Multi-Robot Exploration in the Polygonal Domain

Tomáš Juchelka, Miroslav Kulich, Libor Přeučil

Division of Intelligent and Mobile Robotics - (IMR)
Gerstner Laboratory for Intelligent Decision Making and Control
Department of Cybernetics
Czech Technical University in Prague

10/06/2013
Multi-Robot Exploration in the Polygonal Domain

Outline

• Problem definition
• Exploration framework
• Polygonal domain
• Exploration strategies
• Experimental results
• Conclusion
Mobile Robot Exploration

- Create a map of the environment
- **Frontier**-based approach
  - *Yamauchi (1997)*
- Occupancy grid
  - *Moravec and Elfes (1985)*
- Laser scanner sensor
- **Next-best-view** approach
  - *Select the next robot goal*

Performance metric:
**Time to create the map of the whole environment**

*search and rescue mission*
The Algorithm

Exploration is an iterative procedure:

while unexplored areas exist do
    read current sensor information;
    update map with the obtained data;
    determine new goal candidates;
    assign the goals to the robots;
    plan paths for the robots;
    move the robots towards the goals;
end
Polygonal Domain

- The environment represented as a polygon with holes
- Sensor measurement represented as a polygon:
  - Successive Edge Following
  - Ramer-Douglas-Peucker algorithm
  - Least Squares Fit
- Map is incrementally built as a union of measurements
Map Building

- Vatti clipping algorithm
- Clipper library
- Problem: storing information about edges
- Operations: clipping, offsetting (Minkowski sum): output edges post-processing:

\[ P = p_1 + p_2 + |d_1| + |d_2| \]
Comparison Optimization

• Rules applied:
  • The bounds with $y$-coordinate of its local minimum higher than $y$-coordinate of the top vertex of the output edge can be completely skipped.
  • If $y$-coordinate of the bottom vertex of the edge from the bound is higher than the top vertex of the output edge then the rest of the bound can be skipped.
  • If the penalty value is zero then skip further comparison of the output edge.

**Algorithm performance**

- Clipping
- Clipping with the modifications
- Clipping with the modifications and acceleration
Map Management

- All-in-one map is not robust due to uncertainty of sensors
- Separate maps of the free space and obstacles
- Maps are merge before planning
- The number of vertices grows fast \( \leadsto \)
  Ramer-Douglas-Peucker algorithms for reduction
- The map is used for:
  - Determination of goal candidates
  - Evaluation of goal candidates: Visibility graph + several runs of Dijkstra
  - Planning

\(\textit{We use distance cost only.}\)
Multi-Robot Exploration Strategy

- A set of $m$ robots at positions $R = \{r_1, r_2, \ldots, r_m\}$
- A set of $n$ goal candidates $G = \{g_1, g_2, \ldots, g_n\}$
- The exploration strategy (at each planning step):
  
  Select a goal $g \in G$ for each robot $r \in R$ that will minimize the required time to explore the environment

The problem is formulated as the task-allocation problem

\[
(\langle r_1, g_{r_1} \rangle, \ldots, \langle r_m, g_{r_m} \rangle) = \text{assign}(R, G, \mathcal{M}),
\]

where $\mathcal{M}$ is the map
Comparison - Goal Assignment Strategies

1. **Greedy Assignment**
   
   *Yamauchi B, Robotics and Autonomous Systems 29, 1999*
   
   • Randomized greedy selection of the closest goal candidate

2. **Iterative Assignment**
   
   *Werger B, Mataric M, Distributed Autonomous Robotic Systems 4, 2001*
   
   • Centralized variant of the broadcast of local eligibility algorithm (BLE)

3. **Hungarian Assignment**
   
   • Optimal solution of the task-allocation problem for assignment of $n$ goals and $m$ robots in $O(n^3)$

4. **K-means Clustering**
   
Evaluation Methodology

Experimental setup

• 4, 6, 8, 10 robots, 4 goal-assignment strategies, 4 environments, 30 runs
• sensor range: 5 m, FOV: 270°, SND driver
• Planning period: 1 sec
• CPU 4x3.3GHz, 8GB RAM, x86_64 GNU/Linux kubuntu 3.0.0-20, ROS electric
• total number of runs: 1440 (speeded up 3x: 240 hours)

Performance metrics

• $t_{exp}$ - total required exploration time
• $d_{max}$ - maximal distance:

$$L = \max\{l_1, l_2, \ldots, l_m\},$$

where $m$ is the number of robots and $l_i$ the traveled distance.
Experiments

- a portion of hospital map
- map size: 138x110 m
- 8 robots, sensor range: 10 m
Comparison
Hospital-section Map

- map size: 271x110 m
- 10 robots, Hungarian strategy, sensor range 10 m
- the final map contains 1226 vertices
Graphical Comparison

(a) Greedy

(b) BLE

(c) Hungarian

(d) Kmeans
Comparision of a Polygonal and Grid-based Representation

• Hospital-section map, 1 robot
• size 271x110 m

Polygonal

• 1226 vertices

Occupancy Grid

• 1 cell: 5 cm ⇒ 12 076 800

<table>
<thead>
<tr>
<th>S [m²]</th>
<th>t_{plan} [ms]</th>
<th>t_{total} [ms]</th>
<th>t_{plan} [ms]</th>
<th>t_{total} [ms]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>10</td>
<td>35</td>
<td>274</td>
<td>471</td>
</tr>
<tr>
<td>3000</td>
<td>22</td>
<td>50</td>
<td>985</td>
<td>1224</td>
</tr>
<tr>
<td>5000</td>
<td>14</td>
<td>70</td>
<td>2141</td>
<td>2410</td>
</tr>
<tr>
<td>10000</td>
<td>45</td>
<td>116</td>
<td>3369</td>
<td>3854</td>
</tr>
<tr>
<td>13000</td>
<td>36</td>
<td>132</td>
<td>4902</td>
<td>5611</td>
</tr>
</tbody>
</table>
Conclusion

• Multi-robot exploration framework for the polygonal domain implemented in ROS
• Extension of Clipper with user-defined information
• Implementation of several goal-assignment strategies
• Experimental evaluation

Future work
• Code cleaning \(\leadsto\) code publication
• Statistical evaluation of strategies on real robots
• Implement more strategies
Conclusion

- Multi-robot exploration framework for the polygonal domain implemented in ROS
- Extension of Clipper with user-defined information
- Implementation of several goal-assignment strategies
- Experimental evaluation

Future work
- Code cleaning ⟷ code publication
- Statistical evaluation of strategies on real robots
- Implement more strategies
Thank You!

Technology Agency of the Czech Republic

The presented work is supported under Project No. TE01020197