

Control algorithm for teams of robots factors in moving obstacles

21 April 2016, by Larry Hardesty



MIT researchers will present a new, decentralized planning algorithm for teams of robots that factors in not only stationary obstacles, but moving obstacles, as well. The algorithm still comes with strong mathematical guarantees that the robots will avoid collisions. Credit: Christine Daniloff/MIT

Planning algorithms for teams of robots fall into two categories: centralized algorithms, in which a single computer makes decisions for the whole team, and decentralized algorithms, in which each robot makes its own decisions based on local observations.

With centralized algorithms, if the central computer goes offline, the whole system falls apart. Decentralized algorithms handle erratic communication better, but they're harder to design, because each [robot](#) is essentially guessing what the others will do. Most research on decentralized algorithms has focused on making collective decision-making more reliable and has deferred the problem of avoiding obstacles in the robots' environment.

At the International Conference on Robotics and Automation in May, MIT researchers will present a

new, decentralized planning algorithm for teams of robots that factors in not only stationary obstacles, but moving obstacles, as well. The algorithm also requires significantly less communications bandwidth than existing decentralized algorithms, but preserves strong mathematical guarantees that the robots will avoid collisions.

In simulations involving squadrons of minihelicopters, the decentralized algorithm came up with the same flight plans that a centralized version did. The drones generally preserved an approximation of their preferred formation, a square at a fixed altitude—although to accommodate obstacles the square rotated and the distances between drones contracted. Occasionally, however, the drones would fly single file or assume a formation in which pairs of them flew at different altitudes.

"It's a really exciting result because it combines so many challenging goals," says Daniela Rus, the Andrew and Erna Viterbi Professor in MIT's Department of Electrical Engineering and Computer Science and director of the Computer Science and Artificial Intelligence Laboratory, whose group developed the new algorithm. "Your group of robots has a local goal, which is to stay in formation, and a global goal, which is where they want to go or the trajectory along which you want them to move. And you allow them to operate in a world with static obstacles but also unexpected dynamic obstacles, and you have a guarantee that they are going to retain their local and global objectives. They will have to make some deviations, but those deviations are minimal."

Rus is joined on the paper by first author Javier Alonso-Mora, a postdoc in Rus' group; Mac Schwager, an assistant professor of aeronautics and astronautics at Stanford University who worked with Rus as an MIT PhD student in mechanical engineering; and Eduardo Montijano, a professor at Centro Universitario de la Defensa in Zaragoza,

Spain.

Trading regions

In a typical decentralized group planning algorithm, each robot might broadcast its observations of the environment to its teammates, and all the robots would then execute the same planning algorithm, presumably on the basis of the same information.

But Rus, Alonso-Mora, and their colleagues found a way to reduce both the computational and communication burdens imposed by consensual planning. The essential idea is that each robot, on the basis of its own observations, maps out an obstacle-free region in its immediate environment and passes that map only to its nearest neighbors. When a robot receives a map from a neighbor, it calculates the intersection of that map with its own and passes that on.

This keeps down both the size of the robots' communications—describing the intersection of 100 maps requires no more data than describing the intersection of two—and their number, because each robot communicates only with its neighbors. Nonetheless, each robot ends up with a map that reflects all of the obstacles detected by all the team members.

Four dimensions

The maps have not three dimensions, however, but four—the fourth being time. This is how the algorithm accounts for moving obstacles. The four-dimensional map describes how a three-dimensional map would have to change to accommodate the obstacle's change of location, over a span of a few seconds. But it does so in a mathematically compact manner.

The algorithm does assume that moving obstacles have constant velocity, which will not always be the case in the real world. But each robot updates its map several times a second, a short enough span of time that the velocity of an accelerating object is unlikely to change dramatically.

On the basis of its latest map, each robot calculates the trajectory that will maximize both its local

goal—staying in formation—and its global goal.

The researchers are also testing a version of their algorithm on wheeled robots whose goal is to collectively carry an object across a room where human beings are also moving around, as a simulation of an environment in which humans and robots work together.

This story is republished courtesy of MIT News (web.mit.edu/newsoffice/), a popular site that covers news about MIT research, innovation and teaching.

Provided by Massachusetts Institute of Technology

APA citation: Control algorithm for teams of robots factors in moving obstacles (2016, April 21) retrieved 16 August 2022 from <https://techxplore.com/news/2016-04-algorithm-teams-robots-factors-obstacles.html>

This document is subject to copyright. Apart from any fair dealing for the purpose of private study or research, no part may be reproduced without the written permission. The content is provided for information purposes only.