31st Congress of the International Council
of the Aeronautical Sciences

Belo Horizonte, Brazil; September 09-14, 2018

PERFORMANCE ANALYSIS OF UAV PERSISTENT SURVEILLANCE ALGORITHM

Suwon Lee*, Youdan Kim*
***School of Mechanical and Aerospace Engineering**
Seoul National University
Seoul 08826, Korea

Keywords: *Persistent Coverage Control, Performance Prediction, Multi-agent System, Distributed System, Sensor Network*

Abstract

In this paper, persistent coverage control problem in two-dimensional space is formulated. The monitoring space is discretized into cells with time-increasing ages. Gradient-based and target based persistent coverage controller are designed. A probabilistic prediction model is proposed to predict coverage performance of the designed controllers. Numerical simulation is performed to demonstrate the predicted performance of persistent coverage control algorithms.

1 Introduction

Multi-agent system has been widely studied for decades. Especially, coverage control is one of the important problems in the multi-agent system. The multi-agent coverage control problem can be formulated as follows; allocate sensor network at proper locations that maximize the sensing quality while minimizing the communication costs and control efforts. The sensor network (sensing agents) are controlled to the gradient direction or by the optimal control, which make them converge to certain locations in given space to be monitored [1],[2]. In this sense, the coverage control problem is often called as a static coverage problem.

Meanwhile, for the mission space that is too broad to be fully covered by a static distribution of static sensor network, the sensing agents have to continuously move around the mission space to monitor the whole area. Besides, the characteristics of the mission space may be changing (dynamic). Those types of coverage problems are dynamic coverage problem. The objective of the dynamic coverage problem is to make a one-time sweep of the monitoring space as well as to estimate the dynamic properties of the mission space [3]. When the information of the mission space is not given to the sensing agents beforehand, the problem is called as the exploration problem. In the exploration problem, obstacles in the mission space are typically considered.

In the persistent coverage control problem, the mission space should be monitored persistently [4]. The time-increasing uncertainties are defined and distributed in the mission space that the sensing agents should suppress the uncertainties, while continuously patrolling the mission space. The persistent coverage control problem can be considered as the expansion of the dynamic coverage problem that sweeps the whole mission space repeatedly [5]. Note that it is not necessary for every persistent coverage controller to repeate the same sequence.

The persistent coverage control problem usually discretizes the two-dimensional mission space into cells and allocate time-increasing uncertainty level or age to each cell. The uncertainty can be represented in dynamic equations and combined with the detection model of sensing agents, and optimal control problem is often formulated [6],[7]. A local gradient-based controller was also developed for the persistent coverage problem. Though some persistent coverage control approaches have been developed, the prediction of performance has not been studied. Only the optimal coverage performance in a limited problem formulation was examined [8].

In this study, persistent coverage control problem is formulated, and the performance function representing the monitoring level of the mission space is defined. A prediction model that predicts the performance of persistent coverage algorithm is proposed using probabilistic approach. The prediction model considers binary detection model of sensors and completeness of the persistent coverage algorithm. That is, it is assumed that every cell in the mission space should be detected evenly. Two types of distributed persistent coverage control algorithms are designed, and their coverage performances are compared with the predicted performance obtained from the proposed prediction model.

This paper is organized as follows. In Section 2, the preliminaries includes agent model, mission space description and performance function. In Section 3, the performance prediction model is explained. Target-based and gradient-based persistent coverage algorithms are designed in Section 4, and numerical simulations are carried out in Section 5. Section 6 presents the concluding remarks.

2 Preliminaries

In this section, dynamic model of the agent and mission space discretization are defined.

2.1 Agent and Mission Space Model

Consider a dynamic model of the agent.

$$\begin{aligned} \dot{x}_j(t) &= \mu \cos \theta_j \\ \dot{y}_j(t) &= \mu \sin \theta_j \\ \dot{\theta}_j(t) &= u_j \end{aligned} \tag{1}$$

where $P_j = [x_j, y_j]^T$ is the position of agent $j \in \{1, \cdots, M\}$, and θ_j is its heading angle. The heading angle is controlled by control input u_j, and the speed of each agent μ is assumed to be constant.

Monitoring space, $\Omega = [0,1]^2 \subset \mathbb{R}^2$, is discretized into N number of square cells $c_i, i \in \{1, \cdots, N = n^2\}$, and each cell has its age value denoted as a_i. The age dynamics of cell can be represented as follows,

$$\dot{a}_i(t) = \begin{cases} 1 & \text{if } c_i \notin \mathcal{D} \\ 0 & \text{Otherwise} \end{cases} \tag{2}$$

where $\mathcal{D} = \cup_{j=1}^{M} \mathcal{D}_j$ is the set of cells inside the sensing radii of some agent. For the agent j, $\mathcal{D}_i$ is defined as follows,

$$\mathcal{D}_j = \{c_i | d(P_{c_i}, P_j) \leq r_j\}, \quad i \in \Omega \tag{3}$$

where P_{c_i} and r_j are the position of cell i and sensing radius of agent j, respectively, and $d(\cdot,\cdot)$ is the Euclidean distance operator, i.e., $d(a,b) = \sqrt{(a-b)^T(a-b)}$.

The age of cell c_i is initialized to zero value when $c_i \in \mathcal{D}$. The binary detection model may be more reasonable that other than probabilistic detection models, because sensors usually acquire information very faster than the uncertainty or agent dynamics. That is, the age of certain area increases slower than the speed of sensor's information acquisition.

In the multi-agent situation, the Voronoi tessellation and the Delaunay graph are widely used for coverage control with distributed communication models. The mission space is divided into Voronoi regions $V = \{V_1, \cdots, V_M\}$ as

$$V_j = \{s \in \Omega | d(s, p_j) \leq d(s, p_k), \forall k \neq j\} \tag{4}$$

On the other hand, Delaunay triangulation $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is a dual graph of the Voronoi diagram. The position of each agent is vertices $\mathcal{V} = \{1, \cdots, M\}$, and $\mathcal{E} \subset [\mathcal{V} \times \mathcal{V}]$ is the set of edges. Two vertices are connected and become an edge when they are in the adjacent Voronoi regions, respectively.

Each agent predicts the age of each cell represented by an age vector $\mathcal{A}_j = [\hat{a}_1^j, \cdots, \hat{a}_N^j]^T \in \mathbb{R}^N$, where $\hat{a}_i^j$ is the predicted age of cell i by agent j. The age vector $\mathcal{A}_j$ is predicted based on the age dynamics (1), where element is initialized to zero when the corresponding cell is detected by the agent j. Note that the real age vector $\mathcal{A} = [a_1, \cdots, a_N]^T$ may differ from the predicted age vectors because only the initialization by agent j itself is considered in $\mathcal{A}_j$. When the agent pair $(j,k) = (k,j) \in \mathcal{E}$ is the edge of the Delaunay graph G, agents j, and k share their local information and update the predicted ages to the recent one.

2.2 Performance Function

Performance function $C(t)$ is defined to evaluate the performance of the persistent coverage algorithm at time t as follows,

$$C(t) = 1 - \frac{\sum_{i=1}^{N} \tanh(\rho a_i(t))}{N} \tag{5}$$

where $\rho \in \mathbb{R}$ is a positive scaling constant. The performance function $C(t) \in [0,1]$ is designed to indicate the monitoring level of the entire mission space. To do this, the hyperbolic tangent function is used to bound the age and limit the individual cell's influence on the performance value.

If there does not exist any agent available in the mission space, then the performance C has zero value. On the other hand, $C = 1$ implies the full coverage situation, i.e., $c_i \in \mathcal{D}, \forall i \in \{1, \cdots, N\}$

3 Performance Prediction

In this section, performance function $C(t)$ is predicted based on the information of agents in the mission space. It is assumed that the performance function converges to a steady-state value in finite time. At the steady-state, the age increment and decrement become same. The total increment of ages over the mission space can be written as follows,

$$\Delta \mathbf{a}_{\uparrow} = (N-m)\Delta t \tag{6}$$

where $m(< N)$ is the number of cells in $\mathcal{D}$, and Δt is a small time interval. Likewise, the total decrement of ages over the mission space can be represented as follows,

$$\Delta \mathbf{a}_{\downarrow} = -a_s \mu_c m \Delta t \tag{7}$$

where a_s is the average age of cells being detected by agent at the steady-state. Note that, in this study, the speed of agent with respect to the number of cells, μ_c, is considered and it may be different from the speed of agent, μ, because the square discretization of the mission space makes it dependent to the agent heading angle. In this study, μ_c is approximated as $\mu_c \approx \mu\sqrt{n}$.

The sum of the increment and the decrement becomes zero in the steady-state. From this relation, the average age of cells in the steady-state, a_s, can be computed as follows,

$$\begin{aligned} \Delta \mathbf{a}_{\uparrow} + \Delta \mathbf{a}_{\downarrow} &= \Delta t((N-m) - a_s \mu_c m) \equiv 0 \\ a_s &= \frac{1}{\mu_c}\left(\frac{N}{m} - 1\right) \end{aligned} \tag{8}$$

The steady-state average age a_s is used to compute the probability p, where p is the probability of an individual cell that has been detected during the time interval with length 1. That is, $i \in \mathcal{D}$ at least once during the arbitrary time interval $[t, t+1]$ when $p = 1$. Therefore, for every time interval τ, the probability is $p\tau$.

The predicted age of an individual cell, $\bar{a}_i(t) = \mathbb{E}(a_i)$, can be computed by adding up the expected values of each trial as follows,

$$\mathbb{E}(a_i) = \lim_{n \to \infty} \overset{n}{\mathbb{E}_{k,i}} \tag{9}$$

where

$$\begin{aligned} \mathbb{E}_{0,i} &= p\tau \times 0 \\ \mathbb{E}_{1,i} &= p\tau(1-p\tau) \times \tau \\ \mathbb{E}_{2,i} &= p\tau(1-p\tau)^2 \times 2\tau \\ &\vdots \\ \mathbb{E}_{n,i} &= p\tau(1-p\tau)^n \times n\tau \end{aligned}$$

with $\tau = t/n$. Simplifying the geometric series and substituting the result into (9) yield

$$\begin{aligned} \bar{a}(t) = \mathbb{E}(a_i) &= \lim_{n \to \infty} \sum_{k=0}^{n} p\tau(1-p\tau)^k k\tau \\ &= \frac{1}{p} - e^{-pt}\left(\frac{1}{p} + t\right) \end{aligned} \tag{10}$$

The probability p can be obtained by considering the steady-state value.

$$a_s = \lim_{t \to \infty} \bar{a}(t) = \frac{1}{p} \tag{11}$$

Combining (8), (10) and (11), we have

$$p = \frac{\mu_c m}{N - m} \tag{12}$$

$$\bar{a}(t) = \frac{1}{\mu_c}\left(\frac{N}{m} - 1\right) - \exp\left(\frac{\mu_c m}{N-m} t\right)\left(\frac{1}{\mu_c}\left(\frac{N}{m} - 1\right) + t\right) \tag{13}$$

Substituting it into (5) yields the predicted performance as follows,

$$\bar{C}(t) = 1 - \left(1 - \frac{m}{N}\right)\tanh(\rho \bar{a}(t)) \tag{14}$$

4 Persistent Coverage Controller

In this section, two different control algorhithms for persistent coverage control are presented. Figure 1 describes the mission space and the agents distributed. The target-based algorithm determines the target cell in the mission space, and the agent is guided to the target cell position. The gradient-based algorithm computes the gradient vector based on the newly detected cells, and the agent is guided to the gradient vector direction.

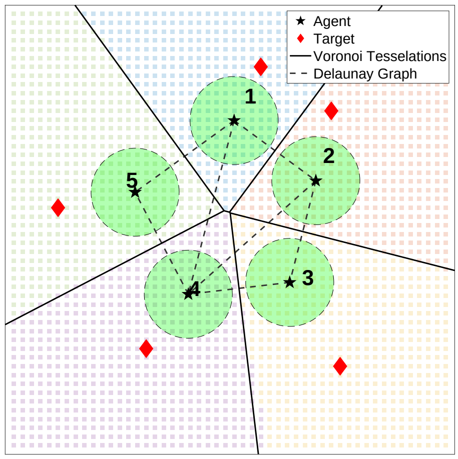
4.1 Target-Based Algorithm

Target based-algorithm determines the target cell of the agent based on the cell ages. The target cell t_j of the agent j is the cell having the biggest age value inside the Voronoi region, i.e.,

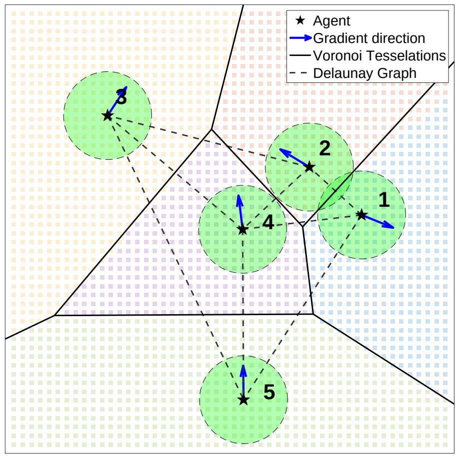
$$t_j = \underset{c_i \in V_j}{\operatorname{argmax}}(\hat{a}_i^j), \quad \hat{a}_i^j \in \mathcal{A}_j \tag{15}$$

Once the target cell t_j is determined, heading angle rate is used to make the agent reach the target cell. Figure 2 shows the geometry between the agent and its target cell. The angular error between heading angle of the agent and target direction is defined as follows,

$$\theta_e = |\theta_e| e \tag{16}$$



(a) Target-based algorithm



(b) Gradient-based algorithm

Fig. 1 Snapshot of the mission space and the agents

where

$$|\theta_e| = \cos^{-1}(\mathbf{v} \cdot \mathbf{v}_{\text{cmd}}) \tag{17}$$

$$e = -\text{sign}(\mathbf{v}_n(3)) \tag{18}$$

$$\mathbf{v} = \begin{bmatrix} \mu \cos\theta_j \\ \mu \sin\theta_j \\ 0 \end{bmatrix} \tag{19}$$

$$\mathbf{v}_{\text{cmd}} = \begin{bmatrix} T_j(1) - P_j(1) \\ T_j(2) - P_j(2) \\ 0 \end{bmatrix} \tag{20}$$

$$\mathbf{v}_n = \mathbf{v} \times \mathbf{v}_{\text{cmd}} \tag{21}$$

To regulate the angular error, the error dynamics can be designed by the following simple

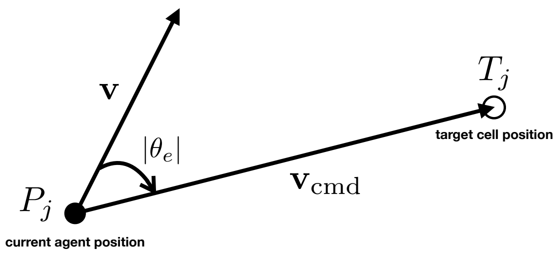


Fig. 2 Geometry between the agent and its target cell

first-order ordinary differential equation.

$$\dot{\theta}_e = \frac{d}{dt}\left(\theta_{\text{cmd}} - \theta_j\right) = -\dot{\theta}_j = -k\theta_e \qquad (22)$$

where $k > 0$ is a scalar gain. Control input u_j considering magnitude saturation $|u_j| \leq u_{\text{lim}}$ can be designed as follows,

$$u_j^0 = \dot{\theta}_j = k|\theta_e|e \qquad (23)$$

$$u_j = \begin{cases} \text{sign}\left(u_j^0\right) \cdot u_{\text{lim}} & \text{if } |u_j^0| \geq u_{\text{lim}} \\ u_j^0 & \text{Otherwise} \end{cases} \qquad (24)$$

4.2 Completeness of Taget-Based Algorithm

For the persistent coverage control problem, the target-based algorithm guarantees the complete surveillance of the given mission space. That is, the target-based persistent coverage algorithm does not leave any blind spots, and eventually sweeps the entire cells in the mission space in finite time.

Theorem 1 *For the bounded mission space $\Omega \subset \mathbb{R}^2$, agent dynamics (1), and control input (24), suppose that $\dot{a}_i$ is constant for all $i \in \{1, \cdots, N\}$. Then, every cell is detected in finite time.*

Proof. Consider arbitrary initial time t_0 and $t_1 > t_0$. All of the cells belong to one of two sets: i) One set, S^D is the set of cells detected by agent at least once before t_1, and ii) another set, S^U is the set of cells not detected until t_1. The upper bound of the elements in S^D can be written as follows,

$$\sup(a^D) = t_1 - t_0, \quad a^D \in S^D \qquad (25)$$

The age of the cell in S^U can be represented as follows,

$$a^U = t_1 - t_0 + a_0^u, \quad a^U \in S^U \qquad (26)$$

where a_0^u is the initial age of the cell at t_0. From Eqs. (25) and (26), following inequality can be obtained.

$$a^U > \sup(a^D) \geq a^D \qquad (27)$$

Therefore, when target-based algorithm, (15), is used, the target cell is selected in the set S^U. Because Ω is bounded, agent reaches the target cell in finite time with the control input, (24). Consequently, as $t_1 \to \infty$, the set $S^U \to \emptyset$.

4.3 Gradient-Based Algorithm

Gradient-based algorithm computes the gradient vector, and the agent is guided to the gradient vector direction. Therefore, the controller can be designed in a similar way of the target-based algorithm. In the gradient-based algorithm, however, the gradient vector is computed based on the predicted ages in the sensing range of the agent. The gradient vector is defined as the vector connecting the center of mass and the agent position. The center of mass is computed as follows,

$$C_j = \frac{L_j}{M_j} \in \Omega \qquad (28)$$

where

$$M_j = \int_{s \in \mathcal{D}_j} \hat{a}_i^j \, ds \qquad (29)$$

$$L_j = \int_{s \in \mathcal{D}_j} \left(s \cdot \hat{a}_i^j\right) ds \qquad (30)$$

The gradient vector $\mathbf{g} = C_j - P_j$ is used in the controller. The velocity vector command $\mathbf{v}_{\text{cmd}}$ in the gradient-based algorithm can be represented as follows,

$$\mathbf{v}_{\text{cmd}} = \begin{bmatrix} \mathbf{g}(1) \\ \mathbf{g}(2) \\ 0 \end{bmatrix} = \begin{bmatrix} C_j(1) - P_j(1) \\ C_j(2) - P_j(2) \\ 0 \end{bmatrix} \qquad (31)$$

5 Numerical Simulation

5.1 Prediction and Simulation

In this study, prediction model is proposed based on several assumptions. However, those assumptions may be different from the numerical simulation situation. The first factor is the approximation of constant μ_c. Because of the square discretization, though the speed of sensing agent is constant, the value of μ_c may not be constant due to the square discretization of mission space as well as heading angle change of the agent. If μ_c is overestimated than the actual value, then the probability in (12) will be also overestimated.

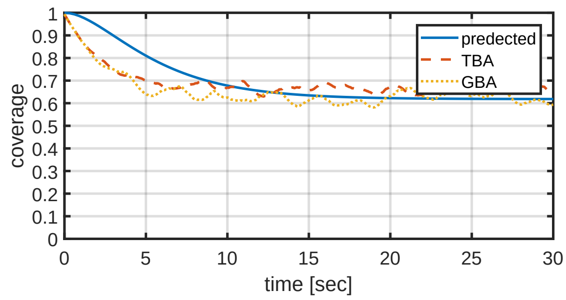
The second factor is the constant m assumption. The prediction model assumes that the number of cells in the detection range of the agent is constant. However, m may be lowered in the situations that some agents are too close to each other and their detection range overlaps. The same situation occurs when the agent is close to the boundary of the mission space. This makes the prediction model overestimate the probability in (12).

5.2 Simulation Case A. Number of Agents

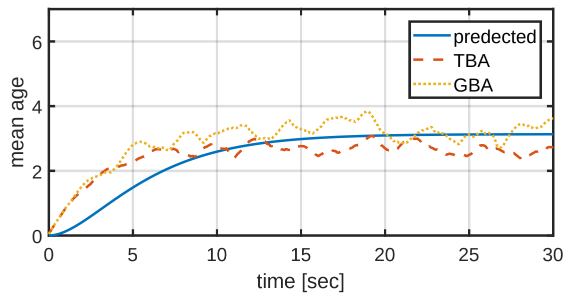
Numerical simulation is performed to compare the performance of the designed controllers and predicted performance. Two different simulation cases are considered. In the first simulation case, different number of agents in the mission space is considered, ($M \in \{3,5,7,9\}$). Intuitively, steady-state coverage performance will be higher if more agents are used. For the prediction, (11) and (12), increasing the number of agents increases m, and the predicted steady-state age is lowered.

The mission space is discretized into $n^2 = 50^2 = 2,500$ number of cells, and the sensing radius of agent is $r = 0.1$. The speed of agent is set $\mu = 0.5$. The homogeneous agents that has the same specifications with respect to each other are considered in the simulation. The design parameters are $\rho = 1/\sqrt{n}$, and $k = u_{\lim} = 10$.

Figures 3 and 4 show the coverage and mean age history for the simulation changing the number of agents. The coverage perforamances of

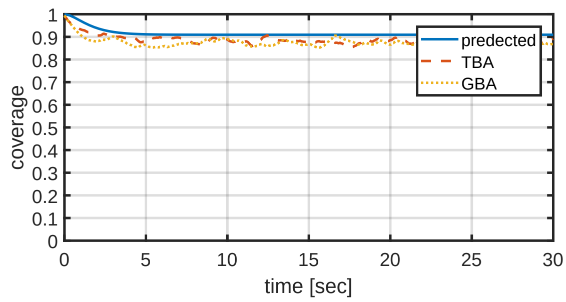


(a) Coverage history

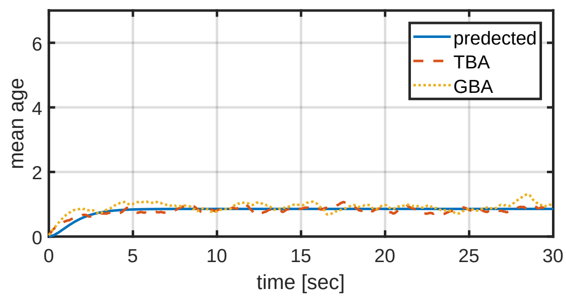


(b) Mean age history

Fig. 3 Coverage and Mean age history (3 agents)



(a) Coverage history

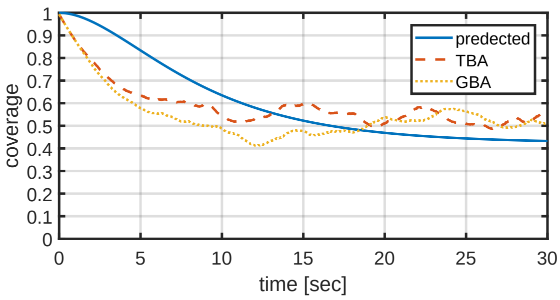


(b) Mean age history

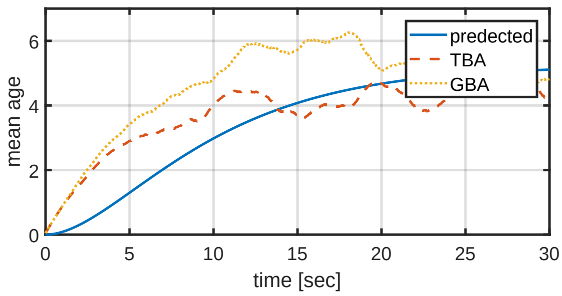
Fig. 4 Coverage and Mean age history (9 agents)

Table 1 Steady-state prediction performance in Cases A and B

	Prediction (%)	**|Prediction-TBA| (%p)**	**|Prediction-GBA| (%p)**
3 agents	61.85	5.30	1.28
5 agents	78.90	0.03	4.84
7 agents	86.65	1.66	3.33
9 agents	90.95	2.40	3.66
$\mu = 0.3$	43.27	12.24	7.11
$\mu = 0.5$	61.85	4.88	3.35
$\mu = 0.7$	71.90	2.24	1.43
$\mu = 0.9$	77.87	0.73	4.38

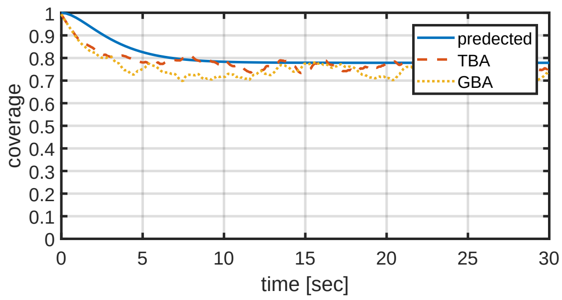


(a) Coverage history

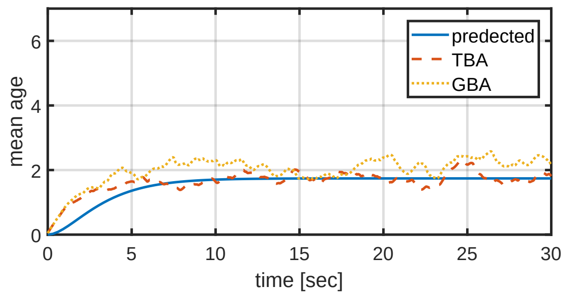


(b) Mean age history

Fig. 5 Coverage and Mean age history ($\mu = 0.3$)



(a) Coverage history



(b) Mean age history

Fig. 6 Coverage and Mean age history ($\mu = 0.9$)

target-based algorith and the gradient-based algorithm converges to steady-state value, but has some fluctuations. Because the initial age of the cells are zero, the initial coverage value is 1 and decreases. For every cases, the predicted age and coverage shows a decent predictions.

5.3 Simulation Case B. Speed of Agents

In the second simulation case, different speed of agent is considered, $\mu \in \{0.3, 0.5, 0.7, 0.9\}$. Simulation settings and design parameters are same as in Case A, except for the number of agent $M = 3$. Predicted steady-state age is lowered as the speed of agent becomes fast as shown in (11) and (12). This prediction is natural because fast agents will cover the mission space more fast and lowers the mean age of the cells in the mission space.

Figures 5 and 6 show the coverage and mean age history for the simulation with various speeds of agents. Steady-state coverage performance increases as the speed of the agent increases.

Simulation results are summarized in Table 1.

6 Conclusion

Persistent coverage control problem was formulated with cell discretization of the two-dimensional mission space and time-increasing age dynamics of the cells. Coverage performance function was proposed to measure the monitoring level of the mission space. The target-based controller and the gradient-based controller were designed for the persistent monitoring over the mission space. The target-based controller does not leave any blind spot and sweeps every cell in a finite time. Agents share their local information based on Voronoi tessellation and Delaunay triangulation graph. The performance of the persistent coverage algorithm was predicted based on agent specifications including the speed, the sensing radius, and the number of agents. The predicted coverage performance was compared with the numerical simulation results of the target-based and the gradient-based controllers. The prediction model shows the decent prediction of the coverage performance for various simulation cases with respect to the speed of agents and the velocity of agents.

References

[1] Mohseni F, Doustmohammadi A, and Menhaj M. B. Centralized receding horizon coverage control for mobile sensory networks. *Third International Conference on Intelligent Systems Modelling and Simulation*, Kota Kinabalu, Malaysia, Feb. 2012.

[2] Mohseni F, Doustmohammadi A, and Menhaj M. B. Distributed receding horizon coverage control for multiple mobile robots. *IEEE Systems Journal*, Vol. 10, No. 1, pp. 198–207, 2016.

[3] Zuo L, Shi Y, and Yan W. Dynamic coverage control in a time-varying environment using bayesian prediction. *IEEE Transactions on Cybernetics*, On-line Publication, Dec. 2017. doi:10.1109/TCYB.2017.2777959

[4] Nigam N. The multiple unmanned air vehicle persistent surveillance problem: a review. *Machines*, Vol. 2, No. 4, pp. 13–72. 2014.

[5] Smith S. L, Schwager M, and Rus D. Persistent robotic tasks: monitoring and sweeping in changing environments. *IEEE Transaction on Robotics*, Vol. 28, No. 2, pp 410–426. 2012.

[6] Lin X, and Cassandras C. G. An optimal control approach to the multi-agent persistent monitoring problem in two-dimensional spaces. *IEEE Transactions on Automatic Control*, Vol. 60, No. 6, pp 1659–1664, 2015.

[7] Cassandras C. G, Lin X, and Ding X. An optimal control approach to the multi-agent persistent monitoring problem. *IEEE Transactions on Automatic Control*, Vol. 58, No. 4, pp 947–961, 2013.

[8] Palacios-Gasos J. M, Talebpour Z, Montijano E, Sagues C, and Martinoli A. Optimal path planning and coverage control for multi-robot persistent coverage in environments with obstacles. *IEEE International Conference on Robotics and Automation (ICRA)*, Singapore, May 29 - June 3, 2017.

Contact Author Email Address

mailto: lsw7169@snu.ac.kr

Copyright Statement