In this paper, we invest an effect of task partitioning as the supplement to the navigation using path integration in a harvesting task of a robotic swarm. We present a simple method based on the response threshold model to allocate an individual robot to a partitioned task and show that the task partitioning will increase system performance by reducing the negative effect of path integration errors. The results of proposed method are analyzed and compared to case that task partitioning is not employed to a given task.

We apply an approach of an autonomous task partitioning in a collective harvesting task in a circular arena as shown in Figure 1. The small empty circle at the center of arena is a nest, and the source area where the clustered objects are placed is located in the right side of the nest in a constant distance. All robots stay around a nest at first, and move randomly to search objects. If robots find objects, robots return with them to nest just using path integration as their only means of navigation. Robots put down carrying objects after arriving at nest and they turn direction to move to an object source to harvest other objects using path integration. Path integration is a process that uses cues to track distance and direction in order to estimate a current position to get home, and is widely used by animals for dead reckoning navigation (Mittelstaedt and Mittelstaedt, 1982).

During the harvesting task, if robots fail to reach the desired destination in the end of navigation, robots move randomly until they arrive at nest or source. If the path integration error is high or the distance of navigation is long, the probability that robots fail to find the nest is high and these wandering behaviors lower an overall system performance. Under these conditions, task partitioning (Jeanne, 1986) will divide harvesting tasks into small tasks with a short distance and reduce the negative effect of path integration errors. There are many examples of task partitioning concerning the collection of food and other materials in insect societies (Ratnieks and Anderson, 1999).

To implement task partitioning, each robot decides its current task based on the response threshold model (Theraulaz et al., 1998). The response threshold model has some similarity with so-called market-based model of task allocation where agents bid for a given task. In our work, robots switch their current tasks on the way from source to nest after harvesting objects by dropping carrying objects in the current positions if the moving distance $D_m$ of robots is higher than a certain threshold $\theta_d$ ($0 < \theta_d < D_m$) and they detect other robots not carrying objects within a detecting range.

Naturally, robot’s threshold of moving distance should be a function of an average moving distance from source to base and the probability that robots detect other robots carrying no objects. Therefore, an optimal threshold for task switching depends on environment components, such as distance between nest and source and density of robots. In addition, the entire environment information is not available to an individual robot, and each robot should choose the best commensurate task with its current states using its own local
knowledge alternatively. Under these limitations, choosing an optimal threshold is a complex problem. In our work, we limit ourselves to a simple method for setting this task switching threshold randomly. Threshold value for task partitioning is selected randomly whenever the robot harvests an object and each robot changes its task based on this value.

We compare the system performance by counting the number of collected objects by all robots. Figure 2 shows the averaged results of 20 repetitions with various path integration errors in two strategies, non-partitioned and randomly partitioned. When the path integration direction error is bigger than a certain error (in this paper, \( \pm 5.2 \) degree that is related to navigation distance and size of nest), the task partitioned method produces more improved performance. On the contrary to this, there is small direction errors, navigation returning to base using task partitioning obtains better performance than without task switching. Frequent task switching causes repetitions of dropping and re-harvesting behaviors and these additional behaviors reduce the number of harvesting objects if the path integration error is low.

Figure 3 shows effects of number of robots and navigation distance when the direction error is set to \( \pm 9 \) degree. As the number of robots is increased, the probability that robots see another robot is increased and task partitioning helps to improve system performances. On the other hand, the number of harvesting objects is decreased as the increasing of navigation distance. In both results, task partitioned navigation shows better results than non-partitioned method.

The goal of our experiments is to investigate whether task partitioning can reduce the negative effects of path integration errors in navigation. We validate our approach using simulation-based experiments and study how task partitioning can improve the performance of our harvesting task. Robots randomly decide the moving distance threshold and the experiments results show that if the path integration error is high, the robots that switch tasks perform better than those that don’t, but if the error is low, non-partitioned strategy is advantageous. The proposed strategy is very simple and far from being an optimal solution, nevertheless we improved the performance when the path integration error is high. There will be an adaptive way to find more optimal task switching function to achieve better performance, and we will leave it as a future work.

Acknowledgements

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MEST) (No. 2012R1A2A4A0105677).

References


