Office de la Propriété Intellectuelle du Canada
Un organisme d'Industrie Canada

Canadian Intellectual Property Office
An agency of Industry Canada

CA 2773712 A1 2011/03/24

(21) **2 773 712**

(12) **DEMANDE DE BREVET CANADIEN**
CANADIAN PATENT APPLICATION

(13) **A1**

(86) Date de dépôt PCT/PCT Filing Date: 2010/09/16
(87) Date publication PCT/PCT Publication Date: 2011/03/24
(85) Entrée phase nationale/National Entry: 2012/03/08
(86) N° demande PCT/PCT Application No.: US 2010/049187
(87) N° publication PCT/PCT Publication No.: 2011/035069
(30) Priorité/Priority: 2009/09/15 (US12/559,698)

(51) Cl.Int./Int.Cl. *B25J 9/16* (2006.01)
(71) Demandeur/Applicant:
HARRIS CORPORATION, US
(72) Inventeurs/Inventors:
BOSSCHER, PAUL MICHAEL, US;
HEDMAN, DANIEL RYAN, US
(74) Agent: GOUDREAU GAGE DUBUC

(54) Titre : APPAREIL ROBOTIQUE METTANT EN OEUVRE UN MECANISME ANTICOLLISION ET PROCEDES ASSOCIES
(54) Title: ROBOTIC APPARATUS IMPLEMENTING COLLISION AVOIDANCE SCHEME AND ASSOCIATED METHODS

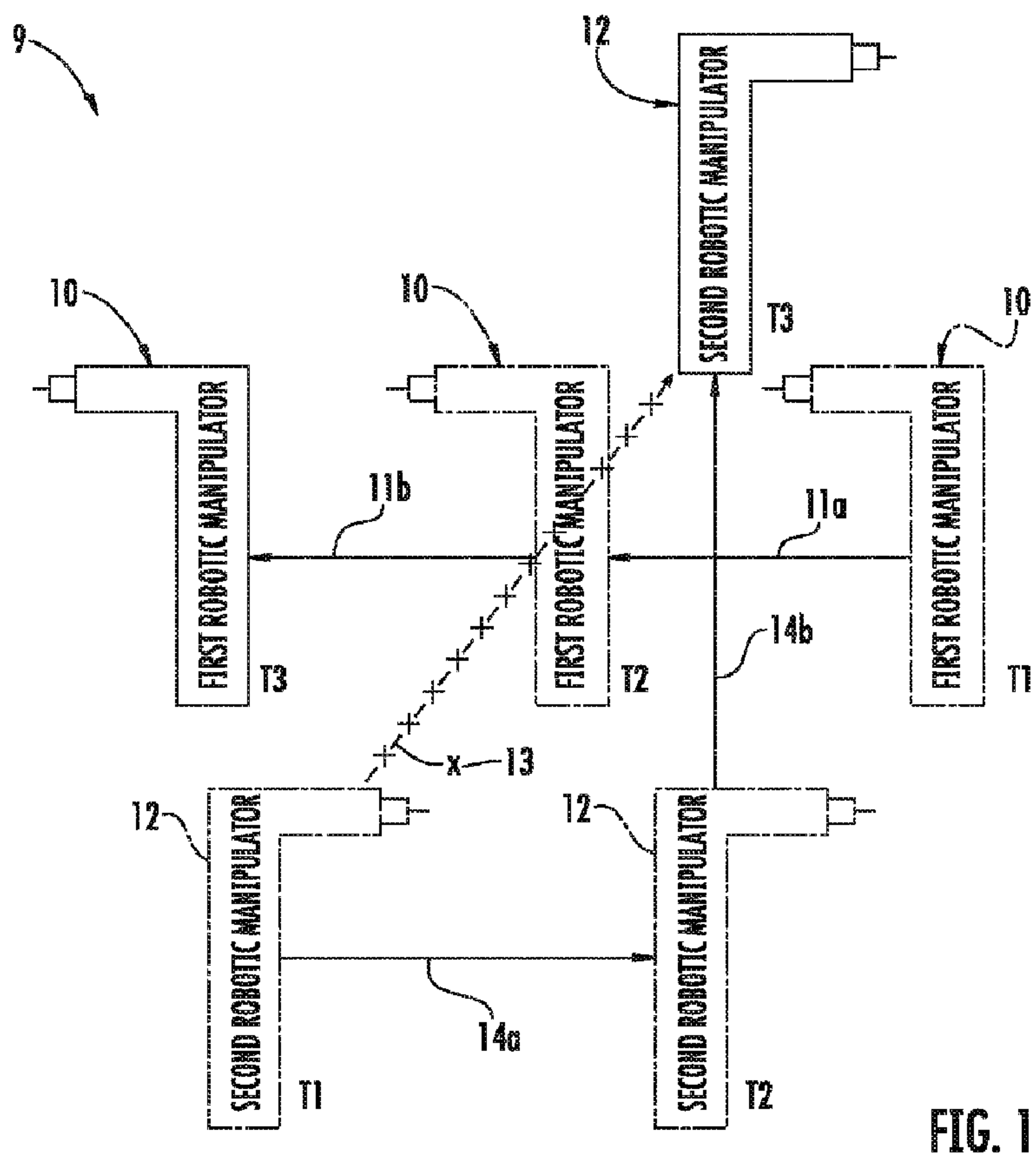


FIG. 1

(57) Abrégé/Abstract:
A robotic system implements a collision avoidance scheme and includes a first robotic manipulator and a first controller configured to control the first robotic manipulator for movement along a first pre-planned actual path. A second controller is configured to

(57) Abrégé(suite)/Abstract(continued):
control movement of a second robotic manipulator for movement along a second pre-planned intended path and deviating therefrom to move in a dodging path away from the first pre-planned actual path based upon determining a potential collision with the first robotic manipulator without prior knowledge of the first pre-planned actual path.

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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
24 March 2011 (24.03.2011)

PCT

(10) International Publication Number
WO 2011/035069 A3

(51) International Patent Classification:
B25J 9/16 (2006.01)

(21) International Application Number:
PCT/US2010/049187

(22) International Filing Date:
16 September 2010 (16.09.2010)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
12/559,698 15 September 2009 (15.09.2009) US

(71) Applicant *(for all designated States except US)*: **HARRIS CORPORATION** [US/US]; 1025 W. NASA Blvd., MS A-11I, Melbourne, Florida 32919 (US).

(72) Inventors: **BOSSCHER, Paul Michael**; 789 Wyeth Street, West Melbourne, Florida 32904 (US). **HEDMAN, Daniel Ryan**; 1147 Brook Street, NE, Palm Bay, Florida 32905 (US).

(74) Agents: **YATSKO, Michael, S.** et al.; Harris Corporation, 1025 W. NASA Blvd., MS A-11I, Melbourne, Florida 32919 (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, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, 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, MK, MT, NL, NO, PL, PT, RO, SE, SI, SK,

[Continued on next page]

(54) Title: ROBOTIC APPARATUS IMPLEMENTING COLLISION AVOIDANCE SCHEME AND ASSOCIATED METHODS

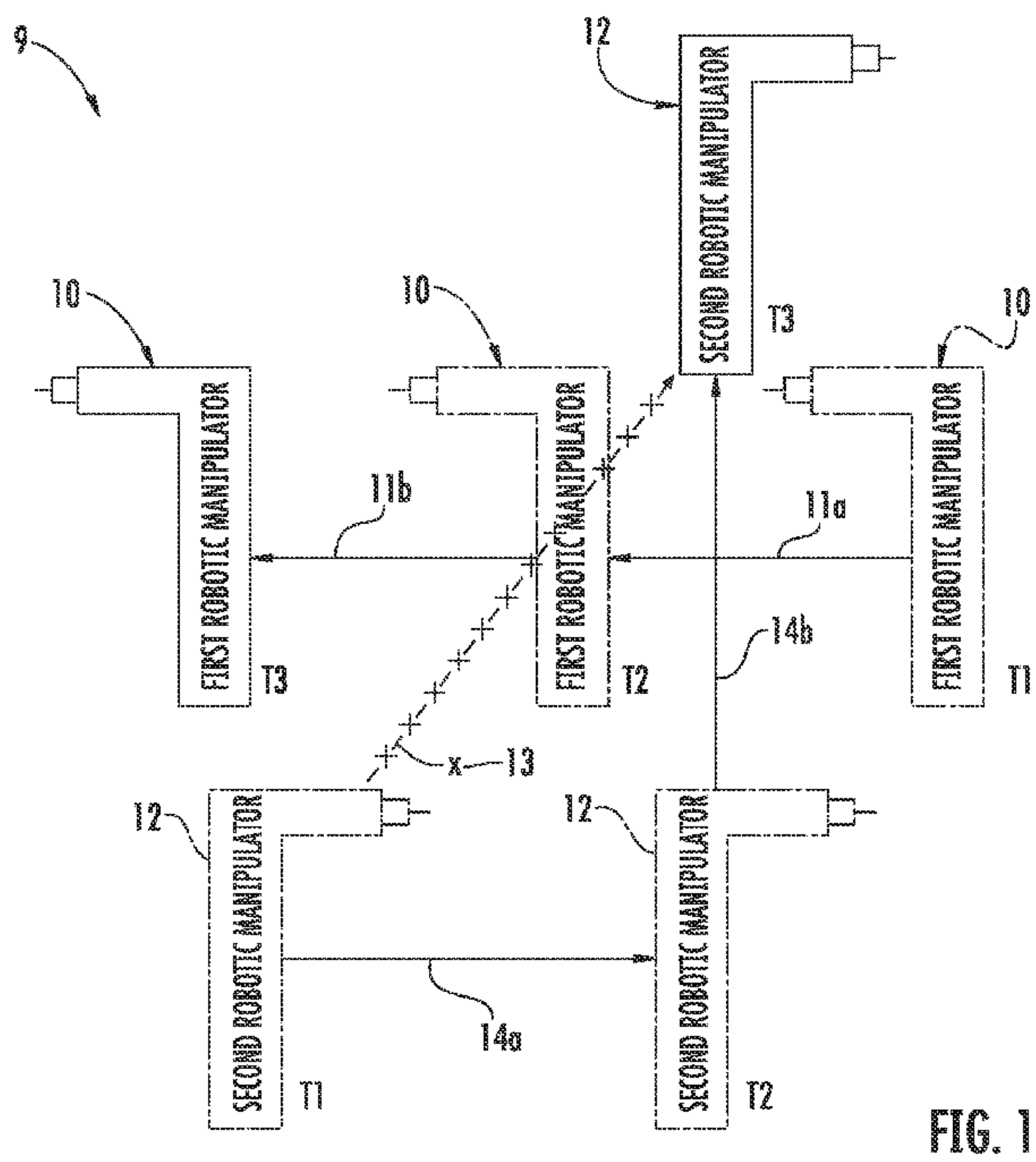


FIG. 1

(57) Abstract: A robotic system implements a collision avoidance scheme and includes a first robotic manipulator and a first controller configured to control the first robotic manipulator for movement along a first pre-planned actual path. A second controller is configured to control movement of a second robotic manipulator for movement along a second pre-planned intended path and deviating therefrom to move in a dodging path away from the first pre-planned actual path based upon determining a potential collision with the first robotic manipulator without prior knowledge of the first pre-planned actual path.

WO 2011/035069 A3

SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

— *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments (Rule 48.2(h))*

— *with information concerning request for restoration of the right of priority in respect of one or more priority claims (Rules 26bis.3 and 48.2(b)(vii))*

(88) Date of publication of the international search report:
5 May 2011

ROBOTIC APPARATUS IMPLEMENTING COLLISION AVOIDANCE SCHEME AND ASSOCIATED METHODS

The present invention relates to the field of robotics, and, more particularly, to collision avoidance for robotic manipulators and related methods. Robotic systems are commonplace in fields such as manufacturing. Indeed, manufacturing plants typically employ robotic systems including numerous robotic manipulators to perform various tasks. To avoid damage to the robotic manipulators, it is helpful to control the robotic manipulators according to a collision avoidance scheme. As such, a variety of collision avoidance schemes for robotic systems have been developed.

Some collision avoidance schemes work by constraining each robotic manipulator to pre-planned collision free paths. For example, one such collision avoidance scheme is disclosed in U.S. Pat. No. 5,204,942 to Otera et al. Such a collision avoidance scheme typically requires reprogramming to accommodate each and every change made to the pre-planned paths of the robotic manipulators. In a manufacturing process that is routinely altered and updated, the collision avoidance system of Otera et al. may be disadvantageous due to the necessary repeated reprogramming thereof.

Other collision avoidance schemes may model a workspace and divide it into different zones. Certain robotic manipulators may be forbidden to enter certain zones, or only one robotic manipulator may be allowed into a given zone at a time. U.S. Pat. No. 5,150,452 to Pollack et al. discloses such a collision avoidance scheme for a robotic system. The robotic system includes a controller storing a model of the workspace that is divided into an occupancy grid. The controller controls the robotic manipulators of the robotic system such that only one robotic manipulator may occupy a cell of the occupancy grid at a given time. This collision avoidance system may reduce the efficiency of a manufacturing plant, particularly if there are a variety of differently sized robotic manipulators and the cell sizes of the occupancy grid are sized to accompany the largest robotic manipulators. Further, since this robotic system operates based upon a model of the workspace, any change to the workplace may require an update of the model, which may be time consuming.

Other attempts at collision avoidance schemes for robotic systems include a controller that actively looks for potential collisions between robotic manipulators. For example, U.S. Pat. No. 4,578,757 to Stark discloses a collision avoidance scheme for a robotic system that models each robotic manipulator of the system as a number of overlapping spheres. As the robotic manipulators move along pre-planned paths, the distance between each sphere of nearby robotic manipulators is calculated by a controller. These calculated distances indicate a risk of collision between two adjacent robotic manipulators. When the risk of collision exceeds a threshold amount, at least one of the robotic manipulators may be slowed down as it travels along its pre-planned path, or even stopped completely. Such a collision avoidance scheme, however, may reduce the efficiency of a manufacturing plant employing the robotic system due to the stopping of robotic manipulators and the associated delays in the manufacturing process.

As explained, these prior approaches may render a manufacturing process employing their respective robotic systems inefficient. Moreover, robotic systems employing these prior approaches may be difficult and/or costly to adapt to new applications or to the addition of additional robotic manipulators. As such, further advances in the field of collision avoidance schemes may be desirable.

In view of the foregoing background, it is therefore an object of the present invention to provide a more efficient collision avoidance scheme for a robotic apparatus.

This and other objects, features, and advantages in accordance with the present invention are provided by a robotic manipulator that detects a potential collision with another robotic manipulator, and moves in a dodging path based upon detection of the potential collision. More particularly, the robotic apparatus may comprise a first robotic manipulator, and a first controller configured to control the first robotic manipulator for movement along a first pre-planned actual path. In addition, the robotic apparatus may include a second robotic manipulator, and a second controller configured to control movement of the second robotic manipulator for movement along a second pre-planned intended path. The second robotic manipulator deviates therefrom to move in a dodging path away from the first pre-planned actual path based upon determining a potential collision with the first robotic manipulator and without prior knowledge of the first pre-planned actual path.

This collision avoidance scheme advantageously allows the first pre-planned actual path of the first robotic manipulator to be reprogrammed without necessitating a reprogramming of the second pre-planned intended path. This may reduce the time it takes to adapt the robotic apparatus to a new application.

The first controller may generate first drive signals for the first robotic manipulator, and the second controller may determine the potential collision based upon the first drive signals. Additionally or alternatively, the first robotic manipulator may include at least one joint, and a joint sensor cooperating with the at least one joint and the first controller for determining positioning of the at least one joint. The second controller may determine the potential collision based upon the positioning of the at least one joint of the first robotic apparatus. This may allow the determination of potential collisions more accurately and with the use of less processing power than through the use of an image sensor. Of course, the second controller may also determine the potential collision based upon an image sensor or joint position sensor in some embodiments.

The second controller may repeatedly determine a distance between the second robotic manipulator and the first robotic manipulator, and, may compare the distance to a threshold distance to thereby determine a potential collision. In addition, the second controller may also repeatedly determine an approach velocity between the second robotic manipulator and the first robotic manipulator, and the second controller may also determine the potential collision based upon the approach velocity.

Furthermore, the second controller may also repeatedly determine an approach velocity between the second robotic manipulator and the first robotic manipulator, and the second controller may also determine the potential collision based upon the approach velocity. The second controller may move the second robotic manipulator at different speeds based upon the approach velocity. The second controller may also repeatedly determine an acceleration of the first robotic manipulator, and the second controller may also determine the potential collision based upon the acceleration.

In addition, the second controller may store a geometric model of the first and second robotic manipulators, and the second controller may determine the distance between the second robotic manipulator and the first robotic manipulator based upon the geometric models. The use of geometric models may greatly reduce the processing power consumed in determining potential collisions.

Each geometric model may include a series of buffer zones surrounding a respective robotic manipulator. The second controller may determine a potential collision between the second robotic manipulator and the first robotic manipulator based upon an overlap between the buffer zones. In addition, the second controller may move the second robotic manipulator at different speeds based upon which respective buffer zones are overlapping.

The second pre-planned intended path may be based upon a sequence of desired velocities. The second controller may move the second robotic manipulator along the dodging path based upon a sequence of dodge velocities to avoid the potential collisions while closely following the sequence of desired velocities of the second pre-planned intended path.

The second robotic manipulator may comprise at least one joint, and the second controller may determine the sequence of dodge velocities based upon a force on the at least one joint. Additionally or alternatively, the second controller may also determine the sequence of dodge velocities upon kinetic energy of the second robotic manipulator. Determining the sequence of dodge velocities based upon a force or torque on the at least one joint, or based upon kinetic energy of the second robotic manipulator, may advantageously constrain the second robotic manipulator from advancing along a dodge velocity or path that would be potentially damaging to its hardware.

The sequence of dodge velocities may comprise at least one velocity in each of a plurality of physical directions. The second controller may determine the sequence of dodge velocities based upon a plurality of convex sets of allowable velocities.

A method aspect is directed to a method of operating a robotic apparatus according to a collision avoidance scheme to avoid a collision with a first robotic manipulator controlled by a first controller for movement along a first pre-planned actual path. The method may comprise controlling a second robotic manipulator with a second controller for movement along a second pre-planned intended path. The method may further include controlling the second robotic manipulator with the second controller for movement in a dodging path away from the first pre-planned actual path based upon determining a potential collision with the first robotic manipulator and without prior knowledge of the first pre-planned actual path.

FIG. 1 is a schematic side view of a robotic apparatus implementing a collision avoidance scheme, in accordance with the present invention.

FIG. 2 is schematic perspective of the robotic apparatus of FIG. 1.

FIG. 3 is a schematic block diagram of a further embodiment of a robotic apparatus implementing a collision avoidance scheme, in accordance with the present invention.

FIG. 4 is a graph illustrating the selection of a dodge velocity by the robotic apparatus of FIG. 3.

FIG. 5 is a schematic block diagram of yet another embodiment of a robotic apparatus implementing a collision avoidance scheme, in accordance with the present invention.

FIG. 6 is a perspective view of the robotic apparatus of FIG. 5.

FIG. 7 is a schematic side view of a robotic manipulator surrounded by a series of buffer zones, in accordance with the present invention.

FIG. 8 is a flowchart of a method of operating a robotic apparatus according to a collision avoidance scheme, in accordance with the present invention.

FIG. 9A is a schematic side view of first and second robotic manipulators.

FIG. 9B is a schematic side view of first and second robotic manipulators including vectors used in determining the dodging path.

FIG. 9C is a schematic side view of first and second robotic manipulators including vectors used in determining the dodging path.

FIG. 10 is a chart illustrating the calculation of an approach velocity using in determining the dodging path.

The present invention will now be described more fully hereinafter with reference to the accompanying drawings, in which preferred embodiments of the invention are shown. This invention may, however, be embodied in many different forms and should not be construed as limited to the embodiments set forth herein. Rather, these embodiments are provided so that this disclosure will be thorough and complete, and will fully convey the scope of the invention to those skilled in the art. Like numbers refer to like elements throughout, and prime and multiple prime notations are used to indicate similar elements in alternative embodiments.

Referring initially to FIGS. 1-2, a robotic apparatus **9** implementing a collision avoidance scheme is now described. The robotic apparatus **9** includes a first robotic manipulator **10** and a first controller **15** configured to control the first robotic manipulator for movement along a first pre-planned actual path **11a, 11b**. In addition, there is a second robotic manipulator **12** and a second controller **20.** The second controller **20** is configured to control movement of the second robotic manipulator **12** for movement along a second pre-planned intended path **13**. The second controller **20**, without prior knowledge of the first pre-planned actual path **11a, 11b**, causes the second robotic manipulator **12** to deviate from the second pre-planned intended path **13** to move in a dodging path **14a, 14b** away from the first pre-planned actual path, and thus the first robotic manipulator **10**, based upon determining a potential collision therewith. Further details of the derivation of the dodging path **14a, 14b** will be given below.

Those skilled in the art will recognize that the robotic apparatus **9** may include any number of robotic manipulators and that the first and second robotic manipulators **10, 12** may be any suitable robotic manipulators, for example, robotic welding arms, or robotic claws for handling objects and/or tools. Of course, the first and second robotic manipulators **10, 12** may be different types of robotic manipulators and may vary in size.

The first pre-planned actual path **11a, 11b,** in other embodiments, may include continuous or discontinuous movement in any direction, and may include movement of joints **26** of the first robotic manipulator **10**. Likewise, the second pre-planned intended path **13** may, in other embodiments include continuous or discontinuous movement in any direction, and may include movement of joints **27** of the second robotic manipulator **12**. In the embodiment as shown in FIG 1, the paths have been simplified to linear movements for clarity of explanation.

The application of this collision avoidance scheme, at three discrete moments in time, is illustrated in FIG 1. Here, the first robotic manipulator **10** is being controlled by the first controller **15** (FIG. 2) for movement along the first pre-planned actual path **11**, including a first segment **11a** beginning at a first time (T1), continuing along a second segment **11b** at a second time (T2) and ending at a third time (T3).

A second controller **20** (FIG. 2) is controlling the second robotic manipulator **12** for attempted movement along the second pre-planned intended path **13**. However, at T1, the second controller **20** detects that the second robotic manipulator **12** would collide with the first robotic manipulator **10**. Therefore, the second controller **20** causes the second robotic manipulator **12** to instead follow the dodge path, segments **14a, 14b**, so that, at T2 and T3, it does not collide with the first robotic manipulator **10**. The dodge path **14am, 14b** not only takes the second robotic manipulator **12** out of danger of colliding with the first robotic manipulator **10**, but may also advantageously follow the second pre-planned intended path **13** as closely as possible. In the illustrated example, the second robotic manipulator **12** illustratively ends up in a same position at T3 as where it would have been had it followed the second pre-planned intended path **13**.

Those skilled in the art will understand that the second controller **20** may repeatedly search for potential collisions with the first robotic manipulator **10**, and may repeatedly adjust the dodge path **13** based thereupon, for example, every 2 milliseconds.

A further embodiment of the robotic apparatus **9'** is now described with reference to FIG. 3. Here, the first controller **15'** comprises a memory **17'** cooperating with a processor **16'** for controlling the first robotic manipulator **10'** for movement along the first pre-planned actual path. The processor **16'** sends drive instructions to the drive signal generator **18'**, which in turn generates and sends first drive signals to the first robotic manipulator **10'**. The first robotic manipulator **10'** moves along the first pre-planned actual path based upon the first drive signals.

The second controller **20'** comprises a processor **21'** and a memory **22'** cooperating for controlling movement of the second robotic manipulator **12'** along the second pre-planned intended path. Here, the second pre-planned intended path is based upon a sequence of desired velocities. The processor **21'** sends drive instructions to the drive signal generator **23'**, which in turn generates and sends second drive signals to the second robotic manipulator **12'**. The second robotic manipulator **12'** moves along the second pre-planned intended path based upon the second drive signals.

The processor **21'** determines a potential collision with the first robotic manipulator **10'** without prior knowledge of the first pre-planned actual path. To effectuate this determination, a communications interface **24'** of the second controller **20'** is coupled to the communications interface **19'** of the first controller **15'** to read the first drive signals. The processor **21'** cooperates with the communications interface **24'** to thereby determine a potential collision based upon the first drive signals. This advantageously allows a quick and accurate determination of the velocity of each portion of the first robotic manipulator **10'** and thus the first pre-planned actual path.

When the processor **21'** determines a potential collision of the second robotic manipulator **12'** and the first robotic manipulator **10'**, it causes the second robotic manipulator to deviate from the second pre-planned path and instead move in a dodging path (away from the first pre-planned actual path and thus the first robotic manipulator) based upon a sequence of dodge velocities.

The processor **21'** may select each of the sequence of dodge velocities from a set of potential dodge velocities. Such a set of potential dodge velocities is illustrated in the two-dimensional graph of FIG. 5. The crosshatched areas in FIG. 5 indicate velocities that, if followed by the second robotic manipulator **12'**, would result in a collision with the first robotic manipulator **10'**. The polygonal non-crosshatched area of FIG. 5 indicates a set of collision free dodge velocities. The second pre-planned intended path may be based upon a sequence of desired velocities, and, in selecting each of the sequence of dodge velocities, the processor **21'** may choose dodge velocities closest to the desired velocities. As shown in FIG 4, the selected velocity SV is shown adjacent to the intended velocity IV.

The dodge velocities may also be based, among other factors, on a torque on a joint of the second robotic manipulator **12'** and/or a kinetic energy of the second robotic manipulator (or based upon some objective function dealing with the robot - some kinematic, dynamic, etc., rule for choosing desired velocities - however optimal is defined for the robot system). For example, certain joints of the second robotic manipulator **12'** may have a torque limit, and moving the robotic manipulator at a velocity that would cause those joints to exceed that torque limit is undesirable. Basing the dodge velocities on a torque on a joint or a kinetic energy of the second robotic manipulator may advantageously help ensure that the second robotic manipulator **12'** is not damaged during movement according to the sequence of dodge velocities, or that a tool or object carried by the second robotic manipulator is not damaged during movement according to a dodge velocity. The determination of these dodge velocities will be described in detail below.

With additional reference to FIG. 5, yet another embodiment of a robotic apparatus **9"** implementing a collision avoidance scheme is now described. The first robotic manipulator **10"** and first controller **15"** are similar in structure and function to the first robotic manipulator **10’** and first controller **15’** described above, and thus no further description thereof is needed.

The second controller **20”** comprises a processor **21”,** memory **22”**, drive signal generator **23”**, and communications interface **24”** as described above with reference to the second controller **20’**. However, here, the memory **22”** stores geometric models of the first and second robotic manipulators. The processor **21”** of the second controller **20”** may generate the geometric models, or this data may be communicated to the second controller and stored in the memory **22”** thereof.

To generate the geometric models, each robotic manipulator **10”, 12”** is first represented as a set of geometric primitives. The geometric primitives include points, line segments, and rectangles. To complete the geometric models, the first and second robotic manipulators **10”, 12”** as then represented as swept spherical bodies. These swept spherical bodies comprise the set of points that are at a specified distance/radius from a respective geometric primitive. If the primitive is a point, the resulting body is a sphere. If the primitive is a line segment, the body is a cylinder with spherical endcaps, also known as a cylisphere. If the primitive is a rectangle, the body is a box with rounded edges. For ease of reading, these bodies will hereinafter be referred to as “cylispherical shells,” but it should be appreciated that they may take other shaped, as described above.

As shown in FIG. 6, a geometrically complex set of objects (here, robotic manipulators) can be approximated by a collection of cylispherical shells. To more clearly illustrate the distinct cylispherical shells, the geometric primitive for each shell is superimposed over the shell. Each robotic manipulator **10”, 12”** in FIG. 6 is approximated by eight cylispherical shells (as can be seen from the eight line segments for each). As a result, the complete geometric model of the robotic manipulators **10”, 12”** and their surroundings at a given instant can be represented by a list of bodies, the respective body types (i.e. point, line segment, or plane), the locations of the corner/end points on each body, and the radius associated with each body.

The primary advantage of using such a method for approximating the robotic manipulators of the robotic apparatus is the simplicity in calculating the distance between the robotic manipulators. The distance $d(j,k)$ between objects j and k is simply

$$d(j,k) = d_p(j,k) - r_j - r_k \qquad (1)$$

where $d_p(j,k)$ is the distance between the primitives of objects j and k and r_j and r_k are the radii of objects j and k respectively. Calculating the distance between the primitives (points, line segments, rectangles) has a relatively straightforward closed-form solution in some cases.

Increasing the number of cylispherical shells used per robotic manipulator can improve the accuracy of the geometric model, but at the cost of increased computation time. Those skilled in the art will appreciate that this is not the only possible method of simplifying the geometric model of the robotic apparatus. Any representation of the robotic apparatus that allows geometric calculations to be performed quickly would be an acceptable substitute.

As will be described below with respect to the calculation of a sequences of dodge velocities, a constraint generation portion of this collision avoidance scheme creates limits on the motion of the second robotic manipulator **12"** based on how close it is to colliding with the first robotic manipulator **10"**. This is accomplished by creating a set of three buffer zones **26", 27", 28"** for each geometric primitive. This is illustrated in FIG. 7, where the geometry of one link of the second robotic manipulator **12"** has been modeled. The geometric primitive chosen is a line segment (not shown). Cylispherical shells with three different radii are then constructed from this single geometric primitive.

In applying this collision avoidance scheme to the robotic apparatus of FIG. 5, the buffer zones **26", 27", 28"** may be the same shape as the geometric models, for example, and the processor **21"** may alter the dodge path based upon detecting an overlap between buffer zones of the first and second robotic manipulators **10", 12"**.

Referring still to FIG. 7, the buffer zones include a reaction buffer zone **26"**, an equilibrium buffer zone **27"**, and an object buffer zone **28"**. The reaction buffer zone **26"** and equilibrium buffer zone **27"** are used to gradually apply changes to the motion of the second robotic manipulator **12"** as it approaches the first robotic manipulator **10"**. The object buffer zone **28"** is the lower bound on how close the second robotic manipulator **12"** should be allowed to be to the first robotic manipulator **10"**. In particular, the object buffer zone **28"** of the second robotic manipulator **12"** should not be allowed to contact the object buffer zone of the first robotic manipulator or a collision may result.

Referring again to FIG. 5, the processor **21"** controls movement of the second robotic manipulator **12"** for movement along the second pre-planned intended path via the drive signal generator **23"**, as described above. One or more visual sensors, illustratively a visual sensor **25",** (can be any type of sensors, position sensor, motion sensor, etc.) are coupled to the communications interface **24".** Those skilled in the art will understand that the image sensor (visual sensor **25")** may be any suitable image sensor, such as a camera, and that there may be a plurality of such sensors. Additionally or alternatively, in other applications, there may be other types of sensors, for example radar or sonar.

The processor **21"** determines a distance between a shell of the second robotic manipulator **12"** and a shell of the first robotic manipulator **10",** based upon the visual sensor **25"**, to detect a potential collision between the first and second robotic manipulators **10", 12"**. The potential collision may be detected based upon a buffer zone of the second robotic manipulator **12"** overlapping a buffer zone of the first robotic manipulator **10"**.

In response to a potential collision, the processor **21"** controls the second robotic manipulator **12"** for movement in a dodge path to avoid a collision with the first robotic manipulator **10"**. The dodge path may depend upon which buffer zones of the first and second robotic manipulators **10", 12"** overlap each other. For example, if the reaction buffer zones **26"** of the first and second robotic manipulators **10", 12"** overlap, the dodge path may take the second robotic manipulator **12"** away from the first robotic manipulator **10"**, but at a lesser speed than the approach speed thereof. Since the reaction **26"** buffer zones may be defined so as to be relatively large in comparison to the geometric model of their respective robotic manipulator, the second robotic manipulator **12"** may simply not need to move at a speed greater or equal to the approach speed of the first robotic manipulator **10"**. Movement at such a lesser speed along the dodge path may conserve power, or may reduce wear and tear on the second robotic manipulator **12"**.

If the equilibrium buffer zones **27"** of the first and second robotic manipulators **10", 12"** overlap each other, the dodge path may take the second robotic manipulator **12"** away from the first robotic manipulator **10"** at the approach speed thereof. This may help to avoid a collision that would otherwise be unavoidable were the second robotic manipulator **12"** to move at a slower speed. If the object buffer zones **28"** of the first and second robotic manipulators **10", 12"** contact each other, the dodge path may be an emergency stop. Moreover, if the object buffer zones **28"** of the first and second robotic manipulators **10", 12"** contact each other, the second controller **20"** may shut down the robotic apparatus **9"**.

For compactness, each geometric primitive and the three shells associated with it shall collectively be referred to as a "body," denoted b. The set of bodies can be divided into two sets. The set of "robotic manipulator" bodies, R, are the bodies that are a part of the second robotic manipulator. The set of "object" bodies, O, are the other bodies (including the first robotic manipulator).

For each robotic manipulator body $b_j \in R$ it is helpful to then determine the set of the bodies $b_k \in (R \cup O)$ that could possibly collide with it. This is done in three steps. The first step is performed off-line prior to executing the collision avoidance scheme, and includes manually removing potential collision pairs (b_j, b_k) from the list of possible collisions. Potential collision pairs removed in this step may not be checked for at any point in the collision avoidance scheme. This is primarily used to allow neighboring bodies to overlap each other without being flagged as a collision.

For example, in FIG. 6 some of the robotic manipulator bodies overlap each other. In this case it would be desirable ignore the overlap between neighboring bodies. However, this is not the case for all pairs of robot bodies. For example, it is desirable to prevent collision between certain bodies and the base of the robotic manipulator, the collision pair for those bodies would not be removed from the list of possible collisions.

The second step is performed after beginning execution of the collision avoidance scheme. The purpose of this step is to quickly check whether potential collision pairs (b_j, b_k) are far enough apart that they can be ignored at this instant. To implement this step, bounding boxes have been constructed around bodies in the workspace. The bounding boxes of each potential collision pair (b_j, b_k) are compared and if the bounding boxes do not overlap, that (b_j, b_k) pair can be excluded from the list of possible colliding bodies. This step is fast and may quickly eliminate potential collisions from consideration.

In the third step the actual distance between the bodies is calculated based upon the pairs (b_j, b_k) that were not eliminated in the first two steps. In particular, the distance between the geometric primitives associated with the bodies is calculated. Because of the simple geometric primitives chosen, this can be calculated very quickly. The output of this step is the shortest distance $d_p(j, k)$ between the primitives of (b_j, b_k), and the points on each of the primitives corresponding to this shortest distance. For the i^{th} pair of (b_j, b_k) considered, this distance is referred to as d_{p_i} and these points are referred to as cp_i (the “collision point”), and ip_i (the “interfering point”), where cp_i is on body b_j and ip_i is on body b_k.

It is worth noting that it may be helpful to modify the collision and distance check method described hereinbefore if a different geometric modeling approach were used.

Given the current state of the robotic apparatus (as represented by the geometric model) and the set of possible collisions, it is helpful to generate constraints on the allowed motion of the robot in order to avoid these collisions. In addition to avoiding collisions (both with the first robotic manipulator and with itself), it is helpful for the motion of the second robotic manipulator to not violate its joint angle limits or joint velocity limits.

To effectuate this, a set of linear inequality constraints on the commanded velocity of the first robotic manipulator is created. For convenience, these limits are formulated in the end-effector (task space) in the implementation described here. Note that because the Jacobian mapping of joint velocities to task space velocities is linear, these constraints can be expressed in either the joint space or the task space and they will still be linear. As a result, the set of allowable velocities of the robot at this instant forms a convex set. This structure is advantageous as it allows formulation of the velocity selection problem as a convex optimization. This optimization can be calculated very efficiently, which may allow collision-free motions to be computed in real time.

The form of the convex optimization problem is

$$\begin{array}{ll} \text{minimize} & f_0(\mathbf{x}) \\ \text{subject to} & f_i(\mathbf{x}) \leq 0, \quad i = 1, \ldots, \kappa \end{array} \qquad (2)$$

where there are κ simultaneous inequality constraints. The objective function $f_0(\mathbf{x})$ can be any function that produces desirable or "optimal" behavior when minimized. It is desirable to minimize the error between the actual end-effector linear and angular velocity $\begin{pmatrix} v^T & \omega^T \end{pmatrix}_{ee}^T$ and the desired end-effector linear and angular velocity $\begin{pmatrix} v^T & \omega^T \end{pmatrix}_{ee,d}^T$. Thus, the following function was chosen.

$$f_0(\mathbf{x}) = ((\mathbf{x} - \mathbf{x}_d)^T (\mathbf{x} - \mathbf{x}_d))^{\frac{1}{2}} \qquad (3)$$

where

$$\mathbf{x} = \mathbf{W} \begin{pmatrix} v \\ \omega \end{pmatrix}_{ee} \quad \mathbf{x}_d = \mathbf{W} \begin{pmatrix} v \\ \omega \end{pmatrix}_{ee,d} \qquad (4)$$

where $\mathbf{W}$ is a weighting matrix with appropriate unit terms such that all of the elements of $\mathbf{x}$ have the same units:

$$\mathbf{W} = \begin{bmatrix} \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \alpha \mathbf{I}_{3\times3} \end{bmatrix}. \qquad (5)$$

For example, a valid choice would be $\alpha = 1\frac{m}{rad}$. In other cases it may be desirable to have additional criteria included in the objective function. For example, with redundant manipulators additional terms can be added to the objective function to represent optimal use of the self-motion of the robot, provided the resulting $f_0(\mathbf{x})$ is a convex, twice-differentiable function.

Given the objective function, it is helpful to define the constraint functions $f_i(\mathbf{x})$. Constraints due to joint angle limits and joint velocity limits are generated.

There are two different cases of joint limits: joint angle limits and joint velocity limits. As explained above, it may be desirable to base the dodge velocities on these limits, so it is helpful to accommodate both cases with a single set of upper and lower joint velocity limits. To do this, a vector that contains all of the upper limits on allowable joint velocities at this instant is formed

$$\dot{\mathbf{q}}_{ul} = \begin{bmatrix} \dot{q}_{ul_1} \\ \vdots \\ \dot{q}_{ul_n} \end{bmatrix} \qquad (6)$$

where the value of each $\dot{q}_{ul_i}$ depends on the current angle of joint i:

$$\dot{q}_{ul_i} = \begin{cases} \dot{q}_{upper_i} & q_i < q_{upper_i} \\ 0 & q_i \geq q_{upper_i}. \end{cases} \qquad (7)$$

That is, the joint i is nominally upper bounded by a user-specified velocity limit (typically the rated velocity limit of the joint), but if joint i has begun to exceed its joint limit its upper velocity limit is 0 and the angle may not be allowed to increase further. A similar vector of lower limits on allowable joint velocities is created:

$$\dot{\mathbf{q}}_{ll} = \begin{bmatrix} \dot{q}_{ll_1} \\ \vdots \\ \dot{q}_{ll_n} \end{bmatrix} \qquad (8)$$

where

$$\dot{q}_{ll_i} = \begin{cases} \dot{q}_{lower_i} & q_i > q_{lower_i} \\ 0 & q_i \leq q_{lower_i}. \end{cases} \qquad (9)$$

Given the values for the upper and lower joint limits in (7) and (9) the instantaneous limit on $\dot{q}_i$ can be stated as

$$\dot{q}_{ll_i} \leq \dot{q}_i \leq \dot{q}_{ul_i}. \tag{10}$$

Equation (10) may be restated using (6) and (8) as

$$\hat{\mathbf{s}}_i^T \dot{\mathbf{q}}_{ll} \leq \hat{\mathbf{s}}_i^T \dot{\mathbf{q}} \leq \hat{\mathbf{s}}_i^T \dot{\mathbf{q}}_{ul} \tag{11}$$

where $\hat{\mathbf{s}}_i$ is a vector that selects the terms corresponding to the i^{th} joint:

$$\hat{\mathbf{s}}_1 = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}^T. \tag{12}$$

Note that $\hat{()}$ notation is used to indicate that a vector is a unit vector. The constraint functions $f_i(\mathbf{x})$ corresponding to the right portion of (11) (i.e. the upper limits) are:

$$\hat{\mathbf{s}}_i^T \dot{\mathbf{q}} \leq \hat{\mathbf{s}}_i^T \dot{\mathbf{q}}_{ul}. \tag{13}$$

using the Jacobian relationship for serial manipulators

$$\begin{pmatrix} v \\ \omega \end{pmatrix}_{ee} = \mathbf{J}\,\dot{\mathbf{q}} \tag{14}$$

and assuming that the second robotic manipulator is not in a singular configuration, $\mathbf{J}$ can be inverted and substitute (14) into (13):

$$\hat{\mathbf{s}}_i^T \mathbf{J}^{-1} \begin{pmatrix} v \\ \omega \end{pmatrix}_{ee} \leq \hat{\mathbf{s}}_i^T \dot{\mathbf{q}}_{ul}. \tag{15}$$

Using the fact that $\mathbf{W}^{-1}\mathbf{W} = \mathbf{I}$, it can be inserted into (15) and group terms:

$$\underbrace{\hat{\mathbf{s}}_i^T \mathbf{J}^{-1} \mathbf{W}^{-1}}_{\mathbf{k}_{f,i}^T} \mathbf{W} \begin{pmatrix} v \\ \omega \end{pmatrix}_{ee} \leq \hat{\mathbf{s}}_i^T \dot{\mathbf{q}}_{ul} \tag{16}$$

where

$$\mathbf{k}_{f,i}^T = \hat{\mathbf{s}}_i^T \mathbf{J}^{-1} \mathbf{W}^{-1}. \tag{17}$$

Dividing through by the magnitude of (17) produces:

$$\underbrace{\frac{\mathbf{k}_{ui}^T}{\|\mathbf{k}_{ui}\|}}_{\hat{\mathbf{a}}_{ui}^T} \underbrace{\mathbf{W}\begin{pmatrix} v \\ \omega \end{pmatrix}_{ee}}_{\mathbf{x}} \le \underbrace{\frac{\hat{\mathbf{s}}_i^T}{\|\mathbf{k}_{ui}\|}\dot{\mathbf{q}}_{ui}}_{b_{ui}}. \tag{18}$$

Thus the upper joint limit on joint i reduces to

$$\hat{\mathbf{a}}_{ui}^T\mathbf{x} \le b_{ui}. \tag{19}$$

The constraint functions $f_i(x)$ corresponding to the left portion of (11) (i.e. the lower limits) may then be created similarly:

$$-\hat{\mathbf{s}}_i^T\dot{\mathbf{q}} \le -\hat{\mathbf{s}}_i^T\dot{\mathbf{q}}_{li} \tag{20}$$

$$-\mathbf{k}_{li}^T\mathbf{W}\begin{pmatrix} v \\ \omega \end{pmatrix}_{ee} \le -\hat{\mathbf{s}}_i^T\dot{\mathbf{q}}_{li} \tag{21}$$

$$\underbrace{-\frac{\mathbf{k}_{li}^T}{\|\mathbf{k}_{li}\|}}_{\hat{\mathbf{a}}_{li}^T} \underbrace{\mathbf{W}\begin{pmatrix} v \\ \omega \end{pmatrix}_{ee}}_{\mathbf{x}} \le \underbrace{-\frac{\hat{\mathbf{s}}_i^T}{\|\mathbf{k}_{li}\|}\dot{\mathbf{q}}_{li}}_{b_{li}} \tag{22}$$

$$\hat{\mathbf{a}}_{li}^T\mathbf{x} \le b_{li}. \tag{23}$$

Combining coefficients of (19) and (24) for all n joints produces

$$\mathbf{A}_{ul} = \begin{bmatrix} \hat{\mathbf{a}}_{ul_1}^T \\ \vdots \\ \hat{\mathbf{a}}_{ul_n}^T \end{bmatrix} \qquad \mathbf{b}_{ul} = \begin{bmatrix} b_{ul_1} \\ \vdots \\ b_{ul_n} \end{bmatrix} \tag{24}$$

$$\mathbf{A}_{ll} = \begin{bmatrix} \hat{\mathbf{a}}_{ll_1}^T \\ \vdots \\ \hat{\mathbf{a}}_{ll_n}^T \end{bmatrix} \qquad \mathbf{b}_{ll} = \begin{bmatrix} b_{ll_1} \\ \vdots \\ b_{ll_n} \end{bmatrix} \tag{25}$$

Then the set of $2n$ constraints due to joint limits simplifies to

$$\mathbf{A}_l\mathbf{x} \le \mathbf{b}_l \tag{26}$$

where

$$\mathbf{A}_l = \begin{bmatrix} \mathbf{A}_{ul} \\ \mathbf{A}_{ll} \end{bmatrix} \qquad \mathbf{b}_l = \begin{bmatrix} \mathbf{b}_{ul} \\ \mathbf{b}_{ll} \end{bmatrix}.$$

As described above, the collision and distance check generates pairs of potential collision points: the collision point $\mathbf{cp}_i$ and the interfering point $\mathbf{ip}_i$. For each potential collision, it desirable to constrain the velocity of cp_i toward ip_i. Graphically, this is illustrated in FIG. 9. In Fig. 9a, a robotic manipulator and an object near it are represented by their respective geometric primitives (line segments in this case). In Figs. 9b and 9c there are dashed lines between the two pairs of possible collision points. Fig. 9b shows the potential collision between an external object and the robotic manipulator, and Fig. 9c shows the potential collision between the robotic manipulator and itself. In each case a unit vector $\hat{e}_i$ in the collision direction (directed from cp_i towards ip_i) is constructed. In addition, the velocity of cp_i is denoted as $\mathbf{v}_{cp_i}$ and the velocity of ip_i is denoted as $\mathbf{v}_{ip_i}$.

The limit on the motion of cp_i towards ip_i may be restated as

$$\hat{e}_i^T \mathbf{v}_{cp_i} \leq v_{ai} \tag{27}$$

where v_{ai} is the greatest allowed "approach velocity" of cp_i towards ip_i. The manner in which v_{ai} is specified can be varied depending on the desired collision avoidance behavior. In the described embodiments, multiple shells (reaction, equilibrium, and safety) have been constructed for each body in order to create a smoothed approach velocity v_{ai}, which is expressed as:

$$v_{ai} = \hat{e}_i^T \mathbf{v}_{ip_i} + \frac{v_{halfi}}{\ln(0.5)} \ln\left(\frac{r_{ri} - d_{pi}}{r_{ri} - r_{ei}}\right) \tag{28}$$

where d_{pi} is the distance between cp_i towards ip_i, r_{ri} is the sum of the reaction radii of the two bodies, r_{ei} is the sum of the equilibrium radii of the two objects, and v_{halfi} is the greatest approach velocity allowed when d_{pi} is halfway between r_{ri} and r_{ei}. The resulting greatest allowed approach velocity is plotted in Fig. 10 as a function of the separation distance. Note that the equation is valid (and thus used) when $r_{ri} > d_i > 0$.

When the objects are separated by exactly the sum of their equilibrium radii, the avoidance velocity should equal the velocity of the interfering point along the collision direction to ensure that they do not get any closer. As the separation distance increases to the sum of the reaction radii, the avoidance velocity increases to infinity, effectively ignoring the constraint. When the objects are closer than the sum of their equilibrium radii, the avoidance velocity is lowered to force them apart. If d_{pi}

drops below r_{si} (the sum of the safety radii of the two bodies) an emergency stop of the robotic apparatus may be triggered. Note that the rule chosen in (29) can be replaced by any equivalent function.

Given v_{si} the left side of (28) should be expanded to express the velocity of cp_i in terms of the robot joint velocities. This is accomplished by treating cp_i as if it were the end-effector of the robotic apparatus and creating a "partial Jacobian matrix" ($\mathbf{J}_{\mathrm{cp}_i}$) for this point. In other words, the robot is virtually truncated at cp_i and the effects of joints between cp_i and the robot base are considered. Utilizing the Jacobian relationship for serial manipulators $\mathbf{v}_{\mathrm{cp}_i}$ is expressed as

$$\mathbf{v}_{\mathrm{cp}_i} = \mathbf{J}_{\mathrm{cp}_i}\dot{\mathbf{q}} \tag{29}$$

In the case shown in Fig. 9b where the potential collision is between the robot and another object $\mathbf{J}_{\mathrm{cp}_i}$ is

$$\mathbf{J}_{\mathrm{cp}_i} = \begin{bmatrix} \frac{(\mathbf{v}_{\mathrm{cp}_i})_x}{q_1} & \cdots & \frac{(\mathbf{v}_{\mathrm{cp}_i})_x}{q_\mu} & 0 & \cdots & 0 \\ \frac{(\mathbf{v}_{\mathrm{cp}_i})_y}{q_1} & \cdots & \frac{(\mathbf{v}_{\mathrm{cp}_i})_y}{q_\mu} & 0 & \cdots & 0 \\ \frac{(\mathbf{v}_{\mathrm{cp}_i})_z}{q_1} & \cdots & \frac{(\mathbf{v}_{\mathrm{cp}_i})_z}{q_\mu} & 0 & \cdots & 0 \end{bmatrix} \tag{30}$$

where there are μ joints between cp_i and the ground. In the case of potential collision between the robotic manipulator and itself, as is shown in Fig. 9c, the motion of the joints below ip_i does not affect the distance between cp_i and ip_i.

Thus the effects of these joints can be ignored and $\mathbf{J}_{\mathrm{cp}_i}$ can be formulated as

$$\mathbf{J}_{\mathrm{cp}_i} = \begin{bmatrix} 0 & \cdots & 0 & \frac{(\mathbf{v}_{\mathrm{cp}_i})_x}{q_\eta} & \cdots & \frac{(\mathbf{v}_{\mathrm{cp}_i})_x}{q_\mu} & 0 & \cdots & 0 \\ 0 & \cdots & 0 & \frac{(\mathbf{v}_{\mathrm{cp}_i})_y}{q_\eta} & \cdots & \frac{(\mathbf{v}_{\mathrm{cp}_i})_y}{q_\mu} & 0 & \cdots & 0 \\ 0 & \cdots & 0 & \frac{(\mathbf{v}_{\mathrm{cp}_i})_z}{q_\eta} & \cdots & \frac{(\mathbf{v}_{\mathrm{cp}_i})_z}{q_\mu} & 0 & \cdots & 0 \end{bmatrix} \tag{31}$$

where there are $\eta - 1$ joints between ip_i and the ground. Substituting (30) into (28) produces

$$\hat{\mathbf{c}}_i^T \mathbf{J}_{\mathrm{cp}_i}\dot{\mathbf{q}} \leq v_{si} \tag{32}$$

Using (14) and substituting for $\dot{q}$

$$\hat{\mathbf{c}}_i^T \mathbf{J}_{cp_i} \mathbf{J}^{-1} \begin{pmatrix} v \\ \omega \end{pmatrix}_{ee} \leq v_{ai} \qquad (33)$$

and using the same approach as (16), $\mathbf{W}^{-1}\mathbf{W}$ is inserted and terms are grouped:

$$\underbrace{\hat{\mathbf{c}}_i^T \mathbf{J}_{cp_i} \mathbf{J}^{-1} \mathbf{W}^{-1}}_{\mathbf{k}_{ci}^T} \mathbf{W} \begin{pmatrix} v \\ \omega \end{pmatrix}_{ee} \leq v_{ai} \qquad (34)$$

where

$$\mathbf{k}_{ci}^T = \hat{\mathbf{c}}_i^T \mathbf{J}_{cp_i} \mathbf{J}^{-1} \mathbf{W}^{-1} \qquad (35)$$

Eqn. (35) is thus reduced to

$$\mathbf{k}_{ci}^T \mathbf{W} \begin{pmatrix} v \\ \omega \end{pmatrix}_{ee} \leq v_{ai} \qquad (36)$$

Dividing through by the magnitude of (36) produces:

$$\underbrace{\frac{\mathbf{k}_{ci}^T}{\|\mathbf{k}_{ci}^T\|}}_{\hat{\mathbf{a}}_{ci}^T} \underbrace{\mathbf{W} \begin{pmatrix} v \\ \omega \end{pmatrix}_{ee}}_{\mathbf{x}} \leq \underbrace{\frac{v_{ai}}{\|\mathbf{k}_{ci}^T\|}}_{b_{ci}} \qquad (37)$$

Thus the limits due to the i^{th} potential collision (28) reduce to

$$\hat{\mathbf{a}}_{ci}^T \mathbf{x} \leq b_{ci} \qquad (38)$$

Combining the coefficients of (39) for m potential collisions produces

$$\mathbf{A}_c = \begin{bmatrix} \hat{\mathbf{a}}_{c1}^T \\ \vdots \\ \hat{\mathbf{a}}_{cm}^T \end{bmatrix} \qquad \mathbf{b}_c = \begin{bmatrix} b_{c1} \\ \vdots \\ b_{cm} \end{bmatrix} \qquad (39)$$

which can be combined with (26)

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_l \\ \mathbf{A}_c \end{bmatrix} \qquad \mathbf{b} = \begin{bmatrix} \mathbf{b}_l \\ \mathbf{b}_c \end{bmatrix} \qquad (40)$$

such that the constraints on the manipulator's velocity due to joint angle limits, joint velocity limits, potential collisions with other objects, and potential collisions with itself are expressed within the single set of equations

$$\mathbf{A}\mathbf{x} \leq \mathbf{b} \qquad (41)$$

To achieve the desired form of the constraints, (42) is rearranged to

$$\mathbf{A}\mathbf{x} - \mathbf{b} \leq \mathbf{0} \qquad (42)$$

where each $f_i(\mathbf{x})$ is represented by the i^{th} row of the left side of (43). Thus the total number of constraints is $\kappa = 2n + m$.

Before continuing, the special cases that cause this method to fail shall be discussed. First, it is helpful for the robotic manipulator to be in a non-singular configuration. This is typically achieved through appropriate selection of joint limits to avoid singular configurations or using additional software to transition the robotic manipulator through the singularity and then resuming the collision avoidance algorithm. Also, in the case where the range space of $\mathbf{J}_{cp_i}$ (i.e. the set of all possible velocities of cp_i) is orthogonal to $\hat{c}_i$, then the left side of (33) equals zero regardless of the choice of $\dot{q}$. In this case the robotic manipulator cannot move cp_i away from ip_i at this instant. This scenario is rare, and can be overcome by adding additional logic that commands the robotic manipulator to move cp_i such that the orthogonality condition changes and the collision avoidance algorithm can resume.

Restating (2), the collision avoidance algorithm has been reduced to a constrained optimization of the form:

$$\begin{array}{ll} \text{minimize} & f_0(\mathbf{x}) \\ \text{subject to} & f_i(\mathbf{x}) \leq 0, \quad i = 1, \ldots, \kappa \end{array} \qquad (43)$$

The objective function $f_0(\mathbf{x})$ and constraint functions $f_1(\mathbf{x}), \ldots, f_\kappa(\mathbf{x})$ are detailed in (3) and (42), respectively. In addition, $f_0, \ldots, f_\kappa : \mathbb{R}^n \rightarrow \mathbb{R}$ are convex and twice continuously differentiable. Thus convex optimization techniques are applicable.

Convex optimization is a subject of extensive study within mathematics. Here, an interior point method of solving the convex optimization has been chosen. In particular, the logarithmic barrier method is employed. This method is well-suited because it is very fast, can handle an arbitrarily large number of inequality constraints, and the accuracy of the result (bounding the error) can be mathematically proven.

Briefly, the optimization is illustrated in Fig. 4, where each constraint $f_i(\mathbf{x})$ creates a set of velocities that are disallowed. When all n constraints are combined, the remaining set of allowable velocities is convex. The basic approach of the logarithmic barrier method is to model each inequality constraint as a logarithmic penalty function that grows to infinity as $\mathbf{x}$ approaches the $f_i(\mathbf{x}) = 0$ barrier. These penalty functions are added to the objective function $f_0(\mathbf{x})$, and an unconstrained optimization of $\mathbf{x}$ is performed via an iterative descent method (typically Newton's method). This value of $\mathbf{x}$ is then refined by scaling the magnitude of the penalty functions and again optimizing $\mathbf{x}$ via the descent method. This process is iterated to find $\mathbf{x}_{opt}$, the optimal value of $\mathbf{x}$. Then using (4) and (14) the optimal joint velocities $\dot{\mathbf{q}}_{opt}$ are

$$\dot{\mathbf{q}}_{opt} = \mathbf{J}^{-1}\mathbf{W}^{-1}\mathbf{x}_{opt} \qquad (44)$$

These joint velocities are commanded to the robotic manipulator. After they are executed for the duration of the specified time interval, the algorithm is repeated.

With respect to flowchart **30** of FIG. 7, a method of operating a robotic apparatus according to a collision avoidance scheme is now described. After the start (at Block **31**), at Block **32** a first robotic manipulator is moved along a first pre-planned actual path. At Block **33**, a second robotic manipulator is moved along a second pre-planned intended path.

At Block **34**, a distance between the second robotic manipulator and the first robotic manipulator is determined and compared to a threshold distance to determine a potential collision. At Block **35**, a decision is made. If there is no potential collision, at Block **36**, the movement of the second robotic manipulator along the second pre-planned intended path is continued. If there is a potential collision, at Block **37**, the second movement of the second robotic manipulator is deviated from the second pre-planned intended path to move in a dodge path away from, but closely following, the first pre-planned actual path. Block **38** indicates the end of the method.

Amended Claims

1. A robotic apparatus (9, 9', 9") implementing a collision avoidance scheme, comprising:
- a first robotic manipulator (10, 10', 10');
- a first controller (15, 15', 15") configured to control said first robotic manipulator for movement along a first pre-planned actual path (11a, 11b) through first drive signals;
- a second robotic manipulator (12, 12', 12"); and
- a second controller (20, 20', 20") configured to control, through second drive signals, movement of said second robotic manipulator for movement along a second pre-planned intended path (13) and deviating therefrom to move in a dodging path (14a, 14b) away from the first pre-planned actual path based upon determining a potential collision with said first robotic manipulator without prior knowledge of the first pre-planned actual path,

wherein the first and the second controllers are configured to communicate with each other through first and second communications interfaces (19', 24') so that the second controller determines a potential collision based upon the first drive signals,

wherein the second pre-planned intended path is based upon a sequence of desired velocities;

wherein said second controller is configured to move said second robotic manipulator along the dodging path based upon a sequence of dodge velocities while choosing the velocities closest to the sequence of desired velocities,

the dodge velocities being based, at least upon a plurality of convex sets of allowable velocities, a kinetic energy of the second robotic manipulator and/or a force on at least one joint of the second robotic manipulator.

2. The robotic apparatus of Claim 1, wherein said first robotic manipulator comprises at least one joint and a joint sensor cooperating with said at least one joint and said first controller for determining positioning of the at least one joint; and wherein said second controller is configured to determine the potential collision based upon the positioning of the at least one joint.

3. The robotic apparatus of Claim 1, wherein said second controller is configured to repeatedly determine a distance between said second robotic

manipulator and said first robotic manipulator and to compare the distance to at least one threshold distance to thereby determine a potential collision.

4. The robotic apparatus of Claim 1, wherein second controller is configured to repeatedly determine a distance between said second robotic manipulator and said first robotic manipulator, and to determine a potential collision with said first robotic manipulator based upon the distance; wherein said second controller is configured to store a geometric model of said first and second robotic manipulators; and wherein said second controller is configured to determine the distance between said second robotic manipulator and said first robotic manipulator based upon the geometric models of said second robotic manipulator and said first robotic manipulator.

5. The robotic apparatus of Claim 4, wherein each geometric model includes a series of buffer zones surrounding a respective robotic manipulator; and wherein said second controller is configured to determine a potential collision between said second robotic manipulator and said first robotic manipulator based upon an overlap between the respective buffer zones of said second robotic manipulator and said first robotic manipulator.

6. The robotic apparatus of Claim 5, wherein said second controller is configured to move said second robotic manipulator at different speeds based upon which respective buffer zones are overlapping.

7. The robotic apparatus of Claim 3, wherein said second controller is also configured to repeatedly determine an approach velocity between said second robotic manipulator and said first robotic manipulator; and wherein said second controller is also configured to determine the potential collision based upon the approach velocity.

8. The robotic apparatus of Claim 3, wherein said second controller is also configured to repeatedly determine an approach velocity between said second robotic manipulator and said first robotic manipulator; and wherein said second controller is also configured to determine the potential collision based upon the approach velocity.

9. The robotic apparatus of Claim 1, wherein said second robotic manipulator comprises at least one joint; and wherein said second controller is configured to determine the sequence of dodge velocities based upon a force on said at least one joint.

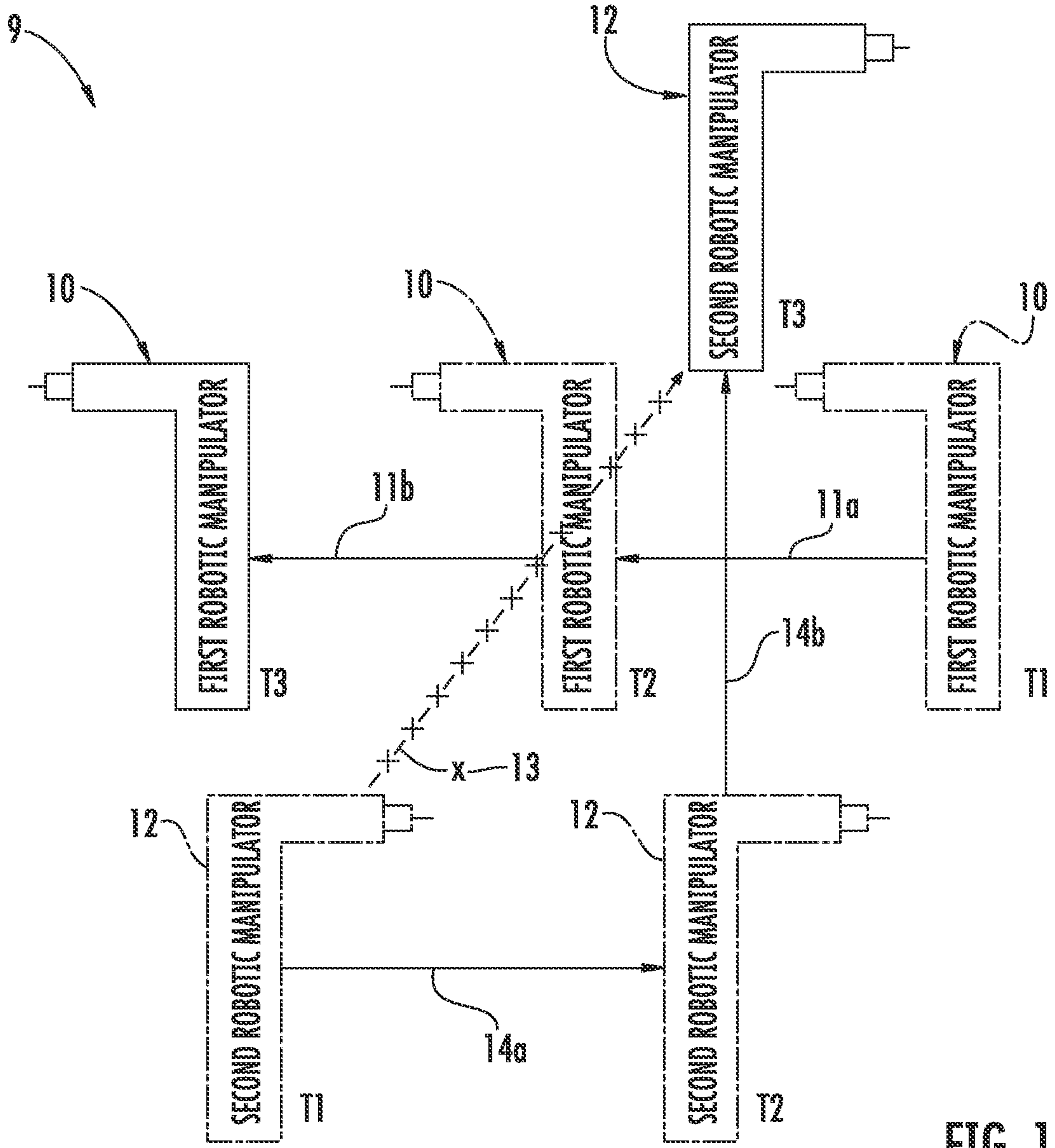

9
12
SECOND ROBOTIC MANIPULATOR
T3
10
FIRST ROBOTIC MANIPULATOR
T3
10
FIRST ROBOTIC MANIPULATOR
T2
10
FIRST ROBOTIC MANIPULATOR
T1
11b
11a
14b
x
13
12
SECOND ROBOTIC MANIPULATOR
T1
12
SECOND ROBOTIC MANIPULATOR
T2
14a

FIG. 1

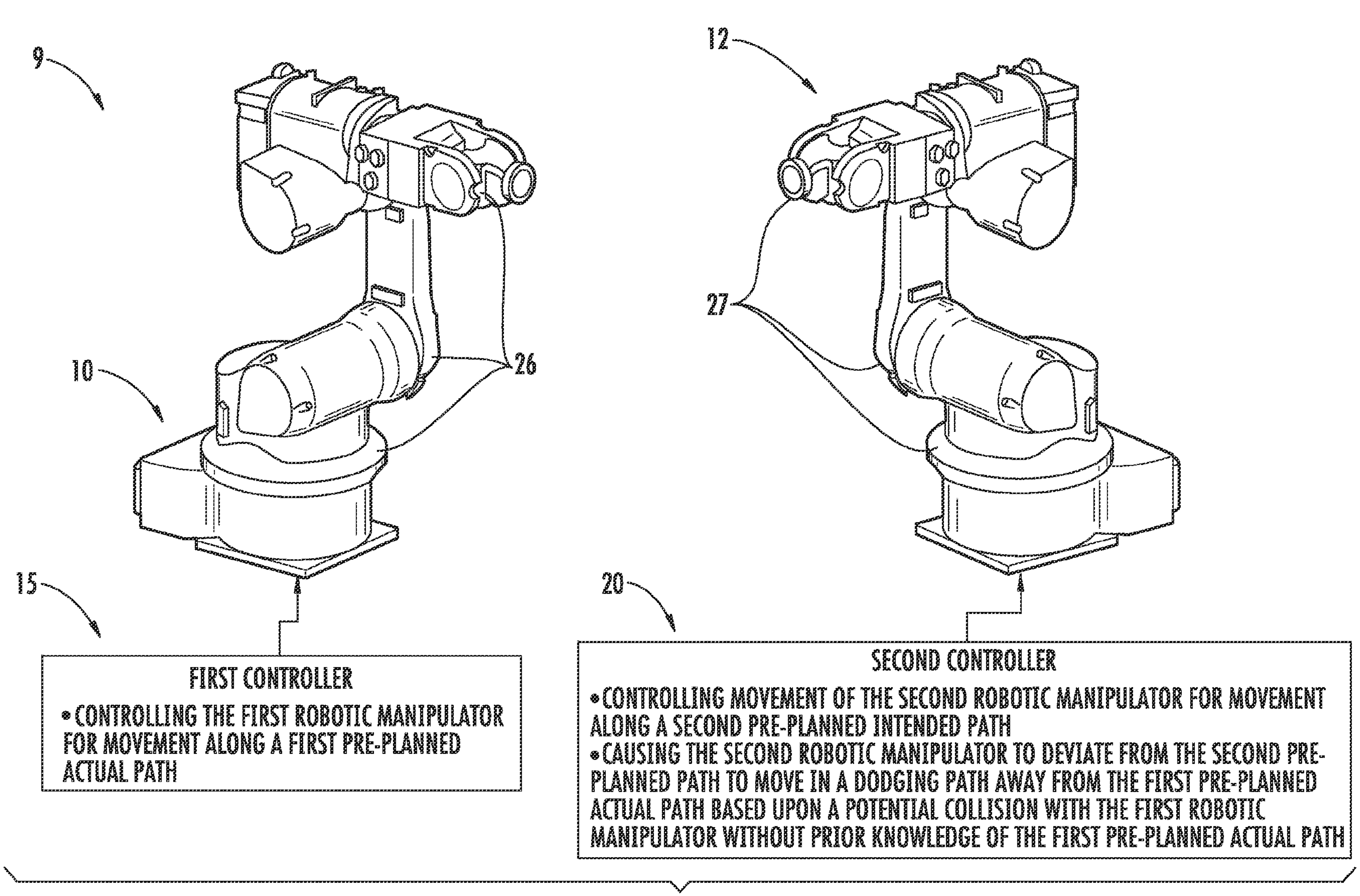

9
10
12
15
20
26
27
FIRST CONTROLLER
•CONTROLLING THE FIRST ROBOTIC MANIPULATOR FOR MOVEMENT ALONG A FIRST PRE-PLANNED ACTUAL PATH
SECOND CONTROLLER
•CONTROLLING MOVEMENT OF THE SECOND ROBOTIC MANIPULATOR FOR MOVEMENT ALONG A SECOND PRE-PLANNED INTENDED PATH
•CAUSING THE SECOND ROBOTIC MANIPULATOR TO DEVIATE FROM THE SECOND PRE-PLANNED PATH TO MOVE IN A DODGING PATH AWAY FROM THE FIRST PRE-PLANNED ACTUAL PATH BASED UPON A POTENTIAL COLLISION WITH THE FIRST ROBOTIC MANIPULATOR WITHOUT PRIOR KNOWLEDGE OF THE FIRST PRE-PLANNED ACTUAL PATH

FIG. 2

3/9

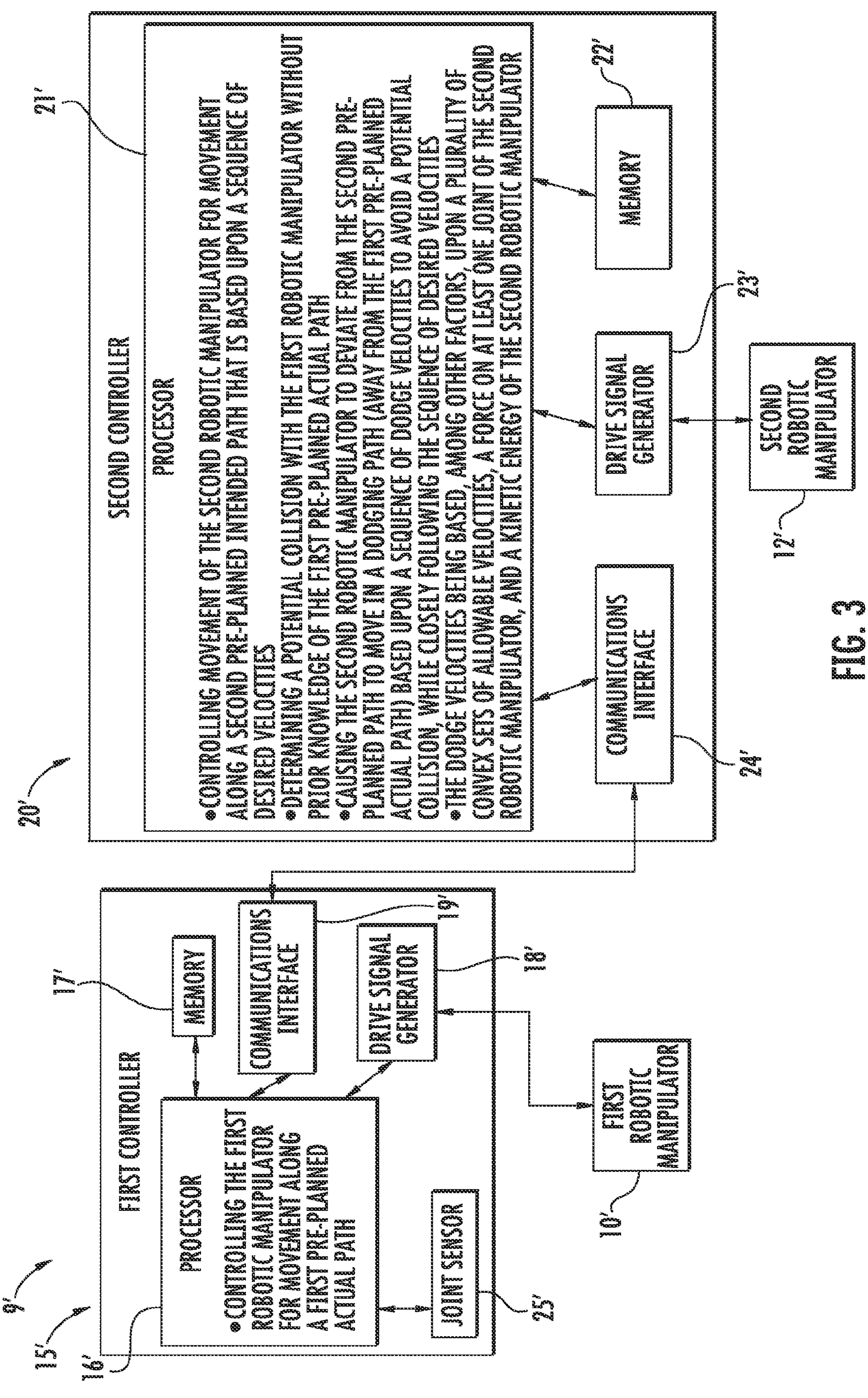

SECOND CONTROLLER
PROCESSOR
●CONTROLLING MOVEMENT OF THE SECOND ROBOTIC MANIPULATOR FOR MOVEMENT ALONG A SECOND PRE-PLANNED INTENDED PATH THAT IS BASED UPON A SEQUENCE OF DESIRED VELOCITIES
●DETERMINING A POTENTIAL COLLISION WITH THE FIRST ROBOTIC MANIPULATOR WITHOUT PRIOR KNOWLEDGE OF THE FIRST PRE-PLANNED ACTUAL PATH
●CAUSING THE SECOND ROBOTIC MANIPULATOR TO DEVIATE FROM THE SECOND PRE-PLANNED PATH TO MOVE IN A DODGING PATH (AWAY FROM THE FIRST PRE-PLANNED ACTUAL PATH) BASED UPON A SEQUENCE OF DODGE VELOCITIES TO AVOID A POTENTIAL COLLISION, WHILE CLOSELY FOLLOWING THE SEQUENCE OF DESIRED VELOCITIES
●THE DODGE VELOCITIES BEING BASED, AMONG OTHER FACTORS, UPON A PLURALITY OF CONVEX SETS OF ALLOWABLE VELOCITIES, A FORCE ON AT LEAST ONE JOINT OF THE SECOND ROBOTIC MANIPULATOR, AND A KINETIC ENERGY OF THE SECOND ROBOTIC MANIPULATOR
MEMORY
DRIVE SIGNAL GENERATOR
COMMUNICATIONS INTERFACE
SECOND ROBOTIC MANIPULATOR
20'
21'
22'
23'
24'
12'
FIRST CONTROLLER
PROCESSOR
●CONTROLLING THE FIRST ROBOTIC MANIPULATOR FOR MOVEMENT ALONG A FIRST PRE-PLANNED ACTUAL PATH
MEMORY
COMMUNICATIONS INTERFACE
DRIVE SIGNAL GENERATOR
JOINT SENSOR
FIRST ROBOTIC MANIPULATOR
9'
15'
16'
17'
18'
19'
25'
10'

FIG. 3

POTENTIAL DODGE VELOCITIES

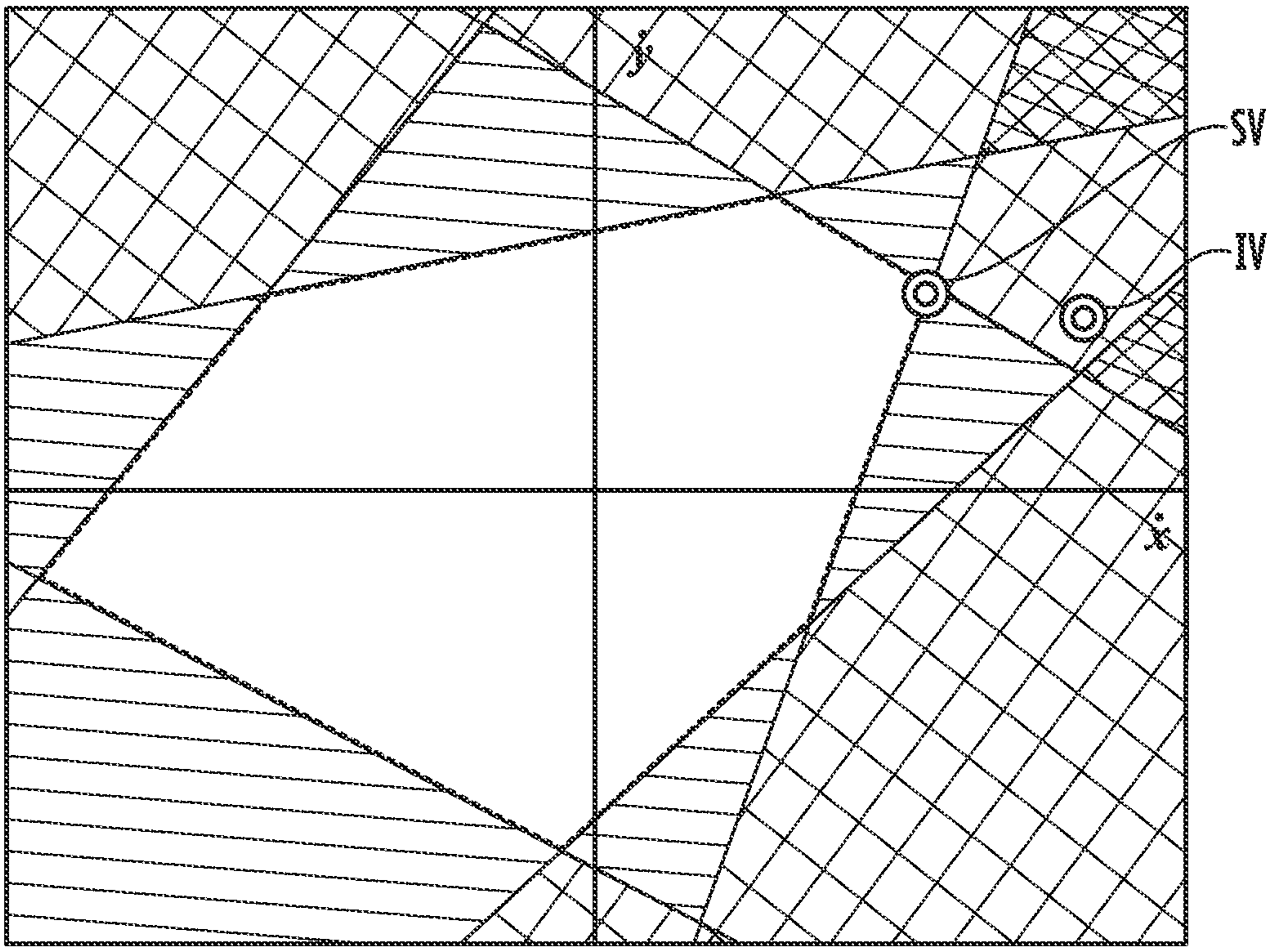


FIG. 4

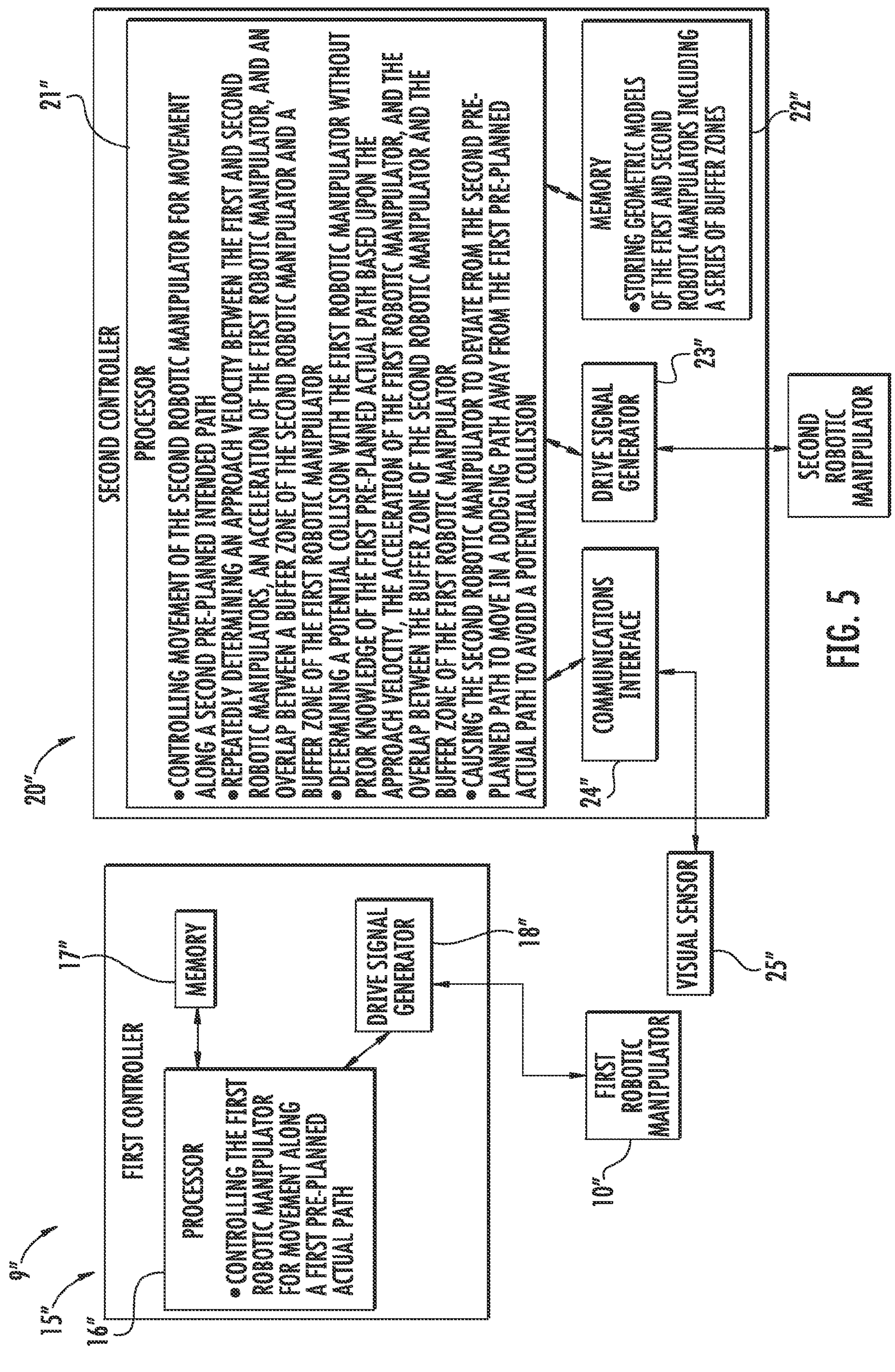

20"
21"
SECOND CONTROLLER
PROCESSOR
●CONTROLLING MOVEMENT OF THE SECOND ROBOTIC MANIPULATOR FOR MOVEMENT ALONG A SECOND PRE-PLANNED INTENDED PATH
●REPEATEDLY DETERMINING AN APPROACH VELOCITY BETWEEN THE FIRST AND SECOND ROBOTIC MANIPULATORS, AN ACCELERATION OF THE FIRST ROBOTIC MANIPULATOR, AND AN OVERLAP BETWEEN A BUFFER ZONE OF THE SECOND ROBOTIC MANIPULATOR AND A BUFFER ZONE OF THE FIRST ROBOTIC MANIPULATOR
●DETERMINING A POTENTIAL COLLISION WITH THE FIRST ROBOTIC MANIPULATOR WITHOUT PRIOR KNOWLEDGE OF THE FIRST PRE-PLANNED ACTUAL PATH BASED UPON THE APPROACH VELOCITY, THE ACCELERATION OF THE FIRST ROBOTIC MANIPULATOR, AND THE OVERLAP BETWEEN THE BUFFER ZONE OF THE SECOND ROBOTIC MANIPULATOR AND THE BUFFER ZONE OF THE FIRST ROBOTIC MANIPULATOR
●CAUSING THE SECOND ROBOTIC MANIPULATOR TO DEVIATE FROM THE SECOND PRE-PLANNED PATH TO MOVE IN A DODGING PATH AWAY FROM THE FIRST PRE-PLANNED ACTUAL PATH TO AVOID A POTENTIAL COLLISION
MEMORY
●STORING GEOMETRIC MODELS OF THE FIRST AND SECOND ROBOTIC MANIPULATORS INCLUDING A SERIES OF BUFFER ZONES
22"
DRIVE SIGNAL GENERATOR
23"
COMMUNICATIONS INTERFACE
24"
SECOND ROBOTIC MANIPULATOR
VISUAL SENSOR
25"
9"
15"
16"
FIRST CONTROLLER
PROCESSOR
●CONTROLLING THE FIRST ROBOTIC MANIPULATOR FOR MOVEMENT ALONG A FIRST PRE-PLANNED ACTUAL PATH
17"
MEMORY
DRIVE SIGNAL GENERATOR
18"
FIRST ROBOTIC MANIPULATOR
10"

FIG. 5

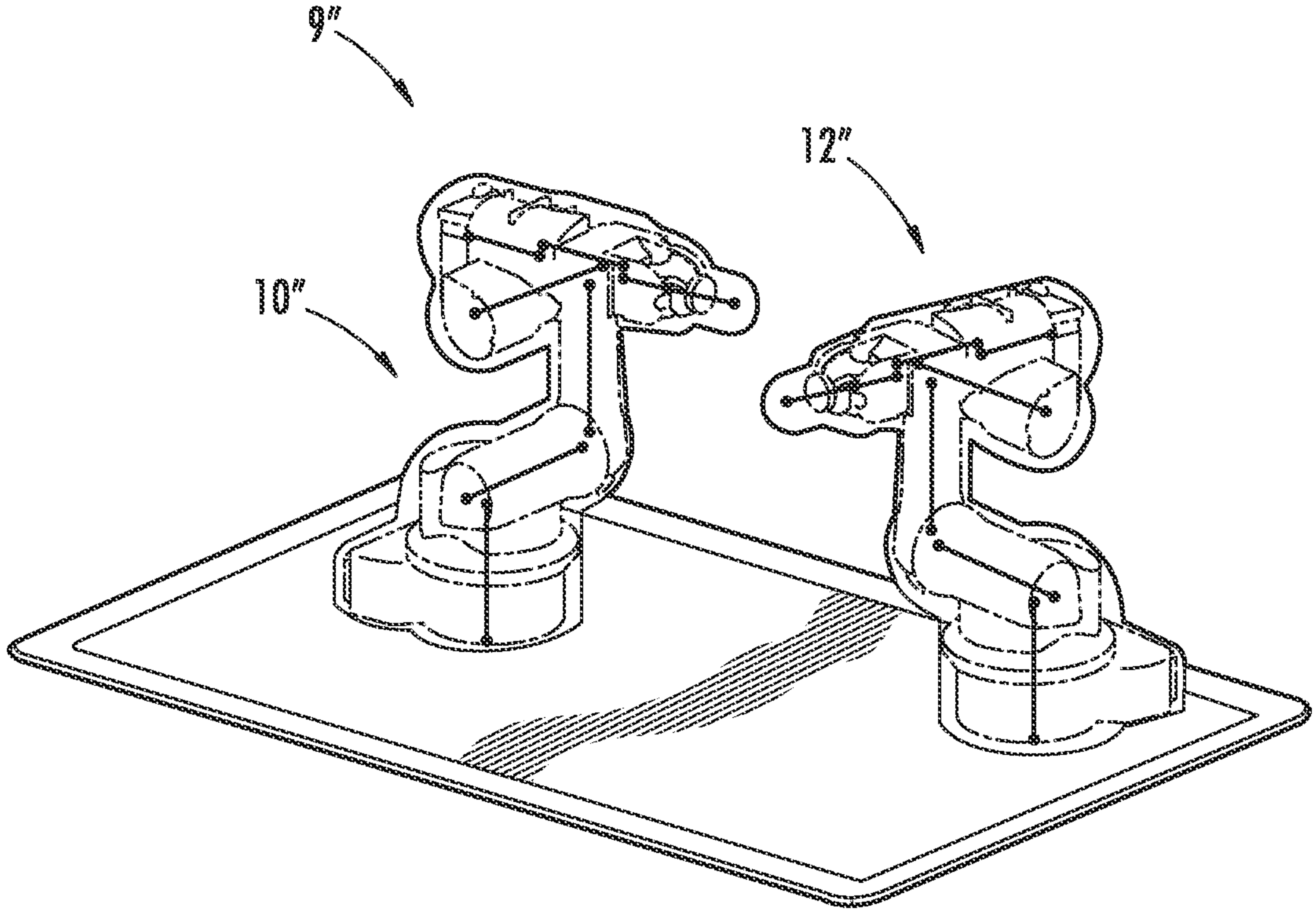


FIG. 6

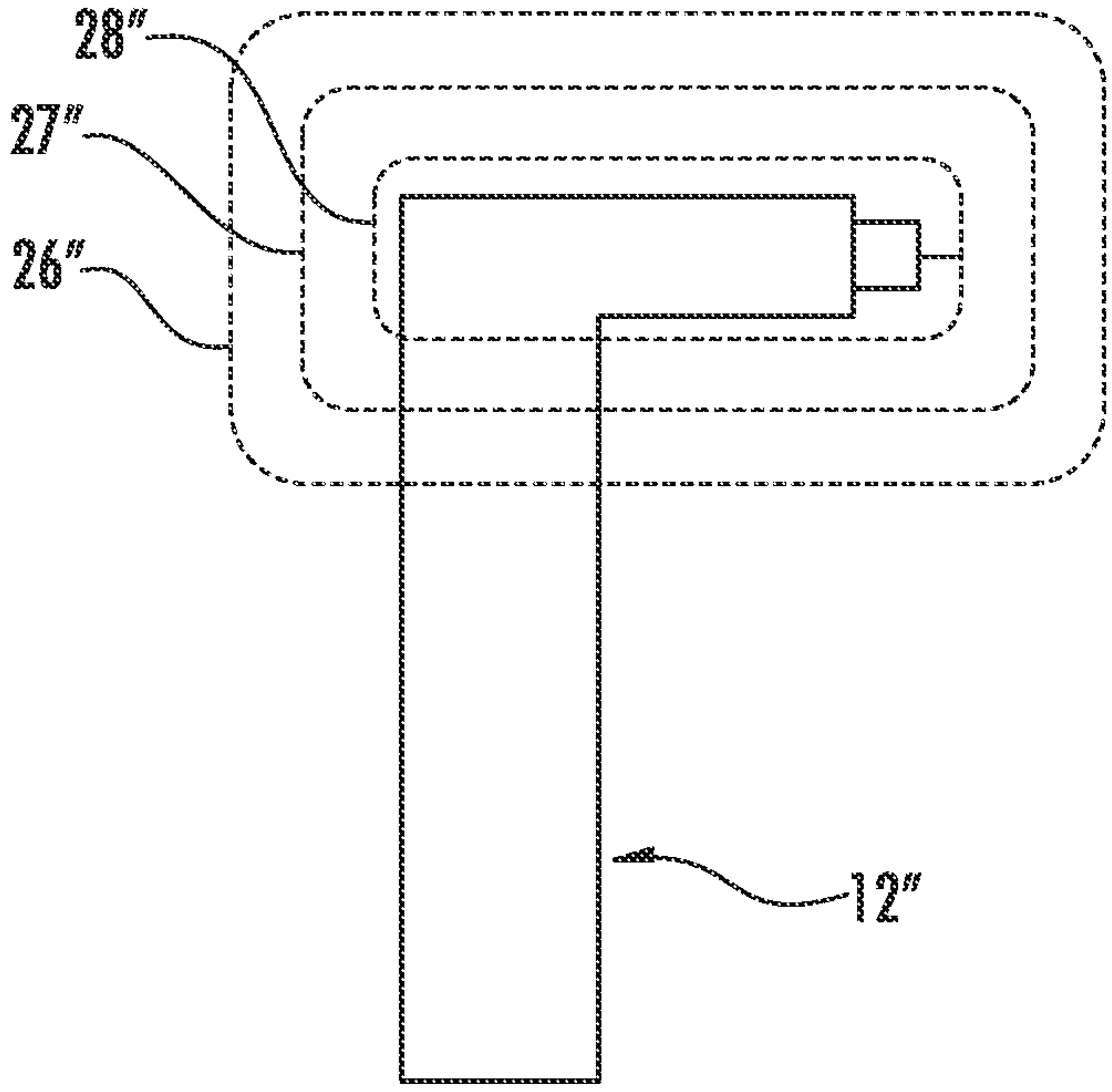


FIG. 7

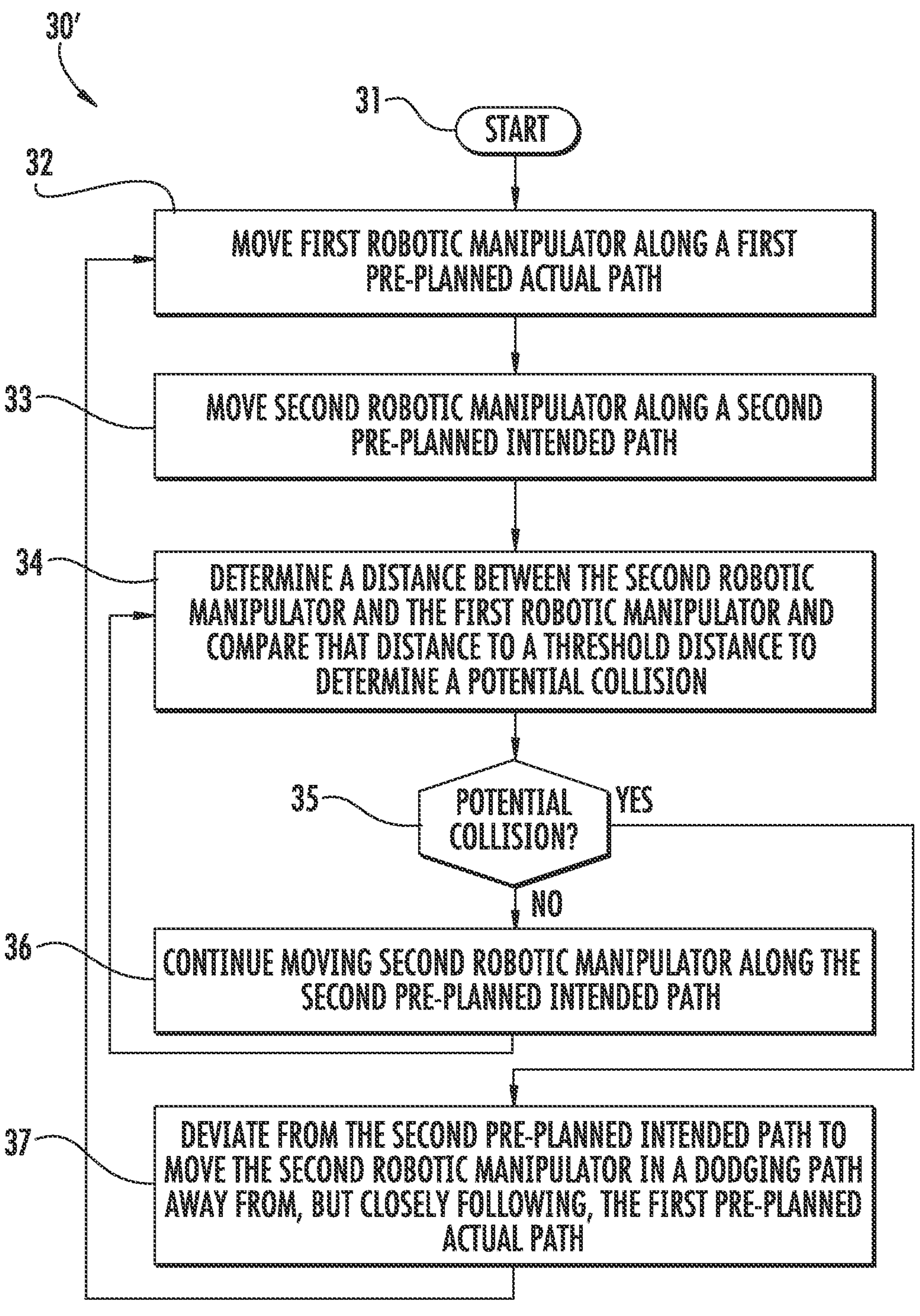

30'
31
START
32
MOVE FIRST ROBOTIC MANIPULATOR ALONG A FIRST PRE-PLANNED ACTUAL PATH
33
MOVE SECOND ROBOTIC MANIPULATOR ALONG A SECOND PRE-PLANNED INTENDED PATH
34
DETERMINE A DISTANCE BETWEEN THE SECOND ROBOTIC MANIPULATOR AND THE FIRST ROBOTIC MANIPULATOR AND COMPARE THAT DISTANCE TO A THRESHOLD DISTANCE TO DETERMINE A POTENTIAL COLLISION
35
POTENTIAL COLLISION?
YES
NO
36
CONTINUE MOVING SECOND ROBOTIC MANIPULATOR ALONG THE SECOND PRE-PLANNED INTENDED PATH
37
DEVIATE FROM THE SECOND PRE-PLANNED INTENDED PATH TO MOVE THE SECOND ROBOTIC MANIPULATOR IN A DODGING PATH AWAY FROM, BUT CLOSELY FOLLOWING, THE FIRST PRE-PLANNED ACTUAL PATH

FIG. 8

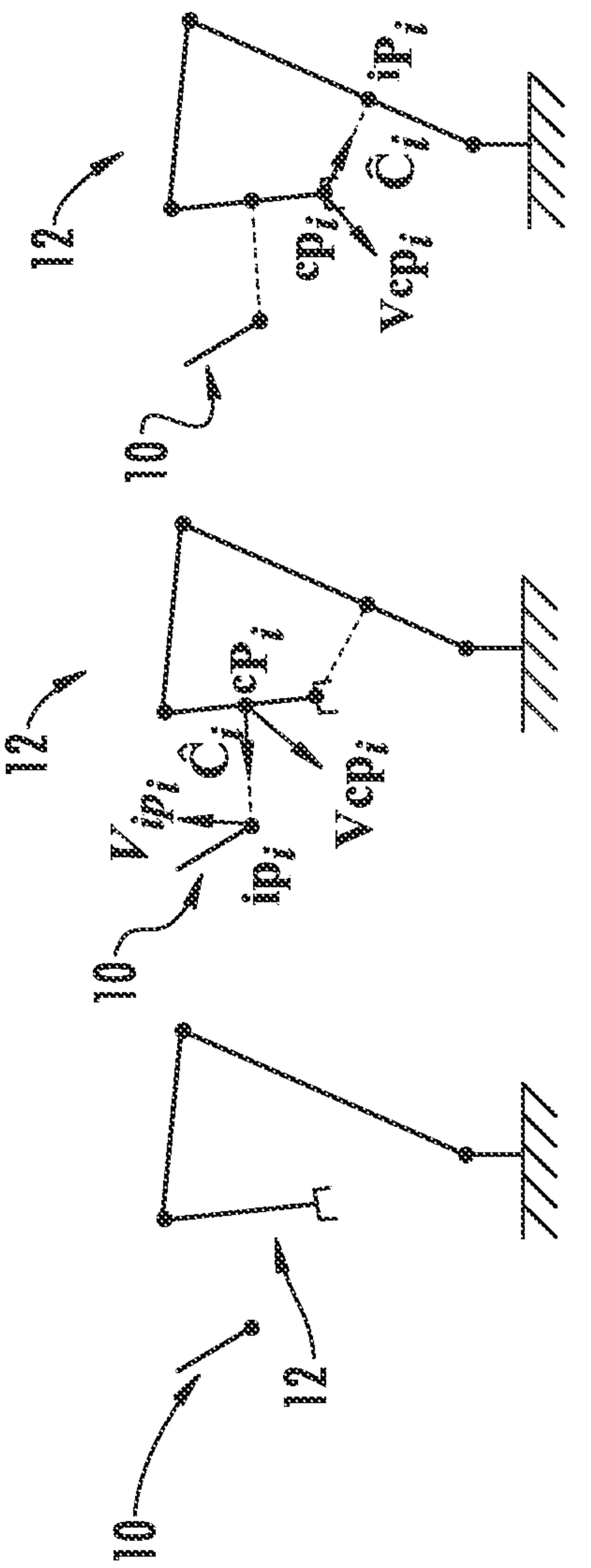


FIG. 9A

FIG. 9B

FIG. 9C

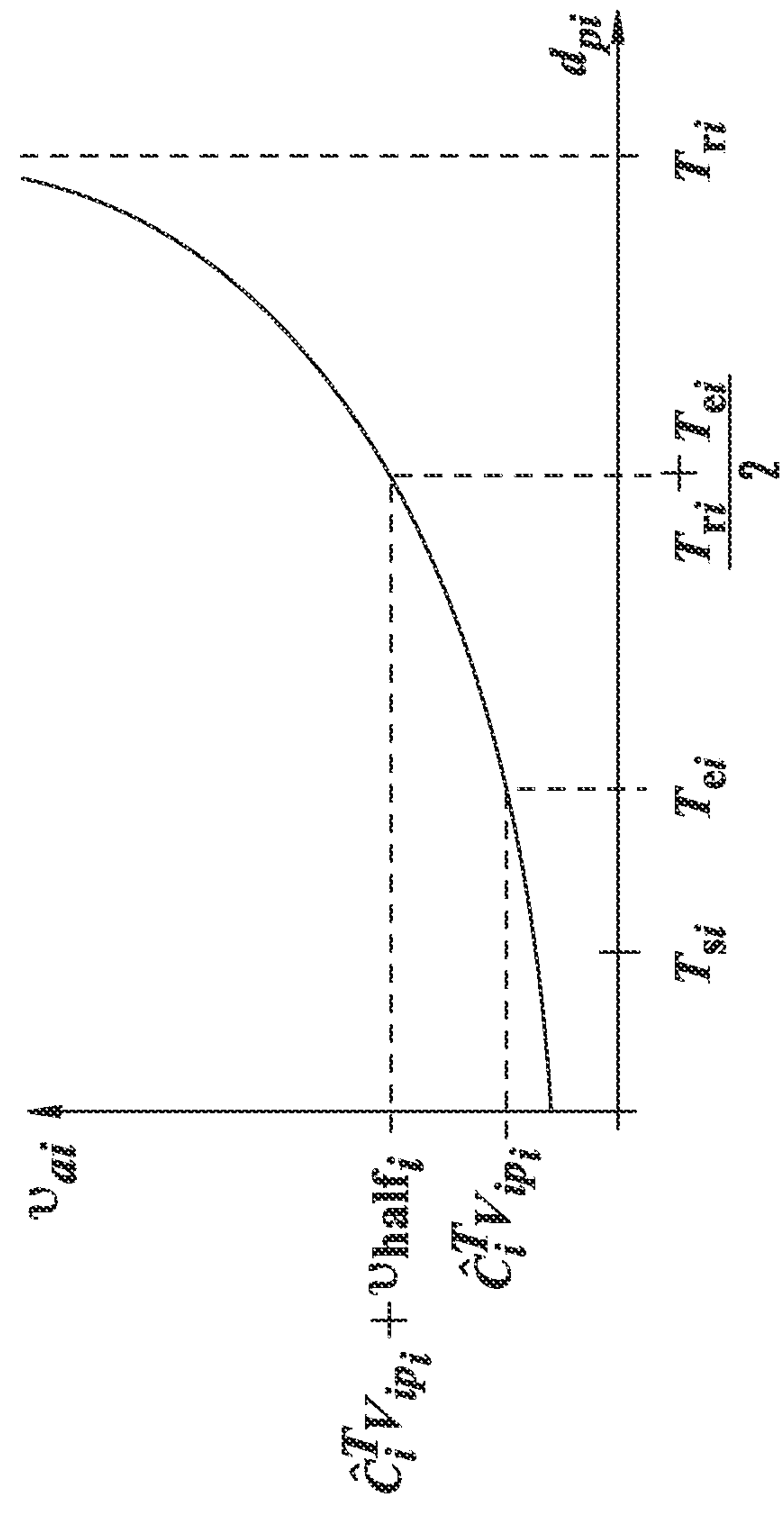
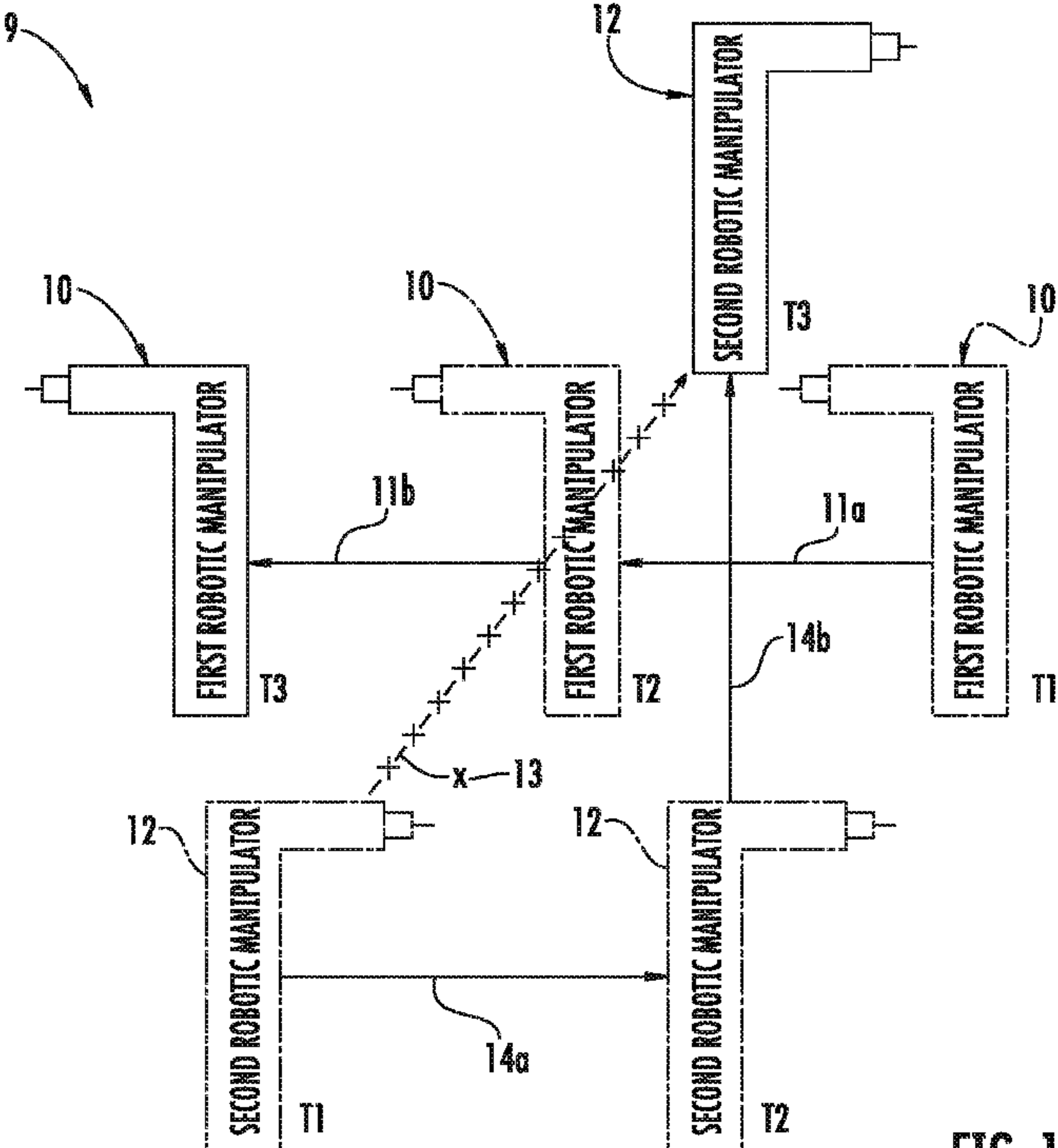


FIG. 10



9
12
SECOND ROBOTIC MANIPULATOR
T3
10
FIRST ROBOTIC MANIPULATOR
T3
10
FIRST ROBOTIC MANIPULATOR
T2
10
FIRST ROBOTIC MANIPULATOR
T1
11b
11a
14b
x
13
12
SECOND ROBOTIC MANIPULATOR
T1
12
SECOND ROBOTIC MANIPULATOR
T2
14a

FIG. 1