Bounded-Cost Search Using Estimates of Uncertainty

Maximilian Fickert, Tianyi Gu, Wheeler Rumal

The Problem Setting

• Motivation: many real-world problems are too hard to solve optimally. Need bounded-cost solution!
• Problem: initial state \((I)\), goal states \((G)\), and a cost bound \(C\).
• Objective: Find a solution with cost at most \(C\) as fast as possible.

Previous Approaches

• Standard heuristic search with pruning on \(C\)
• Potential Search (PTS)\(^1\)
  \(\rightarrow\) best-first search on \(\frac{h}{p}\)
• Bounded-cost Explicit Estimation Search (BEES)\(^2\)
  \(\rightarrow\) focal search:
  • open sorted by \(f\), only nodes with \(g + h \leq C\)
  • focal sorted by \(d\), only nodes with \(g + \hat{h} \leq C\) \(p(n) > 95\%\)

Expected Effort Search

Best-first search on the expected effort: \(\frac{T}{p}\)
- \(T(n)\): search effort to find a solution under \(n\)
- \(p(n)\): probability that \(n\) leads to a solution within \(C\)

\(\begin{align*}
T &= 10 \\
p &= 0.5 \\
\therefore T &= \frac{10}{0.5} \rightarrow 20
\end{align*}\)

\(\begin{align*}
T &= 6 \\
p &= 0.25 \\
\therefore T &= \frac{6}{0.25} \rightarrow 24
\end{align*}\)

How to obtain \(T\) and \(p\)?

- Obtaining \(T\) use distance-to-go \(d\)
- Obtaining \(p\) from belief distributions

Thoretical Analysis

XES optimizes search effort, assuming
- the search explores one subtree at a time,
- subtrees are independent, and
- subtrees are abandoned after spending \(T(n)\) time.

Using \(p\) in BEES

BEES\(^95\):
- open sorted by \(f\), only nodes with \(g + h \leq C\)
- focal sorted by \(d\), only nodes with \(g + \hat{h} \leq C\) \(p(n) > 95\%\)

Experiments

Planning Domains: IPC’18 bounded-cost track

<table>
<thead>
<tr>
<th>Coverage</th>
<th>GBFS</th>
<th>PTS</th>
<th>BEES</th>
<th>BEES(^95)</th>
<th>XES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agricola (20)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Caldera (20)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Caldera-split (20)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DataNetwork (20)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Nurikabe (20)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Settlers (20)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Snake (20)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Spider (20)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Termes (20)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Sum (180)</td>
<td>45</td>
<td>52</td>
<td>60</td>
<td>64</td>
<td>66</td>
</tr>
</tbody>
</table>

Expansions (\(\times 10^3\)) 1.93 3.93 2.10 2.25 1.77

Search Domains:

Bounded-cost algorithms dominate GBFS; XES is best overall.