Reconnaissance into Distant Spaces by Multi-Robot Rescue System with Ad Hoc Networking

† Hisayoshi Sugiyama † Tetsuo Tsujioka ‡ Masashi Murata
† Dept. of Physical Electronics and Informatics, Osaka City University, Osaka, Japan 558-8585
‡ Faculty of Science and Technology, Ryukoku University, Otsu, Japan 520-2194
Email: sugi@info.eng.osaka-cu.ac.jp

***Abstract*—The system procedure of a victim detection system (VDS) is investigated to reconnoiter distance spaces. VDS is a previously proposed multi-robot rescue system, which searches for victims in urban disaster areas, that consists of mobile robots and one monitor station with operators. An ad hoc wireless network, which connects all of them, is continuously maintained by autonomous path recovery (APR) executed by each robot. The proposed VDS procedure adopts autonomous classification of robots into search and relay robots that act based on the behavior algorithms of each class of robot to configure chain networks threading a path to the spaces. The rules of classification and the behavior algorithm refer to the forwarding tables of each robot constructed for ad hoc networking. Simulation results show that VDS can reconnoiter distant spaces autonomously even though two or three spaces exist beyond different corridors stemming from the safety zone where the monitor station is established.**

***Index Terms*—multi-robot, USAR, adhoc network, rescue system.**

I. Introduction

Urban search and rescue (USAR) is becoming one of most urgent investigation subjects due to such disasters as earthquakes, hurricanes, or terrorism against manmade constructions in urban areas [1]. We previously proposed a USAR multi-robot scheme called the victim detection system (*VDS*) [2] that consists of mobile robots and one monitor station (*MS*) with human operators. An ad hoc mobile network based on DSDV [3], which connects all of them, is continuously maintained by autonomous path recovery (APR) executed by each robot. Robots are instructed to search for victims in disaster areas autonomously, while operators at stationary MS monitor information sent from each robot.

In this paper, we investigate a VDS procedure to reconnoiter distance spaces, which are assumed to exist within unknown regions of buildings and that might be distant from the safety zone where the MS is established. The procedure consists of the autonomous classification of robots into search robots (*SR*) and relay robots (*RR*) and behavior algorithms of each class of robot (this method is referred to as *search and relay robot classification* or *SRRC*). SRs reconnoiter the distant spaces, and RRs act as relay terminals between SRs and the MS. They autonomously configure a chain network threading paths to distant spaces even within unknown regions.

II. Search and relay robot classification

Robots must configure a chain network to reconnoiter one of the spaces, and then the chain is transformed to reconnoiter another one. These processes must be repeat until all spaces have been searched.

To allow VDSs to execute these processes, we propose search and relay robot classification (SRRC) in combination with a decision-theoretic approach [4]. In SRRC, robots classify themselves autonomously into SRs and RRs. SR is one whose position is the endpoint of the network. It recognizes its position based on a predetermined rule called the *rule of SR/RR classification*.

RR recognizes itself as not being an SR. Each RR selects its master search robot (*MSR*) among the SRs in the network based on a predetermined rule called the *rule of MSR selection*. RR acts as a relay terminal within the path from its MSR to MS and moves appropriately to maintain the path based on a behavior algorithm.

As mentioned above, there are two rules in SRRC processing: SR/RR classification and MSR selection. In addition, a behavior algorithm is specified that each robot follows depending on its class.

A. SRRC processing rules

The rules for SRRC refer to the forwarding table that each robot stores for the routing process of a DSDV ad hoc network.

1) Forwarding table: Figure 1 shows a VDS with nine robots A∼I and an MS. Among these, robot C is selected to show an example of a forwarding table in Fig. 2. The table consists of entries for each destination (denoted by *dst*). As shown here, elements are classified into transmitted ones and a derived one. The former denotes directly updated elements by update packets, and the latter denotes the element derived from the former ones. In addition, the former is subclassified into inherent elements in DSDV and the introduced ones for SRRC processing (from now called *DSDV elements* and *SRRC elements*, respectively).

DSDV elements consist of *dst*, *nxt*, and *hop*. *nxt* and *hop* indicate the next hop to *dst* and the hop count to there, respectively. In particular, the next hop recorded in the entry for the MS is called *NMS*. The NMS of robot C is robot B, as shown in the first row.

ROBOCOMM 2007, 14th – 16th Oct 2007, Athens, Greece.

ISBN 978-963-9799-08-0
DOI 10.4108/ICST.ROBOCOMM2007. 2105

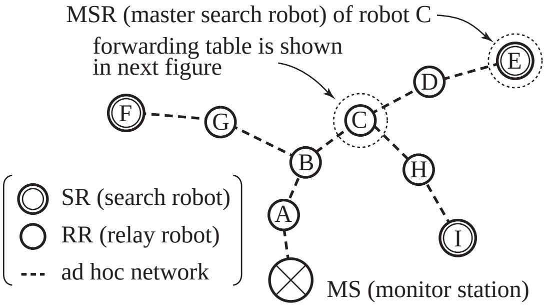

Fig. 1. Example of VDS

NMS (next hop to MS)

dst	nxt	hop	location	inv	cls	msr
MS	B	3	(223.5, 466.8)	–	–	
A	B	2	(210.9, 502.1)		RR	
B	B	1	(222.4, 533.6)		RR	
D	D	1	(293.6, 574.4)	*	RR	
E	D	2	(114.4, 568.4)		SR	*
F	B	3	(151.6, 564.4)		SR	
G	B	2	(189.6, 554.4)		RR	
H	H	1	(271.6, 519.6)	*	RR	
I	H	2	(288.4, 485.6)		SR	

transmitted (dst–cls); derived (msr)
DSDV (dst–hop); SRRC (location–cls)

Fig. 2. Forwarding table of robot C

SRRC elements consist of *location*, *inv*, and *cls*. *location* indicates where the destination is located in the global coordinate system [5]. *inv* indicates an inverted NMS that indicates which robot recognizes robot C as its NMS. *cls* shows the classification results of each robot destination.

Each robot transmits its own *location* and *cls* to all other robots by attaching them to update packets, but it only transmits its NMS to neighboring robots. This information makes *inv* of the neighboring robots.

Derived element *msr* indicates the MSR of robot C if C is classified as an RR. *msr* is selected among SRs listed in the *cls* column.

2) SR/RR classification rule: Classification depends on column *inv* in its forwarding table. If nothing is listed in the column, then the robot recognizes its position as an endpoint and classifies itself as an SR. On the other hand, if one or more indexes are listed in *inv*, then the robot recognizes that its position is not an endpoint and classifies itself as an RR.

In Fig. 1, because robots E, F, and I find no indexes in their *invs*, they classify themselves as SRs. However, since other robots have one or more indexes such as robot C shown in Fig. 2, they classify themselves as RRs.

3) MSR selection rule: Each RR selects its own MSR among the SRs in the network and moves appropriately to maintain the path between the MSR and MS. These robots must configure a chain network where MSR and MS terminate at both endpoints and RR is at some intermediate.

Satisfying this requirement, the MSR of each RR must be the furthest robot from MS because the furthest one might be the front of the chain network and RRs must support this SR to extend the chain as long as possible.

The MSR selection rule concerning these requirements is as follows:

1) Among the SRs indicated in the forwarding tables as destinations, discard those for which the next hop coincide with NMS because SR and MS beyond the same next hop configure an other chain that does not include the RR.

2) Among SRs remaining from the above screening, select one with the largest distance from MS. This distance is measured along the optimal path avoiding obstacles if they exist between each SR and MS.

For example, robot C in Fig. 1 selects robot E as its MSR by the following selection process:

1) Robot C recognizes robots E, F, and I as SRs because they are indicated as SRs in the *cls* of its forwarding table shown in Fig. 2.

2) Among these SRs, robot F is discarded based on the first item of the MSR selection rule because the next hop to F and NMS is the same robot B.

3) Robot C calculates the distances from MS to robots E and I depending on their locations indicated in the forwarding table and the global map of the explored region so far. Then, based on the second item of the MSR selection rule, robot E is finally selected as the MSR of robot C.

B. Behavior algorithm of robots

After the SR/RR classification finishes, each robot acts based on the predetermined behavior algorithm to configure a chain network to reconnoiter one of the spaces and transform it to reconnoiter another one. The details are as follows.

Each RR acts to keep the link to the next hop to its MSR (denoted by *NMSR*). Simultaneously, it acts to keep the link to its NMS. This behavior of RR depends on whether the distance d_{out} to NMSR exceeds the threshold r_{out} and on whether the distance d_{in} to NMS exceeds the threshold r_{in}. r_{out} and r_{in} are called *outer link recovery radius* and *inner link recovery radius*, respectively. These values are fixed in relation to the transmission radius r_t as $r_{in} < r_{out} < r_t$. Because the link to NMS is assumed to be more essential, r_{in} is set smaller than r_{out}. The distances d_{out} and d_{in} can be obtained from the power level of the signal received from NMSR and from NMS, respectively.

On the other hand, each SR moves to its next target. This target is selected according to the decision-theoretic approach [4]. In this approach, Each SR selects the new target among the frontier cells according to the inherent algorithm when it arrives the current target. Besides the movement to the next target, each SR simultaneously acts to keep the link to its NMS when d_{in} exceeds r_{in} as RR does.

The behavior algorithm that each robot follows is shown in Fig. 3 as a flowchart. In this flowchart, the *potential NMS* means the neighboring robots that have paths to MS and are not the NMS yet.

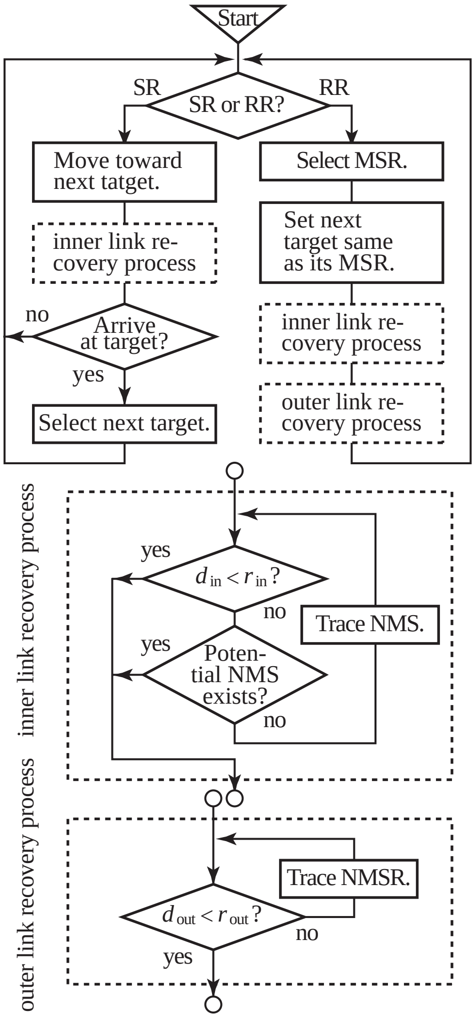

Fig. 3. Behavior algorithm of each robot

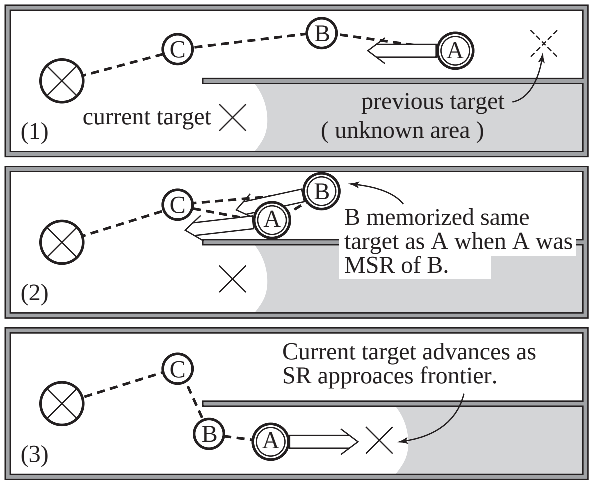

Fig. 4. Transformation of chain network

C. System operation of VDS

At the beginning of system operation, robots are set close around the MS and autonomously perform ad hoc networking. When the network is completed, each SR determines its next target and moves to there. Because the farthest SR from the MS possibly be selected as the MSR of many RRs, this SR and RRs gradually configure a chain network threading a corridor to a distant space.

When the space is searched, the SR updates the next target and the chain network transforms itself to reconnoiter another space beyond the new target. This process of the chain network transformation is explained with Fig. 4 where two blind corridors only exist for the simplicity.

The part (1) of this figure shows when the SR (robot A) reaches the end of a corridor and its next target is updated to one of frontier cells. This target locates in front of unknown region of another corridor indicated by *current target*. At this moment, RRs (B and C) also update their target to the same point as their MSR (A). In the part (2), robot B becomes SR after some movement of A and, because B memorizes the same target as A, both robots moves to the same next target. In the final part (3), as the SR (robot A) approaches the current target within frontier cells, the frontier advances along with the corridor and other robots follow A to maintain the chain network.

III. Computer simulation

Reconnaissance into distant spaces by VDS with the proposed SRRC system procedure was confirmed by computer simulation[1]. The simulation model is shown in Fig. 5. The three spaces indicated by cells 1∼3 exist beyond corridors stemming from a safety zone where an MS is established. A barrier indicated by dashed lines is set when a case with two spaces (cells 1 and 2) is simulated.

Reconnaissance into these cells is estimated by *search rate* and *search time* for each cell. Search rate denotes the percentage of the scanned area to the whole area of each cell, and search time denotes the elapsed time of system operation when the search rate achieves 100%. Here, a point in the cell is assumed to be scanned when the sensor range of a robot includes the point and simultaneously the robot has a path to the MS.

Figure 6 shows the simulation results when the barrier is set in Fig. 5 and the chain transformation from targeting cells 1 to 2 is estimated. The horizontal axis represents the elapsed time of the system operation, and the vertical axis represents the search rate for each cell. Because the chain transformation of one direction is concerned, simulation results indicating the other case (cells 2 to 1) are discarded, and the remaining results are averaged over many simulations.

The normal moving speed of a robot and its accelerated speed when it traces another are set to 1 and 2 [km/h], respectively. The transmission rate of the network is assumed to be high enough that packet collisions do not occur. The

[1]The simulator is constructed based on the modules of an autonomous multi-robot simulator [6] and a DSDV ad hoc network simulator [7].

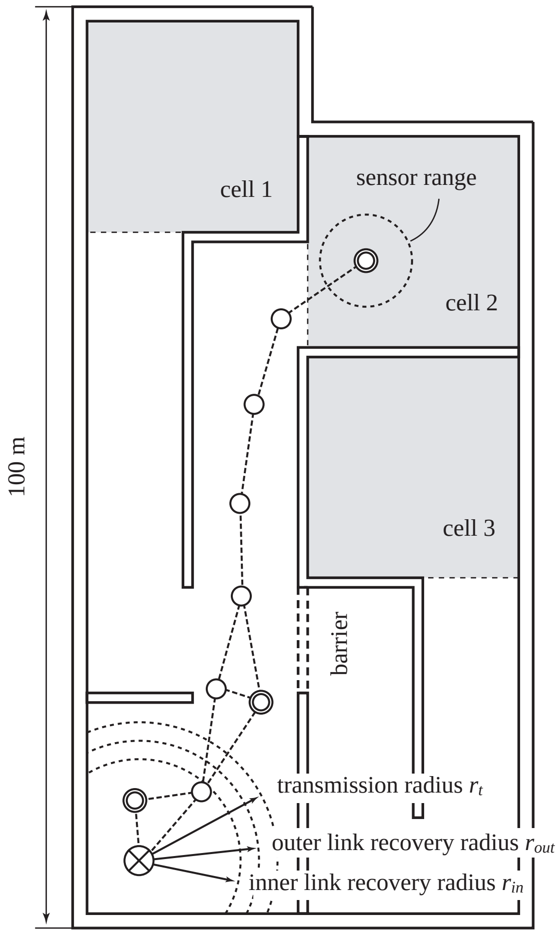

Fig. 5. Simulation model of VDS with SRRC

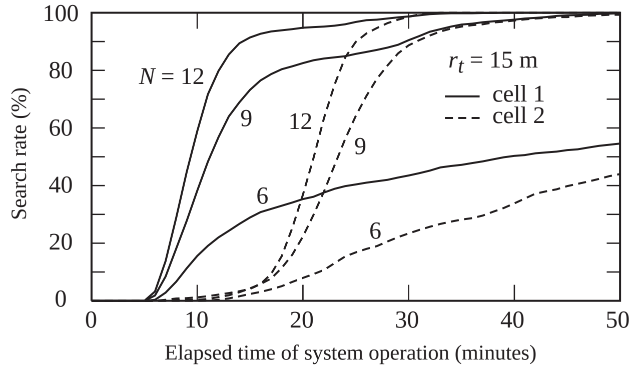

Fig. 6. Search rate for cells 1 and 2

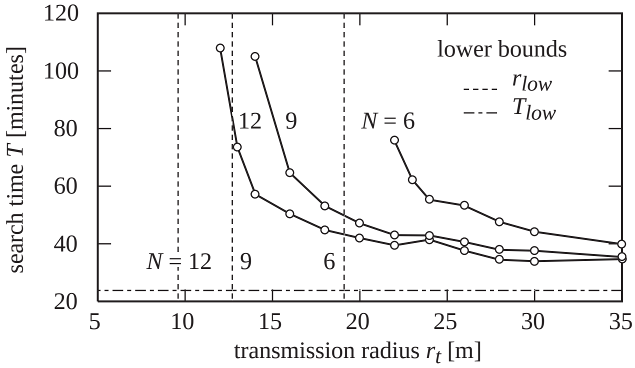

Fig. 7. Search time for all cells

frequency of forwarding table updates is set to ten times per second.

If the number N of robot is enough to reconnoiter either of the cells, first, the chain network accomplishes reconnaissance into cell 1, and second, it transforms to reconnoiter cell 2. The graphs at N *equals 12* almost correspond with this case. On the other hand, when N *equals 6*, because the number may not be sufficient, reconnaissance into a cell is not accomplished in one time. In this case, the transformation is repeated many times and the search rate for either cell hardly increases. The graphs at N *equals 9* show the intermediate case.

Figure 7 shows search time T for all cells 1$\sim$3 versus transmission radius r_t. The barrier is eliminated from the simulation model. The lower bounds r_{low} of r_t and T_{low} of T are indicated. This figure shows that transmission radius r_t possibly compensates for the lack of robots. In the previous Figure 6, N must be large enough to reconnoiter distant spaces. However, when r_t exceeds about 30 [m] in Fig. 7, search time T approaches lower bound T_{low} regardless of N because fewer RRs are necessary to reconnoiter all cells if r_t is set large enough.

On the other hand, as r_t falls below 30 [m], T gradually increases and a difference between N appears. However, until r_t approaches lower bound r_{low} of each N, it is confirmed that reconnaissance into all cells is possible within a reasonable time.

IV. Conclusion

An SRRC system procedure of VDS was investigated to reconnoiter distant spaces within unknown regions. The performance of the procedure was confirmed by computer simulations.

References

[1] R. R. Murphy, “Trial by fire,” *IEEE Robot. Autom. Mag.* , vol. 11, no. 3, pp.50-61, Sept. 2004.

[2] H. Sugiyama, T. Tsujioka, and M. Murata, “Victim detection system for urban search and rescue based on active network operation,” *Design and Application of Hybrid Intelligent Systems, IOS Press*, pp.1104-1113, 2003.

[3] C. E. Perkins and P. Bhagwat, “Highly Dynam-ic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers,” *ACM SIG-COMM’94*, pp.234-244, 1994.

[4] W. Burgard, M. Moors, C. Stachniss, and F. Schneider, “Coordinated multi-robot exploration,” *IEEE Trans. Robot. Autom.* , vol. 21, no. 2, pp. 376-386, June 2005.

[5] D. Fox, J. Ko, K. Konolige, B. Limketkai, D. Schulz, and B. Stewart, “Distributed Multirobot Exploration and Mapping,” *Proc. of IEEE* , vol. 94, no. 7, pp.1325-1339, July 2006.

[6] H. Sugiyama and M. Murata, “AMRS: An Autonomous Mobile Robot Simulator,” *Memoirs of the Faculty of Engineering, Osaka City Univ.*, pp.105-121, 1996.

[7] H. Sugiyama, T. Tsujioka, and M. Murata, “Ad Hoc Network Simulator Based on DSDV Routing Method,” *Memoirs of the Faculty of Engineering, Osaka City Univ.*, pp.43-58, 2002.