Advances in Robotics Research, *Vol. 2, No. 1 (2018) 1-11*
DOI: https://doi.org/10.12989/arr.2018.2.1.001

Bio-inspired robot swarm control algorithm for dynamic environment monitoring

Kyukwang Kim[1a], Hyeongkeun Kim[2b] and Hyun Myung[*1]

[1]*Urban Robotics Laboratory(URL), Korea Advanced Institute of Science and Technology, 291 Daehak-ro, Yuseong-gu, Daejeon 34141, Republic of Korea*
[2]*Department of Mechanical Engineering, Korea Advanced Institute of Science and Technology, 291 Daehak-ro, Yuseong-gu, Daejeon 34141, Republic of Korea*

(Received March 23, 2016, Revised Septemer 29, 2017, Accepted October 8, 2017)

Abstract. To monitor the environment and determine the source of a pollutant gradient using a multiple robot swarm, we propose a hybrid algorithm that combines two bio-inspired algorithms mimicking chemotaxis and pheromones of bacteria. The algorithm is implemented in virtual robot agents in a simulator to evaluate their feasibility and efficiency in gradient maps with different sizes. Simulation results show that the chemotaxis controller guided robot agents to the locations with higher pollutant concentrations, while the pheromone marked in a virtual field increased the efficiency of the search by reducing the visiting redundancy. The number of steps required to reach the target point did not increase proportionally as the map size increased, but were less than those in the linear whole-map search method. Furthermore, the robot agents could function with simple sensor composition, minimum information about the map, and low calculation capacity.

Keywords: bio-inspired; chemotaxis; pheromone; swarm control; environmental monitoring

1. Introduction

Using autonomous robots for environment monitoring and exploration is a widely researched area. Many related works such as monitored area coverage and data collection have been reported (Gabriely and Rimon 2001, Dhariwal and Sukhatme 2004, Trincavelli *et al.* 2008, Fazli and Mackworth 2012, Kim *et al.* 2011, 2014, Silva *et al.* 2015). Generally, an area-coverage algorithm such as a spanning tree has been used for a single robot agent that checks the entire region of interest (Zheng *et al.* 2005). This method is very intuitive to implement and the gathered data is easy to process. However, it becomes inefficient when the coverage region is larger than that can be covered by a single robot agent (Leonard and Olshevsky 2013).

To solve the area coverage problem, multiple robotic agents can be applied together. Multiple

∗Corresponding author, Professor, E-mail: hmyung@kaist.ac.kr
[a]M.Sc Student, E-mail: kkim0214@kaist.ac.kr
[b]Undergraduate Student, E-mail: hkkim1227@kaist.ac.kr

 ISSN: 2287-4976 (Print), 2287-4984 (Online)

agents can search an area and simultaneously decrease the area required to be covered per unit. Spanning tree algorithms have been extended to multi-spanning trees to give deterministic solutions, ensuring every region of the monitoring area is covered with multiple robots. However, the deterministic method is not the optimal solution in some cases. For example, if the goal of the monitoring is to find a pollution source and not the total pollutant state of the area, an algorithm that converges to the source without checking the whole region is a better solution. Furthermore, dynamic environments such as air or water flow can affect the gradient or distribution of the pollutant. In this case, although an entire pollutant map of the region is built, it may quickly be outdated.

Stochastic algorithms that can deal with dynamic changes as well as deterministic problems are introduced (Smith *et al.* 2011). Stochastic methods can also be applied to less intelligent robots or under deficiencies such as platform incompetence or sensor inaccuracy (Renzaglia *et al.* 2011). Among these methods, this paper further investigated and modified the bacteria-inspired controller developed by Oyekan and Hu (2009, 2010), which used a bacterial random walk pattern to solve a pollutant search problem. Furthermore, to control multiple agents with the bacteria inspired algorithm, a multiple-agent controlling method called a pheromone controller suggested by Sauter *et al.* (2005) was modified and applied.

2. Bio-inspired robot control algorithms

2.1 Chemotaxis-inspired search algorithm

Bacteria are known to show a behavior that involves moving toward the chemical gradients that they use as a food source. Bacterial motion consists of two phases: the roll (also known as the tumble phase) and the run. The roll entails random selection of the heading direction, while the run involves moving toward the selected heading. Generally, the frequencies of the roll and run phases are fixed at a certain ratio. However, when bacteria detect a food chemical, the time for the run phase increases, meaning they run toward a certain direction with higher probability of existence of food rather than searching randomly. This mechanism is illustrated in Fig. 1(a).

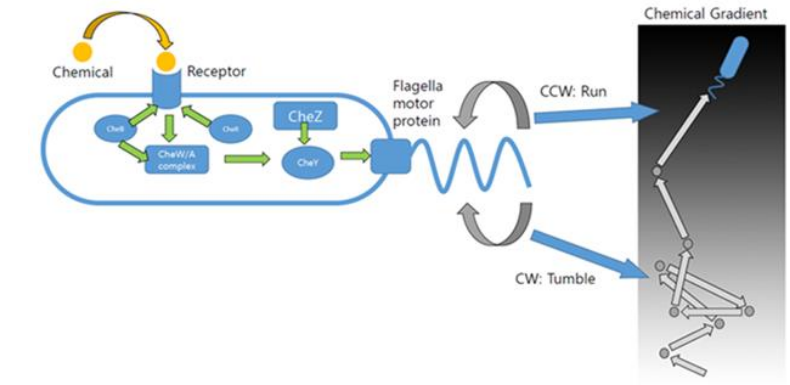
Oyekan and Hu (2009) introduced a mathematical model of chemotaxis based on the Berg and Brown model as follows

$$\tau = \tau_0 \exp\left(\alpha \overline{\frac{dP_b}{dt}}\right) \tag{1}$$

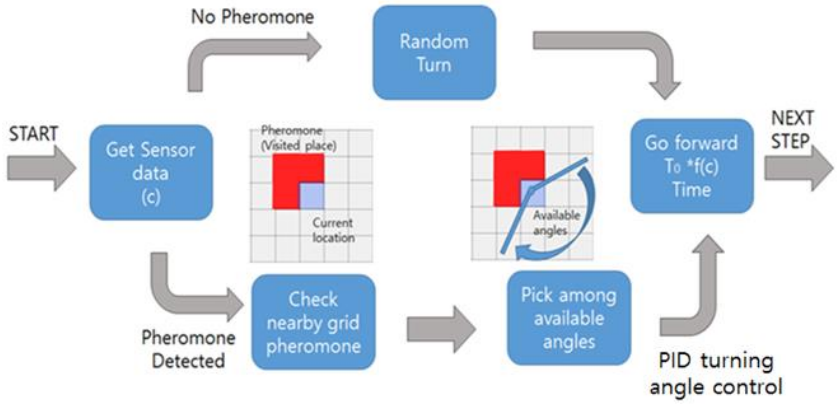
$$\overline{\frac{dP_b}{dt}} = \tau_m^{-1} \int_{-\infty}^{t} \frac{dP_d}{dt'} \exp\left(\frac{(t'-t)}{\tau_m}\right) dt' \tag{2}$$

$$\frac{dP_d}{dt'} = \frac{k_D}{(k_D + C)^2} \frac{dC}{dt} \tag{3}$$

where the value τ is the time spent for the run phase (τ_0 is the minimum run phase time), P_b is the fraction of the chemical bounded receptor under concentration C, with a dissociation constant of k_D, and α, P_d, are τ_m are the system constants. Oyekan and Hu (2010) used this model to simulate a concentration gradient climbing robot by assigning certain values to k_D, α, and τ_0 to optimize the system. However, applying this model directly to real robots is difficult, as robots use



(a)



(b)

Fig. 1 (a) Illustration showing the chemotaxis behavior and (b) Flow chart of the proposed method

sensors instead of receptors, and commercial sensors usually emit a certain voltage or digital values as the output, instead of the constants of the Berg and Brown model. In this paper, the bacterial chemotaxis controller was simplified in the form of the following single equation

$$\tau = \tau_0 \cdot f(c) \tag{4}$$

where the weighting function f receives the sensor value c as an input and gives a weighed output proportional to the measured concentration from the sensor. The proportionally weighted output increases τ, which increases the runtime compared to that of the sampled pollutant concentration at a certain point. This simplified form of the chemotaxis controller is more suitable for actual implementation and can be used for real robotic agents. However, for optimization, changing the formula for weighting function f can give a more appropriate response instead of optimizing three parameters (τ_0, k_D, P_b). The roll phase was implemented for robot agents by randomly selecting a certain angle as the heading angle. The flow chart of the proposed procedure is shown in Fig. 1(b).

Robot agents operate by changing between two phases suitably: the roll phase to determine direction and then the run phase to move toward the selected heading angle. The concentration of the pollutant is then sampled again at the newly moved point, and the run phase time and direction

are re-calculated. Bacterial memory time and other parameters are ignored to simplify the controller.

2.2 Pheromone-inspired compensation algorithm

The bacterial chemotaxis controller can be used independently as a robotic controller. However, it relies on a random selection process, which can cause inefficiency during the operation. Furthermore, to control the multiple robot agents with the bacterial controller, another algorithm to coordinate the interactions between the agents in a swarm is required. For this task, we propose a digital pheromone controller as a multiple agent controller.

A pheromone is a chemical discharged into the surrounding environment functioning as a signal for communication and information exchange to nearby swarm members (Passino 2002, Polverino *et al.* 2012, Bonabeau *et al.* 1999, Parunak 1997). Ants are a well-known entity involved in pheromone use, with one of the better examples of the bio-inspired method being known as the Ant Colony Optimization process (Zhang *et al.* 2014). Bacteria also employ chemical signals to interact with the nearby bacterial cells. Exchanging pheromones and responding to nearby pheromones decrease the complexity of robot agents, as each agent can communicate and coordinate without information exchanging routine or higher-level intelligence. Programming agents simply to response and release pheromone into the shared environment will automatically control them, similar to the self-organizing mechanism.

A digital pheromone is modeled in the virtual setting based on a real pheromone. Sauter *et al.* introduced the terms such as place agents and walker to describe a gridded virtual space corresponding to the real area and robot agents in the virtual place (Sauter *et al.* 2005).

This research implemented the Sauter's model in a simplified form. Grids were generated on a virtual space and N was defined as the number of adjacent grids within one grid distance from an arbitrarily selected grid. Each grid has a certain amount of pheromones that evaporate and propagate to nearby grids. Instead of using the propagation and evaporation factors, the amount of pheromone that flows into and out of the current grid was set to a specific value for simplicity. Moreover, the threshold was not calculated. A simplified form of the pheromone per single grid state is defined by the following equation

$$S(p,t) = S(p,t-1) + N(P_o - P_i) - P_e \tag{5}$$

where N is the number of adjacent grids, P_o is the amount of pheromone propagated to the adjacent grids, P_i is the amount of pheromone that flows into the current grid, and P_e is the amount of evaporation. Strength (S), which was not calculated separately, was defined as the total amount of pheromone per grid (p) at time t. The concept of the pheromone grid used in this research is illustrated in Fig. 2.

If a pheromone marked the already visited paths, the robot would disregard the marked direction during the random heading change process at the roll phase to decrease the overlap of the visited paths between the agents. Robots would also disregard the direction that the pheromones indicated as having a high pollutant level, since this would indicate a pollutant source that was already determined and, therefore, did not need to be visited again.

3. Materials and methods

Two types of simulation environments were prepared to test the hybrid algorithm. The

feasibility of the algorithm was tested by running a single robot with a chemotaxis controller introduced in a small map to search for a light source, first guided and then unguided by the pheromones. In the case of using the chemotaxis controller alone, the robot freely searched the map until a region was found with the strength of light higher than the given threshold. The square-shaped map was divided into an 8-by-8 grid for the case of the pheromone guided robot. Under the same conditions, the robot marked the pheromones in virtual grids visited before. Directions that were marked with pheromones were not considered during the random spinning phase, as described previously. The X- and Y-coordinates of the robot's position at every phase were recorded to track the robot's path to check the converging pattern. The Webots simulator (Michel 2004) was used to set up the simulation environment and to model the virtual robot. A differential wheel robot with a compass sensor, a light sensor, and a GPS was modeled and used in the Webots simulator.

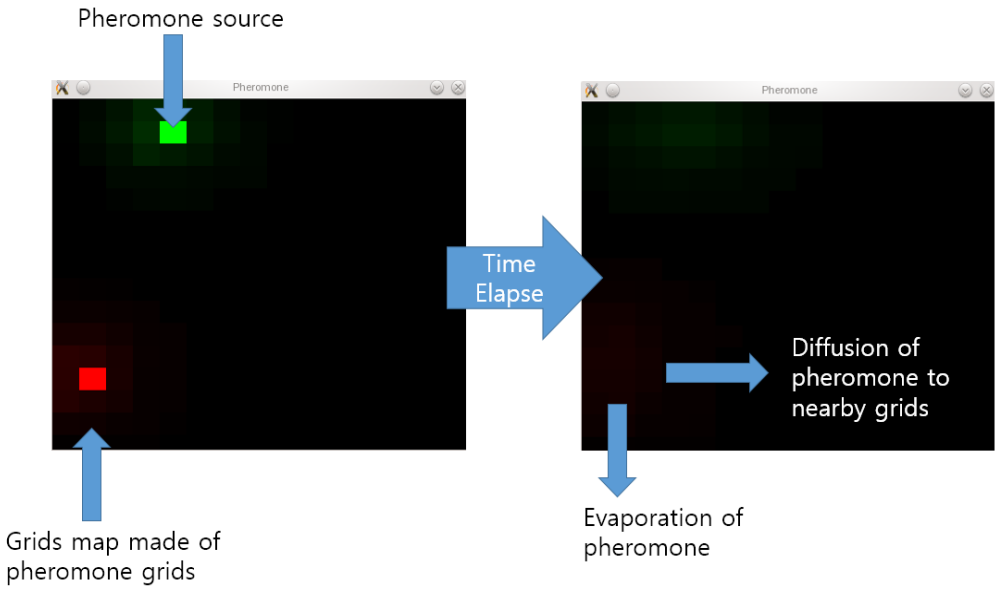


Fig. 2 Pheromone grid setup: Propagation of the pheromone to nearby grids with the elapse of time, diffusion and evaporation eventually remove the old markers

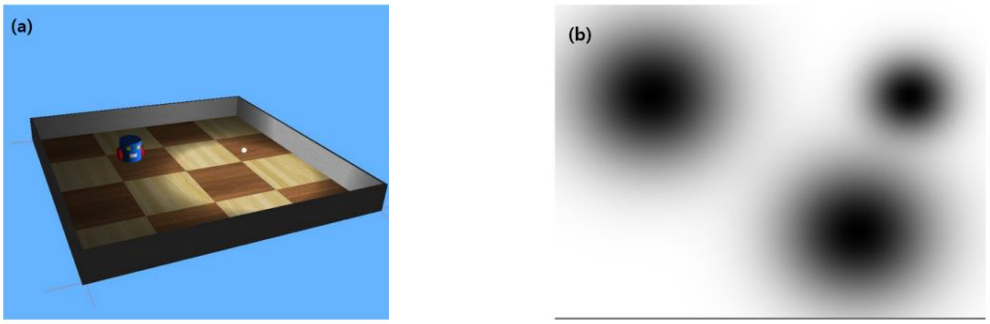


Fig. 3 Experimental setup for simulations, (a) Webots simulator settings and (b) Image mimicking the pollutant sources in the environment, used for the Python-based code

A Python-based code was developed to simulate the algorithm multiple times to check the performance of the pheromone algorithm. As the chemotaxis controller was a stochastic process that included random processes, it was necessary to compare the results of multiple runs and analyze their performance distribution. Grayscale images with multiple color gradients were used to describe the real-world pollutants. The dark regions of the images, where the pixel value was close to 0, were regarded as the polluted regions and the amount (concentration) of the pollutant was defined by subtracting the pixel value from 255 (maximum grayscale pixel value). Pheromone images were divided into a 11-by-11 grid, where each grid contained a pheromone corresponding to the monitoring area. Pheromones representing the visited paths were marked with green color and the pheromones showing high pollutant concentrations were marked with red color. The robot agent was programmed to find all sources existing on the gradient field. The paths of the robot were marked with red/blue squares, which were linked with the lines. The setup of each simulation is shown in Fig. 3.

After multiple runs (total 50 times) in the environment in Fig. 3(b), the number of steps and trajectories required to find all three gradient sources were recorded. The Wilcoxon rank-sum test was used to check whether the simulation cases using the pheromone showed significant differences from the simulations without pheromones.

For comparison with the classical algorithm, the spanning tree area-search method (Gabriely and Rimon 2001) with Prim's algorithm (Prim 1957) was implemented in the Python environment. Like the Webots simulation, an 8-by-8 grid map (changed to 16-by-16 grid if the robot moved 0.5 time step for simulation scaling) was generated. The robot agents were programmed to search the entire map to find the regions with higher chemical (dark pixel) concentration.

4. Results and discussion

Convergence to the gradient source was tested for two simulation environments. The bacterial chemotaxis controller successfully found sources in the maps, both with and without the aid of the pheromone. Fig. 4 illustrates the recorded robot paths on the simulation field.

The robot agents with the chemotaxis controller functioned as expected, though some inefficiency was observed. The two main inefficient cases included the "re-visiting the same place" and "turning back" scenarios. The bacterial controller is designed to move a smaller distance when the detected pollutant concentration is low. At the starting point where the pollutant concentration is low, the robot agents cannot move effectively to reach the goal point. However, if the random heading selection process (roll phase) fortunately chooses a direction that will lead the robots close to the goal point, the increased amount of the pollutant will provide more opportunity to the robots by increasing the distance they can travel. However, in an unfortunate case, the roll phase may select the wrong headings. In the worst case, a 180° movement can be chosen at the first roll and a 360° movement can be selected subsequently, which will eventually cancel the movement of the robot. These reasons can also lead to the robots being stranded at the starting point for a certain period. Furthermore, even after having left the starting point, a random selection of the heading can also cause the problem of sending two robots in a same direction, which then decreases the area covered. Another problem is 'turning back,' which is also caused by random selection of the heading. After escaping and converging to the source, if the gradient level increases, the distance the robot can move increases. However, even if a region with higher

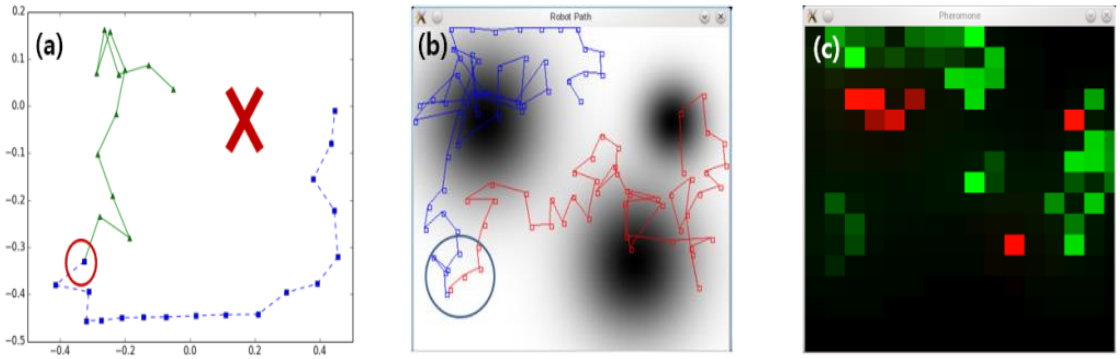


Fig. 4 (a) Robot paths in Webots simulator. The red circle denotes the starting point and the X mark denotes the light source position where the brightness gradient is formed at near fields. The green path denotes the robot without the pheromone while the blue path denotes the robot guided by the pheromone, (b) Multiple robot agents in the second simulation field. The blue circle denotes the starting point of the two robots and (c) Pheromone field corresponding to (b). Reds indicate the pollutant sources and greens indicate visited grids

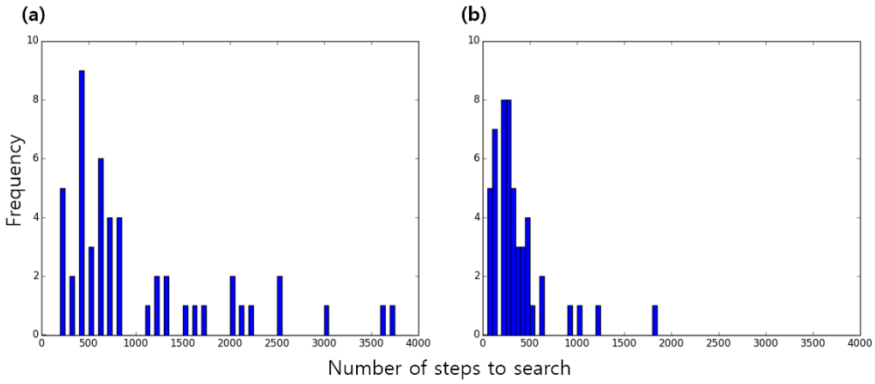


Fig. 5 Number of simulation steps required for multiple robot agents to find all three sources in Fig. 4(b). The simulation was repeated 50 times; the *x*-axis is the step number and the *y*-axis is the number of simulation cases finished within the specified time step. (a) Simulation step distribution of robot agents not guided by the pheromone and (b) Simulation step distribution of robot agents guided by the pheromone

concentration is directly in front of the robot (assuming that the robot's previous heading was 180° and a high pollutant concentration region is located in the same direction), the roll phase can select a 0° heading, which will deviate the robot from the best course. The robot paths shown in Fig. 4(a) illustrate this situation. The green robot overlaps the paths or turns to the wrong region while searching for the light sources.

The use of the pheromone controller helps reduce these inefficiencies. As mentioned before, the modified bacterial algorithm uses pheromone information during the roll phase. Directions marked with the pheromone indicate the visited places (the green boxes in Fig. 4(c)) and are not included in the random heading angle selection. This ensures that visited places in the previous

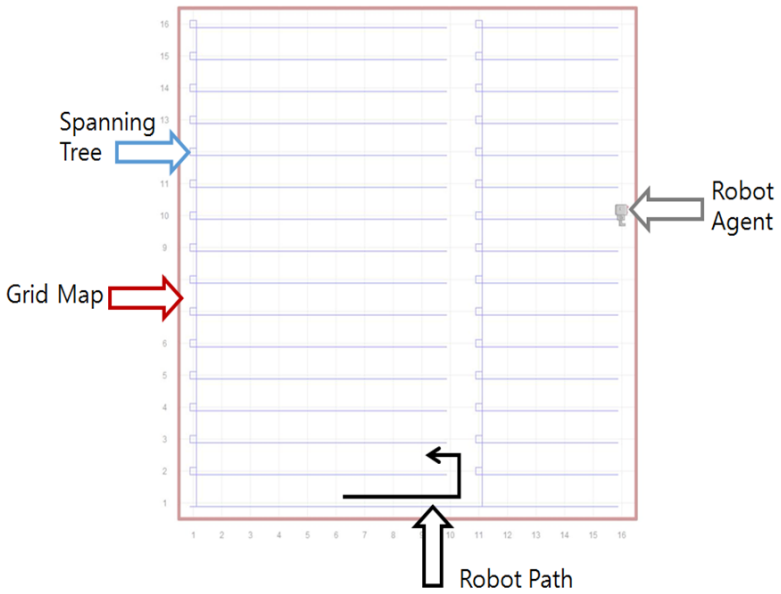


Fig. 6 Robot movement on the grid map with classical whole-map scan algorithm. Blue lines indicated by the blue arrow shows the calculated spanning tree covering the whole map with least overlaps. The robot follows the blue line counterclockwise to visit whole grid

few steps are not revisited again unless the pheromone evaporates after a certain amount of time. This solves the problem of the bacteria controller originating from the random selection process and possibly selecting the worst or bad cases. The pheromone reduces this case by 'memorizing' the previous robot path or the direction data. The starting point contains a high concentration of the green pheromone, thereby causing the robot agent to repel each other from the starting region. In the case of multiple robots, two robots will repel each other as the green pheromone concentration becomes higher when the two robots become close to each other. Area coverage also increases as multiple agents have a low possibility of searching the same area due to pheromone marking. The blue path (robot path guided by pheromone) in Fig. 4(a) shows a straight and non-overlapping path compared to the green path in the same field.

For a more accurate measurement of the performance difference between pheromone guided and unguided cases, multiple simulations were run to obtain the distribution of the results, as the chemotaxis algorithm is a stochastic process. The results of the repeated simulations are shown in Fig. 5.

Pheromone-guided robots showed increased efficiency with the increase in the number of robots. The maximum number of time steps required was 600, disregarding the outliers. Compared to the robots not guided by a pheromone, which recorded over 1,000 steps to find all three sources, the efficiency increased as the number of robots increased. The maximum value of the outlier also decreased compared to the cases of a single robot with the pheromone and multiple robots without the pheromone. A Wilcoxon rank-sum test (Mann and Whitney 1947, Hodges and Lehmann 1963) was conducted to measure the difference between the two distributions. The null hypothesis was set as "two distributions have no difference" and the alternative hypothesis was set as "two distributions have a significant difference". As the number of simulations was sufficiently large

(n>20), the obtained U-statistic (419.5) was used for the Z-distribution assumption (Carine *et al.* 2010), and was converted to Z-statistics (-5). The value of -5 in the Z-statistics was nearly 0, which meant that the U-statistics of the Wilcoxon rank-sum test were sufficiently small to reject the null hypothesis with a p-value<0.05. Thus, it can be confirmed that the usage of the pheromone guides ensures better performance statistically.

Although not simulated in this study, another merit of the controllers used in this research is that they can deal with the moving or dynamic targets. As the target of the monitoring area can be a dynamic source such as air or water, a pollutant source with high concentration regions can move or drift away as the time passes. The pheromone markers in the virtual fields were designed to evaporate as the time passed. If the target slowly moved and changed its location, the robots could relocate to another region by marking a new pheromone after evaporation of the pheromone containing the previous information.

When the robot agent with the spanning tree-based whole-map search algorithm scanned the same grid map (see Fig. 6), 255 steps were required in total for the robot to search the whole grids on the map. Compared to the 12-18 steps performed by the bacteria-inspired controller (Fig. 4(a)) on the same map, the spanning tree-based algorithm is computationally more expensive. Though the spanning tree-based algorithm assures the search of the global maximum, the bacteria controller with a well-tuned threshold can avoid the local maximum, as the concentration gradient has its maximum at its center. The result shows that the stochastic method can reduce the cost gradually in a gradient center searching case.

5. Conclusions

To control multiple robots for environment monitoring, a robot swarm control algorithm and a pollutant source finding algorithm were developed and tested. A hybrid algorithm that combined the simplified bacteria-inspired controller and a pheromone swarm controller was proposed and implemented. The implemented bacterial controller could locate the pollutant sources in the map and the pheromone controller reduced the inefficiency of the bacterial controller caused by the probabilistic process. The simulation results showed that the time step required to complete the mission was decreased, and an increased number of robots were controlled more efficiently in the presence of the pheromone. It is notable that each robot agent was programmed with a simple function: reacting to the sampled pollutant and pheromone concentration. The pheromone controller helped the simple and less intelligent robot swarms in performing the complex task more efficiently, by collaborating with other agents.

The authors expect that these algorithms can be implemented in real robots, and can be adjusted to various types of pollutants and their distributed forms. If real data samples of pollutants are obtained, simulators can be used to determine the optimal weighting function, pheromone evaporation, and propagation constant. Moreover, new weighting functions can be tested or developed, for example, machine learning or evolving methods can be used.

Currently, the simulator is designed as a single thread, calculating the states of the robots and the pheromone grids in the field one-by-one. Increasing the number of robots or pheromone grids can cause heavier loads or time delay to the system. If these states can be calculated in a parallel processor, a large swarm (more than 100 robot agents) can be simulated or operated simultaneously to cover areas larger than what current robots can cover. Multithread or multiprocessor-based parallel calculation and CUDA-based GPU parallelization are expected to be

candidates for this purpose.

Acknowledgements

This research was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (NRF-2015R1A2A1A15056262). This work was also supported by the Ministry of Trade, Industry & Energy (MOTIE, Korea) under Industrial Technology Innovation Program No.10067169, 'Development of Disaster Response Robot System for Lifesaving and Supporting Fire Fighters at Complex Disaster Environment'. The students are supported by Korea Minister of Ministry of Land, Infrastructure and Transport (MOLIT) as "U-City Master and Doctor Course Grant Program".

References

Bonabeau, E., Dorigo, M. and Theraulaz. G. (1999), *Swarm Intelligence: From Natural to Artificial Systems*, Oxford University Press, New York, U.S.A.

Carine, A., Bellera. M., Julien, J. and Hanley, A. (2010), "Normal approximations to the distributions of the Wilcoxon statistics: Accurate to what N? Graphical insights", *J. Statist. Edu.*, **18**(2), 1-17.

Dhariwal, A. and Sukhatme, G. (2004), "Bacterium-inspired robots for environmental monitoring", *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, New Orleans, Louisiana, U.S.A., April.

Fazli, P. and Mackworth, A.K. (2012), "Multi-robot repeated boundary coverage under uncertainty", *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Guangzhou, China, December.

Gabriely, Y. and Rimon, E. (2001), "Spanning-tree based coverage of continuous areas by a mobile robot", *Ann. Math. Artif. Intell.*, **32**(1-4), 77-98.

Hodges, J.L. and Lehmann, E.L. (1963), "Estimation of location based on ranks", *Ann. Math. Stat.*, **34**(2), 598-611.

Kim, C.H., Seo Y.C., Eom H.S., Choi, Y. and Kim, Y. (2014), "Aerial working environment monitoring robot in high radiation area", *Proceedings of the International Conference on Control Automation and Systems (ICCAS)*, Goyang, Republic of Korea, October.

Kim, Y.J., Myeong, H.J., Kim, C.S., Cha, N.J. and Kim, D.H. (2011), "Implementation of smartphone environment remote control and monitoring system for Android operating system-based robot platform", *Proceedings of the International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, Incheon, Republic of Korea, November.

Leonard, N.E. and Olshevsky, A. (2013), "Nonuniform coverage control on the line", *IEEE T. Automat. Control*, **58**(11), 2743-2755.

Mann, H.B. and Whitney, D.R. (1947), "On a test of whether one of two random variables is stochastically larger than the other", *Ann. Math. Stat.*, **18**(1), 50-60.

Michel, O. (2004), "Webots: Professional mobile robot simulation", *J. Adv. Robot. Syst.*, **1**(1), 39-42.

Oyekan, J.O. and Hu, H. (2009), "Toward bacterial swarm for environmental monitoring", *Proceedings of the IEEE International Conference on Automation and Logistics (ICAL)*, Shenyang, China, August.

Oyekan, J.O. and Hu, H. (2010), "Bacteria controller implementation on a physical platform for pollution monitoring", *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Anchorage, U.S.A., May.

Parunak, H.V.D. (1997), "'Go to the ant': Engineering principles from natural agent systems", *Ann. Oper. Res.*, **75**, 69-101.

Passino, K.M. (2002), "Biomimicry of bacterial foraging for distributed optimization and control", *IEEE*

Contr. Syst. Mag., **22**(3), 52-67.
Polverino, G., Abaid, N., Kopman, V., Macrì, S. and Porfiri, M. (2012), "Zebrafish response to robotic fish: Preference experiments on isolated individuals and small shoals", *Bioinspir. Biomim.*, **7**(3), 036019.
Prim, R.C. (1957), "Shortest connection networks and some generalizations", *Bell Labs Tech. J.*, **36**(6), 1389-1401.
Renzaglia, A., Doitsidis, L., Martinelli, A. and Kosmatopoulos, E.B. (2011), "Adaptive-based distributed cooperative multi-robot coverage", *Proceedings of the American Control Conference (ACC) 2011*, San Francisco, California, U.S.A., June.
Sauter, J.A., Matthews, R., Parunak, H. and Brueckner, S.A. (2005), "Performance of digital pheromones for swarming vehicle control", *Proceedings of the International Joint Conference on Autonomous Agents and Multi-Agent Systems*, Utrecht, Netherlands, July.
Silva, J.A.M, Silva, M.F. and Barbosa, R.S. (2015), "Design and implementation of a biological inspired swimming robot", *Adv. Robot. Res.*, **1**(3), 239-250.
Smith, S.L., Schwager, M. and Rus, D. (2011), "Persistent monitoring of changing environments using a robot with limited range sensing", *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China.
Trincavelli, M., Reggente, M., Coradeschi, S., Loutfi, A., Ishida, H. and Lilienthal, A.J. (2008), "Towards Environmental monitoring with mobile robots", *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, Nice, France, September.
Zhang, Z., Gao, C., Liu, Y. and Qian, T. (2014), "A universal optimization strategy for ant colony optimization algorithms based on the Physarum-inspired mathematical model", *Bioinspir. Biomim.*, **9**(3), 036006.
Zheng, X., Jain, S., Koenig, S. and Kempe, D. (2005), "Multi-robot forest coverage", *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, Edmonton, Canada, August.

HM