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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
2 June 2016 (02.06.2016)

WIPO | PCT

(10) International Publication Number
WO 2016/083986 A1

(51) International Patent Classification:
H04L 29/06 (2006.01) *H04L 29/08* (2006.01)

(21) International Application Number:
PCT/IB2015/059060

(22) International Filing Date:
24 November 2015 (24.11.2015)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
5995/CHE/2014 30 November 2014 (30.11.2014) IN

(71) Applicant: **ABB TECHNOLOGY LTD.** [CH/CH]; Affolternstrasse 44, CH-8050 Zurich (CH).

(72) Inventor: **GNANADHAS, Jonathan**; P.O. Box- 15, 417 Ashwini Layout, Karnataka, Gundlupet 571111 (IN).

(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, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, 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, 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, RW, 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, 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).

Declarations under Rule 4.17:

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

Published:

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

(54) Title: METHOD FOR ENSURING EXECUTION OF A PLANT PROCESS IN A DISTRIBUTED CONTROL SYSTEM

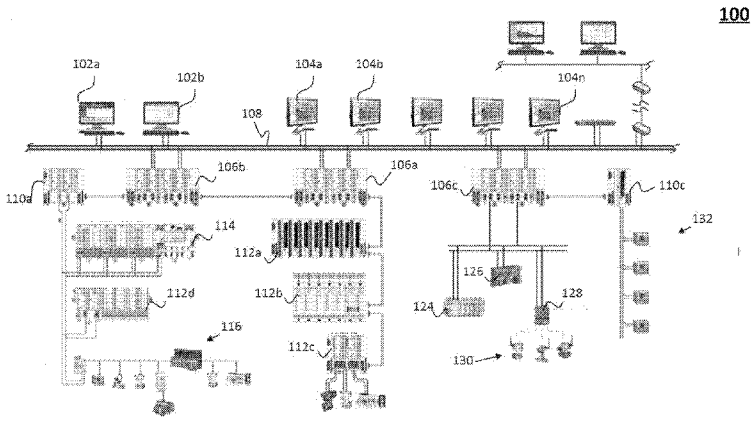

Figure 1

(57) Abstract: A method and a device for ensuring execution of a plant process in a distributed control system during a network attack are disclosed. The networked plant includes an engineering workstation, an operations workplace, and a plurality of input/output (IO) points configured to communicatively couple with a controller. Based on the traffic of the network at the controller, the priority of the communication receive tasks are modified. Further, anomalous condition is determined based on the actual and estimated schedule time of a low priority consumer task and a corrective action is taken.

WO 2016/083986 A1

METHOD FOR ENSURING EXECUTION OF A PLANT PROCESS IN A DISTRIBUTED CONTROL SYSTEM

TECHNICAL FIELD

[001] This present disclosure relates to distributed control systems in a power plant, and more specifically, to a method and a controller device for ensuring execution of a plant process in the distributed control system during a network attack.

BACKGROUND

[002] The importance of distributed control systems has been accepted across the world in various industries. Typically, a plurality of controlling devices connected over an Ethernet network facilitates operations of these systems. However, an attack may occur over the network due to reasons including but not limited to cyber-attack such as Denial of Service (DOS) attack, exceptional plant events such as a trip which can result in a massive amount of packets generated on the network to communicate alarms, unmanaged traffic due to operations like a file transfer, unexpected or non-compliant behaviour of a node in the network and the like.

[003] The network attack can substantially affect the normal functioning of the distributed control systems. For example, an increase in a rate of communication packets to the controlling device may hamper the ability of the controlling device to provide service to other tasks, such as the serial interface tasks, protocol stacks, and the control logics. Further, the controlling device will not be able to efficiently transmit packets to other nodes because the software which can produce these packets are not scheduled or given sufficient execution time due to continuously reception of the communication packets.

[004] Various prior art methods such as rate limiting, packet filtering, Ethernet receive disable, Ethernet receive poll mode and other preventive methods exist to control the network attack. Generally, these methods are employed so that the controlling device does not transit into an unknown or error state during the network attack. However, these methods do not explicitly guarantee a minimal critical data loss and minimal output latency time during the network attack. For example, when the controller device is configured to receive process signals via International Electrotechnical Commission (IEC) 61850 Manufacturing Message Specification (MMS) protocol and the events via IEC 61850 Generic Object Oriented Substation Event (GOOSE), the controller device does not guarantee the consumption of a

packet indicating a trip or an event during a network anomaly due to the limitations in the existing methods for handling the network attack. As a result, performance and efficiency of the distributed control system is severely affected.

[005] Therefore, there exists a need for an efficient method for ensuring functioning of the distributed control systems during the network attack.

SUMMARY

[006] Embodiments of the present disclosure disclose a method and a controlling device for ensuring execution of a plant process in a distributed control system during a network attack. The system comprises a controller configured to execute: at least one communication task comprising instructions for the controller to communicate with at least one field device via an input/output (I/O) module directly or indirectly via a gateway/interface unit; and at least one consumer task comprising instructions corresponding to at least one process application.

[007] In an embodiment, the method includes defining a first state and a second state corresponding to the at least one communication task of the controller. The at least one communication task has a priority greater than the at least one consumer task in the first state and the at least one communication task has a priority smaller than the at least one consumer task in the second state. The method further includes analyzing traffic of a network received at the controller, wherein the traffic comprises a plurality of packets processed during execution of the at least one communication task and the consumer task; processing a scheduling logic configured to change state for the at least one communication task from the first state to the second state on occurrence of a predefined condition corresponding to the traffic of the network; and scheduling an execution of the at least one communication task at the controller after scheduling an execution of the at least one consumer task.

[008] Additional aspects, advantages, features and objects of the present disclosure would be made apparent from the drawings and the detailed description of the illustrative embodiments construed in conjunction with the appended claims that follow.

BRIEF DESCRIPTION OF THE DRAWINGS

[009] The summary above, as well as the following detailed description of illustrative embodiments, is better understood when read in conjunction with the appended drawings. For

the purpose of illustrating the present disclosure, exemplary constructions of the disclosure are shown in the drawings. However, the present disclosure is not limited to specific methods and instrumentalities disclosed herein. Moreover, those skilled in the art will understand that the drawings are not to scale. Wherever possible, like elements have been indicated by identical numbers.

[0010] Embodiments of the present disclosure will now be described, by way of example only, with reference to the following diagrams wherein:

[0011] Figure 1 illustrates an example block diagram of a distributed control system in a plant in accordance with an embodiment of the disclosure;

[0012] Figure 2 illustrates an example block diagram of a controller configured to ensure proper execution of a plant process in the distributed control system during a network attack in accordance with an embodiment of the disclosure;

[0013] Figure 3 illustrates exemplary states for the communication task so that the controller responds appropriately during an anomalous traffic condition in accordance with an embodiment of the disclosure;

[0014] Figure 4 discloses exemplary steps of a method for changing states of the communication task in accordance with traffic condition of the network in accordance with an embodiment of the disclosure;

[0015] Figure 5 illustrates exemplary timing diagrams indicating actual schedule time of a low priority consumer task in accordance with an embodiment of the disclosure; and

[0016] Figure 6 discloses exemplary steps of a method for ensuring execution of the plant process in the distributed control system during the network attack in accordance with an embodiment of the disclosure.

[0017] In the accompanying drawings, an underlined number is employed to represent an item over which the underlined number is positioned or an item to which the underlined number is adjacent. A non-underlined number relates to an item identified by a line linking the non-underlined number to the item. When a number is non-underlined and accompanied by an

associated arrow, the non-underlined number is used to identify a general item at which the arrow is pointing.

DETAILED DESCRIPTION OF EMBODIMENTS

[0018] The following detailed description illustrates embodiments of the present disclosure and ways in which they can be implemented. Although some modes of carrying out the present disclosure have been disclosed, those skilled in the art would recognize that other embodiments for carrying out or practicing the present disclosure are also possible.

[0019] Figure 1 illustrates an example block diagram of a distributed control system 100 in accordance with an embodiment of the disclosure. The distributed control system 100 includes engineering workstations 102 (e.g., 102a, 102b), a plurality of operations workplaces 104 (e.g., 104a, 104b....., and 104n), and a plurality of controllers 106 (e.g., 106a, 106b and 106c) communicatively coupled to each other through a plant network 108.

[0020] In an embodiment, the controller 106 is configured to communicatively couple to a plurality of field devices directly or indirectly via gateway/interface unit 110. As illustrated in Figure. 1, the controller 106a is configured to communicatively couple to one or more field devices through one or more input/output (I/O) modules such as I/O modules 112a, Rack I/O modules 112b and Highway Addressable Remote Transducer (HART) protocol based I/O modules 112c using serial interface. The controller 106b is indirectly connected to the one or more field devices through a gateway/interface unit 110a. As shown in Figure 1, the gateway/interface unit 110a enables the controller 106b to communicate with I/O modules 112d, turbine modules 114, process field bus (PROFIBUS) based devices 116 through the serial interface. Further, the controller 106b is configured to directly communicate with the other controller 106a through the serial interface. In addition, the controller 106c is in direct communication with programmable logic controller (PLC) 124, third party devices 126, wireless gateways 128 via Ethernet network. The wireless gateways 128 enable the controller 106c to control wireless field devices 130. The controller 106c is also in communication with the various IEC 61850 based field devices 132 via a gateway/interface unit 110c.

[0021] The gateway/interface unit 110 (e.g., the 110a, and 110c) is configured to enable the controller 106 to control the operation of the field devices operating at protocols different than

that of the controller 106. The gateway/interface unit 110 is configured to respond to protocols used by these field devices as well as protocols used by the controller 106. As a result, the gateway/interface unit 110 is configured to interpret the signal received from the controller 106 for any of the field devices and vice versa. For example, for IEC 61850 based field device 132, a specific I/O data point may be converted to Generic Object Oriented Substation Events (GOOSE) Signal Sending point (GOOSE Publisher) on the gateway/interface unit 110 in case of a controller output or to a GOOSE Signal Reception Point (GOOSE Subscriber) in case of a controller input in the IEC 61850 network.

[0022] In an embodiment, the controller 106 is configured to receive one or more control logics from the engineering workstation 102 in order to ensure execution of the plant process within the distributed control system 100. For example, the control logics can include instructions to control execution of a specific plant process such as boiler control or management of load shedding application. These control logics are designed and then downloaded to the controller 106 using the engineering workstation 102.

[0023] During an execution of the plant process, the controller 106 is configured to communicate with the field devices through I/O modules 112 in-order to obtain the value of process variables used in the control logic to perform a plant control operation. These variables can correspond to parameters such as alarm, command, technical or other functional parameters required for the execution of the plant process. The controller 106 scans the I/O modules 112; determines the values of these variables and process the information in accordance with the control logic embedded therein. In an example, the controller 106 may communicate the processed information to the operations workplace 104 so that the plant operator may analyze the values and take an appropriate action. In another example, the controller 106 may automatically take an appropriate action based on the information obtained from the I/O modules 112.

[0024] During normal traffic conditions, tasks associated with a communication interface of the controller 106 have higher priorities than other tasks (e.g., consumer tasks or application tasks) so that the controller 106 can effectively communicate with the field devices. Such prioritization of the communication tasks enables the controller 106 to effectively manage the execution of the plant processes, since the controller 106 can communicate control signal, or

other information with the field devices. However, during an occurrence of the network flooding condition, communication related tasks take substantial hold of the computing resources. As a result, other tasks such as the application tasks cannot be scheduled for execution within the controller 106 and thereby, affecting the operational efficiency of the controller 106. The present disclosure discloses a method, system and a device for ensuring proper functioning of the controller 106 during the presence of the network flooding condition.

[0025] In an embodiment, the controller 106 is configured to analyse traffic (such as congestion level) of the network and based on an occurrence of a predefined condition (e.g., excessive rate of communication packets) within the analyzed traffic; the controller 106 is configured to change the state of the communication task 220 to a state wherein the communication task 220 has a lower priority than other critical tasks (e.g., the consumer tasks 216). As a result, the controller 106 is configured to ensure that the critical tasks are scheduled for the execution during an abnormal condition of the network traffic and a fair service or schedule to other functionalities of the controller 106 can be ascertained even in the presence of excessive rate of communication packets. As a result, proper functioning of the distributed control system can be achieved during the abnormal traffic condition. Further, the controller 106 is configured to analyze the traffic behaviour to indicate the type of the network attack and accordingly, take an appropriate action as discussed in later portion of the description.

[0026] Figure 2 illustrates a block diagram of a controller 106 configured to ensure proper execution of the plant process during the network attack in accordance with an embodiment of the disclosure. The controller 106 includes a central processing unit (CPU) 202, support circuits 204, a memory 206 and a traffic monitor 208.

[0027] The traffic monitor 208 is configured to include a sampler 210 for regularly sampling the network traffic and analyzing the traffic samples. The memory 206 includes a task manager 212 and a scheduler 214 configured to schedule one or more consumer tasks 216, low priority (LP) consumer task 218, and communication tasks 220 in accordance with at least one scheduling logic 222 stored therein.

[0028] In an embodiment, the controller 106 is configured to implement a priority scheduled real time operating system platform, in which each of the task or a thread is executed according to its priority. In other words, the scheduler 214 of the controller 106 ensures that a thread or

task with a lower priority is guaranteed not to be scheduled when a task of higher priority than it, is ready to be scheduled. In an embodiment, the task manager 212 is configured to define a first state of the communication task 220 wherein the priority of the communication task 220 is greater than a priority of the consumer task 216. This state can generally be referred to as normal state or State A for the communication task 220. The communication tasks 220 ensures communication of the controller 106 with the field devices via the I/O modules 112 or the gateway/interface unit 110. The communication tasks 220 manage protocols pertaining to the connection as well as receive or send packets from a physical link for the controller 106. Depending on the type of configuration of the controller 106, the communication task 220 is configured to handle both receive (Srx) and transmit (Stx) functionalities as a single task or the communication task 220 can further be split as two individual tasks that handle the receive and send functionalities separately.

[0029] In other words, if S ={f0, f1, f2, ...fn}, where f0, f1, f2.... fn are functions of the controller 106, and Srx = {fa, fb .... fm}, where Srx $\in$ S and fa, fb .... fm are functions corresponding to communication receive tasks of the controller 106, the task manager 212 ensures that the tasks corresponding to the communication receive functions are executed during the normal state or State A. This will ensure the proper communication between the controller 106 and the field devices. The operation of the controller 106 will be further explained in detail with reference to state machine diagram as illustrated in Figure 3.

[0030] Figure 3 illustrates different states for the communication task 220 so as to ensure that the controller 106 responds appropriately during the anomalous traffic conditions in the network attack. In an embodiment, state machine logic (the scheduling logic 222) is embedded in the communication task 220, the LP consumer task 218, as well as in the sampler 210 which can reliably sample the number of communication packets coming from the external input (e.g., network 108) over a period of time. The scheduling logic 222 is configured to trigger the state machine transit from the state A to state B for the communication task 220, if an unexpected rate of packets coming into the controller 106 from the external link is determined.

[0031] The state A is referred to as the normal state wherein the scheduler 214 ensures execution of the functions Srx of the communication task 220. The state B indicates to an anomaly condition in the traffic wherein the priority of the communication task 220 is changed

or inverted. The state B may also be referred to as a load analysis (LA) state wherein execution of the Srx of the communication tasks 220 are halted. The scheduler 214 does not allow execution of Srx of the communication task 220 until all the tasks that are of higher priority to the communication task 220 are executed.

[0032] In an embodiment, the scheduler 214 is configured to achieve state B of the communication task 220 by deferring the execution of the communication task 220 through priority inversion. In the priority inversion, the receiving functionalities of the communication task 220 are forced to wait till completion of the consumer task 216 or application tasks. In another embodiment, the scheduler 214 is configured to achieve the state B of the communication task 220 by dynamically changing the priority of the communication task 220 to a priority lower than the consumer task 216. In a yet another embodiment, the scheduler 214 is configured to prevent the execution of the receive functionalities of the communication task 220 by not scheduling it on the individual task ready queue (if the OS supports such an architecture)

[0033] In State B, the various checks on the traffic is done, such as the time taken for the application tasks to consume the data received, i.e., the time taken from the pre-emption of the Communication Task 220 till the time the consumer task 218 has executed. An anomaly check logic in the application tasks is executed to detect the kind of traffic. If the kind of traffic cannot be identified, the state machine transits to State C or if the traffic can be identified as attack or highly unsuitable for the device, then the state machine transits to State D. From State C, the state machine transits back to State A after a predetermined timeout period. During State C, the communication task 220 is still priority inverted or runs at a lower priority than the consumer task 218.

[0034] In State D, the communication task 220 is priority inverted or runs at a priority less than the consumer task 218. The State D to State A transition happens only when the sampling block identifies that the normal rate (less than or equal to the expected number of packets per unit time) of packets are received from the external link. In State B, the communication task 218 is priority inverted or at a priority that is less than the remaining consumer tasks. In State C and State D, only the critical traffic is consumed. Also in State C and State D, the consumer task 218 is modelled to be at a priority below the priorities of the remaining consumer tasks as well

as application tasks that require to be scheduled ahead of it, which may involve zero or more application tasks.

[0035] Further, the scheduler 214 is configured to identify the execution order and scheduling of the consumer tasks. The scheduler 214 identifies a task, which is of a priority that is below the least priority task among the consumer tasks or priority lesser than the consumer tasks, as the LP (Low Priority) consumer task 218 and schedules the execution of the LP consumer task 218.

[0036] In addition, the traffic monitor 208 is configured to perform traffic analysis during the state B of the communication task 220. The traffic analysis includes but not limited to time taken for the application and consumer tasks to consume the packets and time for the transition of the state A to state B, i.e. the time of pre-emption of communication Task 220 to the time the LP consumer task 218 has executed. Thus the time taken for the state change of the receive functionality of the communication task 220 from the state A to the state B till the completion of the state B indicates CPU utilization time for processing the anomalous network traffic by the application tasks. As a result, the controller 106 can detect the type of the network attack by: monitoring the time for the application tasks and the consumer tasks to consume the data, number of data packets processed by the application and received data.

[0037] In an embodiment, the controller 106 is configured to change the state of the communication task 220 from the State B to State C or from State B to State D depending on the type of the traffic as determined during the State B also referred to as Load Analysis phase. The States C or D may also be referred to as Flood Handle (FH) states.

[0038] If the type of the network traffic cannot be identified, the state of the communication task 220 is transitioned to the State C and if the traffic can be identified as a network attack or highly unsuitable for the controller 106, the State of the communication task 220 is transitioned to the State D. The transition from the State D to State A happens only when the sampler 210 identifies that the normal rate (expected number of packets per unit time) of packets are received from the external link. However, in the States, B, C and D, the Srx of communication task 220 is priority inverted or runs at a lower priority than the LP consumer task 218.

[0039] Further, the controller 106 is configured to handle the network flood condition by setting the priority of the communication task 220 less than the LP consumer task 218, or priority inverted on the LP Consumer Task 218. During the flood handle states, critical data packets are copied to internal buffers and non critical data packets are not consumed more than a certain threshold. For example, the controller 106 is configured to restrict the occupation of a maximum of 30 percent of the non-critical data packets in the receive (RX) input buffer at an instant and 100% occupation of the critical data packets.

[0040] Furthermore, application software or the consumer tasks residing within the controller 106, is configured to process the packets which were received by the receiving functionality of the communication task 220 in State A. Such processing of the data packets facilitates calculating the load at the controller 106. Considering the expected application traffic, the controller 106 can estimate an expected device load or CPU utilization for a given number of packets received on the communication link. Subsequently, the controller 106 is configured to compare an actual device load and the estimated device load to determine kind of network traffic. If the network attack has occurred at the controller 106, a deviation from the expected and the actual execution time can be observed due to packet rejection (that has occurred at TCP/IP level or application level) during data processing. Consequently, the controller 106 determines whether the attack has occurred or not.

[0041] In an embodiment, the controller 106 is configured to determine the actual and estimated device load as described below.

[0042] Consider P is the number of packets received within a unit time frame T. If Testim is the time estimated for the application software to utilize the CPU 202, then the Testim can be calculated using the equation 1.

[0043] Testim = Tos $+\sum_{n=1}^{Prx} Tap\_packet\_rxn + \sum_{n=1}^{Ptx} Tap\_packet\_txn$----- 1

[0044] Where Tos is the time overhead that an operating system consumes to schedule and to perform its kernel activities over the time to consume a packet; Tap_packet_rxn is the time to process a nth received packet by the application; Tap_packet_txn is the time to process and transmit nth packet.by the application; Prx and Ptx are the number of packets that are received and transmitted. If the design of the communication task 220 is that both the receive and

transmit is clubbed into a single task, then the Tap_packet_txn would mean a time to process nth packet to be transmitted, else if the Transmit and receive functionalities are separate, then the Tap_packet_txn would mean the time to process a packet to be transmitted.

[0045] Equation 2 defines the time taken by the application in the state LA

[0046] Tapp-sla = (Tlpts – Thptp) – Tossla -----2

[0047] Where Tapp-sla is the time consumed by the application when the communication task 220 state is the LA state; Tlpts is the time stamp of the low priority task schedule; Thptp is the time stamp taken when the communication task 220 is at high priority and pre-empts in the State A to State B.; and Tossla is the overhead due to operating device during the communication task 220 state LA.

[0048] Equation 3 relates the time stamp Ts during which the sample count of the communication receive packets are got.

[0049] Ts = $\sum_{n=1}^{n=Prx} Tcommrxn + \sum_{n=1}^{n=Ptx-y} Tcommtxn$ + Tapp-snr + Tost ----- 3

[0050] Tapp-snr is the time consumed by the application in State A of communication task 220 during the Time period T; Tcommrxn is the time taken by the communication link to consume the nth packet; Tcommtxn is the time taken by the communication link to transmit nth packet; Tost is the overhead due to operating device during the time frame T; and Ptx-y represents the number of packets transmitted during the time period Ts;

[0051] From equation 3, we arrive at equation 4

[0052] Tapp-snr = Ts – ($\sum_{n=1}^{n=Prx} Tcommrxn + \sum_{n=1}^{n=Ptx-y} Tcommtxn$ + Tapp-snr + Tost)

----- 4

[0053] Tactual = Tapp-sla + Tapp-snr ----- 5

[0054] Based on the Testim and Tactual, the deviation from the estimated time for packet consumption to the actual time for packet consumption can be determined. Consequently, traffic based on the controller 106 is determined.

[0055] The present disclosure facilitates forced scheduling of the other tasks other than the receive functionality of the communication task 220 so that the packets are analyzed and latency reaction of the controller 106 even in the anomalous traffic condition is reduced. During the network flood, it is observed that the application tasks are not treated fairly because the receive functionality of the communication task 220 continuously executes because of large packet rate. The method disclosed herein ensures that the application tasks are scheduled appropriately in order for packets to be consumed. Further, as the transmit functionality of the communication task 220 is available during the network flood, the application tasks are able to take actions or respond at a minimal latency time.

[0056] Figure 4 discloses exemplary steps of a method 400 for ensuring execution of the plant process in the distributed control system 100 during a network attack in accordance with an embodiment of a present disclosure. The method 400 initiates at step 402, and proceeds to step 404.

[0057] At step 404 the method 400 is configured to sample the number of packets received at a controller 106. For example, the sampler 210 of the Figure 2 samples the packets received at the controller 106. At step 406, a determination is made as to whether the number of packets received at the controller 106 is greater than a threshold value. If the number of the packets received is not greater than the threshold value, the method 400 proceeds to step 404.

[0058] The method 400 proceeds to step 408 when the number of packets received at the controller 106 is greater than the threshold value. At step 408, the state of the communication receive task is changed from a State A (i.e., normal state) to a State B (i.e., a load analysis state). For example, if the packet count over time t or the inter frame time between n and n-1th frame averaged over a period of time t, crosses a certain threshold, the state of communication receive task is changed from the normal state to the load analysis state.

[0059] At step 410, packets consumed by the consumer tasks or application tasks are analyzed and time to schedule a low priority (LP) consumer task 218 is measured. At step 412, a determination is made as to whether the actual scheduled time is lesser than the estimated time of schedule for the LP consumer task 218. The method 400 proceeds to step 414 if it is determined that the actual scheduled time deviates from a specific percentage of the estimated time of schedule for the LP consumer task 218. At step 414, an anomalous condition for the

network traffic is detected as the time for the schedule of the LP consumer task 218 is relatively much smaller than the estimated time of the schedule of the LP consumer task 218. This indicates that the packets being received at the controller 106 are not required by the application task or consumer task 216, which have a higher priority than the communication receive task during the state B. It could also indicate that the lower communication stacks such as TCP/IP or UDP could have discarded the anomalous packets and hence a fewer number of packets for the application or consumer tasks to process.

[0060] The method 400 proceeds to step 416 when it is determined that the actual scheduled time is not lesser than the estimated time of schedule for the LP consumer task 218. At step 416 a determination is made as to whether a number of packets processed at the communication layer are much greater than a corresponding number of packets processed at the application layer. The method 400 proceeds to step 414 if it is determined that the number of packets processed at the communication layer is significantly greater than the number of packets processed at the application layer. At step 414 anomaly condition is detected and at step 420, corresponding corrective action such as filtering the source of the packets, raising an alarm to the plant operator, conveying to the controller peers of an attack, etc., is taken.

[0061] The method 400 proceeds to step 418 if it is determined that the number of packets processed at the application layer is not significantly lesser than the number of packets received at the communication layer. At step 418 a determination is made as to whether the data inside the application packet is anomalous. The method 400 proceeds to step 430 if it is found that data in application packets in not anomalous. At step 430, the method changes the State to State C. At step 432, when a predetermined timeout happens, the method proceeds to step 426 and changes the State to State A.

[0062] From step 418, the method 400 proceeds to step 414 if the data in the application packet is found anomalous, and changes the State to State D. At step 414, the anomalous condition is detected and at step 420 a corresponding corrective action is taken.

[0063] At step 422, the method 400 is configured to maintain the State of the communication receive task at the State D). At step 424, a determination is made as to whether the packet rate is normal. The method 400 proceeds to step 422 when the packet rate is still not normal and proceeds to step 426 when the packet rate is found to be normal. At step 426 the method 400

is configured to change the State of the communication receive task from State D to State A. At step 428 the method 400 terminates.

[0064] Figure 5A and 5B illustrate exemplary timing diagrams indicating actual schedule of the LP consumer task 218 with respect to the application (consumer) task at high packet rates with normal and valid data as well as with attack data. Figure. 5A illustrates that during a high packet rate, the controller 106 takes a time t0 to schedule the LP consumer task 218 with respect to the application tasks (consumer tasks), with network traffic containing valid data. Figure 5B illustrates that the controller 106 takes a time t1 to schedule the LP consumer task 218 with respect to the application tasks (consumer tasks) during the attack on the TCP/IP layer. The time t1 is substantially smaller than the time t0.

[0065] Figure. 6 discloses exemplary steps of a method 600 for ensuring execution of a plant process in a distributed control system during a network attack. In an embodiment, the controller 106 is configured to implement the method 600. The method 600 initiates at step 602, wherein the method 600 is configured to define a first state and a second state corresponding to the at least one communication task 220 of the controller 106. The at least one communication task 220 has a priority greater than the at least one consumer task 218 in the first state and the at least one communication task 220 has a priority lesser than the at least one consumer task 218 in the second state. Further, the at least one communication task 220 includes instructions to enable communication of the controller 106 with at least one field device via an input/output (I/O) module 112 directly or indirectly via a gateway/interface unit 110. The at least one consumer task 218 includes instructions corresponding to at least one process application, which the controller 106 executes to perform a plant control action.

[0066] The method 600 proceeds to step 604 to analyse traffic of a network received at the controller 106. The traffic includes a plurality of packets processed during execution of the at least one communication task 220 and the consumer task 216.

[0067] At step 606, the method 600 executes the scheduling logic 222 configured to change state for the at least one communication task 220 from the first state to the second state on occurrence of a predefined condition corresponding to the traffic of the network.

[0068] At step 608, the method 600 schedules an execution of the at least one communication task 220 at the controller 106 after the execution of the at least one consumer task 216 in the second state.

[0069] The methods, systems and devices described herein offer several advantages. Unlike in the prior art, the present disclosure enables the controller device to respond to critical events even in the presence of the anomalous network traffic. The communication receive tasks which are generally scheduled at higher priority during any normal packet rate, are set to lower priorities on identification of an anomalous condition in the network traffic. As a result, the present disclosure ensures that the consumer task within the controller is executed and any critical event is responded by the controller within a minimum latency time. The state change of the communication receive task from high priority to low priority allows receiving of the critical task data and a minimal amount of non-critical data. This ensures a reasonable minimal loss of critical data based on device resources availability. The state change of the communication receive task further ensures a minimal response time by the controller which is very critical in cases of GOOSE application of IED 61850. The state change forces the schedule of the application specific tasks, which creates fairness in scheduling the communication and consumer tasks during the anomalous traffic condition.

[0070] Further, the present disclosure identifies a low priority consumer task among the one or more consumer tasks and assigns priority to the communication task equivalent or lower to the LP consumer task. The present disclosure utilizes the deviation of the actual schedule time for the LP consumer task from its estimated schedule time, and packet rate to indicate the kind of traffic, which can be used to determine whether the device was cyber attacked. Accordingly, a corrective action can be taken.

[0071] The present disclosure can be utilized in an embedded device (including and not limited to controllers, gateways, I/O modules, and the like) in a distributed control system that has Ethernet ports. The present disclosure can be configured to raise alarms or to provide the user (e.g., operations manager) a first level data if an attack has happened and needs further investigation. The present disclosure ensures a relatively higher stability of the device during the network flood without shutting down the Ethernet receive. As a result, the latency period

within which the applications need to respond during the network flood reduces and overall operational efficiency of the device is improved during the network flood.

[0072] It will be appreciated that features of the present disclosure are susceptible to being combined in various combinations without departing from the scope of the present disclosure as defined by the appended claims.

CLAIMS:

1. A method for ensuring execution of a plant process in a distributed control system during a network attack, the system comprises a controller configured to execute: at least one communication task comprising instructions to communicate with at least one field device; and at least one consumer task comprising instructions corresponding to at least one process application associated with the plant process, the method comprising:

defining a first state and a second state corresponding to the at least one communication task of the controller; wherein the at least one communication task has a priority greater than the at least one consumer task in the first state and the at least one communication task has a priority lesser than the at least one consumer task in the second state;

analyzing traffic of a network received at the controller, wherein the traffic comprises a plurality of packets processed during execution of the at least one communication task and the consumer task;

processing a scheduling logic configured to change state for the at least one communication task to one of the first state and the second state, on occurrence of a predefined condition corresponding to the traffic of the network ; and

scheduling, the at least one communication task to execute after the execution of the at least one consumer task in the second state.

2. The method as claimed in claim 1, wherein the predefined condition comprises at least one of:

a count of the plurality of packets received at the controller is greater than a threshold count during a defined interval; and

an inter-frame time between n and n-1th frame averaged over a period of time crosses a defined threshold.

3. The method as claimed in claim 1, wherein scheduling the execution of the at least one communication task further comprising:

identifying a low priority consumer task among a plurality of consumer tasks configured to be executed at the controller wherein the low priority consumer task has a lowest priority among the plurality of consumer tasks;

estimating a schedule time for an execution of the low priority consumer task; and

scheduling an execution of the at least one communication task after execution of the low priority consumer task in the second state.

4. The method as claimed in claim 3, wherein the scheduling logic is further configured to:

change state of the at least one communication task from the second state to one of a third state and a fourth state depending on difference between an actual load and an estimated load at the controller,

wherein the priority of the at least one communication task is less than the priority of the low priority consumer task in the third state and in the fourth state.

5. The method as claimed in claim 4, further comprising:

analyzing the traffic of the network at the controller to change state of the at least one communication task from the fourth state to the first state.

6. The method as claimed in claim 4, further comprising:

comparing the estimated schedule time of the low priority consumer task with an actual schedule time of the low priority consumer task to determine at least one anomalous traffic condition; and

initiating a corrective action in accordance with the at least one anomalous traffic condition.

7. The method as claimed in claim 1, further comprising:

achieving the second state of the at least one communication task by deferring the execution of the at least one communication task through priority inversion.

8. The method as claimed in claim 1, further comprising:

achieving the second state of the at least one communication task by dynamically changing the priority of the at least one communication task to a priority lower than the at least one consumer task.

9. A controller device for ensuring execution of a plant process in a distributed control system during a network attack, wherein the controller device is configured to execute: at least one communication task comprising instructions to communicate with at least one field device via an input/output (I/O) module directly or indirectly via a gateway/interface unit; and at least one consumer task comprising instructions corresponding to at least one process application, the controller device comprising:

a task manager configured to define a first state and a second state corresponding to the at least one communication task of the controller; wherein the at least one communication task has a priority greater than the at least one consumer task in the first state and the at least one communication task has a priority lesser than the at least one consumer task in the second state;

a traffic monitor configured to analyze traffic of a network received at the controller, wherein the traffic comprises a plurality of packets processed during execution of the at least one communication task and the consumer task;

a scheduling logic configured to change state for the at least one communication task from the first state to the second state on occurrence of a predefined condition corresponding to the traffic of the network; and

a scheduler configured to schedule the at least one communication task to execute after an execution of the at least one consumer task in the second state.

10. The controller device as claimed in claim 9, wherein the scheduler along with the scheduling logic, is further configured to:

identify a low priority consumer task among a plurality of consumer tasks configured to be executed at the controller wherein the low priority consumer task has a lowest priority among the plurality of consumer tasks;

estimate a schedule time for an execution of the low priority consumer task; and

schedule an execution of the at least one communication task after execution of the low priority consumer task in the second state.

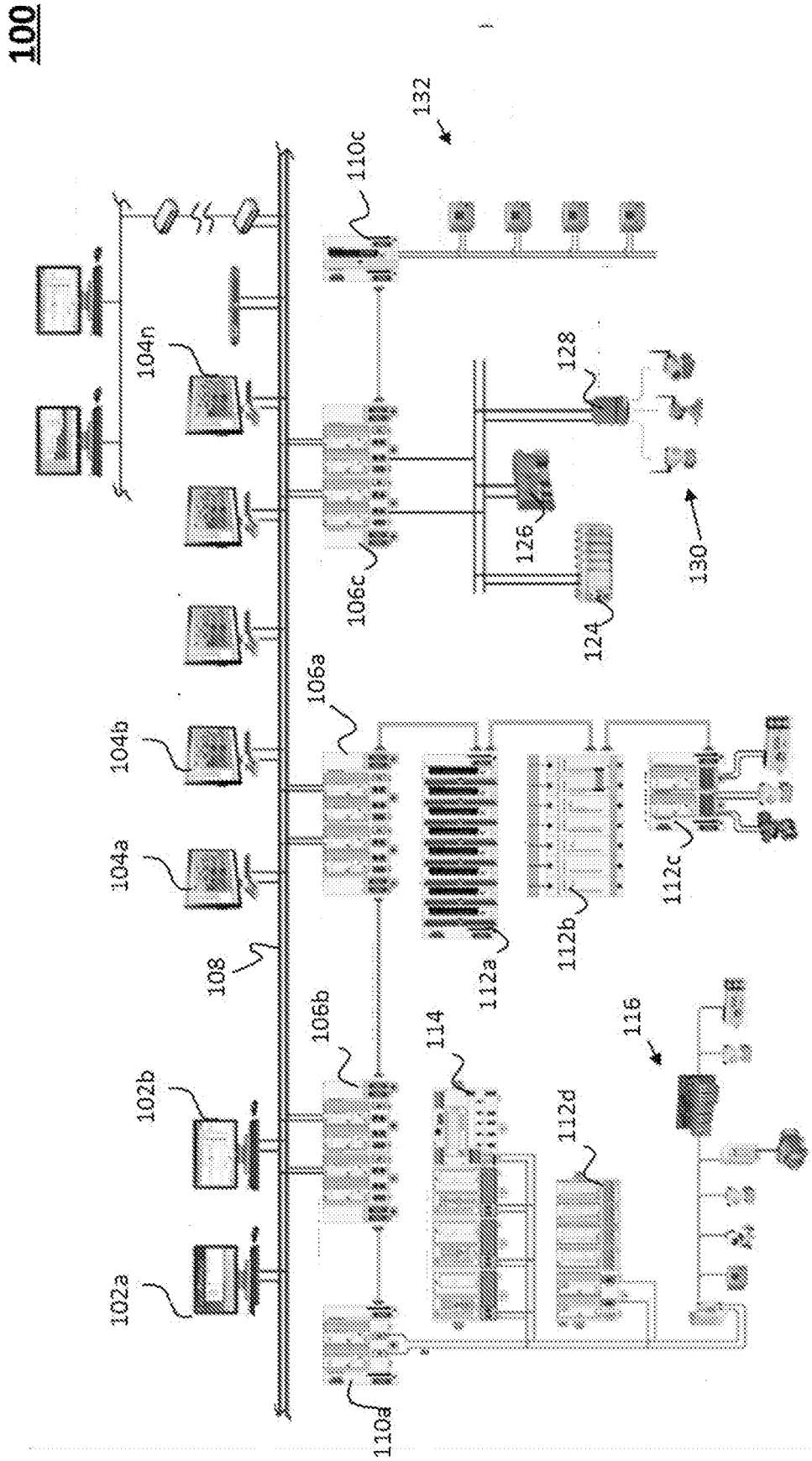
100
102a
102b
104a
104b
104n
106a
106b
106c
108
110a
110c
112a
112b
112c
112d
114
116
124
126
128
130
132

Figure 1

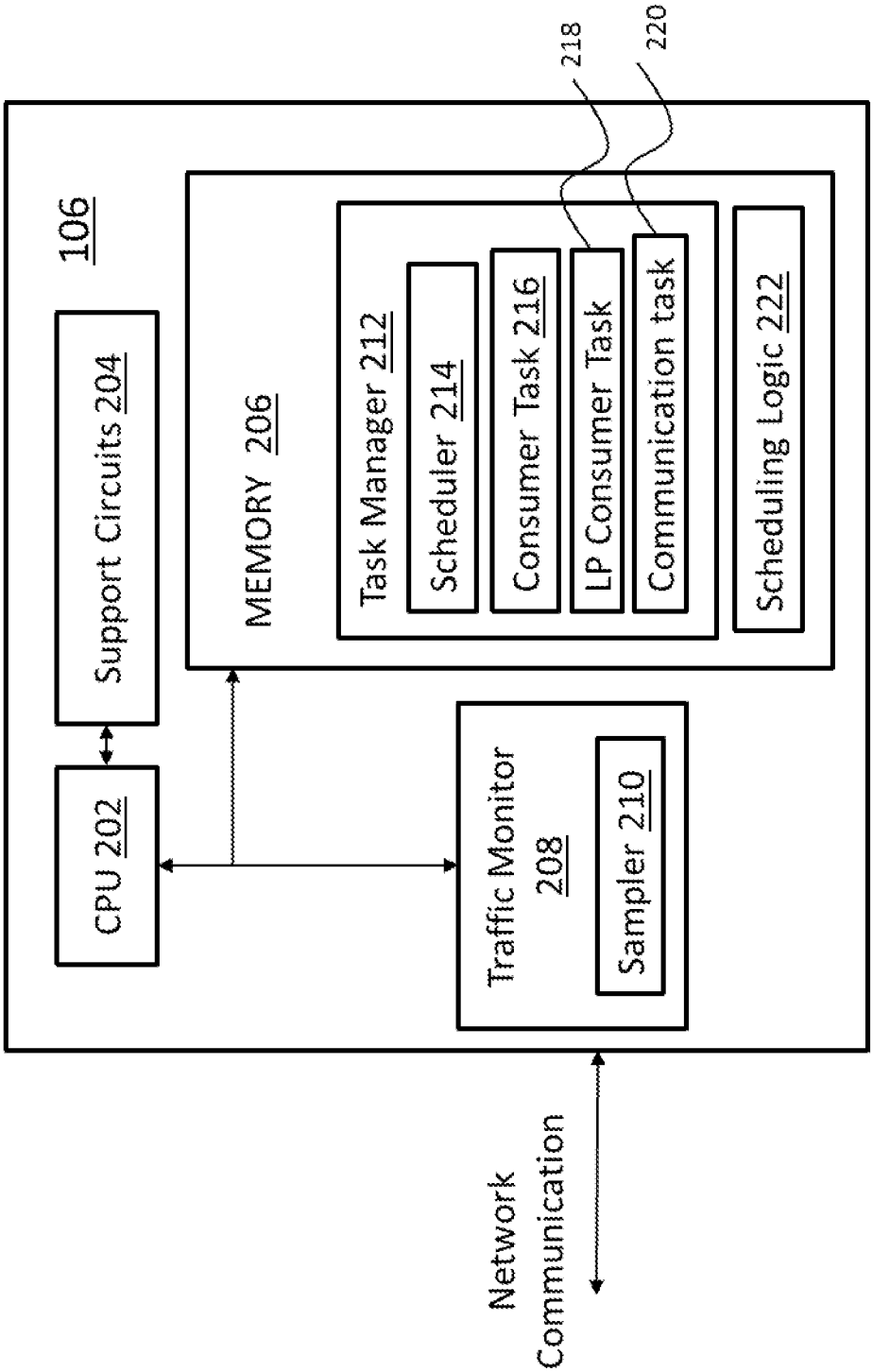
106
Support Circuits 204
CPU 202
MEMORY 206
Task Manager 212
Scheduler 214
Consumer Task 216
LP Consumer Task
Communication task
Scheduling Logic 222
218
220
Traffic Monitor 208
Sampler 210
Network Communication

Figure 2

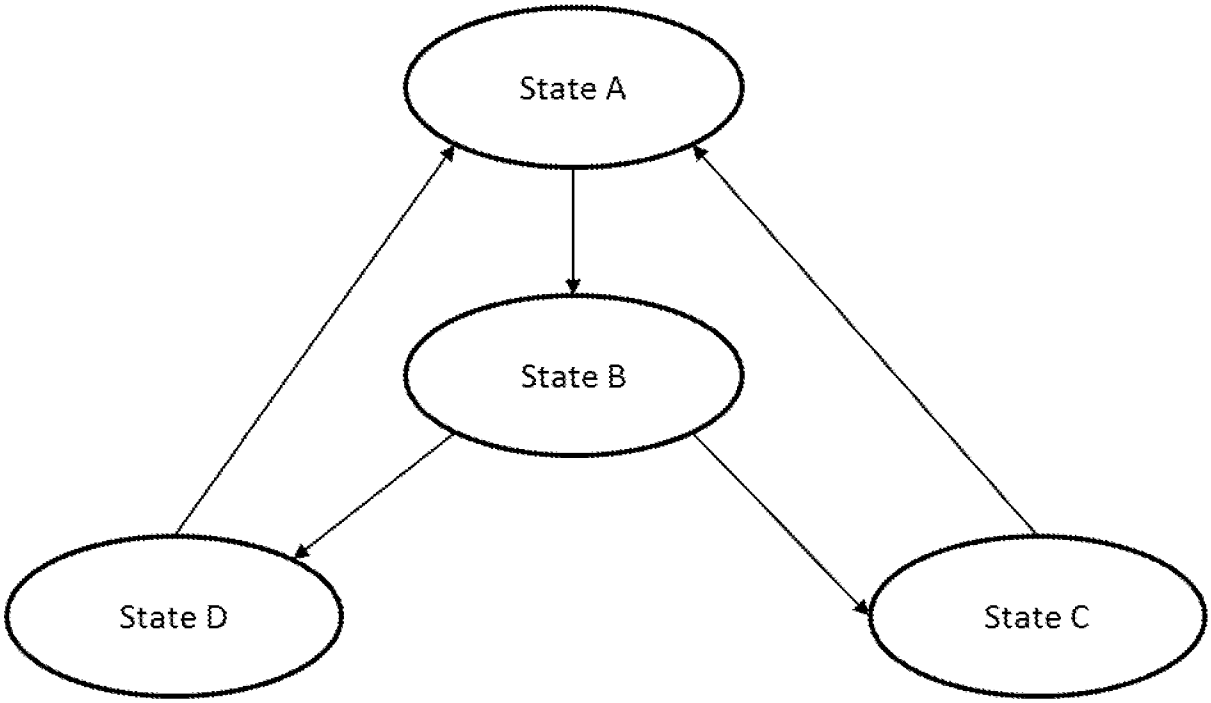

Figure 3

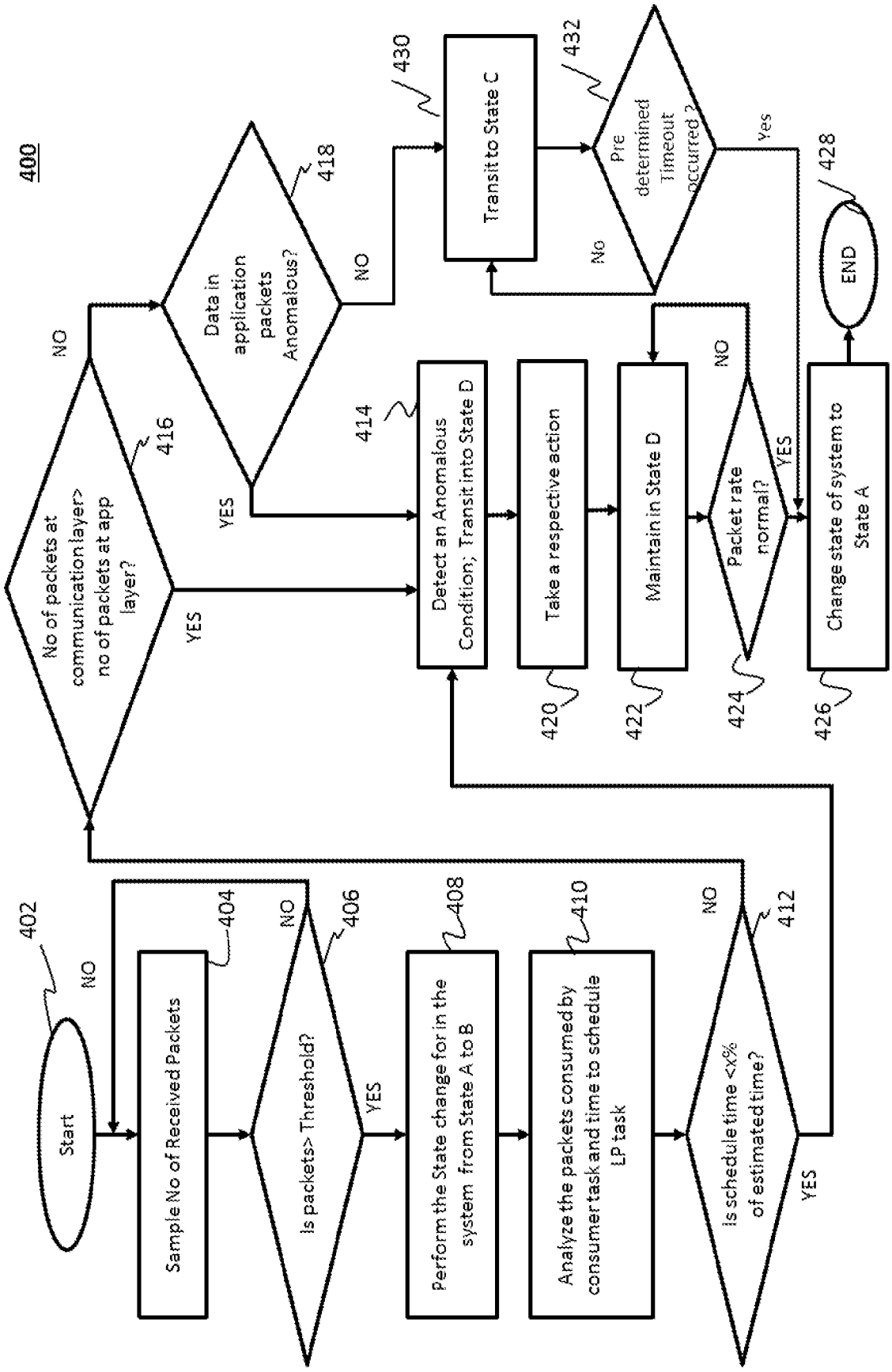
400
Start
402
Sample No of Received Packets
404
Is packets> Threshold?
406
Perform the State change for in the system from State A to B
408
Analyze the packets consumed by consumer task and time to schedule LP task
410
Is schedule time <X% of estimated time?
412
NO
YES
No of packets at communication layer> no of packets at app layer?
416
Data in application packets Anomalous?
418
Transit to State C
430
Pre determined Timeout occurred ?
432
Detect an Anomalous Condition; Transit into State D
414
Take a respective action
420
Maintain in State D
422
Packet rate normal?
424
Change state of system to State A
426
END
428

Figure 4

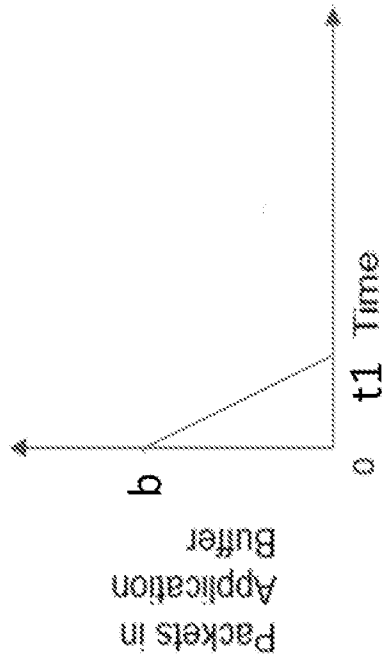

Figure 5B

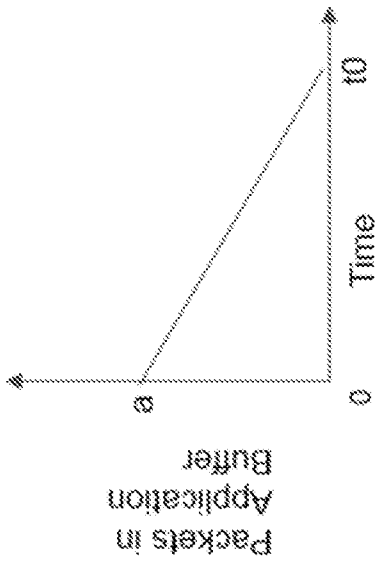

Figure 5A

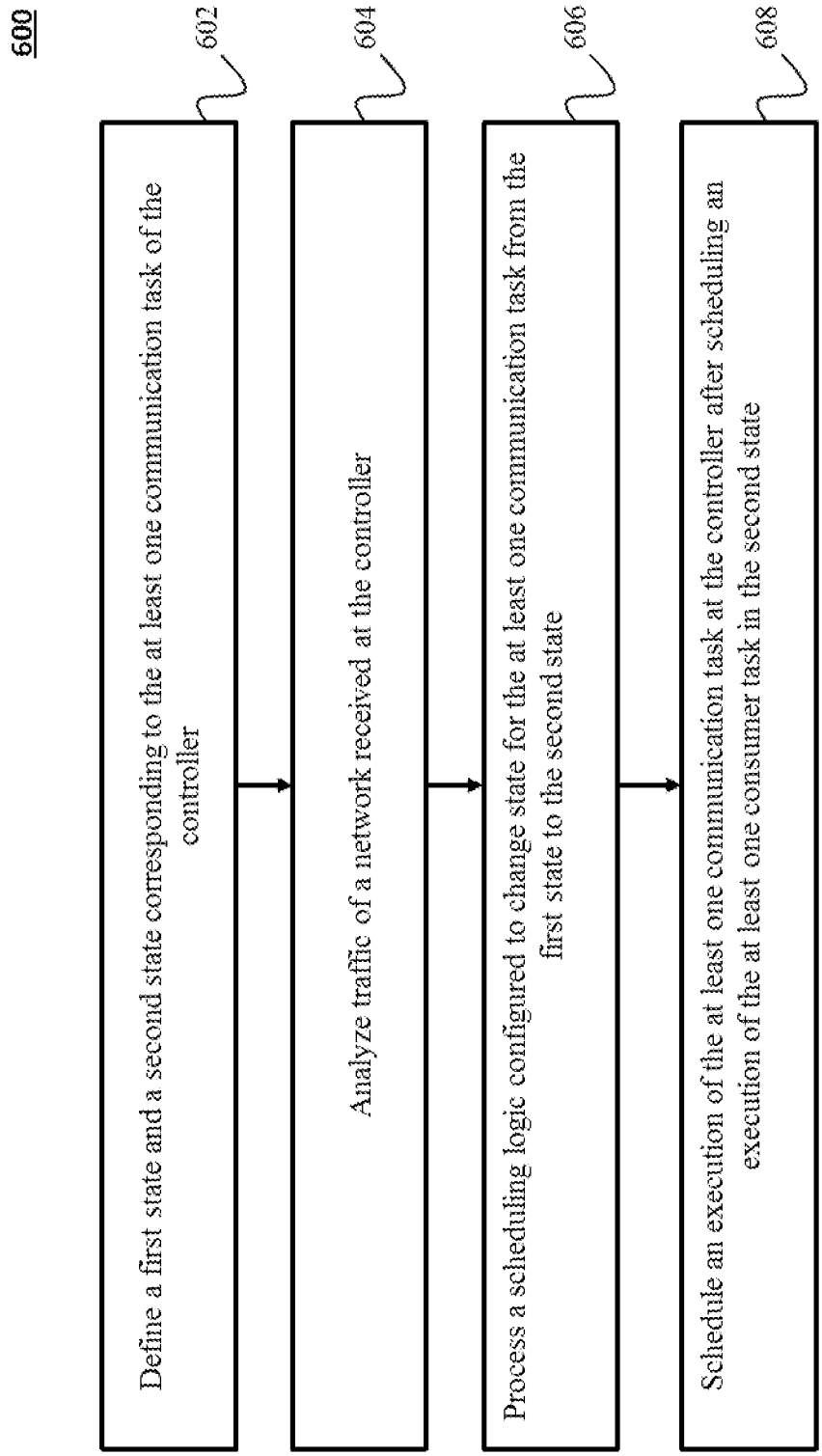
600
602
Define a first state and a second state corresponding to the at least one communication task of the controller
604
Analyze traffic of a network received at the controller
606
Process a scheduling logic configured to change state for the at least one communication task from the first state to the second state
608
Schedule an execution of the at least one communication task at the controller after scheduling an execution of the at least one consumer task in the second state

Figure 6

INTERNATIONAL SEARCH REPORT

International application No
PCT/IB2015/059060

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04L29/06 H04L29/08
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	EP 1 288 748 A2 (HITACHI LTD [JP]) 5 March 2003 (2003-03-05) abstract paragraph [0008] paragraph [0018] paragraph [0024] paragraph [0047] paragraph [0104] - paragraph [0121] paragraph [0141] - paragraph [0144] paragraph [0156] claim 18 figures 21-34, 43 -----	1-10
A	US 2007/050777 A1 (HUTCHINSON THOMAS W [CA] ET AL) 1 March 2007 (2007-03-01) the whole document -----	1-10

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
8 February 2016	16/02/2016

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Bae, Jun-Young

1

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/IB2015/059060

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 1288748 A2	05-03-2003	DE 60217593 T2	25-10-2007
		EP 1288748 A2	05-03-2003
		EP 1746473 A1	24-01-2007
		JP 2003067201 A	07-03-2003
		US 2003046324 A1	06-03-2003
		US 2006059491 A1	16-03-2006
US 2007050777 A1	01-03-2007	NONE	