휴리스틱 접근법에 의한 다수 이동 로봇 에이전트를 이용한 물질 수집 시스템

이영대, 김식

세명대학교 정보통신학과

A Heuristic Approach for Material Collection System by Multiple Mobile Robot Agents

Lee Young Dae* and Kim Sik
* Dept. of Computer and Info. Engineering, Semyung Univ., Jecheon, Chungbuk, 390-711 Korea
Tel : +82-43-649-1416 Fax : +82-43-644-2111 E-mail: ydl@venus.semyung.ac.kr

Abstract

In this paper, we address on the problem of automatic scheduling and motion generation of multiple mobile robots for collecting material parts. We propose a model and solution algorithm for the system. The formulated problem is divided into two kinds of problems; assignment problem and planning problem of robot motion. In this paper, several approaches to solution methods are presented and compared through computer simulation.

1. INTRODUCTION

Typical applications of robot manipulator involve accomplishing tasks within fixed workspace. The introduction of mobility creates an ability, which can be exploited to achieve a broader range of tasks. A mobile robot system is capable of performing a number of tasks in widely separated locations. For example, mobile robots are currently being proposed and developed for a variety of applications including factory automation, planetary exploration, nuclear reactor maintenance, and construction engineering [1][2].

In these applications, a common task description involves acquiring and moving materials from one point to another, which could not be accomplished by a stationary manipulator. Thus, mobility is essential for accomplishing the task and it is desirable to plan efficient robot motion.

The AGVs(Automatic Guided Vehicles) as a kind of typical mobile robots, are being used as a means of delivering parts and materials within highly automated factories[3][4]. The main topics of AGVs address on planning paths to avoid stationary obstacles, navigation, and control, including issues of sensing, kinematics, dynamics and traffic control[5][6].

The motion planning of multiple mobile robot is complicated by task requirements, which impose constraints on the system model for acquiring scattered material. Multiple mobile robots are expected to reduce the total task time and have robustness in comparison with a single robot system.

This paper considers the system rapidly collecting scattered material such as garbage by multiple mobile robot agents. Once a mobile robot is given a task by a supervisory controller, it travels to the source at the location of a material part. Then, it picks up parts at another sources until its container buffer is full. Finally, it delivers to the sink position and stores the collected material into the box. We aim at presenting an approach to the modeling and scheduling algorithm of the system. We formulate an optimization of the problem and development of solution algorithms.

The remainder of the paper is organized as follows. We begin, in Section 2, by modeling the system with several assumptions. The problem is formulated in Section 3. It is divided into several sub-problems to apply suitable solution methods. Then, in Section 4, we present several methods to solve the proposed model. In Section 5, we show the numerical comparison results of the solution methods from computer simulation. The results of this work are summarized in Section 6.

2. SYTEM MODEL

The material collection system using multiple mobile robots considered in this paper is composed of mobile robots, materials, boxes and a supervisory controller on the floor in a room. The system is mainly composed of the following components.

Sources: They are the material parts on the floor. Initially, all materials are randomly and sparsely scattered on the floor in the room.

Sinks: They are the boxes on the floor. The size of each box is limited. The boxes can contain all material.

Robot Agents: The task object of robots is to acquire, to transfer and to deposit all material into boxes. A robot has a container buffer to store material and its capacity is limited. Robots move with constant velocity on the path and the loading and unloading time of material parts are fixed. Robots download from commands and report the status of it to a supervisory controller through the wireless communication facility.

Supervisory controller: A supervisory controller for the system recognizes the positions and status of scattered material parts, boxes and robots from the sensors. It schedules the job of robots that is the main topic of this work, and commands the robots via a wireless communication system.

Next we introduce the following symbols for the convenient description of the system

[Symbols for the system]

O : set of objects. And, $o \in O$ denotes an object, which represent a material part, a box or a robot.

M, B and R : set of material parts, boxes and robots, respectively. Also, We denote m, b and r a material part, a box and a robot, respectively.

$\tilde{O} \subseteq O$: subset of the objects grouped by their types. We denote $\tilde{P}, \tilde{R}, \tilde{B}$ as subset of material parts, robots and boxes, respectively.

$\lambda(P)$, $\lambda(R)$: number of types in material and number of types in robots to handle the kind of material.

$\tilde{R}^{LM}$, $\tilde{R}^{UM}$, $\tilde{R}^{IS}$: subset of robots, that is loaded and moving, unloaded and moving, and idly stopped with no job, respectively.

$\tilde{P}_i^{LM}$, $\tilde{P}_i^{CM}$, $\tilde{P}_i^{RC}$, $\tilde{P}_i^{SB}$: subset of material that is, loaded and moving ones, contracted to be loaded ones, not contracted one and stored ones into boxes, respectively. Where, i is a type index for material.

$|O|$: cardinal number of object set O .

$I(O) = \{1, 2, \cdots, |O|\}$: index set for object O , where $|\bullet|$ denotes cardinal number of a set.

$N(O)$: node set or location points set of objects on the floor.

$d(n_i, n_j) \equiv \|n_i - n_j\|$: shortest distance between two nodes $n_i, n_j \in N$, where $\|\bullet\|$ is Euclidian distance norm .

$d(L)$: distance of a path link L consisted of nodes $L(\tilde{N})$.

v : velocity of a robot, which is regarded constant in this work.

$L(\tilde{N})$: a path link consisted of the nodes $\tilde{N} \subset N$ for a robot to visit for the task.

$T(n_i, n_j) = d(n_i, n_j) / v$: moving time of a robot between two nodes $n_i, n_j \in N$.

$T(L)$: tour time of a robot on the path link L for task.

$T(n_i)$: : service time of a robot at a node.

$T^{idle}(r_i)$: time that a robot is in idle state.

$C(r_i)$: maximum buffer capacity of a robot to store material, which is the same in all robots in this work.

$C(b_i)$: maximum buffer capacity of a box to store material, which is the same in all boxes in this work.

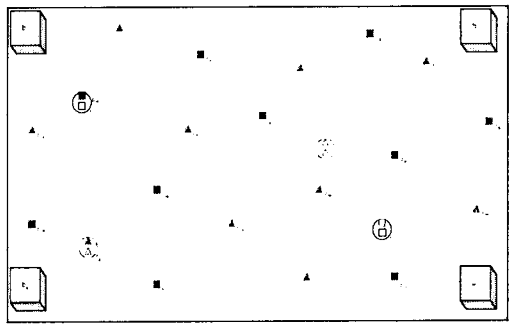

Fig. 1. Schematic diagram of the proposed system

3. PROBLEM FORMULATION

The target problem of this paper is to collect scattered material rapidly as possible using multiple mobile robots. This problem belongs to a kind of general optimization problem. It has complex nature due to highly interrelated sub-problems, which makes it difficult to find the global optimal solution.

In this work, we adopt the divide and conquer strategy to solve the target problem. We aim at finding sub-optimal solution by dividing the target problem into three kinds of sub-problems; assignment of material parts to boxes, assignment of material to a robot and motion planning of that robot. By formulating sub-problems based on some heuristics and by applying solution algorithms to them, a solution for the target problem is derived. We introduce the following assumptions for the system.

1) During the task, no two mobiles robots may pass on the same path of two nodes.

2) All kinds of material can be handled by robots, which means that $\lambda(P)$ equals to $\lambda(R)$.

3) A box or a robot has a limited buffer capacity and it can store any kinds of material.

Next, we will describe the above problem in detail and compare with several solution algorithms.

3.1. Assignment of material parts to box

All scattered material parts should be collected into the boxes. Furthermore, the total material parts scattered on the floor should be contained in boxes at the end. Here we assign material parts near to a box as possible to reduce the delivery time. Then the sub-problem for assigning material parts to a box can be written as

[Sub-problem P1]
Find $\widetilde{P}_i^*$ for all $i \in I(B)$ such that
minimizing the cost $\sum_{j=1}^{|\widetilde{P}_i|} d(b_i, p_j)$
, where $\widetilde{P}_i^*$ is the solution subset of material parts, which are assigned to a box $b_i, i \in I(B)$, $d(b_i, p_j)$ is distance between a box b_i and a material part p_j . And, $m_j, j \in I(\widetilde{P}_i)$ is the j-th material part element in the subset $\widetilde{P}_i$.

3.2. Assignment of material to a robot

After grouping material parts near to the box, the delivery robots should be selected. Generally, the following situation can be considered in this system.

Table 1 Description of situation in the system

$m = \lvert R^{IS} \rvert$	$n = \lvert P^{NC} \rvert$	Description
0	$1 \le n \le \lvert P \rvert$	No idle robot. Awaiting an idle robot
$1 \le m \le \lvert R \rvert$	0	Case of $m \succ n$ or final state of task
$1 \le m \le \lvert R \rvert$	$1 \le n \le \lvert P \rvert$	Typically $n \succ m$, idle robots more than one are ready for task

(m : number of idle robots, n : number of material parts not contracted to move)

Because the number of material parts is larger than that of robots during the task, it is desirable to select the robot that finished its job first and to assign the material parts among the grouped ones. This rule provides the robots, which finished their jobs and awaiting dispatch, with an opportunity to participate in the work. Thus it utilizes the substantial resources of transportation maximally. For rapid transportation of material, we introduce a cost for the assignment of material parts by a robot as follows.

[Sub-problem P2]
Find a robot r_k such that
minimizing the cost $\alpha \cdot [T(L) + \sum_{i=1}^{|\widetilde{N}|} T(n_i)] + \beta \cdot N_c(L)$
,where $r_k \in R^{IS}$ is one of the idle robots that finished their jobs. $T(L)$ is travel time of the robot to tour the path $L(\widetilde{N})$ connecting the node set $\widetilde{N}$. $N_c(L)$ is inserted in the cost function to avoid the collision against the substantial obstacles during the robot tour. And, α and β are weight factors.

4. SOLUTION APPROACH

4.1 Assignment algorithm

Numerous solution strategies can be used for the assignment of a part to a box such as typical random selection or greedy one, etc. In this work, we present the other two methods based upon a typical greedy one and another one based on exhaustive search.
1) RM(Random method): It matches material parts, boxes and robots randomly. It is easy to implement and very fast, but it shows poor performance.
2) GM(Greedy Method)
-GM1(Greedy method 1): It selects material part first and box near to the robot that finished its job. It is ideal if a box has an infinity capacity.
-GM2(Greedy method 2): It makes a graph that connects fist the nodes of material parts, boxes and robots. Next, the robot path is constructed by selecting the shortest distance of material nodes and a box node. Computation time in this method increases, as the number of material part and the size of robot buffer increase.
3) SM(Suggested exhaustive search method): First, it constructs a graph network using the nodes of material parts and boxes. Next, the path is constructed based upon the shortest distance. Although the computation time of the second method is faster than the first one, it also has much computational burden as material parts and boxes increase.

4.2. Planning a robot motion

Our method to plan a robot motion is as follows. First, the straight path from the start point to goal is connected. If no obstacle is found on the path, then the two points are marked and memorized. However if an obstacle is on the path, then an immediate goal point near to an obstacle is generated and the old goal point is put into stack memory of the supervisory controller. This procedure is repeated until there are no obstacles on the path of a robot.
If a collision free path is found, a robot moves from start point to goal one through immediate points that have been memorized. The collision can occur during the task among moving robots. This possibility raises a new technical problem; traffic control of multiple robots. In this work, collision avoidance for moving robots is considered to be performed by the local controller of a robot using the wait and go rule, such that the robot dispatched later waits for a while if it senses that another robot passes the same location of the path.

4.3 Implementation of Algorithm

In this system, multiple robots first identify the maximum number of material parts to be loaded at their current positions. Then, they pick up and transfer the assigned material parts to the selected boxes. This procedure is repeated until all the material parts on the floor are moved into the boxes. The approximate implementation procedure of the proposed algorithms can be describes as follows.

[Implementation Procedure]
S1. Construct a graph network related to the locations of the objects on the floor.
S2. Assign boxes to material parts and select a robot to

carry material.

S3. Perform a sequence cycle of a robot motion

S3.1 Visit the first node of assigned material on the path

S3.2 Load the material

S3.3 Visit the next node and repeat the procedure S3.1-S3.2 until the robot buffer is full

S3.4 Go to the assigned node of a box

S3.5 Unload the material parts

S4. Repeat S3.3-S3.4 until all material parts are removed from the floor.

Computer simulation is carried out to show the validity of the presented system. A graphic simulator is designed to show the proposed system using C++ language code under PC with Pentium 400 Mz processor and Win 98 environment. Fig.2 illustrates the designed graphic simulator for the system.

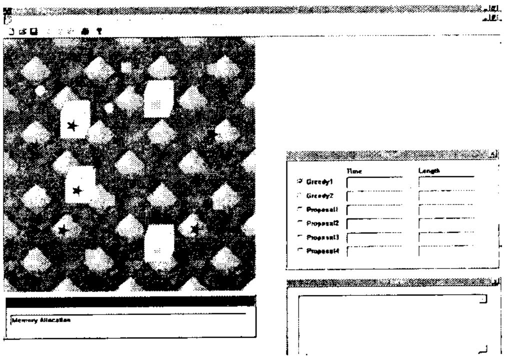

Fig. 2 Graphic simulator for the system

5. NUMERICAL RESULTS

There are several dominant parameters in the system. The following examples show how variation of a parameter and selection strategies affect the system performance. In all tests, service time of a robot such as loading or unloading material at a node is neglected in comparison with travel time for convenience. It means that zero time is associated with pick material from sources and placing material into sources. And, all of the mobile robots are assumed at moving at the same speed, which means that only the tour time of a robot for the task is considered in the test. Tests results are derived on average of ten trials for each example. In our suggested method in selecting and contracting material to move, we divide again the method SM into the followings from SM1 to SM4 considering the distance between the assigned material parts, distance from the box and the material and collision avoidance rule, respectively.

Table 2 Derived suggested methods

Method	Distances of materials	Distances of materials and a box	Collision Avoidance
SM1	O	X	X
SM2	O	O	X
SM3	O	X	O
SM4	O	O	O

[Example 1] Number of boxes

In this example, we investigate how the number of box affects the results. Although there are some differences according to the selection algorithms, it is observed that total task time reduces in proportion to the increase in this parameter.

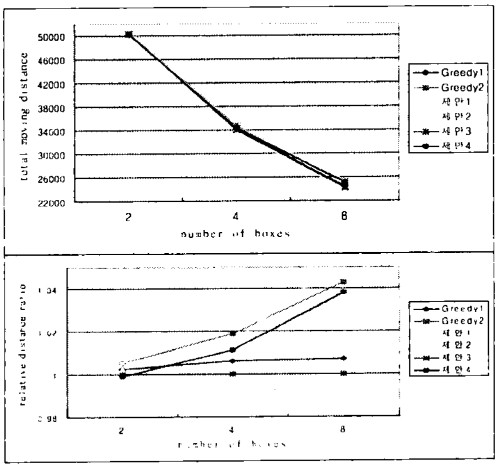

Fig.3 Results by varying the number of boxes, respectively

Table 3 Results by varying the number of boxes

$\|B\|$	2		4		8	
RM	77.9	1.55	76.2	2.24	72.0	2.99
GM1	50.4	1.00	34.2	1.00	24.3	1.01
GM2	50.5	1.00	34.7	1.02	25.2	1.04
SM1	50.4	1.00	34.0	1.00	23.8	0.99
SM2	50.5	1.00	34.6	1.02	25.4	1.05
SM3	50.3	1.00	34.2	1	24.1	1
SM4	50.2	0.99	34.4	1.01	25.1	1.04

(Here $|\lambda(P)| = |R| = C(b) = 2, |P| = 96, |B| = 2,4,8$)

[Example 2] Number of materials and their types

As the number of material parts and material types is increases, the total moving distance of a robot increases in proportion to it. In Fig. 4 and Fig.5, it is observed the performance of the suggested methods is quite better than the greedy search.

Table 4 Results by varying the number of material parts

$\|P\|$	32		64		96		128	
RM	24.2	2.56	48.4	2.81	72.0	2.99	96.8	3.19
GM1	9.9	1.04	17.5	1.02	24.3	1.01	31.1	1.02
GM2	10.3	1.09	18.0	1.05	25.2	1.04	32.0	1.05
SM1	9.6	1.01	16.9	0.98	23.8	0.99	30.6	1.01
SM2	10.9	1.15	18.6	1.08	25.4	1.05	31.9	1.05
SM3	9.5	1	17.3	1	24.1	1	30.4	1
SM4	10.8	1.14	18.3	1.06	25.1	1.04	31.7	1.04

($|\lambda(P)| = |R| = C(b) = C(r) = 2, |B| = 8, |P| = 32,64,96,128$)

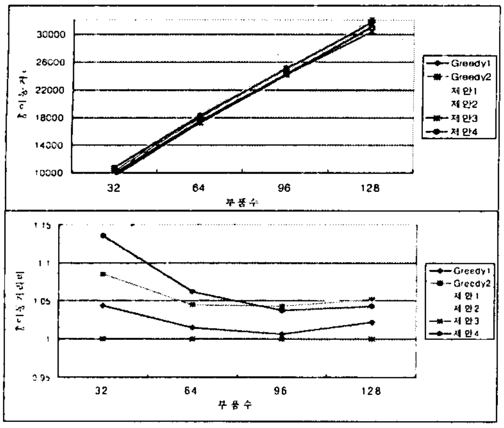

Fig.4 Results by varying the number of material parts

Table 5 Results by varying the number of material types

$\lambda(P)$	1		2		3	
RM	72.6	3.53	72.0	2.99	71.2	2.72
GM1	19.9	0.97	24.3	1.01	27.2	1.04
GM2	21.2	1.03	25.1	1.04	27.6	1.05
SM1	20.7	1.01	23.8	0.99	25.7	0.98
SM2	21.8	1.06	25.4	1.05	28.0	1.07
SM3	20.6	1	24.1	1	26.2	1
SM4	22.1	1.074	25.1	1.04	28.5	1.09

($|R| = C(b) = C(r) = 2$, $|B| = 8$, $|P| = 96$, $|\lambda(P)| = 1,2,3$)

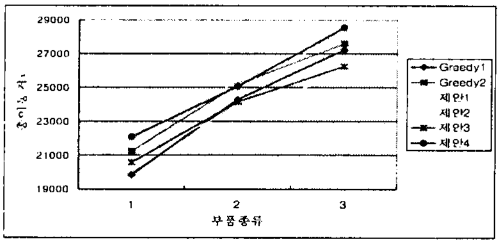

Fig.5 Results by varying the number of material types

[Example 3] Number of robots and their buffer size

As the number of robot increases, the total moving distance and task time decreases. It means the fact that the cooperation by multiple mobile robots is efficient. Also, buffer size of a robot affects the system together with the assignment strategies as shown in Fig. 6

Table 6 Results by varying the number of robots

$\|R\|$	1		2		3		4	
RM	72.2	2.96	72.0	2.99	72.9	3.08	71.4	3.05
GM1	24.3	1.01	24.3	1.01	24.1	1.02	24.0	1.02
GM2	24.1	1.03	25.2	1.04	25.0	1.06	24.9	1.06
SM1	24.0	1.00	23.8	0.99	23.8	1.00	23.4	1.00
SM2	24.9	1.03	25.5	1.05	24.9	1.05	24.5	1.04
SM3	24.1	1	24.1	1	23.7	1	23.5	1
SM4	24.9	1.04	25.1	1.04	25.2	1.07	24.7	1.05

($\lambda(P) = \lambda(R) = 2, |P| = 96, |B| = 8, |R| = 1,2,3,4$)

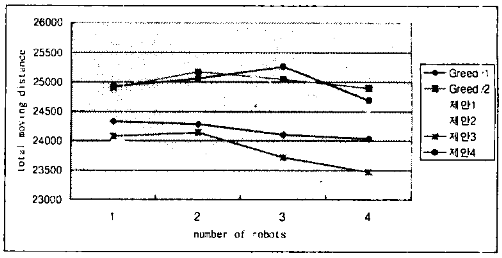

Fig.6 Results by varying the number of robots

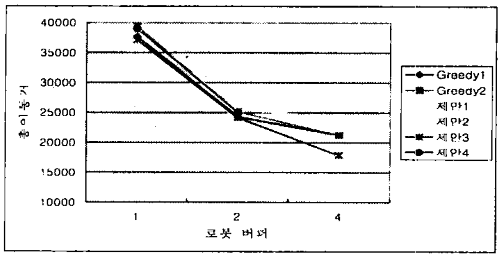

Fig.7 Results by varying varying the buffer size of a robot

Table 7 Results by varying the size of a robot buffer

$C(r)$	1		2		4	
RM	99.0	2.26	72.0	2.99	58.7	3.298
GM1	37.6	1.02	24.3	1.01	21.2	1.19
GM2	39.4	1.06	25.2	1.04	21.2	1.19
SM1	37.1	1.00	23.8	0.99	18.1	1.01
SM2	39.0	1.05	25.4	1.05	24.3	1.37
SM3	37.1	1	24.1	1	17.9	1
SM4	39.0	1.05	25.1	1.04		

($|\lambda(P)| = |R| = 2, |P| = 96$, $|B| = 8$, $|C(r)| = 1,2,4$)

6. CONCLUSION

In this paper, we have shown the possibility to use multiple mobile robots for collecting scattered material. We present a general problem of optimization such that mobile robots a number of tasks to be performed for some minimum cost. We demonstrated that satisfactory solutions are obtained through rigorous search methods. It is observed that random search requires about three times moving distance in comparison with other methods through the tests. The proposed methods from SM1 to SM4 are kinds of the method based on exhaustive search. The search space of them increases exponentially according to the increase in size of robot buffer and etc. So as to reduce this search effort, in can be an alternative idea to restrict the search area of robot based on the current robot position. The parameters of the proposed system, such as buffer size, number of kinds of material parts and number of box and etc., affect the performance of selection algorithms. Therefore, suitable choice of assignment algorithm is important in reducing the task time of the system.

The formulated problem and solution approach are divided into two kinds of problems: assignment problem and planning problem of robot motion. It is similar somewhat to problems on the optimization of computer resources, of which sub-problems fall into two categories: storage and networking. However storage allocation typically deals with a one-dimensional problem instead of two-dimensional floor [7]. For networking, the relevant work involves both geographically distributed systems and multiprocessor supercomputer systems[8]. Instead, the problem of the proposed system is more like to problem on optimization of the problem of a multi-head surface mounting machine in two-dimensional space, of which sub-problems fall into two categories: an assignment problem of resources and pick-and-place sequencing motion planning one of the machine[9].

References

[1] R. Mann, W. Hamel and C. Weisbin, "The Development of an Intelligent Nuclear Maintenance Robot ", *1988 IEEE Int. Conf. Robotics and Automation*, pp.621-623, 1988.

[2] R. Carlton and S. Bartholet, "The Evolution of Application of Mobile Robots to Nuclear Facility Operations and Maintenance", *1987 IEEE Int. Conf. Robotics and Automation*, pp.720-726, 1987.

[3] P. P. Bose, "Basics of AGV systems", *Amer. Mach. Automat., Manuf.*, vol. 130, no.3, pp. 105-122, Mar., 1986.

[4] D. Newton, "Simulation model calculates how many automatic guided vehicles are needed", *Indust. Eng*, vol. 14, 17, no.2, pp.68-78, Feb., 1985.

[5] R. Chattergy, "Some heuristics for the navigation of a robot", *Int. J. Robotics Res.*, vol. 4, no.1,pp 59-66, Spr., 1985.

[6] S. K. Kambhampati and L. S. Davis, "Multiresolution path planning for mobile robots", *IEEE J. Robotics Automation*, vol. RA-2, no.3,pp 135-145, Sept., 1986.

[7] D. C. Grossman and H. Silverman, "Placement of records on a secondary storage device to minimize access time", *J. ACM*, vol. 20, no. 3, pp. 429-438, July, 1973.

[8] D. P. Agrawal, V. K. Janakiram, and G. C. Pathak, "Evaluating the performance of multicomputer configurations", *Computer*, vol. 19, no.5, pp. 23-37, May, 1986.

[9] W.S. Lee, S. H.Lee, B.H. Lee and Y. D. Lee, "A Genetic Approach to Operation of a Multi-head Surface Mounting Machine", *IEICE Trans. Fundamentals*, vol. E84-A, pp. 1748-1756, Sept.,2000.