Performance Analysis of Entropy-based Multi-Robot Cooperative Systems in a MANET

Sang-Chul Kim, Kee Hyun Shin*, Chong-Woo Woo, Yun Shick Eom, and Jae Min Lee

Abstract: This paper proposes two novel algorithms enabling mobile robots to cooperate with each other in a reliability-based system and a time-critical system. In the reliability-based cooperative system, the concepts of a mobile ad hoc network (MANET) and an object entropy are adopted in order to coordinate a specific task. A logical robot group is created based on the exchange of request and reply messages in a robot communication group whose organization depends on transmission range. In the time-critical cooperative system, relational entropy is used to define the relationship between mobile robots. A group leader is selected based on optimizing power consumption. The proposed algorithm has been verified based on the computer-based simulation and soccer robot experiment. The performance metrics are defined. The metrics include the number of messages needed to make a logical robot group and to obtain the relationship of robots and the power consumption to select a group leader. They are verified by simulation and experiment.

Keywords: Autonomous robots, entropy, message complexity, mobile ad hoc networks, robot cooperation.

1. INTRODUCTION

A significant amount of research about systems of multiple mobile robots has been conducted over the past decade. In particular, the cooperative aspect of such systems has been of interest. Cooperation of robots refers to a situation where multiple robots operate to perform a task that either cannot be achieved by a single robot, or whose execution can be improved by using more than one robot, thus obtaining higher performances [1].

Past robot research focused on improving the performance of a single robot. Attention is now being switched to research on multiple mobile robots Systems, and, in particular, to cooperative aspects of the systems. Multiple robot systems have many advantages over a single-robot System, if the robots in the system can cooperate in an effective and efficient manner.

It is possible that one robot may fail to complete its task in a timely manner. The other robots could be kept waiting, delaying or causing the mission to be incomplete [2]. A team of simple and cheap robots can replace a single complex and expensive one [3]. Interference and collisions may occur with higher probability in multi-robot systems than in single-robot systems. This can overshadow the efficiency of multi-robot systems.

To overcome this problem, each robot has to evaluate the feasibility of its behaviors according to environmental conditions. The system needs an organized set of regulations. Cooperative systems are classified in four groups [3]: (1) distributed and unconscious; (2) distributed and conscious; (3) intermediate and hierarchical; (4) centralized.

The proposed multi-robot cooperative system uses a set of heterogeneous mobile robots. The system employs a decentralized autonomous system. In this paper, an entropy concept from information theory is used particularly in robot cooperation. The concept can be extended to relations between humans and robots.

MANETs have been emphasized as an emerging research area due to the growing demands for mobile and pervasive computing, where the dynamic topology for the rapid deployment of independent mobile users such as mobile robots becomes a factor to be considered. One of the outstanding features of

---

Manuscript received June 14, 2007; revised April 25, 2008; accepted June 9, 2008. Recommended by Editorial Board member Jang Myung Lee under the direction of Editor Jae-Bok Song. This work was supported by the Research Program of Konkuk University in 2004.

Sang-Chul Kim are Chong-Woo Woo are with the School of Computer Science, Kookmin University, 861-1, Chongnung-dong, Songbuk-gu, Seoul 136-702, Korea (e-mails: {sckim7, cwwoo}@kookmin.ac.kr).

Kee Hyun Shin, Yun Shick Eom, and Jae Min Lee are with the School of Mechanical Engineering, Konkuk University, 1 Hwayang-dong, Gwangjin-gu, Seoul 143-701, Korea (e-mails: khshin@konkuk.ac.kr, {redwt, ljm1918}@naver.com).

* Corresponding author.

MANETs could be the self-creating, self-administrating, self-organizing, and self-configuring multihop wireless network characteristic. MANETs differ from conventional cellular networks, because all links are wireless and mobile users inter-communicate without a base station. A MANET can be represented as an undirected graph $G(\mathbf{V},\mathbf{E})$ where $\mathbf{V}$ is a finite nonempty set of nodes that can be represented as $\mathbf{V}=\{V_1^G,V_2^G,\cdots,V_W^G\}$ where $|\mathbf{V}|=W$ and $\mathbf{E}$ is a collection of pairs of distinct nodes from $\mathbf{V}$ that form a link that can be represented as $\mathbf{E}=\{E_1^G,$ $E_2^G,\cdots,E_W^G\}$. A connected, acyclic, undirect-ed graph that contains all nodes is defined as a free tree.

Fig. 1 shows MANET nodes at an instant where it is composed of six partitioned sub-graphs. One important issue in MANETs is the time-varying network topology, which may be unpredictable over time. MANET routing algorithms keep updating neighbor discovery and inform nodes of the network topology change with node mobility.

The reduction of routing overhead is a major concern when a MANET routing protocol is developed. An essential measure of the quality of a MANET routing protocol is scalability with an increase of the number of MANET nodes. Message complexity is measured in terms of the number of messages needed to satisfy the algorithm's request.

Shen [4] statistically measures performance of cluster-based topology control (CLTC) protocol using message complexity. The authors in [5] calculate storage complexity and communication complexity to analyze the scalability of various MANET routing protocols and introduce the routing updated LS messages. These follow the order of $O(N^2)$, where N indicates the number of nodes in a MANET.

This paper is organized as follows: Section 2 explains the study of MANETs that are applied to mobile robots. Section 3 describes a detailed explanation of the proposed algorithm, particularly performing a logical robot grouping, to select a group leader and to design the multi-robot reliability-based and time-critical cooperative systems. Section 4 details the numerical experiments and results. Finally, Section 5 summarizes our work and concludes this paper.

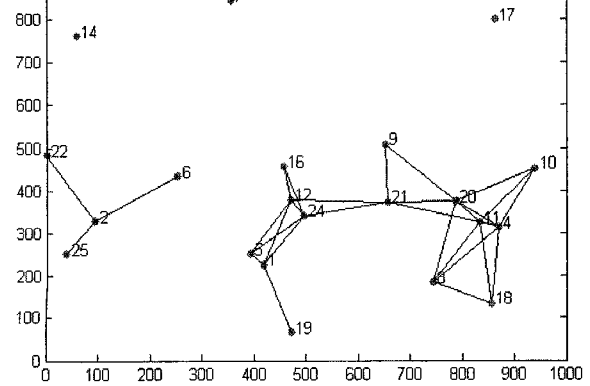


Fig. 1. Mobile ad hoc network (MANET).

2. RELATED WORKS

A fault-tolerant configuration algorithm [6], which makes the robots communicate with one another is proposed, for situations where a MANET is used as the communication method of mobile multi-robot systems. Time complexity measures performance of the proposed algorithm. Winfield used a MANET for robot communication [7]. Robots distributed in an area use wireless transmitters and receivers to transmit and receive sensor data. Various ad hoc protocols were introduced for an efficient data transmission that requires less power than other protocols [8].

Mobile robot distance vector (MRDV) and mobile robot source routing (MRSR) were developed for efficient routing among mobile robots [9]. These protocols have less overhead compared to traditional dynamic source routing (DSR) and ad hoc on demand distance routing (AODV) protocols. An algorithm is proposed to control the mobility of robot to improve communication. The scheduling adopted controls the robot mobility that blocks communi-cation [10].

The authors of [11] define the concept of trust to define a performance metric in a MANET. Trust is represented by entropy, to evaluate trustworthiness of other nodes without centralized authority. Based on recommendations from other nodes where messages include metrics that indicate some nodes have low trust values, a node can obtain a trust relationship. The trust based model can be used for secure ad hoc routing and malicious node detection. The proposed system improves the network throughput as well as detecting malicious behavior.

3. PROPOSED ALGORITHM

The proposed algorithm is composed of two systems, a reliability-based cooperative system and a time-critical cooperative system.

3.1. Reliability-based cooperative system

The literature review indicates that each mobile robot needs to be self-creating, self-administrating, self-organizing, and self-configuring in performing cooperative work. Based on this, MANET technology can be adopted where a MANET node represents a mobile robot. This section introduces the mathematical background of robot grouping to perform a reliability-based cooperative work. In this work, time is not critical; a node is represented as a mobile robot and a route is represented as a robot group. The authors of [12] use the concept of entropy to measure route stability of a MANET.

In this paper, the route stability [12] of a MANET is redefined to perform robot grouping and to calculate group stability, since a route is composed of several member nodes between source and destination nodes and each node in a route has its own value of object entropy (O_E). The most stable route gives the most stable robot grouping. Hence, the robot grouping algorithm (GA) can be derived as below [12].

The velocity vectors of mobile robots m and n are denoted by $v(m,t)$ and $v(n,t)$ respectively at time t. The relative velocity $v(m,n,t)$ between mobile robots m and n at time t is defined as [12],

$$v(m,n,t) = v(m,t) - v(n,t). \tag{1}$$

A variable $a_{m,n}$ can be defined, where N is the discrete number in a time interval Δ_t.

$$a_{m,n} = \frac{1}{N}\sum_{i=1}^{N} |v(m,n,t_i)| \tag{2}$$

Based on the relative velocity and the variable $a_{m,n}$, O_E of mobile robot m of time interval Δ_t, which is expressed as $H_m(t,\Delta_t)$, can be represented as [12],

$$H_m(t,\Delta_t) = \frac{-\sum_{k\in F_m} P_k(t,\Delta_t)\log P_k(t,\Delta_t)}{\log C(F_m)}, \tag{3}$$

where $P_k(t,\Delta_t) = a_{m,k}/\sum_{i\in F_m} a_{m,i}$, F_m is a set of neighboring robots of a mobile robot m and $C(F_m)$ is a degree of the set F_m.

If the value of $H_m(t,\Delta_t)$ is small, the robot m is stable, if the value of $H_m(t,\Delta_t)$ is large, the robot m is unstable. In general, MANET consists of several member robots between source and destination robots. The grouping algorithm can be represented as

$$\gamma^j = GA_{k,l}^j(t,\Delta_t) = \prod_{i=1}^{N_r}[H_i(t,\Delta_t)]\Bigg|_j, \tag{4}$$

where N_r indicates the total number of member robots including the source robot (k) and destination robot (l) of a single robot grouping j.

Based on $\gamma^j = GA_{k,l}^j(t,\Delta_t)$, O_E of each member robot is multiplied and represented as γ^j. Then γ^j will be used to select the robot group has the best γ value among several robot groups, for a specific task.

Several robot groups in a robot society can undertake a specific task. After calculating the γ value of each robot group, the group having the lowest γ value is selected. The member robots of the group will perform the specific task since the group has the maximum potential capability to perform the task.

3.2. Time-critical cooperative system

This section proposes another concept to define relationship among mobile robots, since time-critical cooperative work among robots is constructed based on robot's relation status, as well as robot's own status.

The robot's own status is represented as an O_E and the relationship between robots is represented as a relational entropy (R_E).

Even though the robot's own status may be good, the relationship between two robots, R_E, may not be good. The cooperation should be reconsidered to achieve efficient task utilization. This is analogous to human teams cooperatively performing tasks.

Analyzing two systems, such as time-critical and reliability-based cooperative systems, explains the need of R_E where the number of hops (H) is defined as R_E.

The number of hops is calculated as

$$H = \lceil D/T \rceil, \tag{5}$$

where D indicates the distance from a source robot to a destination robot and T indicates the transmission range of robots. Since the value of H is assigned to a robot with uncertainty based on the random values of D and T, H is used, for simplicity in this paper, as the R_E among many possible metrics. The value of H can be normalized when it is divided by the largest number of hops in a robot group.

3.3. Group leader selection giving optimized power consumption

Power consumption in battery-operated robots is an important factor to be considered. The group leader might be overloaded compared to the member robots. It must control the member robots' behavior, sending control messages. Scalability should be considered to reduce message generation, and so doing prolong battery life for the robots including the group leader.

The binary integer programming (BIP) equation for the optimum power consumption configuration will be introduced, where robots are required to send information packets to their peer respective receiver robots, subject to a constraint on the signal to-interference-and-noise ratio (SINR). The objective is to minimize the total power of the robot cooperative communication system, when all robots transmit signaling packets initiated from a source robot. This can be represented as

$$\begin{aligned} &\textit{minimize} \sum_{m\,links} P_{xy} \\ &\textit{subject to} \\ &\mathrm{SINR}_{xy} \geq \beta \\ &\sum_{m\,links} R_{xy} = N_r - w,\ \forall_{m\,links} \in N_r - w, \end{aligned} \tag{6}$$

where P_{xy} indicates the power used between peer member robots x and y in a robot group j, β indicates the required SINR threshold for the desired quality of communication, the binary integer variable R_{xy} represents the connection status between member robots x and y. $R_{xy}=0$ when robots x and y are not connected and $R_{xy}=1$ when robots x and y are connected. N_r indicates the total number of member robots in the robot group j.

Due to the transmission range and robot mobility of member robot, the robots that are not included in robot groups in a MANET are not considered. They are located out of the transmission range of the member robots of each robot groups, represented as w. A transmission involving m links is admissible iff there is a set of transmission powers, $P_{xy} \geq 0$.

The transmitted power from a robot x to a robot y can be described as

$$P_{xy} = d(x,y)^4 \cdot P_r, \tag{7}$$

where $d(x,y)$ is the distance between the peer robots x and y, and P_r is the minimum power required by the receiver robot y. For example, the receiver sensitivity in *Bluetooth* is -70dBm. It is used as the minimum power required in this simulation [13]. 250m is used as the maximum transmission range between two nodes.

Group Leader Selection Algorithm

When the *MIN* function yields the minimum number of robots in a robot group, $MIN(s_1,s_2,\cdots,s_n)$ represents $min_i\{s_i\}, i=1,2,\cdots,n$ where multiple source robots and group leaders of multiple robot groups are considered. $MIN(d_1,d_2,\cdots,d_m)$ can be represented as $min_u\{d_j\}, u=1,2,\cdots,m$.

Definition 1: When s is defined as the number of robots in a robot group (a communication path) from a source robot to a group leader and the variable d is defined as the number of robots in a robot group from a group leader to a destination robot, a group leader is selected with the constraint of

$$min_i\{s_i + min_j\{d_j^i\}\} i=1,2,\cdots,n, j=1,2,\cdots,m, \tag{8}$$

where n indicates the total number of source robots. The selected group leader gives the lower bound of the minimum number of messages, when a source robot initiates a message, where m indicates the total number of member robots in a robot group.

4. SIMULATION AND EXPERIMENT

A computer-based simulator has been developed to analyze the reliability-based and time-critical cooperative systems. A standalone mobile robot environment is implemented, where the robots have no connection to an external network, such as the Internet. The discrete-event simulator was developed in *Matlab* to verify the various network topologies and to calculate message complexity [14] and power consumption. Fig. 2 shows the algorithm implemented in the simulator for the reliability-based cooperative system introduced in Section 3.1. In the computer-based simulator, robots are randomly distributed with uniform density in a network area of 1km^2. The

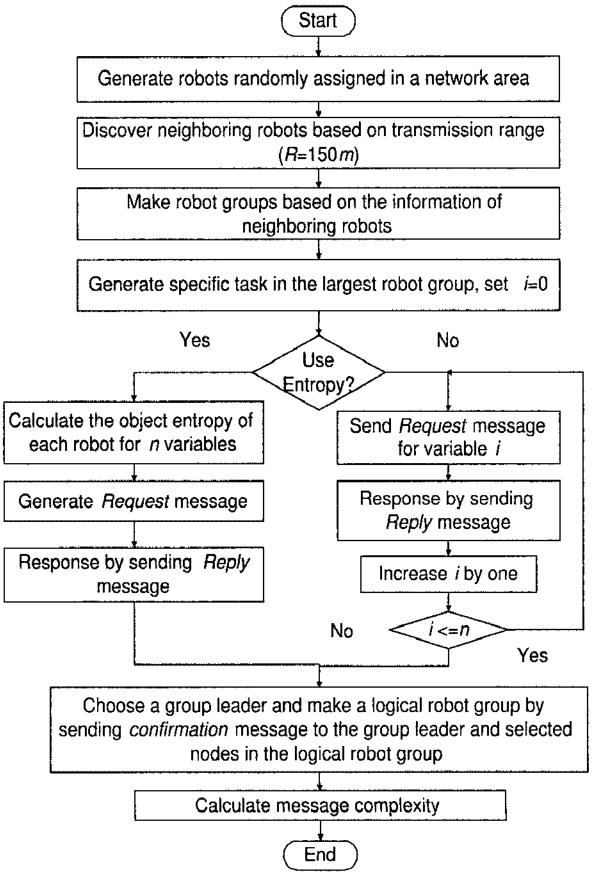


Fig. 2. Algorithm for a reliability-based cooperative system.

transmission range (*T*) of a robot is defined as 150m.

4.1. Reliability-based cooperative system

The simulator comprises several modules (Fig. 2). In this paper, variable status of a robot, which represents its own power, mobile speed, the capability for a certain task is represented as a single O_E. Equation (3) represents O_E composed only of the speed variable.

First, a random node generator in the simulator generates random robots. Each robot detects its neighboring robot based on the transmission range. This constitutes a communication robot group (C_R). After the group is constructed, only a single robot, which needs to obtain cooperation from other robots, is selected to trigger a specific task and transmits a *request* message to the group.

Other robots in the group check the required value of O_E, which is a threshold, from the source robot. Only the robots whose current O_E values are below the threshold send a *reply* message back to the source robot. Robots, responding by sending the *reply* message, consist of a logical robot grouping within the communication robot group. They assist the source robot t perform the specified task. All the robots generating a *reply* message consist of a logical robot group (L_R).

The relation $n(L_R) \le n(C_R)$, between C_R and L_R, is constructed where $n(A)$ represents the number of elements in set A. When not using O_E, as shown in Fig. 2, it is assumed that n variables need to be checked in order to perform a specific task. A *request* message for each single variable i is generated and sent to all robots.

Whenever a source robot issues a *request* message, all the robots receiving the *request* message send a *reply* message back to the source robot. The source robot should compare the status of all robots for the specific variable, such as a remaining power or mobile speed.

Based on n iterations of the above procedure, the source robot can determine robots that comprise a logical robot group destined to perform the specific task required. After deciding the members, the source robot sends a *confirmation* message to the member robots selected in the logical robot group by unicasting.

Using O_E, since all nodes represent status as a single O_E variable, there is no need for the source robot to repeat the iterative procedure n times to form the logical robot grouping. That is, the n variables are represented as a single O_E value in this case. Therefore, one procedure to send a *request* message and receive *reply* messages is enough for the source robot to check the status of all robots to perform the task. Similarly, in the no-entropy case, after deciding the members, the source robot sends a *confirmation* message to the member robots of the logical robot group by unicasting.

In the simulation, it is assumed that only a single variable can be included in a *request* message in order to consider the maximum length of the *request* message. It is assumed that the value of O_E of each robot is randomly assigned and the value is kept as a constant, when each robot performs the logical robot grouping and a group leader selected.

The message will not be lost at the intermediate nodes. In this simulation, the number of messages that is used to construct the communication robot grouping is not considered as the performance metric of the network resource. The number of messages used to establish the logical robot grouping is counted as the performance metric of the network resource and depicted in the following result graphs.

Based on a random network topology scenario, the number of nodes and the network topology are unchanged during the simulation to analyze the message complexity of each variable and compare the simulation results.

For simplicity, five variables, (n equals 5), are selected in this simulation. Fig. 3 shows the results of message complexity of the five different variables respectively.

Monte Carlo simulation performing 400 iterations at each variable is used to evaluate various network topologies for random scenarios and to calculate the average number of messages for each variable, where

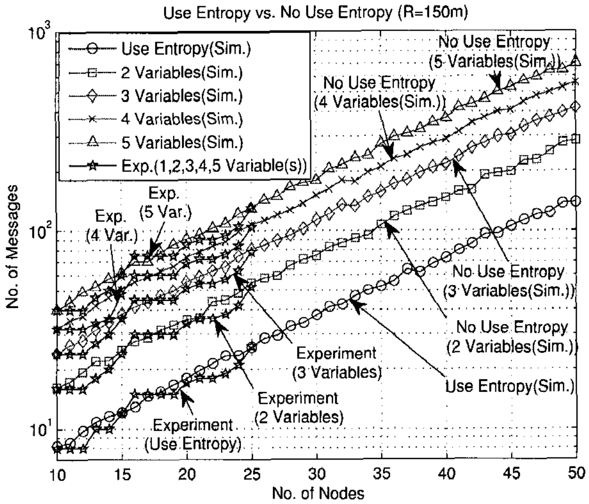


Fig. 3. Message complexity (simulation & experiment).

each topology is composed of a different number of sub networks.

The random robot generator used in this simulation and simulator performance were verified 100, 125, 150, and 175 number of robots per cluster as well as several specs in the adaptive dynamic backbone (*ADB*) algorithm [4] matched with the results in [4], which was performed by *QualNet*, with less than a 1% difference for almost all cases [13].

In Fig. 3, the x axis shows the number of robots in a communication robot group and the y axis shows the number of messages to perform a logical robot grouping. The graph marked as Use Entropy indicates the scenario when all robots are prepared with O_E when a source robot generates a *request* message. The graphs marked as No Use Entropy (n variables) indicates the scenarios when robots do not use the concept of O_E. As it can be expected and shown in Fig. 3, message complexity of the O_E scenario is much less than the message complexities of the other scenarios without entropy.

When the number of variables required increases in the scenarios not using entropy, the number of messages used also increases. O_E makes a logical robot grouping more efficient, since the network resource such as the number of message using the contention-based wireless communication channel can be significantly reduced and the power used to send and receive messages is also saved.

During the simulation, each topology is composed of a different number of communication robot groups. Even though several communication robot groups, based on the transmission range of robot exist in a network topology, only the largest communication robot group is considered in the simulation. Therefore, from the graph of the Use Entropy, it is shown that the number of messages is not so high when the number of the robots is 10. When using O_E, n numbers of variables are represented by a single O_E.

4.2. Experiment

In order to verify the simulation, an experiment using soccer robots was conducted. 10 to 25 soccer robots were randomly assigned in the network of 1.8m x 1.2m with broadcast transmission radius 30 cm.

Several experiments were conducted to match the number of robots against the average number of robots in the simulation located in the network area. Robot movement is captured by CCD camera and its image is processed by METROX vision board. C++ MFC Windows application program measures robot cooperation. Fig. 4 shows the robot images captured by the vision board, where the color is used to identify each robot. Fig. 5 indicates the coordinates of the robots processed by the MFC program in real time.

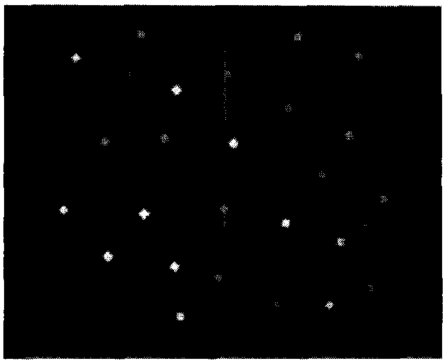
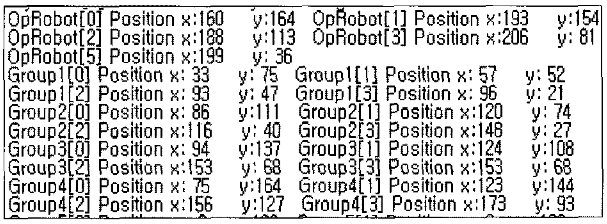


Fig. 4. Robot image captured by METROX vision board.



```
OpRobot[0] Position x:160   y:164  OpRobot[1] Position x:193   y:154
OpRobot[2] Position x:188   y:113  OpRobot[3] Position x:206   y: 81
OpRobot[5] Position x:199   y: 36
Group1[0] Position x: 33   y: 75  Group1[1] Position x: 57   y: 52
Group1[2] Position x: 93   y: 47  Group1[3] Position x: 96   y: 21
Group2[0] Position x: 86   y:111  Group2[1] Position x:120   y: 74
Group2[2] Position x:116   y: 40  Group2[3] Position x:148   y: 27
Group3[0] Position x: 94   y:137  Group3[1] Position x:124   y:108
Group3[2] Position x:153   y: 68  Group3[3] Position x:153   y: 68
Group4[0] Position x: 75   y:164  Group4[1] Position x:123   y:144
Group4[2] Position x:156   y:127  Group4[3] Position x:173   y: 93
```

Fig. 5. Robots' coordinates processed by the MFC program.

The experimental results are plotted in Fig. 3, where the simulation results are also included. As shown in Fig. 3, the average differences between the simulation and experiment are 8.8%, 8.9%, 9.0%, 9.8%, and 8.5% in each case of Use Entropy and No Use Entropy.

4.3. Time-critical cooperative system

When a *request* message is propagated to a robot group, only robots satisfying both O_E and R_E at the same time can transmit a *reply* message to the source robot, since there is no need for each robot to respond with a *reply* message back to a source robot. Due to the time criticality, a *confirmation* message is not issued and the robots move to a source robot, as soon as they send the *reply* message. The simulation emphasizes the network resource needed to perform the above scenario.

Figs. 6 and 7 depict the message complexities, where the O_E values of 0.1 and 0.3 are selected, when R_E is increased from 1 to 9. It can be shown that when the message complexity of R_E is increased to 9 in both Figs. 6 and 7, the message complexities reach an upper bound calculated from the maximum length path obtained from the largest communication robot group. Figs. 8 and 9 show the message complexities, where the R_E values of 1 and 3 are plotted respectively, when O_E is increased from 0.1, 0.3, 0.5, 0.7, 0.9 to 1.

Fig. 10 shows the absolute difference between the simulation and experiment, when the value of O_E is 0.1 with R_E of 0.1. The graph is indicates that there is

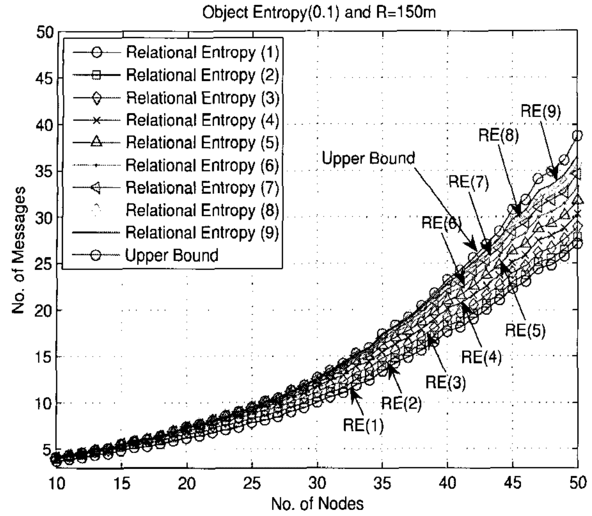


Fig. 6. Message complexity of OE(0.1): simulation.

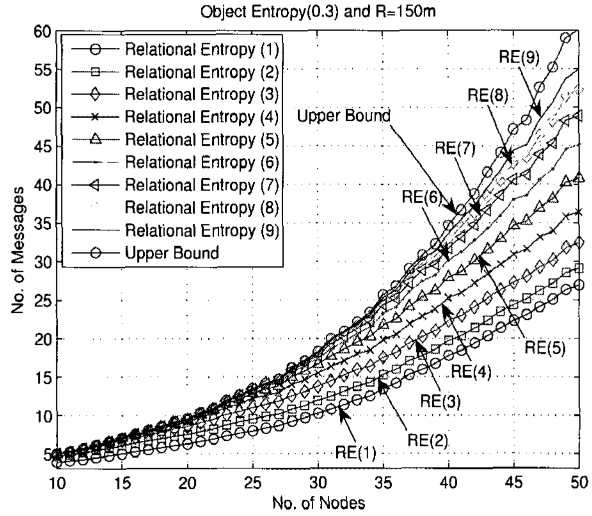


Fig. 7. Message complexity of OE(0.3).

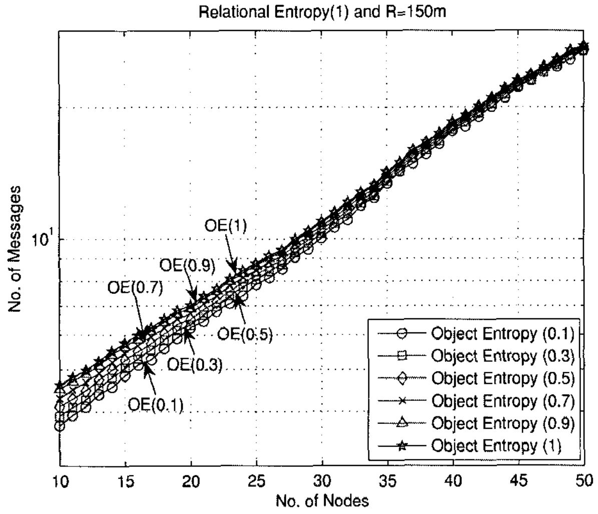


Fig. 8. Message complexity of RE(1).

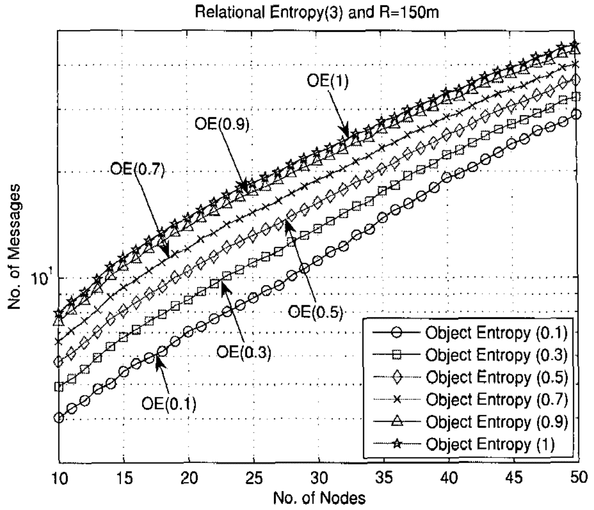


Fig. 9. Message complexity of RE(3).

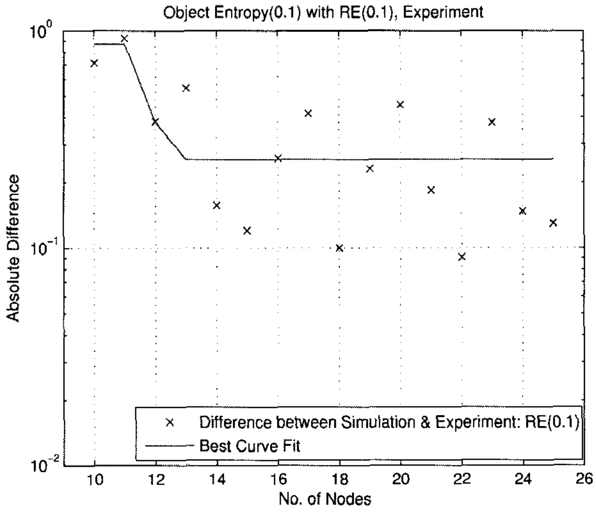


Fig. 10. Comparison of message complexity (OE(0.1) with RE(0.1)) between simulation and experiment.

Table 1. Percentage overhead of $R_E(3)$ [%].

O_E	0.1	0.3	0.5	0.7	0.9	1
Overhead	10.22	30.23	48.98	66.87	84.17	92.51

little difference between the results of the simulation and experiment. The difference lies between 0 and 1, and the experiment follows the trend of the simulation.

Table 1 shows the average values of the percentage overhead of $R_E(3)$ compared to $R_E(1)$, for each value of O_E. Based on Table 1 reveals that for a given value of O_E, $R_E(3)$ uses 10.22%, 30.23%, 48.98%, 66.87%, 84.17%, and 92.51% more network resource than $R_E(1)$. Finding robots that satisfy a requirement (O_E) within 3 hops is much easier than finding robots that satisfy a requirement (O_E) within 1 hop. As all the robots qualifying for the requirement ((O_E) and R_E) of $R_E(3)$ send the *reply* message, message complexity of $R_E(3)$ is higher than for $R_E(1)$.

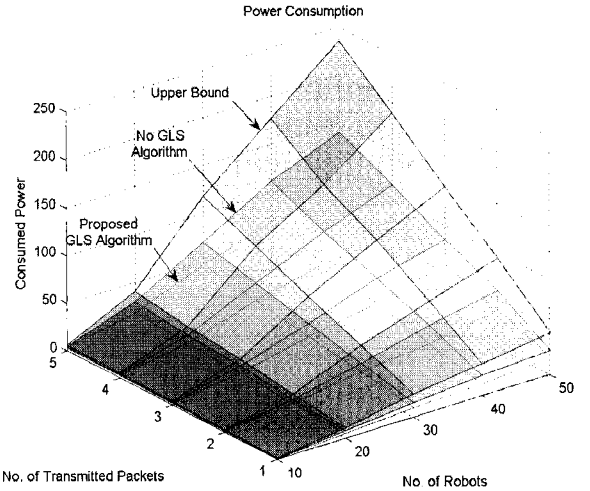


Fig. 11. Power consumption of GLS algorithm.

4.4. Minimized-power consumption system

The algorithm introduced in Definition 1 is implemented to select the group leader minimizing power consumption. A scenario is considered where a single source robot transmits several control messages to control member robots. The control message initiated from a source robot is delivered to a group leader and the message is delivered to the designated member robots. In Fig. 11, the x axis shows the number of robots in a robot group, the y axis shows the number of control messages transmitted from a source robot, and the z axis shows the power consumed in a MANET. The proposed group leader selection algorithm uses the least power compared to power consumption where a group leader is randomly selected (Fig. 11). The largest length of the spanning tree yields the upper bound.

5. CONCLUSIONS

This paper proposes novel algorithms that enable mobile robots to cooperate. The concepts of object entropy, relational entropy, and MANET are introduced to enable a robot grouping to perform a specific task. The entropy value is used to represent the current robot status. The MANET is introduced to implement self-creating, self-administrating, self-organizing, and self-configuring mobile robot group, using a multihop wireless network.

This paper details the process of forming the logical robot group. A route with the lowest value of $\gamma^j = GA^j_{k,l}(t,\Delta_t)$ is selected for the reliability-based cooperative system, when there are several routes between source and destination robots. It proposes relation entropy to consider a time-critical cooperative system. The proposed group leader selection algorithm described in Definition 1 conserves power in mobile robots scattered over a network, since the algorithm reduces control overhead. As demonstrated in the experiment and simulation, the algorithm uses the network resources efficiently. It uses a contention-based wireless communication channel. It can easily support cooperative rescues in an emergency.

REFERENCES

[1] F. R. Noreils, "Toward a robot architecture integrating cooperation between mobile robots: Application to indoor environment," *International Journal of Robotics Research*, vol. 12. pp. 79-98, 1993.

[2] R. R. Murphy, C. L. Kisetti, R. Tardif, L. Irish, and A. Gage, "Emotion-based control of cooperating heterogeneous mobile robots," *IEEE Trans. on Robotics and Automation*, vol. 18, pp. 744-757, 2002.

[3] A. Borkowski, M. Gnatowski, and J. Malec, "Mobile robot cooperation in simple environments," *Proc. of Second Workshop on Robot Motion and Control*, pp. 109-114, 2001.

[4] C-.C. Shen, C. Srisathapornphat, R. L. Z. Huang, C. Jaikaeo, and E. L. Lloyd, "CLTC: A cluster-based topology control framework for ad hoc networks," *IEEE Trans. on Mobile Computing*, vol. 3, no. 1, pp. 18-32, 2004.

[5] X. Hong, K. Xu, and M. Gerla, "Scalable routing protocol for mobile ad hoc networks," *IEEE Network*, vol. 16, no. 4, pp. 11-21, 2002.

[6] P. Basu and J. Redi, "Movement control algorithms for realization of fault-tolerant ad hoc robot networks," *IEEE Network*, vol. 18, no. 4, pp. 36-44, 2004.

[7] A. F. T. Winfield, "Distributed sensing and data collection via broken ad hoc wireless connected networks of mobile robots," In L.E. Parker, G.W. Bekey, and J. Barhen, editors, *Distributed Autonomous Robotic Systems*, vol. 4, pp. 273-282, 2000.

[8] Z. Wang, M. Zhou, and N. Ansari, "Ad-hoc robot wireless communications," *Proc. of IEEE Conference on System, Man and Cybernetics*, vol. 4, pp. 4045-4050, 2003.

[9] S. M. Das, Y. C. Hu, C. S. Lee, and Y. H. Lu, "Efficient unicast messgae for mobile robots," *Proc. of IEEE Conference on Robotics and Automation*, pp. 93-98, 2005.

[10] P, Bracka, S. Midonnet, and G. Roussl, "Trajectory based communication in an ad hoc network of robots," *Proc. of IEEE Conference on Wireless and Mobile Computing, Networking and Communication*, vol. 3, pp. 1-8, 2005.

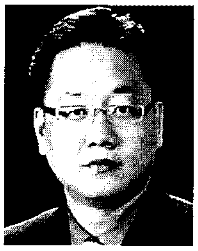
[11] Y. L. Sun, W. Yu, Z. Han, and K. J. Ray Liu, "Information theoretic framework of trust modeling and evaluation for ad hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 305-317, 2006.

[12] B. An and S. Papavassiliou, "An entropy-based

model for supporting and evaluating route stability in mobile ad hoc wireless networks," *IEEE Comm. Letters*, vol. 6, no. 8, pp. 328-330, 2002.

[13] S.-C. Kim, "An algorithm for selecting a group leader in mobile robots realized by mobile ad hoc networks and object entropy," *Proc. of International Conference on Adaptive and Natural Computing Algorithms*, pp. 659-666, 2007.

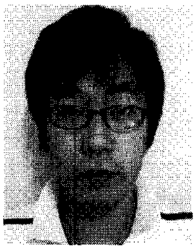
[14] S.-C. Kim and J.-M. Chung, "Message complexity analysis of mobile ad hoc network address autoconfiguration protocols," *IEEE Trans. on Mobile Computing*, vol. 7, no. 3, pp. 358-371, 2008.



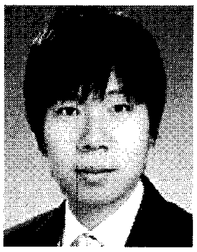
Sang-Chul Kim received the B.S., M.S., and Ph.D. degrees in Electrical Engineering from Kyungpook National University, Computer Science from Changwon National University and Electrical & Computer Engineering from Oklahoma State University in U.S.A. in 1994, 1998, and 2005 respectively. During 1994-1999, he worked in Samsung SDS as a System Engineer. In 2006, he was Post Doctoral Research Fellow in Computer Science of Univ. of Nevada Las Vegas. He is now with the School of Computer Science in Kookmin University Seoul, Korea.

Kee Hyun Shin received the B.S. degree in Mechanical Engineering from Seoul National University in 1981, the M.S. degree in Mechanical Design from Seoul National University in 1983, and the Ph.D. degree in Mechanical Engineering from Oklahoma State University in 1991. During 1990 to 1992, he worked in web handling research center at Oklahoma State University as a Researcher. He is now a Professor in mechanical engineering department, Konkuk University, Seoul, Korea.

Chong-Woo Woo received the Ph.D. degree in Computer Science from Illinois Institute of Technology in U.S.A. He is now with the School of Computer Science, Kookmin University, Seoul, Korea.



Yun Shick Eom received the B.S. degree in Mechanical Engineering from Konkuk University in 2005 and he studies now in graduated school of mechanical design.



Jae Min Lee studies now in undergraduate school mechanical engineering department Konkuk University, Seoul, Korea.