Abstract

Recently, robotics research has been gained a lot of attentions, due to its wide applications, especially in some places that human beings can not survive, like in a planet or fire. There are a lot of interesting applications of multiple robots, such as satellite exploration and surveillance missions. However, in such system one can not possibly rely on assuming fail-proof software or hardware, especially when such robot systems are expected to operate in hazardous or harsh environment. Therefore, the issue of resilience to failure becomes essential. Perhaps surprisingly, this aspect of multiple robot systems has been explored to very little extent so far.

Flocking, as one of important applications of coordination of multiple robots, has a lot of applications, like moving a box from one place to another place, or explores some unknown places. During flocking, mobile robots group to form a desired pattern and move together while maintaining that formation. One difficulty is how to distinguish the crashed robots from those who are staying in "waiting" state since they all don't move during some period. Another difficulty is to make a group of robots move together to form a geographic graph in distributed way. To address the above questions, our research goal is to achieve the effective coordinated flocking even with the crash of mobile robots. More specifically, we focus on solving distributed geographic agreement problems for groups of mobile robots.

First, we proposed a fault tolerant flocking in asynchronous model. Our algorithm ensures that the crash of faulty robots does not bring the formation to a permanent stop, and that the correct robots are thus eventually allowed to reorganize and continue moving together. Furthermore, the algorithm makes no assumption on the relative speeds at which the robots can move. The algorithm relies on the assumption that robots' activations follow a k-bounded asynchronous scheduler, in the sense that the beginning and end of activations are not synchronized across robots (asynchronous), and that while the slowest robot is activated once, the fastest robot is activated at most k times (k-bounded).

By analyzing the above algorithm carefully, we find one flaw in the algorithm due to the limit of design method. That is the formation formed by robots can not be rotated freely. Therefore, in the following part, we design a new method to lift such limit by allowing the formation to move to any direction, including its rotation, yet in a semi-synchronous model. Further analysis demonstrates that the proposed algorithm can make formation freely rotation, and has very good maneuverability.

In practical applications, some parts of a robot, like sensor, moving actuator, or memory etc., is prone to transient crash due to the influence of complex environment. We discuss the (im) possibility of self-stabilization of flocking algorithms with memory corruption in different system models.

Finally, we propose a non-fault tolerant flocking algorithm in order to compare the performance with the above fault-tolerant ones. The described algorithm can effectively adapt to the environment to avoid the collision among robots and obstacles. Furthermore, one interesting thing is when there is no obstacle in the environment; the robots can keep the desired distance with their neighbors.

In all, to the best of our knowledge, our work is the first to consider the fault tolerant issue during robots dynamic flocking. Also it opens some new interesting topics on robots dynamic coordination, like robots coordination in the model of crash and recovery.

Keywords: Mobile robot, decentralized coordination, local interactions, flocking, formation generation, self-organization