

Multiple Robots Task Allocation for Cleaning and Delivery

Seohyun Jeon, Minsu Jang, Daeha Lee, Young-Jo Cho,
Jaehong Kim and Jaeyeon Lee

Abstract This paper presents a mathematical formulation of the problem of cleaning and delivery task in a large public space with multiple robots, along with a procedural solution based on task reallocation. The task in the cleaning problem is the cleaning zone. A group of robots are assigned to each cleaning zones according to the environmental parameters. Resource constraints make cleaning robots stop operation periodically, which can incur a mission failure or deterioration of the mission performance. In our solution approach, continuous operation is assured by replacing robots having resource problems with standby robots by task reallocation. Two resource constraints are considered in our formulation: the battery capacity and the garbage bin size. This paper describes and compares the performance of three task reallocation strategies: All-At-Once, Optimal-Vector, and Performance-Maximization. The performance measures include remaining garbage volume, cleaning quality, and cleaning time. Task allocation algorithms are tested by simulation in an area composed of 4 cleaning zones, and the Performance-Maximization strategy marked the best performance. Hence, a delivery task is added to the cleaning task. The delivery request operates as a new perturbation factor for the reallocation. The task allocation procedure for the delivery task includes the switching of tasks of the delivery robot itself as well as exchanging among cleaning robots to meet the balance of the cleaning performance. The experiment was conducted with 9 robots with the soft-

S. Jeon (✉) · M. Jang · D. Lee · Y.-J. Cho · J. Kim · J. Lee
Human Robot Interaction Research Team, Electronics and Telecommunications
Research Institute (ETRI), 218 Gajeongro, Yuseonggu, Daejeon 34129, Korea
e-mail: happyseohyun@etri.re.kr

M. Jang
e-mail: minsu@etri.re.kr

D. Lee
e-mail: bigsum@etri.re.kr

Y.-J. Cho
e-mail: youngjo@etri.re.kr

J. Kim
e-mail: jhkim504@etri.re.kr

J. Lee
e-mail: leeje@etri.re.kr

ware architecture that enables multi-functional of a robot and they performed both pseudo-clean and delivery task successfully.

1 Introduction

The problem of cleaning and delivery with multiple robots has been handled with various approaches. A common approach for cleaning with multiple robots is partitioning the space [9]. Ahmadi and Stone researched about partitioning the space for robots with different speed profile to minimize the overall cleaning time [2]. Once the space is partitioned, robots are distributed to their individual partitioned space to clean. Also, according to Ahmadi, the problem of cleaning a large public space can be defined as a *continuous area sweeping* task [2]. This means that the robot has to visit a place repeatedly rather than visiting once since the garbage is continuously piled on an open space. Similar application can be found for patrol and surveillance missions which sweeps the area continuously to detect the event of interest [7, 11].

Another approach for cleaning is a multi-robot coverage path planning within a single cleaning zone [10, 12, 17]. When multiple robots are assigned in a single space, paths of robots have to be generated, not to overlap with each other, to minimize the cleaning time. Also, a different approach for multi-robot cleaning is a formation control. If the cleaning target is impossible to be cleaned with one robot, a group of robots are coordinated to clean the target. Zahugi et al. researched about using a group of robots in a formation to clean the oil-spill [19].

Recently, Avidbots corp. introduced a autonomous cleaning robot system for a public space [1]. Two robots cooperate as a team for cleaning. Robots are able to

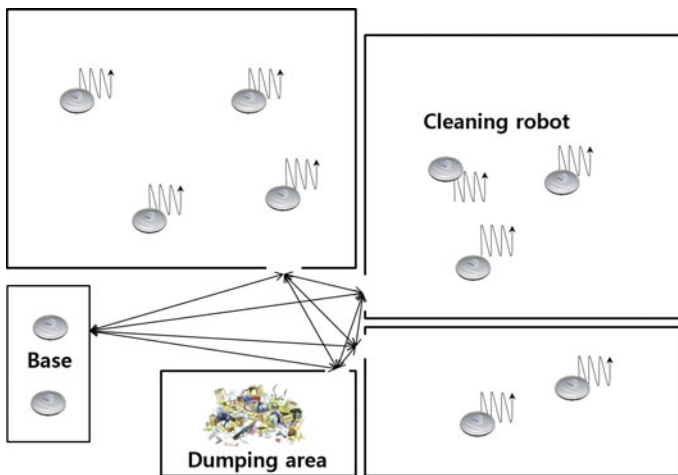


Fig. 1 Multi-robot cleaning as a team

autonomously localize, plan paths and sweep the floor. This system is just the beginning of the commercialized cooperative cleaning robot. More multi-robot systems will be used to clean a large public space in near future.

While operating cleaning robots, inoperative robots can be occurred for the reasons as discharge of battery or filled dust bin. Since such events as the battery and the bin can be solved after charging and emptying, these events are treated as normal inoperative situations. The advantage of using multiple robots stands out in this circumstance, that inoperative robots can be replaced with redundant robots.

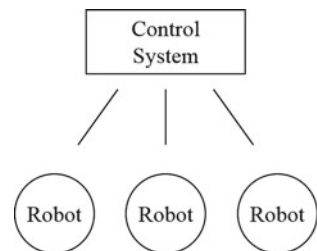
Thus, the approach of this paper for multi-robot cleaning is task allocation. This approach is sending robots as a team to the cleaning zone and reallocate robots when an inoperative robot occurs (Fig. 1). The cleaning zone is derived from the topological map. Methods as doors detection or Voronoi diagram are used to generate the topological map from the 2D grid map [15, 18]. Each vertex of the topological map becomes the cleaning zone. Allocating robot as a team to the cleaning zones and planning the path of robots inside the cleaning zone reduces the computational complexity compared to the partitioning the space or planning paths as a whole workspace. Each vertex of the topological map contains cleaning information of the zone such as area or the garbage amount which affects the task allocation.

In addition, a delivery task can be added to the robot. A delivery task is one of the most demanding task for automation. The necessary technology for a delivery robot is not very far from the current stage. In fact, several robots are already commercialized in the field such as Aethon and Swisslog in the healthcare center, or Kiva systems in the warehouse. The key issue of the delivery task is the task switching within the multi-functional robot and load-balancing among all deployed robots in case of the absent (for delivery) robot. According to the multi-robot task allocation (MRTA) taxonomy, this problem is MT-MR-IA [13].

Centralized control of the multi-robot system is used to find an optimal allocation as in Fig. 2. The control system allocates tasks to robots and the robot autonomously interprets the task and performs the assigned task.

This paper describes the problem of cleaning and delivery with multiple robots in a large public space and proposes MRTA algorithms to allocate tasks to robots. The final goal of the solution is providing an allocation vector that represents the number of robots to be placed in the zone and selecting an appropriate robot for delivery while maintaining the cleaning performance. The mathematical formulation of the

Fig. 2 Centralized multi-robot system



cleaning problem is described in Sect. 2 in which the task allocation strategies are introduced, and the simulated result and its analysis are discussed. The multi-robot delivery problem is explained in Sect. 3, followed by the conclusion in Sect. 4.

2 Multi-Robot Cleaning

2.1 Problem Description

In a large public area such as airport or department store, multiple robots are needed to clean the whole space. After the large space is divided into several cleaning zones, it needs to be decided how many robots are required to clean the single zone. If the garbage information of the environment and the specification of the cleaning robot is provided, the number of robot team for each cleaning zone can be decided. Thus, the object of the multi-robot cleaning problem is to find an appropriate team member of robots at each cleaning zones regarding the changing environment.

Since the battery discharges and the dust bin fills over time while the environment has constant inflow of the dust, the robot become absent to charge the battery and empty the bin. In this absent, robots become inoperative to conduct the assigned cleaning task. Inoperative robots have to be replaced to available robots to maintain the constant cleaning performance. Such replacement is defined as an allocation step. These constraints are assumed in this problem.

Constraints for the Inoperative Robot

- Battery The robot has limited battery level that discharges linearly.
 Dust bin The robot has limited capacity of the dust bin.

Goal The goal of this multi-robot cleaning problem is to find an allocation vector for each allocation step. The allocation vector is the number of robot in a team of a cleaning zone. The allocation vector at each reallocation step is denoted by \mathbf{A}_k .

$$\mathbf{A}_k = [r_k^1, r_k^2, \dots, r_k^n] \quad (1)$$

The total number of working (cleaning) robots at k th step is r_k ($r_k = \sum_{i=1}^n r_k^i$). The number of standby robots at k th step is r_{ks} , and the number of inoperative robots is r_{ki} . Then, the available robots r_{ka} at k th step is the sum of standby robots and the working robots. And, this is the same to subtracting inoperative robots from the total number of robots ($r_{ka} = r_{ks} + r_k = r_t - r_{ki}$). When previously inoperative robot becomes ready after charging the battery or emptying the bin, the robot's status becomes standby robot.

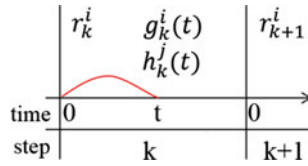


Fig. 3 The timeline used in the cleaning problem. k denotes the reallocation step, and t denotes the time passed after the reallocation. r_k^i and $g_k^i(t)$ is the number of assigned robots and the remaining garbage amount of i th zone, respectively, and $h_k^j(t)$ is the collected garbage amount of j th robot at time t of the step k

2.2 Problem Formulation

2.2.1 Timeline

To allocate robots at each zone, the environment and the robot should be modeled mathematically. To begin with, understanding timeline of the problem should be preceded. There are two types of timeline in this problem. One is the discrete time, and the other is the continuous time as in Fig. 3. Every reallocation steps is symbolized as k in the discrete time. The continuous time is used to show the time passed after reallocation, and it is symbolized as t . t is reset to 0 at every reallocation, and increases as time passes.

2.2.2 Robot

A set of r number of cleaning robots is represented by $R = \{R^j | j \in 1, \dots, r\}$, and each cleaning robot (R^j) has four attributes (A_R^j) and two parameters (P_R^j), and is defined as:

$$R^j = \langle A_R^j, P_R^j \rangle \quad (2)$$

$$A_R^j = \{C, M, T_w, T_c\}, \quad P_R^j = (b_k^j(t), h_k^j(t)) \quad (3)$$

where C is the cleaning capability of a robot, which is the volume of the collected garbage by the robot per hour. M is the maximum capacity of the dust bin of a robot, T_w is the operating hours of a cleaning robot with the fully charged battery, and T_c is the time it takes to fully charge the empty battery. $b_k^j(t)$ is the battery level, and $h_k^j(t)$ is the collected garbage amount of j th robot at time t in the allocation step k . $h_k^j(0)$ becomes the initial amount of the collected garbage at the allocation step k .

$$h_k^j(t) = h_k^j(0) + \frac{h_k^j(t)}{dt} t \quad (4)$$

$$\frac{h_k^j(t)}{dt} = \min \left(C, \frac{g_k^i(t)}{r_k^i} \right) = \begin{cases} C, & \frac{g_k^i(t)}{r_k^i} \geq C \\ \frac{g_k^i(t)}{r_k^i}, & \frac{g_k^i(t)}{r_k^i} < C \end{cases} \quad (5)$$

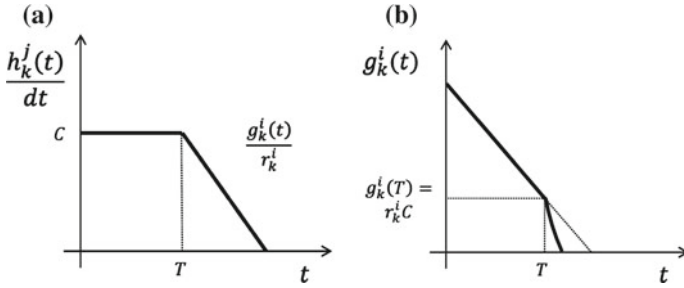


Fig. 4 If the remaining garbage of a cleaning zone is decreased below $r_k^i \cdot C$, the collecting garbage amount of a robot per hour is reduced to $\frac{g_k^i(t)}{r_k^i}$. In this case, $\frac{h_k^j(t)}{dt} = C$ at $t \leq T$, and $\frac{h_k^j(t)}{dt} = \frac{g_k^i(t)}{r_k^i}$ at $t > T$. **a** The collecting garbage amount of a robot. **b** The remaining garbage amount of a cleaning zone

Note that $\min(\cdot)$ term in Eq. 5 is used to satisfy the condition that if there are enough garbage in the cleaning zone for robots which can sufficiently collect the garbage of their capability, then the collecting garbage amount of a robot is C . Otherwise, the collecting garbage amount per hour is reduced to the remaining garbage amount of the zone divided by the number of robots assigned to the zone ($\frac{g_k^i(t)}{r_k^i}$). Figure 4a shows that after the time T , the collecting garbage amount is reduced to $\frac{g_k^i(t)}{r_k^i}$ from C since there are fewer garbage as in Fig. 4b. Thus, the decrease rate of remaining garbage amount of the cleaning zone after time T shows second order curved line as in Fig. 4b.

Also, the collected garbage amount of a robot cannot exceed the maximum capacity of the dust bin mounted on the robot from the constraints for inoperative robots. ($h_k^j(t) \leq M$) If $h_k^j(t) = M$, the robot has to stop cleaning and return to empty the dust bin. Similarly, let B_M be the marginal value of the discharged battery of a robot, then, if $b_k^j(t) \leq B_M$, the robot has to return to the charging station.

2.2.3 Environment

A cleaning zone of the total workspace can autonomously be extracted and numbered from a grid map generated by a robot by the graph cut algorithm [5]. An example of a cleaning area can be seen in Fig. 5. Figure 5a shows the space is divided into 4 cleaning zones and an auxiliary zone. The auxiliary zone is a hallway connecting each cleaning zones in which exists a charging station and a dumping area. Corresponding topological map can be seen in Fig. 5b with 4 nodes. The edge of the topological map represents the connection and the distance to the entrance between nodes. The allocation is conducted based on these nodes of the topological map.

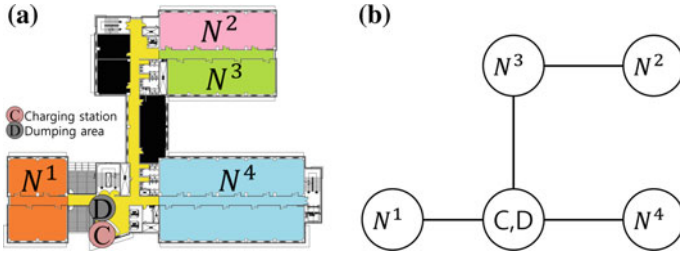


Fig. 5 An indoor 2D floor map and corresponding topological map of the cleaning area. **a** A 2D map with 4 cleaning zones. **b** Generated topological map from the 2D map

If there are n number of nodes $N = \{N^i | i \in 1, \dots, n\}$, each node of the topological map contains two attributes (A_N^i) and parameters (P_N^i) about the cleaning zones and can be defined as:

$$N^i = \langle A_N^i, P_N^i \rangle \quad (8)$$

$$A_N^i = \{A^i, \Delta^i\}, \quad P_N^i = (g_k^i(t), r_k^i) \quad (9)$$

where A^i is the area of the cleaning zone, Δ^i is the garbage increasing rate of the zone, g_k^i is the remaining garbage amount of i th node at time t in k th step, and r_k^i is the number of robots assigned for i th node at k th step.

There is various approaches to model the dirt distribution on the floor. Hess modeled the dirt distribution map of the floor with Poisson distribution of the grid-cell [8]. However, this paper assumed uniform distribution of the garbage that has linear increasing rate of (Δ^i).

$$g_k^i(t) = g_k^i(0) + A^i \Delta^i t - r_k^i \frac{h_k^i(t)}{dt} t \quad (8)$$

$$g_k(t) = \sum_{i=1}^n g_k^i(t) \quad (9)$$

In Eq. 8, $g_k^i(0)$ is the initial garbage volume amount in the i th node at the beginning of the step k . The second term $A^i \Delta^i t$ is the garbage volume occurrence in the i th node over time, while the last term indicates the total garbage volume collected by robots operating in the i th node in the duration of $t = [0, t]$.

Total number of robots Since there are limited number of robots, the user has to decide how many robots are needed at the workspace to clean continuously. If there are large amount of garbage occurring every time, it cannot be cleaned with small number of robots. Since the user cannot afford to buy infinite number of robots, there is a demand to know an optimal number of robots after comparing the cost and the cleaning performance. Hence, we want to maximize the performance of multiple robots with the limited number of robots. To achieve this, a task allocation algorithm is considered and this will be discussed in the Sect. 2.3.2.

This paper proposes the minimum number of robots required to clean the workspace. Given the information of robots and the cleaning space, the total minimum number of robots (r_t) required to clean the whole space can be defined as Eq. 10.

$$r_t = \sum_{i=1}^n \left\{ \left(\frac{A^i \Delta^i}{C} + \frac{g_0^i(0)}{C t_d} \right) \times \left(1 + \frac{T_w}{T_c} \right) \right\} \quad (10)$$

where t_d is a user defined parameter, which is the desired time of the space to be cleaned, and it is normally set to $t_d = 24$ h.

2.3 Task Allocation

2.3.1 The Cleaning Procedure

The multi-robot system for cleaning a large public space is controlled as the following procedure.

1. Initialize parameters ($g_0^i(0)$, $h_0^j(0)$).
2. Allocate robots (\mathbf{A}_k).
3. Perform cleaning.
4. Monitor parameters for the demand for reallocation ($b_k^j(t)$, $h_k^j(t)$).
If yes, go to step 2.

After initializing the environmental and robotic parameters, the control system decides the allocation vector (\mathbf{A}_k), that is the number of robots to be assigned at each cleaning zone. The allocation vector is derived by receiving the garbage amount ($g_k^i(t)$) and the number of available robots (r_{ka}) at the reallocation step. By applying allocation strategies, which will be explained in the following Sect. 2.3.2, the allocation vector is decided.

After deciding the number of robots, specific robots has to be selected to send to the target zone. Robots are selected based on the cost function (U_k^j). The cost function is a combination form of the distance to the target (d^j), remaining battery ($b_k^j(t)$) and the bin level ($M - h_k^j(t)$) as Eq. 11. Robots with minimum cost (U_k^*) will be assigned to the target zone. Since scales of parameters in the cost function are different, normalization is necessary. The cost function is a user-defined form, so weights of parameters can be defined by the user. Once the allocation vector is derived, the corresponding number of robots will move to the assigned zones and perform cleaning. The overall procedure can be seen in Fig. 6

$$U_k^j = f(d^j, b_k^j(t), M - h_k^j(t)) \quad (11)$$

$$U_k^* = \min_j (U_k^j) \quad (12)$$

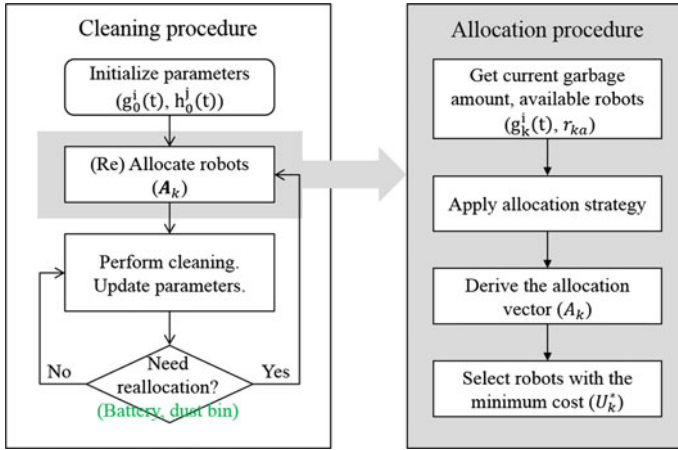


Fig. 6 Cleaning and its allocation procedure

When an inoperative robot (r_{ki}) occurs due to the battery discharge or full of the bin as discussed in Sect. 2.1, the robot cannot perform the cleaning task anymore, but has to return to the charging station or to the dumping area to resolve the situation. The control system monitors such encountered situation, and firstly searches for a substitute robot among standby robots (r_{ks}). If there is an available robot, the system replaces the inoperative robot to the standby robot. If not, the system has to decide whether to wait or to extract a robot from other cleaning zone, for persistent cleaning. This is decided by the task allocation strategy, and the reallocation result can be differed according to the strategy. This will be explained in the following section.

2.3.2 Task Allocation Strategy

Provided that environmental and robotic parameters are known, the control system decides the best allocation vector to clean the space since the performance of cleaning varies according to the initial allocation vector. Four different allocation vectors are tested as $A1_0$, $A2_0$, $A3_0$, and $A4_0$. The larger the number of allocation vector, the more robots are assigned at initial stage. The garbage parameters for the node 1, 2, 3 and 4 are set differently as in Table 1. It can be seen from the Fig. 7 that allocating many robots at initial stage does not always result in a good performance. Note that, even though the r_0 of $A4_0$ is larger than $A3_0$, the sum of remaining garbage of all nodes is not cleaned. (The line of the graph stays higher in $A4_0$ than $A3_0$.) This is because when initially many robots are allocated, there are not enough standby robots to substitute when former cleaning robots become inoperative and return to base for charging. In other words, intentionally keeping some standby robots initially will increase the performance of the cleaning. How many initial robots are needed to

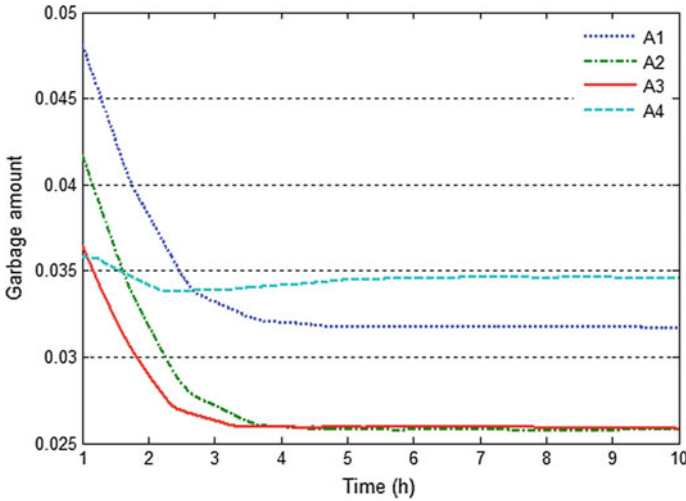


Fig. 7 The moving average of sum of the remaining garbage amount of all cleaning zones over time with different allocation strategies ($\mathbf{A1}_0 = [2\ 3\ 3\ 6]$, $\mathbf{A2}_0 = [3\ 4\ 4\ 7]$, $\mathbf{A3}_0 = [4\ 5\ 5\ 8]$, $\mathbf{A4}_0 = [5\ 6\ 6\ 9]$). The result of $\mathbf{A3}_0$ allocation shows the best performance with the least remaining garbage amount

be left for standby is decided by several allocation strategies. Considering this, three different allocation strategies are proposed as following.

All-at-once allocation (AOA)

This allocation method is relatively simple which is allocating all robots at once. All robot means robots that are available at the allocation moment. This allocation strategy does not leave standby robots. The number of robots for allocation at each node is as Eq. 13

$$r_k^{i*} = \frac{g_k^i(0)}{g_k(0)} \times r_{ka} \tag{13}$$

Optimal-vector allocation (OVA)

This method predefines the optimal number of robots for each cleaning zone and allocates robots only the number as defined even though there are remaining standby robots. This way standby robots is left and become substitutes when previously cleaning robots become inoperative. The allocation is as Eq. 14

$$r_k^{i*} = \frac{A^i \Delta^i}{C} + \frac{g_k^i(0)}{Ct_d} + 1 \tag{14}$$

Performance-maximization allocation (PMA)

In this allocation, the number of robots at each cleaning zone is decided by the remaining garbage amount at the allocation step. This method tries to maximize the performance as Eq. 15.

First, by defining the base robot set ($r_{kb}^i = \frac{g_k^i(t)}{g_k(t)} \times r_{ks}$) and altering the number of robots from the base set by combinatorial search, the performance and the remaining garbage amount ($g_k(t)$) can be calculated. The r_k^i with the best performance ($\min(g_k(t))$) is finally selected. This is the same as maximizing the collecting garbage amount per robot ($\max(h_k^j(t))$).

$$r_k^{i*} = \min_{r_k^i}(g_k(t)) = \max_{r_k^i}(h_k^j(t)) \quad (15)$$

Due to several reasons, such as the longer charge time, the distance between cleaning zones, and the limited number of robots, often there are few standby robots when reallocation is required. In this case, the control system regenerates the appropriate allocation vector based on the number of available robots (r_{ka}) at the reallocation step and this can result the change of assigned cleaning zone.

2.4 Simulation

2.4.1 Experimental Setup

Proposed allocation strategies are tested in the simulation. Parameters of each cleaning zone in the simulation are set as Table 1 with the area, the garbage increasing rate, and the initial garbage amount. Homogeneous robots are used for cleaning, and their attributes are set as $C = 0.001 \text{ m}^3 = 1 \text{ L/h}$, $M = 0.004 \text{ m}^3 = 4 \text{ L}$, $T_w = 4 \text{ h}$, and $T_c = 6 \text{ h}$. ($A_R^j = \{0.001, 0.004, 4, 6\}$)

Based on the attributes of the cleaning zone as Table 1, the total number of robots (r_t) required for the space to be cleaned is 40. Therefore, 40 robots are used in the simulation. The garbage increasing rate (Δ^i) is assumed to be static and uniformly distributed within the cleaning zone. Initial garbage amount is set as Table 1, which is 5 hours passed from the zero amounts. The charging station and the dumping area is separated, so a robot has to visit two different places according to its status. The traverse distance among cleaning zones are not taken into account in the simulation since it does not affect the result of the performance comparison among strategies.

Table 1 Simulation environment

Cleaning zone	Area ($A^i (m^2)$)	Garbage increasing rate ($\Delta^i (m^3/h)$)	Initial garbage amount ($g_0^i(0)(m^3/h)$)
N^1	1,000	2×10^{-6}	0.01
N^2	1,500	1.5×10^{-6}	0.0125
N^3	2,000	1.5×10^{-6}	0.015
N^4	2,500	2×10^{-6}	0.025

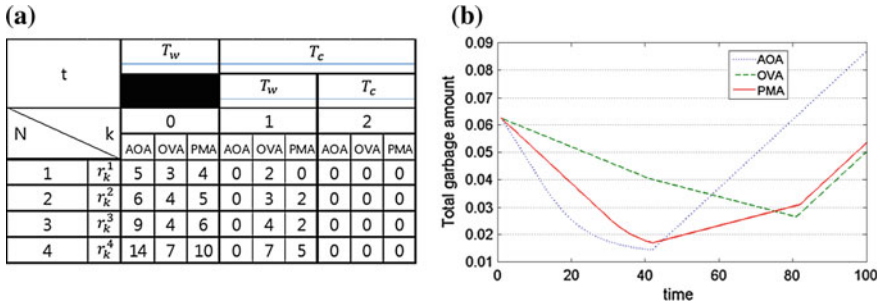


Fig. 8 Allocation of three strategies at first three allocation steps. **a** Allocation vector. The number of robots for each allocation strategies. **b** Garbage amount of the first allocation

2.4.2 Result and Analysis

The number of robots allocated to each cleaning zone varies according to the allocation strategy. The number of assigned robots at each cleaning zone and corresponding remaining garbage amount resulting from the allocation vector can be seen in Fig. 8. It shows the first allocation cycle of AOA, OVA, and PMA. The allocation cycle comes at every $(T_w + T_c)$ time. The allocation vector is repeated within the cycle for AOA and OVA, whereas PMA’s changes since the garbage amount decreases and corresponding collecting garbage amount of a robot decreases below the capability (C). After first assigned group of cleaning robots returns for charging the battery or emptying the bin, the second group moves for cleaning at allocation step $k + 1$, and this is continued for the next cycle.

Figure 8b shows the first cycle of the remaining garbage of all cleaning zones at $k = 0, 1$, and 2 for three strategies. Note that the remaining garbage of AOA increased dramatically since there are no cleaning robots at $k = 1$ and 2 while garbage is still increasing.

The overall cleaning performance of each allocation strategy is measured by three factors: The average boundary of the garbage amount, the cleaning quality, and the cleaning time.

During the simulation, the remaining garbage amount is increased and decreased within upper and lower boundaries. The lower the average between boundaries, the cleaner the area is. From Table 2, OVA and PMA is maintained cleaner than AOA.

Table 2 Cleaning performance comparison for three allocation strategies

Strategy	Average boundary of garbage	Cleaning quality	Cleaning time
AOA	0.0570	5.21×10^{-9}	∞
OVA	0.0228	1.09×10^{-9}	86.2h
PMA	0.0228	0.87×10^{-9}	81.0h

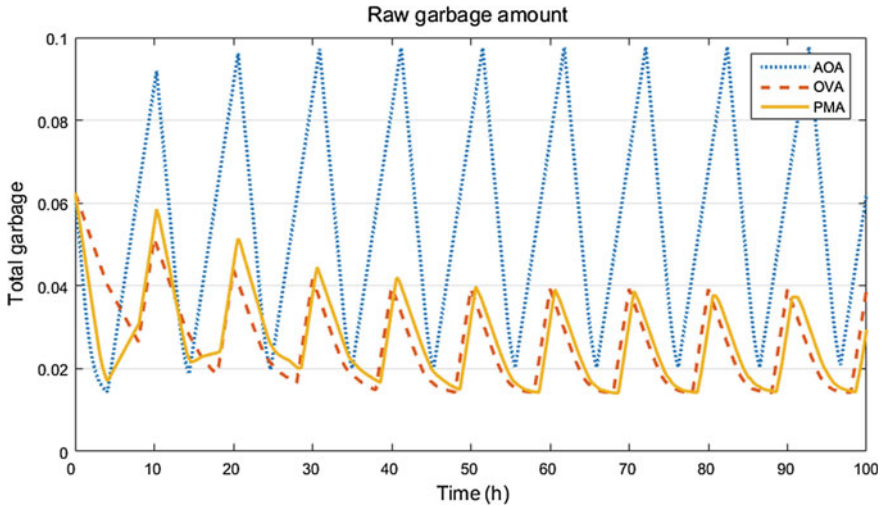


Fig. 9 The simulation result of the garbage amount of AOA, OVA, and PMA strategies

Although the average boundaries is low, if the difference of upper and lower boundaries is large, it can be said that the cleaning quality is low. It means the zone is cleaned but contaminated easily. This value is measured by calculating the deviation of two boundaries. The difference of the boundaries can be seen in Fig. 9 as saw-toothed shaped. From Table 2, the best cleaning quality is provided by PMA and it can be seen as orange solid line in Fig. 9.

The cleaning time is the time consumed to clean the garbage below the marginal amount. The marginal amount is set to $1.1 \times \Delta^i$ in this paper. Since the garbage increases continuously, there is no end of the cleaning, but, the first time that the garbage amount reaches the marginal amount is defined as the cleaning time. The calculated cleaning time of the PMA is slightly faster than OVA as in Table 2.

The simulation result shows that the initial allocation vector affects the cleaning performance, and leaving standby robots for replacement results in better performance as can be seen from PMA and OVA task allocation strategies.

3 Multi-Robot Delivery

3.1 Problem Description

Delivery is one of the highly demanding tasks for automation. For example, hospital staffs require frequent turn-arounds to deliver materials such as medicine, mails, and linens. Nurses walk maximum 10 Km a day, not only to care patients, but also, to get supplies [3]. To replace such mobility work inside a healthcare center, robots have been developed by Aethon and Swisslog to task over the delivery task [16]. Also, Kiva systems in the warehouse is one of the succeeded robotic system that

used for delivering packages. They insisted that by improving the algorithms of coordinating robots, the number of robots required in the warehouse can be reduced to 5–10 % [6]. Amazon is now extending their robotic application to deliver outside of the warehouse, directly to customers. From the aforementioned examples, it can be tested for robots to add a delivery task besides the cleaning task to maximize the usage. To achieve this, the robot should be multi-functional and the control system should select an appropriate robot for delivery while assigning the cleaning zone.

The performance of the delivery task can be evaluated by several indicators, in scopes of cost and time. The standard for carriers and logistics services providers categorized the key performance indicator into time, alert, security and efficiency [14]. The problem of robot for cleaning and delivery, in this paper, only considered the time and efficiency indicators since the experiment is conducted in the laboratory and the purpose of using robots are multi-functional which is not dedicated only for the logistics. Therefore, this problem disregards the alert and security problems. The time considered in this problem is waiting time of the pick-up position until a robot arrives after the delivery is requested. The efficiency is considered in scope of cleaning performance.

The delivery request is made with the simple input data of positions for pick-up and drop-off. Therefore, the problem of delivery is to find a nearest robot to the delivery position among cleaning robots while maintaining the cleaning performance.

3.2 Task Allocation

The reallocation due to the delivery request is similar to the battery and dust bin events in the cleaning problem. The task allocation procedure for delivery request is as in Fig. 10. When there is a request for a delivery, a nearest robot to the pick-up position is selected for the delivery. When delivery task is finished in the drop-off position, the robot returns to the cleaning task on the nearest cleaning zone from the finished position. To balance the cleaning performance, a robot from the drop-off position is selected to change the assigned zone with the delivery robot and is allocated to the new cleaning zone where the delivery robot has left. This way the cleaning performance is maintained continuously.

The delivery task while cleaning is conducted by following procedure:

1. A delivery is requested from the pick-up (PU) position to the drop-off (DO) position.
2. The task allocator selects a robot for the delivery and **reallocates** the cleaning tasks.
3. The selected robot moves to the PU position from the current position.
4. The robot arrives at the PU position and loads the package.
5. The robot moves to the DO position and unloads the package.
6. The robot returns to the cleaning state and perform cleaning on the newly assigned cleaning zone.

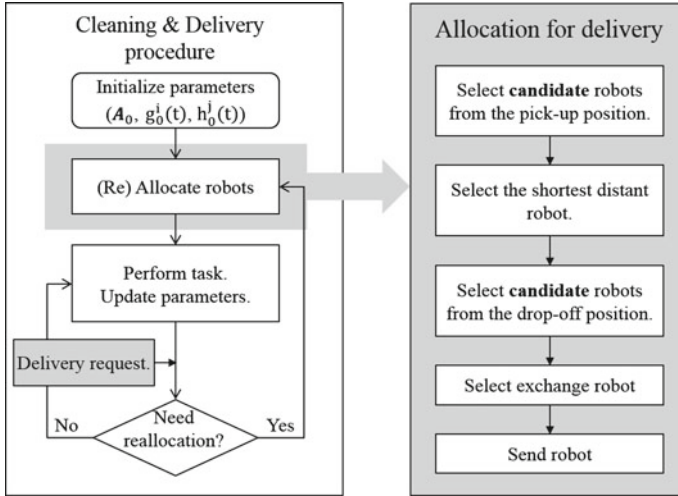


Fig. 10 The task allocation algorithm for cleaning and delivery

The task allocator considers the time efficiency, thus the cost function of the delivery task measures the distance (d).

$$\min_r(d) \tag{16}$$

Normally, the robot from the nearest cleaning zone that presents the minimum distance to the PU position is selected for the delivery. The absence of the delivery robot from the cleaning zone incur perturbation of previous allocation and becomes another factor for the task reallocation. The key point of this strategy that differs from other problem is that the second procedure from above that the task allocator reallocates the cleaning zone of all robots to maintain the balance of the cleaning task. In this procedure, a robot might change its assigned cleaning zone from the previous assigned cleaning zone to the new zone where the delivery robot has left. This way, the delivery robot can save time of returning to the cleaning task compared with going to the original cleaning zone, while maintaining the overall cleaning performance. This scenario is depicted in Fig. 11.

3.3 Experimental Setup

3.3.1 Hardware Setting

The experiment for the cleaning and delivery mission is conducted with 9 robots. The robot used for the experiment is Kobuki as in Fig. 12. Although the base platform came from the robot cleaner, this research robot does not have broom for cleaning.

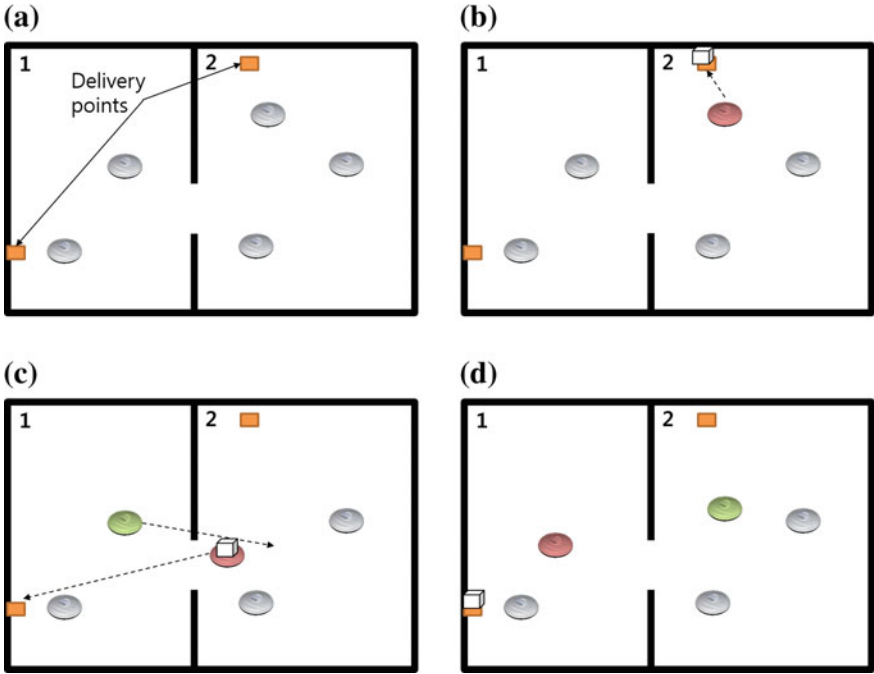
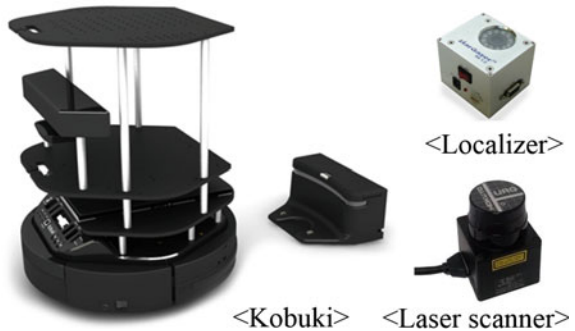


Fig. 11 Cleaning and delivery task scenario. **a** Initial allocation for cleaning. **b** A delivery request, and a (*red*) robot is selected for the delivery. **c** A (*green*) robot is selected for the substitute. **d** Two robot have changed their assigned cleaning zone after the delivery

Fig. 12 The robot used for the experiment



Since the only difference is the cleaning equipment, the movement of robot is programmed similar to the cleaning situation. Thus, it can be said that this robot performs pseudo-cleaning. The robot is mounted with a localizer which looks marker of the ceiling and a laser scanner for obstacle avoidance. A button for the user-interface of the delivery task and a light tower is used to indicate the status. All robots are connected via wifi to the Remote Operating Center (ROC).

3.3.2 Software Setting

The software architecture controlling robots and ROC is named as Cooperative Intelligent Robotic System (CIROS). CIROS is consisted of the Message processors, Context memory, Task controller, Individual task interpreter, Controller, Navigator, Robot platform, Monitor, and the Reporter as in Fig. 13.

When a mission is commanded from ROC, the Message processor receives the command, and the message is interpreted and matched from the Context Memory. If the message is a task, the Task controller receives the message and selects an appropriate task. Then, the Individual task interpreter takes action for the robot. If the command is cleaning, the Controller commands the robot to clean. And, if the command is delivery, the robot is controlled to follow the delivery action sequence. The uRON library is used for the Navigator which generates and follows the path with the localizer and avoids obstacle autonomously [4]. The final motion behavior of the command reaches to the hardware of the Robot platform to control sensor and actuator. All movements is monitored in the Monitor, and after matching it to the Context memory, it sends out the status of robot to ROC from the Reporter.

3.4 Result and Analysis

Nine robots are deployed in 3 zones on the 16 m × 6 m space for the laboratory experiment as in Fig. 14. Seven robots are initially deployed for cleaning and 2 robots

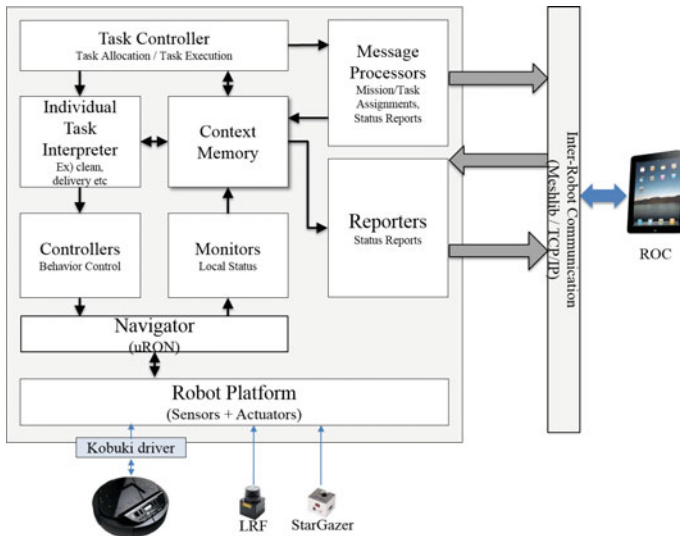


Fig. 13 Software architecture of a robot

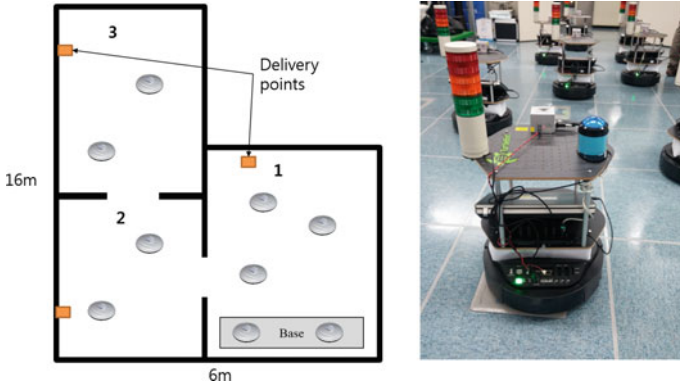


Fig. 14 Real application of multiple robots for cleaning and delivery mission

remained at the base for standby. Deployed robots move through the path generated by the boustrophedon method mimicking cleaning robot. Delivery positions are localized off-line and the delivery request is randomly called on-line. When a delivery request is made from zone 1 to zone 2, the nearest robot to the delivery position from zone 1 is selected for the delivery and moves to the pick-up position while a robot from zone 2 is selected to change the cleaning zone to 1. After loading the package, the delivery robot successfully moves to the drop-off position at zone 2 following the delivery task sequence while a robot from zone 2 is moved to zone 1 to fill out the absence. After the delivery is finished, the delivery robot stays in zone 2 for cleaning, and the exchange of robots was performed successfully (Fig. 14).

4 Conclusion

This paper describes the problem of cleaning and delivery mission in a large public space with the multi-robot system based on the task allocation approach. The multi-robot system seeks for optimization, so it inevitably follows the centralized control system.

This paper first describes the problem of cleaning. The goal of the cleaning problem is to allocate a group of robots to each cleaning zone according to the robotic and environmental status. Every reallocation is occurred due to limitations of garbage bin and battery amount of a robot. The problem is formulated mathematically regarding the robotic action and environment. The minimum number of required robots for cleaning the given space is suggested. Cleaning procedure with reallocation is explained. Three allocation strategies are proposed and tested by simulation. The number of robots allocated to each cleaning zone varies according to the allocation strategy. The result is compared with respect to the remaining garbage, the cleaning quality, and the overall cleaning time. It can be concluded that the performance-

maximization allocation (PMA) strategy performs better than the optimal-vector allocation (OVA) or the all-at-once allocation (AOA) for cleaning a large space. The PMA and OVA strategies have in common that they intentionally remain standby robots, and this leads to better performance since the cleaning task can be successfully carried over to standby robots when the former robots become inoperative to perform the task.

And, the problem complexity is increased by adding the delivery task to the cleaning task. The delivery request becomes another factor for the reallocation. The delivery robot is selected to minimize the waiting time. The selected delivery robot follows the pick-up and drop-off task sequence. To balance the absent of the delivery robot and minimize the return time to cleaning task, the control system reallocates the cleaning zone to robots, and an exchange among robots of the assigned zone can occur to maintain the cleaning performance. The experiment with 9 robots in the laboratory shows the allocation of switching between two tasks was successfully conducted.

The future direction of this work will be developing the allocation strategy while increasing the cleaning performance and analyzing the effect of switching robots for frequent delivery request.

Acknowledgments This work was supported by the Robot R&D program of MOTIE/KEIT. [10051155, The Development of Robot Based Logistics Systems Applicable to Hospital-wide Environment]

References

1. Ackerman, E.: Avidbots wants to automate commercial cleaning with robots. *IEEE Spectr.* (2014)
2. Ahmadi, M., Stone, P.: A multi-robot system for continuous area sweeping tasks. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1724–1729 (2006)
3. Bardram, J.E., Bossen, C.: Mobility work: the spatial dimension of collaboration at a hospital. *Comput. Support. Coop. Work (CSCW)* **14**(2), 131–160 (2005)
4. Choi, S., Lee, J.Y., Yu, W.: uRON v1.5: A device-independent and reconfigurable robot navigation library. In: *Proceedings of the IEEE Workshop on Advanced Robotics and its Social Impacts (ARSO)*, pp.170–175 (2010)
5. Choi, J., Choi, M., Nam, S.Y., Chung, W.K.: Autonomous topological modeling of a home environment and topological localization using a sonar grid map. *Auton. Robots* **30**(4), 351–368 (2011)
6. D'Andrea, R.: Guest editorial: a revolution in the warehouse: a retrospective on kiva systems and the grand challenges ahead. *IEEE Trans. Autom. Sci. Eng.* **49**, 638–639 (2012)
7. Elmaliach, Y., Agmon, N., Kaminka, G.A.: Multi-robot area patrol under frequency constraints. *Ann. Math. Artif. Intell.* **57**(3–4), 293–320 (2009)
8. Hess, J., Beinhofer, M., Kuhner, D., Ruchti, P., Burgard, W.: Poisson-driven dirt maps for efficient robot cleaning. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp.2245–2250 (2013)
9. Jager, M., Nebel, B.: Dynamic decentralized area partitioning for cooperating cleaning robots: In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp.3577-3582 (2002)

10. Janchiv, A., Batsaikhan, D., Kim, B.S., Lee, W.G., Lee, S.G.: Time-efficient and complete coverage path planning based on flow networks for multi-robots. *Int. J. Control Autom. Syst. (IJCAS)*, **11**(2), 369–376 (2013)
11. Kolling, A., Carpin, S.: Multi-robot surveillance: an improved algorithm for the graph-clear problem. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2360–2365 (2008)
12. Kong, C.S., Peng, N.A., Rekleitis, I.: Distributed coverage with multi-robot system. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2423–2429 (2006)
13. Korsah, G.A., Stentz, A., Dias, M.B.: A comprehensive taxonomy for multi-robot task allocation. *Int. J. Robot. Res. (IJRR)* **32**(12), 1495–1512 (2013)
14. Koscielniak, T., et. al.: Key performance indicators for carriers & logistics service providers. Odette International Limited, p. 8 (2007)
15. Lee, J.H., Choi, J.S., Lee, B.H., Lee, K.W.: Complete coverage path planning for cleaning task using multiple robots. In: *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pp. 3618–3622 (2009)
16. Niechwiadowicz, K., Khan, Z.: Robot based logistics system for hospitals-survey. In: *IDT Workshop on Interesting Results in Computer Science and Engineering* (2008)
17. Rekleitis, I., Lee-Shue, V., New, A.P., Choset, H.: Limited communication, multi-robot team based coverage. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3462–3468 (2004)
18. Santos, F.N., Moreira, A.P.: Towards extraction of topological maps from 2D and 3D occupancy grids. *Lecture Notes in Computer Science (LNCS)*, *Prog. Artif. Intell.* **8154**, 307–318 (2013)
19. Zahugi, E.M.H., Shanta, M.M., Prasad, T.V.: Design of multi-robot system for cleaning up marine oil spill. *Int. J. Adv. Inf. Technol. (IJAIT)*, **2**(4), 33–43 (2012)