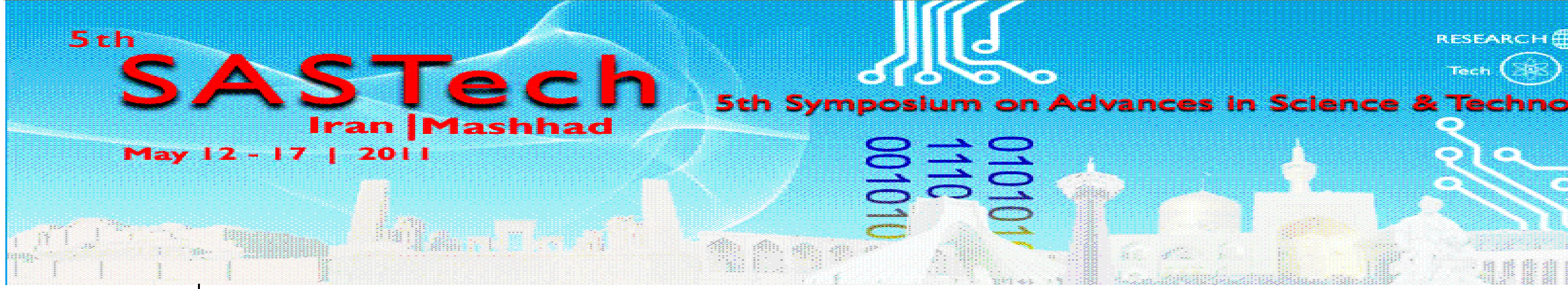

Radial FTP

Z. Moezkarimi (zmoezkarimi@aut.ac.ir),
A. Bagheri (ar_bagheri@aut.ac.ir)
Amirkabir university of technology
Paper Reference Number: 0702-671
Name of the Presenter: Z. Moezkarimi

Abstract

The Freeze Tag Problem (FTP) arises in swarm robotics. In FTP one active robot should awakes some inactive robots. Robots that are awaked may help to awake other inactive robots. For activating an inactive robot in FTP, the awakening robot should be in the physical proximity of it. In this paper, we study a variant of FTP in which for activating inactive robots it's sufficient to go to a given distance L of the robots. In fact a robot only has to get “close” to another robot in order to activate it. We call this problem "Radial FTP" and give an approximation algorithm for it.

Key words: Freeze Tag Problem, swarm robotics, approximation algorithm, optimization.

1. Introduction

In Freeze Tag Problem there are a large number of inactive robots and one active robot and the goal is to awaken all of the inactive robots in the shortest possible time. The new awakening robot can assist in activating other sleeping robots. For activating an inactive robot in FTP, the awakening robot should be in the physical proximity of it. Now we consider a variant of FTP in which for activating inactive robots it's sufficient to go to a given distance L of the robots. We call this problem *Radial FTP*. For example one application of Radial FTP arises in the context of distributing data in wireless sensor networks. Imagine that we have a large number of robots equipped with wireless sensors. The range sensors are limited. We can awake a robot only by entering the region of the range sensors. In order to awakening an inactive robot, the active robot moves toward it. Once the active robot touches the side of the range sensor of the inactive robot, it can awake the inactive robot. This problem has been mentioned in [2] for future research.

The FTP is similar to some problems like the minimum broadcast time problem [8], the multicast problem [3], and the related minimum gossip time problem [7] but the proximity required in the FTP leads to significant differences.

FTP is NP-hard, even for the case of star graphs with an equal number of robots at each vertex [2]. Many solutions for FTP have been proposed including approximation algorithms, heuristic strategies, local search and computational intelligence.

Some solution proposed for star graphs [2], Ultrametrics [2] and Online FTP [6]. There are some Density-Based strategies and Sibling-Based strategies [1]. The heuristic strategies for this problem divide into two groups: the greedy strategies and the sector-based strategies. The main idea in greedy strategies is select the closest robot in each step. The greedy strategy is not a fully defined heuristic. What remains to be specified is how conflicts among robots are resolved, since two robots may have the same closest neighbor. In [9] Sztainberg et al. describe three methods for resolving these conflicts: claims, refresh, and delayed target choice. The weakness of greedy strategy is sometimes a robot prefers to travel a longer distance to obtain a better payoff. Thus some other heuristic strategies were designed. We call these strategies sector-based strategies because the authors use the sectors around the robots for making decisions. The robot can detect the closest robot within each of its sectors. Three methods that have been proposed in [10] are Bang-for-the-Buck, Random Sector Selection and Opposite Cone. In the field of local search algorithms and computational intelligence three are methods for the FTP: Alternating Path Algorithm [5], a solution based on genetic algorithms [5] and a solution based on ant algorithms [4]. Recently Zoraida proposed a new method for FTP based on DNA-computing [12]. There are also some open problems related to FTP (see [2, 11]). But it's the first time that the Radial FTP has been studied.

The Radial FTP is equivalent to touch a circle with radius L around the robots. We can describe the problem as follows. In Euclidean space, there are n circles and the goal is to touch all of them in the shortest possible time. If L=0, Radial FTP converts to the FTP. Figure 1 illustrates the awakening process. In this figure robot A awakes robot B when get close to B within distance L, then travels toward robot F. This operation will be continued until the last robot awake.

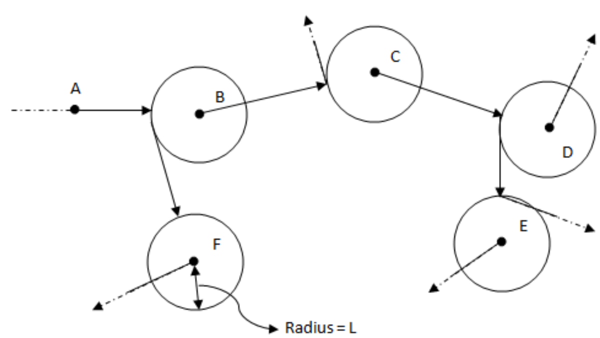

Fig 1: activating inactive robots by going to a given distance L of them.

2.Comparing FTP and Radial FTP

In Radial FTP, makespan time for a set of robots may be very different with FTP. For example in figure 2 the distance between robot V_r and the edge of cycle around robot A is ε. The distance between robot A and the edge of cycle around robot B is ε too. The distance between robots V_r and A and between robots A and B is α. If we consider FTP for this instance and solve it, then the makespan time will be 2α. If we consider Radial FTP for this instance and solve it, then the makespan time will be 2ε. Robot V_r should go to distance L from A, at this time new awakening robot A can go to the distance L from B and awake it. If the ratio of α to ε was too much, the difference between the solutions will be too much.

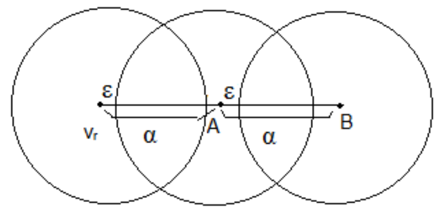

Fig 2: The difference between the solutions for FTP and Radial FTP.

Another example is when the robots locate in the place such that all of the circles around the robots have overlay. See figure 3. It means that each robot have the distance smaller that L with at least one robot. For the FTP the solution will be at least the radius of the set of robots. If we consider Radial FTP the makespan time will be zero!!

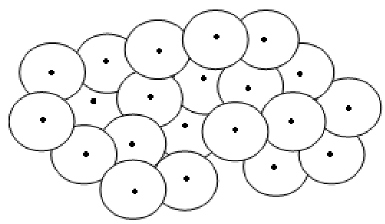

Fig 3: The case where the circles around robots have overlay.

3. An approximation algorithm for Radial FTP

As mentioned, in Radial FTP for awakening robots it's sufficient to go to a given distance L of the robots. Thus the distances should be travelled in Radial FTP and the makespan time is less than which is in FTP. We imagine that each robot moves with unit speed so the distances travelled by robots is equivalent to the time required for wake-up process. For analyzing the relation between the makespans of the Radial FTP and FTP consider the following scenario.
Imagine that robots A and B are two robots from a set of robots in FTP (see figure 4). In some steps of the solution if one robot wants to awake robots A and B, it should goes to the point A, awakes robot A and then travels the edge a and then awakes robot B. If the active robot wants to awake robots A and B in Radial FTP, it sufficient to travels distance b. As shown in the figure 4, in the case Radial FTP nearly two times of L has been decreased from the distance should be traveled in FTP.

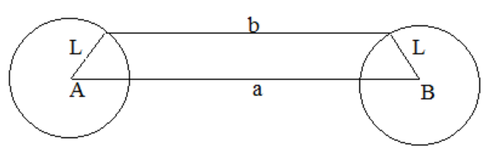

Fig 4: Deference between distances should be traveled in FTP and Radial FTP.

Thus, using triangular inequality we have:

$$a \leq b + 2L$$

Now we can analyze parameter L. A PTAS algorithm has been proposed for FTP in [2]. This algorithm divides the space into pixels of side 1/m. This algorithm rescales the coordinates of n points (robots) so that they lie in a square of unit side. Then the algorithm divides the square into an m-by-m grid of pixels, each of side length 1/m. Then it chooses a representative point for each pixel and assumes that all of robots in the pixel are in the location of the representative point of that pixel. After that it enumerates the wake-up trees on the set of representative points and selects the pseudo-balanced wake-up tree with minimum makespan time. Then the algorithm converts the wake-up tree on the representative points into a wake-up tree on the set of all robots by using an O(1)-approximation [2] in each pixel. In this PTAS algorithm, for awakening each 5 representative points, we should travel a distance equal to one pixel side. See figure 5.

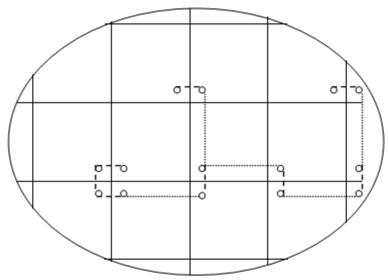

Fig 5: A scheme of awakening process in PTAS algorithm for Euclidean space.

Thus we can have a relation between the makespan time and the number of edges in the longest path in wake-up tree. Let N_h be the number of edges in the longest path in wake-up tree. So we have:

$$N_h = (\text{dotted makespan} *5)/(1/m)$$

in which 1/m is the size of pixel side and dotted makespan is the makespan for FTP. In Radial FTP there is no need to travel all pixel side and it's just need to go to the distance L of the sleeping robot. Thus we can rewrite the above relation. In the following relation we assume that $L < 1/m$ and the minimum distance between each pair of robots is L and radial makespan is the makespan time for Radial FTP:

$$N_h = (\text{radial makespan} *5) / (1/m - L)$$

We know that, using PTAS algorithm for FTP, the solution is a factor of $(1+\varepsilon)$ of the optimal solution. Also the difference between the edge lengths in FTP and Radial FTP is at most 2L. Thus we will have the following relations between the makespan of FTP and Radial FTP and the makespan of FTP corresponding Radial FTP. The dotted makespan corresponding Radial FTP is a solution of Radial FTP that we convert to the FTP.

$$\begin{aligned}
\text{dotted makespan} &\leq (1+\varepsilon)\,(\text{dotted makespan corresponding optimal radial makespan}) \\
&\leq (1+\varepsilon)\,(\text{optimal radial makespan} + 2LN_h) \\
&\leq (1+\varepsilon)\,(\text{optimal radial makespan} + 2L\,((\text{radial makespan}*5) / (1/m - L))) \\
&\leq (1+\varepsilon)\,(1+ 10L / (1/m - L))(\text{ optimal radial makespan})
\end{aligned}$$

In the above phrases $\varepsilon=O(1/m)$. In the last sentence if L=0, the solution will be the solution for FTP.

Thus there is an approximation algorithm for Radial FTP. The assumption of minimum distance L between each two robots leads to the solution be valid for Radial FTP because no two robots wake up in a moment. We can rewrite the following relation for approximation factor:

$$\text{dotted makespan} \leq (1+\varepsilon)\ (1+ 10L/(1/m - L))(\text{ optimal radial makespan})$$

If the parameter L be small enough, the Radial FTP get closer to the FTP and we can use the algorithms proposed for FTP for solving Radial FTP. But if the parameter L be large we should consider special algorithms for Radial FTP.

4. Conclusions

In this paper we proposed an approximation algorithm for Radial FTP. This algorithm is based on a PTAS algorithm. Our algorithm is useful when the L is small and the Radial FTP is closed to the FTP. Otherwise the factor of the algorithm may be not enough good and special algorithms for Radial FTP should be considered.

References

Arkin, E. M. Bender, M. A. Ge, D. He, S. & Mitchell, J. S. B. (2003). Improved approximation algorithms for the freeze-tag problem. *In Proc. 15th ACM Symposium on Parallel Algorithms and Architectures*. California, USA. 295-303.

Arkin, E. M. Bender, M. A. Fekete, S. P. Mitchell J. S. B. & Skutella, M. (2006). The Freeze-Tag Problem: How to Wake Up a Swarm of Robots. *Algorithmica*. 46(2), 193-221.

Bar-Noy, A. Guha, S. Naor, J. & Schieber, B. (1998). Multicasting in heterogeneous networks. *In Proceedings of the 30th Annual ACM Symposium on Theory of Computing (STOC)*. 448–453.

Bucatanschi, D. G. (2004). The Ant Colony System for the Freeze-Tag Problem. *Midstates Conference on Undergraduate Research in Mathematics and Computer Science*. MCURCSM, Ohio, USA. 61-69.

Bucantanschi, D. Hoffmann, B. Hutson K. R. & Kretchmar, R. M. (2007). A Neighborhood Search Technique for the Freeze Tag Problem. *Extending the Horizons: Advances in Computing, Optimization, and Decision Technologies*. 37, 97-113.

Hammar, M. Nilsson B. J. & Persson, M. (2006). The Online Freeze-Tag Problem. *7th Latin American Symposium: Theoretical Informatics*. 3887, 569-579.

Hedetniemi, S. M. Hedetniemi, S. T. & Liestman, A. (1988). A survey of gossiping and broadcasting in communication networks. *Networks*. 18, 319-349.

Ravi, R. (1994). Rapid rumor ramification: approximating the minimum broadcast time. *In Proceedings of the* 35*th Annual Symposium on Foundations of Computer Science* (*FOCS*). 202–213.

Sztainberg, M. O. Arkin, E. M. Bender M. A. & Mitchell, J. S. B. (2002). Analysis of Heuristics for the Freeze-Tag Problem. *Proceeding of the 8th Scandinavian Workshop on Algorithm Theory*. 270-279.

Sztainberg, M. O. Arkin, E. M. Bender M. A. & Mitchell, J. S. B. (2004). Theoretical and Experimental Analysis of Heuristics for the Freeze-Tag Robot Awakening Problem. *IEEE Transactions on Robotics and Automation*. 20(4), 691-701.

The Open Problems Project. URL: http://maven.smith.edu/~orourke/TOPP/P35.html#Problem.35.

Zoraida, B. S. E. Arock, M. Ronald, B. S. M. & Ponalagusamy, R. (2010). DNA algorithm employing temperature gradient for Freeze-Tag Problem in swarm robotics. *Transactions of the Institute of Measurement and Control*.