Learning Portfolios of Automatically Tuned Planners

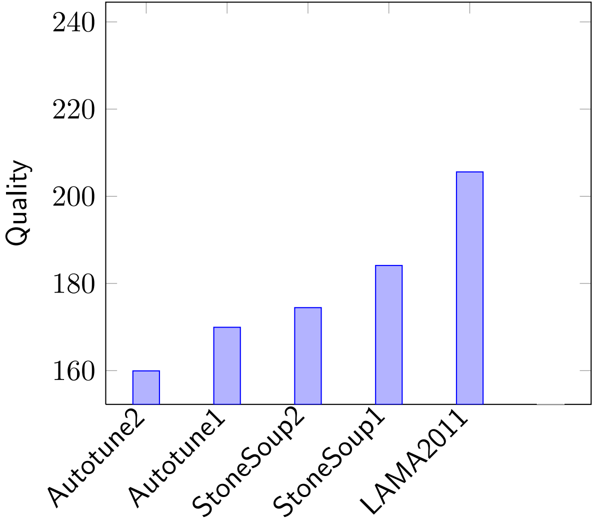
Jendrik Seipp [1] Manuel Braun [1] Johannes Garimort [1]
Malte Helmert [2]

[1]Albert-Ludwigs-Universität Freiburg, Germany

[2]Universität Basel, Switzerland

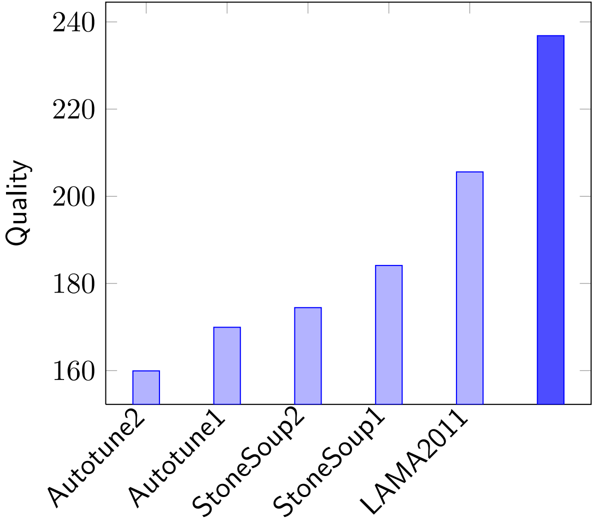
June 2012

IPC 2011 – Sequential Satisficing Track
Results



IPC 2011 – Sequential Satisficing Track

Results



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

- Training set of 21 former IPC domains (1998–2006)
- 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^{FF} (12x), h^{LM} (11x), h^{cg} (6x), h^{cea} (4x), h^{add} (1x)

Tuning Results

coverage		Planners: optical-t	pathways	pipes-t	tpp	...
Domains	optical-t (48)	**21**	0	3	0	...
	pathways (30)	22	**30**	29	**30**	...
	pipes-t (50)	26	39	**42**	38	...
	tpp (30)	24	**30**	**30**	**30**	...
	...	...	...	...	...	...

Portfolio Learning

Portfolio Generators

- Input: planners, results on training set, total time limit
- Output: {depot: 18s, gripper: 65s, ...}

Stone Soup

- Hill-climbing in the portfolio space
- Start: $\{$depot: 0, gripper: 0, $\dots\}$
- Successors:
  $\{$depot: g, gripper: 0, $\dots\}$, $\{$depot: 0, gripper: g, $\dots\}$, $\dots$
- Choose best and repeat

Uniform

- Run all planners for same amout of time
- Result: $\{$depot: 85, gripper: 85, ...$\}$

Selector

- 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

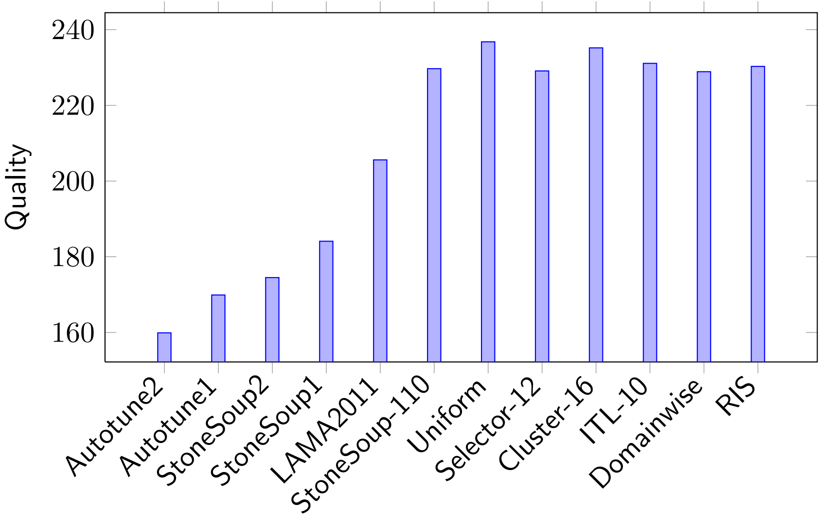
- Iteratively retrieve domain with highest improvement potential
- Give the fastest improving planner the needed time
- Continue until total time limit reached or no more domains can be improved

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



Different timeouts

1, 3, 5, 15 minutes

- 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

Summary

- Tuning for domains is effective
- Tuned planners yield very good results in portfolio