Reduced Time-Expansion Graphs and Goal Decomposition for Solving Cooperative Path Finding Sub-optimally

Pavel Surynek
Charles University in Prague
Czech Republic

24th International Joint Conference on Artificial Intelligence (IJCAI 2015), Buenos Aires, Argentina
Cooperative Path Finding

- CPF
  - a group of agents (robots, cars, units in RTS, ...)
  - each agent has unique **start** and **goal** location
  - **collisions** must be avoided
  - environment - undirected graph

CPF $\Sigma = (G, \{a_1, a_2, a_3\}, \alpha_0, \alpha_+)$
Solving CPF by Reducing it to SAT

- **expand** (copy) graph $G$ over time
  - the number of expansions $n$ is specified
  - **represent arrangements** of agents in time
  - **encode** relocation of agents through expanded graph as a propositional formula $F(n)$
    - constraints to check validity of transitions between arrangements at time-steps
  - **ask SAT solver** whether $F(n)$ is solvable
Standard Time Expansion

- each expansion corresponds to a time step
  - placement of each agent at each time step is explicitly represented
- too many expansions in case of long makespan
  - can be used for makespan optimal solving of CPF

CPF \( \Sigma = (G = (V, E), \{a_1, a_2\}, \alpha_0, \alpha_+ ) \)

= \begin{align*}
\alpha_0 & \quad \alpha_1 \quad \alpha_1 \quad \alpha_3 \quad \alpha_+ = \alpha_+ \\
\alpha_+ & \quad \alpha_0 \quad \alpha_1 \quad \alpha_1 \quad \alpha_3
\end{align*}

Exp_T(G, 4)
Reduced Time Expansion

- expansions correspond **avoidance** among agents
  - movements of agents are represented as **vertex disjoint** paths
  - **few expansions** for small interaction among agents
    - even if makespan is large
  - can used for makespan **suboptimal** CPF solving

CPF $\Sigma=(G=(V,E), \{a_1,a_2\}, \alpha_0, \alpha_+)$
Goal Decomposition

• observation
  – few expansions are needed if there is little difference between the initial and goal arrangement

• place agents one by one (UniROBOT)
  – solve a separate CPF for single agent placement
    • few expansions ⇒ small propositional formula
      ⇒ easy SAT
    • merge solutions into an overall solution of the original CPF
Experimental Evaluation

- **setup**
  - 4-connected grid, with obstacles
  - SAT-based solving with various propositional encodings is compared with A*-based algorithms

### Average runtime | Grid 8x8 | 20% obstacles

<table>
<thead>
<tr>
<th></th>
<th>OD+ID</th>
<th>INVERSE</th>
<th>ALL-DIFFERENT</th>
<th>DIRECT</th>
<th>MATCHING</th>
<th>SIMPLIFIED</th>
<th>UNIROBOT</th>
<th>WHCA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Makespan</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>optimal</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WHCA*</td>
<td>5.6</td>
<td>9.3</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>UniROBOT</td>
<td>9.3</td>
<td>15.8</td>
<td>33.0</td>
<td>49.3</td>
<td>83.4</td>
<td>96.1</td>
<td>131.4</td>
<td>154.1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>4</th>
<th>8</th>
<th>12</th>
<th>16</th>
<th>20</th>
<th>24</th>
<th>28</th>
<th>32</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

![Runtime (seconds) vs |A| graph](image)