

CS 69.13, Fall 2022, Elliot Potter

Method Description

This system is an extension on the leader-follower programming assignment. The major changes are: 1. The waypoint broadcaster module now broadcasts a list of possible robot locations, as well as a list of terminal nodes that the robot must cover. I relax the assumption that the total number of nodes in the graph is equal to the number of available robots, and instead I assume that the number of robots is sufficient to cover the minimum Steiner graph of those terminal nodes. 2. The leading robot calls two functions: `get_graph()` and `solve_steiner_tree()`. - `get_graph` converts the given waypoints into a connected NetworkX graph. - `solve_steiner_tree()` calls the NetworkX tree solver with the graph and terminal nodes

You can configure the connectivity distance in the launch file, and the evaluated graph in `waypoint_broadcaster`

Evaluation

This system worked pretty well. Given a large graph, the Steiner optimizer succeeds in finding a much smaller graph that maintains connectivity between nodes.