Network Based Multiple Mobile Robots with Subsumption Architecture Supporting Swarm Behaviors

Maki K Habib[1], Fusaomi Nagata[2], Akimasa Otsuka[2] and Keigo Watanabe[3]

[1]The American University in Cairo, Egypt
[2]Tokyo University of Science, Yamaguchi, Japan
[3]Okayama University, Yamaguchi, Japan

[1]maki@ieee.org

Abstract: This paper presents the use of dynamic sampling period to evaluate the basic behavior performance of network-based multiple mobile robots system with cooperative swarm behaviors. Network-based subsumption architecture with PC server is proposed to minimize the total cost for designing multiple mobile robots system by maximizing the group performance of robots with limited hardware and software capabilities rather than optimizing the behavior performance of a individual robot. This enables to develop high-level group behavior architecture such as a complex schooling behavior. Such capability is based on decomposing a complex behavior into simple and basic behaviors that are organized into layers of subsumption architecture. Finally, this paper, the basic performance of the network-based subsumption architecture is experimentally evaluated in association with the measurement of the dynamic sampling period.

Keywords: Multiple mobile robots, Subsumption architecture, Swarm intelligence, Group behaviors.

1 INTRODUCTION

Nature continually aspire the development of new ideas and system. Many simple biological species influence directly the development of new multi robot systems. Some of these species include insects such as ants [1 - 4], bees, wasps, and termites [5]. A multi robot system is a widely progressive research fields that went from simulation, prototyping and recently real applications [`1, 6]. It represents a team of robots that are either heterogeneous or homogeneous in their physical structure and functionality. The importance of multi robot systems is reflected in its high task reliability, fault tolerance and flexibility, spatially distributed, cost effective by using simpler individual structures, high application demand and the research challenges facing it [6, 22].

While robotics community witness recognizable progress in research and developments in the field of multi robot systems and its applications, there are still many research challenges which need to be addressed and efficiently solved by applying new innovative techniques. These research challenges may include: task allocation, cooperative mapping and localizations, sensor integration and real time sensory information fusion, real time based control architectures, local and global behaviors, resources utilization and deadlock solving, heterogeneous and distributed intelligence (think, reason, decide and learn), effective reconfigurable modular functionality, autonomy and cooperation, awareness and coordination, inter process communication and human machine interface, etc. [1aa, 1a, 1b].

The application areas of multi robot systems may cover areas such as, military and security, underwater and space exploration, hazardous environments, service robotics in both public and private domains, entertainment, and so forth, can benefit from the use of multi-robot systems. In these challenging application domains, multi-robot systems can often deal with tasks that are difficult, if not impossible, to be accomplished by an individual robot. The following paragraph highlights some examples of the work in the field of multi robot systems.

Parsons and Canny proposed an algorithm for planning the motions of several mobile robots which share the same workspace containing polygonal obstacles [7]. Each robot has an ability of independent translational motion in two dimensions. The algorithm computes a path for each robot which avoids all obstacles in the workspace as well as the other robots. Barman et al. developed an extensible facility for multiple mobile robots [8]. The system consists of nine radio-controlled mobile robots, two CCD color video cameras, a video transmitter and tuner, radio controllers, and so on. Software for tracking control is described. Kube and Zhang examined the problem of controlling multiple behavior-based autonomous robots [9], [10]. Based on observations made from the study of social insects, they proposed five simple mechanisms used to invoke group behavior in simple sensor based mobile robots. They also constructed a system of five homogeneous sensor-based mobile robots with capability of achieving simple collective task. Noreils described architecture for cooperative and autonomous mobile robots [11]. The cooperation is composed of two phases. One is the collaboration where a task is decomposed into subtasks. The other is the coordination where robots coordinate their activities to fulfill the initial task using the notion of coordinated protocols. This architecture showed benefits of modularity, robustness and programmability.

In addition, Azarm and Schmidt presented a novel approach to do conflict-resolution for multiple mobile robots [12]. A framework for negotiation is developed by using the online motion planning, which permits quick decentralized and parallel decision-making. The key objective of the negotiation procedure is dynamic assignment of robot motion priorities. The performance was evaluated experimentally using only two mobile robots. Bennewitz and Burgard considered the problem of path planning for teams of mobile robots [13] using a decoupled and prioritized approach to coordinate the movements of the mobile robots in their environment. The proposed algorithm computes the paths for the individual robots in the configuration-time space. To estimate the risk of colliding with other robots, it uses a probabilistic model of the robots motions. Guo and Parker proposed a distributed and optimal motion planning algorithm for multiple robots, in which computation cost was decomposed into two modules, i.e., path planning and velocity planning [14]. The D* search method was applied in both modules, based on either geometric formulation or schedule formulation. The algorithm was implemented and tested in a group of Nomad 200 indoor robots. Parker outlined the project that demonstrated a team of 100+ heterogeneous robots solving an indoor reconnaissance and surveillance task [15]. The focus was the impact of heterogeneity on the collaborative solution approach that the robot team must take. Pimentel and Campos addressed the problem of multi-mobile robot cooperation with strict communication constraints which are considered indispensable for successful task execution [16]. The problem is modeled as a minimization of an energy functional which accounts for network connectivity, other relevant robot and task requirements in order to select locally optimal actions for each robot.

Antonelli et al. presented two experimental case studies performed using a multi-robot system made of six Khepera II mobile robots [17]. The experiments are aimed at testing the performances and the robustness of a behavior-based technique, called the null-space-based behavioral control (NSB). The NSB approach was developed to control a generic team of autonomous vehicles and it was implemented on a centralized architecture to control a platoon of autonomous mobile robots at a kinematic level. Also, the experimental validation of the NSB in the presence of static and dynamic obstacles was evaluated with a team of grounded mobile robots [18].

In this paper, network-based subsumption architecture with PC server is proposed to minimize the total cost for designing multiple mobile robots system by maximizing the group performance of robots with limited hardware and software capabilities rather than optimizing the behavior performance of individual robot. This enables to develop high-level group behavior. A server supervisory control with networked-based subsumption architecture is implemented and tested to realize a schooling behavior by relying only on information from the PSD sensors. Further, the dynamic sampling period is introduced to evaluate the basic performance of developed system.

2 MOBILE ROBOT WITH SIX PSD SENSORS

A. Basic Hardware

Figure 1 shows the developed mobile robot [19], [20]. The robot is an omnidirectional mobile robot with three wheels driven by DC motors and six PSD (Position Sensitive Detector) sensors.. It is produced by TosaDenshi LTD. A MicroConverter ADuC814ARU provided by Analog Devices is mounted on the control board of the mobile robot. In order to measure the distance to any object in real time, $\boldsymbol{d}_i(k) = [d_{i1}(k)...d_{i6}(k)]^T$ is the distance vector of the i-th mobile robot at the discrete time k. The PSD sensor is mainly composed of LEDs, electrical resistances and photodiodes, and can calculate the distance to an object through triangulation technique. Figure 2 shows the measurement graph of the PSD sensor, In order to cope with the problem of narrow directivity, the required number of the PSD sensor was decided to be six. In order to reduce the dead zone angle in the sensor view, more PSD sensors is required. Besides, each robot has a Bluetooth wireless device to communicate with the PC server.

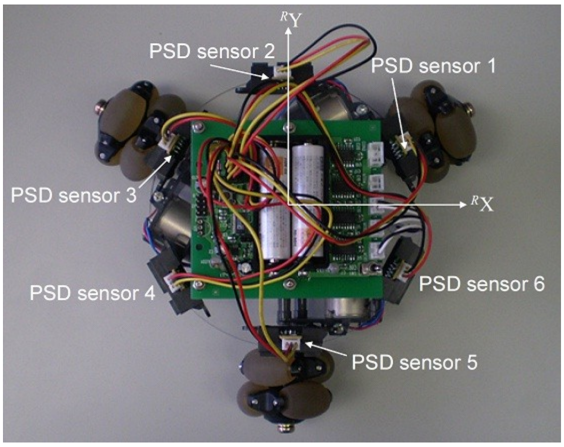

Fig. 1. The mobile robot with three wheels and six PSD sensors.

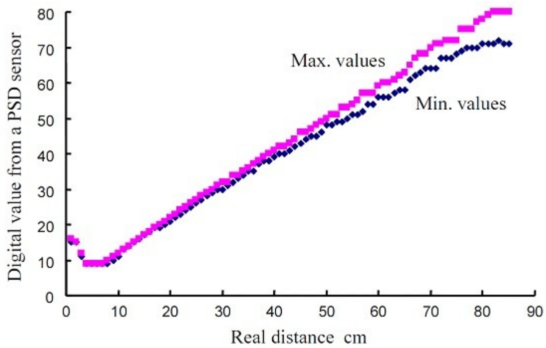

Fig.2. Shows the functional graph of PSD distance measurements.

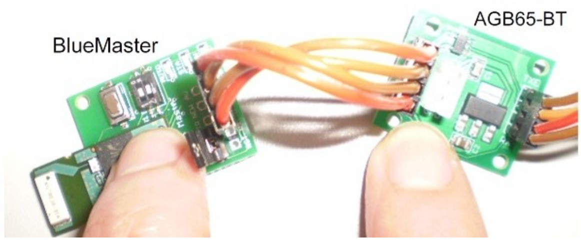

Fig.3. Bluetooth module through a serial interface AGB65-BT.

The Bluetooth is connected to each mobile robot through serial port to support data communication, so that text codes can be transmitted to and received from the Bluetooth module called BlueMaster through a serial interface AGB65-BT as shown in Fig. 3. The small Bluetooth device is provided by Asakusagiken Co., Ltd. The AGB65-BT is connected to the serial port of MicroConverter ADuC814ARU mounted on a mobile robot.

B. Kinematic Control of Three-Wheeled Mobile Robot

Next, the kinematic control method of the mobile robot is explained. Figure 4 illustrates the kinematic model of the mobile robot in robot coordinate system $\Sigma R(O - {}^{R}X^{R}Y)$. ω_i $(i = 1, 2, 3)$ is the angular velocity of each wheel. Also, by using the radius r of each wheel, $v_{ri} = r\omega_{ri}$ is the forward velocity of each wheel. If the position and orientation vector of the robot is given by $[x_r \quad y_r \quad \phi_r]^T$, then the velocity is represented by $v_r = [\dot{x}_r \quad \dot{y}_r \quad \dot{\phi}_r]^T$. The following kinematic relation is obtained from Fig. 3 [21].

$$\begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{pmatrix} = \frac{1}{r} \begin{pmatrix} -\frac{1}{2} & \frac{\sqrt{3}}{2} & L \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} & L \\ 1 & 0 & L \end{pmatrix} \begin{pmatrix} \dot{x}_r \\ \dot{y}_r \\ \dot{\phi}_r \end{pmatrix} \qquad (1)$$

where L is the distance between the center O of the robot and the center of each wheel. By using Eq. (1), the robot can be controlled kinematically, i.e., the is desired behavior designed by $v_r = [\dot{x}_r \quad \dot{y}_r \quad \dot{\phi}_r]^T$ can be performed by making three wheels rotate with the angular velocity vector $\boldsymbol{\omega} = [\omega_1 \; \omega_2 \; \omega_3]^T$. As special cases, Table 1 shows the basic velocity components to move in the direction of each PSD sensor. When designing the schooling mode using multiple mobile robots, the six basic velocities are used. The important point is that the direction of velocity, a mobile robot generates in ΣR, depends on the ratio $\dot{x}_{ri} : \dot{y}_{ri}$. The velocity norm can be arbitrarily changed as $\alpha\omega = [\alpha\omega_1 \; \alpha\omega_2 \; \alpha\omega_3]^T$ with a scalar α.

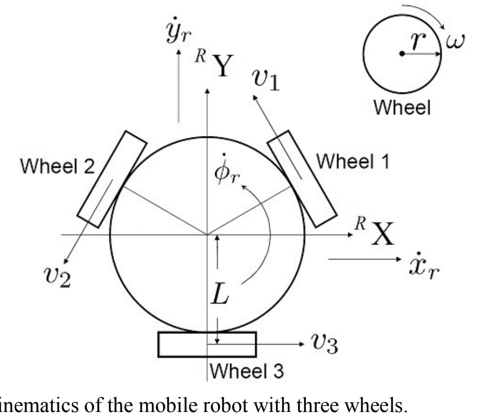

Fig. 4. Kinematics of the mobile robot with three wheels.

Table 1
Velocity components to move in the direction of each PSD sensor.

i	1	2	3	4	5	6
$\dot{x}_{ri}$	$\sqrt{3}$	0	$-\sqrt{3}$	$-\sqrt{3}$	0	$\sqrt{3}$
$\dot{y}_{ri}$	1	2	1	-1	-2	-1

3 NETWORK-BASED MULTI ROBOT SYSTEM

A. Software Development Environment and Its Limitation

The software development environment of the selected mobile robot using a free C language has two limitations since we are using robots with limited capabilities. The first is having limited memory and communication overhead, the flash ROM of the mobile robot is only 8 kB, so that it is difficult to deal with huge amount of state information. The second is, the mathematical functions within the standard library such as "exp ( )" is not supported by the complier while the computation resources is not enough to realize advance behavior. Thus, for example, it is impossible to directly program potential field technique supporting navigation and planning. In order to cope with such limitations of the software development environment, a n efficient strategy was adopted to have two levels of behaviors. The first level of behaviors constitutes simple reflexive swarm behaviors implemented directly on the mobile robot itself. The other level of behaviors represents a complex set of smarm behaviors implemented on a PC server expressing a supervisory control scheme. In the server control mode, each mobile robot collects all sensory information from the six PSD sensors and transmits it to the PC server, $\boldsymbol{d_i}(k) = [d_{i1}(k)...d_{i6}(k)]^T$ through Bluetooth communication. The subscript i denotes the identification number of a robot. Then the server decides and returns a simple swarm command/behavior to the robot while considering the overall swarm behavior (current and next). By means of the proposed supervisory control of the PC server, a complex swarm behavior can be executed be deciding the selection of such behavior by the PC server according to the situation needs and decompose it into sets of simple behaviors that are sent to each robot to execute. In addition, functions demanding high computation, such as potential field technique, can be placed at the PC server side where the Windows Visual Studio runs (representing the software development environment).

B. Server Supervisory Control Based on Subsumption Architecture

The proposed PC server based supervisory control is designed coordinate the required behavior for multiple homogeneous mobile robots each with three wheels and six PSD sensors. This system is used to support study needs of fourth year students at Tokyo University of Science, Yamaguchi, such as, to learn the subsumption control architecture for schooling behavior. The subsumption control architecture was first proposed by Brooks [23]. Students can practically know the basic concept and effectiveness of subsumption control architecture which provides a method for structuring reactive systems from lower level to higher level using layered sets of rules, i.e., reactive behaviors according to the change in the situation within robot's environment. Accordingly, the PC server returns a set of simple behavior associated with short execution time, e.g., 200 ms to the corresponding robot. Nine kinds of the most simply subdivided reaction behaviors, i.e., reflex actions, are implemented for the mobile robots as tabulated in Table 2. When a set of behavior codes and an execution time is transmitted from the PC server to a mobile robot, the mobile robot conducts the motion exactly within the specified execution time.

Three agents called "Avoid objects ", "Turn to left or right" and "Move forward " are designed as a composite of a set of basic behaviors/commands/actions A_i (i = 0, 1, 2, ... , 8) shown in Table 2 that are used to realize a schooling behavior.

4 SUBSUMPTION ARCHITECTURE IMPLEMENTED ON PC SERVER

The software development environment is Windows Visual Studio C#, which is used to develop and implement high level software architecture such as subsumption control architecture according to application requirements and can be adapted based on robot's hardware capabilities. Subsumption is an efficient way to decompose complex behavior into a set of simple behaviors. Based on sensory information, only one behavior is selected as a highest priority when a new set of sensory information is presented. Figure 5 shows the subsumption control architecture implemented on the supervisory PC server. The controller includes the three agents. The upper level agent has a higher priority to be dispatched. This section introduces the three kinds of agents and the corresponding output command codes from the list shown in Table 2. The nine commands shown in Table II are simple and basic motions but important reflex actions for each mobile robot to consequently produce the competences relevant to the three agents. The PC server receives PSD sensory information $\boldsymbol{d_i}(k)(1 \le i \le N)$ from all mobile robots periodically every dynamic sampling period, in which N is the number of the available mobile robots. By analyzing $\boldsymbol{d_i}(k)$, the controller dispatches the current execution to one of the three agents for the i-th mobile robot according to triggering priority.

In the schooling mode, all mobile robots try regularly move along the inner of a circular fence keeping the distance to both the fence and other mobile robots. This mode enables the robots to behave like carps in a Japanese artificial circular pond.

TABLE 2
The basic list of simple reactive behaviors/commands/Actions that can be executed at the robot side together with the corresponding motion.

Cmd. code	Corresponding motion
0	Halt at the position
1	Move to the direction of PSD sensor 1
2	Move to the direction of PSD sensor 2
3	Move to the direction of PSD sensor 3
4	Move to the direction of PSD sensor 4
5	Move to the direction of PSD sensor 5
6	Move to the direction of PSD sensor 6
7	Rotate to clockwise direction
8	Rotate to counterclockwise direction

Figure 6 and 7 show the layout configuration and the experimental scene of schooling behavior respectively, in which multiple mobile robots are controlled based on the subsumption control architecture incorporated in the PC server associated with supervisory control capability as shown in Fig. 5. It was confirmed from the experiment that the multiple mobile robots performed the desirable schooling behavior successfully.

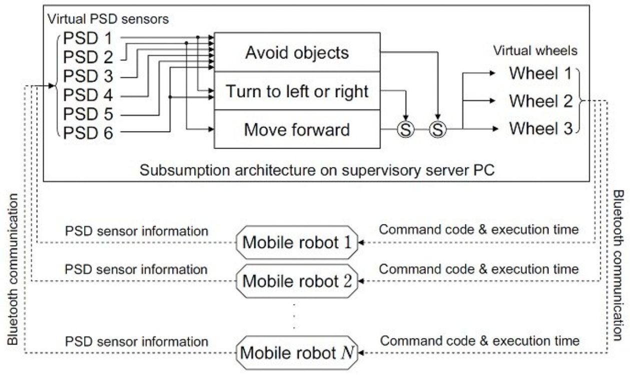

Fig. 5. Network-based subsumption architecture for a schooling behavior, which is implemented on a PC server.

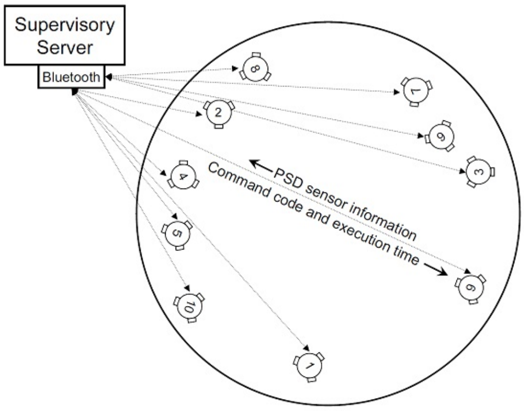

Fig. 6. Layout configuration of the developed system.

Fig. 7. Experimental school behavior demonstration.

To coordinate the relation between each robot and the PC server, there is an agent dispatcher supervises the controller shown in Fig. 4. The dispatcher does not immediately move the execution right to a higher priority agent when a lower priority agent is running, because each agent works as a simple reactive behavior every sampling period according to the sensory information. Instead of this, whenever a reaction behavior is executed during a specified execution time, the dispatcher checks and updates the activity of each agent and gives the next execution right to a newly updated active agent with the highest priority.

5 DYNAMIC SAMPLING PERIOD

The dynamic sampling period is an important factor to evaluate the performance of the multiple mobile robots system. The dynamic sampling period is defined for a PC server as a variable time starting from submitting a command packet to a mobile robot until receiving a response packet including the new sensory information. Figure 9 shows the handshake process to measure the sampling period $T_k = t_{k+1} - t_k$. The processor time t_k at the discrete time k in the PC server is used as a timestamp to synchronize the time between the PC server and each mobile robot. The processor time t_k can be obtained by referring the Windows parameter "int smillisec = Environment.TickCount;" in a timer interrupt process at the PC server side. The size of the packet used in Bluetooth communication between the PC server and a mobile robot is 30 bytes. When a command packet is transmitted from the PC server to a mobile robot, necessary information about the command is written at the offset positions from byte 0 to 11. On the other hand, when returning to the PC server from the mobile robot, the sensory information of the six PSD sensor are set at the offset positions starting from byte 12 to 29. It should be noted that the execution time in Fig. 4 is set to 0 ms in this measurement process. It is observed experimentally that there exists some time dispersion around 62.4 ms and the resolution of the time dispersion is about 15.6 ms. The timer interval of Microsoft Visual C# used in the measurement.

6 CONCLUSION

In this paper, network-based subsumption architecture has been presented to realize high level behavior such as, schooling behavior using only information from PSD sensors. Experimental results showed interesting behavior among the multiple mobile robots, such as following, avoidance, and schooling. Further, how to simply measure the dynamic sampling period has been introduced to evaluate the basic performance of network-based multiple mobile robots system. The dynamic sampling period is defined for the PC server as the variable time. It is expected that the dynamic sampling period becomes one of important criterion for network-based multiple mobile robots system.

REFERENCES

[1] Asama, H, Habib, M K, Endo, I, Ozaki, K, Mastumoto, A Ishida, Y, "Functional distribution among multiple mobile robots in an autonomous and decentralized robot system", IEEE Robotics and Automation 1991 conf. (ICRA 1991), USA, pp.1921-1926.

[2] Moffet, M W, "Cooperative Food Transport by an Asiatic Ant," National Geographic Research, 4(3), 1988, 386-394.

[3] Anderson, C and Franks, N R., "Teams in animal societies," Behavioral Ecology, 12(5), 2001, 534-540.

[4] Gautrais, J., Theraulaz, G., Deneubourg, J.L., and Anderson, C., "Emergent polyethism as a consequence of increased colony size in insect societies," Journal of Theoretical Biology, 215, 2002, 363-373.

[5] Mori, H, Saito, Y, and Tho, Y P, "Cooperative group predation in a sit and wait cheyletid mite," Experimental and Applied Acarology, 23, 1999, 643-651.

[6] Chow, M-Y, Chiaverini, S, and Kitts, C, "Introduction to the Focused Section on Mechatronics in Multirobot Systems," IEEE/ASME Transactions on Mechatronics, VOL. 14, No. 2, 2009, pp.133 – 140.

[7] Parsons, D and Canny, J, "A motion planner for multiple mobile robots," in Proc. IEEE International Conference on Robotics and Automation, Cincinnati, OH, USA, May, 1990, 8–13.

[8] A. R. Barman, A R, Kingdon, S J, Little, J J, et al, "DYNAMO: Real-time experiments with multiple mobile robots," Intelligent Vehicles Symposium, Tokyo, Japan, July, 1993, 261–266.

[9] Kube, C R and Zhang, H, "Collective robotic intelligence," in Proc. The Second International Workshop on Simulation of Adaptive Behavior, 1992, 460–468.

[10] Kube, C R and Zhang, H, "Collective robotics: From social insects to robots," Adaptive Behavior, vol. 2, no. 2, 1993, 189–219.

[11] Noreils, F R, "An architecture for cooperative and autonomous mobile robots," in Proc. IEEE International Conference on Robotics and Automation, Nice , France, May, 1992, 2703–2710.

[12] Azarm, K and Schmidt, G, "Conflict-free motion of multiple mobile robots based on decentralized motion planning and negotiation," in Proc. IEEE International Conference on Robotics and Automation, Albuquerque, NM , USA, April, 1997, 3526–3533.

[13] Bennewitz, M and Burgard, W, "A probabilistic method for planning collision-free trajectories of multiple mobile robots", Proc. of the Workshop Service Robotics - Applications and Safety Issues in an Emerging Market at the 14th European Conference on Artificial Intelligence, Berlin, Germany, August, 7 pages, 2000.

[14] Guo, Y and Parker, L E, "A distributed and optimal motion planning approach for multiple mobile robots", Proc. IEEE International Conference on Robotics and Automation, Washington, DC, USA, 2002, 2612–2619.

[15] Parker, L E, "The effect of heterogeneity in teams of 100+ mobile robots," Multi-Robot Systems: From Swarms to Intelligent Automata, Volume II, Kluwer Academic Publishers, 2003, 205–215.

[16] Pimentel, B and Campos, M, "Cooperative communication in ad hoc networked mobile robots," in Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, Las Vegas, NE, Oct., 2003, 2876–2881.

[17] Antonelli, G, Arrichiello, F, Chakraborti, S and Chiaverini, S, "Experiences of formation control of multi-robot systems with the null-spacebased behavioral control," in Proc. IEEE International Conference on Robotics and Automation, Roma, April, 2007, 1068–1073.

[18] G. Antonelli, F. Arrichiello and S. Chiaverini, "Experiments of formation control with multirobot systems using the null-space-based behavioral control," IEEE Transactions on Control Systems Technology, Vol. 17, No. 5, pp. 1173–1182, 2009.

[19] F. Nagata, T. Yamashiro and K. Watanabe, "Cooperative swarm control for multiple mobile robots using only information from PSD sensors," Artificial Life and Robotics, vol.16, no. 1, pp. 116–120, 2011.

[20] N. Kitahara, F. Nagata, A. Otsuka, K. Sakakibara, K. Watanabe and M. K. Habib, "A proposal of experimental education system of mechatronics," in Proc. of 17th International Symposium on Artificial Life and Robotics, Beppu, Oita, Japan, January, pp. 166–169, 2012.

[21] K. Watanabe, Y. Shiraishi, S. G. Tzafestas, J. Tang and T. Fukuda, "Feedback control of an omni-directional autonomous platform for mobile service robots," Journal of Intelligent & Robotic Systems, vol. 22. nos. 3/4, pp. 315–330, 1998.

[22] Asama, H , Ishida, Y, Ozaki, K, Habib, MK, Matsumoto, A, Kaetsu, H, and Endo, I "Communication system between multiple robotic agents.," the 1992 Japan-USA Symposium on Flexible Automation Part 1(of 2), 1992, 647-654.

[23] R. A. Brooks, "A robust layered control system for a mobile robot," IEEE Journal of Robotics and Automation, vol. 2, no. 1, pp. 14–23, 1986.