Formation of Mobile Robots with Inaccurate Sensor Information

Gunhee Kim, Doo Yong Lee, and Kyungno Lee

Abstract: This paper develops a control method for some generic formation tasks of multiple mobile robots with inaccurate sensor information. Inaccurate sensor information means that all the robots have only local sensors that cannot accurately measure absolute distances and directions of objects. That is, all the sensors have limitation on the range, and uncertainty in the values. Therefore, more robust and reliable control logic is proposed and implemented. The logic is developed considering generic situations and increasing the number of robots participating in the formation. Petri nets are used for modeling and design of the control logic, which can visualize the control models and make it easy to check the states of each robot. Physically homogeneous mobile robots are designed and built to evaluate the developed logic. Each robot is equipped with eighteen infrared sensors and a UHF transceiver module. The experiment results are analyzed quantitatively by using the data of the relative distances and angles between the robots. And the trajectories of the robots during the formation are also evaluated. The developed control approach is demonstrated with experiments to be successful and efficient for the formation of autonomous mobile robots.

Keywords: multiple mobile robots, robot formation, Petri nets

I. Introduction

In the last decade, many researches on robotic systems have aimed at achieving autonomous multiple robot systems exhibiting cooperative behavior [1]. Such systems offer some important advantages over a single robot. First, inherently complex tasks for a single robot can be accomplished efficiently. Second, building robotic systems is cheaper and more flexible because of using several simple and small-sized robots [2]. Third, losses caused by failure of one or more robots can be minimized [3].

Formation control is one of the most important problems in the research of multiple mobile robots. Solving this problem is quite important for several reasons. First, the solutions to formation are useful primitives for larger tasks such as moving a large object by multiple mobile robots [4]. Second, formation allows a group to maximize the efficiency in understanding environment. Individual team members are permitted to concentrate on only a portion of the environment since their partners cover the rest. For example, a group of robots is often required to move with keeping formation when it explores an unknown territory especially in military applications [5]. It is also useful to detect unusual situation such as trespasses, gas leakage and fire occurrence in surveillance area [6].

For these reasons, there have been much research activities toward solving the formation problems from various approaches.

Parker [7] studies formation to develop control laws governing the behavior of individual agents using a combination of local and global knowledge. She mainly concerns with the proper balance between local and global control, and proposes some guidelines.

Balch and Arkin [5] solve the formation-keeping problems using a behavior-based approach. By using this technique their systems are able to integrate several behaviors simultaneously such as navigating to waypoint, avoiding obstacles, and keeping formation. They also demonstrate the validity of their approach by implementation on two Nomad 150 mobile robots in laboratory, and on four-wheel-drive Unmanned Ground Vehicles outdoors.

Desai et al. [8][9] study control strategies for the formation of mobile robots using nonlinear control theory and graph theory. They concentrate on the navigation of multiple robots in a workspace involving obstacles or narrow passageways. In their research, the multi-robot motion planning is performed by a central planner which uses sensor information collected by each robot.

Yamaguchi [6] presents distributed control method that enables adaptation of formation. By using formation vector, the mobile robot group makes several formations even though some of them are broken. His method is verified by computer simulation.

Suzuki et al. [10][11] investigate a number of significant formation problems of geometric pattern for a distributed system of homogeneous mobile robots. They propose algorithms that make ideal mobile robots form a simple geometric object and distribute themselves uniformly within a particular area. They also study the convergence of mobile robots on a single point within finite steps, and characterize which geometric patterns can be achievable from their initial configuration. However, it is difficult to apply the algorithms to real systems because they assume each robot is a point and the time it takes for a robot to move to its new position is negligible. Yun et al. [12] extend Suzuki and Yamashita's results and consider some realistic physical properties, i.e., robots with physical dimensions have range sensors and satisfy nonholonomic constraints They develop algorithms for various types of line and circle formations.

Although many formation methods are proposed as reviewed above, these techniques still have some problems. In the most of the previous researches, it is assumed that robots and their sensors are ideal. That is, it is supposed that the information used by robots has no uncertainty, and each robot can know the exact positions of all other robots. Therefore, the control of formation is accomplished based on simple computation of exact robot coordinates.

In this paper, only inaccurate sensor information is available for coordinating the formation of multiple mobile robots. In-

Manuscript received: Aug. 29, 2001., Accepted: Nov. 19, 2001.
Gunhee Kim: Advanced Robotics Research Center, KIST (knir38@kist.re.kr)
Doo Yong Lee, Kyungno Lee: Dept. of Mechanical Engineering, KAIST (leedy@kaist.ac.kr/ jeje@discrete.kaist.ac.kr)

accurate sensor information means that all the robots have only local sensors that cannot accurately measure absolute distances and directions of objects. That is, all the sensors have limitation on the range and uncertainty in the values. Therefore, more robust and reliable control logic is proposed and implemented.

This paper develops a control method for generic formation tasks of multiple mobile robots. A lead robot becomes a reference of the formation, and the others determine their position in relation to the lead robot. The control logic is developed considering generic situations and increasing the number of robots participating in the formation.

Physically homogeneous robots equipped with UHF transceiver modules and infrared sensors are designed and implemented to evaluate the developed logic. UHF transceiver modules permit explicit communication to each robot. Communication generally improves performance of the work and extends the domain of tasks that a robot team can perform since it allows robots to exchange their internal states and circumstances with other members. Mataric [13] deals with communication in the cooperative box pushing with two autonomous six-legged robots. She demonstrates that a simple cooperative strategy with communication greatly outperforms both a single-robot alternative and an approach with two non-communicating robots. The robots can also identify other members by using frequency-modulated infrared sensors. This allows a robot to recognize another robot and its behavior.

Petri nets are used for modeling and the design of control logic in this paper. Petri net models can visualize the control logic and make it easy to check the states of each robot.

II. System architecture of mobile robots

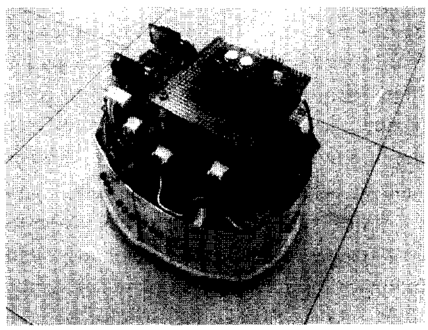
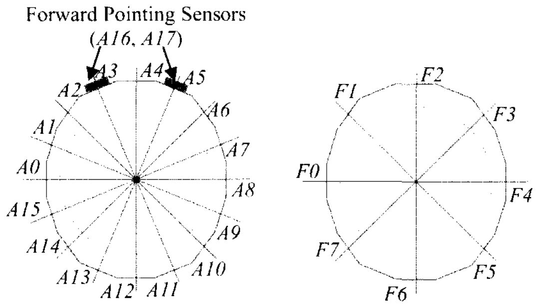


Fig. 1. A mobile robot.

Figure 1 shows a mobile robot built for experiments. It is 16 cm × 13 cm × 16 cm, and its wheelbase is 8 cm. It has a microprocessor, *80C196KC* produced by *Intel Corp.*, for the fundamental processing functions and two stepping motors to perform seven types of motions, *stop, moving forward, moving backward, turning left, turning right, rotating left,* and *rotating right.* The robot can estimate its moved distance and direction from the initial position using encoders. Note that only relative distance and direction from a specific point can be recognized.

The infrared sensor used in this research consists of two infrared-emitting diodes for illumination and one phototransistor for absorption of the reflected light. It is inherently not adequate to measure the precise distance since its value is extremely sensitive to the reflection angle. However, it is possible to estimate the distance roughly by measuring the voltage level at the phototransistor.



(a)Amplitude detectors (b) Frequency detectors

Fig. 2. Arrangement of the infrared sensors.

The arrangement of sensors is shown in Figure 2. There are two forward-pointing sensors and sixteen sensors around the robot. *A* denotes the amplitude detector, and *F* is the frequency detector. To reduce power consumption, the frequency detector circuits are not employed to all sixteen sensors contrary to the amplitude detectors. Though all sixteen sensors emit light with a particular frequency, only eight sensors can detect the frequency of the absorbed beam. Forward pointing sensors, *A16* and *A 17*, are mounted in the same places with *A3* and *A5*, but the altitude is different.

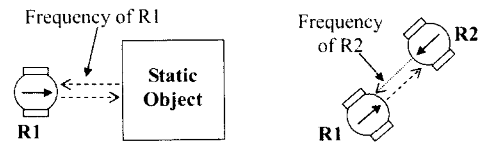


Fig. 3. Object recognition.

If a robot detects an unknown object, it must determine whether the object is another member or a static object such as a wall or an obstacle. It is possible to recognize types of the detected object by using the frequency of the light as robot ID. Each robot has a unique fixed sensor frequency. If a robot receives its own frequency, it recognizes the detected object is static as depicted in Figure 3. If a robot receives a frequency of another robot, it can recognize the ID, position and moving direction of the detected robot. The surfaces of all robots are covered with black tapes to prevent reflection of light. Therefore, a robot cannot receive its own frequency reflected by another robot.

All the mobile robots have UHF transceiver modules to share their information. They are implemented by using *Bim-433-F* fabricated by *Radiometrix Ltd.* This module is capable of half duplex data transmission at maximum speed up to 40 Kbit/s, i.e., it is impossible to receive and transmit data simultaneously. All the robots share a single frequency of 433Hz for data transmission with no centralized hardware robots. It means that only one robot can transmit data at a time and communication process operates autonomously. To eliminate the probability of collision, i.e., simultaneous transmission, the right to use communication channel is assigned to each robot using a round-robin method.

III. Control of generic formation tasks

A general control logic for the formation of two robots is developed, and then, it is extended to three-robot formation, keeping the logic as general as possible. Two formations considered in this paper are described in Figure 4. The first formation is a line formation of two mobile robots. If a lead robot discovers another member, they make a row formation. The second formation is that three robots make one column and one row.

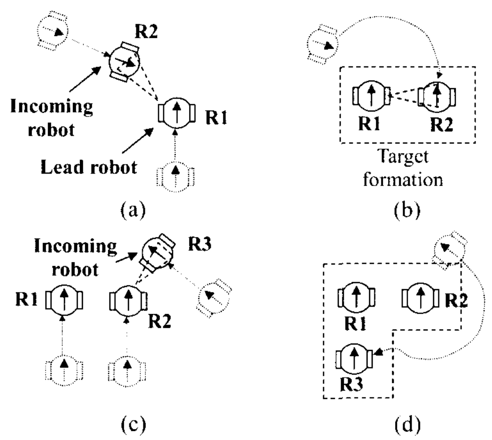


Fig. 4. Two formations.

The proposed formation process consists of three subtasks, i.e., *formation-maintenance*, *identification*, and *formation-generation*. The robots in formation proceed keeping the formation through the subtask *formation-maintenance*. If a new object is detected, it is checked whether is a new robot by the subtask *identification*. The subtask *formation-generation* is executed by the new incoming robot when it detects a formation of robots.

When the robots move together, they maintain the relative distances with each other. The follower robots are responsible for the *formation-maintenance*. The control logic of the subtask *formation-maintenance* is designed by using Petri net models as presented in Figure 5 and Figure 6. The interpretation of places and transitions is given in Table 1 and Table 2. The detailed introduction to Petri nets is provided in many references such as [14] and [15]. Each robot corrects its position errors by comparing its sensor states with those of the leader. The logic deals with two cases, i.e., following the leader in the right side and in the back.

If the robot follows the leader in the right side, it uses five sensors of itself from *A14* to *A2* and five ones of the leader from *A6* to *A10*. In the logic, "a sensor is ON" means that its value exceeds the threshold and becomes larger than the other values. If these ten values are all zero for some fixed period, it can be concluded the formation becomes broken. Then, the robots stop moving and execute *formation-generation* again.

The logic for the follower in the back of the leader is performed in the similar way. Only difference is in the use of six values *A3*, *A4*, and *A5* of itself; and *A11*, *A12*, and *A13* of the leader.

The *identification* process consists of two stages. The first step is the realization by the robots in formation, and second is confirmation by the incoming robot. The robots in formation checks roughly whether a new robot is found by analyzing frequency and amplitude values of the sensors. If a new one is discovered, they stop moving for a fixed period of time to allow the incoming robot to join the formation. The confirmation process starts after the incoming robot receives a message noting the completion of the first step from the leader.

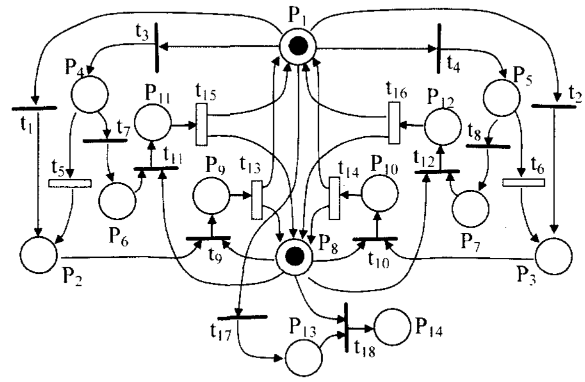


Fig. 5. Petri net model of the subtask *formation-maintenance* for the follower robot in the right side of the lead robot.

Table 1. Interpretation of places and transitions of Figure 5 for the follower robot in the right side of the lead robot.

Places	Interpretation
P_1	Stand-by
P_2 (P_3)	Select the behavior *Turning right* (*Turning left*)
P_4 (P_5)	Watch other sensor states
P_6 (P_7)	Select *Moving forward* with acceleration (with deceleration)
P_8 (P_{14})	Moving forward (Stop)
P_9 (P_{10})	*Turning right* (*Turning left*)
P_{11} (P_{12})	*Moving forward* with acceleration (with deceleration)
P_{13}	Formation becomes broken
Transitions	Interpretation
t_1 (t_2)	*A1* (*A15*) is ON or *A7* (*A9*) of the reference robot is ON
t_3 (t_4)	*A2* (*A14*) is ON
t_5 (t_6)	Time delay for sensor check
t_7 (t_8)	*A10* (*A6*) of the reference robot is ON
t_9 (t_{10})	Command *Turning right* (*Turning left*)
t_{11} (t_{12})	Command *Moving forward* with acceleration (with deceleration)
t_{13} (t_{14})	Time delay for Turning right (*Turning left*)
t_{15} (t_{16})	Time delay for Moving forward with acceleration (with deceleration)
t_{17}	Message from the leader, "Detection is lost"
t_{18}	Emergency stop

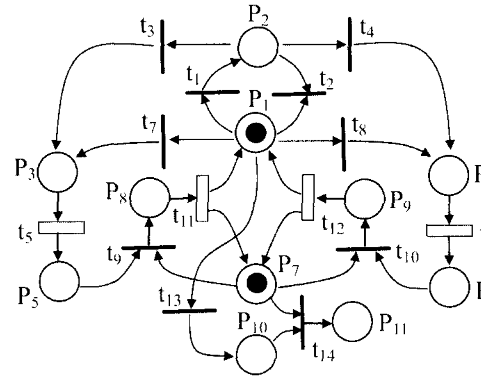


Fig. 6. Petri net model of the subtask *formation-maintenance* for the follower robot in the back of the lead robot.

Table 2. Interpretation of places and transitions of Figure 6 for the follower robot in the back of the lead robot.

Places	Interpretation
P_1	Stand-by
P_2	Watch other sensor states
P_3 (P_4)	Select *Moving forward* for a while, then *Turning right* (*Turning left*)
P_5 (P_6)	Select the behavior *Turning right* (*Turning left*) starts
P_7 (P_{11})	*Moving forward* (*Stop*)
P_8 (P_9)	*Turning right* (*Turning left*)
P_{10}	Formation becomes broken
Transitions	Interpretation
t_1 (t_2)	*A4* is ON (OFF)
t_3 (t_4)	*A11* (*A13*) of the reference robot is ON
t_5 (t_6)	Time delay for *Moving forward* before *Turning right* (*Turning left*)
t_7 (t_8)	*A5* (*A3*) is ON
t_9 (t_{10})	Command *Turning right* (*Turning left*)
t_{11} (t_{12})	Time delay for *Turning right* (*Turning left*)
t_{13}	Message from the leader, "Detection is lost"
t_{14}	Emergency stop

The subtask *formation-generation* is executed by the incoming robot, and the robots in the formation become the reference of the new formation. The incoming robot moves to its target position by exchanging sensor information with the robots already in the formation. The incoming robot determines whether it will turn clockwise (*CW mode*) or counterclockwise (*CCW mode*) around the reference robot to the target position. The direction mode that makes the moving distance shorter is selected.

Figure 7 shows the procedure of the subtask *formation-generation* when the incoming robot determines to turn *CW* (*CW* mode). The basic idea is that the incoming robot moves in the tangential direction of the sensible area of the reference robots. First, the robot rotates left until *A8* or *A7* is ON. Then, it repeats the following three steps until it reaches its target position.

1) If *A8* or *A7* is ON, the robot goes forward for some fixed period *P1*.

2) Then it turns right until *A8* or *A7* is ON. In this process, there is a time limit, *P2*, to escape a deadlock, i.e., keeping turning right without detection.

3) If A6 or A5 is ON, it turns left a little.

In the three-robot formation, if the incoming robot discovers the second robot in the formation at first, it must move to the lead robot keeping the detection. Figure 7.(d) and Figure 7.(e) show the scheme of changing the reference robots when the incoming robot turns *CW*. The incoming robot turns around the reference robot by repeating the three steps. If it detects the frequency of the leader by *F1* or *F2* on its way, it rotate left until *A8* is ON and *F4* detects the frequency of the leader. Then, it executes the three steps again to reach the target position.

When the incoming robot turns *CCW* (*CCW* mode), the subtasks are executed in the similar procedure though the moving directions and sensor numbers are different. The detail is specified in the Petri net model of the control logic of subtask *formation-generation* as shown in Figure 8. Table 3 explains places and transitions of Figure 8.

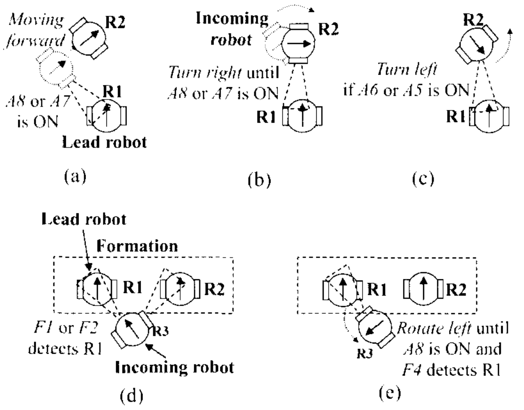


Fig. 7. Control logic of *formation-generation*.

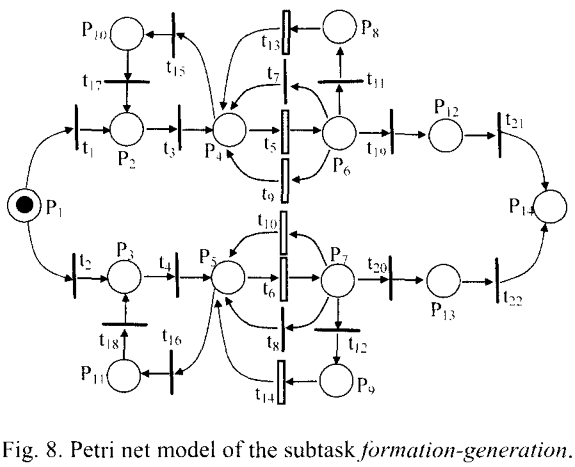


Fig. 8. Petri net model of the subtask *formation-generation*.

Table 3. Interpretation of places and transitions of Petri net model of the subtask *formation-generation*.

Places	Interpretation
P_1 (P_{14})	Initial (Last) state of subtask
P_2 (P_3)	*Rotating right* in *CCW mode* (*Rotating left* in *CW mode*)
P_4 (P_5)	*Moving forward* in *CCW mode* (in *CW mode*)

P_6 (P_7)	*Turning left* in *CCW mode* (*Turning right* in *CW mode*)
P_8 (P_9)	*Turning right* in *CCW mode* (*Turning left* in *CW mode*)
P_{10} (P_{11})	*Rotate right* in *CCW mode* (*Rotate left* in *CW mode*) for changing the reference robot
P_{12} (P_{13})	*Rotating left* (*Rotating right*) for changing moving direction
Transitions	Interpretation
t_1 (t_2)	Determine *CCW mode* (*CW mode*) around the reference robot
t_3, t_7	*A0* or *A15* is ON
t_4, t_8	*A7* or *A8* is ON
t_5, t_6	Time delay for *Moving forward*
t_9, t_{10}	Time limit for escaping the deadlock
t_{11} (t_{12})	*A2* or *A3* (*A5* or *A6*) is ON
t_{13} (t_{14})	Time delay for *Turning right* (*Turning left*)
t_{15} (t_{16})	*F2* or *F3* (*F1* or *F2*) detects the frequency of leader
t_{17} (t_{18})	*A0* (*A8*) is ON and *F0* (*F4*) detects the frequency of leader
t_{19}, t_{20}	Reach target position (It means that *A8* of the leader is ON in two-robot formation and *A12* of leader is ON in three-robot formation.)
t_{21}, t_{22}	Become parallel with lead robot (It means that *A0* is ON in two-robot formation and *A4* is ON in three-robot formation.)

IV. Experiments and results

Two different experiments are performed in order to validate the proposed control strategies in the formation tasks. The maximum detectable range of the infrared sensor is about 50 cm. The normal moving speed of each robot is 12 cm/s, and it accelerates or decelerates between 6 cm/s and 16 cm/s.

1. Two-robot formation

The formation of two robots is performed successfully by the proposed control method. The robots are initially distributed in an open workspace without any guidelines and landmarks. And the initial distance between robots is not too long to lower the possibility of passing each other.

Figure 9 shows the trajectories of two robots during formation. The trajectories are plotted by attaching markers to the bases of the robots and recording their trajectories on the floor covered with 30 × 30 cm tiles. The initial and final configuration of the robots and the points where detection occurs are also shown in this figure. The paths of each of them are not smooth since the formation progresses by repeating the proposed three steps and the sensors are arranged discretely.

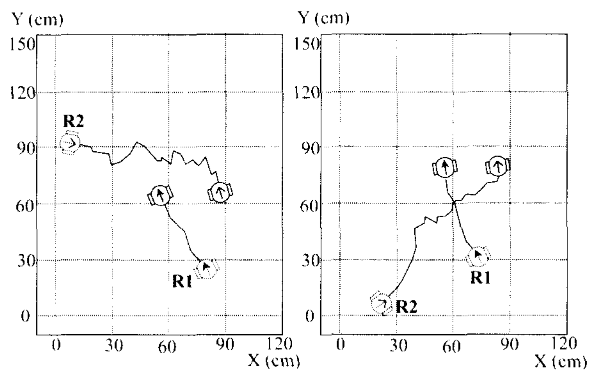


Fig. 9. Paths of two robots during the formation.

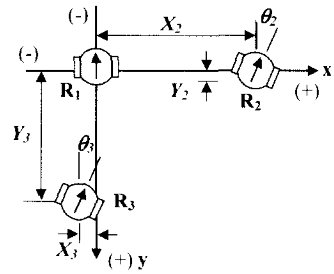


Fig. 10. Measurement of the results of the formation.

Table 4. The results of formation experiments of two robots.

Experiments number	Robot 2		
	X_2	Y_2	θ_2
1	33	-2	5
2	36	8	-8
3	29	-1	0
4	27	5	13
5	33	-3	15
6	36	7	21
7	27	3	15
8	33	-1	5
9	31	0	-3
10	25	1	20

(Xn, Yn : cm, θ_n : degrees)

The formation is analyzed quantitatively by the relative distances and angles between the lead robot and the follower robots. They are measured from the center of each robot as shown in Figure 10. The results of experiments are shown in Table 4. The initial positions of the robots are varied in each experiment. In an ideal formation, the robots will form a line without y-deviation and they become parallel with each other. However, the results listed in Table 4 have some error bounds. Each robot maintains the relative distances by comparing measured sensor values with the threshold. However, the distances cannot be maintained exactly in all experiments since the uncertainty exists in the sensor values. The sensors are arranged with 22.5 degrees apart around the robot, and hence the deviation of angels between robots can occur within this limit.

2. Three-robot formation

Experiments for the formation of three robots are conducted in the same conditions as in the two-robot formation. The formation is also performed successfully by the proposed control method.

The trajectories of three robots during formation are shown in Figure 11. The results of the three-robot formation are also quantified by the relative distances and angles. Initially, the distance between R1 and R2 is 30 cm, and the directions of the robots are parallel with each other. They move along keeping the formation until a new robot is detected. The results of three-robot formation are shown in Table 5.

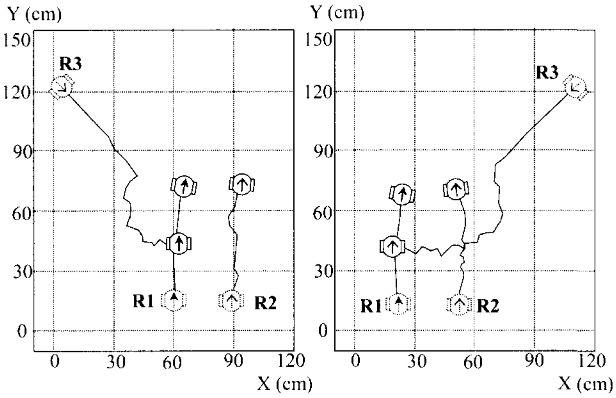


Fig. 11. Paths of three robots during the formation.

Table 5. The results of formation experiments of three robots.

Experiments number	Robot 2			Robot 3		
	X_2	Y_2	θ_2	X_3	Y_3	θ_3
1	31	1	-18	-2	27	-8
2	27	-2	11	-3	28	-18
3	29	4	10	4	30	-23
4	26	3	-26	-1	28	0
5	33	2	-6	3	30	2
6	31	-1	-9	-1	32	8
7	32	-2	5	6	35	-15
8	28	-2	7	3	31	9
9	27	3	-13	-4	29	-13
10	30	1	8	-2	34	5

(Xn, Yn : cm, θ_n : degrees)

V. Conclusion

This paper develops a control method for some generic formation tasks with inaccurate sensor information. All the sensors have limitation on range and uncertainty in their values. Since the formation control in the most previous researches is accomplished based on computation of exact robot coordinates, they cannot be applied to highly uncertain environment and the robotic systems with inaccurate sensor information. A few generic formation tasks are considered in this paper to generalize the control logic for varying number of the participating robots. The control logic is developed by using Petri nets models.

The experimental results are quantitatively analyzed using relative distances and angles between the lead robot and the follower robots. The developed control approach is demonstrated with experiments to be successful and efficient for the formation of autonomous mobile robots.

References

[1] M. J. Mataric, "Issues and approaches in the design of collective autonomous agents," *Robotics and Autonomous Systems*, vol. 16, no. 2, pp. 321-331, Dec., 1995.

[2] Y. U. Cao, A. S. Fukunaga, A. B. Kahng, and F. Meng, "Cooperative mobile robotics : antecedents and directions," *Proc. of IEEE International Conf. on Intelligent Robots and Systems*, pp. 226-234, 1995.

[3] J. S. Bay, "Design of the "army-ant" cooperative lifting robot," *IEEE Robotics and Automation Magazine*, pp. 36-43, Mar., 1995.

[4] D. J. Stilwell and J. S. Bay, "Toward the development of a material transport system using swarms of ant-like robots," *Proc. of IEEE International Conf. on Robotics and Automation*, pp. 766-771, 1993.

[5] T. Balch and R. C. Arkin, "Behavior-based formation control for multirobot teams," *IEEE Trans. on Robotics and Automation*, vol. 14, no. 6, pp. 926-939, Dec., 1998.

[6] H. Yamaguchi, "Adaptive formation control for distributed autonomous mobile robot groups," *Proc. of IEEE International Conf. on Robotics and Automation*, pp. 2300-2305, 1997.

[7] L. E. Parker, "Designing control laws for cooperative agent teams," *Proc. of IEEE International Conf. on Robotics and Automation*, pp. 582-587, 1993.

[8] J. P. Desai, V. Kumar, and J. P. Ostrowski, "Controlling formations of multiple mobile robots," *Proc. of the IEEE International Conf. on Robotics and Automation*, pp. 2864-2869, 1998.

[9] J. P. Desai, V. Kumar, and J. P. Ostrowski, "Control of changes in formation for a team of mobile robots," *Proc. of IEEE International Conf. on Robotics and Automation*, pp. 1556-1561, 1999.

[10] I. Suzuki and M. Yamashita, "Distributed anonymous mobile robots : formation of geometric patterns," *SIAM Journal on Computing*, vol. 28, no. 4, pp. 1347-1363, Mar., 1999.

[11] K. Sugihara and I. Suzuki, "Distributed algorithms for formation of geometric patterns with many mobile robots," *Journal of Robotic Systems*, vol. 13, no. 3, pp. 127-139, Mar., 1996.

[12] X. Yun, G. Alptekin, and O. Albayrak, "Line and circle formation of distributed physical mobile robots," *Journal of Robotic Systems*, vol. 14, no. 2, pp. 63-76, Feb., 1997.

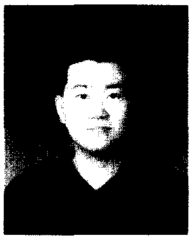
[13] M. J. Mataric, M. Nilsson, and K. T. Simsarian, "Cooperative multi-robot box-pushing," *Proc. of IEEE International Conf. on Robotics and Automation*, pp. 556-561, 1995.

[14] T. Murata, "Petri nets : properties, analysis, and application," *Proc. of IEEE*, vol. 77, no.4, pp. 542-580, Apr., 1989.

[15] R. Zurawski and M. Zhou, "Petri nets and industrial applications : A Tutorial," *IEEE Trans. on Industrial Electronics*, vol. 41, no.6, pp.567-583, Dec., 1994.

Gunhee Kim
He received the B.S. and M.S. degrees in Mechanical Engineering from the Korea Advanced Institute of Science and Technology, in 1999 and 2001, respectively. He is currently a research scientist in Advanced Robotics Research Center, Korea Institute of Science and Technology. His research interests include cooperative control of multiple robot systems, and mobile robot control.

Kyungno Lee
He received the B.S. degree in Mechanical Engineering from Yonsei University in 1996, and the M.S. degree in Mechanical Engineering from KAIST in 1998. Currently, he is a Ph.D. student in the Department of Mechanical Engineering, KAIST, Korea. His research interests are robotics, virtual reality, and discrete event system control.



Doo Yong Lee
He earned the B.S. degree from the Department of Control and Instrumentation Engineering, Seoul National University, Seoul, Korea, and the M.S. and Ph.D. degrees from the Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute, Troy, New York, U.S.A. He is an Associate Professor of the Department of Mechanical Engineering of Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea. He was a postdoctoral Research Associate at Information Technology Services, Rensselaer Polytechnic Institute from 1993 to 1994. He was a Visiting Scholar at Osaka University, Osaka, Japan in 1997.

His research interests include Discrete Event Systems, robotics, virtual reality, and manufacturing automation.

He received The Charles M. Close Doctoral Prize from Rensselaer Polytechnic Institute in 1993, and the Baek-Am Paper Award from the Korean Society of Mechanical Engineers in 1999. He is a Senior Member of IEEE, a Senior Member of the Society of Manufacturing Engineers (SME), a Member of the Korean Society of Mechanical Engineers, and a Member of the Institute of Control, Automation and Systems Engineers, Korea.

He serves as an Associate Editor of the IEEE Transactions on Systems, Man, and Cybernetics, Part B; the KSME International Journal; and the Journal of Control, Automation and Systems Engineering.