Ming Hsieh Department of Electrical Engineering

USC Viterbi School of Engineering

RiverSwarm: Topology-Aware Distributed Planning for Obstacle Encirclement in Connected Robotic Swarms

Pradipta Ghosh, Jie Gao, Andrea Gasparri and Bhaskar Krishnamachari

ANRG

Problem Introduction

Scenario: Collective behavior of a robotic swarm moving in a cluttered environment.

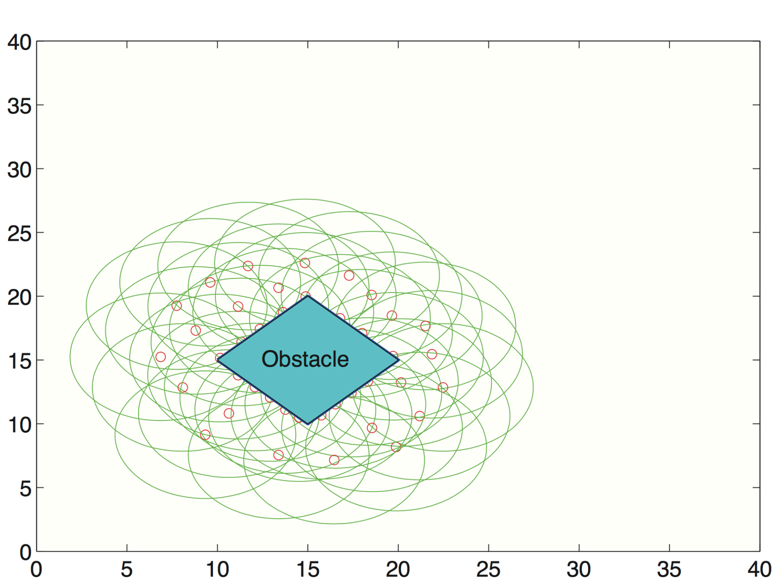
Challenges:

- *Disconnection:* network *partitioning*
- Inefficiency in movement due to connectivity constraints

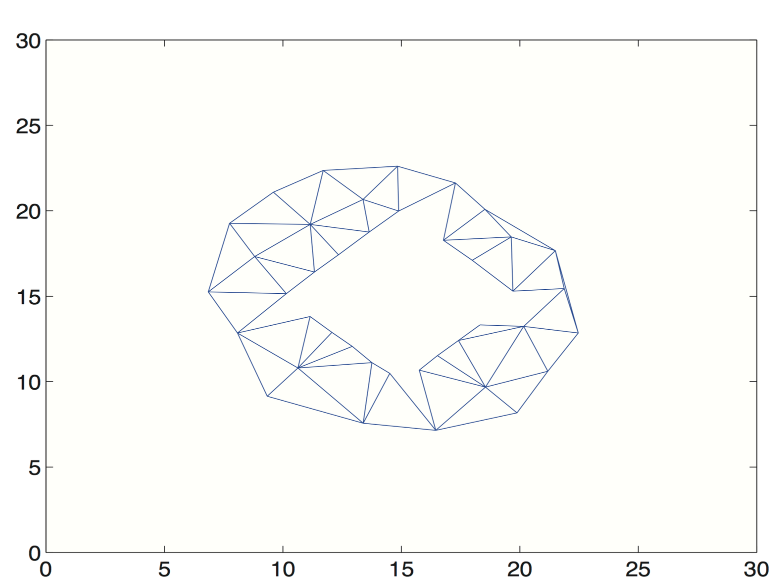
Proposed Solution

- Run a *global connectivity* maintenance algorithm
- Use the Gauss Bonnet theorem based technique to detect if the swarm has *encircled an obstacle*
- If needed *disconnect* the swarm from the *rear of the obstacle*

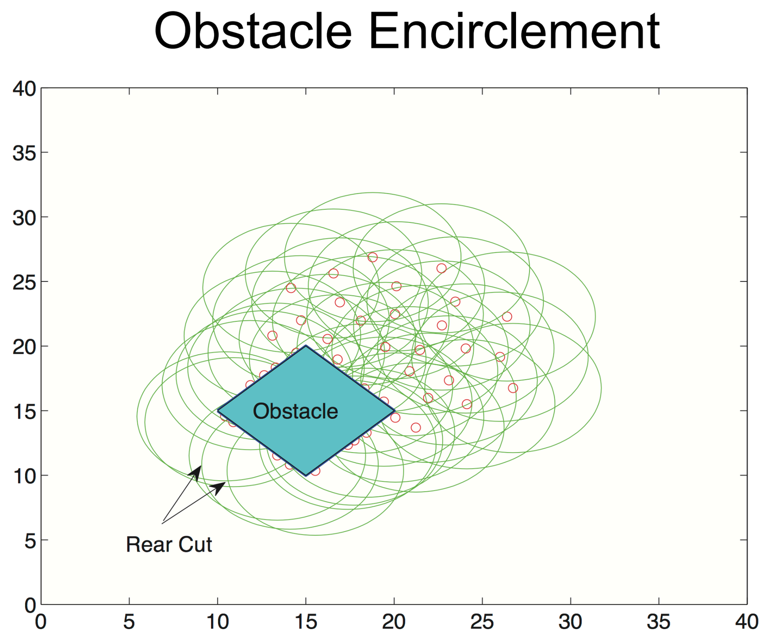
Algorithm Steps



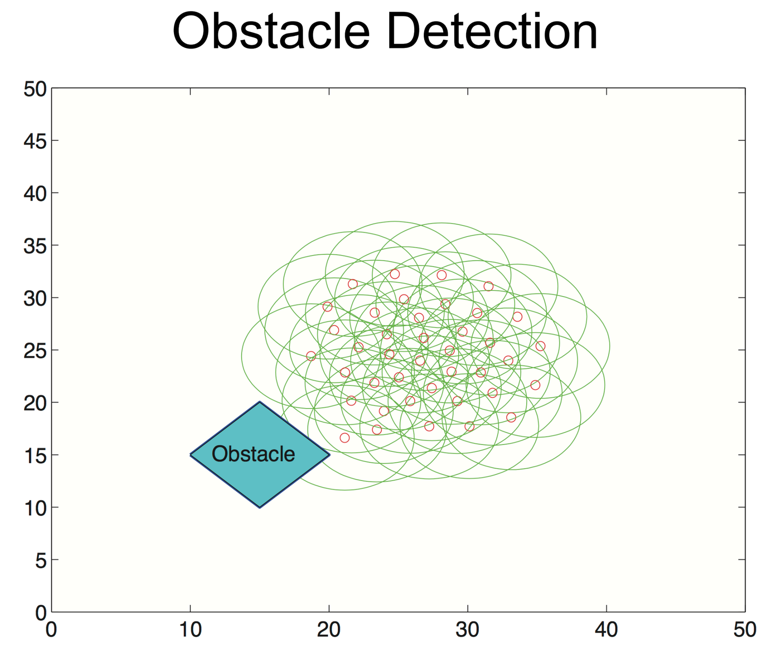
Obstacle Encirclement



Obstacle Detection



Initiate Detach



Obstacle Passed

Global Connectivity Maintenance

- Describe the network topology through the Laplacian matrix
- Relate the gradient of the algebraic connectivity to the robots locations
- Design a control law to maximize this gradient
- It can be computed in a distributed fashion

The Gauss Bonnet Theorem

The *total curvature* of a surface M is a *topological invariance*:

$$C(M) = 2\pi\chi(M)$$

where:

$C(M)$ is the Gaussian Curvature of the surface

$\chi(M)$ is the Euler Characteristics of the surface

Obstacle/Hole Detection

- Perform *Delaunay Triangulation* of the network topology
- Now, the Euler characteristic is:

$$\chi(M) = 2 - Z$$

with Z number of holes in the network

- The Gaussian curvature of the Delaunay Triangulation is:

$$C(M) = \sum_{v_i \varepsilon V} C(i)$$

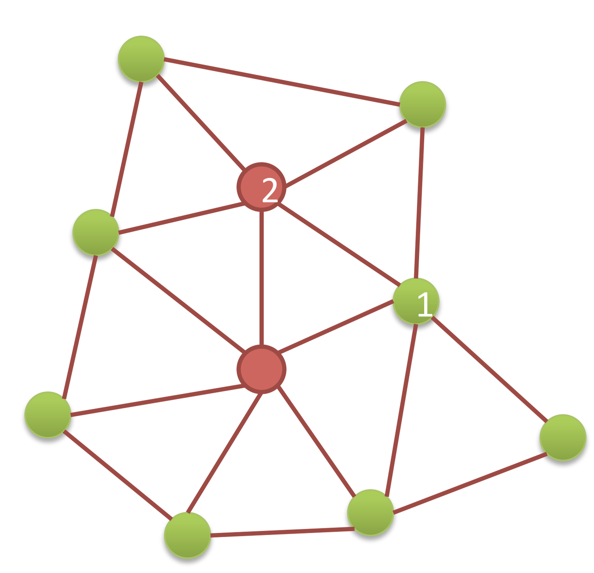
where $C(i)$ is the Gaussian Curvature at vertex *i*

- **Combining them**

$$\sum_{v_i \varepsilon V} C(i) = 2\pi(2 - Z)$$

In our case:

- The Gaussian Curvature $C(i)$ at vertex *i* is computed as:



Boundary Vertex

$$C(i) = \pi - \sum_k \theta_i(k)$$

Interior Vertex

$$C(i) = 2\pi - \sum_k \theta_i(k)$$

With $\theta_i(k)$ the corner angle of the k[th] triangle to vertex i is adjacent to

$$C(1) = \pi - \sum_{k=1}^{4} \theta_1(k) = \pi - 4\times\frac{\pi}{3} = (-\frac{\pi}{3})$$

$$C(2) = 2\pi - \sum_{k=1}^{5} \theta_2(k) = 2\pi - 5\times\frac{\pi}{3} = \frac{\pi}{3}$$

Future Works

- Design a distributed control law with provable theoretical guarantees
- Validation on a real test-bed scenario

pradiptg@usc.edu, jgao@cs.stonybrook.edu, gasparri@dia.uniroma3.it, bkrishna@usc.edu

Ming Hsieh Institute
Ming Hsieh Department of Electrical Engineering