Fast Sensor Scheduling with Communication Costs for Sensor Networks

Shogo Arai, Yasushi Iwatani, and Koichi Hashimoto

Abstract—This paper is devoted to sensor scheduling for a class of sensor networks whose sensors are spatially distributed and measurements are influenced by state dependent noise. Sensor scheduling is required to achieve power saving since each sensor operates with a battery power source. The sensor scheduling problem is formulated as model predictive control which minimizes a quadratic cost function with communication costs, since communications among sensors take much power. A fast and optimal sensor scheduling algorithm is proposed for a class of sensor networks. From a theoretical standpoint, we present that computation time of the proposed algorithm increases exponentially with the number of sensor types, while that of standard algorithms is exponential in the number of the sensors. The proposed algorithm is faster than standard one, since the number of sensor types is always less than or equal to the number of sensors.

I. INTRODUCTION

A sensor network is a collection of spatially distributed sensors that communicate with each other. Applications of sensor networks include habitat monitoring, animal tracking, forest-fire detection, precision farming, and disaster relief applications [16], [18]. Sensor networks have been also implemented in control systems such as robot control systems [4], [12] and target tracking systems [10], [22]. Sensors in a sensor network are connected wirelessly, and each sensor operates with a battery power source. It is therefore desired for each sensor to prolong the battery life, or equivalently, to achieve power saving [11]. It is important to restrict communications of sensors for power saving since communications take much power. One of approaches to solve the problem is to select available sensors dynamically [2], [14]. This process is called sensor scheduling.

One of the major problems on sensor scheduling is to reduce computation time, since the number of possible sensor sequences increases exponentially with the number of the sensors. In particular, a predictive control method [5], a branch and bound method [7], a stochastic scheduling algorithm [9], and a sub-optimal method based on relaxed dynamic programming [1] have been proposed for sensor scheduling. In addition, a sensor scheduling strategy for continuous-time systems has been provided in [13]. These approaches assume that sensor characteristics are different from each other, that is, each sensor observes a different state or the covariance matrices for the sensor model are different from each other.

The existing approaches cannot be applied to sensor scheduling for sensor networks for the following two reasons. First, a sensor network usually consists of a few types of sensors [18]. In other words, many sensors in a sensor network have the same characteristics, while the existing approaches assume that sensors have different characteristics as mentioned before. Thus the existing approaches cannot provide a reasonable solution for the sensor scheduling problems for sensor networks. Second, sensors in a sensor network are spatially distributed. Each measurement noise may depend on the position of a target relative to the position of the sensor. In particular, measurements taken by cameras or radar sensors are influenced by state dependent noise [6], [15], [17]. The existing works have not provided any optimal sensor scheduling algorithm for systems with state dependent noise.

To solve the problems, we have proposed a fast and optimal sensor scheduling algorithm for sensor networks whose measurements are influenced by state dependent noise and which have various kind of sensor nodes [3]. The sensor scheduling problem has been formulated as a model predictive control problem with single sensor measurement per time. The proposed scheduling algorithm minimizes a given quadratic cost function at each time, where the concept of sensor types has been introduced to classify sensor characteristics. From a theoretical standpoint, we have shown that its computation time increases exponentially with the number of sensor types, while the computation time of a primitive algorithm is exponential in the number of sensors. Thus computational cost of the proposed algorithm is expected to be much less than a primitive one since a networked control system usually consists of a few sensors types. However, the proposed algorithm does not consider any communication costs. Thus, it cannot provide appropriate sensor scheduling when communication costs are considered.

This paper is devoted to sensor scheduling with communication costs and observational ranges. An optimal scheduling algorithm is proposed for a class of sensor networks. Computation time of the proposed algorithm increases exponentially with the number of the sensor types, while a primitive one is represented by an exponential function whose base is the number of sensors and whose exponent is twice the length of the prediction horizon. Thus it is guaranteed that computation time of our algorithm is less or equal to that of a primitive one.

This paper uses the following notation. The Kronecker delta is denoted by $\delta_{m\ell}$. The expectation operator is represented by $E[\cdot]$. For a square matrix $A$, $\text{tr}[A]$ stands for the trace of $A$. 

S. Arai and K. Hashimoto are with the Department of System Information Sciences, Tohoku University, Aramaki Aza Aoba 6-6-01, Sendai, Japan (e-mail: arai@ic.is.tohoku.ac.jp; koichi@ic.is.tohoku.ac.jp).

Yasushi Iwatani is with the Graduate School of Science and Technology, Hirosaki University, Bunkyo 3, Hirosaki, Aomori, Japan (e-mail: iwatani@cc.hirosaki-u.ac.jp).
II. SYSTEM DESCRIPTION

This paper considers a class of systems with \( N \) sensors as illustrated in Fig. 1. The sensors are labeled from 1 to \( N \). Each sensor can communicate with some other sensors. To simplify notations, it is assumed that one sensor is available for observation at each time. However, our algorithm presented in this paper is easily extended for the systems in which multiple sensors are available at each time, when all possible combinations of sensors are labeled. The observation data is transmitted over the sensor network, and one of the sensors sends the data to the controller. Details of the network are described in the next section.

A. Sensor Model

Let us first describe the sensor model considered in this paper. The sensor model is of the form:

\[
y_i(k) = C_i x_p(k) + \sum_{\ell=1}^{q_i} d_{i\ell}(x(k)) v_{i\ell}(k),
\]

where \( y_i(k) \in \mathbb{R}^p \) is the measurement taken by sensor \( i \), \( x_p(k) \in \mathbb{R}^n \) is the state of the controlled object, \( d_{i\ell} \) is a vector function that is differentiable, \( x \) is defined by

\[
x(k) = \left[ x_p^T(k) \ x_c^T(k) \right]^T,
\]

and \( x_c \) is the state of the controller. The noise \( v_i(k) = [v_{i1}(k), \ldots, v_{iq_i}(k)]^T \) is white, Gaussian and zero mean with a covariance matrix

\[
E[v_i(k)v_i^T(\tau)] = V_i \delta_{k\tau}.
\]

The vector function \( d_{i\ell}(x) \) is a function of \( x \) not of only \( x_p \). This helps to develop a camera model presented in Example 2. Note that state-dependent sensor scheduling is required due to the function \( d_{i\ell}(x) \) even when \( C_i = C_\ell \) and \( V_i = V_\ell \) hold for all pairs of \( (i, \ell) \). The sensor model (1) is more general than models considered in other papers. For example, \cite{7} considers a class of systems described by

\[
y_i(k) = C_i x(k) + D_i v_i(k)
\]

where \( C_i \) and \( D_i \) are constant matrices. Our models (1) and (9) presented in Section II-B include the above class. In addition, our approach is available for systems with both sensor models (1) and (4).

The following examples show several sensors whose mathematical models can be written by (1).

Example 1: Consider radar sensors that measure the position of a target in the \((x, y)\) plane. Two standard models have been developed for radar sensors. One is given by

\[
y_i = \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} \cos \theta_i - \sin \theta_i \\ \sin \theta_i \cos \theta_i \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & r_i \end{bmatrix} v_i
\]

\cite{17}, and the other is given by

\[
y_i = \begin{bmatrix} x \\ y \end{bmatrix} + a_{i}(r_i) \begin{bmatrix} \cos \theta_i - \sin \theta_i \\ \sin \theta_i \cos \theta_i \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & b \end{bmatrix} v_i
\]

\cite{6}, where \( y_i \) is the measurement, \( \theta_i \) the angle between the \( x \) axis and the vector joining sensor \( i \) to the target, \( r_i \) the distance from sensor \( i \) to the target, \( a_i(r_i) \) a function such that \( a_i^2(r_i) \) is a quadratic function of \( r_i \), \( b \) a constant, and \( v_i \) a white, Gaussian and zero mean noise (see Fig. 2). It is clear that (5) and (6) are described by (1).

Example 2 (\cite{13}): Let a target and two cameras be set at \((x, y), (p_{x1}, p_{y1})\) and \((p_{x2}, p_{y2})\), respectively, as illustrated in Fig. 3. The optical axes of cameras \( \ell \) and \( m \) are parallel to the \( y \) axis and the \( x \) axis, respectively. The combination of cameras \( \ell \) and \( m \) is labeled by \( i \). Then the camera model is represented by

\[
y_i = \begin{bmatrix} x \\ y \end{bmatrix} + \frac{1}{f} \begin{bmatrix} y_c - p_{ty} \\ x_c - p_{tx} \end{bmatrix} v_i,
\]

where \( f \) is the focal length of the cameras and \((x_c, y_c)\) is an estimate of \((x, y)\). Equation (7) is represented by (1), since \( d_{i\ell} \) is a function of \( x \).

B. Sensor Type

The concept of sensor types is a key to developing a fast sensor scheduling algorithm. A function \( \sigma \) is defined as a surjection from \( \{1, 2, \ldots, N\} := \mathbb{N} \) onto \( \{1, 2, \ldots, M\} := \mathbb{M} \) for some \( M \) that satisfies

\[
C_i = C_\ell \text{ and } V_i = V_\ell \Rightarrow \sigma(i) = \sigma(\ell).
\]

Each value of \( \sigma \) represents a sensor type. In other words, a set of sensors with the same measurement matrix and the same covariance matrix constitutes a type of sensors. The number of sensor types is always less than or equal to the number of
sensors, that is, $M \leq N$. A sensor network usually consists of a few sensor types [18]. Thus $M$ is far less than $N$ for a class of sensor networks. The concept of sensor types is effective for our scheduling algorithm proposed in Section IV-B. In order to clarify the sensor types, we rewrite $C_t$ and $V_t$ as $C_{\sigma(i)}$ and $V_{\sigma(i)}$, respectively. Hence, we have

$$y_t(k) = C_{\sigma(i)} x_p(k) + \sum_{\ell=1}^{q_{\sigma(i)}} d_{\ell \ell}(x(k)) v_{\ell \ell}(k),$$

$$E[w_t(k)w_t^\top(\tau)] = V_{\sigma(i)} \delta_{t \tau}.$$  

The sensor type $\sigma(i)$ is sometimes simply written by $\sigma$.

C. Closed Loop System

The controlled object is represented as a discrete-time linear time-invariant system

$$x_p(k+1) = A_p x_p(k) + B_p u(k) + w(k),$$

where $x_p(k) \in \mathbb{R}^{n_p}$ is the state vector, $u(k) \in \mathbb{R}^r$ the control input, and $w(k)$ the process noise. The noise $w(k)$ is white, Gaussian and zero mean with a covariance matrix $W$. The time index $k$ is sometimes omitted to simplify notation. The initial state $x_p(0)$ is a random variable whose expectation value and covariance matrix are known constants.

The controller is given by

$$x_c(k+1) = A_c x_c(k) + B_c y_c(k) + B_1 u(k),$$

$$u(k) = C_c x_c(k) + D_c y_c(k).$$

The closed loop system described by (9)–(13) is of the form:

$$x(k+1) = A_x x(k) + B_x \sum_{\ell=1}^{q_x} d_{\ell \ell}(x) v_{\ell \ell}(k) + \begin{bmatrix} w(k) \\ 0 \end{bmatrix},$$

where

$$A_x = \begin{bmatrix} A_p + B_p D_{c \sigma} C_{c} & B_c C_{c} \\ B_1 C_{c} + B_2 C_{c} D_{c} C_{c} & A_{c} + B_2 C_{c} D_{c} C_{c} \end{bmatrix},$$

$$B_x = \begin{bmatrix} B_p D_{c} \\ B_1 C_{c} + B_2 C_{c} D_{c} \end{bmatrix}.$$  

We also define $n := n_p + n_c$ and $\bar{x}_0 := E[x(0)]$.

III. SENSOR SCHEDULING PROBLEM

The network considered in this paper works as follows. Sensor $i(k)$ is used for observation at time $k$. The observation data is transmitted over the network, and sensor $j(k)$ sends it to the controller. Sensors available for observation are restricted. The set of available sensors for observation at time $k$ is denoted by $I_c(k)$. Sensors that can communicate with the controller are also restricted, and the set of such sensors at time $k$ is denoted by $J_c(k)$. It is assumed that an optimal path from sensor $i$ to sensor $j$ is known for all pairs of $i$ and $j$. The communication cost between sensors $i$ and $j$ is denoted by $\gamma_{ij} \geq 0$. The communication cost is given by $\gamma_{ij} = \infty$, when sensors $i$ and $j$ are disconnected.

The sensor scheduling problem is formulated as follows.  

Problem 1: Find optimal pairs of $\iota^*(k) = (i^*(k), j^*(k))$ such that

$$\{\iota^*(0), \iota^*(1), \cdots, \iota^*(T)\} = \arg \min_{\iota(0) \in \mathcal{K}(0), \cdots, \iota(T) \in \mathcal{K}(T)} J_c,$$

where

$$J_c = E\left[ \sum_{k=0}^{T} \{ x_p^\top(k) Q_p x_p(k) + u^\top(k) R u(k) \\ + (x(k) - m_{j(k)})^\top(k) \Gamma (x(k) - m_{j(k)}) \} \\ + x_p^\top(T+1) P x_p(T+1) \right] + \sum_{k=0}^{T} \gamma(k)(k),$$

where $T$ is a positive integer, $Q_p \in \mathbb{R}^{n_p \times n_p}$, $R \in \mathbb{R}^{r \times r}$ and $\Pi \in \mathbb{R}^{n \times n}$ are positive definite symmetric matrices and $\Gamma \in \mathbb{R}^{n \times n}$ is a semi-positive definite symmetric matrix and $m_{j} \in \mathbb{R}^n$ is a constant vector.

The cost $J_c$ includes a communication cost between the controller and sensor $j$ represented by the third term in the right hand side of (19). If there exist no communication costs, then $\Gamma = 0$ and $\gamma_{ij} = 0$ for all $i$ and $j$. In this case, we obtain a finite-time quadratic cost function

$$J = E\left[ \sum_{k=0}^{T} \{ x_p^\top(k) Q_p x_p(k) + u^\top(k) R u(k) \} \\ + x_p^\top(T+1) P x_p(T+1) \right].$$

In our previous paper [3], a fast sensor scheduling algorithm that minimizes $J$ has been derived on the assumption that $i(k) = j(k)$ and $I_c(k) = J_c(k) = \mathbb{N}$ for all $k$.

This paper assumes that model predictive control is implemented. Problem 1 is solved at each time. Thus a fast algorithm for solving Problem 1 is desired. One primitive method to solve Problem 1 is as follows: Calculate values of the cost function for all possible sequences of pairs of sensors from time 0 to $T$ and compare these values. This is called the exhaustive search in this paper. The exhaustive search requires $N^{2(T+1)}$ sensor sequences to determine the optimal solution, when $I_c(k) = \mathbb{N}$ and $J_c(k) = \mathbb{N}$ for all $k$. The exhaustive search is not suitable for model predictive control from the point of view of computation time.

When model predictive control is applied, $E[x(k)]$ is required to solve Problem 1 at time $k$. Therefore we have to estimate $E[x_p(k)]$. We use (12) as an observer for estimation in a numerical example presented in this paper. Then $x_c$ converges to $E[x_p]$ when $\sigma$ is fixed and all the eigenvalues of $A_x$ are in the open unit disk, since

$$E[x(k+1)] = A_x E[x(k)].$$

IV. FAST SENSOR SCHEDULING

A. Fast Sensor Selection in a Given Sequence of Sensor Types

The key idea to obtain a fast sensor scheduling algorithm is to separate the original sensor scheduling problem into
two scheduling problems: Scheduling of sensor types and scheduling of sensors in a given sequence of sensor types. The later is formulated as follows.

**Problem 2:** Suppose that a sensor type $\sigma_k$ is selected for observation and a sensor type $\rho_k$ is chosen for communication with the controller at time $k \in \{0, 1, \cdots, T\}$. Find

$$\{\ell_1^*, \ell_2^*, \cdots, \ell_T^*\} = \arg \min_{\ell_k^*(0) \in \mathbb{K}_k, \cdots, \ell_k^*(T) \in \mathbb{K}_k} J_c$$

where

$$\mathbb{K}_k = \{1 \cdots 1\} \times \{1 \cdots 1\}$$

This section proposes a fast sensor selection method that gives $\ell_1^*(k)$ in (22). It is assumed throughout this paper that one of the above two holds to derive theoretical results.

1) There exist constant matrices $S_{\sigma \ell} \in \mathbb{R}^{p \times \kappa}$ and $s_{\ell i} \in \mathbb{R}^{p \times \sigma}$ such that

$$d_{\ell i}(x) = S_{\sigma \ell} x + s_{\ell i},$$

$$\forall i \in \{1, 2, \cdots, N\}, \forall \ell \in \{1, 2, \cdots, q_{\sigma}\}. \quad (26)$$

2) There exist constant matrices $S_{\ell i} \in \mathbb{R}^{p \times \kappa}$ and $s_{\ell i} \in \mathbb{R}^{p \times \sigma}$ such that

$$d_{\ell i}(x)d_{\ell m}^T = s_{\ell i} s_{\ell m}^T + S_{\ell i}(x - x_0)s_{\ell m}^T,$$

$$\forall i \in \{1, 2, \cdots, N\}, \forall \ell, m \in \{1, 2, \cdots, q_{\sigma}\}. \quad (27)$$

Roughly speaking, (28) implies that the magnitude of the sensor noise is proportional to the distance between the controlled object and sensor $i$, and the covariance matrices $S_{\sigma \ell}$ do not depend directly on sensor $i$. The sensor model (7) satisfies (28), although (28) does not hold for (5) or (6).

We can derive all the results that will be provided in the paper when we replace (26) with the following:

$$d_{\ell i}(x)d_{\ell m}^T(x) = \sum_{j=1}^{Si_{\ell m}} (S_{\sigma \ell j} x + s_{\ell j} (S_{\sigma \ell j} x + s_{i_{\ell m}}))^T,$$

$$\forall i \in \{1, 2, \cdots, N\}, \forall \ell, m \in \{1, 2, \cdots, q_{\sigma}\}. \quad (28)$$

Note that (26) is not equivalent to (28). For example, the sensor model (9) with $q_{\sigma} = 1$ and $d_{i1} = \sqrt{(x - p_{i1})^2 + (y - p_{i1})^2}$ satisfies (28) but not (26).

The second assumption (27) is valid for all systems in a neighborhood of $x = \bar{x}_0$. In fact, we obtain (27) when we set

$$s_{\ell i} = d_{\ell i}(\bar{x}_0), \quad S_{\ell i} = \left. \frac{\partial d_{\ell i}}{\partial x} \right|_{x = \bar{x}_0}$$

and use a linear approximation of $d_{\ell i}(x)d_{\ell m}^T(x)$ around $\bar{x}_0$. Hence the proposed sensor scheduling algorithm with the linear approximation of the sensor model is always available if $d_{\ell i}$ is differentiable. The following theorem gives a solution for Problem 2.

**Theorem 1:** Suppose that sensor types $\sigma_k$ and $\rho_k$ are selected at time $k \in \{0, 1, \cdots, T\}$ for observation and communication with the controller, respectively.

1) If (26) holds, then the optimal pair of sensors $\ell_1^*(k)$ satisfies

$$\ell_1^*(k) = \arg \min_{\ell_k^*(k) \in \mathbb{K}_k} \{ \text{tr} [P(k) \Psi_i(k)] + \text{tr} [\Phi_i(k)] \} + g_j(k) + \gamma_1 \gamma_2,$$  \quad (30)

where

$$g_j(k) = m_j^T \Gamma m_j - m_j^T \Gamma A(k) \bar{x}_0 - (m_j^T \Gamma A(k) \bar{x}_0)^T,$$  \quad (31)

$$P(T + 1) = \begin{bmatrix} \Pi & 0 \\ 0 & 0 \end{bmatrix},$$  \quad (32)

$$P(k) = Q_{\sigma_k} + A_{\sigma_k}^T P(k + 1) A_{\sigma_k} + \sum_{\ell = 1}^{q_{\sigma_k}} \sum_{m = 1}^{q_{\sigma_k}} V_{\sigma_k \ell m} S_{\sigma_k \ell m}^T B_{\sigma_k \ell}^T P(k + 1) B_{\sigma_k \ell} S_{\sigma_k \ell},$$  \quad (33)

$$\Psi_i(k) = \sum_{\ell = 1}^{q_{\sigma_k}} \sum_{m = 1}^{q_{\sigma_k}} V_{\sigma_k \ell m} B_{\sigma_k \ell} (S_{\sigma_k \ell} A(k) \bar{x}_0 s_{\ell m})^T + s_{\ell i} (S_{\sigma_k \ell m} A(k) \bar{x}_0)^T + s_{\ell i} s_{\ell m}^T B_{\sigma_k \ell}^T + \Omega,$$  \quad (34)

$$\Phi_i(k) = D_{\sigma_k \ell} R_{\sigma_k \ell} \sum_{\ell = 1}^{q_{\sigma_k}} \sum_{m = 1}^{q_{\sigma_k}} V_{\sigma_k \ell m} S_{\sigma_k \ell m}^T B_{\sigma_k \ell} + \Omega,$$  \quad (35)

$$Q_{\sigma_k} = \begin{bmatrix} Q_p & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} C_{\sigma_k \ell}^T D_{\sigma_k \ell}^T \\ C_{\sigma_k \ell} \end{bmatrix} R \begin{bmatrix} D_{\sigma_k \ell} C_{\sigma_k \ell} \end{bmatrix} + \sum_{\ell = 1}^{q_{\sigma_k}} \sum_{m = 1}^{q_{\sigma_k}} V_{\sigma_k \ell m} s_{\sigma_k \ell m} D_{\sigma_k \ell} R D_{\sigma_k \ell} S_{\sigma_k \ell} + \Omega,$$  \quad (36)

$$\Omega = \begin{bmatrix} W & 0 \\ 0 & 0 \end{bmatrix},$$  \quad (37)

$$A(k) = A_{\sigma_k \ell} A_{\sigma_k \ell},$$  \quad (38)

with $A(0) = I$ and $V_{\sigma_k \ell m}$ is the $(\ell, m)$ element of $V_{\sigma_k}$.

2) Suppose that (27) is satisfied. Then (30) is also true when $P, \Psi_i, \Phi_i$ and $Q_{\sigma_k}$ are replaced with

$$P(k) = Q_{\sigma_k} + A_{\sigma_k}^T P(k + 1) A_{\sigma_k},$$  \quad (39)

$$\Psi_i(k) = \sum_{\ell = 1}^{q_{\sigma_k}} \sum_{m = 1}^{q_{\sigma_k}} V_{\sigma_k \ell m} B_{\sigma_k \ell} (S_{\ell i} (A(k) - I) \bar{x}_0 s_{\ell m}^T + s_{\ell i} s_{\ell m}^T) B_{\sigma_k \ell}^T + \Omega,$$  \quad (40)
\[ \Phi_i(k) = D_{c\sigma}^T R D_{c\sigma} \sum_{\ell=1}^{q_{\sigma}} \sum_{m=1}^{V_{\sigma}^{\ell m}} \left( S_{\ell m}(A(k) - I) \bar{x}_0 s_{\ell m}^T + S_{\ell m}(A(k) - I) \bar{x}_0 + s_{\ell m} s_{\ell m}^T \right), \]

Furthermore, Step 3 compares \( M^{2(T+1)} \) values to obtain \( m^* \). Thus we obtain the following theorem.

**Theorem 2:** Assume \( I_{c}(k) = \mathbb{N} \) for all \( k \). Then the computational cost of the proposed algorithm is given by \( O(TN^2M^{2T}) \).

Recall that the exhaustive search requires \( N^{2(T+1)} \) sensor sequences to determine the optimal sensor sequence. Thus the proposed algorithm is effective for fast sensor scheduling for sensor networks such that the number of sensor types, \( M \), is far less than the number of sensors, \( N \).

**Example 3:** Consider a vehicle that travels on the two-dimensional plane. Let \((x, y)\) be the position of the vehicle. The goal here is to control the vehicle using a sensor network which consists of forty nine radar sensors. No controller is mounted on the vehicle. One of the sensors has a controller, and the input signal is sent over the network\(^1\).

The state equation of the vehicle in continuous time is given by

\[ \dot{x}_p = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} x_p + \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \bar{u} + \bar{w}, \]  

where \( \bar{x} := [x \ y \ \dot{x} \ \dot{y}]^T \in \mathbb{R}^4 \) and \( \bar{u} \in \mathbb{R}^2 \). The covariance matrix of \( \bar{w} \) is set to \( 0.001I_4 \). The state equation (46) is discretized with a sampling period 0.1. An observer (12) and a controller (13) are designed such that the poles of (14) are set to \( 0.91 \pm 0.055j, 0.92 \pm 0.030j, 0.86 \pm 0.091j \) and \( 0.86 \pm 0.091j \), where \( j \) denotes the imaginary unit.

Radar sensors 1, 2, \ldots, 49 with model (5) are set at

\( (0, 0), (0, 0.25), \ldots, (0, 1.5), (0.25, 0), \ldots, (1.5, 1.5), \) as illustrated in Fig. 4. The covariance matrices are given by

\[ V_i = \begin{bmatrix} 0.002 & 0 \\ 0 & 0.01 \end{bmatrix}, \]

for all \( i \in \{1, 2, \ldots, 49\} \). In this example, the number of sensors is forty nine, and the number of sensor types is one.

It is assumed that each sensor can communicate with four neighborhood sensors, and the communication cost of the network is set to

\[ \gamma_{ij} = 0.4(\|p_{ix} - p_{jx}\| + \|p_{iy} - p_{jy}\|), \]

where \( (p_{ix}, p_{iy}) \) denotes the position of sensor \( i \). The observation range of the sensors is defined by

\[ \left\| \begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} p_{ix} \\ p_{iy} \end{bmatrix} \right\| \leq 0.4, \]

for all \( i \), and \( \mathbb{I}_c(k) = \{i \mid (49) \text{ holds}\} \). The controller is set at \( (0.75, 0) \), and \( j(k) = 22 \), or equivalently, \( \mathbb{I}_c(k) = \{22\} \) for all \( k \). Other parameters are set to

\[ Q_p = I_4, \quad R = I_2, \quad \Pi = I_4, \quad \Gamma = 0, \quad T = 4. \]

\(^1\)In a networked control system, the controller and the controlled object are sometimes connected through a network [21].
The radar sensor model is linearized by (29), so that the proposed algorithm can be applied. The left of Fig. 4 shows a sample path of the vehicle for
\[
\begin{align*}
x_p(0) &= \begin{bmatrix} 1.21 & 1.21 & 0.01 & 0.01 \end{bmatrix}^T, \\
x_c(0) &= \begin{bmatrix} 1.2 & 1.2 & 0 & 0 \end{bmatrix}^T,
\end{align*}
\]
and \(E[x_p(0)] = x_c(0)\). The right of Fig. 4 illustrates the suboptimal sensor sequences with the communication costs and without the communication costs, that is, \(\gamma_{ij} = 0\) and \(\Gamma = 0\). The suboptimal sensor sequences are given by \(41 \rightarrow 33 \rightarrow 25 \rightarrow 17 \rightarrow 16 \rightarrow 15 \rightarrow 9 \rightarrow 15 \rightarrow 8\), with the communication costs, and \(41 \rightarrow 40 \rightarrow 33 \rightarrow 32 \rightarrow 25 \rightarrow 17 \rightarrow 10 \rightarrow 9 \rightarrow 8 \rightarrow 1\), without the communication costs. Each selected sensor in scheduling with the communication costs is equal to or closer to the controller than one without the communication costs to reduce the communication costs.

The proposed method needs to compare at most 12 values of the cost function to obtain \(i^*(0)\) at each time, and it takes \(7.3 \times 10^{-3} \) [s] to get \(i^*(0)\) on average. On the other hand, the exhaustive search method takes more than one hour to get \(i^*(0)\) where Monte Carlo method is implemented to compute the values of the cost function. The proposed method is useful for large-scale sensor networks. The programs ran in MATLAB 7.1 on a PentiumD 3.2 GHz PC with 2 GB of RAM.

V. Conclusion

In this paper, a sensor scheduling problem with communication costs for a class of sensor networks whose measurements are influenced by state dependent noise was addressed. The sensor scheduling problem was formulated as a model predictive control problem, and we proposed a fast and optimal sensor scheduling algorithm for a class of sensor networks. The computation time of the proposed algorithm increases exponentially with the number of sensor types, while the computation time of a primitive algorithm is represented by an exponential function whose base is the number of sensors and whose exponent is twice the length of the prediction horizon. A numerical example demonstrated that the proposed algorithm is effective for large-scale sensor networks.