Learning Portfolios of Automatically Tuned Planners

Jendrik Seipp\textsuperscript{1} \quad Manuel Braun\textsuperscript{1} \quad Johannes Garimort\textsuperscript{1} \quad Malte Helmert\textsuperscript{2}

\textsuperscript{1}Albert-Ludwigs-Universität Freiburg, Germany
\textsuperscript{2}Universität Basel, Switzerland

June 2012
IPC 2011 – Sequential Satisficing Track

Results

<table>
<thead>
<tr>
<th>Team</th>
<th>Quality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Autotune2</td>
<td>160</td>
</tr>
<tr>
<td>Autotune1</td>
<td></td>
</tr>
<tr>
<td>StoneSoup2</td>
<td>180</td>
</tr>
<tr>
<td>StoneSoup1</td>
<td></td>
</tr>
<tr>
<td>LAMA2011</td>
<td>240</td>
</tr>
</tbody>
</table>
Motivation

- **Tuned planners:**
  - Tune for **complete** benchmark set
  - Commit to **single** planner

- **Portfolio planners:**
  - Manually select planners
  - Calculate times greedily

- **Our approach:**
  - Tune **one planner for each domain in training set** automatically
  - Evaluate **multiple** portfolio generation methods
Overview

- Domain Tuning
- Portfolio Learning
Domain Tuning
Tuning Procedure – Domains

- Tune Fast Downward with ParamILS for each domain
**Tuning Procedure – Configurations**

- **Heuristics:** $h^{FF}$, $h^{add}$, $h^{cg}$, $h^{cea}$, $h^{LM}$
- **Searches:** eager, lazy
- **Type of landmarks, cost-handling, preferred operators**
- **Numerous combination options and conditional parameters**
  $\rightarrow 2.99 \cdot 10^{13}$ configurations
Tuning Results – Trends

- Preferred operators (19/21)
- Lazy search (20x), eager search (1x)
- Most configurations use one (10x) or two (9x) heuristics
  - $h_{\text{FF}}$ (12x), $h_{\text{LM}}$ (11x), $h_{\text{cg}}$ (6x), $h_{\text{cea}}$ (4x), $h_{\text{add}}$ (1x)
### Tuning Results

<table>
<thead>
<tr>
<th>coverage</th>
<th>optical-t</th>
<th>pathways</th>
<th>pipes-t</th>
<th>tpp</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>optical-t (48)</td>
<td>21</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>...</td>
</tr>
<tr>
<td>pathways (30)</td>
<td>22</td>
<td>30</td>
<td>29</td>
<td>30</td>
<td>...</td>
</tr>
<tr>
<td>pipes-t (50)</td>
<td>26</td>
<td>39</td>
<td>42</td>
<td>38</td>
<td>...</td>
</tr>
<tr>
<td>tpp (30)</td>
<td>24</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>...</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Portfolio Learning
Portfolio Generators

- **Input:** planners, results on training set, total time limit
- **Output:** \{depot: 18s, gripper: 65s, ... \}
Hill-climbing in the portfolio space

Start: \{depot: 0, gripper: 0, \ldots\}

Successors:
\{depot: g, gripper: 0, \ldots\}, \{depot: 0, gripper: g, \ldots\}, \ldots

Choose best and repeat
Run all planners for same amount of time
Result: {depot: 85, gripper: 85, ...}
Brute force
For all subset sizes \{1, \ldots, 21\} compute best portfolio with equal time shares
Cluster

- Find $k$ clusters with $k$-means
- Cluster by quality
- From each cluster choose best planner
- Give all planners equal time shares
Increasing Time Limit

- Iteratively increase the portfolio time limit
- Get problems that can be solved in that limit
- Find best planner for these problems
- Give it the needed time
- Repeat until no more problems solvable or time limit exceeded
Domain-wise

- Iteratively retrieve domain with highest improvement potential
- Give the fastest improving planner the needed time
- Continue until total time limit reached or no more domains can be improved
Randomized Iterative Search

- Use any existing portfolio as initialization (e.g. uniform)
- Successors:
  - Swap time slice between planners
  - Collect time from all planners and give it to single one
- Commit to first successor improving score
- Run until score stagnates long enough
Portfolio Results
30 minutes

![Quality Chart]

- Autotune2
- Autotune1
- StoneSoup2
- StoneSoup1
- LAMA2011
- StoneSoup-110
- Uniform
- Selector-12
- Cluster-16
- ITL-10
- Domainwise
- RIS
Uniform portfolio outperforms LAMA even in 3 min setting
Other portfolios are even better
Less planners in portfolio when less time is available
No portfolio dominates others for all timeouts
*Cluster* and *Increasing Time Limit* among best performers
*Randomized Iterative Search* prone to overfitting
Outlook

- Promising initial results for optimal configurations
- Adaptively select next configuration
- Use more heterogeneous planners
- Apply automatic portfolio diversification in other areas
Tuning for domains is effective
Tuned planners yield very good results in portfolio