Learning the Delaunay Triangulation of Landmarks From a Distance Ordering Sensor

Max Katsev    Steven M. LaValle

Department of Computer Science
University of Illinois at Urbana-Champaign, USA

IROS 2011
Problem

Simple point robot in a plane:
- 1 weak sensor
- 2 simple motion primitives
- No GPS, no odometry, no compass, etc.

Problem:
- What can the robot learn?
- What can the robot do?
Robot Model

- Planar environment with fixed distinguishable landmarks
- Distance ordering sensor
  - Returns a sequence of landmark labels in the order of ascending distance from the robot position
- Motion primitives:
  - TOWARD($l$)
  - AWAY($l$)
Motivation

- Analyze capabilities of a low-cost minimalistic robot
- Avoid state estimation
- Long-term goal: compare the power of sensors
Motivation

- Analyze capabilities of a low-cost minimalistic robot
- Avoid state estimation
- Long-term goal: compare the power of sensors
Related Work

- **Bug algorithms**

- **Topological mapping**

- **Landmark order-based navigation**
  - B. Tovar et al., 2006, 2007
Robot Model

- Planar environment with fixed distinguishable landmarks
- Distance ordering sensor
  - Returns a sequence of landmark labels in the order of ascending distance from the robot position
- Motion primitives:
  - $\text{TOWARD}(l)$
  - $\text{AWAY}(l)$
Robot Capabilities

- Identify the type of a triangle formed by 3 landmarks
- Locate its circumcenter
- Construct the Delaunay triangulation of a set of landmarks
- Compute the convex hull
- Patrolling
Identifying the Triangle Type
Identifying the Triangle Type

TOWARD(1)

TOWARD(1)
Identifying the Triangle Type

TOWARD(2) is executed until 1 and 2 switch places in the sensor output
Identifying the Triangle Type

Type of the angle can be determined from the position of the corresponding landmark in the distance ordering.
Locating the Circumcenter

- Construct a directional vector field guiding the robot to the goal
- Specific field depends on the type of a triangle
- Easy for acute and right triangles:
  - Always move toward the most distant landmark (last in the distance ordering)
  - In other words, follow the antigradient of
    \[ f(r) = \max_{i \in \{1, 2, 3\}} \rho(r, l_i) \]
Locating the Circumcenter

- Construct a directional vector field guiding the robot to the goal
- Specific field depends on the type of a triangle
- Easy for acute and right triangles:
  - Always move toward the most distant landmark (last in the distance ordering)
  - In other words, follow the antigradient of $f(r) = \max_{i \in \{1,2,3\}} \rho(r, l_i)$
For an obtuse triangle case some modifications are required
Use $\text{AWAY}(2)$ to avoid getting stuck between landmarks 1 and 3
Extra step: navigate to the allowed starting position first (for example, interior of the triangle)
Computing the Delaunay Triangulation

- Delaunay triangulation of set of points $L$ is formed by triangles with an empty circle property.
- Circumscribed circle of any triangle should not contain points of $L$ in its interior.

Delaunay

1

2

3

4

not Delaunay

1

2

3

4
Computing the Delaunay Triangulation

- Delaunay triangulation of set of points $L$ is formed by triangles with an empty circle property
- Circumscribed circle of any triangle should not contain points of $L$ in its interior

![Delaunay Diagram](image1)

![Not Delaunay Diagram](image2)
Computing the Delaunay Triangulation

- Delaunay triangulation of set of points $L$ is formed by triangles with an empty circle property.
- Circumscribed circle of any triangle should not contain points of $L$ in its interior.

\[
[(1, 3, 4), 2]
\]

\[
[(4, (1, 2, 3)]
\]
Computing the Convex Hull

- A triangulation of a set of landmarks can be used to infer its convex hull
- Edges of the hull boundary belong to exactly 1 triangle
- Collected information allows to compute the convex hull for any subset of landmarks
Computing the Convex Hull

- A triangulation of a set of landmarks can be used to infer its convex hull.
- Edges of the hull boundary belong to exactly 1 triangle.
- Collected information allows to compute the convex hull for any subset of landmarks.
Computing the Convex Hull

- A triangulation of a set of landmarks can be used to infer its convex hull
- Edges of the hull boundary belong to exactly 1 triangle
- Collected information allows to compute the convex hull for any subset of landmarks
Demonstration

Triangle (0, 1, 2) ... checking type
Demonstration

Triangle (0, 1, 2) ... checking type
Conclusions

- We consider a robot in a plane equipped with a landmark distance ordering sensor
- The robot is able to compute the Delaunay triangulation for a set of landmarks
  - can be used, for example, to patrol any subset of landmarks

Future work:

- What other tasks can the robot perform?
- Consider sensing/actuation uncertainty
- Consider a more general comparison function
- Compare with different minimalistic sensor models
Conclusions

- We consider a robot in a plane equipped with a landmark distance ordering sensor.
- The robot is able to compute the Delaunay triangulation for a set of landmarks.
  - can be used, for example, to patrol any subset of landmarks.

Future work:
- What other tasks can the robot perform?
- Consider sensing/actuation uncertainty.
- Consider a more general comparison function.
- Compare with different minimalistic sensor models.