Path Planning for Swarms by Combining Probabilistic Roadmaps and Potential Fields

Alex Wallar
School of Computer Science
University of St Andrews
St Andrews, Scotland

Erion Plaku
Dept. of Electrical Engineering
and Computer Science
Catholic University of America
Washington DC, USA

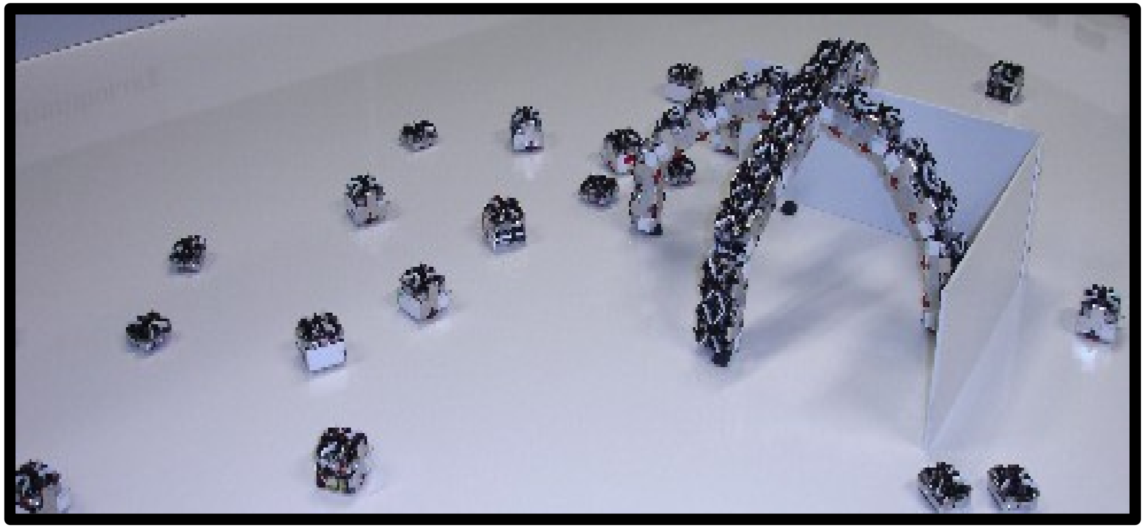
http://aw204.host.cs.st-andrews.ac.uk

University of St Andrews

CUA

Motivation

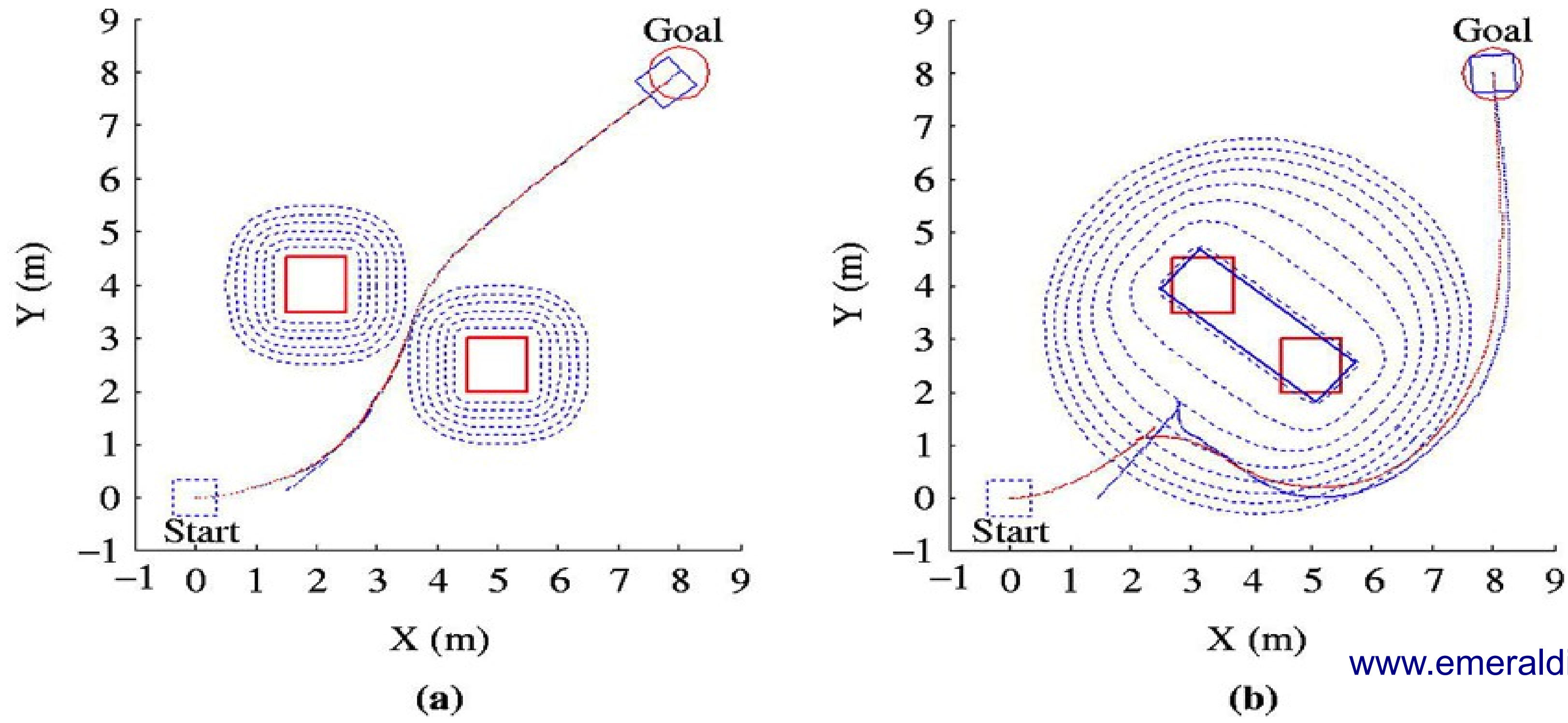
- Enables a large group of robots to complete complex tasks through simple interactions
- Used for
  - Monitoring
  - Mapping
  - Inspection
  - Surveying
  - Exploration
  - Search and Rescue



www.eggshell-robotics.com

Path Planning for Swarms via Potential Fields

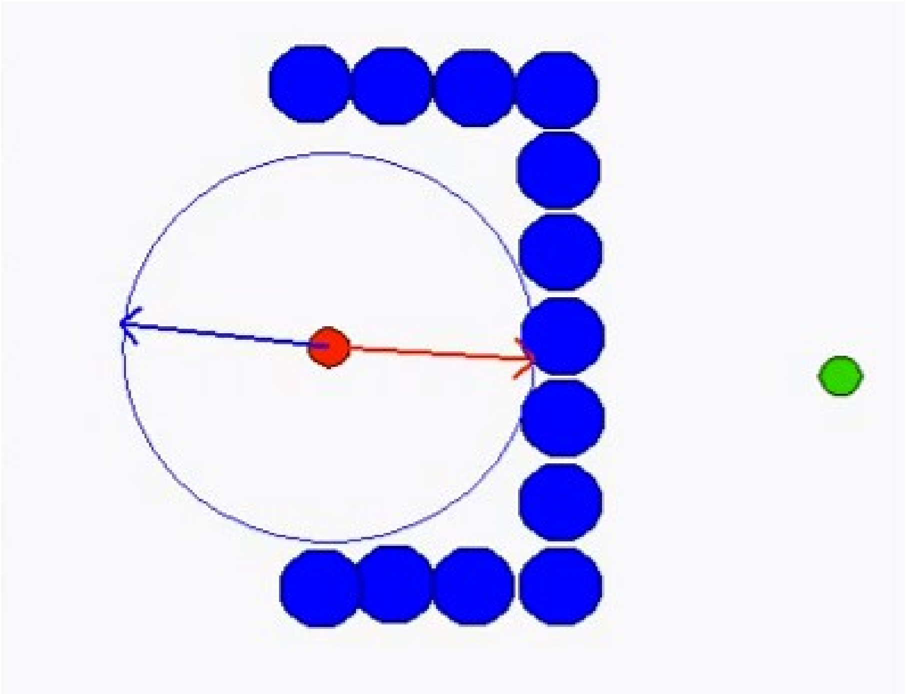
- Attractive potential field around the goal
- Repulsive potential field around obstacles
- Very fast field computation



www.emeraldinsight.com

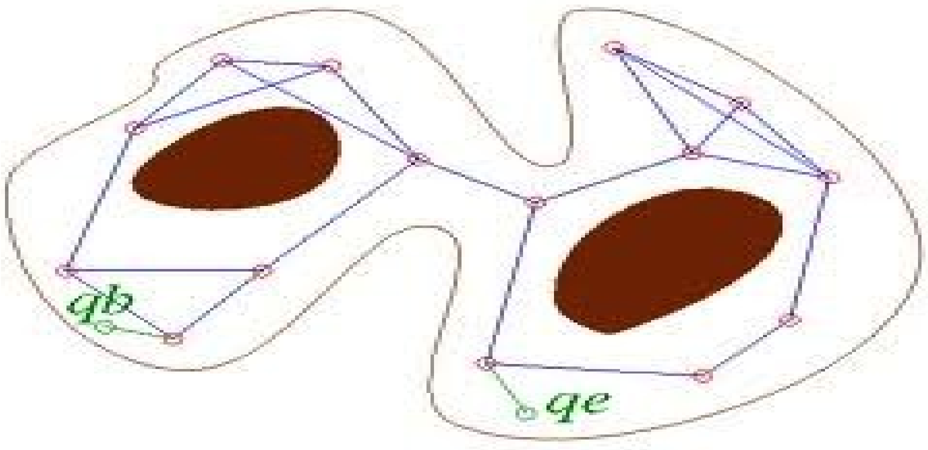
Limitations of Path Planning via Potential Fields

- Local minima problem
- Potential fields often lead robots into local minima especially when dealing with complex environments and large swarms



Path Planning for Swarms via Probabilistic Roadmaps (PRMs)

- Randomly distribute points in the environment
- Connect collision free points
- Use a shortest path algorithm to determine the path from initial to final configurations
- Captures the connectivity of the environment



robotics.stanford.edu

Limitations of Path Planning via PRMs

- As the number of robots increases
  - It becomes difficult to sample collision-free configurations
  - It becomes difficult to generate collision free paths
  - Larger roadmaps are needed to capture the connectivity of the sample space
- PRMs do not promote fluidity in movement

Proposed Approach

- Combine roadmaps and potentials for swarms (CRoPS) to enable fast path planning for swarms
- Potential fields provide scalability
- Probabilistic roadmaps provide high-level guidance of how swarm should move towards the goal
- Bots will move in cohesion due to potential fields

Swarm Motion

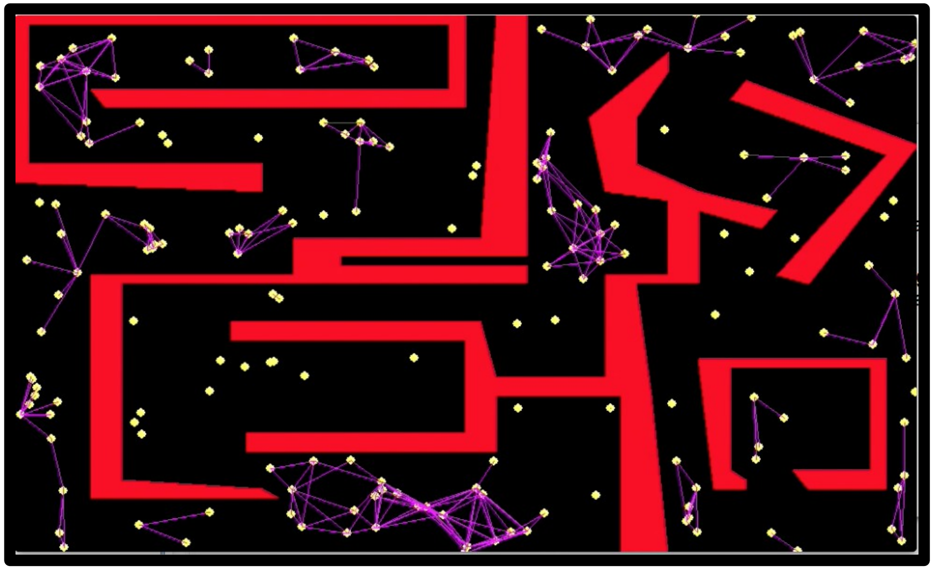
- There is long range attraction to intermediate goals and final destination
- Robots are repulsed from obstacles
- Robots move as a swarm while keeping some separation from one another
- A robot’s heading is influenced by the headings of its neighbors

CRoPS: Combined Roadmaps and Potential Fields

- Construct roadmap using PRM
- Search roadmap to compute intermediate goals for each the swarm
- Repeat until solved
  - assign next goal to boids that have reached current goal
  - compute overall potential for each boid
  - compute new heading based on overall potential
  - update positions

Roadmap Construction

- Nodes are randomly distributed in environment
- Nodes are connected to k collision free nodes within a radius
- Nodes that spawn within a minimum distance from an obstacle are discarded



Roadmap Weights

- In order to bias the swarm towards less cluttered areas, nodes further away from obstacles are assigned higher weights

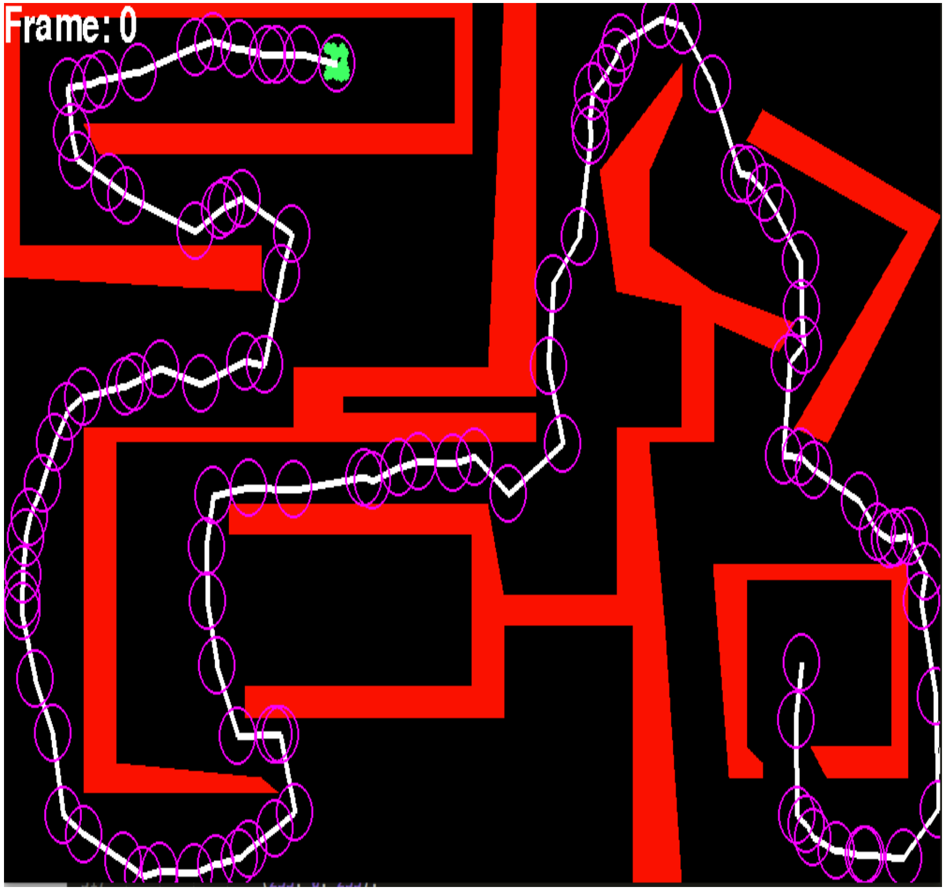
$$w(q_i) = (\sum_{o \in Obstacles} dist(q_i, o))^3$$

- Edge costs reflect not only the distance among the endpoints, but also their clearance away from obstacles, i.e.,

$$w(q_i, q_j) = ||q_i, q_j||_2 / \min(w(q_i), w(q_j))$$

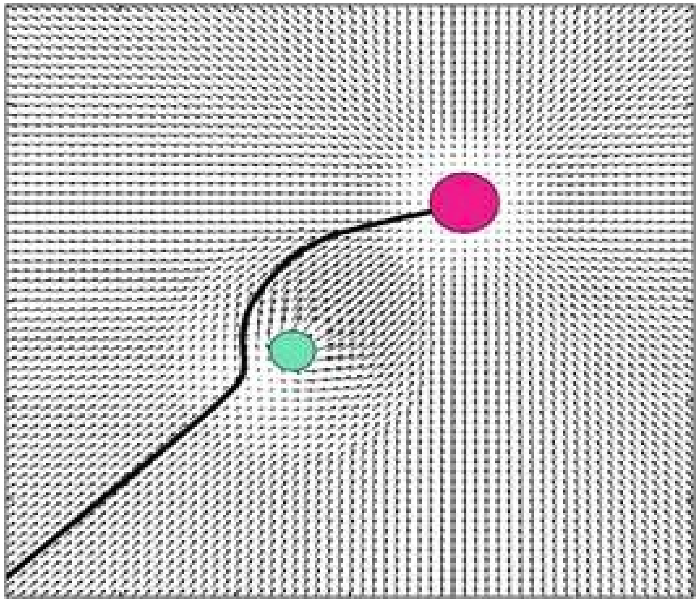
Guiding the Swarm through the Shortest Roadmap Path

- Shortest roadmap path serves as high-level guide
- Computed using Dijkstra's algorithm
- Guide provides a series of intermediate goals defined as circles centered at the path vertices
- CRoPS seeks to move the swarm to the final destination by passing each bot through these intermediate goals



Potential Fields

- **CRoPS** uses five potential fields
  - Repulsion from obstacles
  - Repulsion from other robots
  - Attraction to current intermediate goal
  - Neighbor heading influence
  - Random walks



www.cs.mcgill.ca

Repulsion From Obstacles

- A strong potential field is used to repel bots in the swarm from obstacles

$$P_{obst}(b,o) = \frac{1}{(dist(pos(b),o) - radius(b))^2}$$

- The repulsion is only computed for obstacles within a certain distance from the bot

$$PF_{obst}(b) = \sum_{\substack{o \in Obstacles \\ dist(b,o) \leq \Delta_{obst}}} (pos(b) - ClosestPoint(o, pos(b)))P_{obst}(b,o)$$

Repulsion From Other Robots

- Robots should not come too close or too far from each other
- Uses weak sigmoidal potential function to show that obstacle field is dominant

$$P_{sep}(b_i, b_j) = \frac{1}{1 + \exp(\delta_{sep} ||pos(b_i), pos(b_j)||_2)}$$

- Similar to the obstacles, robots that are far away should not influence this field.

$$PF_{sep}(b) = \sum_{\substack{b_i \in Robots - \{b\} \\ dist(b, b_i) \leq \Delta_{sep}}} (pos(b) - pos(b_i)) P_{sep}(b, b_i)$$

Attraction to Immediate Goal

- $igoal(b)$ represents the next immediate goal defined in the a shortest path for a boid b.
- A weak sigmoidal function is used to increase potential as the robot gets closer to the goal
- This allows the swarm to increase speed and promotes fluidity

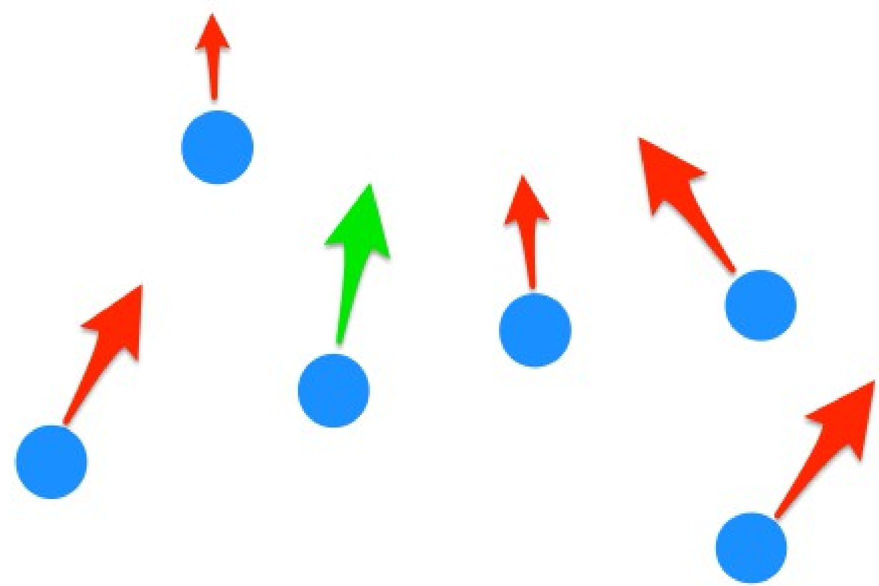
$$PF_{igoal}(b) = \frac{igoal(b) - pos(b)}{1 + \exp(\delta_{igoal} ||pos(b), igoal(b)||_2)}$$

Neighbor Heading Influence

- The heading of a robot is influenced by the headings of its neighbors
- The neighbors are chosen to be not too far nor too close
- Moreover, bots that are “stuck” are unable to be chosen

$$\gamma(b, b_i) = \exp\left(\frac{-(||pos(b), pos(b_i)||_2 - \mu)^2}{2\sigma^2}\right)$$

$$PF_{heading}(b) = \sum_{b_i \in Neighs(b)} heading(b_i)$$



Escaping Local Minima

- Each robot keeps track of past its past locations
- Bot is considered stuck if it has moved very little in the last l time steps

$$stuck(b) = \begin{cases} 1, & \text{if } ||pos(b) - prev_{\ell}(b)||_2 < \Delta_{stuck} \\ 0, & \text{otherwise,} \end{cases}$$

- If the bot is stuck, a random walk in the form of another potential field is applied

$$PF_{\text{escape}}(b) = stuck(b)(r_x, r_y)$$

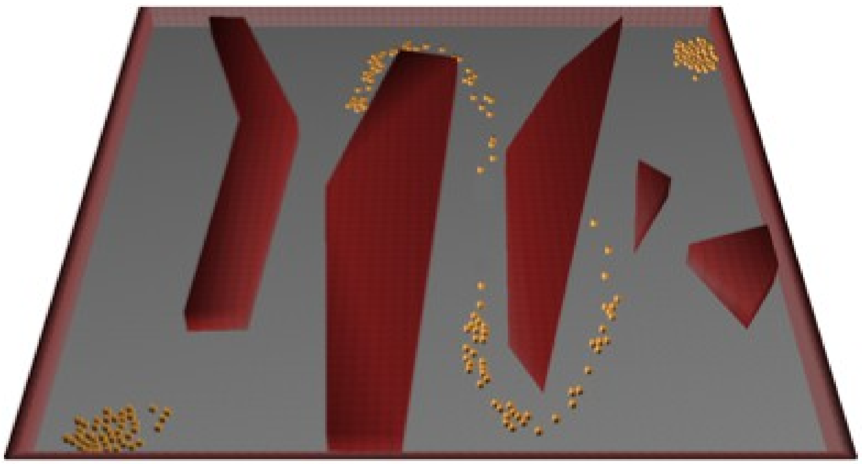
Superimposition of Potential Fields

- Different potential fields are superimposed to obtain the overall force vector applied to the robot
- This superimposition ensures that the subfield with the highest potential has the most influence

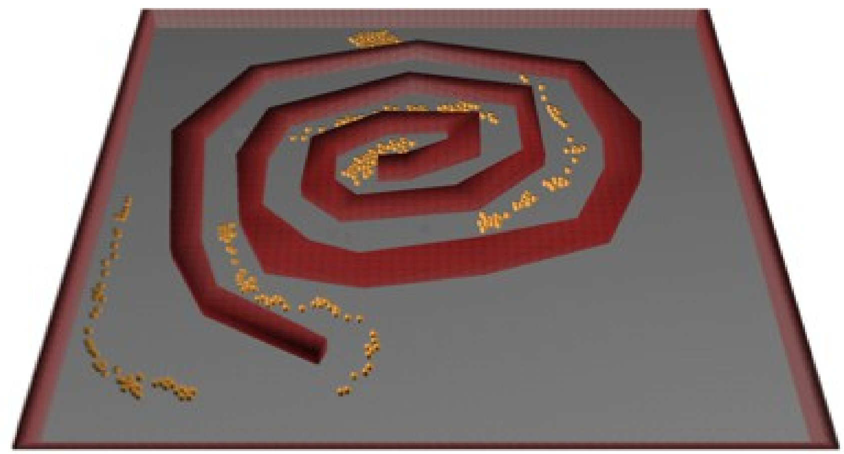
$$PF(b) = \frac{\sum_{\phi \in fields} (||PF_\phi(b)||_2 PF_\phi(b))}{\sum_{\phi \in fields} ||PF_\phi(b)||_2}$$

$$fields = \{obst, sep, igoal, heading, escape\}$$

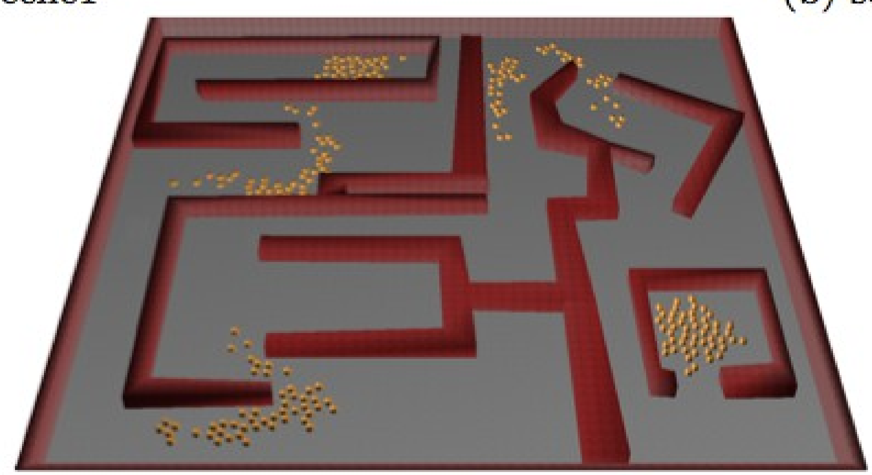
Experiments and Results



(a) scene1



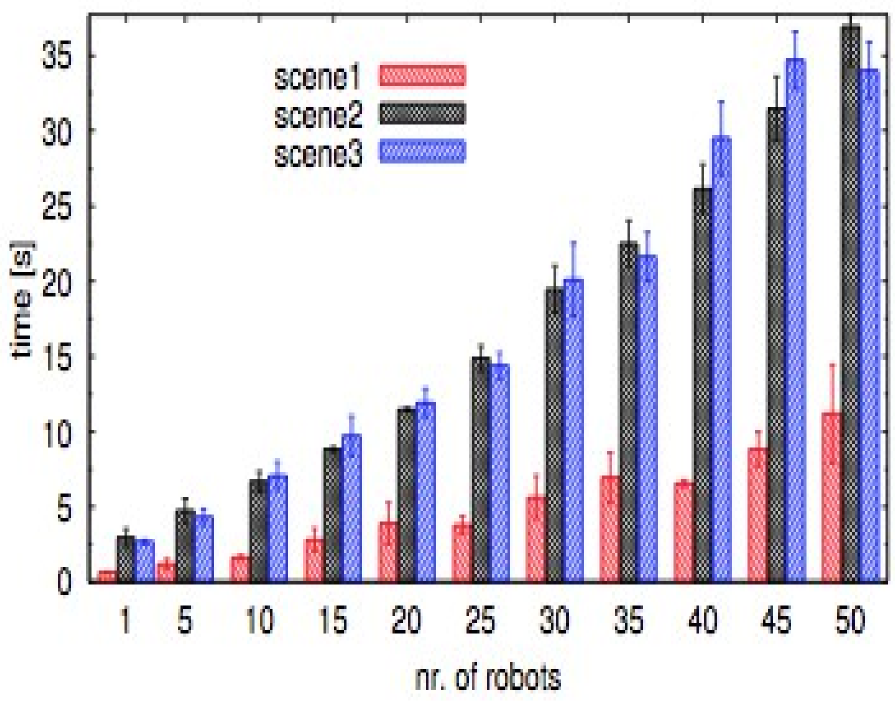
(b) scene2

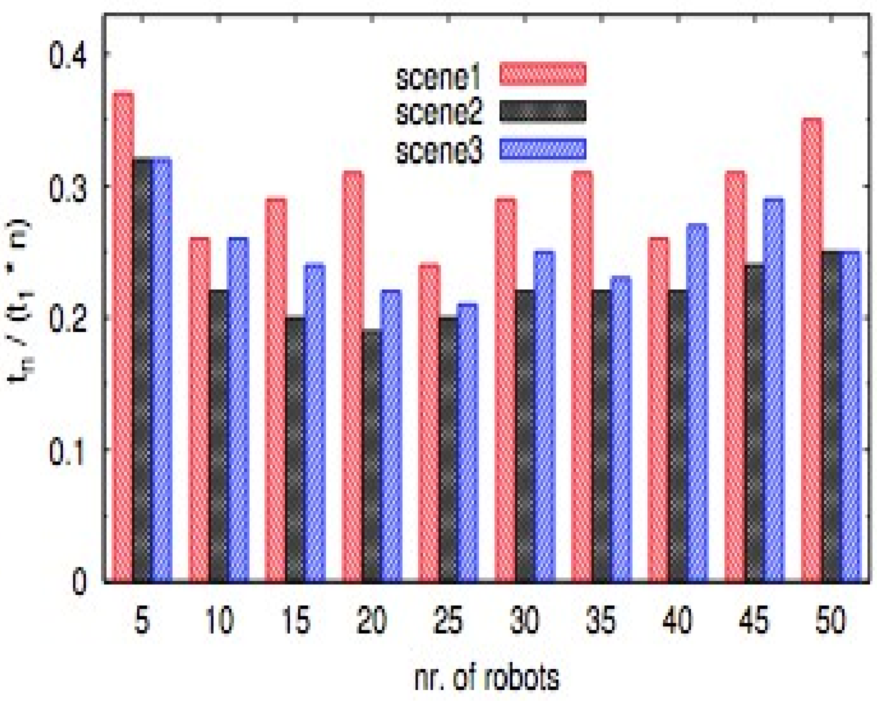


(c) scene3

Time Analysis

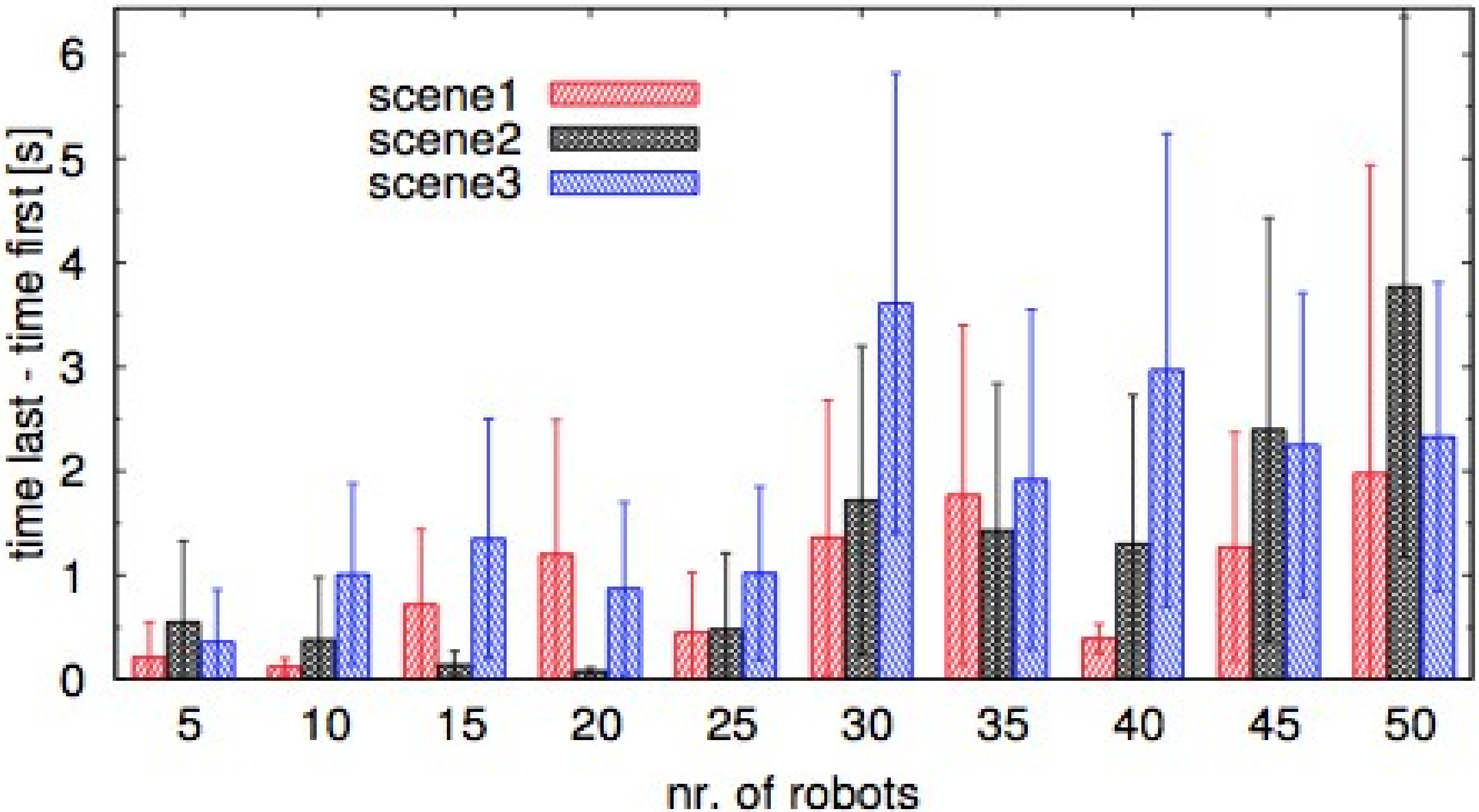
- Graph 1 & 2 show that as the number of robots increase, the time needed to navigate through each environment increases linearly





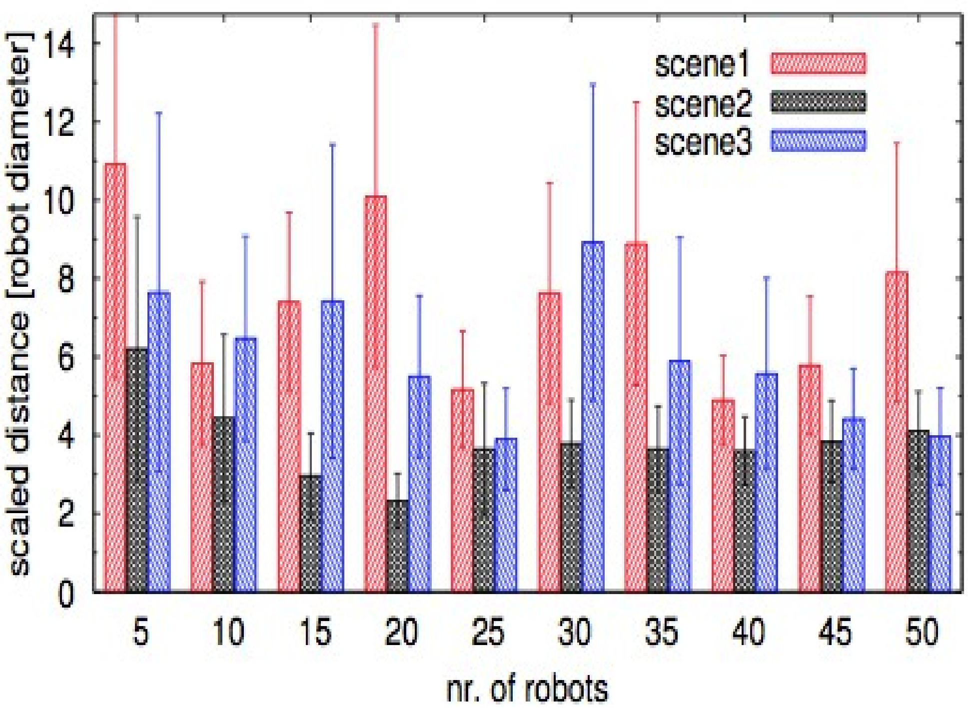
Swarm Analysis 1

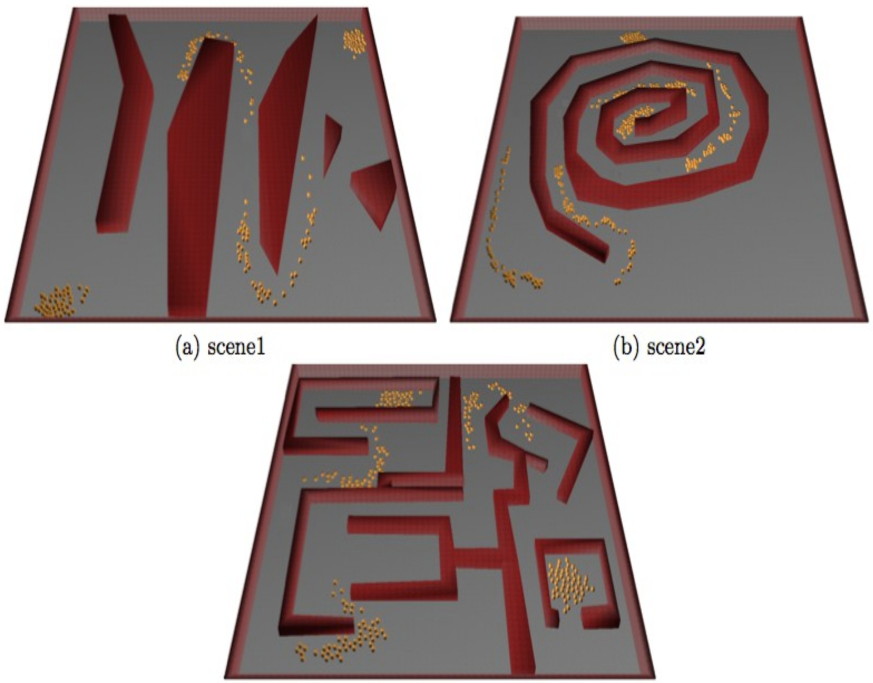
- Figure shows that robots reach the last goal at generally the same time on different environments



Swarm Analysis 2

- Varied levels of separation based on the environment
- Separation generally constant regardless of number of bots





(a) scene1

(b) scene2

(c) scene3

Research Direction

- Incorporate moving obstacles
- Improve the interplay between the global path planner and the potential fields
- Deal with probabilistic environments
- Create threat maps based on amount of movements in an area
- Increase the dimensions of the environment

Conclusion

- The combination between potential fields and low dimensional roadmaps enable fast swarm planning
- **CRoPS** presents an efficient, computationally cheap algorithm for swarm path planning
- **CRoPS** is scalable due to the linear time complexity
- Bots still act as a swarm, obeying the four base principles

Questions