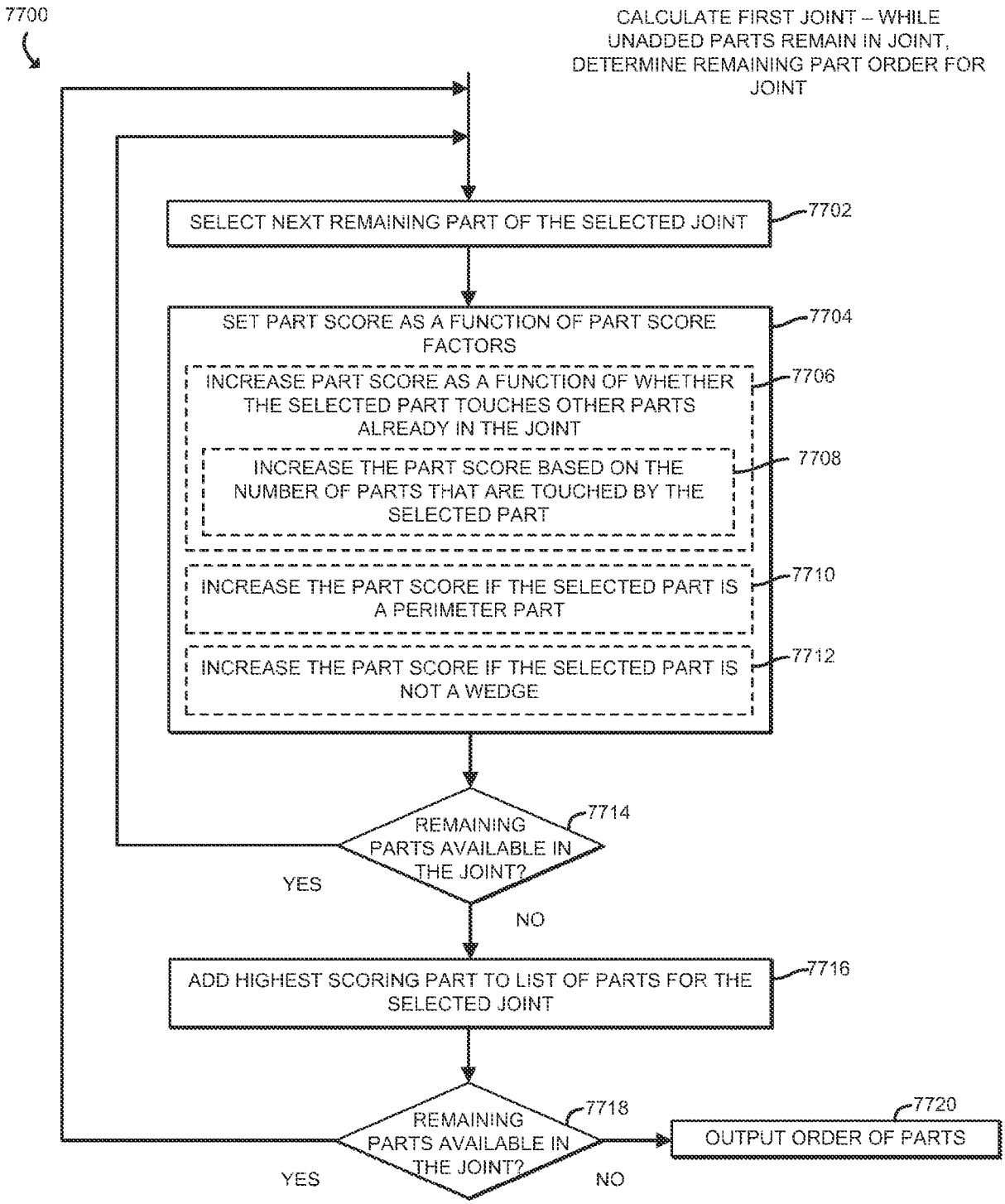

7700
CALCULATE FIRST JOINT – WHILE UNADDED PARTS REMAIN IN JOINT, DETERMINE REMAINING PART ORDER FOR JOINT
SELECT NEXT REMAINING PART OF THE SELECTED JOINT
7702
SET PART SCORE AS A FUNCTION OF PART SCORE FACTORS
7704
INCREASE PART SCORE AS A FUNCTION OF WHETHER THE SELECTED PART TOUCHES OTHER PARTS ALREADY IN THE JOINT
7706
INCREASE THE PART SCORE BASED ON THE NUMBER OF PARTS THAT ARE TOUCHED BY THE SELECTED PART
7708
INCREASE THE PART SCORE IF THE SELECTED PART IS A PERIMETER PART
7710
INCREASE THE PART SCORE IF THE SELECTED PART IS NOT A WEDGE
7712
REMAINING PARTS AVAILABLE IN THE JOINT?
7714
YES
NO
ADD HIGHEST SCORING PART TO LIST OF PARTS FOR THE SELECTED JOINT
7716
REMAINING PARTS AVAILABLE IN THE JOINT?
7718
YES
NO
OUTPUT ORDER OF PARTS
7720

FIG. 77

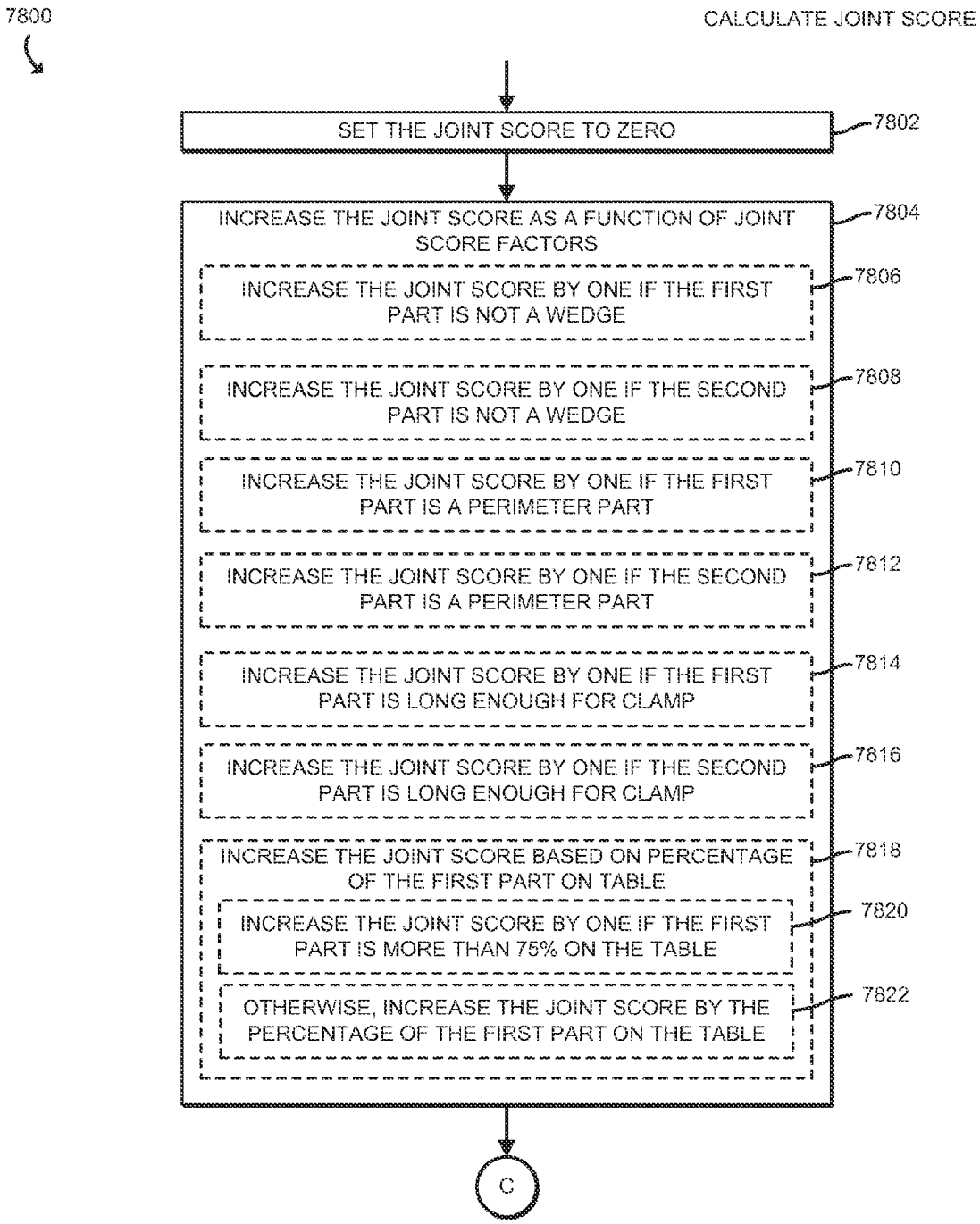

7800
CALCULATE JOINT SCORE
SET THE JOINT SCORE TO ZERO
7802
INCREASE THE JOINT SCORE AS A FUNCTION OF JOINT SCORE FACTORS
7804
INCREASE THE JOINT SCORE BY ONE IF THE FIRST PART IS NOT A WEDGE
7806
INCREASE THE JOINT SCORE BY ONE IF THE SECOND PART IS NOT A WEDGE
7808
INCREASE THE JOINT SCORE BY ONE IF THE FIRST PART IS A PERIMETER PART
7810
INCREASE THE JOINT SCORE BY ONE IF THE SECOND PART IS A PERIMETER PART
7812
INCREASE THE JOINT SCORE BY ONE IF THE FIRST PART IS LONG ENOUGH FOR CLAMP
7814
INCREASE THE JOINT SCORE BY ONE IF THE SECOND PART IS LONG ENOUGH FOR CLAMP
7816
INCREASE THE JOINT SCORE BASED ON PERCENTAGE OF THE FIRST PART ON TABLE
7818
INCREASE THE JOINT SCORE BY ONE IF THE FIRST PART IS MORE THAN 75% ON THE TABLE
7820
OTHERWISE, INCREASE THE JOINT SCORE BY THE PERCENTAGE OF THE FIRST PART ON THE TABLE
7822
C

FIG. 78

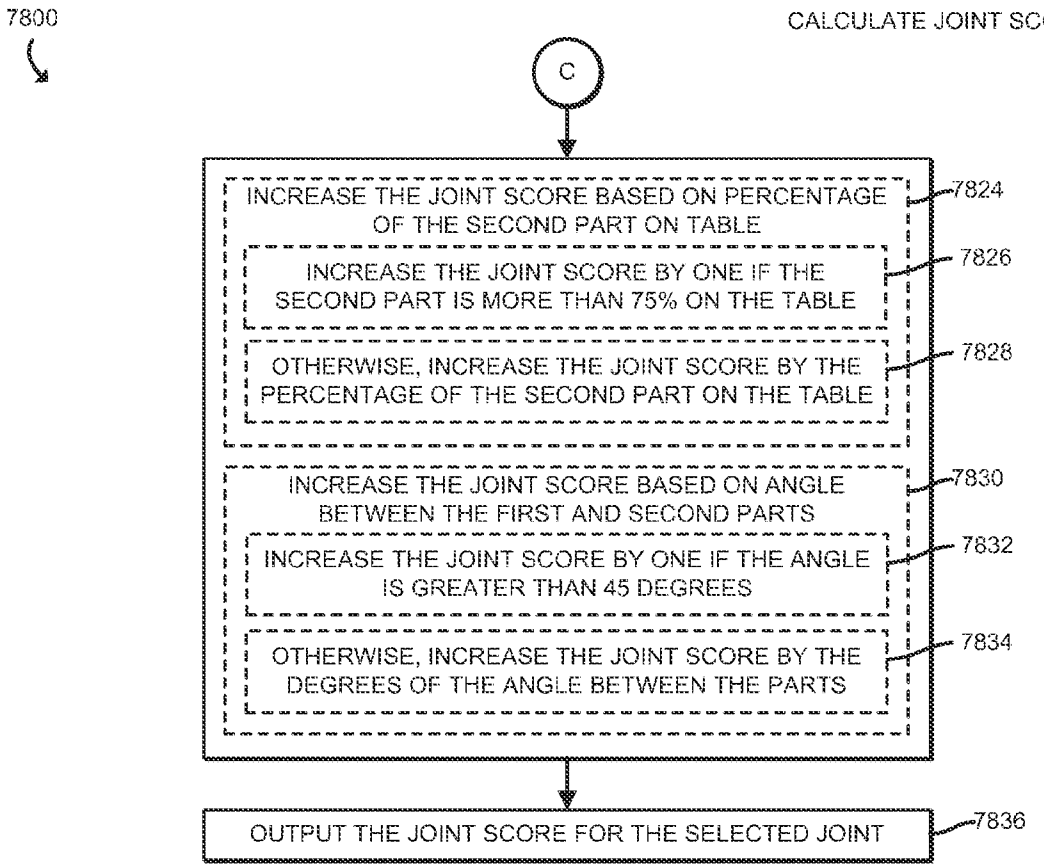

7800
CALCULATE JOINT SCORE
C
INCREASE THE JOINT SCORE BASED ON PERCENTAGE OF THE SECOND PART ON TABLE
7824
INCREASE THE JOINT SCORE BY ONE IF THE SECOND PART IS MORE THAN 75% ON THE TABLE
7826
OTHERWISE, INCREASE THE JOINT SCORE BY THE PERCENTAGE OF THE SECOND PART ON THE TABLE
7828
INCREASE THE JOINT SCORE BASED ON ANGLE BETWEEN THE FIRST AND SECOND PARTS
7830
INCREASE THE JOINT SCORE BY ONE IF THE ANGLE IS GREATER THAN 45 DEGREES
7832
OTHERWISE, INCREASE THE JOINT SCORE BY THE DEGREES OF THE ANGLE BETWEEN THE PARTS
7834
OUTPUT THE JOINT SCORE FOR THE SELECTED JOINT
7836

FIG. 79

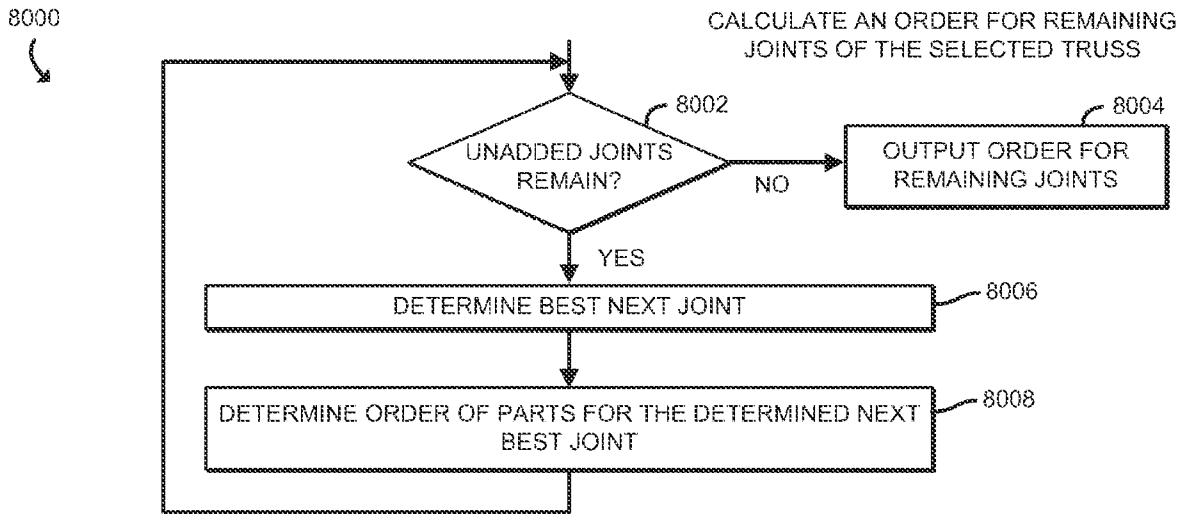

8000
CALCULATE AN ORDER FOR REMAINING JOINTS OF THE SELECTED TRUSS
8002
UNADDED JOINTS REMAIN?
NO
8004
OUTPUT ORDER FOR REMAINING JOINTS
YES
DETERMINE BEST NEXT JOINT
8006
DETERMINE ORDER OF PARTS FOR THE DETERMINED NEXT BEST JOINT
8008

FIG. 80

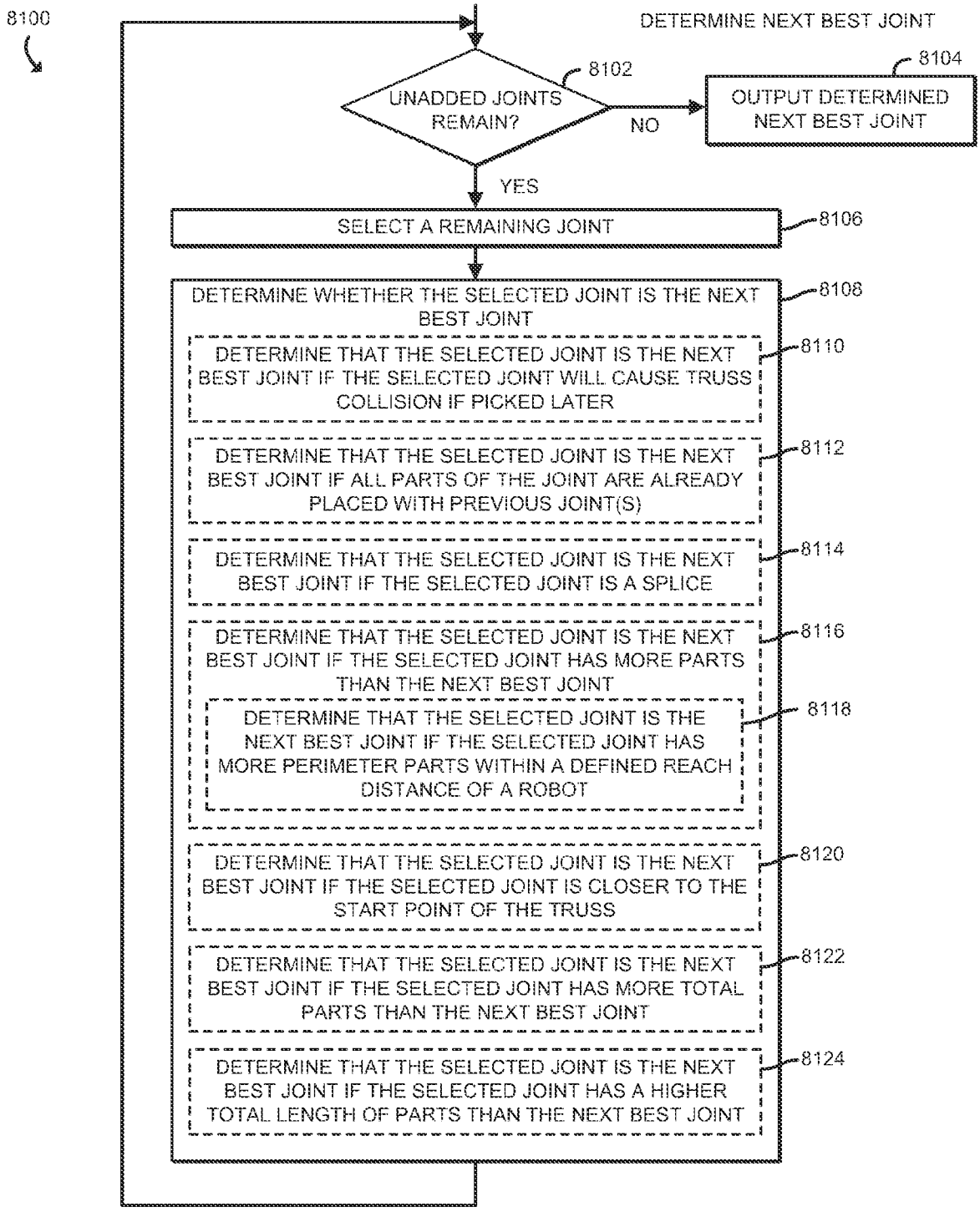

8100
DETERMINE NEXT BEST JOINT
8102
UNADDED JOINTS REMAIN?
NO
8104
OUTPUT DETERMINED NEXT BEST JOINT
YES
SELECT A REMAINING JOINT
8106
DETERMINE WHETHER THE SELECTED JOINT IS THE NEXT BEST JOINT
8108
DETERMINE THAT THE SELECTED JOINT IS THE NEXT BEST JOINT IF THE SELECTED JOINT WILL CAUSE TRUSS COLLISION IF PICKED LATER
8110
DETERMINE THAT THE SELECTED JOINT IS THE NEXT BEST JOINT IF ALL PARTS OF THE JOINT ARE ALREADY PLACED WITH PREVIOUS JOINT(S)
8112
DETERMINE THAT THE SELECTED JOINT IS THE NEXT BEST JOINT IF THE SELECTED JOINT IS A SPLICE
8114
DETERMINE THAT THE SELECTED JOINT IS THE NEXT BEST JOINT IF THE SELECTED JOINT HAS MORE PARTS THAN THE NEXT BEST JOINT
8116
DETERMINE THAT THE SELECTED JOINT IS THE NEXT BEST JOINT IF THE SELECTED JOINT HAS MORE PERIMETER PARTS WITHIN A DEFINED REACH DISTANCE OF A ROBOT
8118
DETERMINE THAT THE SELECTED JOINT IS THE NEXT BEST JOINT IF THE SELECTED JOINT IS CLOSER TO THE START POINT OF THE TRUSS
8120
DETERMINE THAT THE SELECTED JOINT IS THE NEXT BEST JOINT IF THE SELECTED JOINT HAS MORE TOTAL PARTS THAN THE NEXT BEST JOINT
8122
DETERMINE THAT THE SELECTED JOINT IS THE NEXT BEST JOINT IF THE SELECTED JOINT HAS A HIGHER TOTAL LENGTH OF PARTS THAN THE NEXT BEST JOINT
8124

FIG. 81

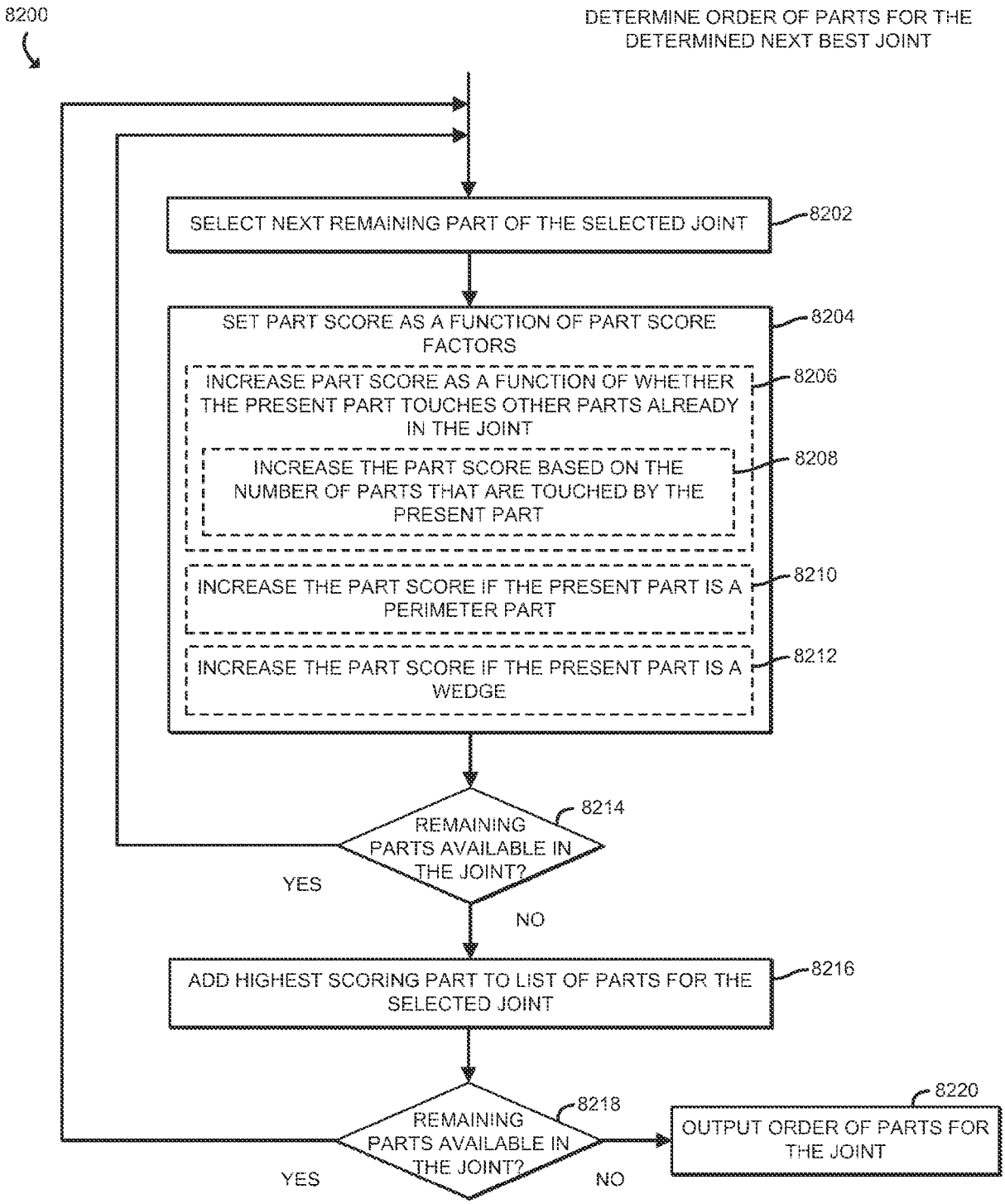

8200
DETERMINE ORDER OF PARTS FOR THE DETERMINED NEXT BEST JOINT
SELECT NEXT REMAINING PART OF THE SELECTED JOINT
8202
SET PART SCORE AS A FUNCTION OF PART SCORE FACTORS
8204
INCREASE PART SCORE AS A FUNCTION OF WHETHER THE PRESENT PART TOUCHES OTHER PARTS ALREADY IN THE JOINT
8206
INCREASE THE PART SCORE BASED ON THE NUMBER OF PARTS THAT ARE TOUCHED BY THE PRESENT PART
8208
INCREASE THE PART SCORE IF THE PRESENT PART IS A PERIMETER PART
8210
INCREASE THE PART SCORE IF THE PRESENT PART IS A WEDGE
8212
REMAINING PARTS AVAILABLE IN THE JOINT?
8214
YES
NO
ADD HIGHEST SCORING PART TO LIST OF PARTS FOR THE SELECTED JOINT
8216
REMAINING PARTS AVAILABLE IN THE JOINT?
8218
YES
NO
OUTPUT ORDER OF PARTS FOR THE JOINT
8220

FIG. 82

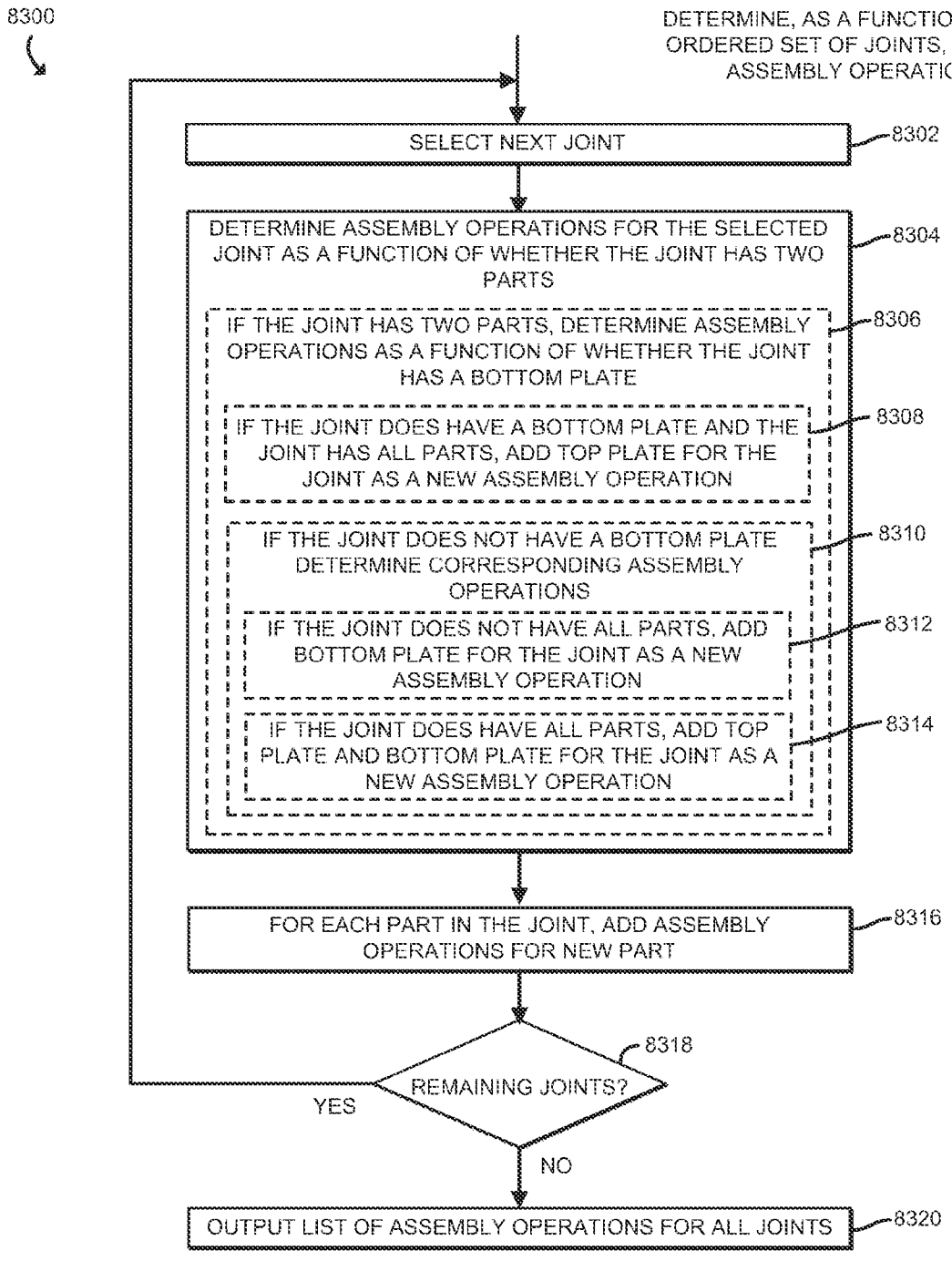

8300
DETERMINE, AS A FUNCTION OF THE ORDERED SET OF JOINTS, A SET OF ASSEMBLY OPERATIONS
SELECT NEXT JOINT
8302
DETERMINE ASSEMBLY OPERATIONS FOR THE SELECTED JOINT AS A FUNCTION OF WHETHER THE JOINT HAS TWO PARTS
8304
IF THE JOINT HAS TWO PARTS, DETERMINE ASSEMBLY OPERATIONS AS A FUNCTION OF WHETHER THE JOINT HAS A BOTTOM PLATE
8306
IF THE JOINT DOES HAVE A BOTTOM PLATE AND THE JOINT HAS ALL PARTS, ADD TOP PLATE FOR THE JOINT AS A NEW ASSEMBLY OPERATION
8308
IF THE JOINT DOES NOT HAVE A BOTTOM PLATE DETERMINE CORRESPONDING ASSEMBLY OPERATIONS
8310
IF THE JOINT DOES NOT HAVE ALL PARTS, ADD BOTTOM PLATE FOR THE JOINT AS A NEW ASSEMBLY OPERATION
8312
IF THE JOINT DOES HAVE ALL PARTS, ADD TOP PLATE AND BOTTOM PLATE FOR THE JOINT AS A NEW ASSEMBLY OPERATION
8314
FOR EACH PART IN THE JOINT, ADD ASSEMBLY OPERATIONS FOR NEW PART
8316
REMAINING JOINTS?
8318
YES
NO
OUTPUT LIST OF ASSEMBLY OPERATIONS FOR ALL JOINTS
8320

FIG. 83

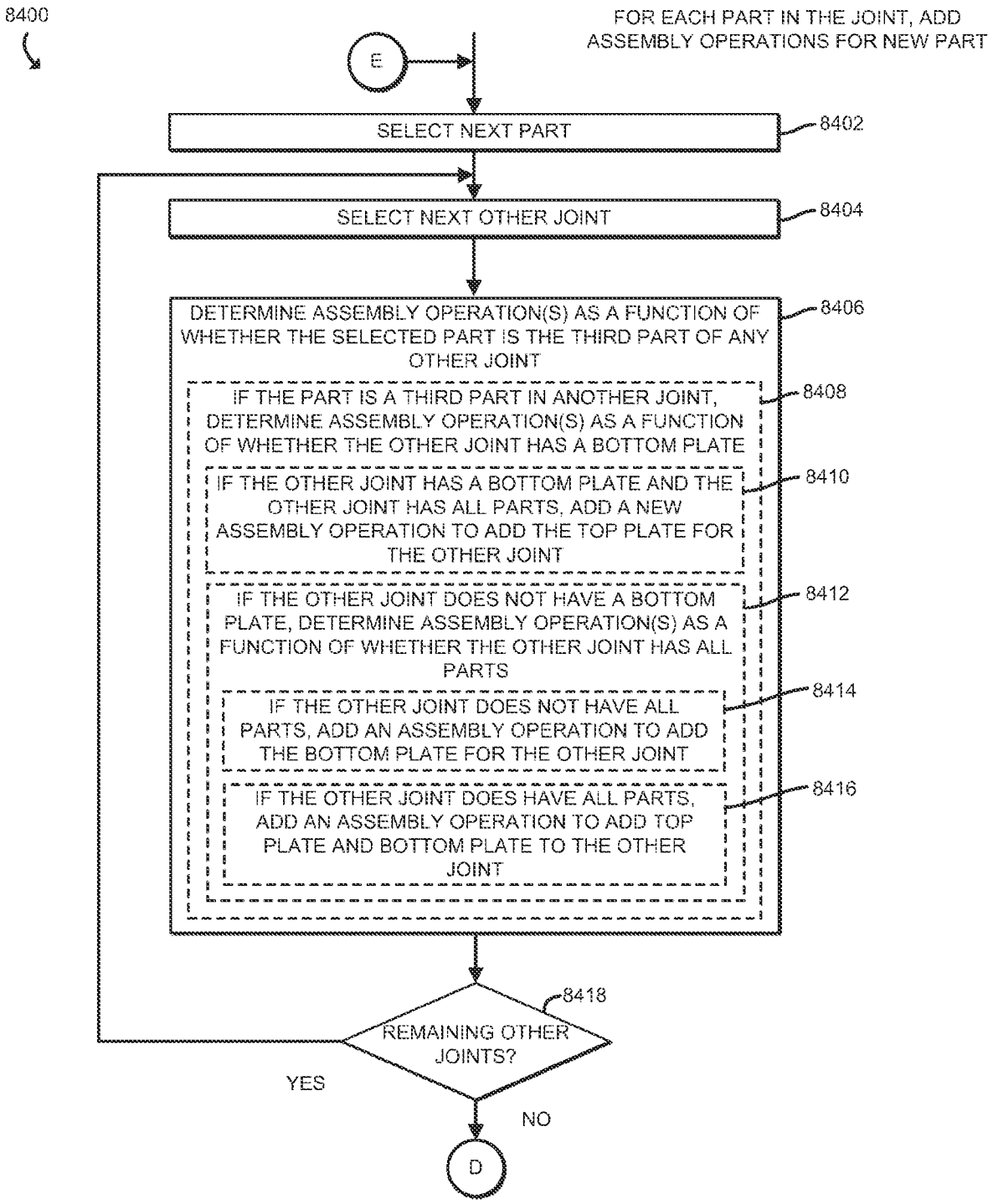

8400
FOR EACH PART IN THE JOINT, ADD ASSEMBLY OPERATIONS FOR NEW PART
E
SELECT NEXT PART
8402
SELECT NEXT OTHER JOINT
8404
DETERMINE ASSEMBLY OPERATION(S) AS A FUNCTION OF WHETHER THE SELECTED PART IS THE THIRD PART OF ANY OTHER JOINT
8406
IF THE PART IS A THIRD PART IN ANOTHER JOINT, DETERMINE ASSEMBLY OPERATION(S) AS A FUNCTION OF WHETHER THE OTHER JOINT HAS A BOTTOM PLATE
8408
IF THE OTHER JOINT HAS A BOTTOM PLATE AND THE OTHER JOINT HAS ALL PARTS, ADD A NEW ASSEMBLY OPERATION TO ADD THE TOP PLATE FOR THE OTHER JOINT
8410
IF THE OTHER JOINT DOES NOT HAVE A BOTTOM PLATE, DETERMINE ASSEMBLY OPERATION(S) AS A FUNCTION OF WHETHER THE OTHER JOINT HAS ALL PARTS
8412
IF THE OTHER JOINT DOES NOT HAVE ALL PARTS, ADD AN ASSEMBLY OPERATION TO ADD THE BOTTOM PLATE FOR THE OTHER JOINT
8414
IF THE OTHER JOINT DOES HAVE ALL PARTS, ADD AN ASSEMBLY OPERATION TO ADD TOP PLATE AND BOTTOM PLATE TO THE OTHER JOINT
8416
8418
REMAINING OTHER JOINTS?
YES
NO
D

FIG. 84

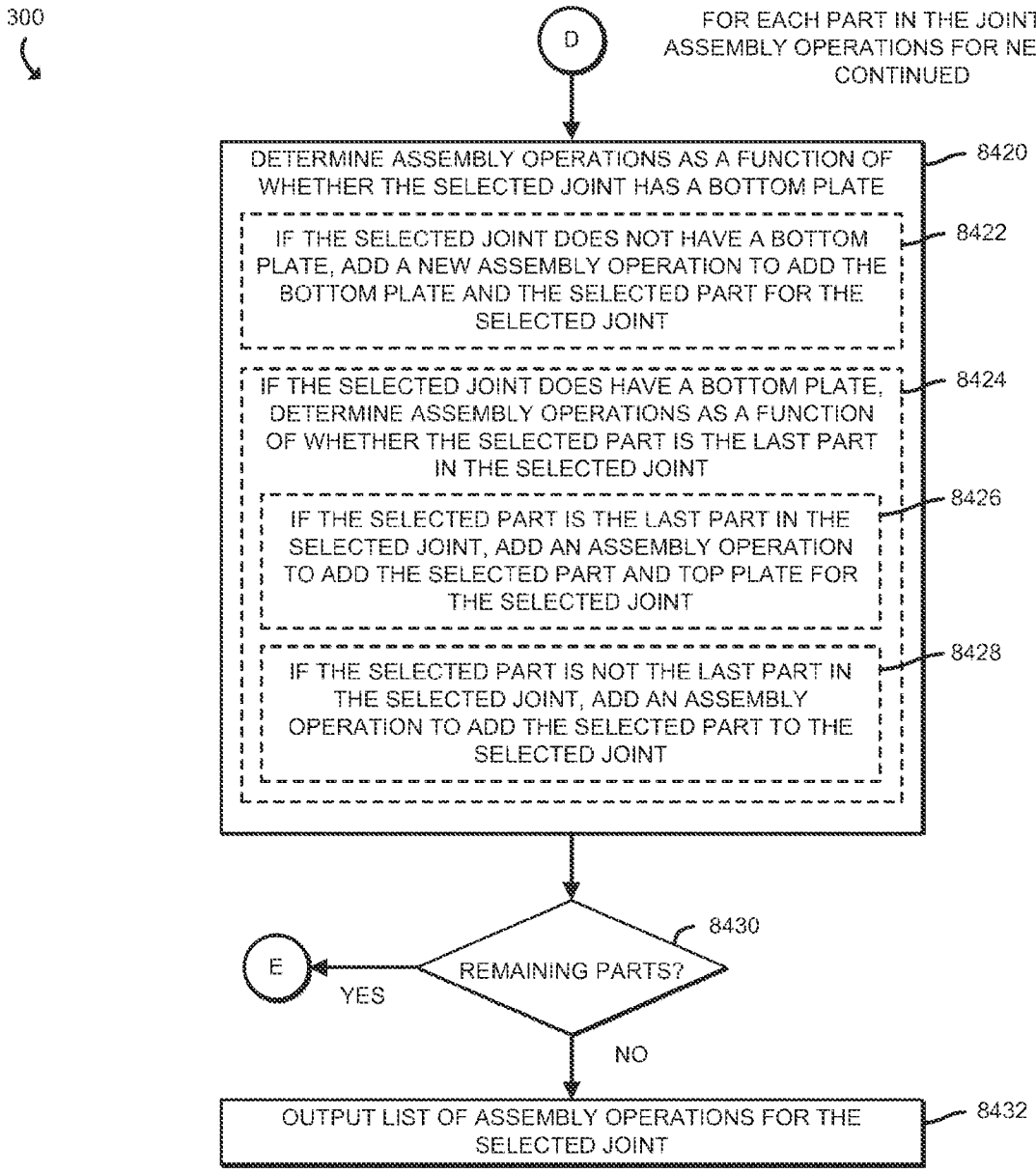

300
D
FOR EACH PART IN THE JOINT, ADD ASSEMBLY OPERATIONS FOR NEW PART - CONTINUED
DETERMINE ASSEMBLY OPERATIONS AS A FUNCTION OF WHETHER THE SELECTED JOINT HAS A BOTTOM PLATE
8420
IF THE SELECTED JOINT DOES NOT HAVE A BOTTOM PLATE, ADD A NEW ASSEMBLY OPERATION TO ADD THE BOTTOM PLATE AND THE SELECTED PART FOR THE SELECTED JOINT
8422
IF THE SELECTED JOINT DOES HAVE A BOTTOM PLATE, DETERMINE ASSEMBLY OPERATIONS AS A FUNCTION OF WHETHER THE SELECTED PART IS THE LAST PART IN THE SELECTED JOINT
8424
IF THE SELECTED PART IS THE LAST PART IN THE SELECTED JOINT, ADD AN ASSEMBLY OPERATION TO ADD THE SELECTED PART AND TOP PLATE FOR THE SELECTED JOINT
8426
IF THE SELECTED PART IS NOT THE LAST PART IN THE SELECTED JOINT, ADD AN ASSEMBLY OPERATION TO ADD THE SELECTED PART TO THE SELECTED JOINT
8428
REMAINING PARTS?
8430
E
YES
NO
OUTPUT LIST OF ASSEMBLY OPERATIONS FOR THE SELECTED JOINT
8432

FIG. 85

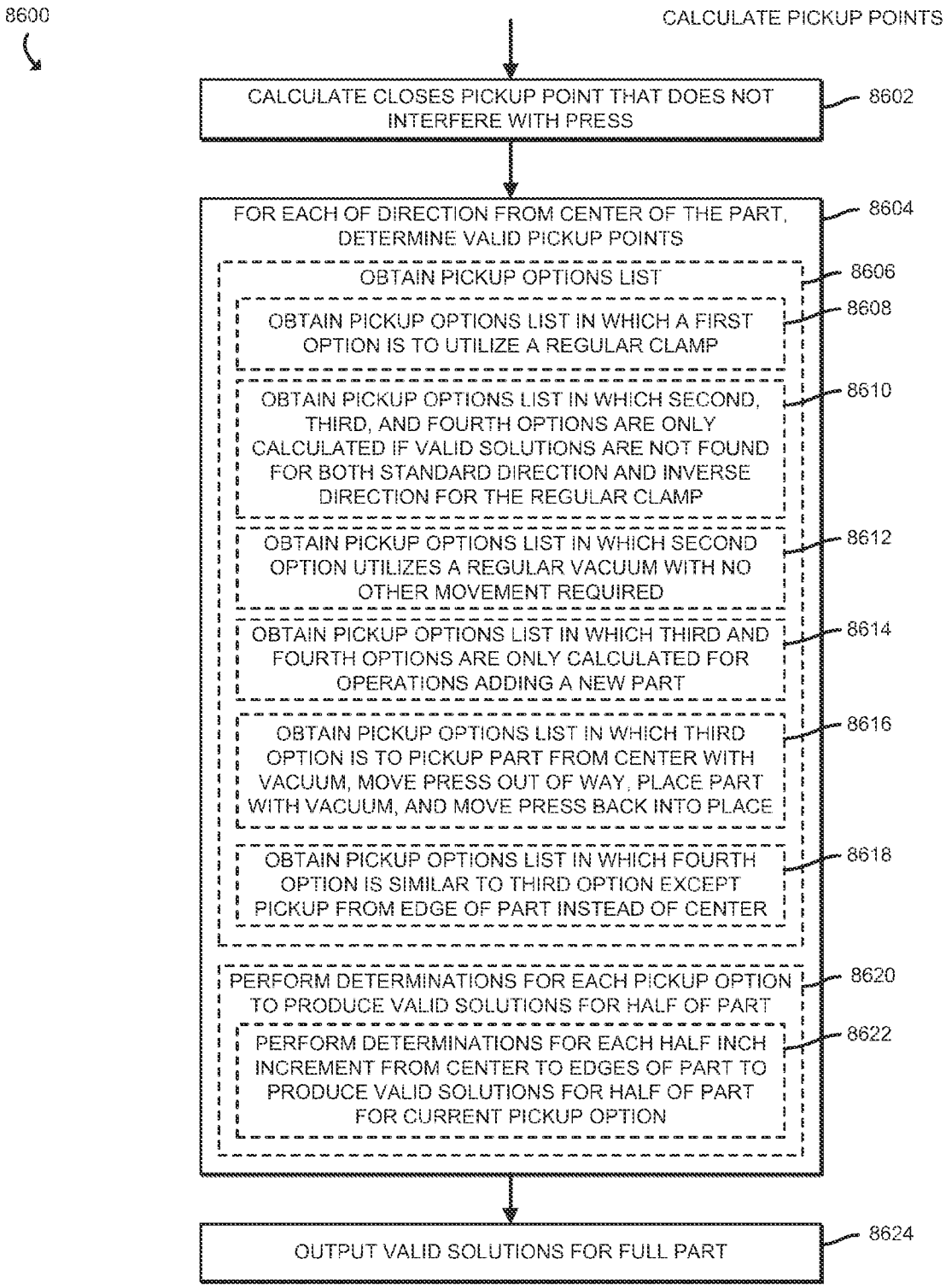

8600
CALCULATE PICKUP POINTS
CALCULATE CLOSES PICKUP POINT THAT DOES NOT INTERFERE WITH PRESS
8602
FOR EACH OF DIRECTION FROM CENTER OF THE PART, DETERMINE VALID PICKUP POINTS
8604
OBTAIN PICKUP OPTIONS LIST
8606
OBTAIN PICKUP OPTIONS LIST IN WHICH A FIRST OPTION IS TO UTILIZE A REGULAR CLAMP
8608
OBTAIN PICKUP OPTIONS LIST IN WHICH SECOND, THIRD, AND FOURTH OPTIONS ARE ONLY CALCULATED IF VALID SOLUTIONS ARE NOT FOUND FOR BOTH STANDARD DIRECTION AND INVERSE DIRECTION FOR THE REGULAR CLAMP
8610
OBTAIN PICKUP OPTIONS LIST IN WHICH SECOND OPTION UTILIZES A REGULAR VACUUM WITH NO OTHER MOVEMENT REQUIRED
8612
OBTAIN PICKUP OPTIONS LIST IN WHICH THIRD AND FOURTH OPTIONS ARE ONLY CALCULATED FOR OPERATIONS ADDING A NEW PART
8614
OBTAIN PICKUP OPTIONS LIST IN WHICH THIRD OPTION IS TO PICKUP PART FROM CENTER WITH VACUUM, MOVE PRESS OUT OF WAY, PLACE PART WITH VACUUM, AND MOVE PRESS BACK INTO PLACE
8616
OBTAIN PICKUP OPTIONS LIST IN WHICH FOURTH OPTION IS SIMILAR TO THIRD OPTION EXCEPT PICKUP FROM EDGE OF PART INSTEAD OF CENTER
8618
PERFORM DETERMINATIONS FOR EACH PICKUP OPTION TO PRODUCE VALID SOLUTIONS FOR HALF OF PART
8620
PERFORM DETERMINATIONS FOR EACH HALF INCH INCREMENT FROM CENTER TO EDGES OF PART TO PRODUCE VALID SOLUTIONS FOR HALF OF PART FOR CURRENT PICKUP OPTION
8622
OUTPUT VALID SOLUTIONS FOR FULL PART
8624

FIG. 86

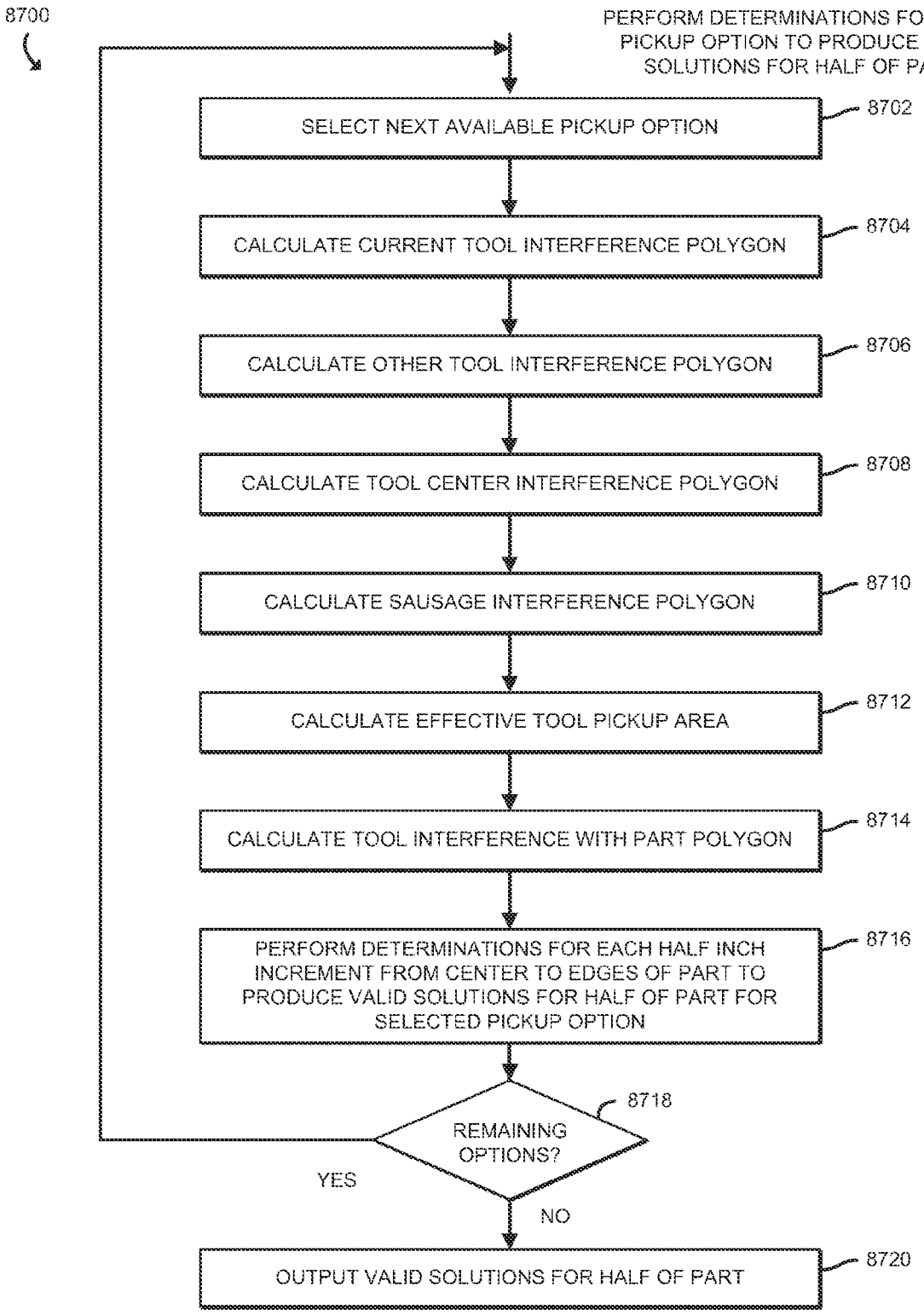

8700
PERFORM DETERMINATIONS FOR EACH PICKUP OPTION TO PRODUCE VALID SOLUTIONS FOR HALF OF PART
SELECT NEXT AVAILABLE PICKUP OPTION
8702
CALCULATE CURRENT TOOL INTERFERENCE POLYGON
8704
CALCULATE OTHER TOOL INTERFERENCE POLYGON
8706
CALCULATE TOOL CENTER INTERFERENCE POLYGON
8708
CALCULATE SAUSAGE INTERFERENCE POLYGON
8710
CALCULATE EFFECTIVE TOOL PICKUP AREA
8712
CALCULATE TOOL INTERFERENCE WITH PART POLYGON
8714
PERFORM DETERMINATIONS FOR EACH HALF INCH INCREMENT FROM CENTER TO EDGES OF PART TO PRODUCE VALID SOLUTIONS FOR HALF OF PART FOR SELECTED PICKUP OPTION
8716
REMAINING OPTIONS?
8718
YES
NO
OUTPUT VALID SOLUTIONS FOR HALF OF PART
8720

FIG. 87

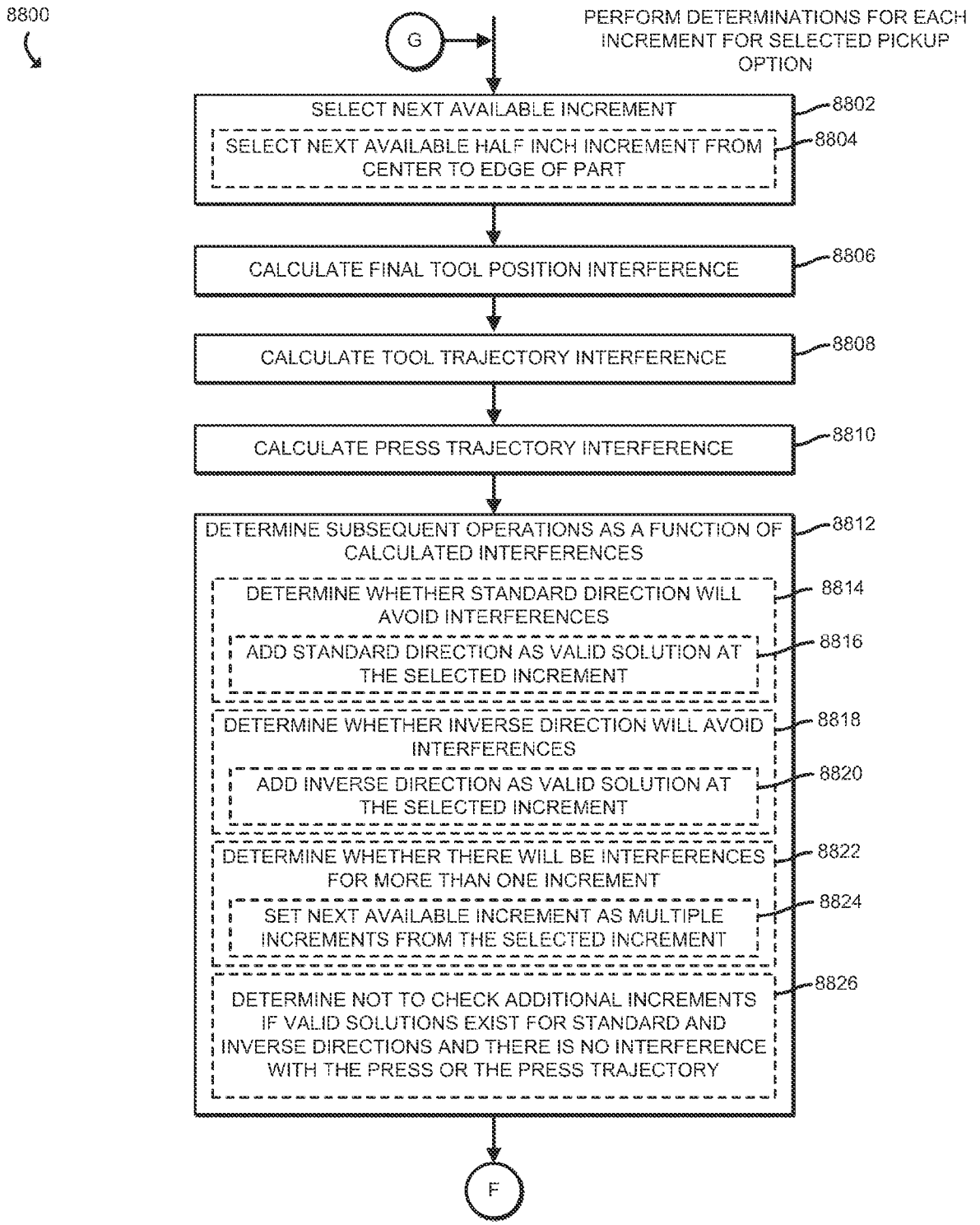

8800
G
PERFORM DETERMINATIONS FOR EACH INCREMENT FOR SELECTED PICKUP OPTION
SELECT NEXT AVAILABLE INCREMENT
8802
SELECT NEXT AVAILABLE HALF INCH INCREMENT FROM CENTER TO EDGE OF PART
8804
CALCULATE FINAL TOOL POSITION INTERFERENCE
8806
CALCULATE TOOL TRAJECTORY INTERFERENCE
8808
CALCULATE PRESS TRAJECTORY INTERFERENCE
8810
DETERMINE SUBSEQUENT OPERATIONS AS A FUNCTION OF CALCULATED INTERFERENCES
8812
DETERMINE WHETHER STANDARD DIRECTION WILL AVOID INTERFERENCES
8814
ADD STANDARD DIRECTION AS VALID SOLUTION AT THE SELECTED INCREMENT
8816
DETERMINE WHETHER INVERSE DIRECTION WILL AVOID INTERFERENCES
8818
ADD INVERSE DIRECTION AS VALID SOLUTION AT THE SELECTED INCREMENT
8820
DETERMINE WHETHER THERE WILL BE INTERFERENCES FOR MORE THAN ONE INCREMENT
8822
SET NEXT AVAILABLE INCREMENT AS MULTIPLE INCREMENTS FROM THE SELECTED INCREMENT
8824
DETERMINE NOT TO CHECK ADDITIONAL INCREMENTS IF VALID SOLUTIONS EXIST FOR STANDARD AND INVERSE DIRECTIONS AND THERE IS NO INTERFERENCE WITH THE PRESS OR THE PRESS TRAJECTORY
8826
F

FIG. 88

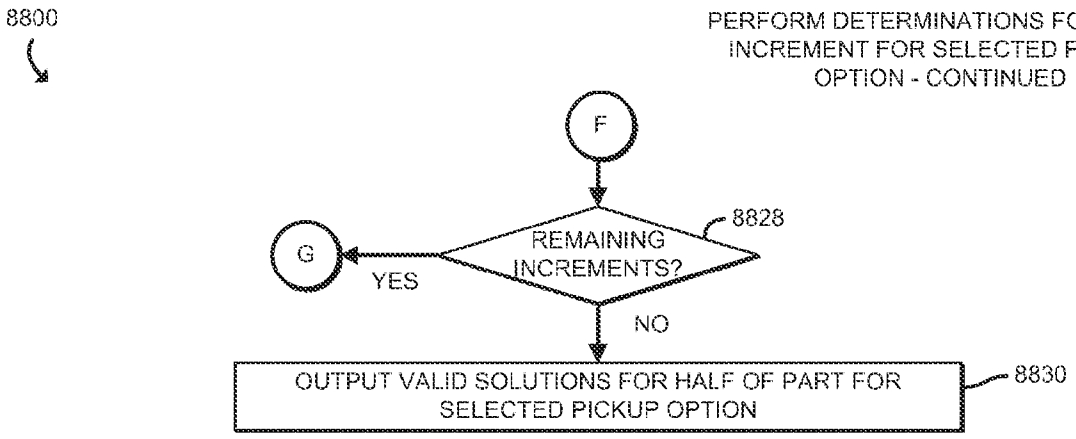

8800
PERFORM DETERMINATIONS FOR EACH INCREMENT FOR SELECTED PICKUP OPTION - CONTINUED
F
REMAINING INCREMENTS?
8828
G
YES
NO
OUTPUT VALID SOLUTIONS FOR HALF OF PART FOR SELECTED PICKUP OPTION
8830

FIG. 89

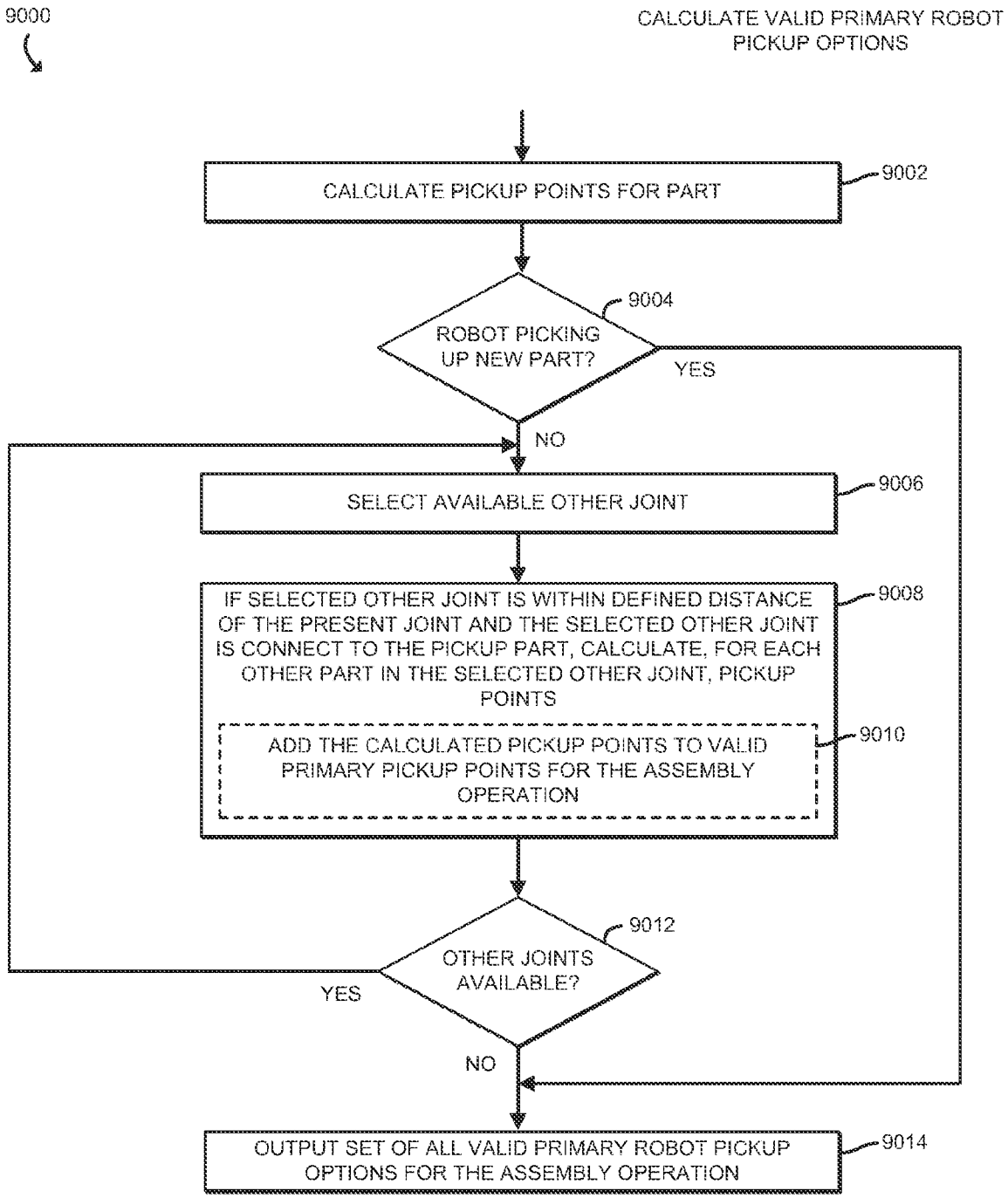

9000
CALCULATE VALID PRIMARY ROBOT PICKUP OPTIONS
CALCULATE PICKUP POINTS FOR PART
9002
ROBOT PICKING UP NEW PART?
9004
YES
NO
SELECT AVAILABLE OTHER JOINT
9006
IF SELECTED OTHER JOINT IS WITHIN DEFINED DISTANCE OF THE PRESENT JOINT AND THE SELECTED OTHER JOINT IS CONNECT TO THE PICKUP PART, CALCULATE, FOR EACH OTHER PART IN THE SELECTED OTHER JOINT, PICKUP POINTS
9008
ADD THE CALCULATED PICKUP POINTS TO VALID PRIMARY PICKUP POINTS FOR THE ASSEMBLY OPERATION
9010
OTHER JOINTS AVAILABLE?
9012
YES
NO
OUTPUT SET OF ALL VALID PRIMARY ROBOT PICKUP OPTIONS FOR THE ASSEMBLY OPERATION
9014

FIG. 90

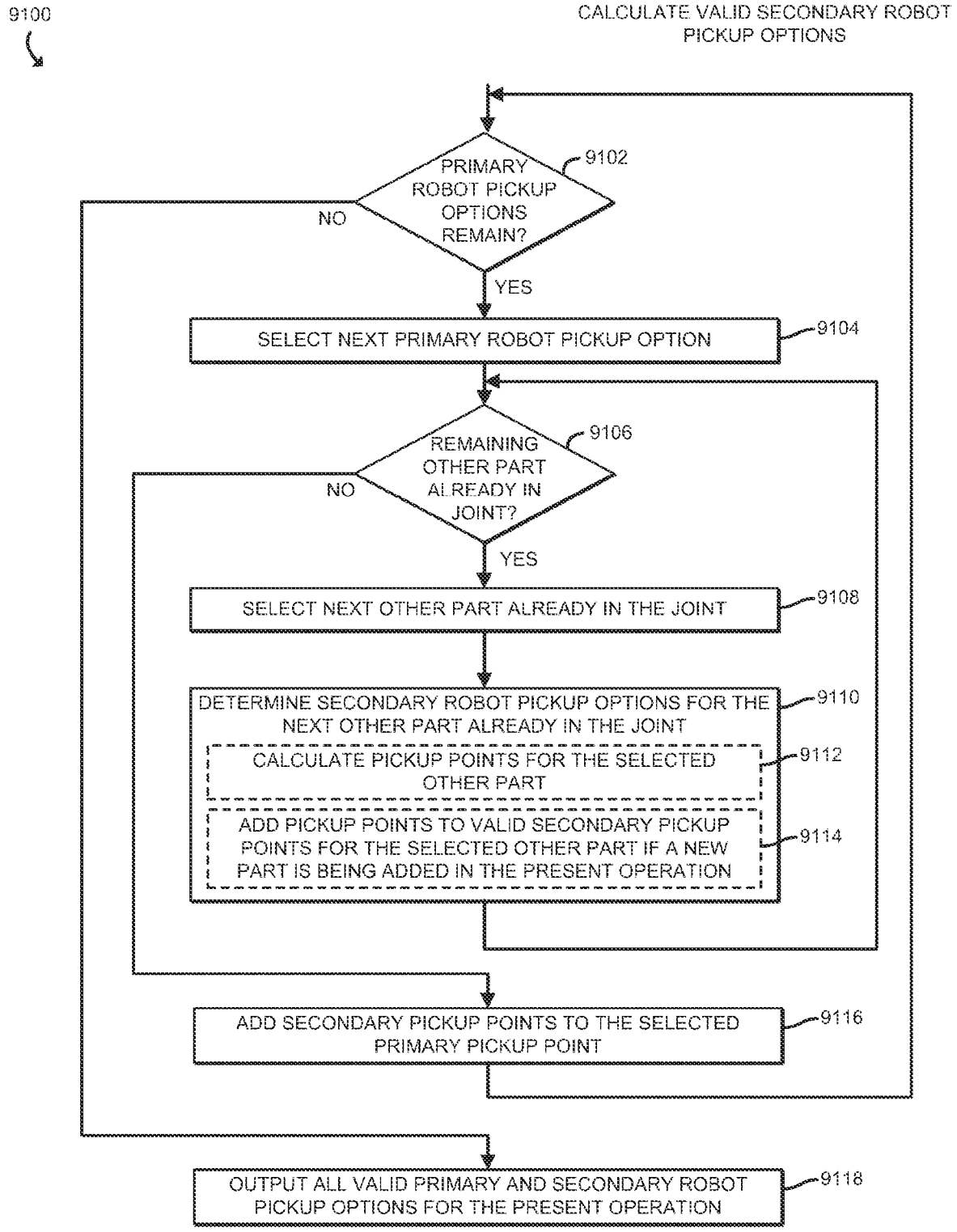

9100
CALCULATE VALID SECONDARY ROBOT PICKUP OPTIONS
PRIMARY ROBOT PICKUP OPTIONS REMAIN?
9102
NO
YES
SELECT NEXT PRIMARY ROBOT PICKUP OPTION
9104
REMAINING OTHER PART ALREADY IN JOINT?
9106
NO
YES
SELECT NEXT OTHER PART ALREADY IN THE JOINT
9108
DETERMINE SECONDARY ROBOT PICKUP OPTIONS FOR THE NEXT OTHER PART ALREADY IN THE JOINT
9110
CALCULATE PICKUP POINTS FOR THE SELECTED OTHER PART
9112
ADD PICKUP POINTS TO VALID SECONDARY PICKUP POINTS FOR THE SELECTED OTHER PART IF A NEW PART IS BEING ADDED IN THE PRESENT OPERATION
9114
ADD SECONDARY PICKUP POINTS TO THE SELECTED PRIMARY PICKUP POINT
9116
OUTPUT ALL VALID PRIMARY AND SECONDARY ROBOT PICKUP OPTIONS FOR THE PRESENT OPERATION
9118

FIG. 91

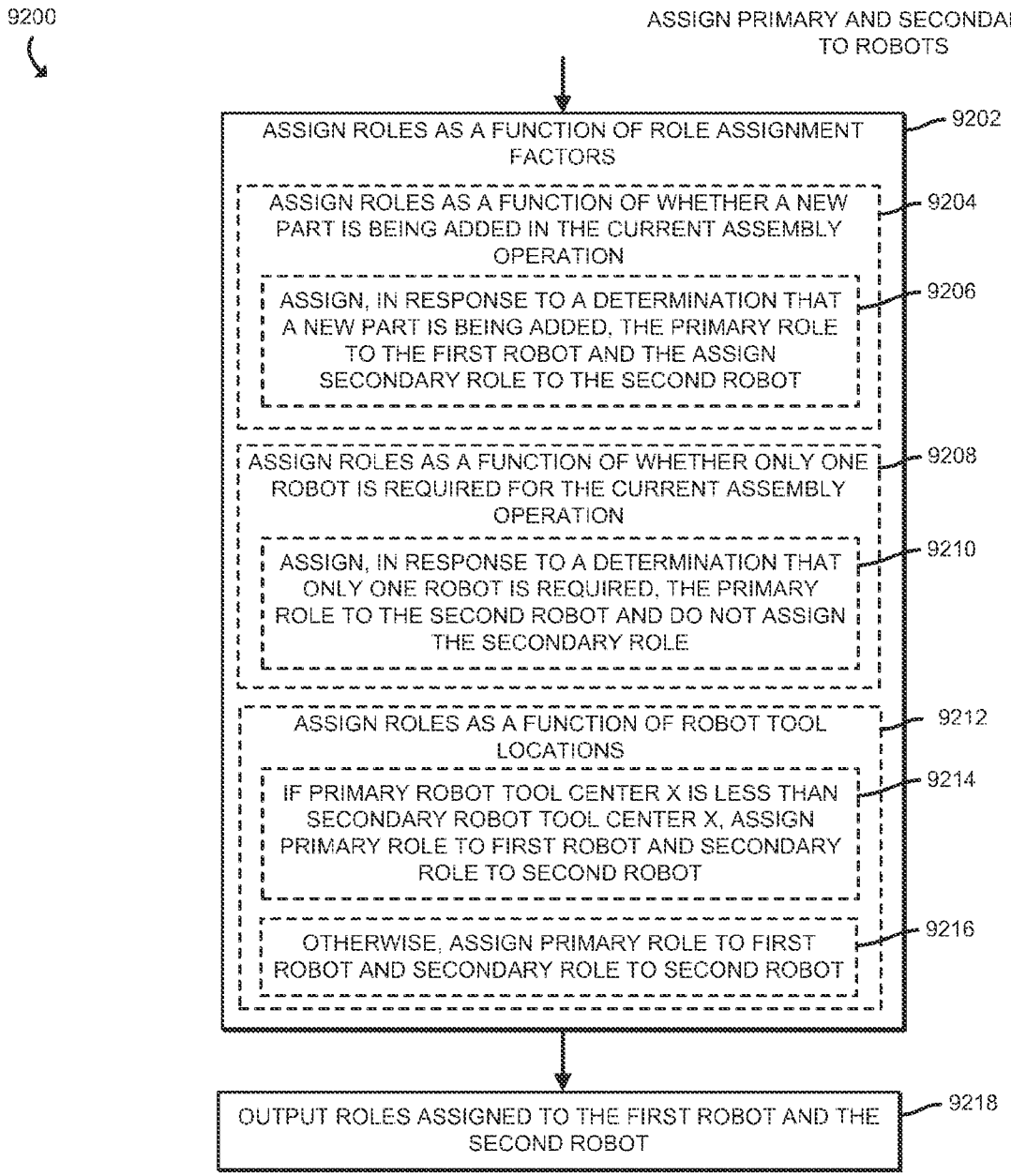

9200
ASSIGN PRIMARY AND SECONDARY ROLES TO ROBOTS
ASSIGN ROLES AS A FUNCTION OF ROLE ASSIGNMENT FACTORS
9202
ASSIGN ROLES AS A FUNCTION OF WHETHER A NEW PART IS BEING ADDED IN THE CURRENT ASSEMBLY OPERATION
9204
ASSIGN, IN RESPONSE TO A DETERMINATION THAT A NEW PART IS BEING ADDED, THE PRIMARY ROLE TO THE FIRST ROBOT AND THE ASSIGN SECONDARY ROLE TO THE SECOND ROBOT
9206
ASSIGN ROLES AS A FUNCTION OF WHETHER ONLY ONE ROBOT IS REQUIRED FOR THE CURRENT ASSEMBLY OPERATION
9208
ASSIGN, IN RESPONSE TO A DETERMINATION THAT ONLY ONE ROBOT IS REQUIRED, THE PRIMARY ROLE TO THE SECOND ROBOT AND DO NOT ASSIGN THE SECONDARY ROLE
9210
ASSIGN ROLES AS A FUNCTION OF ROBOT TOOL LOCATIONS
9212
IF PRIMARY ROBOT TOOL CENTER X IS LESS THAN SECONDARY ROBOT TOOL CENTER X, ASSIGN PRIMARY ROLE TO FIRST ROBOT AND SECONDARY ROLE TO SECOND ROBOT
9214
OTHERWISE, ASSIGN PRIMARY ROLE TO FIRST ROBOT AND SECONDARY ROLE TO SECOND ROBOT
9216
OUTPUT ROLES ASSIGNED TO THE FIRST ROBOT AND THE SECOND ROBOT
9218

FIG. 92

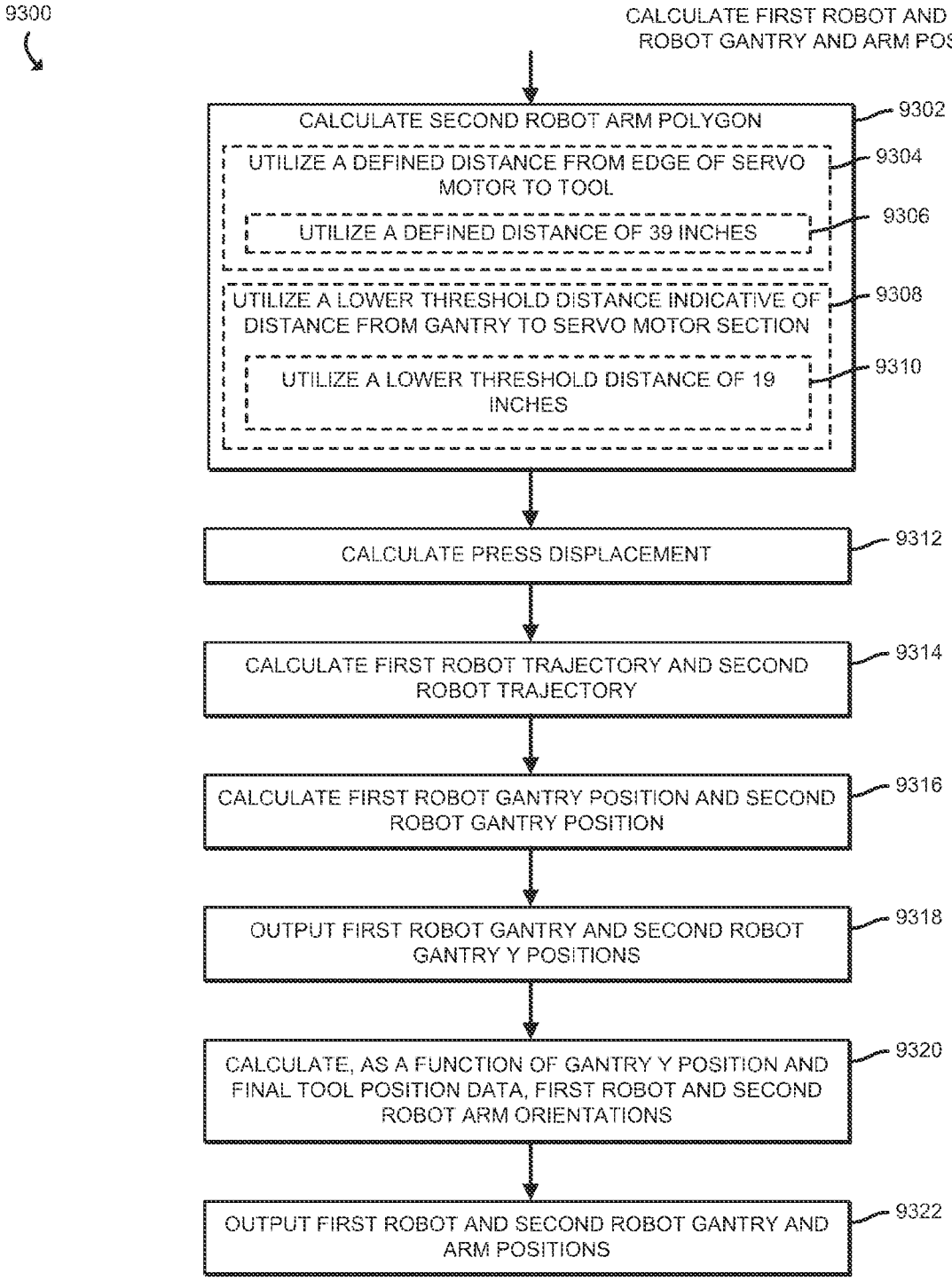

9300
CALCULATE FIRST ROBOT AND SECOND ROBOT GANTRY AND ARM POSITIONS
CALCULATE SECOND ROBOT ARM POLYGON
9302
UTILIZE A DEFINED DISTANCE FROM EDGE OF SERVO MOTOR TO TOOL
9304
UTILIZE A DEFINED DISTANCE OF 39 INCHES
9306
UTILIZE A LOWER THRESHOLD DISTANCE INDICATIVE OF DISTANCE FROM GANTRY TO SERVO MOTOR SECTION
9308
UTILIZE A LOWER THRESHOLD DISTANCE OF 19 INCHES
9310
CALCULATE PRESS DISPLACEMENT
9312
CALCULATE FIRST ROBOT TRAJECTORY AND SECOND ROBOT TRAJECTORY
9314
CALCULATE FIRST ROBOT GANTRY POSITION AND SECOND ROBOT GANTRY POSITION
9316
OUTPUT FIRST ROBOT GANTRY AND SECOND ROBOT GANTRY Y POSITIONS
9318
CALCULATE, AS A FUNCTION OF GANTRY Y POSITION AND FINAL TOOL POSITION DATA, FIRST ROBOT AND SECOND ROBOT ARM ORIENTATIONS
9320
OUTPUT FIRST ROBOT AND SECOND ROBOT GANTRY AND ARM POSITIONS
9322

FIG. 93

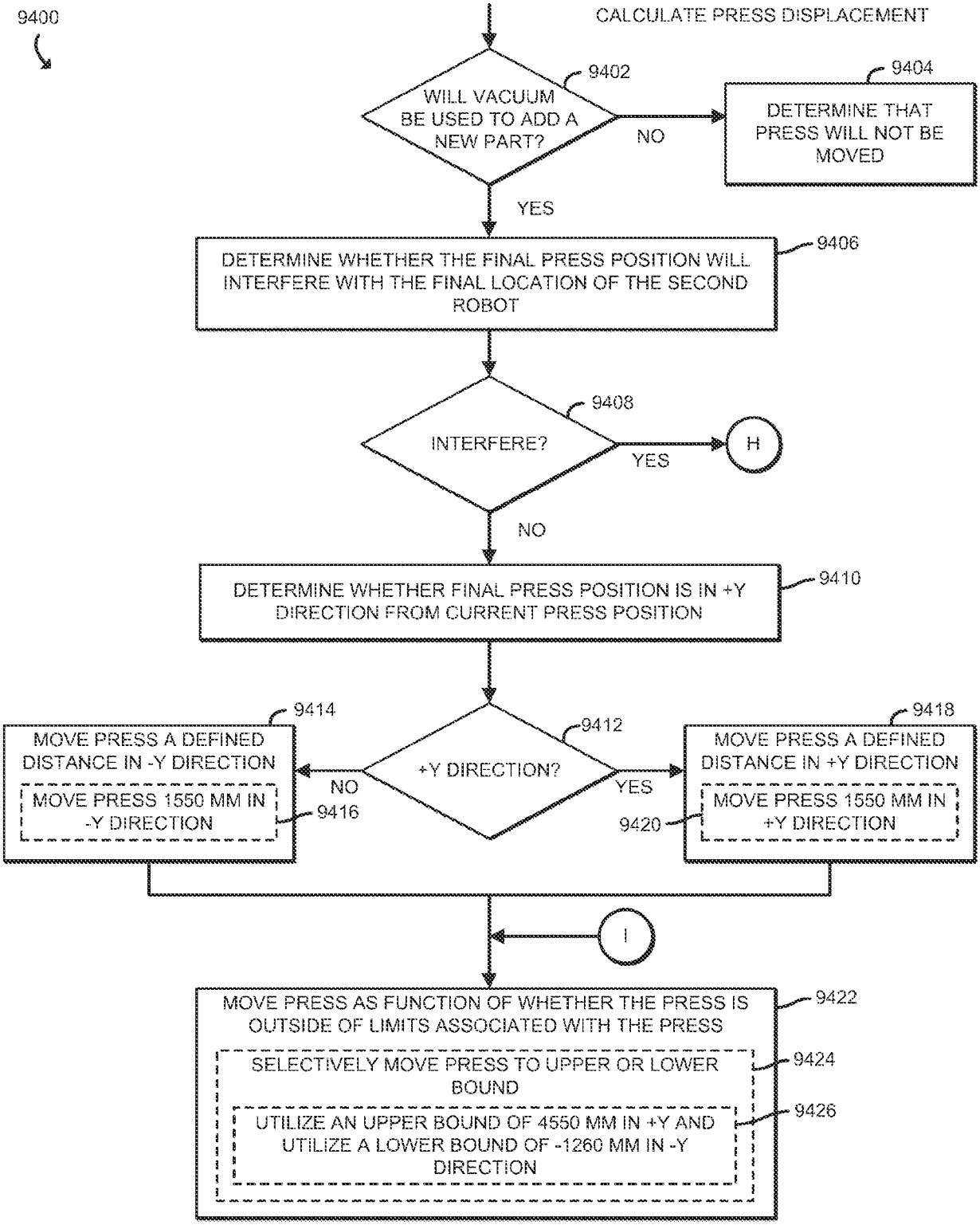

9400
CALCULATE PRESS DISPLACEMENT
9402
WILL VACUUM BE USED TO ADD A NEW PART?
NO
9404
DETERMINE THAT PRESS WILL NOT BE MOVED
YES
DETERMINE WHETHER THE FINAL PRESS POSITION WILL INTERFERE WITH THE FINAL LOCATION OF THE SECOND ROBOT
9406
9408
INTERFERE?
YES
H
NO
DETERMINE WHETHER FINAL PRESS POSITION IS IN +Y DIRECTION FROM CURRENT PRESS POSITION
9410
9414
MOVE PRESS A DEFINED DISTANCE IN -Y DIRECTION
MOVE PRESS 1550 MM IN -Y DIRECTION
9416
NO
9412
+Y DIRECTION?
YES
9418
MOVE PRESS A DEFINED DISTANCE IN +Y DIRECTION
MOVE PRESS 1550 MM IN +Y DIRECTION
9420
I
MOVE PRESS AS FUNCTION OF WHETHER THE PRESS IS OUTSIDE OF LIMITS ASSOCIATED WITH THE PRESS
9422
SELECTIVELY MOVE PRESS TO UPPER OR LOWER BOUND
9424
UTILIZE AN UPPER BOUND OF 4550 MM IN +Y AND UTILIZE A LOWER BOUND OF -1260 MM IN -Y DIRECTION
9426

FIG. 94

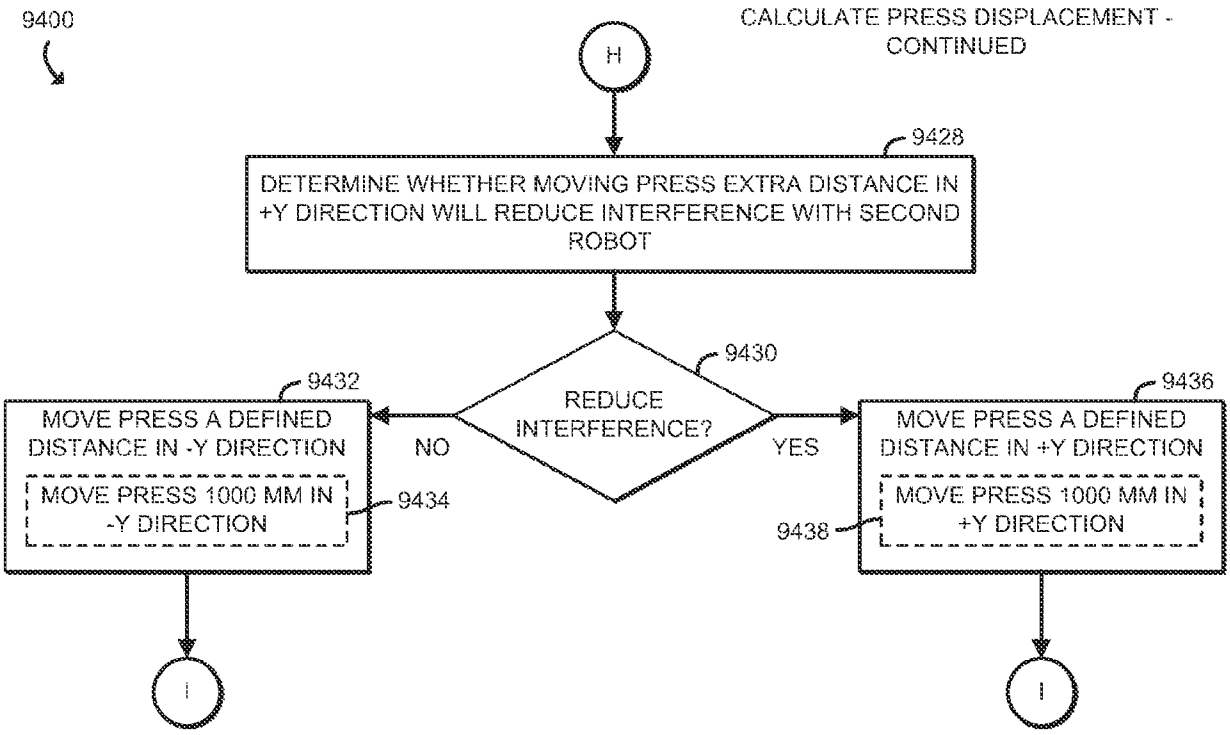

9400
H
CALCULATE PRESS DISPLACEMENT - CONTINUED
9428
DETERMINE WHETHER MOVING PRESS EXTRA DISTANCE IN +Y DIRECTION WILL REDUCE INTERFERENCE WITH SECOND ROBOT
9430
REDUCE INTERFERENCE?
NO
YES
9432
MOVE PRESS A DEFINED DISTANCE IN -Y DIRECTION
MOVE PRESS 1000 MM IN -Y DIRECTION
9434
9436
MOVE PRESS A DEFINED DISTANCE IN +Y DIRECTION
MOVE PRESS 1000 MM IN +Y DIRECTION
9438
I
I

FIG. 95

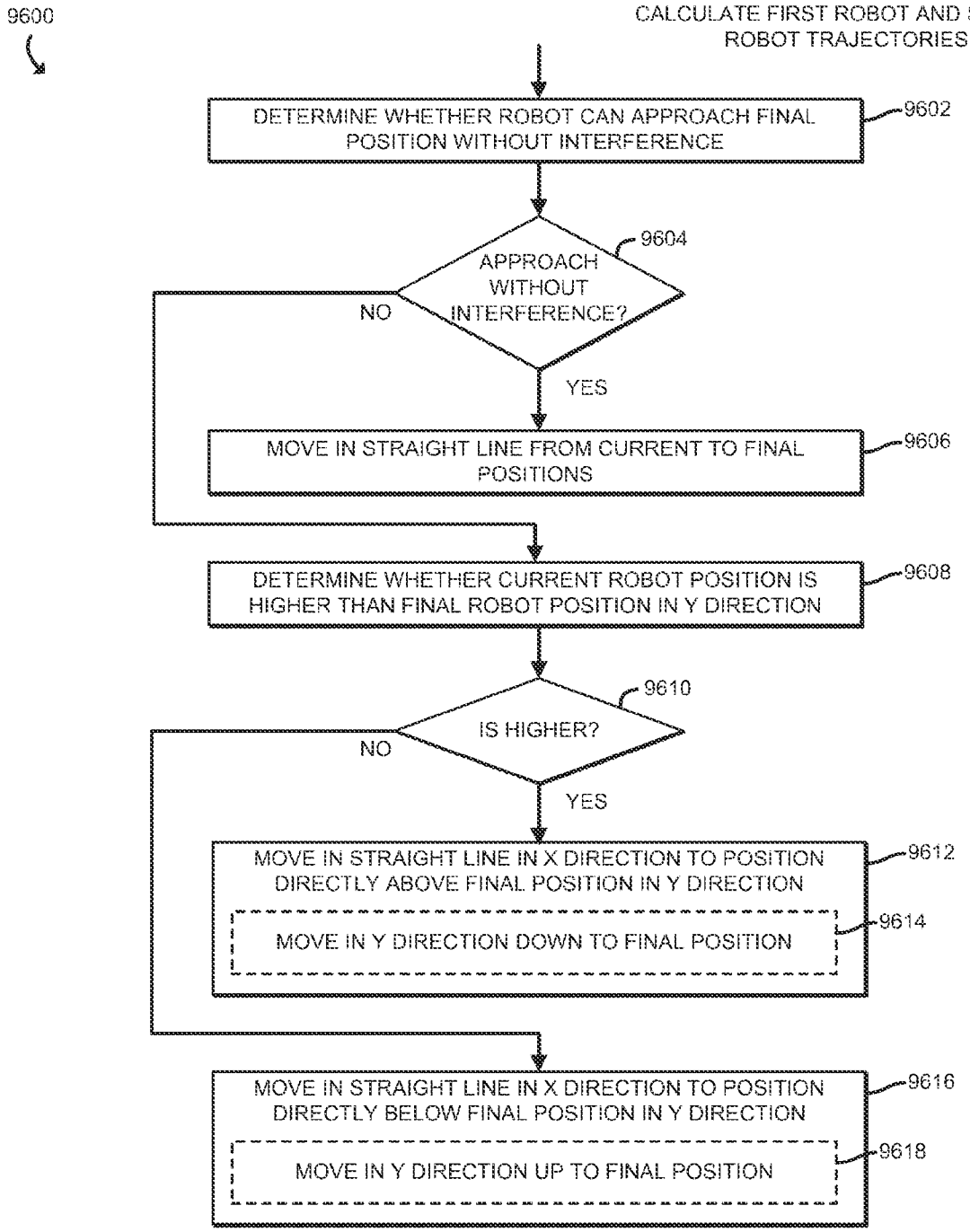

9600
CALCULATE FIRST ROBOT AND SECOND ROBOT TRAJECTORIES
DETERMINE WHETHER ROBOT CAN APPROACH FINAL POSITION WITHOUT INTERFERENCE
9602
APPROACH WITHOUT INTERFERENCE?
9604
NO
YES
MOVE IN STRAIGHT LINE FROM CURRENT TO FINAL POSITIONS
9606
DETERMINE WHETHER CURRENT ROBOT POSITION IS HIGHER THAN FINAL ROBOT POSITION IN Y DIRECTION
9608
IS HIGHER?
9610
NO
YES
MOVE IN STRAIGHT LINE IN X DIRECTION TO POSITION DIRECTLY ABOVE FINAL POSITION IN Y DIRECTION
9612
MOVE IN Y DIRECTION DOWN TO FINAL POSITION
9614
MOVE IN STRAIGHT LINE IN X DIRECTION TO POSITION DIRECTLY BELOW FINAL POSITION IN Y DIRECTION
9616
MOVE IN Y DIRECTION UP TO FINAL POSITION
9618

FIG. 96

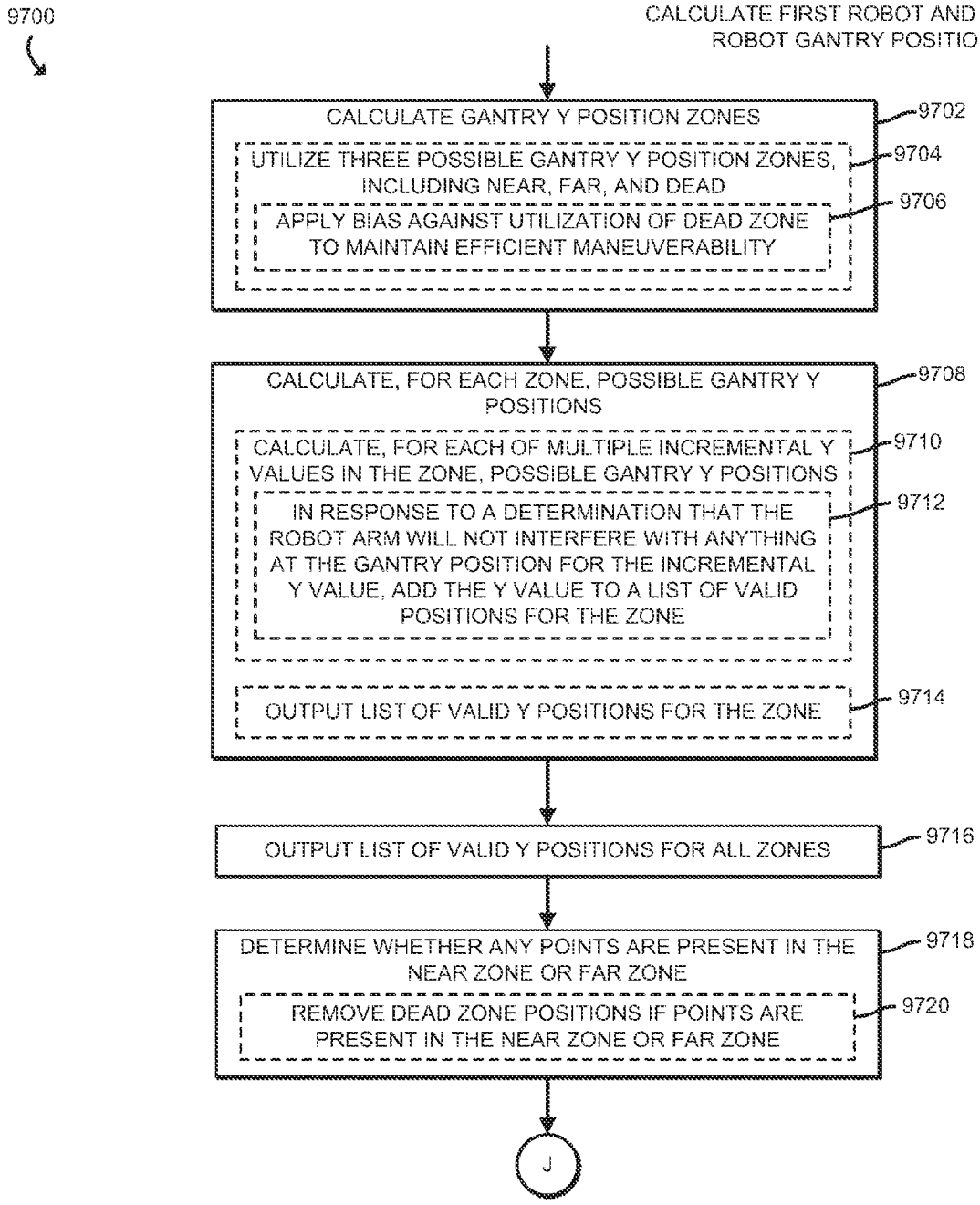

9700
CALCULATE FIRST ROBOT AND SECOND ROBOT GANTRY POSITIONS
CALCULATE GANTRY Y POSITION ZONES
9702
UTILIZE THREE POSSIBLE GANTRY Y POSITION ZONES, INCLUDING NEAR, FAR, AND DEAD
9704
APPLY BIAS AGAINST UTILIZATION OF DEAD ZONE TO MAINTAIN EFFICIENT MANEUVERABILITY
9706
CALCULATE, FOR EACH ZONE, POSSIBLE GANTRY Y POSITIONS
9708
CALCULATE, FOR EACH OF MULTIPLE INCREMENTAL Y VALUES IN THE ZONE, POSSIBLE GANTRY Y POSITIONS
9710
IN RESPONSE TO A DETERMINATION THAT THE ROBOT ARM WILL NOT INTERFERE WITH ANYTHING AT THE GANTRY POSITION FOR THE INCREMENTAL Y VALUE, ADD THE Y VALUE TO A LIST OF VALID POSITIONS FOR THE ZONE
9712
OUTPUT LIST OF VALID Y POSITIONS FOR THE ZONE
9714
OUTPUT LIST OF VALID Y POSITIONS FOR ALL ZONES
9716
DETERMINE WHETHER ANY POINTS ARE PRESENT IN THE NEAR ZONE OR FAR ZONE
9718
REMOVE DEAD ZONE POSITIONS IF POINTS ARE PRESENT IN THE NEAR ZONE OR FAR ZONE
9720
J

FIG. 97

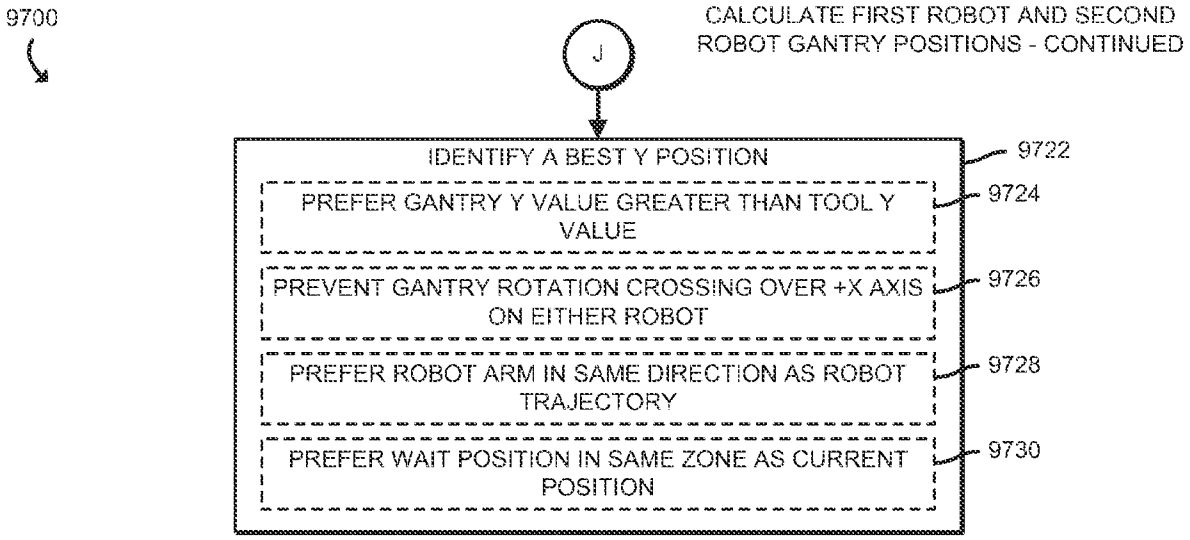

9700
J
CALCULATE FIRST ROBOT AND SECOND ROBOT GANTRY POSITIONS - CONTINUED
IDENTIFY A BEST Y POSITION
9722
PREFER GANTRY Y VALUE GREATER THAN TOOL Y VALUE
9724
PREVENT GANTRY ROTATION CROSSING OVER +X AXIS ON EITHER ROBOT
9726
PREFER ROBOT ARM IN SAME DIRECTION AS ROBOT TRAJECTORY
9728
PREFER WAIT POSITION IN SAME ZONE AS CURRENT POSITION
9730

FIG. 98

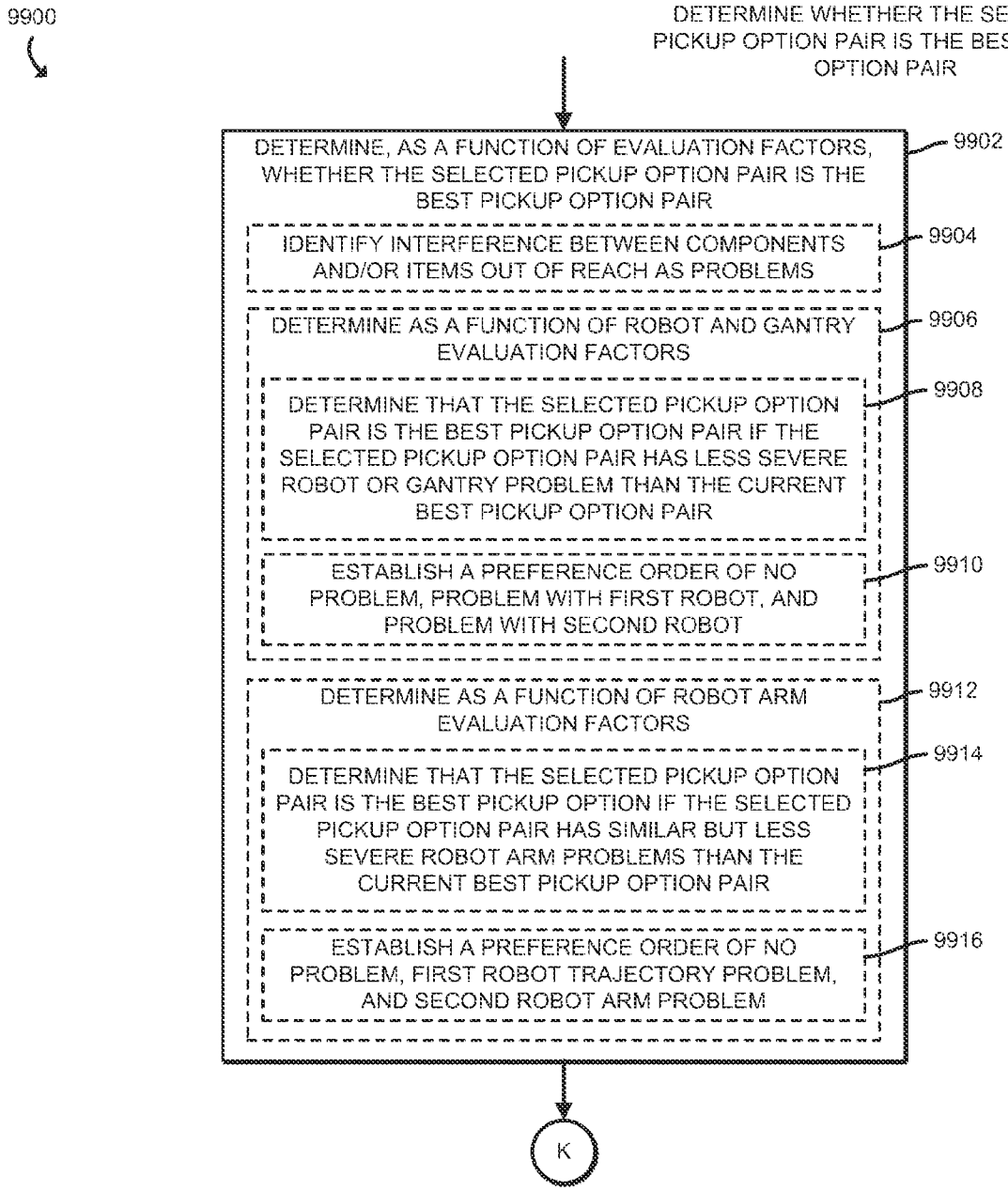

9900
DETERMINE WHETHER THE SELECTED PICKUP OPTION PAIR IS THE BEST PICKUP OPTION PAIR
DETERMINE, AS A FUNCTION OF EVALUATION FACTORS, WHETHER THE SELECTED PICKUP OPTION PAIR IS THE BEST PICKUP OPTION PAIR
9902
IDENTIFY INTERFERENCE BETWEEN COMPONENTS AND/OR ITEMS OUT OF REACH AS PROBLEMS
9904
DETERMINE AS A FUNCTION OF ROBOT AND GANTRY EVALUATION FACTORS
9906
DETERMINE THAT THE SELECTED PICKUP OPTION PAIR IS THE BEST PICKUP OPTION PAIR IF THE SELECTED PICKUP OPTION PAIR HAS LESS SEVERE ROBOT OR GANTRY PROBLEM THAN THE CURRENT BEST PICKUP OPTION PAIR
9908
ESTABLISH A PREFERENCE ORDER OF NO PROBLEM, PROBLEM WITH FIRST ROBOT, AND PROBLEM WITH SECOND ROBOT
9910
DETERMINE AS A FUNCTION OF ROBOT ARM EVALUATION FACTORS
9912
DETERMINE THAT THE SELECTED PICKUP OPTION PAIR IS THE BEST PICKUP OPTION IF THE SELECTED PICKUP OPTION PAIR HAS SIMILAR BUT LESS SEVERE ROBOT ARM PROBLEMS THAN THE CURRENT BEST PICKUP OPTION PAIR
9914
ESTABLISH A PREFERENCE ORDER OF NO PROBLEM, FIRST ROBOT TRAJECTORY PROBLEM, AND SECOND ROBOT ARM PROBLEM
9916
K

FIG. 99

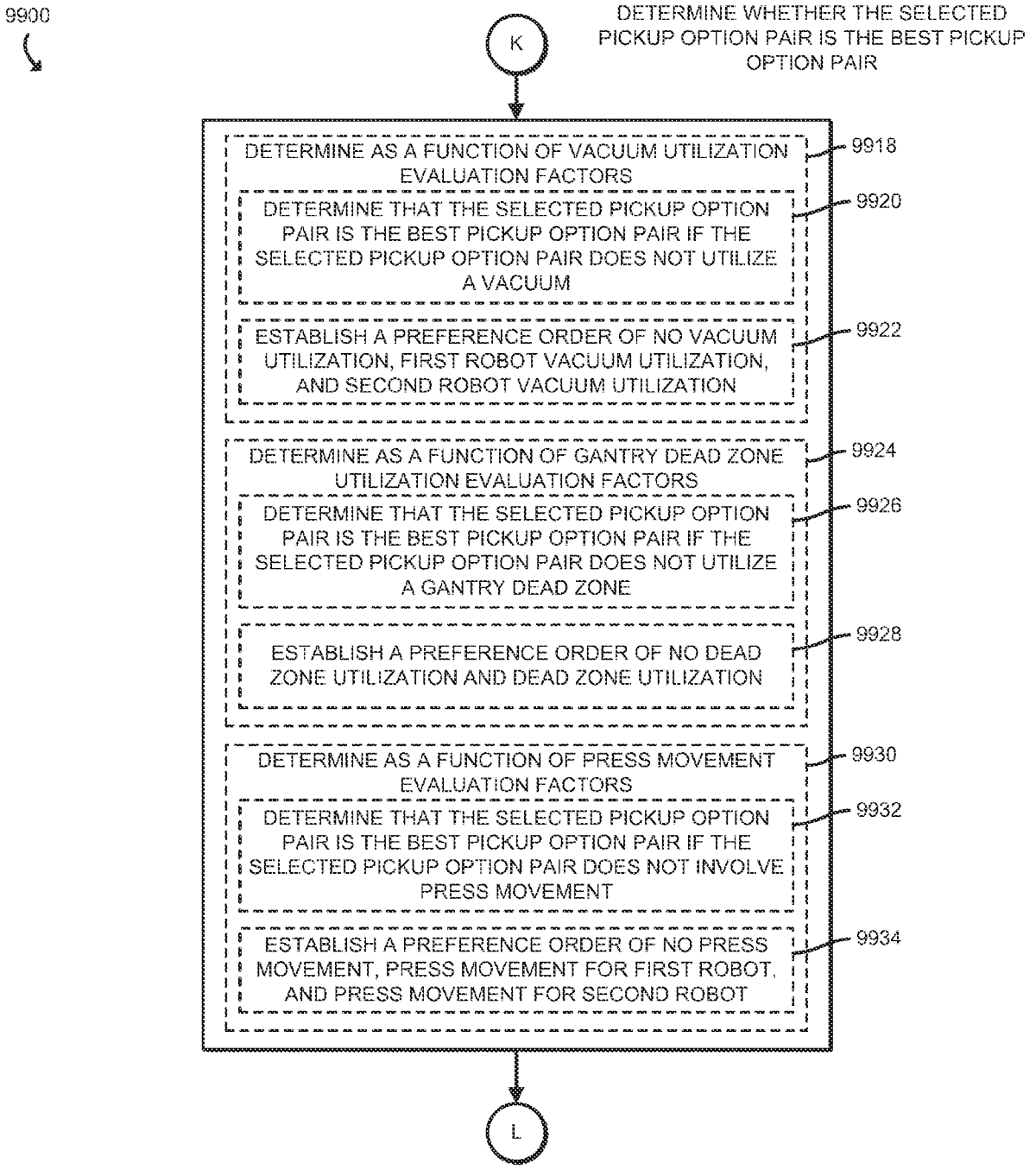

9900
K
DETERMINE WHETHER THE SELECTED PICKUP OPTION PAIR IS THE BEST PICKUP OPTION PAIR
DETERMINE AS A FUNCTION OF VACUUM UTILIZATION EVALUATION FACTORS
9918
DETERMINE THAT THE SELECTED PICKUP OPTION PAIR IS THE BEST PICKUP OPTION PAIR IF THE SELECTED PICKUP OPTION PAIR DOES NOT UTILIZE A VACUUM
9920
ESTABLISH A PREFERENCE ORDER OF NO VACUUM UTILIZATION, FIRST ROBOT VACUUM UTILIZATION, AND SECOND ROBOT VACUUM UTILIZATION
9922
DETERMINE AS A FUNCTION OF GANTRY DEAD ZONE UTILIZATION EVALUATION FACTORS
9924
DETERMINE THAT THE SELECTED PICKUP OPTION PAIR IS THE BEST PICKUP OPTION PAIR IF THE SELECTED PICKUP OPTION PAIR DOES NOT UTILIZE A GANTRY DEAD ZONE
9926
ESTABLISH A PREFERENCE ORDER OF NO DEAD ZONE UTILIZATION AND DEAD ZONE UTILIZATION
9928
DETERMINE AS A FUNCTION OF PRESS MOVEMENT EVALUATION FACTORS
9930
DETERMINE THAT THE SELECTED PICKUP OPTION PAIR IS THE BEST PICKUP OPTION PAIR IF THE SELECTED PICKUP OPTION PAIR DOES NOT INVOLVE PRESS MOVEMENT
9932
ESTABLISH A PREFERENCE ORDER OF NO PRESS MOVEMENT, PRESS MOVEMENT FOR FIRST ROBOT, AND PRESS MOVEMENT FOR SECOND ROBOT
9934
L

FIG. 100

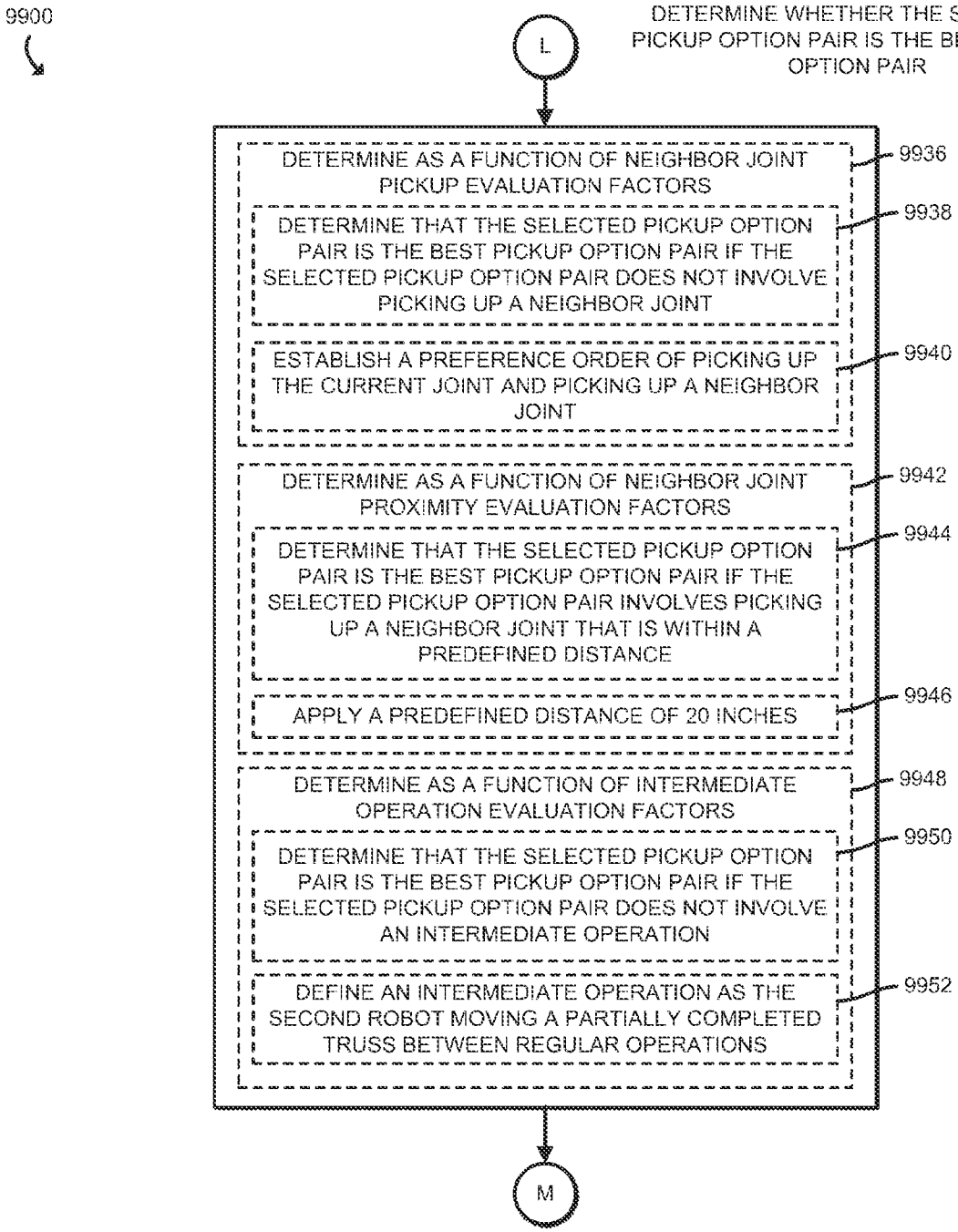

9900
L
DETERMINE WHETHER THE SELECTED PICKUP OPTION PAIR IS THE BEST PICKUP OPTION PAIR
DETERMINE AS A FUNCTION OF NEIGHBOR JOINT PICKUP EVALUATION FACTORS
9936
DETERMINE THAT THE SELECTED PICKUP OPTION PAIR IS THE BEST PICKUP OPTION PAIR IF THE SELECTED PICKUP OPTION PAIR DOES NOT INVOLVE PICKING UP A NEIGHBOR JOINT
9938
ESTABLISH A PREFERENCE ORDER OF PICKING UP THE CURRENT JOINT AND PICKING UP A NEIGHBOR JOINT
9940
DETERMINE AS A FUNCTION OF NEIGHBOR JOINT PROXIMITY EVALUATION FACTORS
9942
DETERMINE THAT THE SELECTED PICKUP OPTION PAIR IS THE BEST PICKUP OPTION PAIR IF THE SELECTED PICKUP OPTION PAIR INVOLVES PICKING UP A NEIGHBOR JOINT THAT IS WITHIN A PREDEFINED DISTANCE
9944
APPLY A PREDEFINED DISTANCE OF 20 INCHES
9946
DETERMINE AS A FUNCTION OF INTERMEDIATE OPERATION EVALUATION FACTORS
9948
DETERMINE THAT THE SELECTED PICKUP OPTION PAIR IS THE BEST PICKUP OPTION PAIR IF THE SELECTED PICKUP OPTION PAIR DOES NOT INVOLVE AN INTERMEDIATE OPERATION
9950
DEFINE AN INTERMEDIATE OPERATION AS THE SECOND ROBOT MOVING A PARTIALLY COMPLETED TRUSS BETWEEN REGULAR OPERATIONS
9952
M

FIG. 101

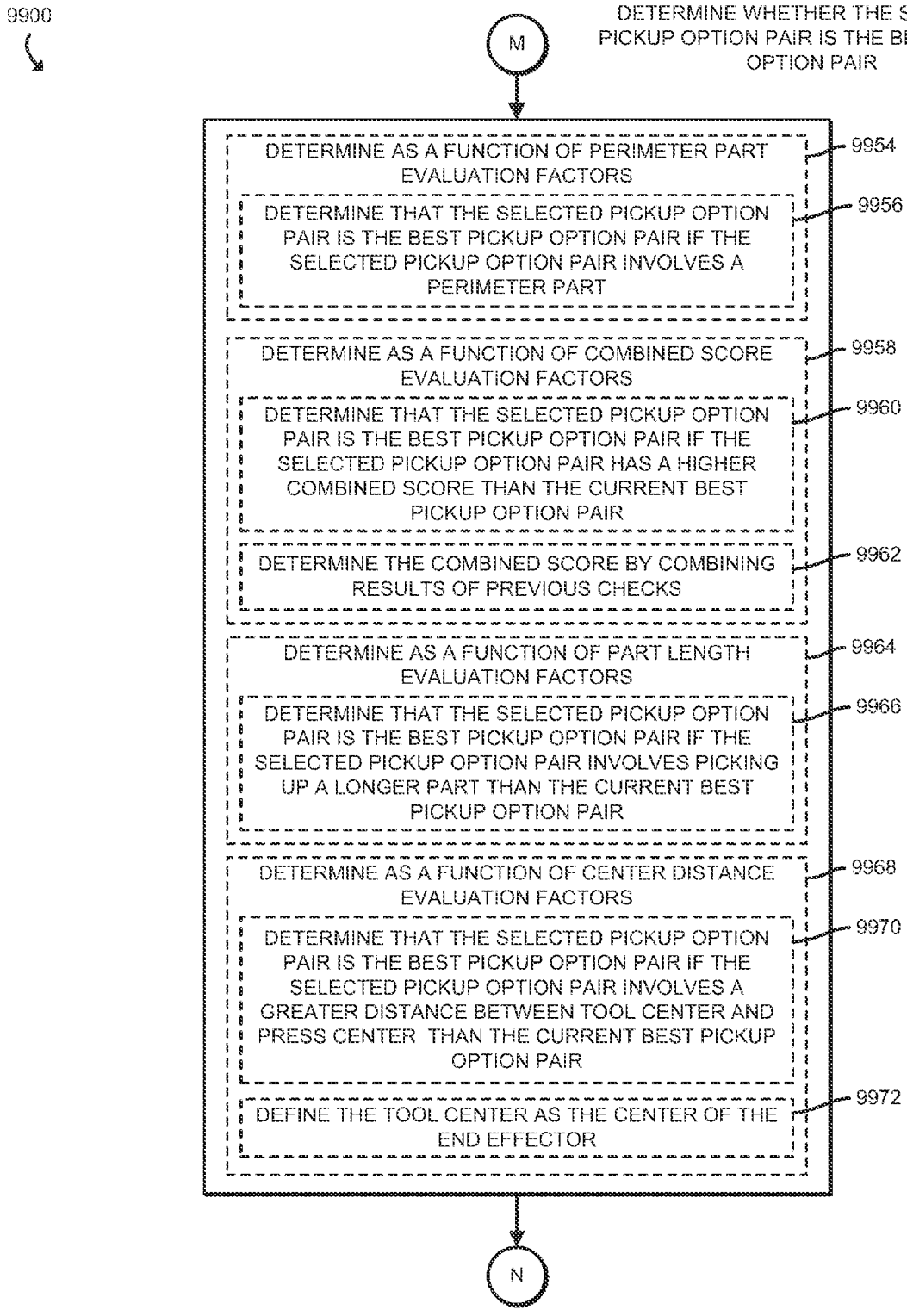

9900
M
DETERMINE WHETHER THE SELECTED PICKUP OPTION PAIR IS THE BEST PICKUP OPTION PAIR
DETERMINE AS A FUNCTION OF PERIMETER PART EVALUATION FACTORS
9954
DETERMINE THAT THE SELECTED PICKUP OPTION PAIR IS THE BEST PICKUP OPTION PAIR IF THE SELECTED PICKUP OPTION PAIR INVOLVES A PERIMETER PART
9956
DETERMINE AS A FUNCTION OF COMBINED SCORE EVALUATION FACTORS
9958
DETERMINE THAT THE SELECTED PICKUP OPTION PAIR IS THE BEST PICKUP OPTION PAIR IF THE SELECTED PICKUP OPTION PAIR HAS A HIGHER COMBINED SCORE THAN THE CURRENT BEST PICKUP OPTION PAIR
9960
DETERMINE THE COMBINED SCORE BY COMBINING RESULTS OF PREVIOUS CHECKS
9962
DETERMINE AS A FUNCTION OF PART LENGTH EVALUATION FACTORS
9964
DETERMINE THAT THE SELECTED PICKUP OPTION PAIR IS THE BEST PICKUP OPTION PAIR IF THE SELECTED PICKUP OPTION PAIR INVOLVES PICKING UP A LONGER PART THAN THE CURRENT BEST PICKUP OPTION PAIR
9966
DETERMINE AS A FUNCTION OF CENTER DISTANCE EVALUATION FACTORS
9968
DETERMINE THAT THE SELECTED PICKUP OPTION PAIR IS THE BEST PICKUP OPTION PAIR IF THE SELECTED PICKUP OPTION PAIR INVOLVES A GREATER DISTANCE BETWEEN TOOL CENTER AND PRESS CENTER THAN THE CURRENT BEST PICKUP OPTION PAIR
9970
DEFINE THE TOOL CENTER AS THE CENTER OF THE END EFFECTOR
9972
N

FIG. 102

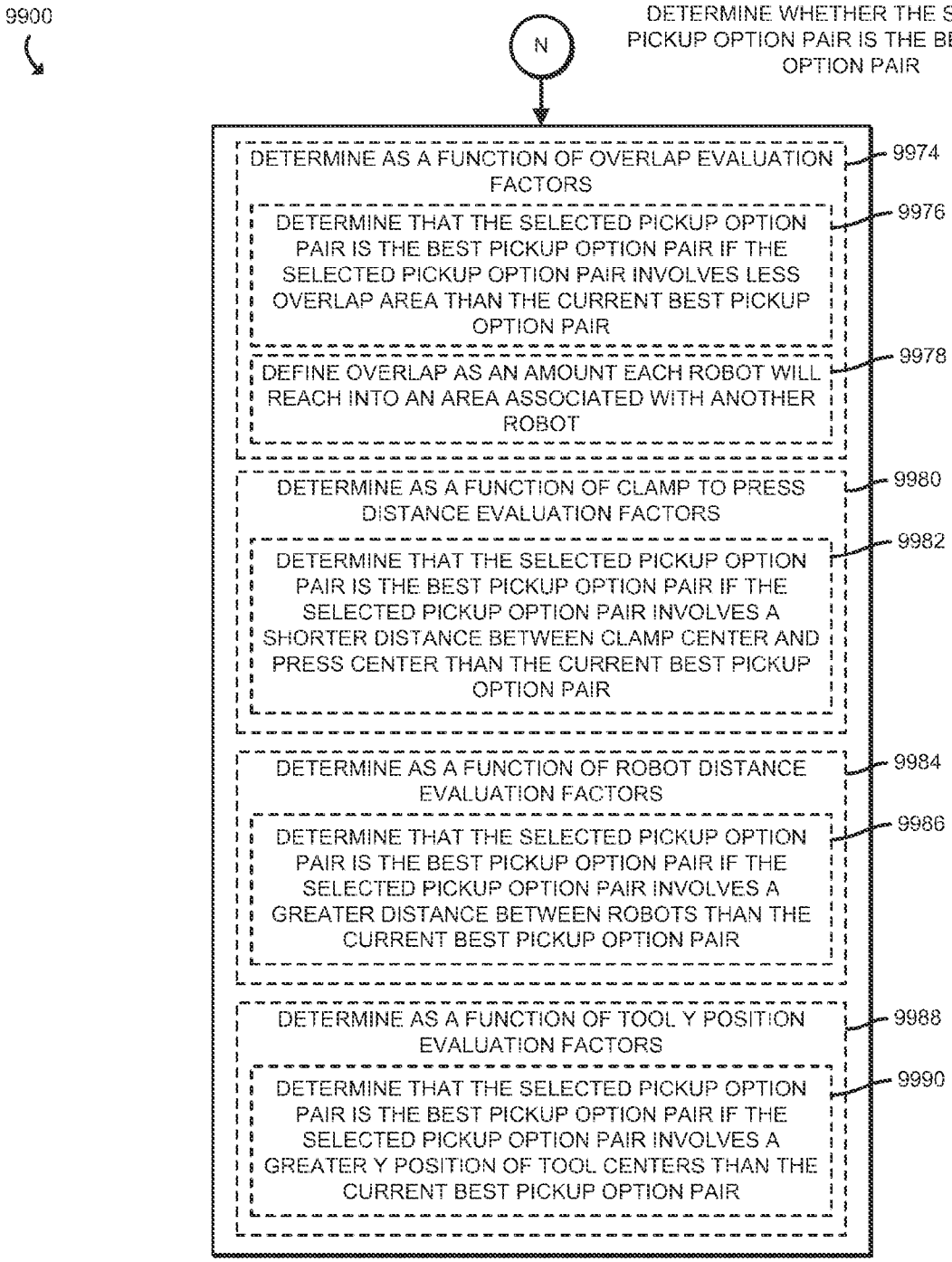

9900
N
DETERMINE WHETHER THE SELECTED PICKUP OPTION PAIR IS THE BEST PICKUP OPTION PAIR
DETERMINE AS A FUNCTION OF OVERLAP EVALUATION FACTORS
9974
DETERMINE THAT THE SELECTED PICKUP OPTION PAIR IS THE BEST PICKUP OPTION PAIR IF THE SELECTED PICKUP OPTION PAIR INVOLVES LESS OVERLAP AREA THAN THE CURRENT BEST PICKUP OPTION PAIR
9976
DEFINE OVERLAP AS AN AMOUNT EACH ROBOT WILL REACH INTO AN AREA ASSOCIATED WITH ANOTHER ROBOT
9978
DETERMINE AS A FUNCTION OF CLAMP TO PRESS DISTANCE EVALUATION FACTORS
9980
DETERMINE THAT THE SELECTED PICKUP OPTION PAIR IS THE BEST PICKUP OPTION PAIR IF THE SELECTED PICKUP OPTION PAIR INVOLVES A SHORTER DISTANCE BETWEEN CLAMP CENTER AND PRESS CENTER THAN THE CURRENT BEST PICKUP OPTION PAIR
9982
DETERMINE AS A FUNCTION OF ROBOT DISTANCE EVALUATION FACTORS
9984
DETERMINE THAT THE SELECTED PICKUP OPTION PAIR IS THE BEST PICKUP OPTION PAIR IF THE SELECTED PICKUP OPTION PAIR INVOLVES A GREATER DISTANCE BETWEEN ROBOTS THAN THE CURRENT BEST PICKUP OPTION PAIR
9986
DETERMINE AS A FUNCTION OF TOOL Y POSITION EVALUATION FACTORS
9988
DETERMINE THAT THE SELECTED PICKUP OPTION PAIR IS THE BEST PICKUP OPTION PAIR IF THE SELECTED PICKUP OPTION PAIR INVOLVES A GREATER Y POSITION OF TOOL CENTERS THAN THE CURRENT BEST PICKUP OPTION PAIR
9990

FIG. 103

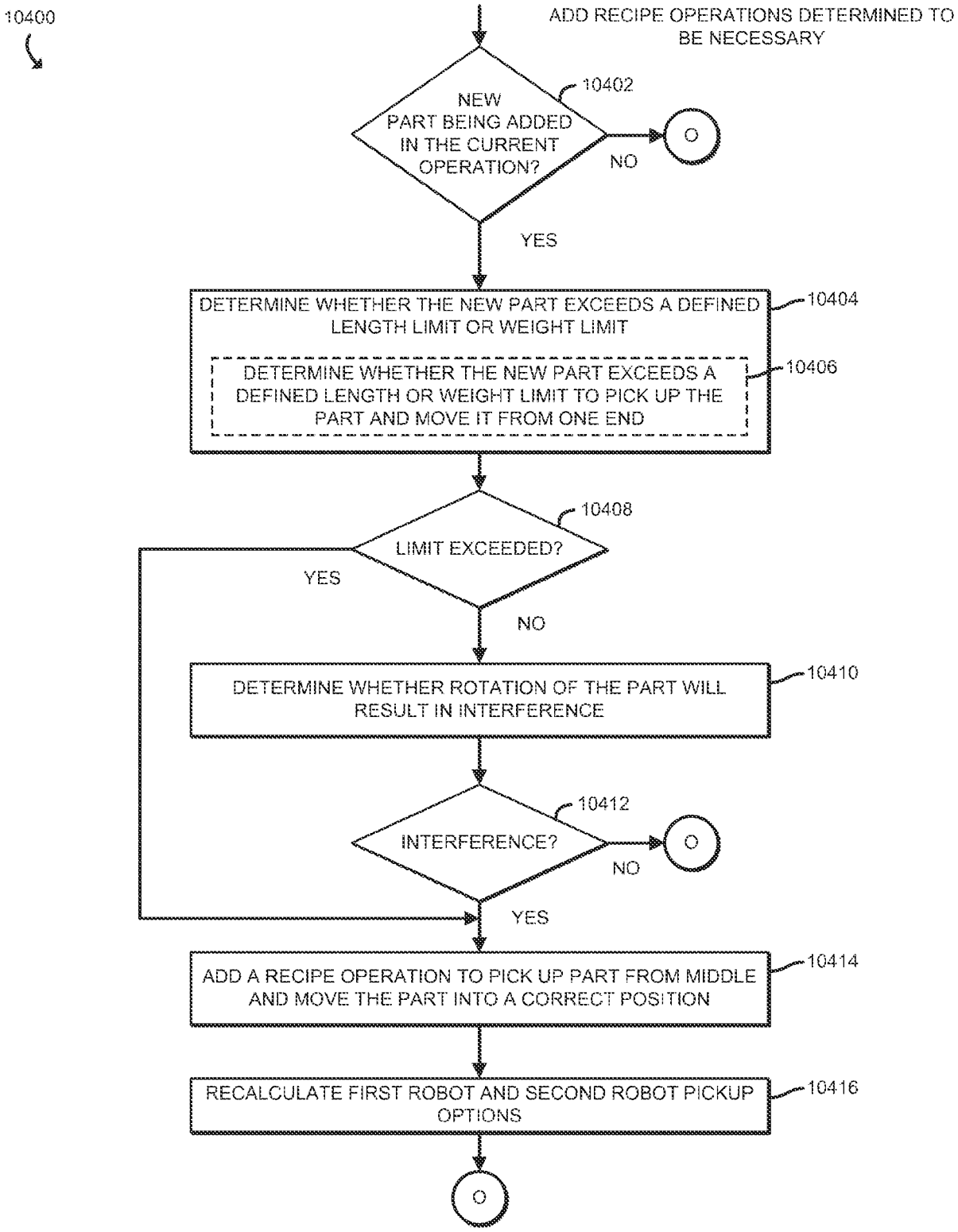

10400
ADD RECIPE OPERATIONS DETERMINED TO BE NECESSARY
NEW PART BEING ADDED IN THE CURRENT OPERATION?
10402
NO
O
YES
DETERMINE WHETHER THE NEW PART EXCEEDS A DEFINED LENGTH LIMIT OR WEIGHT LIMIT
10404
DETERMINE WHETHER THE NEW PART EXCEEDS A DEFINED LENGTH OR WEIGHT LIMIT TO PICK UP THE PART AND MOVE IT FROM ONE END
10406
LIMIT EXCEEDED?
10408
YES
NO
DETERMINE WHETHER ROTATION OF THE PART WILL RESULT IN INTERFERENCE
10410
INTERFERENCE?
10412
NO
O
YES
ADD A RECIPE OPERATION TO PICK UP PART FROM MIDDLE AND MOVE THE PART INTO A CORRECT POSITION
10414
RECALCULATE FIRST ROBOT AND SECOND ROBOT PICKUP OPTIONS
10416
O

FIG. 104

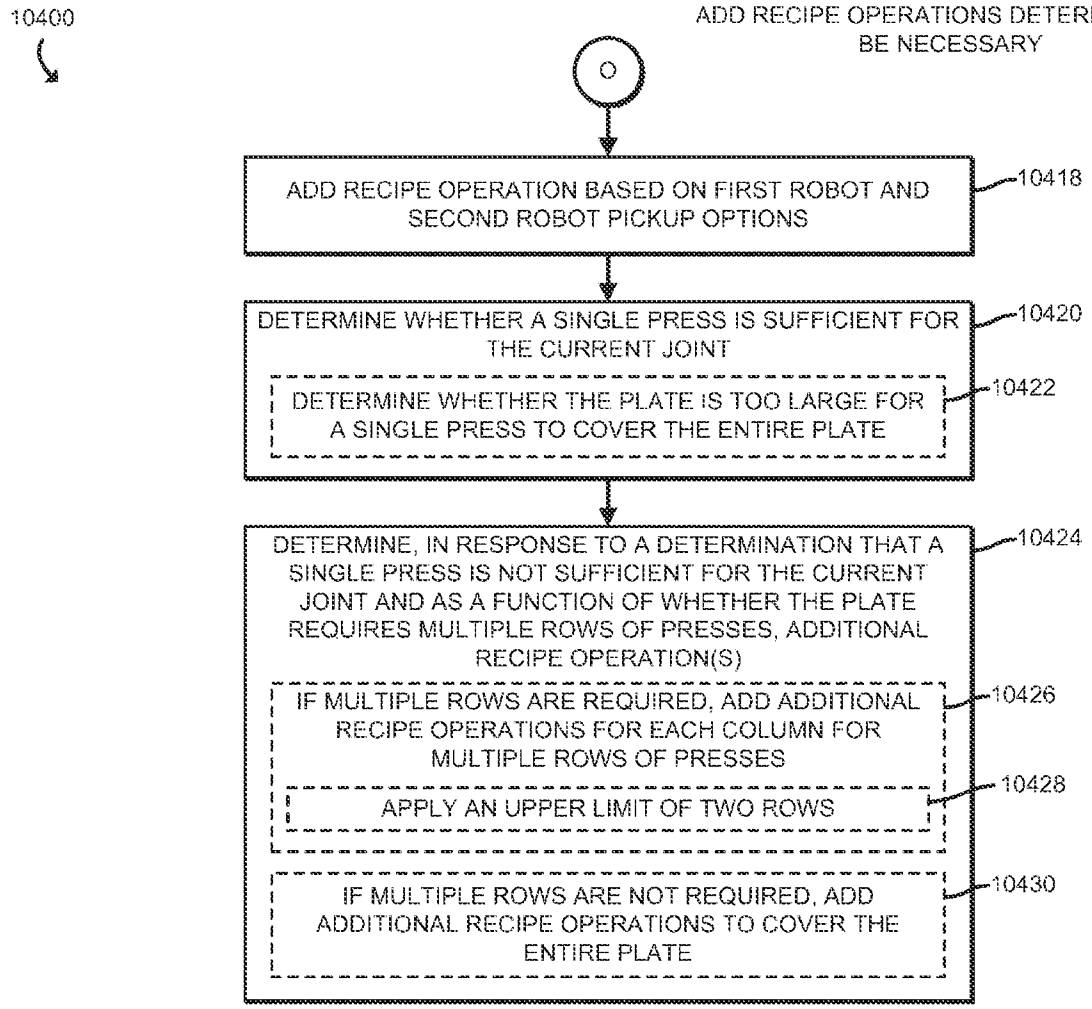

10400
ADD RECIPE OPERATIONS DETERMINED TO BE NECESSARY
O
ADD RECIPE OPERATION BASED ON FIRST ROBOT AND SECOND ROBOT PICKUP OPTIONS
10418
DETERMINE WHETHER A SINGLE PRESS IS SUFFICIENT FOR THE CURRENT JOINT
10420
DETERMINE WHETHER THE PLATE IS TOO LARGE FOR A SINGLE PRESS TO COVER THE ENTIRE PLATE
10422
DETERMINE, IN RESPONSE TO A DETERMINATION THAT A SINGLE PRESS IS NOT SUFFICIENT FOR THE CURRENT JOINT AND AS A FUNCTION OF WHETHER THE PLATE REQUIRES MULTIPLE ROWS OF PRESSES, ADDITIONAL RECIPE OPERATION(S)
10424
IF MULTIPLE ROWS ARE REQUIRED, ADD ADDITIONAL RECIPE OPERATIONS FOR EACH COLUMN FOR MULTIPLE ROWS OF PRESSES
10426
APPLY AN UPPER LIMIT OF TWO ROWS
10428
IF MULTIPLE ROWS ARE NOT REQUIRED, ADD ADDITIONAL RECIPE OPERATIONS TO COVER THE ENTIRE PLATE
10430

FIG. 105

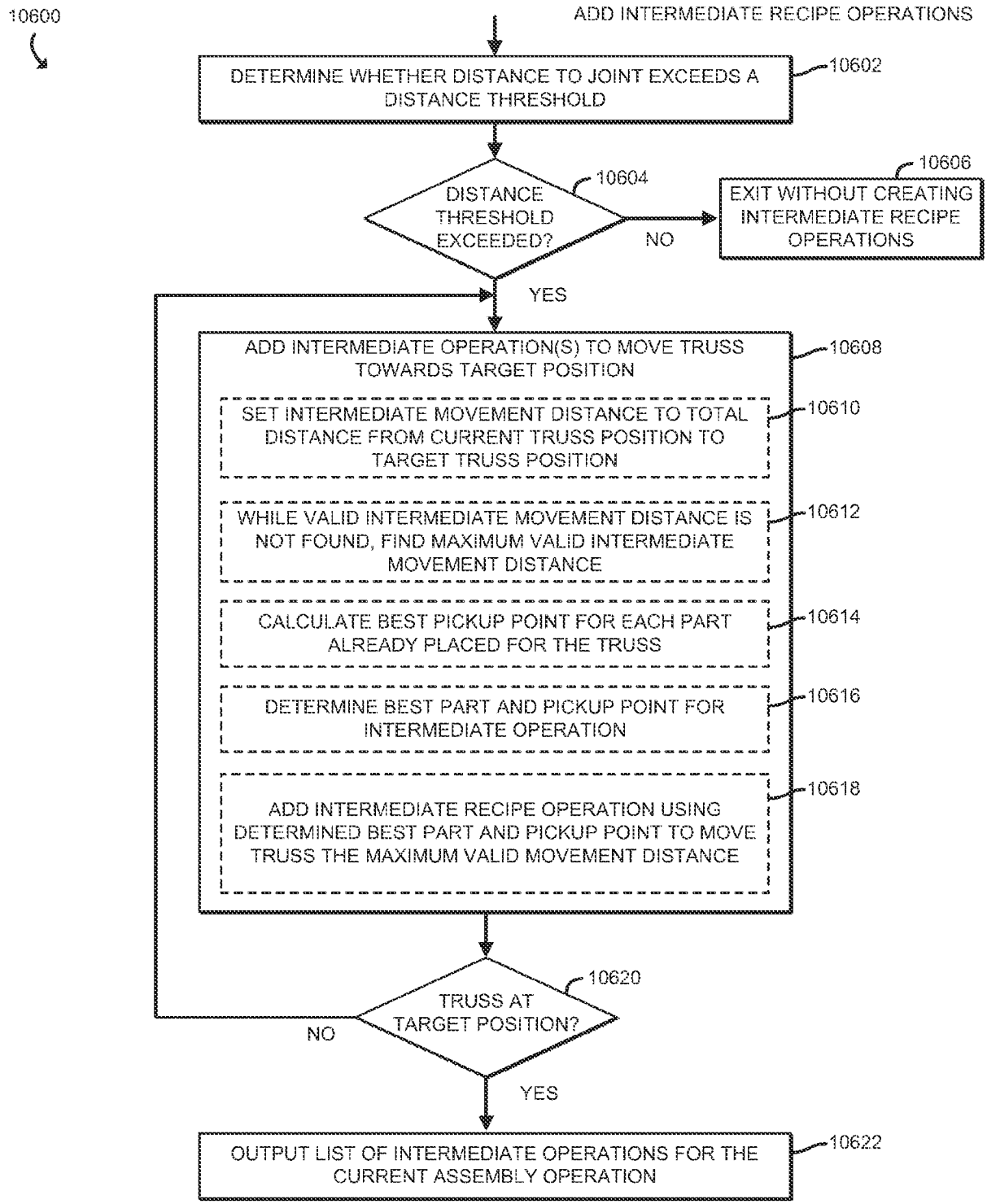

10600
ADD INTERMEDIATE RECIPE OPERATIONS
DETERMINE WHETHER DISTANCE TO JOINT EXCEEDS A DISTANCE THRESHOLD
10602
DISTANCE THRESHOLD EXCEEDED?
10604
NO
EXIT WITHOUT CREATING INTERMEDIATE RECIPE OPERATIONS
10606
YES
ADD INTERMEDIATE OPERATION(S) TO MOVE TRUSS TOWARDS TARGET POSITION
10608
SET INTERMEDIATE MOVEMENT DISTANCE TO TOTAL DISTANCE FROM CURRENT TRUSS POSITION TO TARGET TRUSS POSITION
10610
WHILE VALID INTERMEDIATE MOVEMENT DISTANCE IS NOT FOUND, FIND MAXIMUM VALID INTERMEDIATE MOVEMENT DISTANCE
10612
CALCULATE BEST PICKUP POINT FOR EACH PART ALREADY PLACED FOR THE TRUSS
10614
DETERMINE BEST PART AND PICKUP POINT FOR INTERMEDIATE OPERATION
10616
ADD INTERMEDIATE RECIPE OPERATION USING DETERMINED BEST PART AND PICKUP POINT TO MOVE TRUSS THE MAXIMUM VALID MOVEMENT DISTANCE
10618
TRUSS AT TARGET POSITION?
10620
NO
YES
OUTPUT LIST OF INTERMEDIATE OPERATIONS FOR THE CURRENT ASSEMBLY OPERATION
10622

FIG. 106

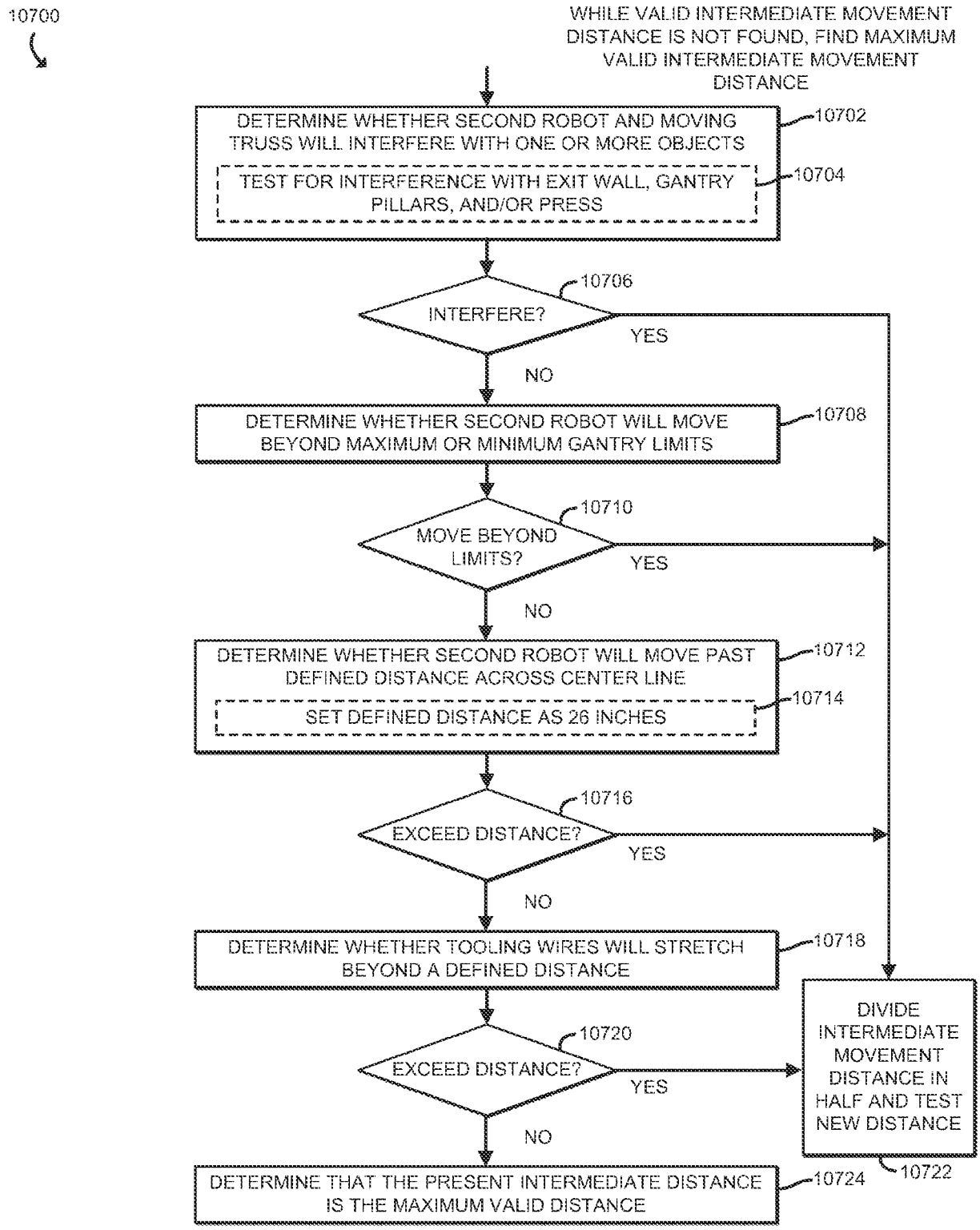

10700
WHILE VALID INTERMEDIATE MOVEMENT DISTANCE IS NOT FOUND, FIND MAXIMUM VALID INTERMEDIATE MOVEMENT DISTANCE
DETERMINE WHETHER SECOND ROBOT AND MOVING TRUSS WILL INTERFERE WITH ONE OR MORE OBJECTS
10702
TEST FOR INTERFERENCE WITH EXIT WALL, GANTRY PILLARS, AND/OR PRESS
10704
INTERFERE?
10706
YES
NO
DETERMINE WHETHER SECOND ROBOT WILL MOVE BEYOND MAXIMUM OR MINIMUM GANTRY LIMITS
10708
MOVE BEYOND LIMITS?
10710
YES
NO
DETERMINE WHETHER SECOND ROBOT WILL MOVE PAST DEFINED DISTANCE ACROSS CENTER LINE
10712
SET DEFINED DISTANCE AS 26 INCHES
10714
EXCEED DISTANCE?
10716
YES
NO
DETERMINE WHETHER TOOLING WIRES WILL STRETCH BEYOND A DEFINED DISTANCE
10718
EXCEED DISTANCE?
10720
YES
NO
DIVIDE INTERMEDIATE MOVEMENT DISTANCE IN HALF AND TEST NEW DISTANCE
10722
DETERMINE THAT THE PRESENT INTERMEDIATE DISTANCE IS THE MAXIMUM VALID DISTANCE
10724

FIG. 107

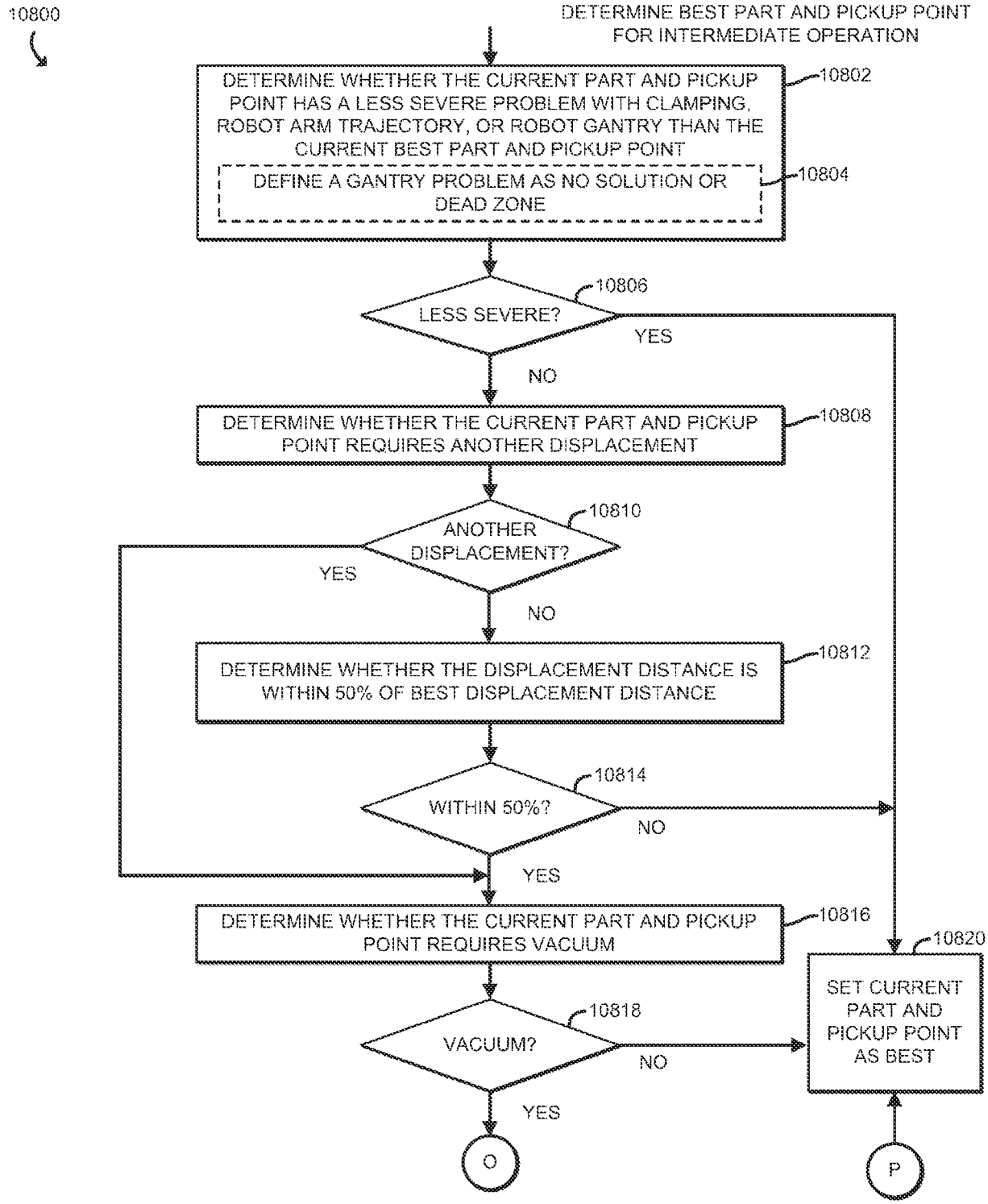

10800
DETERMINE BEST PART AND PICKUP POINT FOR INTERMEDIATE OPERATION
DETERMINE WHETHER THE CURRENT PART AND PICKUP POINT HAS A LESS SEVERE PROBLEM WITH CLAMPING, ROBOT ARM TRAJECTORY, OR ROBOT GANTRY THAN THE CURRENT BEST PART AND PICKUP POINT
10802
DEFINE A GANTRY PROBLEM AS NO SOLUTION OR DEAD ZONE
10804
LESS SEVERE?
10806
YES
NO
DETERMINE WHETHER THE CURRENT PART AND PICKUP POINT REQUIRES ANOTHER DISPLACEMENT
10808
ANOTHER DISPLACEMENT?
10810
YES
NO
DETERMINE WHETHER THE DISPLACEMENT DISTANCE IS WITHIN 50% OF BEST DISPLACEMENT DISTANCE
10812
WITHIN 50%?
10814
NO
YES
DETERMINE WHETHER THE CURRENT PART AND PICKUP POINT REQUIRES VACUUM
10816
VACUUM?
10818
NO
YES
SET CURRENT PART AND PICKUP POINT AS BEST
10820
O
P

FIG. 108

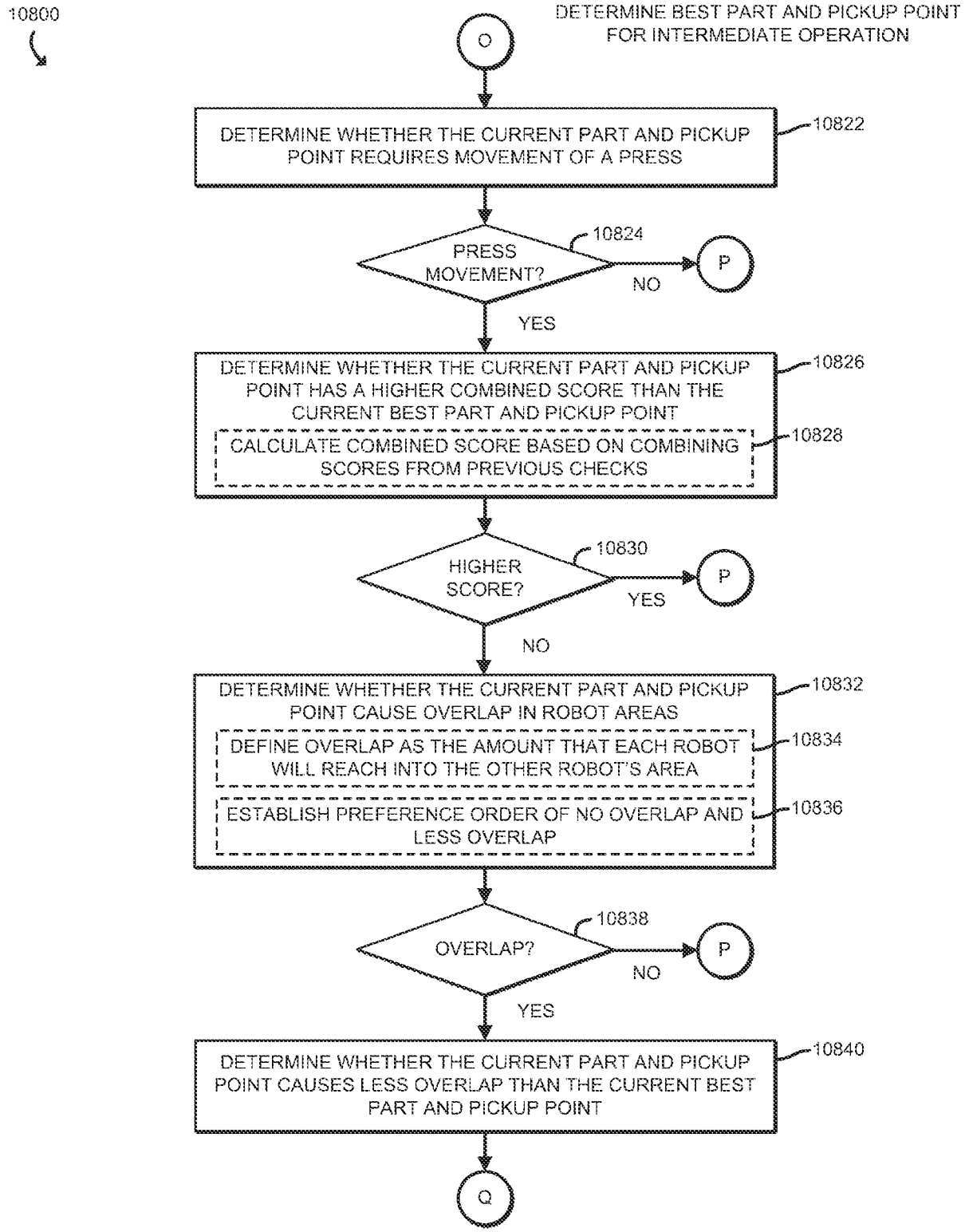

10800
DETERMINE BEST PART AND PICKUP POINT FOR INTERMEDIATE OPERATION
O
DETERMINE WHETHER THE CURRENT PART AND PICKUP POINT REQUIRES MOVEMENT OF A PRESS
10822
PRESS MOVEMENT?
10824
NO
P
YES
DETERMINE WHETHER THE CURRENT PART AND PICKUP POINT HAS A HIGHER COMBINED SCORE THAN THE CURRENT BEST PART AND PICKUP POINT
10826
CALCULATE COMBINED SCORE BASED ON COMBINING SCORES FROM PREVIOUS CHECKS
10828
HIGHER SCORE?
10830
YES
P
NO
DETERMINE WHETHER THE CURRENT PART AND PICKUP POINT CAUSE OVERLAP IN ROBOT AREAS
10832
DEFINE OVERLAP AS THE AMOUNT THAT EACH ROBOT WILL REACH INTO THE OTHER ROBOT'S AREA
10834
ESTABLISH PREFERENCE ORDER OF NO OVERLAP AND LESS OVERLAP
10836
OVERLAP?
10838
NO
P
YES
DETERMINE WHETHER THE CURRENT PART AND PICKUP POINT CAUSES LESS OVERLAP THAN THE CURRENT BEST PART AND PICKUP POINT
10840
Q

FIG. 109

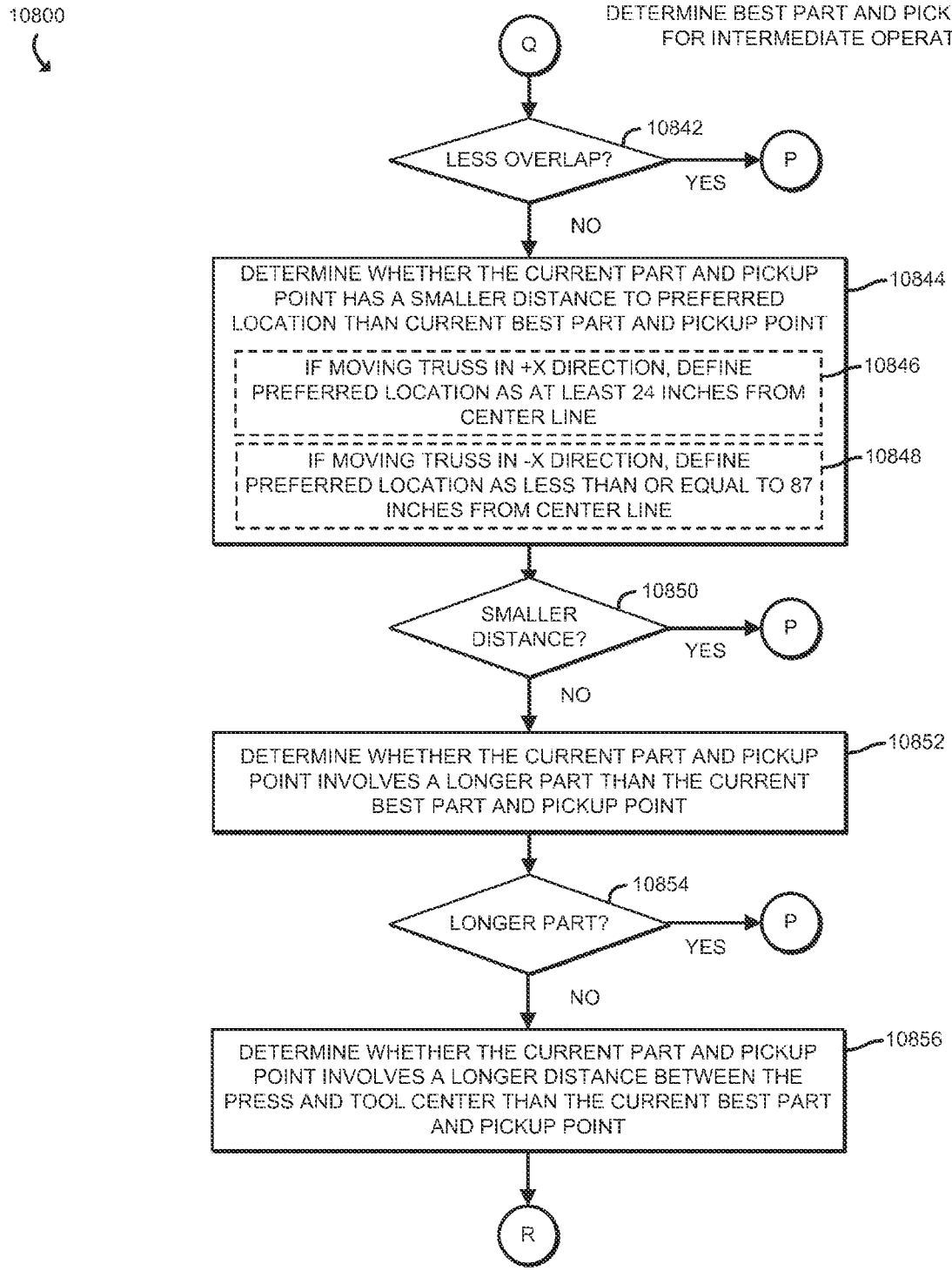

10800
DETERMINE BEST PART AND PICKUP POINT FOR INTERMEDIATE OPERATION
Q
LESS OVERLAP?
10842
YES
P
NO
DETERMINE WHETHER THE CURRENT PART AND PICKUP POINT HAS A SMALLER DISTANCE TO PREFERRED LOCATION THAN CURRENT BEST PART AND PICKUP POINT
10844
IF MOVING TRUSS IN +X DIRECTION, DEFINE PREFERRED LOCATION AS AT LEAST 24 INCHES FROM CENTER LINE
10846
IF MOVING TRUSS IN -X DIRECTION, DEFINE PREFERRED LOCATION AS LESS THAN OR EQUAL TO 87 INCHES FROM CENTER LINE
10848
SMALLER DISTANCE?
10850
YES
P
NO
DETERMINE WHETHER THE CURRENT PART AND PICKUP POINT INVOLVES A LONGER PART THAN THE CURRENT BEST PART AND PICKUP POINT
10852
LONGER PART?
10854
YES
P
NO
DETERMINE WHETHER THE CURRENT PART AND PICKUP POINT INVOLVES A LONGER DISTANCE BETWEEN THE PRESS AND TOOL CENTER THAN THE CURRENT BEST PART AND PICKUP POINT
10856
R

FIG. 110

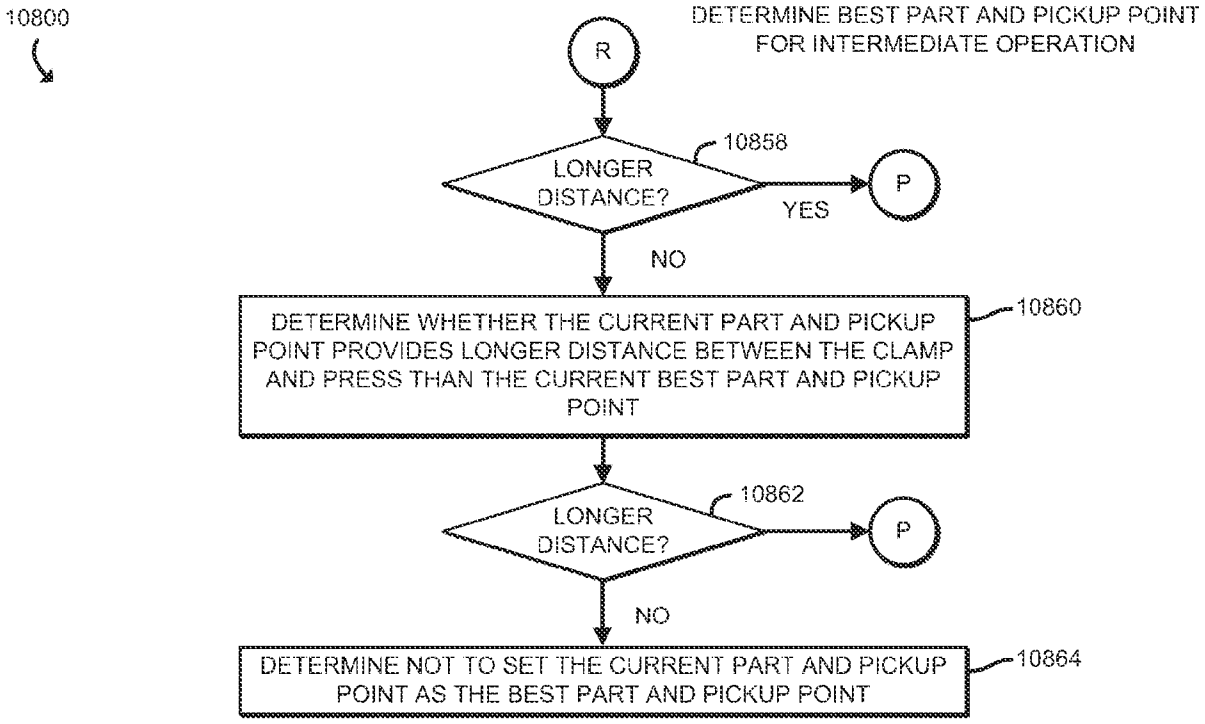

10800
DETERMINE BEST PART AND PICKUP POINT FOR INTERMEDIATE OPERATION
R
LONGER DISTANCE?
10858
YES
P
NO
DETERMINE WHETHER THE CURRENT PART AND PICKUP POINT PROVIDES LONGER DISTANCE BETWEEN THE CLAMP AND PRESS THAN THE CURRENT BEST PART AND PICKUP POINT
10860
LONGER DISTANCE?
10862
P
NO
DETERMINE NOT TO SET THE CURRENT PART AND PICKUP POINT AS THE BEST PART AND PICKUP POINT
10864

FIG. 111

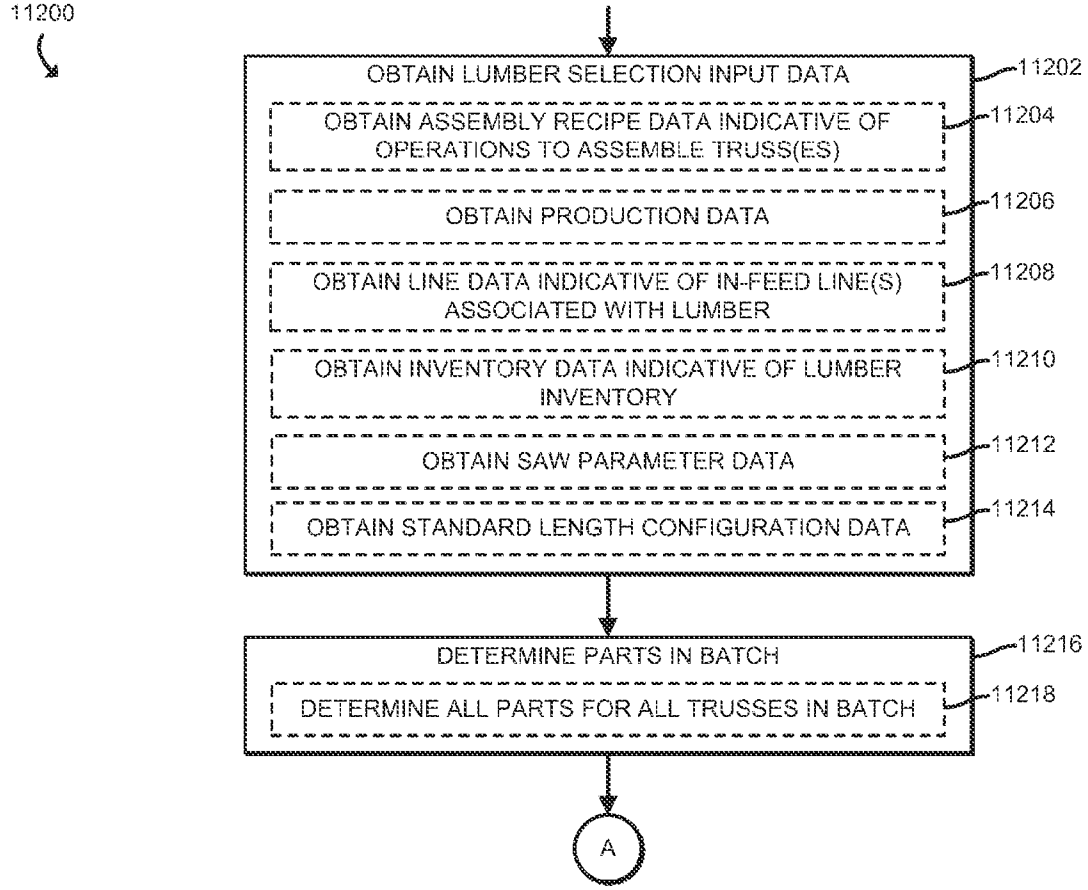

11200
OBTAIN LUMBER SELECTION INPUT DATA
11202
OBTAIN ASSEMBLY RECIPE DATA INDICATIVE OF OPERATIONS TO ASSEMBLE TRUSS(ES)
11204
OBTAIN PRODUCTION DATA
11206
OBTAIN LINE DATA INDICATIVE OF IN-FEED LINE(S) ASSOCIATED WITH LUMBER
11208
OBTAIN INVENTORY DATA INDICATIVE OF LUMBER INVENTORY
11210
OBTAIN SAW PARAMETER DATA
11212
OBTAIN STANDARD LENGTH CONFIGURATION DATA
11214
DETERMINE PARTS IN BATCH
11216
DETERMINE ALL PARTS FOR ALL TRUSSES IN BATCH
11218
A

FIG. 112

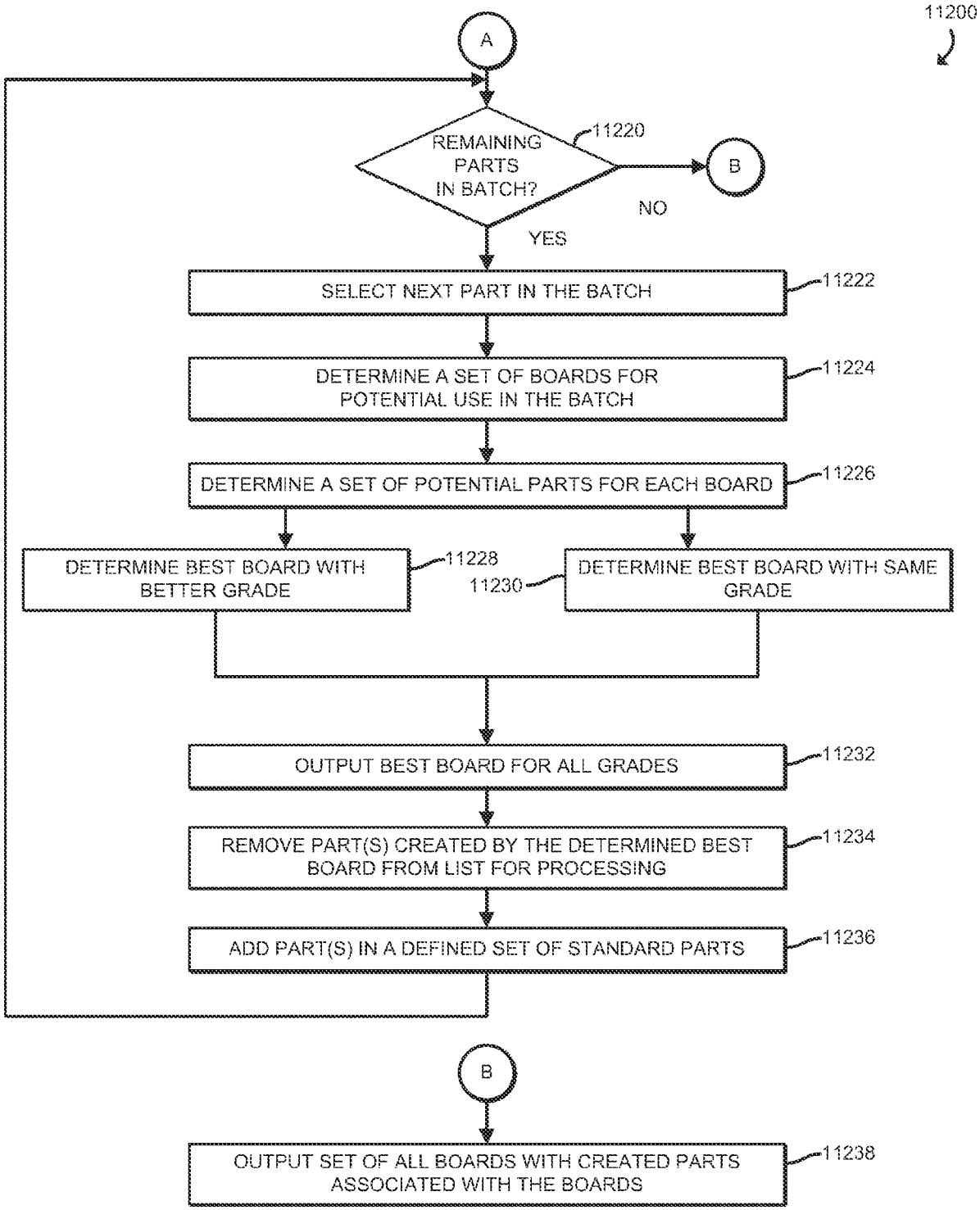

11200
A
REMAINING PARTS IN BATCH?
11220
B
NO
YES
SELECT NEXT PART IN THE BATCH
11222
DETERMINE A SET OF BOARDS FOR POTENTIAL USE IN THE BATCH
11224
DETERMINE A SET OF POTENTIAL PARTS FOR EACH BOARD
11226
DETERMINE BEST BOARD WITH BETTER GRADE
11228
11230
DETERMINE BEST BOARD WITH SAME GRADE
OUTPUT BEST BOARD FOR ALL GRADES
11232
REMOVE PART(S) CREATED BY THE DETERMINED BEST BOARD FROM LIST FOR PROCESSING
11234
ADD PART(S) IN A DEFINED SET OF STANDARD PARTS
11236
B
OUTPUT SET OF ALL BOARDS WITH CREATED PARTS ASSOCIATED WITH THE BOARDS
11238

FIG. 113

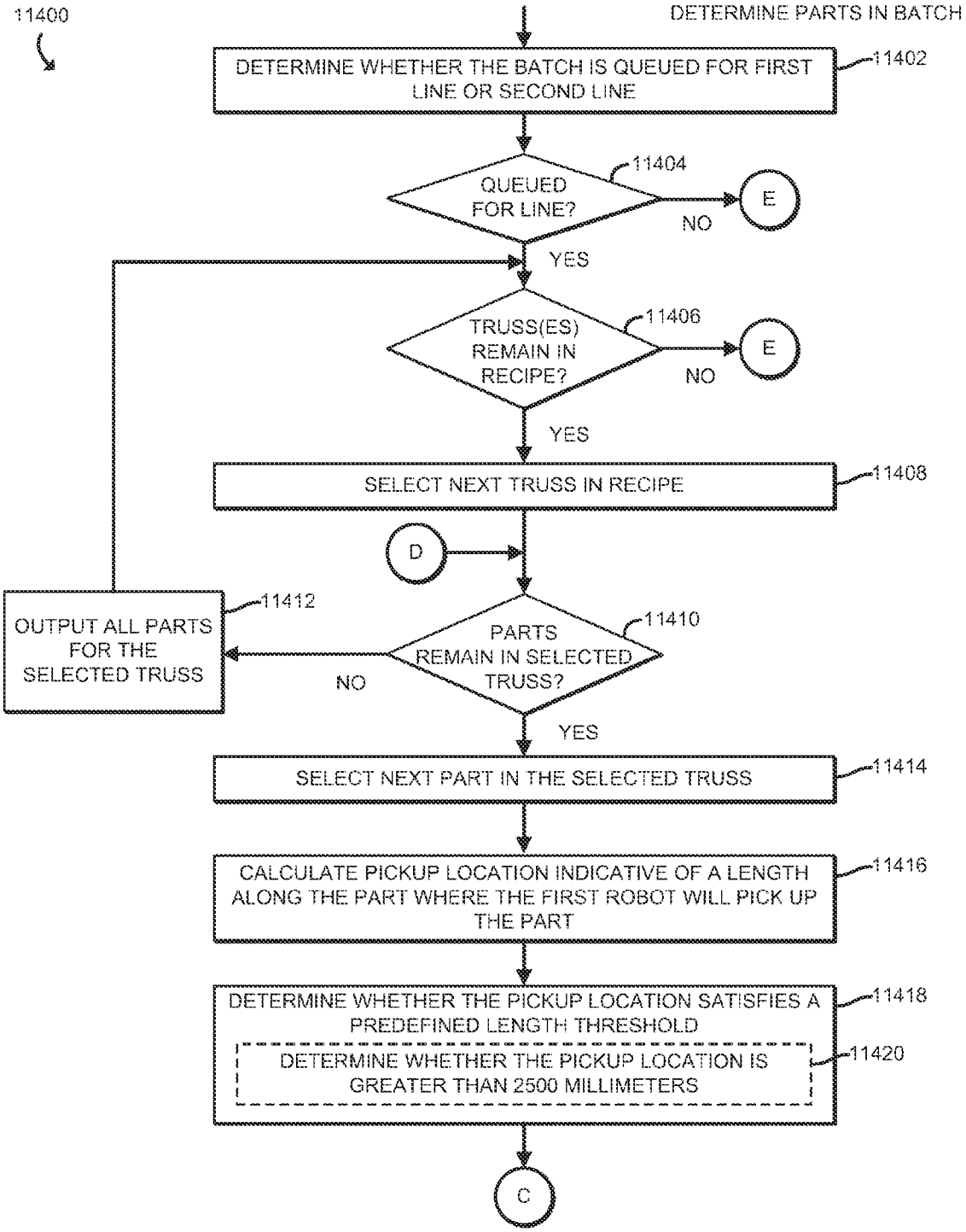

11400
DETERMINE PARTS IN BATCH
DETERMINE WHETHER THE BATCH IS QUEUED FOR FIRST LINE OR SECOND LINE
11402
QUEUED FOR LINE?
11404
NO
E
YES
TRUSS(ES) REMAIN IN RECIPE?
11406
NO
E
YES
SELECT NEXT TRUSS IN RECIPE
11408
D
OUTPUT ALL PARTS FOR THE SELECTED TRUSS
11412
PARTS REMAIN IN SELECTED TRUSS?
11410
NO
YES
SELECT NEXT PART IN THE SELECTED TRUSS
11414
CALCULATE PICKUP LOCATION INDICATIVE OF A LENGTH ALONG THE PART WHERE THE FIRST ROBOT WILL PICK UP THE PART
11416
DETERMINE WHETHER THE PICKUP LOCATION SATISFIES A PREDEFINED LENGTH THRESHOLD
11418
DETERMINE WHETHER THE PICKUP LOCATION IS GREATER THAN 2500 MILLIMETERS
11420
C

FIG. 114

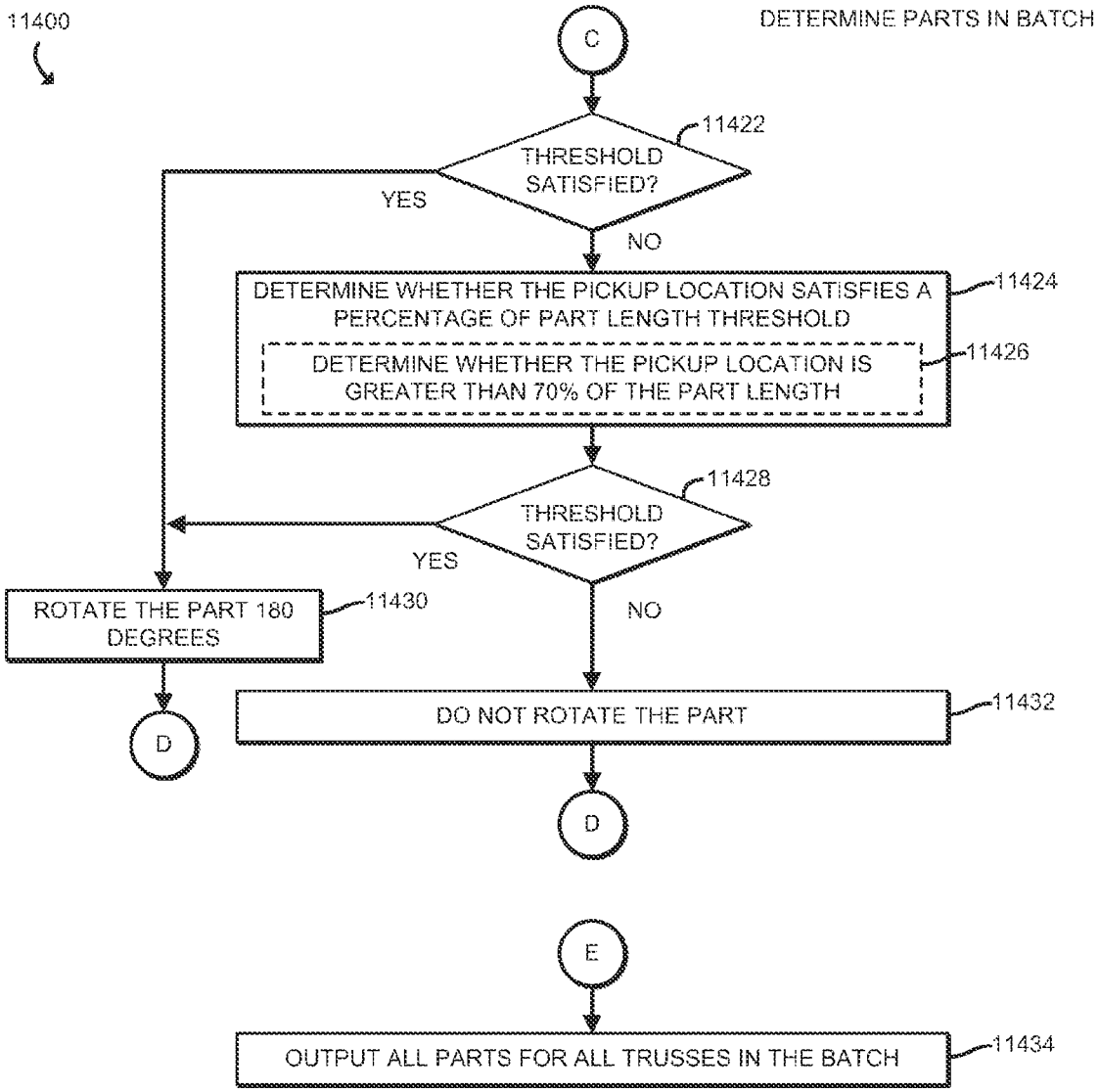

11400
DETERMINE PARTS IN BATCH
C
11422
THRESHOLD SATISFIED?
YES
NO
DETERMINE WHETHER THE PICKUP LOCATION SATISFIES A PERCENTAGE OF PART LENGTH THRESHOLD
11424
DETERMINE WHETHER THE PICKUP LOCATION IS GREATER THAN 70% OF THE PART LENGTH
11426
11428
THRESHOLD SATISFIED?
YES
NO
ROTATE THE PART 180 DEGREES
11430
D
DO NOT ROTATE THE PART
11432
D
E
OUTPUT ALL PARTS FOR ALL TRUSSES IN THE BATCH
11434

FIG. 115

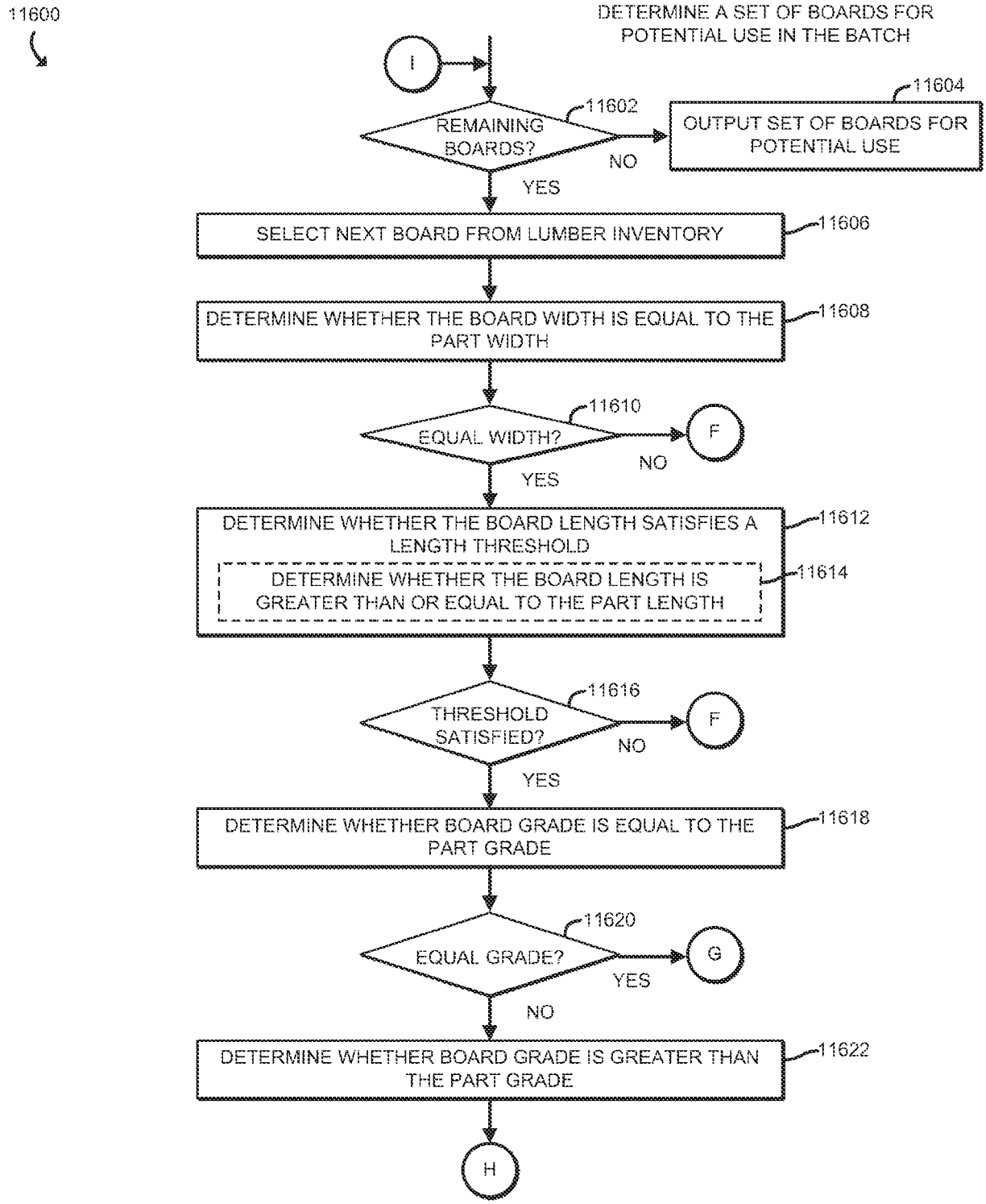

11600
DETERMINE A SET OF BOARDS FOR POTENTIAL USE IN THE BATCH
I
11602
REMAINING BOARDS?
NO
11604
OUTPUT SET OF BOARDS FOR POTENTIAL USE
YES
SELECT NEXT BOARD FROM LUMBER INVENTORY
11606
DETERMINE WHETHER THE BOARD WIDTH IS EQUAL TO THE PART WIDTH
11608
11610
EQUAL WIDTH?
F
NO
YES
DETERMINE WHETHER THE BOARD LENGTH SATISFIES A LENGTH THRESHOLD
11612
DETERMINE WHETHER THE BOARD LENGTH IS GREATER THAN OR EQUAL TO THE PART LENGTH
11614
11616
THRESHOLD SATISFIED?
F
NO
YES
DETERMINE WHETHER BOARD GRADE IS EQUAL TO THE PART GRADE
11618
11620
EQUAL GRADE?
G
YES
NO
DETERMINE WHETHER BOARD GRADE IS GREATER THAN THE PART GRADE
11622
H

FIG. 116

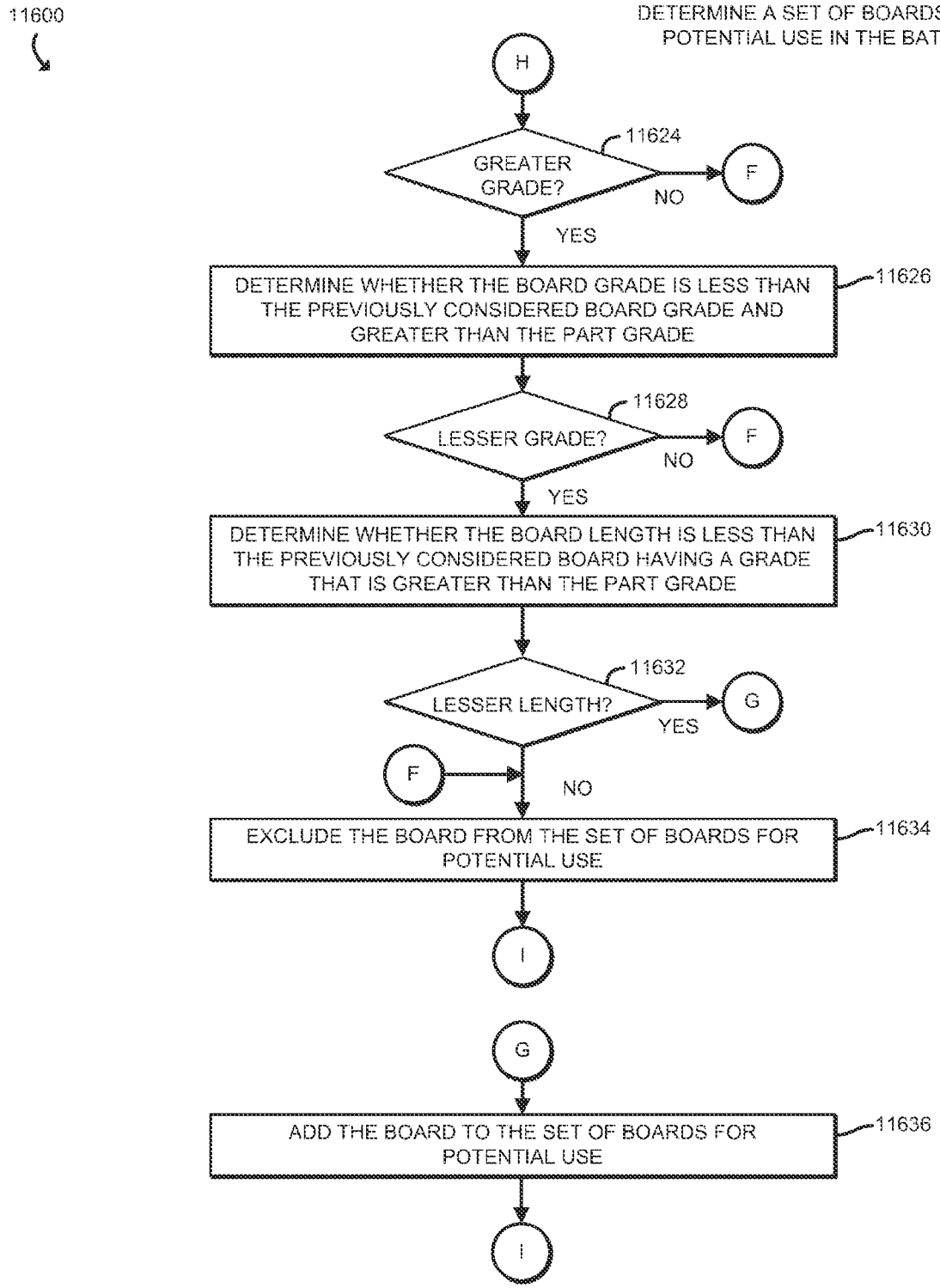

11600
DETERMINE A SET OF BOARDS FOR POTENTIAL USE IN THE BATCH
H
11624
GREATER GRADE?
NO
F
YES
DETERMINE WHETHER THE BOARD GRADE IS LESS THAN THE PREVIOUSLY CONSIDERED BOARD GRADE AND GREATER THAN THE PART GRADE
11626
11628
LESSER GRADE?
NO
F
YES
DETERMINE WHETHER THE BOARD LENGTH IS LESS THAN THE PREVIOUSLY CONSIDERED BOARD HAVING A GRADE THAT IS GREATER THAN THE PART GRADE
11630
11632
LESSER LENGTH?
YES
G
F
NO
EXCLUDE THE BOARD FROM THE SET OF BOARDS FOR POTENTIAL USE
11634
I
G
ADD THE BOARD TO THE SET OF BOARDS FOR POTENTIAL USE
11636
I

FIG. 117

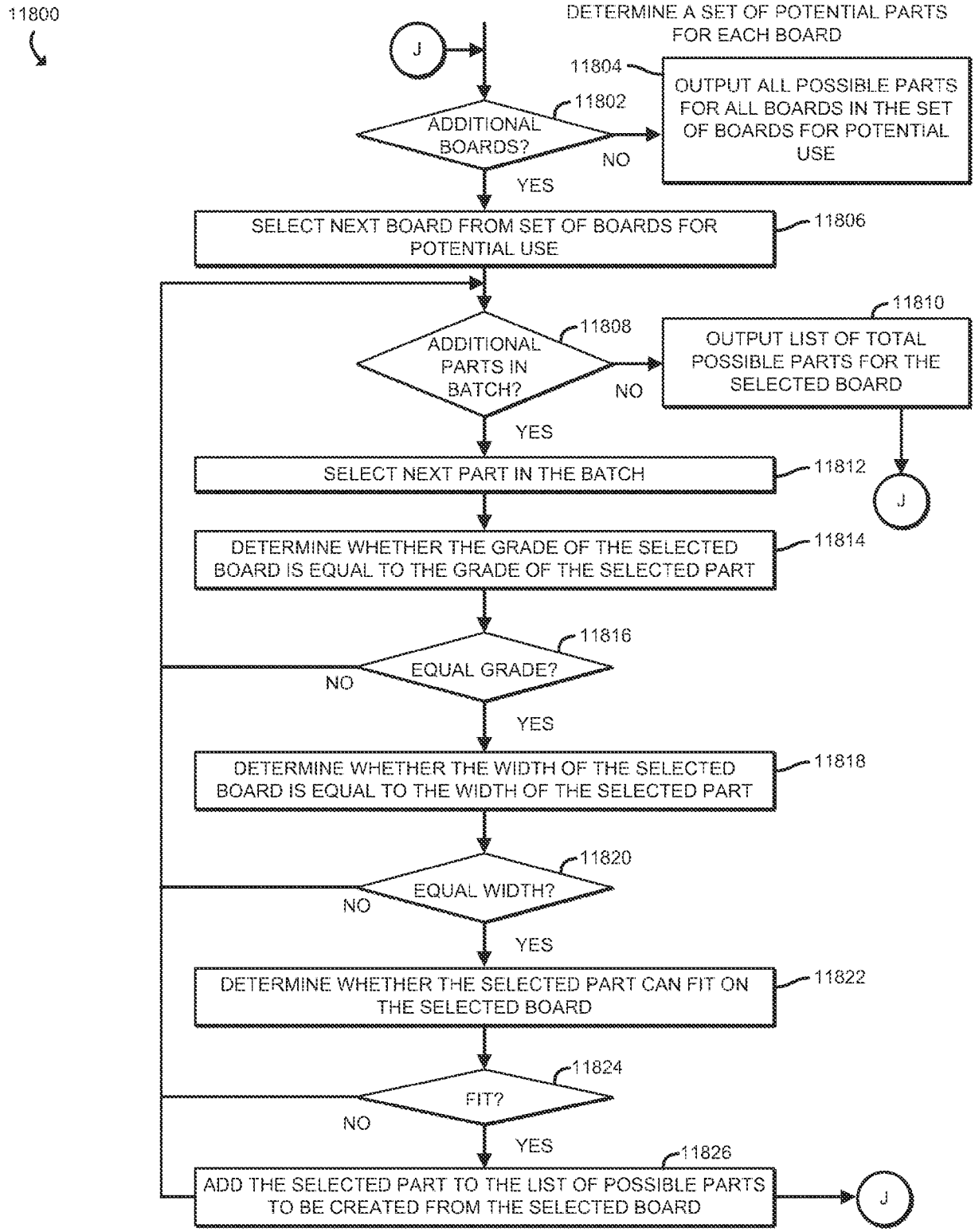

11800
DETERMINE A SET OF POTENTIAL PARTS FOR EACH BOARD
J
11804
OUTPUT ALL POSSIBLE PARTS FOR ALL BOARDS IN THE SET OF BOARDS FOR POTENTIAL USE
11802
ADDITIONAL BOARDS?
NO
YES
SELECT NEXT BOARD FROM SET OF BOARDS FOR POTENTIAL USE
11806
11810
11808
ADDITIONAL PARTS IN BATCH?
NO
OUTPUT LIST OF TOTAL POSSIBLE PARTS FOR THE SELECTED BOARD
YES
SELECT NEXT PART IN THE BATCH
11812
J
DETERMINE WHETHER THE GRADE OF THE SELECTED BOARD IS EQUAL TO THE GRADE OF THE SELECTED PART
11814
11816
EQUAL GRADE?
NO
YES
DETERMINE WHETHER THE WIDTH OF THE SELECTED BOARD IS EQUAL TO THE WIDTH OF THE SELECTED PART
11818
11820
EQUAL WIDTH?
NO
YES
DETERMINE WHETHER THE SELECTED PART CAN FIT ON THE SELECTED BOARD
11822
11824
FIT?
NO
YES
11826
ADD THE SELECTED PART TO THE LIST OF POSSIBLE PARTS TO BE CREATED FROM THE SELECTED BOARD
J

FIG. 118

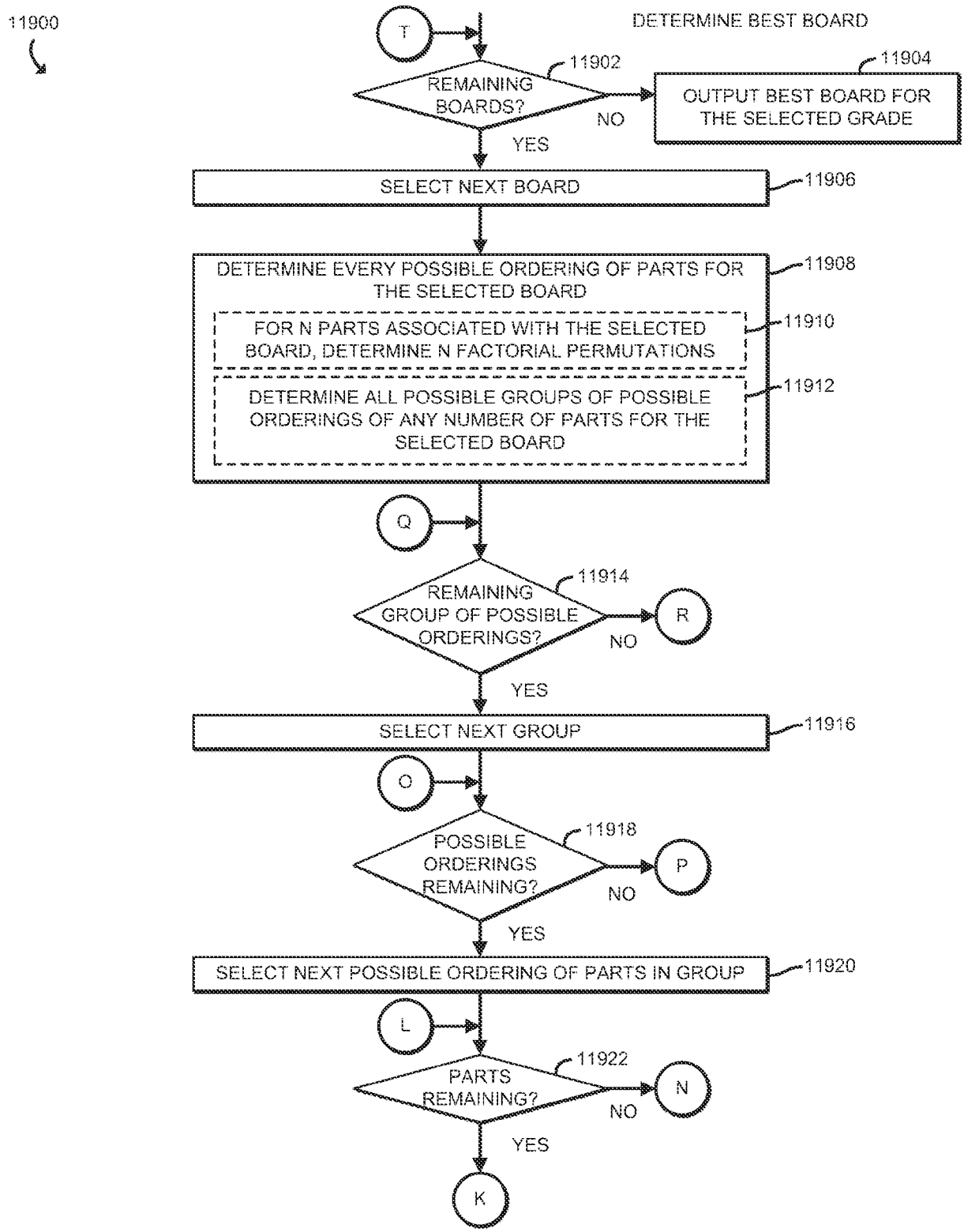

11900
T
DETERMINE BEST BOARD
11902
REMAINING BOARDS?
NO
11904
OUTPUT BEST BOARD FOR THE SELECTED GRADE
YES
SELECT NEXT BOARD
11906
DETERMINE EVERY POSSIBLE ORDERING OF PARTS FOR THE SELECTED BOARD
11908
FOR N PARTS ASSOCIATED WITH THE SELECTED BOARD, DETERMINE N FACTORIAL PERMUTATIONS
11910
DETERMINE ALL POSSIBLE GROUPS OF POSSIBLE ORDERINGS OF ANY NUMBER OF PARTS FOR THE SELECTED BOARD
11912
Q
11914
REMAINING GROUP OF POSSIBLE ORDERINGS?
NO
R
YES
SELECT NEXT GROUP
11916
O
11918
POSSIBLE ORDERINGS REMAINING?
NO
P
YES
SELECT NEXT POSSIBLE ORDERING OF PARTS IN GROUP
11920
L
11922
PARTS REMAINING?
NO
N
YES
K

FIG. 119

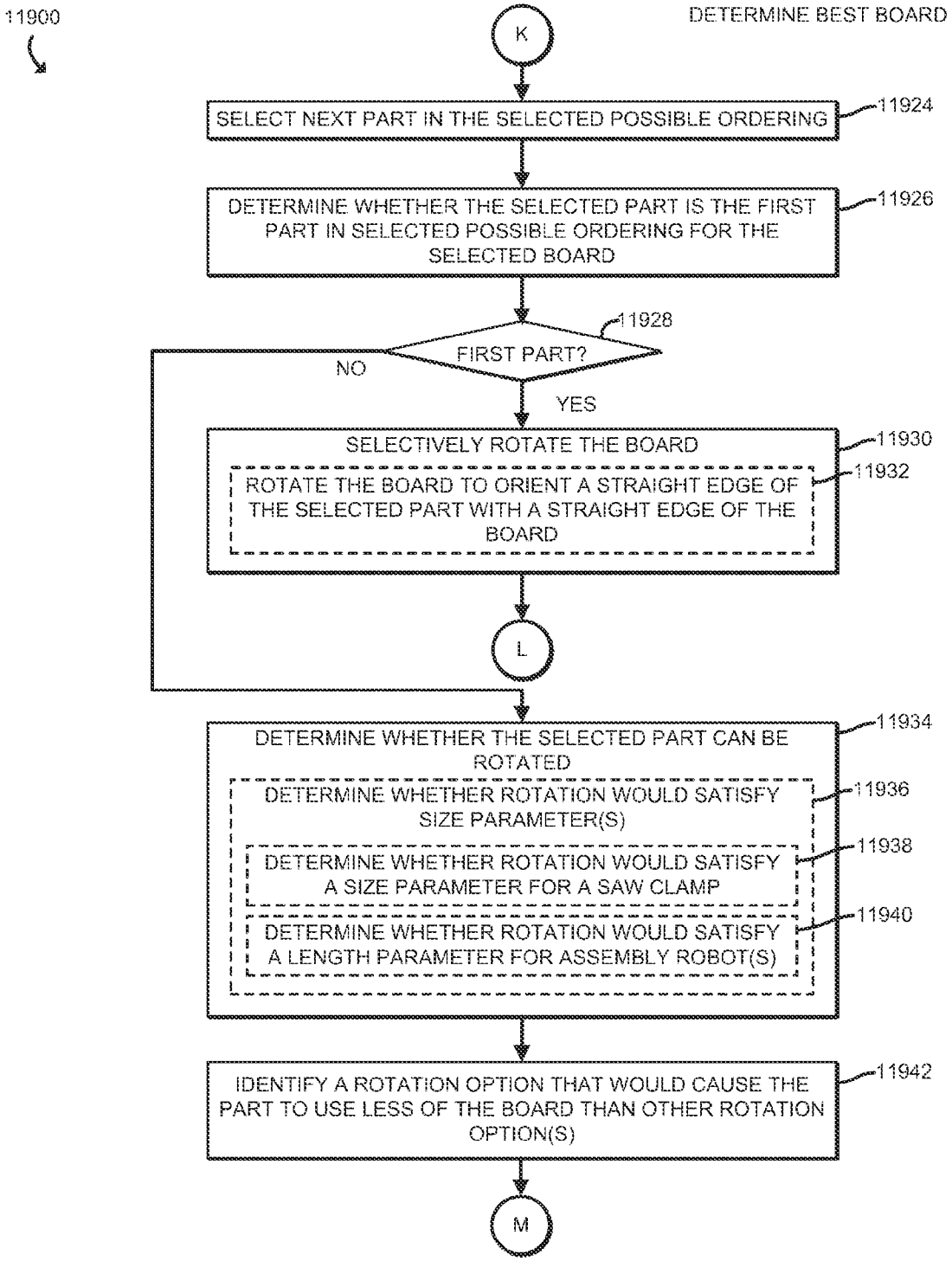

11900
DETERMINE BEST BOARD
K
SELECT NEXT PART IN THE SELECTED POSSIBLE ORDERING
11924
DETERMINE WHETHER THE SELECTED PART IS THE FIRST PART IN SELECTED POSSIBLE ORDERING FOR THE SELECTED BOARD
11926
11928
FIRST PART?
NO
YES
SELECTIVELY ROTATE THE BOARD
11930
ROTATE THE BOARD TO ORIENT A STRAIGHT EDGE OF THE SELECTED PART WITH A STRAIGHT EDGE OF THE BOARD
11932
L
DETERMINE WHETHER THE SELECTED PART CAN BE ROTATED
11934
DETERMINE WHETHER ROTATION WOULD SATISFY SIZE PARAMETER(S)
11936
DETERMINE WHETHER ROTATION WOULD SATISFY A SIZE PARAMETER FOR A SAW CLAMP
11938
DETERMINE WHETHER ROTATION WOULD SATISFY A LENGTH PARAMETER FOR ASSEMBLY ROBOT(S)
11940
IDENTIFY A ROTATION OPTION THAT WOULD CAUSE THE PART TO USE LESS OF THE BOARD THAN OTHER ROTATION OPTION(S)
11942
M

FIG. 120

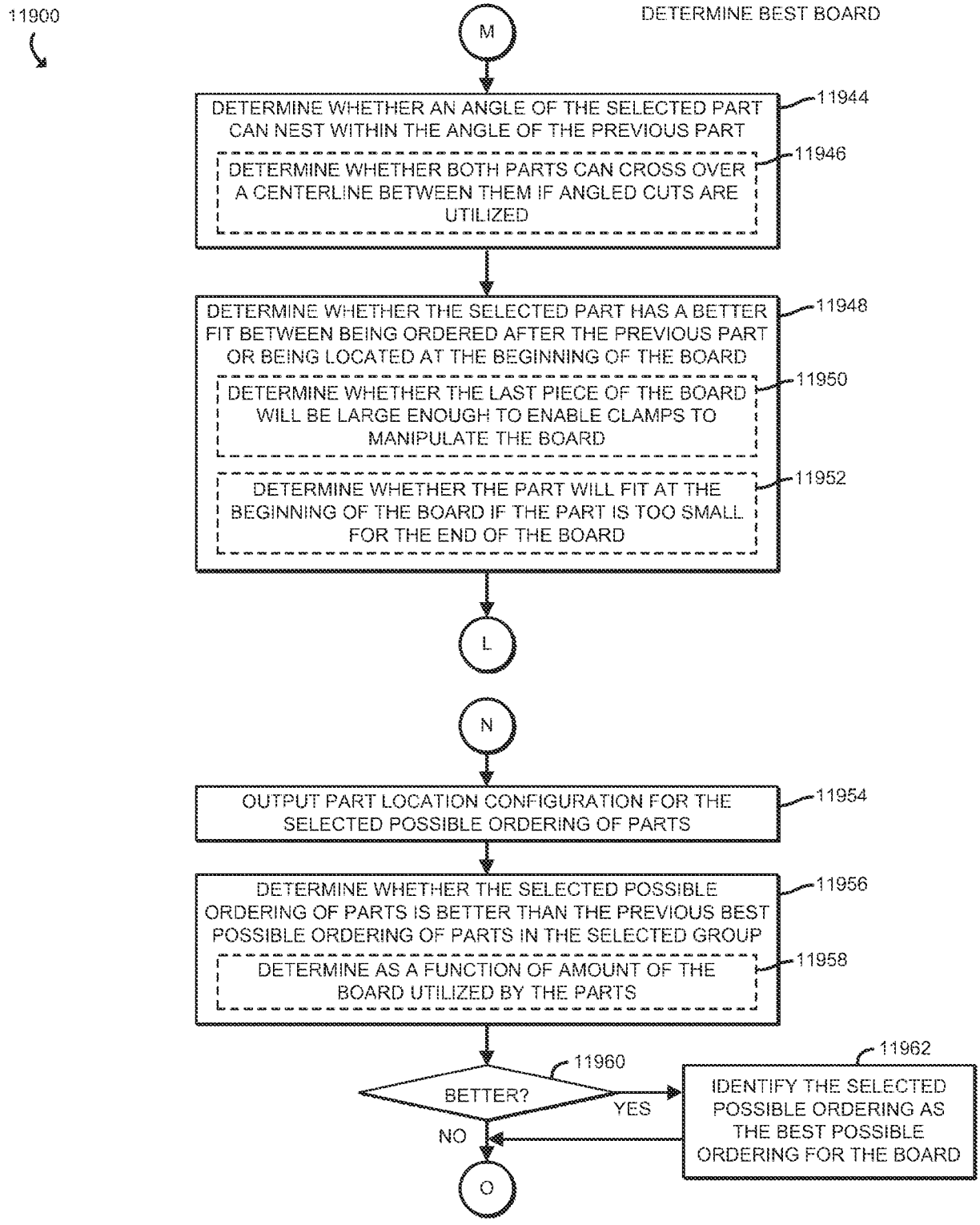

11900
DETERMINE BEST BOARD
M
DETERMINE WHETHER AN ANGLE OF THE SELECTED PART CAN NEST WITHIN THE ANGLE OF THE PREVIOUS PART
11944
DETERMINE WHETHER BOTH PARTS CAN CROSS OVER A CENTERLINE BETWEEN THEM IF ANGLED CUTS ARE UTILIZED
11946
DETERMINE WHETHER THE SELECTED PART HAS A BETTER FIT BETWEEN BEING ORDERED AFTER THE PREVIOUS PART OR BEING LOCATED AT THE BEGINNING OF THE BOARD
11948
DETERMINE WHETHER THE LAST PIECE OF THE BOARD WILL BE LARGE ENOUGH TO ENABLE CLAMPS TO MANIPULATE THE BOARD
11950
DETERMINE WHETHER THE PART WILL FIT AT THE BEGINNING OF THE BOARD IF THE PART IS TOO SMALL FOR THE END OF THE BOARD
11952
L
N
OUTPUT PART LOCATION CONFIGURATION FOR THE SELECTED POSSIBLE ORDERING OF PARTS
11954
DETERMINE WHETHER THE SELECTED POSSIBLE ORDERING OF PARTS IS BETTER THAN THE PREVIOUS BEST POSSIBLE ORDERING OF PARTS IN THE SELECTED GROUP
11956
DETERMINE AS A FUNCTION OF AMOUNT OF THE BOARD UTILIZED BY THE PARTS
11958
11960
BETTER?
YES
NO
11962
IDENTIFY THE SELECTED POSSIBLE ORDERING AS THE BEST POSSIBLE ORDERING FOR THE BOARD
O

FIG. 121

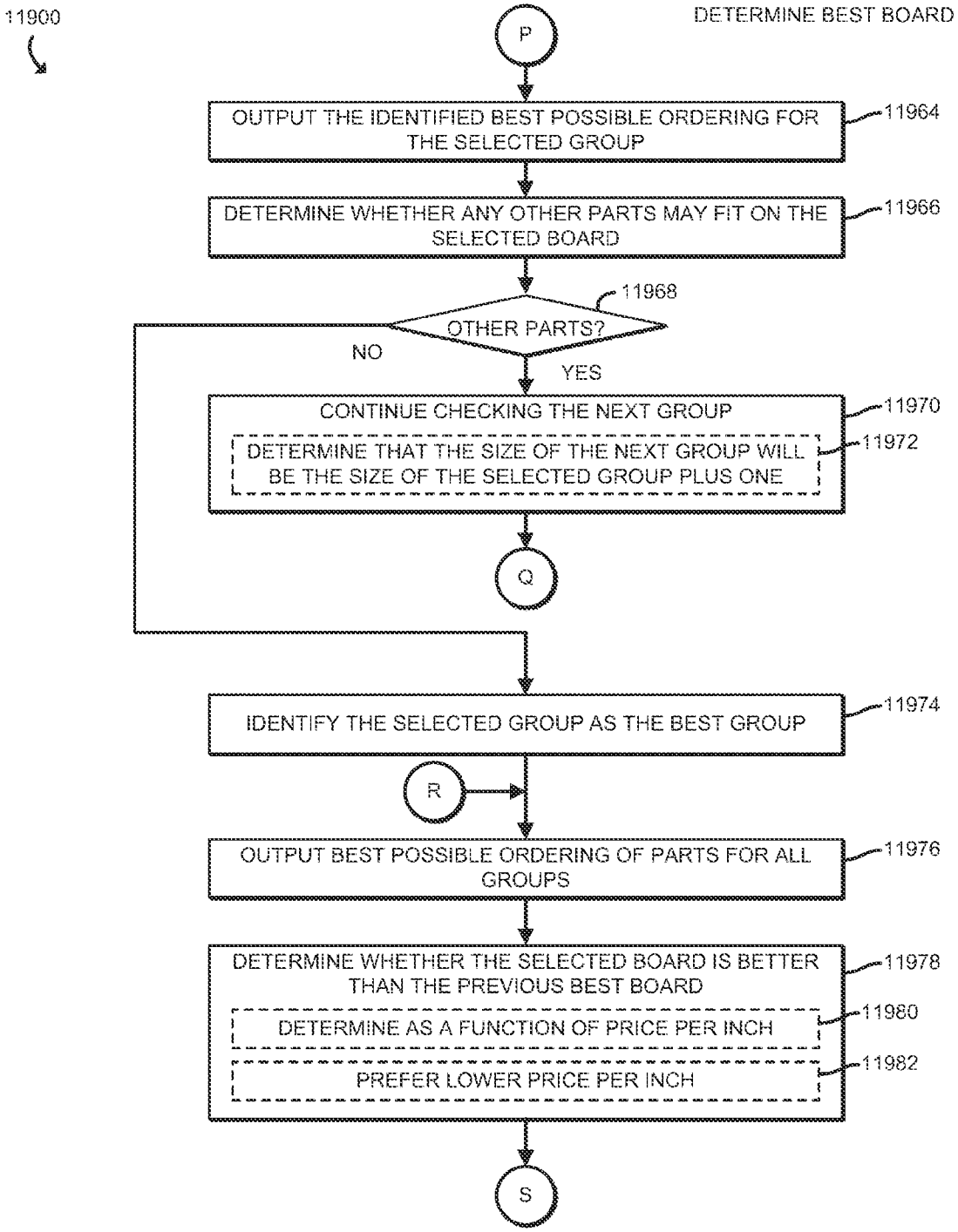

11900
DETERMINE BEST BOARD
P
OUTPUT THE IDENTIFIED BEST POSSIBLE ORDERING FOR THE SELECTED GROUP
11964
DETERMINE WHETHER ANY OTHER PARTS MAY FIT ON THE SELECTED BOARD
11966
11968
OTHER PARTS?
NO
YES
CONTINUE CHECKING THE NEXT GROUP
11970
DETERMINE THAT THE SIZE OF THE NEXT GROUP WILL BE THE SIZE OF THE SELECTED GROUP PLUS ONE
11972
Q
IDENTIFY THE SELECTED GROUP AS THE BEST GROUP
11974
R
OUTPUT BEST POSSIBLE ORDERING OF PARTS FOR ALL GROUPS
11976
DETERMINE WHETHER THE SELECTED BOARD IS BETTER THAN THE PREVIOUS BEST BOARD
11978
DETERMINE AS A FUNCTION OF PRICE PER INCH
11980
PREFER LOWER PRICE PER INCH
11982
S

FIG. 122

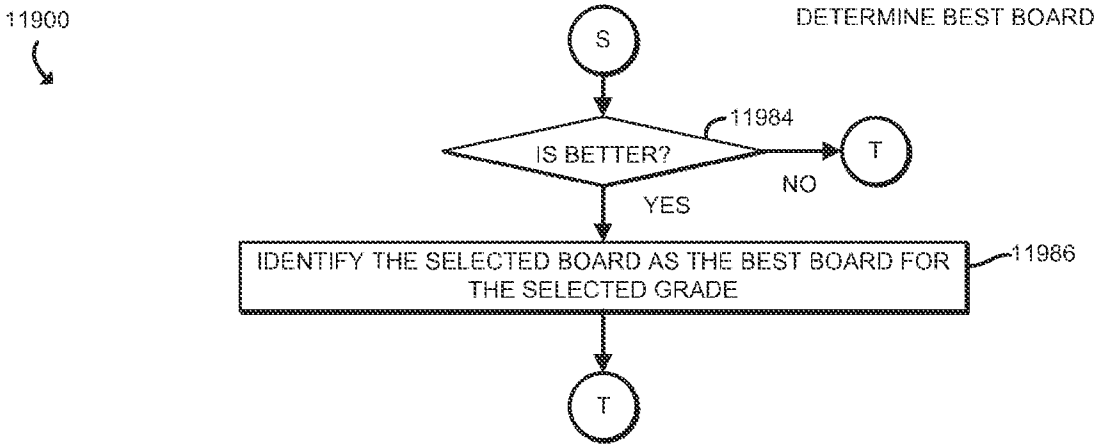

11900
DETERMINE BEST BOARD
S
11984
IS BETTER?
T
NO
YES
IDENTIFY THE SELECTED BOARD AS THE BEST BOARD FOR THE SELECTED GRADE
11986
T

FIG. 123

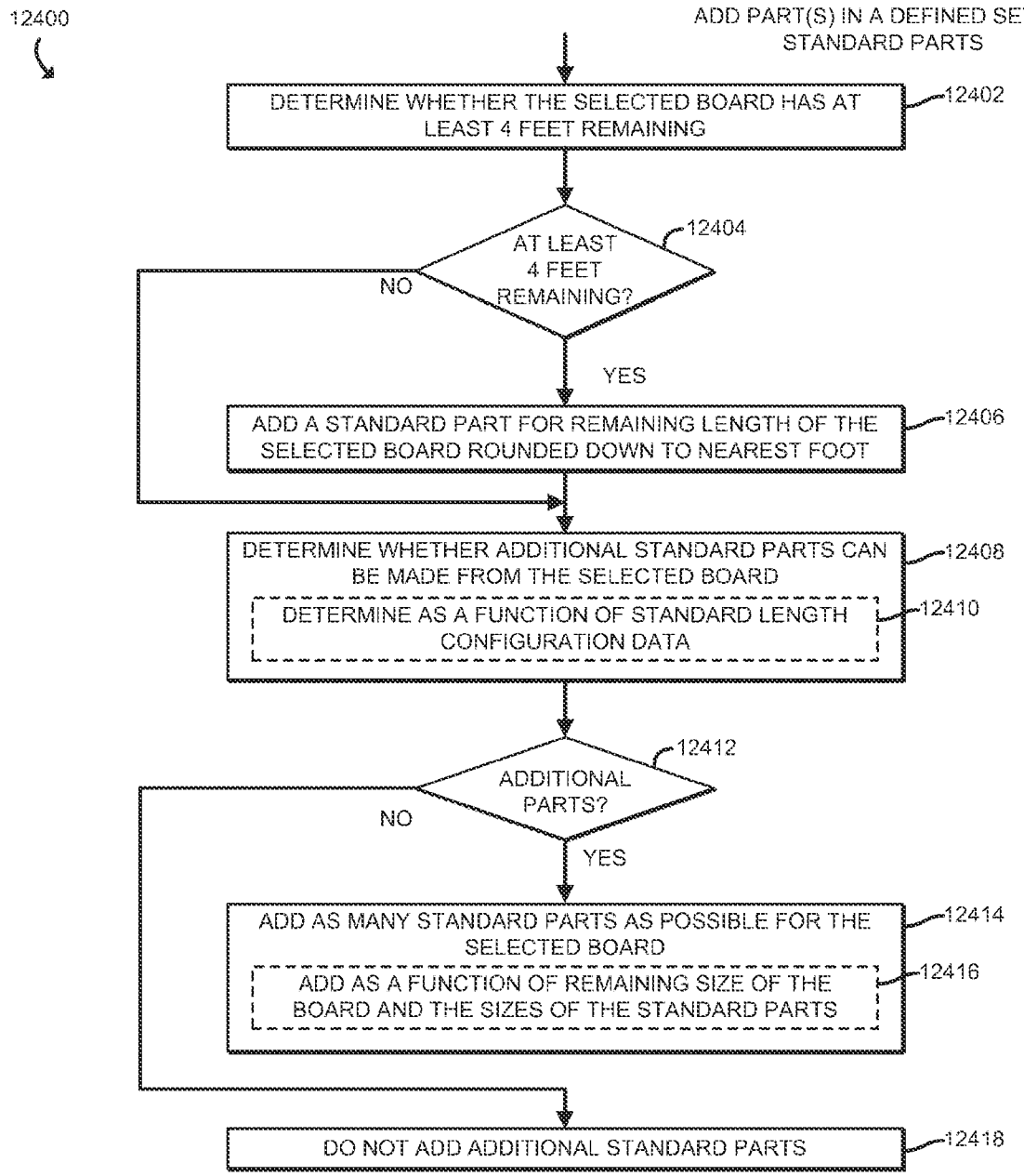

12400
ADD PART(S) IN A DEFINED SET OF STANDARD PARTS
DETERMINE WHETHER THE SELECTED BOARD HAS AT LEAST 4 FEET REMAINING
12402
AT LEAST 4 FEET REMAINING?
12404
NO
YES
ADD A STANDARD PART FOR REMAINING LENGTH OF THE SELECTED BOARD ROUNDED DOWN TO NEAREST FOOT
12406
DETERMINE WHETHER ADDITIONAL STANDARD PARTS CAN BE MADE FROM THE SELECTED BOARD
12408
DETERMINE AS A FUNCTION OF STANDARD LENGTH CONFIGURATION DATA
12410
ADDITIONAL PARTS?
12412
NO
YES
ADD AS MANY STANDARD PARTS AS POSSIBLE FOR THE SELECTED BOARD
12414
ADD AS A FUNCTION OF REMAINING SIZE OF THE BOARD AND THE SIZES OF THE STANDARD PARTS
12416
DO NOT ADD ADDITIONAL STANDARD PARTS
12418

FIG. 124

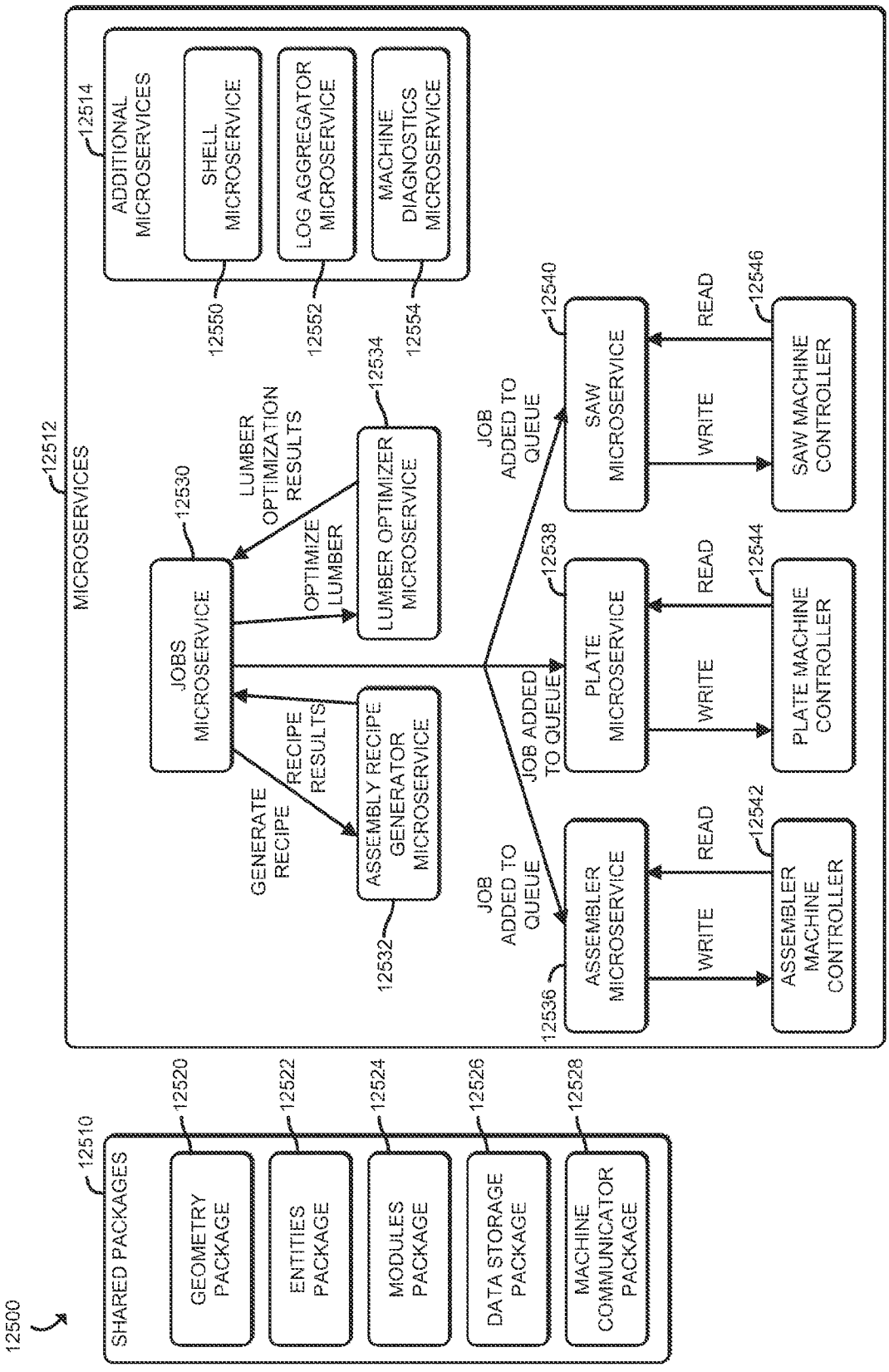

12500
SHARED PACKAGES
12510
GEOMETRY PACKAGE
12520
ENTITIES PACKAGE
12522
MODULES PACKAGE
12524
DATA STORAGE PACKAGE
12526
MACHINE COMMUNICATOR PACKAGE
12528
MICROSERVICES
12512
ADDITIONAL MICROSERVICES
12514
SHELL MICROSERVICE
12550
LOG AGGREGATOR MICROSERVICE
12552
MACHINE DIAGNOSTICS MICROSERVICE
12554
JOBS MICROSERVICE
12530
LUMBER OPTIMIZATION RESULTS
OPTIMIZE LUMBER
LUMBER OPTIMIZER MICROSERVICE
12534
GENERATE RECIPE
RECIPE RESULTS
ASSEMBLY RECIPE GENERATOR MICROSERVICE
12532
JOB ADDED TO QUEUE
ASSEMBLER MICROSERVICE
12536
WRITE
READ
ASSEMBLER MACHINE CONTROLLER
12542
JOB ADDED TO QUEUE
PLATE MICROSERVICE
12538
WRITE
READ
PLATE MACHINE CONTROLLER
12544
JOB ADDED TO QUEUE
SAW MICROSERVICE
12540
WRITE
READ
SAW MACHINE CONTROLLER
12546

FIG. 125

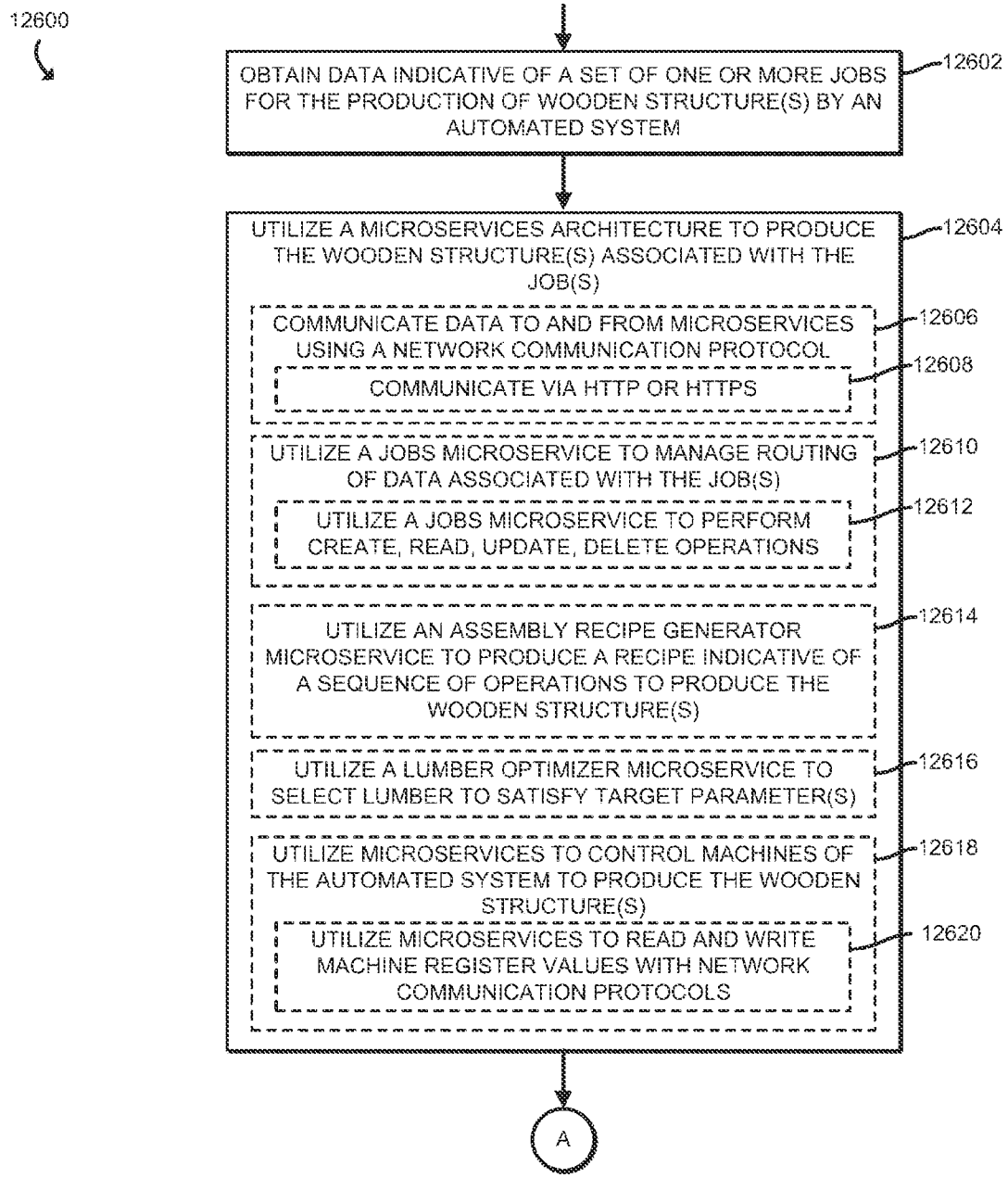

12600
OBTAIN DATA INDICATIVE OF A SET OF ONE OR MORE JOBS FOR THE PRODUCTION OF WOODEN STRUCTURE(S) BY AN AUTOMATED SYSTEM
12602
UTILIZE A MICROSERVICES ARCHITECTURE TO PRODUCE THE WOODEN STRUCTURE(S) ASSOCIATED WITH THE JOB(S)
12604
COMMUNICATE DATA TO AND FROM MICROSERVICES USING A NETWORK COMMUNICATION PROTOCOL
12606
COMMUNICATE VIA HTTP OR HTTPS
12608
UTILIZE A JOBS MICROSERVICE TO MANAGE ROUTING OF DATA ASSOCIATED WITH THE JOB(S)
12610
UTILIZE A JOBS MICROSERVICE TO PERFORM CREATE, READ, UPDATE, DELETE OPERATIONS
12612
UTILIZE AN ASSEMBLY RECIPE GENERATOR MICROSERVICE TO PRODUCE A RECIPE INDICATIVE OF A SEQUENCE OF OPERATIONS TO PRODUCE THE WOODEN STRUCTURE(S)
12614
UTILIZE A LUMBER OPTIMIZER MICROSERVICE TO SELECT LUMBER TO SATISFY TARGET PARAMETER(S)
12616
UTILIZE MICROSERVICES TO CONTROL MACHINES OF THE AUTOMATED SYSTEM TO PRODUCE THE WOODEN STRUCTURE(S)
12618
UTILIZE MICROSERVICES TO READ AND WRITE MACHINE REGISTER VALUES WITH NETWORK COMMUNICATION PROTOCOLS
12620
A

FIG. 126

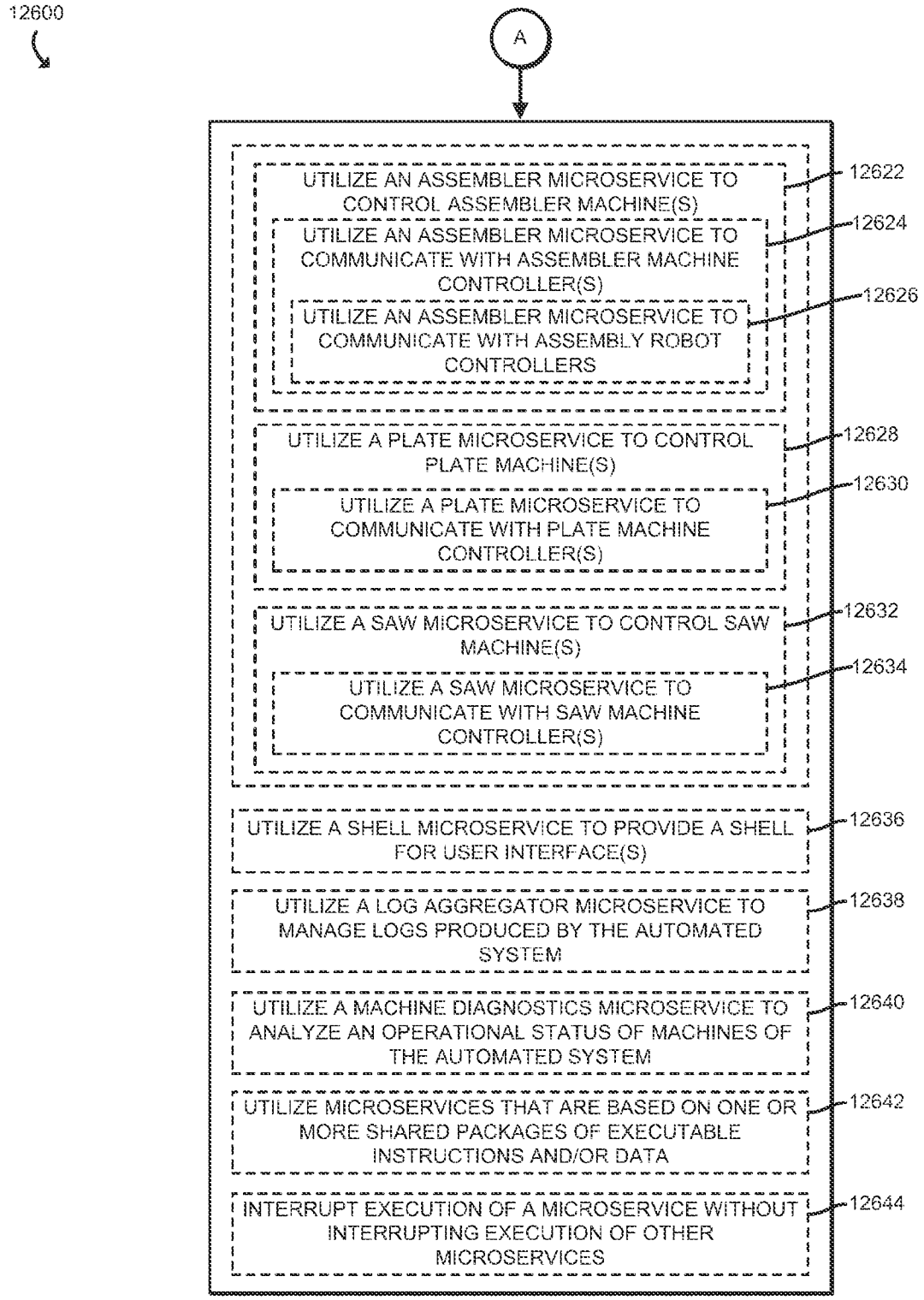

12600
A
UTILIZE AN ASSEMBLER MICROSERVICE TO CONTROL ASSEMBLER MACHINE(S)
12622
UTILIZE AN ASSEMBLER MICROSERVICE TO COMMUNICATE WITH ASSEMBLER MACHINE CONTROLLER(S)
12624
UTILIZE AN ASSEMBLER MICROSERVICE TO COMMUNICATE WITH ASSEMBLY ROBOT CONTROLLERS
12626
UTILIZE A PLATE MICROSERVICE TO CONTROL PLATE MACHINE(S)
12628
UTILIZE A PLATE MICROSERVICE TO COMMUNICATE WITH PLATE MACHINE CONTROLLER(S)
12630
UTILIZE A SAW MICROSERVICE TO CONTROL SAW MACHINE(S)
12632
UTILIZE A SAW MICROSERVICE TO COMMUNICATE WITH SAW MACHINE CONTROLLER(S)
12634
UTILIZE A SHELL MICROSERVICE TO PROVIDE A SHELL FOR USER INTERFACE(S)
12636
UTILIZE A LOG AGGREGATOR MICROSERVICE TO MANAGE LOGS PRODUCED BY THE AUTOMATED SYSTEM
12638
UTILIZE A MACHINE DIAGNOSTICS MICROSERVICE TO ANALYZE AN OPERATIONAL STATUS OF MACHINES OF THE AUTOMATED SYSTEM
12640
UTILIZE MICROSERVICES THAT ARE BASED ON ONE OR MORE SHARED PACKAGES OF EXECUTABLE INSTRUCTIONS AND/OR DATA
12642
INTERRUPT EXECUTION OF A MICROSERVICE WITHOUT INTERRUPTING EXECUTION OF OTHER MICROSERVICES
12644

FIG. 127

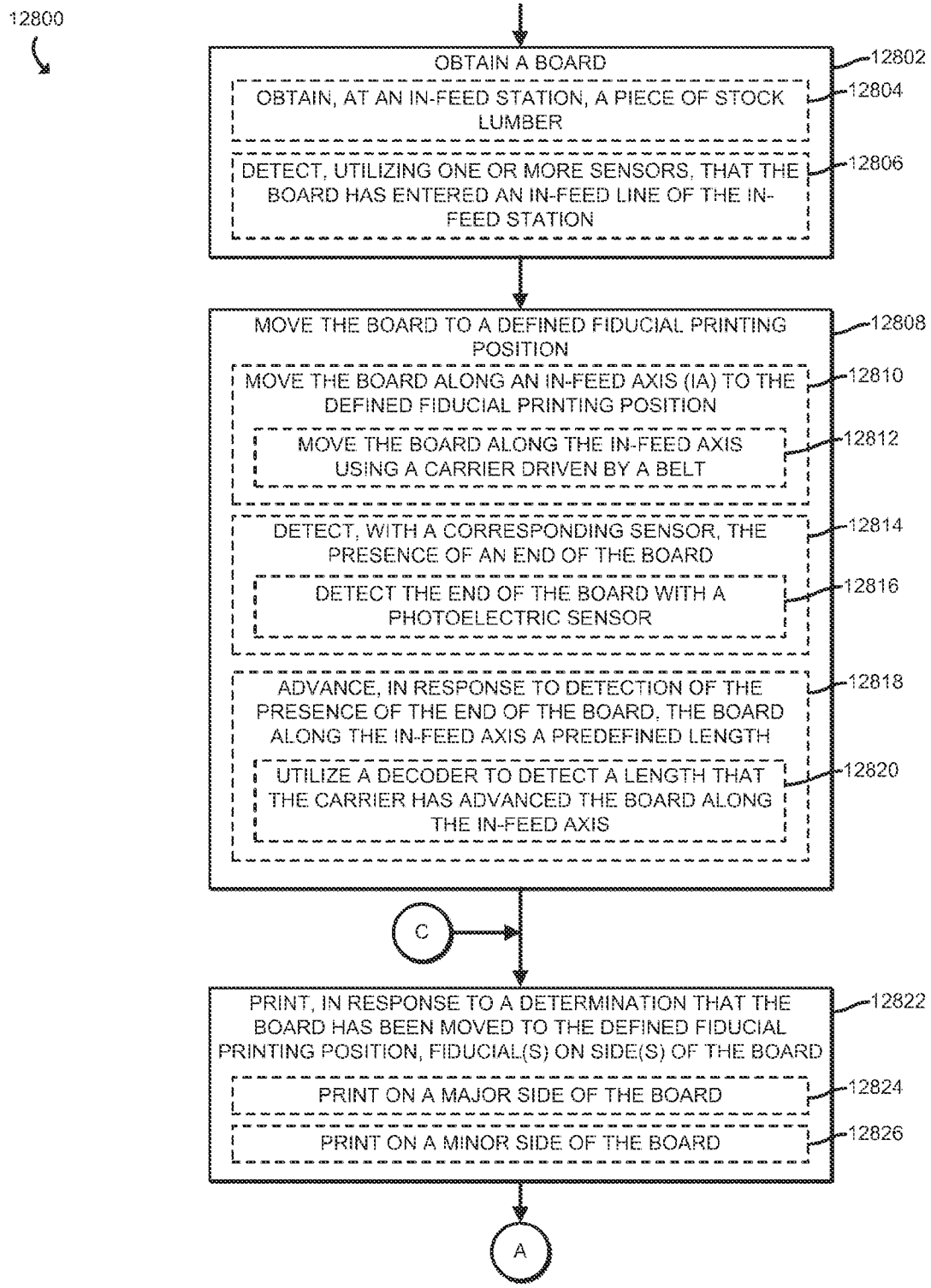

12800
OBTAIN A BOARD
12802
OBTAIN, AT AN IN-FEED STATION, A PIECE OF STOCK LUMBER
12804
DETECT, UTILIZING ONE OR MORE SENSORS, THAT THE BOARD HAS ENTERED AN IN-FEED LINE OF THE IN-FEED STATION
12806
MOVE THE BOARD TO A DEFINED FIDUCIAL PRINTING POSITION
12808
MOVE THE BOARD ALONG AN IN-FEED AXIS (IA) TO THE DEFINED FIDUCIAL PRINTING POSITION
12810
MOVE THE BOARD ALONG THE IN-FEED AXIS USING A CARRIER DRIVEN BY A BELT
12812
DETECT, WITH A CORRESPONDING SENSOR, THE PRESENCE OF AN END OF THE BOARD
12814
DETECT THE END OF THE BOARD WITH A PHOTOELECTRIC SENSOR
12816
ADVANCE, IN RESPONSE TO DETECTION OF THE PRESENCE OF THE END OF THE BOARD, THE BOARD ALONG THE IN-FEED AXIS A PREDEFINED LENGTH
12818
UTILIZE A DECODER TO DETECT A LENGTH THAT THE CARRIER HAS ADVANCED THE BOARD ALONG THE IN-FEED AXIS
12820
C
PRINT, IN RESPONSE TO A DETERMINATION THAT THE BOARD HAS BEEN MOVED TO THE DEFINED FIDUCIAL PRINTING POSITION, FIDUCIAL(S) ON SIDE(S) OF THE BOARD
12822
PRINT ON A MAJOR SIDE OF THE BOARD
12824
PRINT ON A MINOR SIDE OF THE BOARD
12826
A

FIG. 128

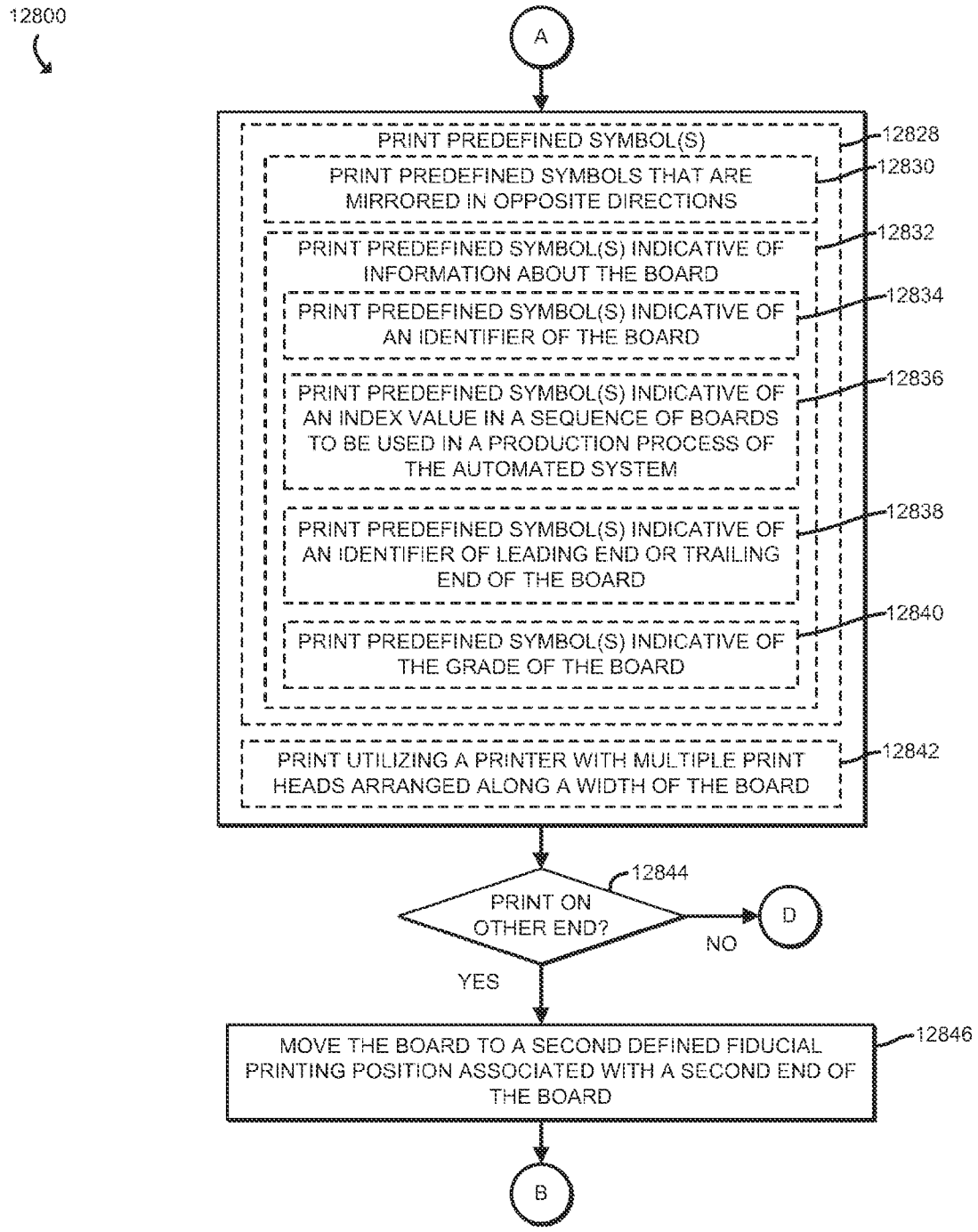

12800
A
PRINT PREDEFINED SYMBOL(S)
12828
PRINT PREDEFINED SYMBOLS THAT ARE MIRRORED IN OPPOSITE DIRECTIONS
12830
PRINT PREDEFINED SYMBOL(S) INDICATIVE OF INFORMATION ABOUT THE BOARD
12832
PRINT PREDEFINED SYMBOL(S) INDICATIVE OF AN IDENTIFIER OF THE BOARD
12834
PRINT PREDEFINED SYMBOL(S) INDICATIVE OF AN INDEX VALUE IN A SEQUENCE OF BOARDS TO BE USED IN A PRODUCTION PROCESS OF THE AUTOMATED SYSTEM
12836
PRINT PREDEFINED SYMBOL(S) INDICATIVE OF AN IDENTIFIER OF LEADING END OR TRAILING END OF THE BOARD
12838
PRINT PREDEFINED SYMBOL(S) INDICATIVE OF THE GRADE OF THE BOARD
12840
PRINT UTILIZING A PRINTER WITH MULTIPLE PRINT HEADS ARRANGED ALONG A WIDTH OF THE BOARD
12842
12844
PRINT ON OTHER END?
NO
D
YES
MOVE THE BOARD TO A SECOND DEFINED FIDUCIAL PRINTING POSITION ASSOCIATED WITH A SECOND END OF THE BOARD
12846
B

FIG. 129

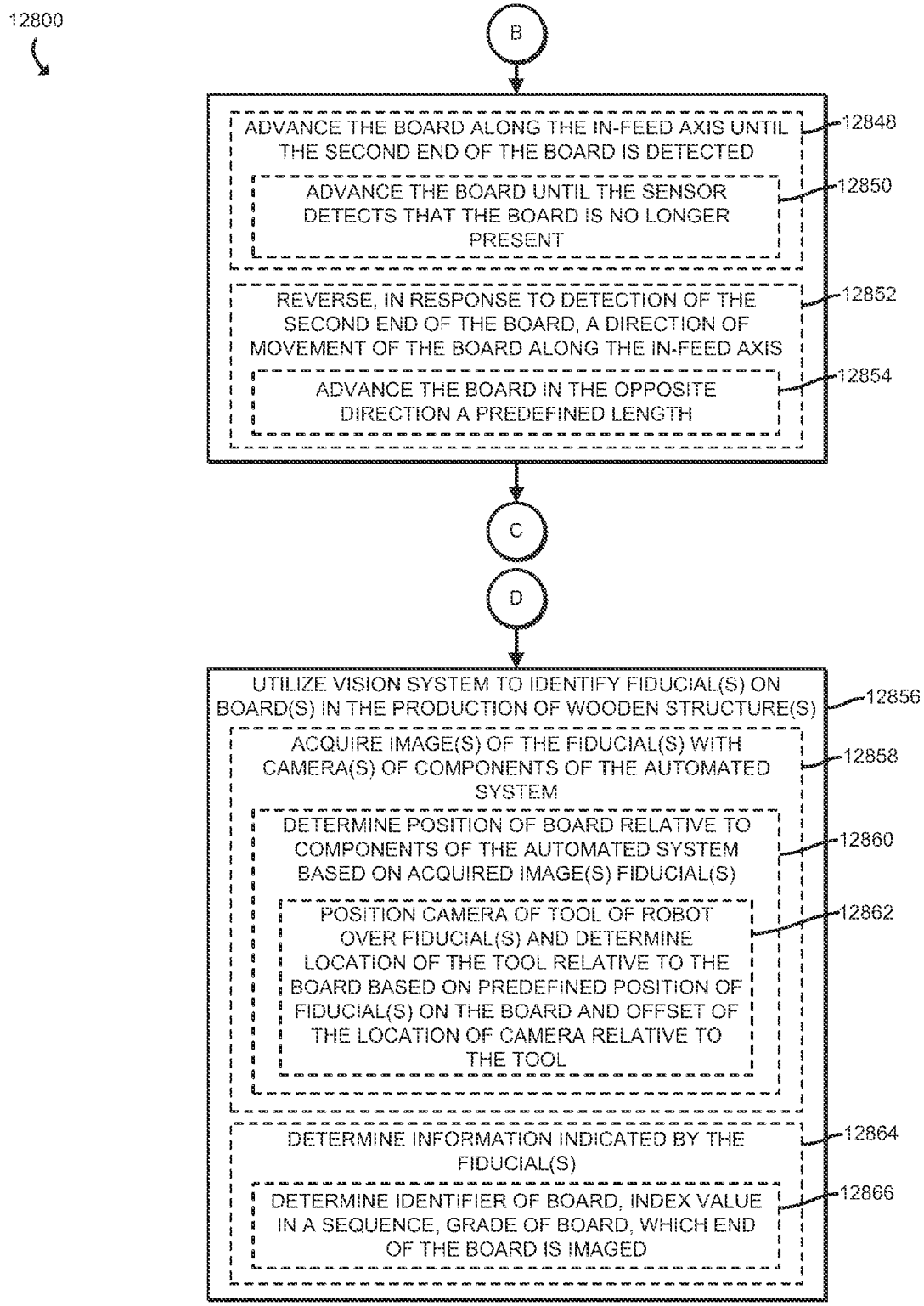

12800
B
ADVANCE THE BOARD ALONG THE IN-FEED AXIS UNTIL THE SECOND END OF THE BOARD IS DETECTED
12848
ADVANCE THE BOARD UNTIL THE SENSOR DETECTS THAT THE BOARD IS NO LONGER PRESENT
12850
REVERSE, IN RESPONSE TO DETECTION OF THE SECOND END OF THE BOARD, A DIRECTION OF MOVEMENT OF THE BOARD ALONG THE IN-FEED AXIS
12852
ADVANCE THE BOARD IN THE OPPOSITE DIRECTION A PREDEFINED LENGTH
12854
C
D
UTILIZE VISION SYSTEM TO IDENTIFY FIDUCIAL(S) ON BOARD(S) IN THE PRODUCTION OF WOODEN STRUCTURE(S)
12856
ACQUIRE IMAGE(S) OF THE FIDUCIAL(S) WITH CAMERA(S) OF COMPONENTS OF THE AUTOMATED SYSTEM
12858
DETERMINE POSITION OF BOARD RELATIVE TO COMPONENTS OF THE AUTOMATED SYSTEM BASED ON ACQUIRED IMAGE(S) FIDUCIAL(S)
12860
POSITION CAMERA OF TOOL OF ROBOT OVER FIDUCIAL(S) AND DETERMINE LOCATION OF THE TOOL RELATIVE TO THE BOARD BASED ON PREDEFINED POSITION OF FIDUCIAL(S) ON THE BOARD AND OFFSET OF THE LOCATION OF CAMERA RELATIVE TO THE TOOL
12862
DETERMINE INFORMATION INDICATED BY THE FIDUCIAL(S)
12864
DETERMINE IDENTIFIER OF BOARD, INDEX VALUE IN A SEQUENCE, GRADE OF BOARD, WHICH END OF THE BOARD IS IMAGED
12866

FIG. 130