International Journal of Control, Automation and Systems VV(X) (YYYY) 1-8
http://dx.doi.org/10.1007/s12555-xxx-xxxx-x
ISSN:1598-6446 eISSN:2005-4092
http://www.springer.com/12555

MPC-Based Co-Design of Control and Routing for Wireless Sensor and Actuator Networks

Dai Satoh, Koichi Kobayashi*, and Yuh Yamashita

Abstract: A wireless sensor and actuator network (WSAN) is a class of networked control systems. In WSANs, sensors and actuators are located in a distributed way, and communicate to controllers through a wireless communication network such as a multi-hop network. In this paper, we propose a model predictive control (MPC) method for co-design of control and routing of WSANs. MPC is an optimal control strategy based on numerical optimization. The control input is calculated by solving the finite-time optimal control problem at each discrete time. In the proposed method, a WSAN is modeled by a switched linear system. In the finite-time optimal control problem, a control input and a mode corresponding to a communication path are optimized simultaneously. The proposed method is demonstrated by a numerical example.

Keywords: co-design of control and routing, mixed logical dynamical system, model predictive control, wireless sensor and actuator networks.

1. INTRODUCTION

During the last decade, networked control systems (NCSs) have attracted much attention (see, e.g., [1, 2]). In an NCS, the control input and the measured signal are transmitted to plants and controllers through a communication network, respectively. Needless to say, it is important to consider a wireless network in NCSs. For large-scale NCSs, it is useful to utilize multi-hop wireless networks (see, e.g., [3–5]). A multi-hop wireless network is generally composed of a large number of wireless nodes deployed randomly in a two- (or three-) dimensional space. A wireless sensor and actuator network (WSAN) is well known as a control system using a multi-hop wireless network (see, e.g., [6–20]). A WSAN is a control system where components such as sensors and actuators are located in a distributed way, and communicate to controllers through a wireless communication network. There are several applications such as environmental monitoring/control [6, 7], healthcare [14], and light control [20] (see also [17]).

On the other hand, model predictive control (MPC) has been widely studied for control of complex systems (see,e.g., [21]). MPC is an optimal control strategy based on numerical optimization. The control input is calculated by solving the finite-time optimal control problem at each discrete time. MPC is well known as a control method of constrained linear/nonlinear/hybrid systems, and has many applications such as automobile, gauge and tension control in rolling processes, and control of demand and supply in smart grid (see, e.g., [21]). For MPC of hybrid systems, a mixed logical dynamical (MLD) system plays an important role (see, e.g., [22]). Advanced MPC methods have been recently proposed, e.g., MPC for time-varying delay systems [23], robust MPC [24,25], fuzzy predictive control [26,27]. Analysis and control for WSANs have been studied so far. In [10], a WSAN was analyzed using a colored Petri net. In [12], several topics were discussed, e.g., real-time scheduling algorithms and a simulator. In [15], event-triggered control has been applied to a WSAN. In [16], a WSAN was analyzed based on the set packing algorithm and the traveling salesman problem. In [18], a PID controller was utilized. In [19], compensation of packet loss in a WSAN was studied. In [28, 29], state estimation for more general networks including WSANs was studied.

MPC methods for WSANs have also been studied so far. In [9], the conventional MPC method was applied to hydronic systems implemented by a WSAN. Also in [8], the conventional MPC method was applied to a feedback process trainer in which the plant and the controller are connected through a wireless communication network. In [11], the conventional MPC method and the extended Kalman filter were combined, and a WirelessHART architecture was utilized in implementation of a WSAN. In

Manuscript received XX XX, 201X; revised XX XX, 201X; accepted XX XX, 201X. Recommended by Associate Editor XXXX under the direction of Editor YYYY. This work was partly supported by the Telecommunications Advancement Foundation and JSPS KAKENHI Grant Numbers 17K06486, 16H04380.
Dai Satoh, Koichi Kobayashi, and Yuh Yamashita are with the Graduate School of Information Science and Technology, Hokkaido University, Sapporo 060-0814, Japan (e-mail: {d_satoh@stl.,k-kobaya@,yuhyama@}ssi.ist.hokudai.ac.jp).
* Corresponding author.

[13], the MPC problem considering both the control performance and delays over a WSAN was studied. However, in these MPC methods, simultaneous optimal design of control and routing has not been studied. In [11], both routing and control was considered. In routing design, both a primary path and a backup path are introduced, but optimization of communication paths is not discussed. In order to improve quality of control, it is important to consider designing both control and routing simultaneously. Since the candidates of communication paths can be enumerated (see, e.g., [30–32]), it is important to consider the problem of choosing the optimal paths. To the best of our knowledge, such simultaneous optimization problem for WSANs has not been studied so far.

In this paper, we propose a new method of MPC for co-design of control and routing of WSANs. In the simultaneous optimization problem for WSANs, not only state/input constraints but also constraints on network structure must be imposed. Since MPC can also deal with constraints on network structure, it is appropriate as a control method for the simultaneous optimization problem. First, we propose a switched linear system model for a WSAN. Here, based on conventional sampled-data control, we assume that the time that the control input and the communication path are computed is given in advance. Under this assumption, a time sequence of the state in the intersampling interval is modeled as a linear system. The time that the control input is updated is changed depending on the communication path. Since the different linear system is obtained for each communication path, the overall model is given by a switched linear system. The mode (the discrete state) in this model corresponds to the communication path. A switched linear system is frequently used as a model of NCSs (see, e.g., [3, 33, 34]). Next, consider the finite-time optimal control problem for the obtained switched linear system. In this problem, both the control input and the communication path are optimized simultaneously. By rewriting the switched linear system into the MLD system, this problem can be reduced to a mixed integer quadratic programming (MIQP) problem, which can be solved by a commercial/free solver. Finally, the proposed method is demonstrated by a numerical example.

The main contributions of this paper are as follows:

(i) A switched linear system model for a WSAN is proposed.
(ii) The finite-time optimal control problem in which the control input and the communication path are optimized is formulated, and is reduced to an MIQP problem.
(iii) The effectiveness of the proposed method is validated by a numerical example.

This paper is organized as follows. In Section 2., the notion of WSANs is defined. In Section 3., the proposed MPC method is explained. In Section 4., a modeling method of WSANs for MPC is proposed. In Section 5., the finite-time optimal control problem is formulated, and is reduced to an MIQP problem. In Section 6., a numerical example is presented. In Section 7., we conclude this paper.

Notation: Let $\mathcal{R}$ denote the set of real numbers. Let $\{0,1\}^n$ denote the set of n-dimensional vectors, which consists of elements 0 and 1. Let I_n, $0_{m\times n}$ denote the $n\times n$ identity matrix, the $m\times n$ zero matrix, respectively. For simplicity, we sometimes use the symbol 0 instead of $0_{m\times n}$, and the symbol I instead of I_n. For the vector v, let $v^\top$ denote the transpose of v. For the finite set A, let $|A|$ denote the number of elements of A.

2. WIRELESS SENSOR AND ACTUATOR NETWORKS

In this section, a WSAN is explained. A WSAN studied here is illustrated by an undirected graph such as Fig. 1.

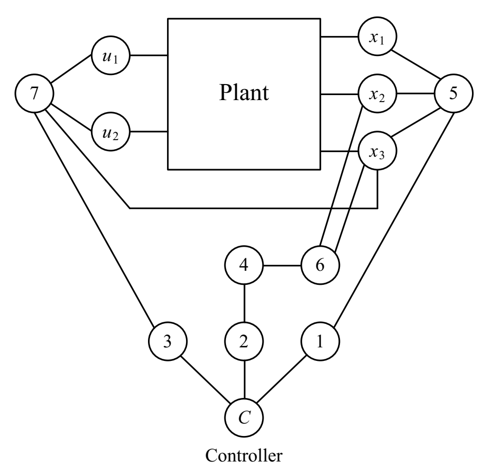

Fig. 1. Example of wireless sensor and actuator networks.

Nodes in WSANs consist of five kinds of nodes, i.e.,

(i) plant nodes that correspond to plants,
(ii) sensor nodes that correspond to sensors,
(iii) actuator nodes that correspond to actuators,
(iv) controller nodes that correspond to controllers,
(v) communication nodes that communicate messages about sensing and actuating from plants (controllers) to controllers (plants).

As an example, consider the WSAN given by the undirected graph in Fig. 1. In this example, we suppose that the number of plants is 1, and the plant has three states and two control inputs. A controller node is given by node C, and the number of controllers is given by 1. Sensor nodes are given by nodes x_1, x_2, and x_3. The sensor nodes x_1, x_2, and x_3 measure the first, second, and third elements of the state, respectively. Actuator nodes are given by nodes u_1 and u_2. Communication nodes are given by $1, 2, \ldots, 7$.

In this paper, only when the controller requests information about the measured state, each sensor node sends the message about the measured state. The messages about control inputs are sent from the controller node to the actuator nodes through communication nodes. After the actuator nodes receive the messages, the control input is immediately updated. We assume that the plant node is connected to only sensor and actuator nodes, and is not directly connected to communication nodes. We also suppose that a routing protocol is given by a proactive routing protocol (see, e.g., [35]), and the set of available paths from a controller (sensor and actuator nodes) to sensor and actuator nodes (a controller) is given by the undirected graph of the WSAN in advance. Such set can be frequently computed (see, e.g., [30–32]). Furthermore, several results on implementation of WSANs have been obtained so far (see, e.g., [6,8]). Using the existing results on implementation, the closed-loop system of WSANs can be implemented.

Here, a WSAN is formally defined as follows.

Definition 1: A WSAN is given by a tuple

$$\mathcal{N} = (\mathcal{G}, \Sigma),$$

where:

- $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is an undirected connected graph that expresses the radio connectivity of the network, where $\mathcal{V}$ is the set of vertices (nodes), and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges. The set $\mathcal{V}$ is decomposed to $\mathcal{V} = \mathcal{P} \cup \mathcal{S} \cup \mathcal{A} \cup \mathcal{C} \cup \mathcal{H}$, where
    - $\mathcal{P}$ is the set of plant nodes (assume $|\mathcal{P}| = 1$),
    - $\mathcal{S}$ is the set of sensor nodes,
    - $\mathcal{A}$ is the set of actuator nodes,
    - $\mathcal{C}$ is the set of controller nodes (assume $|\mathcal{C}| = 1$),
    - $\mathcal{H}$ is the set of communication nodes.

  For two nodes i, j, if i is connected to j (i.e., j is connected to i), then the message included in i (j) may be sent to j (i).
- Σ is the dynamics of plants, that is,

  $$\Sigma: \ x(k+1) = Ax(k) + Bu(k),$$

  where $k \in \{0, 1, \ldots\}$ is the discrete time, and $x(k) \in [\underline{x}, \overline{x}] \subseteq \mathcal{R}^n$ and $u(k) \in [\underline{u}, \overline{u}] \subseteq \mathcal{R}^m$ are the state and the control input of the plant, respectively (the sets $[\underline{x}, \overline{x}]$ and $[\underline{u}, \overline{u}]$ are given). The matrices $A \in \mathcal{R}^{n \times n}$ and $B \in \mathcal{R}^{n \times m}$ are given in advance. An actuator node is assigned to each element of the control input. That is, $|\mathcal{A}| = m$ holds. A sensor node is assigned to each element of the state. That is, $|\mathcal{S}| = n$ holds.

This definition is given by reference to [15, 16, 18, 20]. The definition of multi-hop control networks [3,34] is also used as reference.

In control of WSANs, co-design of control and routing must be considered. In this paper, a new co-design method using model predictive control (MPC) for WSANs is proposed (see, e.g., [21] for details of MPC).

3. MODEL PREDICTIVE CONTROL FOR CO-DESIGN OF CONTROL AND ROUTING

In this section, we explain the proposed MPC method for co-design of control and routing.

The proposed procedure of MPC for WSANs is presented as follows. Let L denote the criterial length of the paths from the controller to sensor/actuator nodes. The length L may be given by the maximum length of the paths.

Procedure of MPC for WSANs:

Step 0: Preset the initial control input for the plant in advance.

Step 1: Calculate

(i) control inputs,

(ii) paths from the controller to actuator nodes for sending messages about control inputs,

(iii) paths from sensor nodes to the controller for sending messages about measured states

by solving the finite-time optimal control problem. Set $t = 0$.

Step 2: Send messages about the control input to actuator nodes through the chosen path. Send messages about the request of the measured state to sensor nodes through the chosen path. These messages reach until $t = L$.

Step 3: After the message about the next control input reaches to actuator nodes, apply the next control input to the plant. After the message about the request of the measured state reaches to sensor nodes, measure the state.

Step 4: Send messages about the measured state to the controller, where paths used are the same as paths used in sending messages about the request.

Step 5: Wait until time $t = 2L$. Then, the controller can receive messages about the measured state. Return to Step 1.

In the proposed procedure, the time that messages to actuator and sensor nodes are sent is the same.

Hereafter, in Section 4., we will explain details of the modeling method. In Section 5., we will explain details of the finite-time optimal control problem.

4. MODELING METHOD

A WSAN is modeled by using the following procedure.

Procedure of modeling a WSAN:

Step 1: Enumerate the candidates of modes based on the candidates of communication paths.

Step 2: For each mode, derive a linear system in the time interval $[0, 2L]$.

Using the example in Fig. 1, we explain the above procedure.

First, we explain Step 1. From Fig. 1, we suppose that L is given by $L = 4$. Here, we assume that (i) a communication node can store only one message, (ii) one path is assigned for each message. From these assumptions, we see that the controller cannot communicate to both actuator nodes u_1 and u_2 simultaneously. Then, we can obtain the following cases.

(i) The controller communicates to the sensor nodes x_1, x_2, and x_3.

(ii) The controller communicates to the sensor nodes x_1, x_2, and the actuator node u_1.

(iii) The controller communicates to the sensor nodes x_2, x_3, and the actuator node u_1.

(iv) The controller communicates to the sensor nodes x_1, x_3, and the actuator node u_1.

(v) The controller communicates to the sensor nodes x_1, x_2, and the actuator node u_2.

(vi) The controller communicates to the sensor nodes x_2, x_3, and the actuator node u_2.

(vii) The controller communicates to the sensor nodes x_1, x_3, and the actuator node u_2.

In this paper, each case is called a mode. We can consider several methods for setting the mode. The above method is one of them. Depending on devices implementing communication nodes, we may consider other methods.

Next, we explain Step 2. As an example, consider modeling a linear system for mode (ii). Noting that $2L = 8$, we can obtain

$$\begin{aligned}
x(1) &= Ax(0) + B\begin{bmatrix} u_1(-1) \\ u_2(-1) \end{bmatrix}, \\
x(2) &= Ax(1) + B\begin{bmatrix} u_1(-1) \\ u_2(-1) \end{bmatrix}, \\
x(3) &= Ax(2) + B\begin{bmatrix} u_1(-1) \\ u_2(-1) \end{bmatrix}, \\
x(4) &= Ax(3) + B\begin{bmatrix} u_1(0) \\ u_2(-1) \end{bmatrix}, \\
x(5) &= Ax(4) + B\begin{bmatrix} u_1(0) \\ u_2(-1) \end{bmatrix}, \\
x(6) &= Ax(5) + B\begin{bmatrix} u_1(0) \\ u_2(-1) \end{bmatrix}, \\
x(7) &= Ax(6) + B\begin{bmatrix} u_1(0) \\ u_2(-1) \end{bmatrix}, \\
x(8) &= Ax(7) + B\begin{bmatrix} u_1(0) \\ u_2(-1) \end{bmatrix},
\end{aligned}$$

where $u_1(-1)$ and $u_2(-1)$ are the initial control input given in advance, and $u_1(0)$ is the control input, which is a decision variable in the finite-time optimal control problem. In the above expressions, $u_2(0)$ is not included, but is included as a dummy decision variable in the finite-time optimal control problem. From the above expressions, we can obtain the following linear system for mode (ii):

$$\hat{x}(1) = \hat{A}\hat{x}(0) + B_2^0 u(0) + B_2^{-1} u(-1),$$

where

$$\hat{x}(1) = \begin{bmatrix} x(1) \\ x(2) \\ x(3) \\ x(4) \\ x(5) \\ x(6) \\ x(7) \\ x(8) \end{bmatrix}, \quad \hat{x}(0) = \begin{bmatrix} x(-7) \\ x(-6) \\ x(-5) \\ x(-4) \\ x(-3) \\ x(-2) \\ x(-1) \\ x(0) \end{bmatrix},$$

$$u(0) = \begin{bmatrix} u_1(0) \\ u_2(0) \end{bmatrix}, \quad u(-1) = \begin{bmatrix} u_1(-1) \\ u_2(-1) \end{bmatrix},$$

and defining $[\, B_1 \;\; B_2 \,] := B$ ($B_1, B_2 \in \mathcal{R}^n$), the matrices $\hat{A}$, B_2^0, and B_2^{-1} are given by

$$\hat{A} = \begin{bmatrix} 0 & \cdots & 0 & A \\ 0 & \cdots & 0 & A^2 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & A^8 \end{bmatrix},$$

$$B_2^0 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ B_1 \\ B_1 + AB_1 \\ (I + A + A^2)B_1 \\ (I + A + A^2 + A^3)B_1 \\ (I + A + A^2 + A^3 + A^4)B_1 \end{bmatrix},$$

$$B_2^{-1} = \begin{bmatrix} B \\ (I + A)B \\ (I + A + A^2)B \\ B_2 + (A + A^2 + A^3)B \\ AB_2 + (A^2 + A^3 + A^4)B \\ A^2B_2 + (A^3 + A^4 + A^5)B \\ A^3B_2 + (A^4 + A^5 + A^6)B \\ A^4B_2 + (A^5 + A^6 + A^7)B \end{bmatrix},$$

respectively. Thus, for the fixed mode, we can obtain a linear system.

Consider deriving a general form of a linear system for each mode. Let $l \in \{0, 1, 2, \ldots, \}$ denote the label for periods of communication. For each l, both the control input and the path are calculated. In general, the linear system for the mode α can be obtained by

$$\hat{x}(l+1) = \hat{A}\hat{x}(l) + B_{\alpha}^{0}u(l) + B_{\alpha}^{-1}u(l-1), \tag{1}$$

where

$$\hat{x}(l) = \begin{bmatrix} x(2Ll-2L+1) \\ x(2Ll-2L+2) \\ \vdots \\ x(2Ll) \end{bmatrix}, \quad \hat{A} = \begin{bmatrix} 0 & \cdots & 0 & A \\ 0 & \cdots & 0 & A^2 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & A^{2L} \end{bmatrix}.$$

The matrices B_{α}^{0} and B_{α}^{-1} are different for each mode. The matrix $\hat{A}$ in (1) does not depend on the mode. The mode α in (1) may be different for each l. Hereafter, the mode α in (1) is replaced with $\alpha(l)$, and the set of modes is denoted by $\mathcal{M} = \{1,2,\ldots,M\}$. Thus, a WSAN can be modeled by the switched linear system with delay (1).

5. FINITE-TIME OPTIMAL CONTROL PROBLEM

In this section, we formulate the finite-time optimal control problem for a WSAN. The finite-time optimal control problem studied in this paper is to find a time sequence of the control input and a time sequence of the mode minimizing a given cost function.

For the switched linear system with delay (1) expressing a WSAN, consider the following finite-time optimal control problem.

Problem 1: For the switched linear system with delay (1) expressing a WSAN, suppose that the initial control input $u(-1)$, the initial state $x(0)$, and the prediction horizon N are given. Then, find a control input sequence $u(0), u(1), \ldots, u(N-1)$ and a mode sequence $\alpha(0), \alpha(1), \ldots, \alpha(N-1)$ minimizing the following cost function:

$$J = \sum_{l=0}^{N-1} \left\{ \hat{x}^{\top}(l) Q \hat{x}(l) + u^{\top}(l) R u(l) \right\}, \tag{2}$$

where $Q \geq 0$ and $R > 0$ are given weighting matrices.

In MPC, the first value of the control input sequence obtained is applied to the plant. This problem can be rewritten as an MIQP problem. In order to explain this fact, we consider transforming (1) into an MLD system [22].

First, define $\bar{x}(l) := [\, \hat{x}^{\top}(l) \;\; u^{\top}(l-1) \,]^{\top}$. Then, (1) can be rewritten as

$$\bar{x}(l+1) = \bar{A}_{\alpha(l)}\bar{x}(l) + \bar{B}_{\alpha(l)}u(l), \tag{3}$$

where

$$\bar{A}_{\alpha(l)} = \begin{bmatrix} \hat{A} & B_{\alpha(l)}^{-1} \\ 0 & 0 \end{bmatrix}, \quad \bar{B}_{\alpha(l)} = \begin{bmatrix} B_{\alpha(l)}^{0} \\ I \end{bmatrix}.$$

Next, a binary variable $\delta_i(l) \in \{0,1\}$, $i \in \mathcal{M}$ is defined by the following relation:

$$[\alpha(l) = i] \leftrightarrow [\delta_i(l) = 1], \;\; i \in \mathcal{M}.$$

Then, the equality constraint

$$\delta_1(l) + \delta_2(l) + \cdots + \delta_M(l) = 1, \;\; l = 0,1,\ldots,N-1 \tag{4}$$

must be imposed. Using a binary variable, (3) can be rewritten as

$$\bar{x}(l+1) = \sum_{i=1}^{M} z_i(l), \tag{5}$$

$$z_i(l) = \delta_i(l)\left\{\bar{A}_i\bar{x}(l) + \bar{B}_i u(l)\right\}. \tag{6}$$

In addition, (6) is equivalent to the following linear inequalities

$$g_{\min}\delta_i(l) \leq z_i(l) \leq g_{\max}\delta_i(l), \tag{7}$$

$$\begin{aligned}&(\bar{A}_i\bar{x}(l) + \bar{B}_i u(l)) - g_{\max}(1-\delta_i(l)) \leq z_i(l) \\ &\leq (\bar{A}_i\bar{x}(l) + \bar{B}_i u(l)) - g_{\min}(1-\delta_i(l)).\end{aligned} \tag{8}$$

where $g_{\min}$ and $g_{\max}$ are constant vectors such that for any $x(k) \in [\underline{x}, \overline{x}]$ and $u(k) \in [\underline{u}, \overline{u}]$, the inequality $g_{\min} \leq \bar{A}_i\bar{x}(l) + \bar{B}_i u(l) \leq g_{\max}$, $i \in \mathcal{M}$ is satisfied. Thus, the MLD system expressing (1) can be obtained as (4), (5), (7), and (8). We remark here that the MLD system obtained is linear with respect to $\bar{x}(l)$, $z_i(l)$, and $\delta_i(l)$.

Using the MLD system obtained, Problem 1 can be rewritten as the following problem.

Problem 2: For the MLD system (4), (5), (7), and (8), suppose that the initial state $\bar{x}(0)$ and the prediction horizon N are given. Then, find continuous decision variables $u(0), u(1), \ldots, u(N-1)$, $z_i(0), z_i(1), \ldots, z_i(N-1)$ and binary decision variables $\delta_i(0), \delta_i(1), \ldots, \delta_i(N-1)$ minimizing the following cost function:

$$J = \sum_{l=0}^{N-1} \left\{ \bar{x}^{\top}(l) \begin{bmatrix} Q & 0 \\ 0 & 0 \end{bmatrix} \bar{x}(l) + u^{\top}(l) R u(l) \right\}.$$

By a simple calculation, Problem 2 can be equivalently transformed into an MIQP problem. As was pointed out in Section 2., the closed-loop system of WSANs can be implemented by using the existing techniques. On the other hand, an MIQP problem must be solved in the controller node. Hence, the controller node must be implemented by using the personal computer that can solve an MIQP problem. As software in the controller node, we can use e.g., IBM ILOG CPLEX with MATLAB.

6. NUMERICAL EXAMPLE

As an example, consider the WSAN in Fig. 1 again, where matrices A and B in the dynamics Σ are given by

$$A = \begin{bmatrix} 0.996 & 0.02 & 0.01 \\ 0.01 & 1.005 & 0 \\ 0 & 0.04 & 1.001 \end{bmatrix}, \quad B = \begin{bmatrix} 0.6 & 0 \\ 0.8 & 0.5 \\ 0 & 0.7 \end{bmatrix}.$$

As was explained in Section 4., this WSAN is modeled by the 7-mode switched linear system. Next, setting of the finite-time optimal control problem is explained. The initial control input for the plant is given by $[\,0 \;\; 0\,]^\top$. The initial state is given by $x(0) = [\,20 \;\; 10 \;\; 5\,]^\top$. The prediction horizon N is given by $N = 2$. The weighting matrices Q and R are given by $10I$ and I, respectively.

We present the computational results. Fig. 2 and Fig. 3 show times responses of the state and the control input, respectively. The mode sequence is obtained as

$$\begin{aligned}&2 \to 4 \to 2 \to 4 \to 4 \to 7 \to 2 \to 6 \to 1 \to 6\\&\to 2 \to 5 \to 6 \to 4 \to 6.\end{aligned}$$

From Fig. 2, we see that the state converges to a neighborhood of the origin. Circles in Fig. 3 imply the control inputs computed at this time.

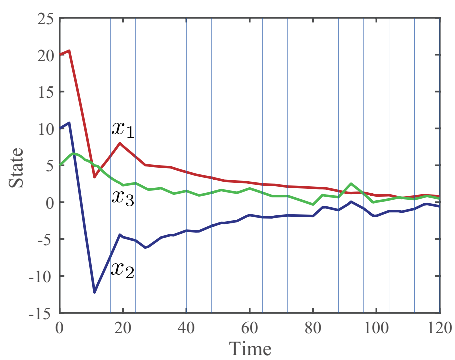

Fig. 2. Time response of the state. Red line: the first element of the state (x_1). Blue line: the second element of the state (x_2). Green line: the third element of the state (x_3).

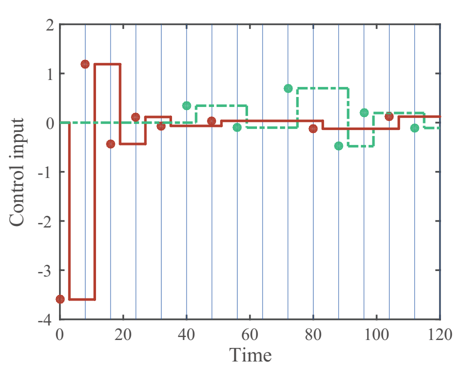

Fig. 3. Time response of the control input. Solid line: the control input u_1. Dash line: the control input u_2. Circles imply the control inputs computed at $2Ll$, $l = 0, 1, 2, \ldots$. After the message about the control input reaches the plant, computed control inputs are applied to the plant.

For example, noting that the initial mode is obtained as 2, the first element u_1 of the control input is updated at time 3. The mode at time 40 is 7, and the second element u_2 of the control input is updated at time 43. In this example, since $x_1(0)$ is relatively large, u_1 is updated preferentially in the transient response. Next, the mode at time 64 is 1. This implies that the control input is not updated in the time interval $[64, 72)$. In the proposed method, when update of the control input is not needed, the message about the control input may not be sent. From these observations, we see that routing and control are optimized simultaneously. Since such optimization is not considered in the existing methods, routing may be fixed. As a result, communication is periodic, and may be redundant. In the proposed method, since routing is adjusted depending on the current state, communication is generally aperiodic. Moreover, in the existing method, communication paths are not optimized. Hence, under the cost function (2) in Problem 1, the control performance of the proposed method is equal to or better than that of the existing methods. Thus, the proposed method provides us a better performance from the viewpoints of both communication and control.

Finally, we explain the computation time for solving Problem 2 (i.e., the MIQP problem). In this example, Problem 2 was solved 15 times. Then, the mean computation time and the worst computation time are 0.72 sec and 1.16 sec, respectively, where we used IBM ILOG CPLEX Optimizer 12.6.2 as an MIQP solver on the computer with Intel Core i7-4770K 3.50GHz processor and 32GB memory. Since the candidates of paths are enumerated off-line, Problem 2 can be solved fast. Developing a faster computation method for more complex WSANs is future work. Furthermore, if we suppose that the target computation time is about 1 sec, we see that $N = 2$ is appropriate. According to the conventional MPC method (see, e.g., [21]), a longer N enables us a higher control performance. However, we must consider the computation time.

7. CONCLUSION

In this paper, we proposed an MPC method for WSANs. A WSAN is modeled by a switched linear system, where the mode (the discrete state) corresponds to the paths from the controller to sensor and actuator nodes. In the proposed procedure of MPC, the control input and the paths are re-computed at a given time, where the time interval between re-computations is given by a constant.

The proposed MPC-based co-design method provides us a basic result for control of WSANs.

There are several open problems. In the proposed method, the time that the control and the path are re-computed is given, but it should be dynamically changed based on the state. It is important to combine the proposed MPC method with event-triggered/self-triggered control methods (see, e.g., [36] for details of event-triggered/self-triggered control). For each control input, we may consider different timing of re-computation. It is also important to consider multirate sampled-data control. Technical issues on model uncertainty and disturbances must be overcome. Then, the existing robust MPC methods (see, e.g., [25]) will help us development of MPC for WSANs under uncertainty. Finally, an application of the proposed method to real systems is important to show the further effectiveness of the proposed method.

REFERENCES

[1] C. T. Abdallah and H. G. Tanner, "Complex networked control systems: Introduction to the special section," *IEEE Control Systems Magazine*, vol. 27, no. 4, pp. 30–32, 2007.

[2] P. Antsaklis and J. Baillieul, "Special issue on technology of networked control systems," *Proc. of the IEEE*, vol. 95, no. 1, pp. 5–8, 2007.

[3] R. Alur, A. D'Innocenzo, K. H. Johansson, G. J. Pappas, and G. Weiss, "Compositional modeling and analysis of multi-hop control networks," *IEEE Trans. on Automatic Control*, vol. 56, no. 10, pp. 2345–2357, 2011.

[4] B. Radunović and J.-Y. Le Boudec, "Rate performance objectives of multihop wireless networks," *IEEE Trans. on Mobile Computing*, vol. 3, no. 4, pp. 334–349, 2004.

[5] I. Saha, L. K. Sambasivan, S. K. Ghosh, and R. K. Patro, "Distributed fault-tolerant topology control in wireless multi-hop networks," *Wireless Networks*, vol. 16, no. 6, pp. 1511–1524, 2010.

[6] X. Cao, J. Chen, Y. Xiao, and Y. Sun, "Building-environment control with wireless sensor and actuator networks: Centralized versus distributed," *IEEE Trans. on Industrial Electronics*, vol. 57, no. 11, pp. 3596–3605, 2011.

[7] J. Chen, X. Cao, P. Cheng, Y. Xiao, and Y. Sun, "Distributed collaborative control for industrial automation with wireless sensor and actuator networks," *IEEE Trans. on Industrial Electronics*, vol. 57, no. 12, pp. 4219–4230, 2010.

[8] P. Gil, A. Paulo, L. Palma, A. Amâncio, and A. Cardoso, "Model Based Predictive Control over Wireless Sensor and Actuator Networks," *Proc. of the 37th Annual Conf. of the IEEE Industrial Electronic Society*, pp. 2600–2605, 2011.

[9] M. B. Kane, J. Scruggs, and J. P. Lynch, "Model-predictive control techniques for hydronic systems implemented on wireless sensor and actuator networks," *Proc. of the 2014 American Control Conf.*, pp. 3542–3547, 2014.

[10] S. A. Kumar and K. T. F. Simonsen, "Towards a model-based development approach for wireless sensor-actuator network protocols," *Proc. of the 4th ACM SIGBED Int'l Workshop on Design, Modeling, and Evaluation of Cyber-Physical Systems*, pp. 35–39, 2014.

[11] B. Li, Y. Ma, T. Westenbroek, C. Wu, H. Gonzalez, C. Lu, "Wireless routing and control: a cyber-physical case study," *Proc. of the ACM/IEEE 7th Int'l Conf. on Cyber-Physical Systems*, 2016.

[12] C. Lu, A. Saifullah, B. Li, M. Sha, H. Gonzalez, D. Gunatilaka, C. Wu, L. Nie, Y. Chen, "Real-time wireless sensor-actuator networks for industrial cyber-physical systems," *Proccedings of the IEEE*, vol. 104, no. 5, pp. 1013–1024, 2016.

[13] A. E.-D. Mady and G. Provan, "Co-design of wireless sensor-actuator networks for building controls," *Proc. of the IEEE Conf. on Decision and Control and European Control Conf.*, pp. 5266–5273, 2011.

[14] D. Martínez, F. Blanes, J. Simo, and A. Crespo, "Wireless sensor and actuator networks: Characterization and case study for confined spaces healthcare applications," *Proc. of the Int'l Multiconf. on Computer Science and Information Technology*, pp. 687–693, 2008.

[15] M. Mazo, Jr. and P. Tabuada, "Decentralized event-triggered control over wireless sensor/actuator networks," *IEEE Trans. on Automatic Control*, vol. 56, no. 10, pp. 2456–2461, 2011.

[16] H. Nakayama, Z. Md. Fadlullah, N. Ansari, and N. Kato, "A novel scheme for WSAN sink mobility based on clustering and set packing techniques," *IEEE Trans. on Automatic Control*, vol. 56, no. 10, pp. 2381–2389, 2011.

[17] R. Verdone, D. Dardari, G. Mazzini, and A. Conti, *Wireless Sensor and Actuator Networks: Technologies, Analysis and Design*, Academic Press, 2010.

[18] F. Xia, Y.-C. Tia, Y. Li, and Y. Sun, "Wireless sensor/actuator network design for mobile control applications," *Sensors*, vol. 7, no. 10, pp. 2157–2173, 2007.

[19] F. Xia, X. Kong, and Z. Xu, "Cyber-physical control over wireless sensor and actuator networks with packet loss," *Wireless Networking Based Control*, S. K. Mazumder Ed., Springer, 2011.

[20] L.-W. Yeh, C.-Y. Lu, C.-W. Kou, Y.-C. Tseng, and C.-W. Yi, "Autonomous light control by wireless sensor and actuator networks," *IEEE Sensor Journal*, vol. 10, no. 6, pp. 1029–1041. 2010.

[21] E. F. Camacho and C. B. Alba, *Model Predictive Control*, Springer, 2008.

[22] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, no. 3, pp. 407–427, 1999.

[23] S. Bououden, M. Chadli, L. Zhang, and T. Yang, "Constrained model predictive control for time-varying delay systems: Application to an active car suspension," *Int'l Journal of Control, Automation and Systems*, vol. 14, no. 1, pp. 51–58, 2016.

[24] S. Bououden, M. Chadli, H. R. Karimi, "A robust predictive control design for nonlinear active suspension systems," *Asian Journal of Control*, vol. 18, no. 1, pp. 122–132, 2016.

[25] D.Q. Mayne, M.M. Seron, and S.V. Raković, "Robust model predictive control of constrained linear systems with bounded disturbances," *Automatica*, vol. 41, no. 2, pp. 219–224, 2005.

[26] S. Bououden, M. Chadli, A. Fouad, and S. Filali, “A new approach for fuzzy predictive adaptive controller design using particle swarm optimization algorithm,” *Int’l Journal of Innovative Computing, Information & Control*, vol. 9, no. 9, pp. 3741–3758, 2013.

[27] S. Bououden, M. Chadli, H. R. Karimi, “An ant colony optimization-based fuzzy predictive control approach for nonlinear processes,” *Information Sciences*, vol. 299, pp. 143–158, 2015.

[28] Y. Xu, R. Lu, H. Peng, K. Xie, and A. Xue, “Asynchronous dissipative state estimation for stochastic complex networks with quantized jumping coupling and uncertain measurements,” *IEEE Trans. on Neural Networks and Learning Systems*, vol. 28, no. 2, pp. 268–277, 2017.

[29] Y. Xu, R. Lu, P. Shi, J. Tao, and S. Xie, “Robust estimation for neural networks with randomly occurring distributed delays and Markovian jump coupling,” *IEEE Trans. on Neural Networks and Learning Systems*, available online.

[30] S. Han, Z. Zhong, H. Li, G. Chen, E. Chan, and A. K. Mok, “Coding-aware multi-path routing in multi-hop wireless networks,” *Proc. of the IEEE Int’l Performance Computing and Communications Conf.* pp. 93–100, 2008.

[31] H. Okada, N. Nakagawa, T. Wada, T. Yamazato, and M. Katayama, “Multi-route coding in wireless multi-hop networks,” *IEICE Trans. on Communications*, vol. E89-B, pp. 1620–1626, 2006.

[32] H. Okada, T. Wada, K. Ohuchi, M. Saito, T. Yamazato, and M. Katayama, “Throughput evaluation of ARQ scheme for multi-route coding in wireless multi-hop networks,” *Proc. of the 2006 IEEE 63rd Vehicular Technology Conf.*, pp. 668–672, 2006.

[33] M. C. F. Donkers, W. P. M. H. Heemels, N. van de Wouw, and L. Hetel, “Stability analysis of networked control systems using a switched linear systems approach,” *IEEE Trans. on Automatic Control*, vol. 56, no. 9, pp. 2101–2115, 2011.

[34] K. Kobayashi and K. Hiraishi, “Optimal control of multihop control networks based on the MLD framework,” *IEEJ Trans. on Electrical and Electronic Engineering*, vol. 10, no. 6, pp. 699-705, 2015.

[35] I. F. Akyildiz, X. Wang, and W. Wang, “Wireless mesh networks: a survey,” *Computer Networks*, vol. 47, no. 4, pp. 445–487, 2005.

[36] W. P. M. H. Heemels, K. H. Johansson, and P. Tabuada, “An introduction to event-triggered and self-triggered control,” *Proc. of the 51st IEEE Conf. on Decision and Control*, pp. 3270–3285, 2012.

Dai Satoh received the B.E. degree in 2016 from Hokkaido University. Since 2016, he has been pursuing the M.S. degree with the Graduate School of Information Science and Technology, Hokkaido University. His research interests include networked control systems.

Koichi Kobayashi received the B.E. degree in 1998 and the M.E. degree in 2000 from Hosei University, and the D.E. degree in 2007 from Tokyo Institute of Technology. From 2000 to 2004, he worked at Nippon Steel Corporation. From 2007 to 2015, he was an Assistant Professor at Japan Advanced Institute of Science and Technology. Since 2015, he has been an Associate Professor at the Graduate School of Information Science and Technology, Hokkaido University. His research interests include analysis and control of discrete event and hybrid systems. He is a member of the IEEE, IEEJ, IEICE, ISCIE, and SICE.

Yuh Yamashita received his B.S., M.S., and Ph.D. degrees from Hokkaido University, Japan, in 1984, 1986, and 1993, respectively. In 1988, he joined the faculty of Hokkaido University. From 1996 to 2004, he was an Associate Professor at the Nara Institute of Science and Technology, Japan. Since 2004, he has been a Professor of the Graduate School of Information Science and Technology, Hokkaido University. His research interests include nonlinear control and nonlinear dynamical systems. He is a member of SICE, ISCIE, IEICE, SCEJ, and IEEE.