On Improving Plan Quality via Local Enhancements

Tomáš Balyo*, Roman Barták*, and Pavel Surynek*, **
{tomas.balyo, roman.bartak, pavel.surynek}@mff.cuni.cz

*Faculty of Mathematics and Physics
Charles University in Prague
Czech Republic

**Graduate School of Maritime Sciences
Kobe University, Japan

1. Background and Objectives

What is Planning?
- Finding plans – sequences of actions
- Input
  - A set of actions with preconditions and effects
  - Descriptions of the initial state and the goal state
- Output
  - Plan = a valid sequence of actions that transform the world from the initial state to the goal state

Planning Algorithms
- There are already many successful planners
- **Optimal planners** (find shortest possible plans) are **slow** and cannot handle large problems
- **Suboptimal planners** (produce longer plans) are a lot **faster** and can find plans for harder problems
- We need to choose between quality and performance

Our Goal
- Combine the planning approaches to have both **performance** and plans of **good quality**

2. The Proposed Method

Our Approach – The Basic Idea
A) Find a sub-optimal plan \( P \)
B) Select a sub-plan (sub-sequence) of \( P \)
C) Replace it with an improved subplan (thus improving \( P \) itself)
D) Keep repeating B) and C) until the entire plan is optimal or time is out

How do we do that?
A) A fast sub-optimal planner finds the initial plan \( P \) (we used LPG, but any fast planner is suitable)
B) The sub-plans are selected by systematically shifting a window of increasing size through \( P \)
C) The subplan optimization is formulated as a planning problem and solved by an optimal planner (we used the SAT-based SASE approach)

Window shifting methods: Halfstep (left) and Fullstep (right)

3. Results and Conclusions

Experiments
- Cumulative results of eight classical STRIPS domains from the International Planning Competitions
- Compared the new method with the fast planner LPG and the optimal planner SASE

<table>
<thead>
<tr>
<th>Method</th>
<th>Makespan</th>
<th>( \Delta )LPG</th>
<th>( \Delta )SASE</th>
</tr>
</thead>
<tbody>
<tr>
<td>LPG</td>
<td>71.27</td>
<td>0.00</td>
<td>-75.38</td>
</tr>
<tr>
<td>SASE</td>
<td>146.65</td>
<td>75.38</td>
<td>0.00</td>
</tr>
<tr>
<td>Expo-fullstep</td>
<td>170.41</td>
<td>99.14</td>
<td>23.76</td>
</tr>
<tr>
<td>Turbo-halfstep</td>
<td>179.53</td>
<td>108.25</td>
<td>32.87</td>
</tr>
</tbody>
</table>

The makespan score of a planner indicates the number and quality of the produced plans. Higher value = better performance

The comparison of three window enlargement strategies:
- turbo = increase by one; expo = increase by a factor of 1.5;
- random = random size between 2 and 20

Does it work? – Conclusion
- We can solve as many problems as the **fastest planning algorithm**
- According to our experiments the plans are always **significantly improved**, moreover an **optimal** (or almost optimal) plan is often produced
- It is a successful anytime algorithm capable of finding optimal plans

SoCS 2012: The Fifth Annual Symposium on Combinatorial Search