Learning Portfolios of Automatically Tuned Planners: Detailed Results

Jendrik Seipp
Albert-Ludwigs-Universität Freiburg
seipp@informatik.uni-freiburg.de

Manuel Braun
Albert-Ludwigs-Universität Freiburg
braun@informatik.uni-freiburg.de

Johannes Garimort
Albert-Ludwigs-Universität Freiburg
garimort@informatik.uni-freiburg.de

Malte Helmert
Universität Basel
malte.helmert@unibas.ch

Albert-Ludwigs-Universität Freiburg, Technische Fakultät, Technical Report 268, March 2012

Introduction

This technical report provides additional experimental data for the ICAPS 2012 paper "Learning portfolios of automatically tuned planners" [9].

Specifically, it contains the following information:

- Which configurations of Fast Downward are found in each training domain by the tuning process (Table 1)?
- How many problems in the training set does each of these configurations solve in each domain (Table 2)?
- Which portfolios are learned for different overall timeouts between 1–30 minutes (Tables 3–7)?
- How well do the learned portfolios for these different timeouts perform on the evaluation set, compared to each other and to LAMA 2011 with the same timeout (Tables 8–12)? For the IPC time limit of 30 minutes, we also report results for some other IPC planners related to our approach (Table 12).
- What is the coverage of our 30-minute portfolio on the evaluation set, compared to each other and some related IPC 2011 participants (Table 13)?

In the following sections, we describe this data in some more detail.

In the appendix, we give the parameter settings for the tuned configurations according to the command-line syntax used by Fast Downward (March 2012 version).

Tuning Results

The configurations we obtained from automatic tuning can be seen in detail in Table 1. The most eye-catching characteristic is that lazy best-first search is vastly preferred over eager best-first search (20:1). Nineteen of the 21 configurations use preferred operators.

Furthermore Keyder et al. landmarks [6] (11x) are preferred over those from Richter et al. [8] (4x). Also the configurations are mostly using either one (10x) or two (9x) heuristics. The heuristics used are h^{FF} (12x), landmarks (11x), h^{cg} (6x), h^{cea} (4x) and h^{add} (1x).

All landmark heuristics used were of the admissible kind (i.e., used cost partitioning) due to a bug in the configuration file used for ParamILS. Parameters that are not shown ended up at their default value in each experiment or referred to settings that are irrelevant given the other parameter choices. (For example, the "use conjunctive landmarks" option never applies because no h^m landmarks with $m > 1$ were used in any of the domains.)

domain	heuristics	heuristic cost type	landmarks	remove LM orders?	keep disjunctive LMs?	eager or lazy search?	search cost type	reopening?	heuristic weight	preferred op. heuristics	preferred queue boost
airport	h^{cea}, h^{FF}, h^{lm}	2	RHW	false	true	lazy	2	false	∞	h^{cea}, h^{lm}	0
depot	h^{cea}, h^{lm}	2	h^1	false		lazy	2	false	∞	h^{cea}	100
driverlog	h^{FF}, h^{lm}	1	RHW	true	true	lazy	2	false	∞	h^{lm}	0
freecell	h^{lm}	0	h^1	true		eager	1	true	5		
grid	h^{cg}, h^{FF}, h^{lm}	2	RHW	true	true	lazy	2	false	10	h^{cg}, h^{lm}	1000
logistics-2000	h^{cea}	1				lazy	1	false	10	h^{cea}	2000
miconic-full	h^{FF}	1				lazy	1	false	∞	h^{FF}	5000
mprime	h^{cea}	2				lazy	2	false	5	h^{cea}	1000
optical-telegraphs	h^{cg}, h^{FF}	1				lazy	1	false	10	h^{cg}	100
pathways	h^{FF}, h^{lm}	2	h^1	true		lazy	0	false	∞	h^{lm}	5000
philosophers	h^{cg}	2				lazy	2	false	10	h^{cg}	0
pipes-no-tankage	h^{FF}, h^{lm}	0	RHW, h^1	false	false	lazy	2	false	10	h^{FF}	500
pipes-tankage	h^{FF}	1				lazy	1	false	7	h^{FF}	5000
psr-large	h^{add}	0				lazy	0	true	∞	h^{add}	0
rovers	h^{FF}, h^{lm}	0	h^1	false		lazy	2	true	10+ε	h^{lm}	200
satellite	h^{cg}	2				lazy	2	true	∞	h^{cg}	0
schedule	h^{FF}, h^{lm}	0	h^1	true		lazy	1	false	∞	h^{lm}, h^{FF}	1000
storage	h^{cg}, h^{lm}	1	h^1	true		lazy	1	false	∞	h^{cg}	0
tpp	h^{FF}, h^{lm}	2	h^1	true		lazy	0	false	10	h^{lm}	500
trucks	h^{FF}	1				lazy	1	true	1.5	h^{FF}	5000
zenotravel	h^{cg}	1				lazy	1	true	2+ε		

Table 1: The tuned configurations and their features. The columns are, from left to right: (1) the domain on which the configuration was tuned; (2) the set of heuristics used for search (h^{add} = additive heuristic [1]; h^{cea} = context-enhanced additive [3]; h^{cg} = causal-graph [2]; h^{FF} = FF heuristic [4]; h^{lm} = admissible landmark heuristic [8, 5]); (3) how did the heuristics treat costs? (0 = unmodified; 1 = treat as unit-cost; 2 = add +1 to each operator cost unless all operator costs are 1); (4) how were landmarks generated? (RHW = Richter/Helmert/Westphal [8]; h^1 = h^m-based with $m = 1$ [6]; column only applies if h^{lm} was used); (5) were landmark orderings discarded before search? (column only applies if h^{lm} was used); (6) were disjunctive landmarks kept for search? (column only applies if h^{lm} with RHW landmarks was used); (7) was the search algorithm eager or lazy [7]?; (8) how did the search heuristics treat costs for purposes of path cost computation? (same values as in column 3); (9) were closed search nodes reopened when finding a cheaper path to their states?; (10) how were heuristic values h weighted relative to path costs g? (∞ = greedy best-first search; $x + \varepsilon$ = weight h by x and prefer lower h values to break ties); (11) which heuristics were used to compute preferred operators?; (12) how large was the boost value for preferred operator queues [7]? (column only applies if preferred operators were used)

	airport	depot	driverlog	freecell	grid	logistics00	miconic-full	mprime	optical-t	pathways	philosophers	pipes-nt	pipes-t	psr-large	rovers	satellite	schedule	storage	tpp	trucks	zenotravel
airport (50)	47	42	**48**	25	42	40	35	42	37	44	26	36	35	37	46	26	43	27	47	34	28
depot (22)	18	**19**	18	12	18	17	**19**	**19**	16	16	13	**19**	**19**	14	17	12	18	17	16	15	12
driverlog (20)	**20**	**20**	**20**	16	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	18	18
freecell (80)	78	78	**80**	**80**	**80**	77	79	76	77	**80**	66	**80**	**80**	76	**80**	70	77	**80**	**80**	77	43
grid (5)	**5**	**5**	**5**	3	**5**	**5**	**5**	**5**	**5**	**5**	4	**5**	**5**	3	**5**	4	**5**	4	**5**	4	**5**
logistics00 (28)	**28**	**28**	**28**	**28**	**28**	**28**	**28**	**28**	**28**	**28**	**28**	**28**	**28**	**28**	**28**	**28**	**28**	**28**	**28**	**28**	**28**
miconic-full (150)	0	0	0	0	0	**139**	138	**139**	138	0	137	0	138	138	0	134	0	0	0	137	134
mprime (35)	**35**	**35**	**35**	21	**35**	**35**	**35**	**35**	**35**	**35**	**35**	**35**	**35**	34	34	**35**	**35**	**35**	34	**35**	**35**
optical-t (48)	0	0	0	0	0	4	2	4	**21**	0	2	0	3	1	0	1	0	0	0	2	2
pathways (30)	23	29	29	5	29	28	29	28	22	**30**	7	**30**	29	14	**30**	7	**30**	11	**30**	6	7
philosophers (48)	0	0	0	0	0	**48**	45	**48**	**48**	0	**48**	0	45	**48**	0	**48**	0	0	0	5	5
pipes-nt (50)	43	43	43	29	43	42	44	42	41	44	30	44	44	32	43	23	**45**	35	43	43	28
pipes-t (50)	25	37	30	17	40	34	41	33	26	39	21	40	**42**	20	38	17	38	17	38	38	20
psr-large (50)	0	0	0	0	0	31	15	31	26	0	33	0	20	**36**	0	33	0	0	0	19	31
rovers (40)	39	**40**	**40**	18	**40**	**40**	**40**	**40**	34	**40**	32	**40**	**40**	32	**40**	32	**40**	32	**40**	23	28
satellite (36)	**36**	**36**	**36**	11	**36**	**36**	**36**	**36**	**36**	**36**	**36**	34	**36**	**36**	**36**	**36**	**36**	**36**	**36**	35	27
schedule (150)	143	**150**	144	61	**150**	144	148	127	145	**150**	61	**150**	149	65	146	62	**150**	**150**	149	99	26
storage (30)	18	18	19	19	19	19	20	20	20	**21**	17	18	20	17	**21**	17	20	**21**	**21**	17	19
tpp (30)	25	28	**30**	10	**30**	29	**30**	26	24	**30**	27	**30**	**30**	26	**30**	26	**30**	26	**30**	8	10
trucks (30)	15	16	13	7	16	16	20	16	16	13	12	18	20	18	13	11	15	9	13	**23**	10
zenotravel (20)	**20**	**20**	**20**	15	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**

Table 2: Coverage (number of solved problems) of all tuned planner configurations (columns) on all domains (rows). Best values are highlighted in bold, diagonal entries are underlined.

		airport	depot	driverlog	freecell	grid	logistics00	miconic-full	mprime	optical-t	pathways	philosophers	pipes-nt	pipes-t	psr-large	rovers	satellite	schedule	storage	tpp	trucks strips	zenotravel
Cluster	k=12	5	—	5	—	5	5	—	—	—	5	5	—	5	5	5	—	—	—	5	5	5
	k=7	—	8	—	—	8	8	—	—	—	—	—	—	8	—	8	8	—	—	—	8	—
DW		—	6	3	—	—	1	1—	2	—	8	1	—	—	—	1—	5	15	—	1	3	1
Inc. Time L.	t=1s	—	—	—	—	—	8	1—	3	—	12	—	—	1	8	11	—	7	—	—	—	—
	t=5s	—	—	—	—	—	1—	16	—	—	—	—	—	5	16	13	—	—	—	—	—	—
RIS		1	1	—	1	—	3	—	—	1	—	—	7	3	8	21	2	2	—	5	5	—
Selector		—	—	—	—	—	12	—	—	—	12	—	12	—	12	—	—	—	—	—	12	—
Stone Soup	g=1—	—	—	—	—	—	19	—	—	—	—	—	—	1—	—	19	—	—	—	—	1—	—
	g=2—	—	—	—	—	—	19	—	—	—	—	—	—	2—	—	19	—	—	—	—	—	—
Uniform		2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2

Table 3: Portfolios learned for a time limit of 1 minute.

		airport	depot	driverlog	freecell	grid	logistics00	miconic-full	mprime	optical-t	pathways	philosophers	pipes-nt	pipes-t	psr-large	rovers	satellite	schedule	storage	tpp	trucks strips	zenotravel
Cluster	k=1—	—	—	—	—	18	18	—	—	—	18	18	18	18	18	18	—	—	—	18	18	—
DW		—	6	3	—	—	13	53	13	—	8	1—	—	—	5	2—	7	45	—	1	3	1
Inc. Time Limit	t=1s	—	—	23	—	—	8	19	19	15	18	17	—	21	8	11	—	13	8	—	—	—
RIS		4	1	9	2	—	—	32	29	12	—	—	7	1	9	2	1	14	22	12	2—	3
Selector		18	—	—	—	18	—	18	18	18	18	18	—	—	18	—	—	18	—	—	18	—
Stone Soup	g=1—	8	—	—	9	—	29	27	—	—	—	1—	—	1—	9	4—	—	1—	—	—	2—	—
Uniform		8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8

Table 4: Portfolios learned for a time limit of 3 minutes.

		airport	depot	driverlog	freecell	grid	logistics00	miconic-full	mprime	optical-t	pathways	philosophers	pipes-nt	pipes-t	psr-large	rovers	satellite	schedule	storage	tpp	trucks strips	zenotravel
Cluster	k=13	—	23	23	23	—	23	23	—	23	—	—	23	23	23	—	23	—	23	23	23	0
DW		—	6	10	—	52	13	53	13	11	8	53	—	—	10	20	7	45	—	1	3	1
Inc. Time Limit	t=1s	—	—	35	—	37	8	32	19	15	18	44	—	21	8	11	—	30	22	—	—	0
RIS		12	3	35	12	2	26	32	12	12	—	10	7	3	11	12	1	30	22	5	50	3
Selector		33	—	33	—	—	—	33	33	33	—	—	—	—	33	—	—	33	—	33	33	0
Stone Soup	g=10	12	10	9	9	—	26	36	—	—	—	10	7	10	9	40	—	10	20	5	50	10
Uniform		14	14	14	14	14	14	14	14	14	14	14	14	14	14	14	14	14	14	14	14	14

Table 5: Portfolios learned for a time limit of 5 minutes.

		airport	depot	driverlog	freecell	grid	logistics00	miconic-full	mprime	optical-t	pathways	philosophers	pipes-nt	pipes-t	psr-large	rovers	satellite	schedule	storage	tpp	trucks strips	zenotravel
Cluster	k=13	69	—	69	—	69	69	—	—	—	69	69	69	69	69	—	—	69	—	69	69	69
DW		—	6	68	—	52	13	53	13	11	353	53	—	—	141	10	60	45	—	1	3	1
Inc. Time Limit	t=1s	—	15	139	—	52	8	104	144	15	80	69	—	21	137	11	53	30	22	—	—	0
RIS		41	22	55	65	52	5	36	29	51	20	—	33	14	13	140	94	30	11	2	91	96
Selector		69	—	69	69	69	—	—	69	69	69	69	69	—	69	—	—	69	—	—	69	69
Stone Soup	g=40	40	—	35	36	—	5	31	197	26	—	—	23	38	11	40	—	40	—	79	80	73
Uniform		42	42	42	42	42	42	42	42	42	42	42	42	42	42	42	42	42	42	42	42	42

Table 6: Portfolios learned for a time limit of 15 minutes.

		airport	depot	driverlog	freecell	grid	logistics00	miconic-full	mprime	optical-t	pathways	philosophers	pipes-nt	pipes-t	psr-large	rovers	satellite	schedule	storage	tpp	trucks strips	zenotravel
Cluster	k=12	—	150	—	—	150	150	—	150	—	—	150	150	150	150	—	150	150	150	—	150	0
	k=16	112	112	—	112	112	—	—	112	112	—	—	112	112	112	112	112	112	112	112	112	112
DW		—	6	25	—	52	13	53	156	11	353	53	298	214	137	10	60	281	—	1	3	1
Inc. Time L.	t=10s	—	169	139	—	52	20	104	144	61	259	69	170	214	137	25	—	237	—	—	—	0
	t=5s	—	169	139	—	52	10	104	144	38	266	53	196	214	137	13	—	237	—	28	—	0
RIS		40	3	143	64	92	6	10	256	58	19	—	208	214	31	142	1	43	30	26	311	103
Selector		150	—	150	150	150	—	—	150	150	150	—	150	—	150	—	—	150	—	—	150	150
Stone Soup	g=100	81	—	139	64	90	—	—	261	49	—	—	298	79	60	97	—	43	—	—	91	96
	g=110	108	—	139	64	90	—	—	197	82	—	—	208	214	—	104	—	110	—	—	91	96
Uniform		85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85

Table 7: Portfolios learned for a time limit of 30 minutes.

Quality	**LAMA**	**Stone Soup**		**Uniform**	**Selector**	**Cluster**		**Inc. Time Limit**		**DW**	**RIS**
	2011	g=10	g=20			k=12	k=7	t=5s	t=1s		
barman	17.49	0.00	0.00	18.16	18.12	18.20	3.60	10.41	**18.29**	18.29	3.55
elevators	6.85	**8.47**	7.99	7.00	7.33	7.48	7.17	6.60	8.26	7.16	5.05
floortile	2.50	1.98	2.13	3.14	2.41	3.03	2.67	**3.37**	3.28	3.14	3.28
nomystery	9.66	16.66	15.62	16.62	14.70	**16.67**	16.64	14.47	14.53	15.67	16.58
openstacks	**10.35**	5.76	6.30	4.38	3.34	6.04	6.97	7.31	5.31	4.52	2.41
parcprinter	**19.95**	18.57	18.58	19.74	18.86	19.26	19.45	18.74	18.71	19.49	19.74
parking	2.72	0.00	0.00	0.75	0.78	2.75	**2.85**	0.00	0.77	1.42	0.00
pegsol	18.49	18.34	17.93	18.56	18.14	18.09	18.63	17.55	17.71	18.32	**19.01**
scanalyzer	15.16	15.36	15.15	16.35	15.30	**16.41**	15.74	14.70	14.94	16.05	15.68
sokoban	10.44	10.45	**13.89**	7.51	11.17	10.78	9.27	11.18	9.18	9.90	10.09
tidybot	9.87	**10.98**	10.71	0.00	8.30	3.77	5.73	7.93	6.51	6.21	7.95
transport	7.36	8.57	8.21	3.18	8.69	7.48	4.44	5.92	**9.01**	7.89	7.96
visitall	10.86	**19.90**	**19.90**	11.90	17.90	13.90	15.90	17.90	17.90	16.90	**19.90**
woodworking	13.90	11.21	11.30	13.37	15.14	14.84	12.26	**15.15**	14.52	11.26	15.13
Sum	155.60	146.24	147.73	140.67	**160.20**	158.70	141.34	151.25	158.94	156.22	146.33

Table 8: The qualities of our portfolios (right) compared to LAMA (left) on the IPC 2011 sequential satisficing track domains with a time limit of 1 minute. Best planners in each domain are highlighted in bold.

Portfolio Learning Results

We learned portfolios for different time limits (1 min., 3 min. 5 min., 15 min. and 30 min.). The times that were assigned to each planner can be seen in tables 3 to 12. We noticed that most generators assign some planners (the ones for *pipes*, *mprime*) quite a large proportion of time in the 30 minutes setting, while they get much less time in the settings with low time limits. With 1 minute time limit, some planners (*storage*, *freecell*, *optical-telegraphs*) get virtually no time at all. This indicates that these planners are not useful in such settings. Generally, it can be seen that the portfolios get more sparse with less time available.

Results for Different Timeouts

The results on the IPC 2011 benchmarks for different time limits are shown in Tables 8–12. While the uniform portfolio still outperforms LAMA with a time limit of 3 minutes, it doesn't seem to be the best choice when even less time is available. Though no portfolio generator clearly dominates the others, the *Cluster* and *Increasing Time Limit* generators seem to perform quite well. *Randomized Iterative Search* (RIS) achieves the best training score in all settings but seems to be somewhat prone to over-fitting.

Detailed Results for Main Experiment

In Table 12 we compare the solution qualities of our portfolios with the ones obtained by IPC 2011 competitors. In some domains the quality highly differs between different (portfolio) planners due to the used heuristics. For instance, in the *visitall* domain performance greatly depends on landmark usage: All configurations using Keyder et al. landmarks solve 17 or more instances, those using Richter et al. landmarks solve 8–14, and those without landmarks only solve 1–5 problems.

As can be seen in the table all of our portfolios perform better than LAMA 2011, the winner of the IPC 2011 challenge. Interestingly the best performer is the uniform portfolio that runs all tuned configurations for the same amount of time.

The competition winner achieves 206 points, our uniform portfolio 237, perfection would be 280. That's 42% of the delta to perfection, and LAMA 2011 is a strong planner. Moreover, LAMA 2011 was informed by experience with the IPC 2008, with many of the IPC 2008 domains reused in the IPC 2011. Our portfolio however, does not use any IPC 2008 results.

Table 13 shows the coverage for the same set of planners on the IPC 2011 domains. All of our portfolios solve more instances than LAMA 2011 with the uniform portfolio solving the most instances.

References

[1] Blai Bonet and Héctor Geffner. Planning as heuristic search. *Artificial Intelligence*, 129(1):5–33, 2001.

Quality	**LAMA**	**Stone Soup**	**Uniform**	**Selector**	**Cluster**	**Inc. Time Limit**	**DW**	**RIS**
	2011	g=10			k=10	t=5s		
barman	17.49	13.01	18.41	**18.46**	18.27	18.43	18.33	13.24
elevators	8.77	10.64	8.59	8.44	9.58	8.60	**11.14**	9.73
floortile	3.55	3.28	3.28	3.86	**4.01**	3.28	3.28	2.89
nomystery	9.70	17.63	16.67	16.63	**17.65**	15.60	16.54	17.63
openstacks	**14.11**	9.55	8.60	10.71	10.15	10.57	9.48	9.93
parcprinter	**19.95**	19.25	19.78	19.60	19.43	19.61	19.49	19.78
parking	6.12	2.75	2.95	5.55	5.08	**7.06**	4.69	3.78
pegsol	**19.30**	18.87	19.19	18.75	19.07	18.07	18.90	18.84
scanalyzer	15.70	16.78	16.56	15.81	15.97	15.98	17.00	**17.27**
sokoban	11.22	13.20	12.87	12.29	13.23	12.62	13.36	**13.57**
tidybot	12.92	12.58	8.40	10.94	11.09	10.98	10.69	**13.56**
transport	8.98	10.86	10.32	**10.97**	10.32	10.65	7.67	9.66
visitall	14.02	**19.90**	16.90	**19.90**	**19.90**	18.90	**19.90**	16.90
woodworking	14.60	15.47	15.81	**15.82**	15.26	15.67	13.90	15.76
Sum	176.45	183.79	178.35	187.73	**189.02**	186.03	184.38	182.54

Table 9: The qualities of our portfolios (right) compared to LAMA (left) on the IPC 2011 sequential satisficing track domains with a time limit of 3 minutes. Best planners in each domain are highlighted in bold.

Quality	**LAMA**	**Stone Soup**	**Uniform**	**Selector**	**Cluster**	**Inc. Time Limit**	**DW**	**RIS**
	2011	g=10			k=13	t=1s		
barman	17.49	13.20	18.48	13.71	13.13	**18.61**	18.60	13.31
elevators	10.67	**11.30**	9.71	10.12	9.52	10.99	11.16	11.05
floortile	4.13	3.28	**4.26**	3.89	4.07	3.82	3.88	3.28
nomystery	9.70	**17.67**	17.63	16.69	16.66	15.60	16.62	**17.67**
openstacks	**16.20**	10.26	10.09	8.32	11.57	13.69	13.15	10.07
parcprinter	**19.95**	19.58	19.78	19.25	19.78	19.73	19.75	19.78
parking	8.09	3.55	6.28	7.38	8.71	11.52	**13.61**	7.23
pegsol	**19.41**	18.97	19.25	18.65	18.80	18.22	19.19	19.07
scanalyzer	17.53	17.49	16.58	17.05	16.54	17.70	**17.81**	17.38
sokoban	11.58	14.88	12.96	14.51	**14.89**	13.31	14.11	13.85
tidybot	13.02	14.29	13.33	12.52	14.27	**14.67**	13.41	13.98
transport	9.10	11.30	10.81	**11.78**	10.81	11.14	10.92	9.60
visitall	15.57	**19.90**	17.90	**19.90**	**19.90**	**19.90**	**19.90**	**19.90**
woodworking	14.61	15.68	**15.91**	15.90	15.59	15.70	15.70	15.90
Sum	187.06	191.38	192.98	189.67	194.23	204.61	**207.84**	192.06

Table 10: The qualities of our portfolios (right) compared to LAMA (left) on the IPC 2011 sequential satisficing track domains with a time limit of 5 minutes. Best planners in each domain are highlighted in bold.

Quality	**LAMA**	**Stone Soup**	**Uniform**	**Selector**	**Cluster**	**Inc. Time Limit**	**DW**	**RIS**
	2011	g=40			k=13	t=1s		
barman	17.49	13.61	**18.94**	18.66	18.69	18.91	18.63	18.75
elevators	10.67	14.71	12.91	13.85	13.32	**14.91**	12.84	12.34
floortile	4.81	5.03	5.16	4.76	5.18	4.91	4.88	**5.93**
nomystery	9.98	17.73	17.73	16.67	16.76	15.73	16.69	**17.75**
openstacks	**18.56**	10.58	14.31	12.41	13.52	14.35	13.48	14.53
parcprinter	**19.95**	19.36	19.79	19.74	19.54	19.87	19.79	19.80
parking	12.04	10.54	12.20	15.94	15.07	15.21	15.27	**16.69**
pegsol	**19.95**	19.22	19.46	19.13	19.13	19.15	19.47	19.40
scanalyzer	17.73	18.39	18.84	18.85	18.92	18.48	18.80	**19.47**
sokoban	14.99	16.28	16.31	15.72	**16.60**	16.41	15.73	14.82
tidybot	13.53	14.59	**16.02**	14.81	15.59	15.26	14.97	14.84
transport	9.84	14.26	13.35	16.49	**16.84**	16.80	16.73	15.53
visitall	15.63	**19.90**	**19.90**	**19.90**	**19.90**	**19.90**	**19.90**	**19.90**
woodworking	14.61	15.91	**15.92**	15.85	15.70	15.70	15.74	15.91
Sum	199.79	210.10	220.84	222.80	224.78	225.60	222.92	**225.66**

Table 11: The qualities of our portfolios (right) compared to LAMA (left) on the IPC 2011 sequential satisficing track domains with a time limit of 15 minutes. Best planners in each domain are highlighted in bold.

Quality	**LAMA**	**FD-Autotune**		**FD Stone Soup**		**Stone Soup**		**Uniform**	**Selector**	**Cluster**		**Inc. Time Limit**		**DW**	**RIS**
	2011	1	2	1	2	g=100	g=110			k=16	k=12	t=10s	t=5s		
Training Score	–	–	–	–	–	19.39	19.39	19.22	19.28	19.18	19.17	18.73	18.74	19.07	19.49
barman (20)	17.62	13.23	5.40	6.31	5.60	17.06	18.27	19.02	18.96	17.05	**19.14**	19.02	19.01	18.88	19.08
elevators (20)	11.11	10.63	14.37	11.82	13.48	15.18	15.71	15.27	14.96	15.70	15.80	15.25	14.90	15.30	**16.38**
floortile (20)	4.81	6.56	**8.87**	5.87	6.37	5.93	5.80	5.93	6.01	5.93	5.63	5.18	5.22	5.20	5.95
nomystery (20)	10.81	10.38	16.52	12.33	12.51	18.77	18.75	17.73	17.76	18.75	17.76	16.65	16.65	16.67	**18.82**
openstacks (20)	**18.71**	12.99	17.06	12.48	12.35	11.11	11.26	16.33	10.72	16.76	15.36	15.29	14.49	14.14	11.41
parcprinter (20)	19.51	19.48	13.61	19.12	18.17	19.56	19.54	19.88	18.82	**19.89**	19.74	19.87	19.86	19.88	**19.89**
parking (20)	11.48	3.74	3.49	15.23	14.85	17.16	**17.40**	16.89	16.98	17.05	16.20	15.57	15.57	13.32	14.79
pegsol (20)	**20.00**	19.28	19.72	17.04	14.61	19.50	19.38	19.54	19.38	19.64	19.29	19.19	19.19	19.53	19.43
scanalyzer (20)	17.89	16.71	15.87	18.65	17.38	19.12	18.37	19.19	18.92	19.00	18.75	18.77	18.83	19.25	**19.69**
sokoban (20)	16.68	17.16	10.51	17.31	15.55	18.59	18.55	17.26	17.37	17.35	17.25	18.26	18.22	**18.64**	17.55
tidybot (20)	14.13	13.86	12.51	14.69	14.63	15.08	15.03	**16.40**	15.87	16.21	14.49	15.60	15.58	15.29	15.88
transport (20)	12.64	9.48	8.49	9.29	9.59	15.94	15.88	17.55	**17.60**	16.00	17.11	16.74	16.95	17.12	15.58
visitall (20)	15.55	1.71	3.29	3.98	0.92	**19.90**	**19.90**	**19.90**	**19.90**	**19.90**	**19.90**	**19.90**	**19.90**	**19.90**	**19.90**
woodworking (20)	14.65	14.72	10.25	**19.99**	18.43	15.92	15.82	15.94	15.88	15.92	15.71	15.80	15.79	15.77	15.91
Sum (280)	205.59	169.94	159.94	184.12	174.45	228.82	229.67	**236.84**	229.14	235.17	232.15	231.10	230.17	228.89	230.28

Table 12: The qualities of our portfolios (right) compared to IPC 2011 competitors (left) on the IPC 2011 sequential satisficing track domains. All planners were allowed to use up to 30 minutes for each problem. Best planners in each domain are highlighted in bold. The first row shows the training scores of our portfolios.

Quality	**LAMA**	**FD-AT**		**FDSS**		**Stone Soup**		**Uniform**	**Selector**	**Cluster**		**Inc. Time L.**		**DW**	**RIS**
	2011	1	2	1	2	g=100	g=110			k=16	k=12	t=10s	t=5s		
barman (20)	**20**	14	8	7	6	18	19	**20**	**20**	18	**20**	**20**	**20**	**20**	**20**
elevators (20)	**20**	19	15	19	19	17	18	19	17	18	19	18	19	18	19
floortile (20)	6	8	**9**	7	7	7	7	7	7	7	7	6	6	6	7
nomystery (20)	11	11	17	13	13	**19**	**19**	18	18	**19**	18	17	17	17	**19**
openstacks (20)	**20**	17	18	15	15	14	14	19	14	**20**	18	18	17	17	14
parcprinter (20)	**20**	**20**	14	**20**	**20**	**20**	**20**	**20**	19	**20**	**20**	**20**	**20**	**20**	**20**
parking (20)	15	7	4	19	**20**	**20**	**20**	**20**	**20**	**20**	**20**	19	19	17	17
pegsol (20)	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**
scanalyzer (20)	**20**	**20**	17	**20**	**20**	**20**	19	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**
sokoban (20)	18	**19**	11	**19**	17	**19**	**19**	18	18	18	18	**19**	**19**	**19**	18
tidybot (20)	16	16	14	16	**17**	16	16	**17**	**17**	**17**	16	**17**	**17**	**17**	**17**
transport (20)	15	11	10	11	14	18	18	**20**	**20**	18	**20**	**20**	**20**	**20**	18
visitall (20)	**20**	2	4	4	3	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**
woodworking (20)	**20**	**20**	14	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**
Sum (280)	241	204	175	210	211	248	249	**258**	250	255	256	254	254	251	249

Table 13: The coverage (number of solved problems) of our portfolios (right) compared to IPC 2011 competitors (left) on the IPC 2011 sequential satisficing track domains. All planners were allowed to use up to 30 minutes for each problem. Best planners in each domain are highlighted in bold.

[2] Malte Helmert. A planning heuristic based on causal graph analysis. In Shlomo Zilberstein, Jana Koehler, and Sven Koenig, editors, *Proceedings of the Fourteenth International Conference on Automated Planning and Scheduling (ICAPS 2004)*, pages 161–170. AAAI Press, 2004.

[3] Malte Helmert and Héctor Geffner. Unifying the causal graph and additive heuristics. In Jussi Rintanen, Bernhard Nebel, J. Christopher Beck, and Eric Hansen, editors, *Proceedings of the Eighteenth International Conference on Automated Planning and Scheduling (ICAPS 2008)*, pages 140–147. AAAI Press, 2008.

[4] Jörg Hoffmann and Bernhard Nebel. The FF planning system: Fast plan generation through heuristic search. *Journal of Artificial Intelligence Research*, 14:253–302, 2001.

[5] Erez Karpas and Carmel Domshlak. Cost-optimal planning with landmarks. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI 2009)*, pages 1728–1733, 2009.

[6] Emil Keyder, Silvia Richter, and Malte Helmert. Sound and complete landmarks for and/or graphs. In Helder Coelho, Rudi Studer, and Michael Wooldridge, editors, *Proceedings of the 19th European Conference on Artificial Intelligence (ECAI 2010)*, pages 335–340. IOS Press, 2010.

[7] Silvia Richter and Malte Helmert. Preferred operators and deferred evaluation in satisficing planning. In Alfonso Gerevini, Adele Howe, Amedeo Cesta, and Ioannis Refanidis, editors, *Proceedings of the Nineteenth International Conference on Automated Planning and Scheduling (ICAPS 2009)*, pages 273–280. AAAI Press, 2009.

[8] Silvia Richter, Malte Helmert, and Matthias Westphal. Landmarks revisited. In *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence (AAAI 2008)*, pages 975–982. AAAI Press, 2008.

[9] Jendrik Seipp, Manuel Braun, Johannes Garimort, and Malte Helmert. Learning portfolios of automatically tuned planners. In *Proceedings of the Twenty-Second International Conference on Automated Planning and Scheduling (ICAPS 2012)*. AAAI Press, 2012.

Appendix

We report the configurations found in the different training domains in the form in which they are generated by our automated tool. In many cases, these could be expressed more simply by exploiting default values or omitting redundant options.

- airport:

```
--landmarks lmg=lm_rhw(only_causal_landmarks=false,
                       disjunctive_landmarks=true,
                       conjunctive_landmarks=true,
                       no_orders=false,lm_cost_type=2,cost_type=2)
--heuristic hCea=cea(cost_type=2)
--heuristic hLM,hFF=lm_ff_syn(lmg,admissible=true)
--search lazy(alt([single(hLM),single(hLM,pref_only=true),
                   single(hFF),single(hFF,pref_only=true),
                   single(hCea),single(hCea,pref_only=true)],
                  boost=0),
              preferred=[hLM,hCea],reopen_closed=false,cost_type=2)
```

- depot:

```
--landmarks lmg=lm_hm(only_causal_landmarks=false,
                      disjunctive_landmarks=true,
                      conjunctive_landmarks=true,
                      no_orders=false,m=1)
--heuristic hCea=cea(cost_type=2)
--heuristic hLM=lmcount(lmg,admissible=true)
--search lazy(alt([single(hLM),single(hLM,pref_only=true),
                   single(hCea),single(hCea,pref_only=true)],
                  boost=100),
              preferred=[hCea],reopen_closed=false,cost_type=2)
```

- driverlog:

```
--landmarks lmg=lm_rhw(only_causal_landmarks=false,
                       disjunctive_landmarks=true,
                       conjunctive_landmarks=true,
                       no_orders=true,lm_cost_type=2,cost_type=1)
--heuristic hLM,hFF=lm_ff_syn(lmg,admissible=true)
--search lazy(alt([single(hLM),single(hLM,pref_only=true),
                   single(hFF),single(hFF,pref_only=true)],
                  boost=0),
              preferred=[hLM],reopen_closed=false,cost_type=2)
```

- freecell:

```
--landmarks lmg=lm_hm(only_causal_landmarks=false,
                      disjunctive_landmarks=true,
                      conjunctive_landmarks=true,
                      no_orders=true,m=1)
--heuristic hLM=lmcount(lmg,admissible=true)
--search eager(single(sum([g(),weight(hLM,5)])),
               preferred=[],reopen_closed=true,pathmax=false,
               cost_type=1)
```

- grid:

```
--landmarks lmg=lm_rhw(only_causal_landmarks=false,
                       disjunctive_landmarks=true,
                       conjunctive_landmarks=true,
                       no_orders=true,lm_cost_type=1,cost_type=2)
--heuristic hCg=cg(cost_type=2)
--heuristic hLM,hFF=lm_ff_syn(lmg,admissible=true)
--search lazy(alt([single(sum([g(),weight(hLM,10)])),
                   single(sum([g(),weight(hLM,10)]),pref_only=true),
                   single(sum([g(),weight(hFF,10)])),
                   single(sum([g(),weight(hFF,10)]),pref_only=true),
                   single(sum([g(),weight(hCg,10)])),
                   single(sum([g(),weight(hCg,10)]),pref_only=true)],
                  boost=1000),
              preferred=[hLM,hCg],reopen_closed=false,cost_type=2)
```

- logistics-2000:

```
--heuristic hCea=cea(cost_type=1)
--search lazy(alt([single(sum([g(),weight(hCea,10)])),
                   single(sum([g(),weight(hCea,10)]),pref_only=true)],
                  boost=2000),
              preferred=[hCea],reopen_closed=false,cost_type=1)
```

- miconic-fulladl:

```
--heuristic hFF=ff(cost_type=1)
--search lazy(alt([single(hFF),single(hFF,pref_only=true)],
                  boost=5000),
              preferred=[hFF],reopen_closed=false,cost_type=1)
```

- mprime:

```
--heuristic hCea=cea(cost_type=2)
--search lazy(alt([single(sum([g(),weight(hCea,5)])),
                   single(sum([g(),weight(hCea,5)]),pref_only=true)],
                  boost=1000),
              preferred=[hCea],reopen_closed=false,cost_type=2)
```

- optical-telegraphs:

```
--heuristic hCg=cg(cost_type=1)
--heuristic hFF=ff(cost_type=1)
--search lazy(alt([single(sum([g(),weight(hFF,10)])),
                   single(sum([g(),weight(hFF,10)]),pref_only=true),
                   single(sum([g(),weight(hCg,10)])),
                   single(sum([g(),weight(hCg,10)]),pref_only=true)],
                  boost=100),
              preferred=[hCg],reopen_closed=false,cost_type=1)
```

- pathways:

```
--landmarks lmg=lm_hm(only_causal_landmarks=false,
                      disjunctive_landmarks=true,
                      conjunctive_landmarks=false,
                      no_orders=true,m=1,lm_cost_type=0,cost_type=2)
--heuristic hLM,hFF=lm_ff_syn(lmg,admissible=true)
--search lazy(alt([single(hLM),single(hLM,pref_only=true),
                   single(hFF),single(hFF,pref_only=true)],
                  boost=5000),
              preferred=[hLM],reopen_closed=false,cost_type=0)
```

- philosophers:

```
--heuristic hCg=cg(cost_type=2)
--search lazy(alt([single(sum([g(),weight(hCg,10)])),
                   single(sum([g(),weight(hCg,10)]),pref_only=true)],
                  boost=0),
              preferred=[hCg],reopen_closed=false,cost_type=2)
```

- pipesworld-no-tankage:

```
--landmarks lmg=lm_merged([lm_rhw(),lm_hm(m=1)],
                          only_causal_landmarks=false,
                          disjunctive_landmarks=false,
                          conjunctive_landmarks=true,
                          no_orders=false)
--heuristic hFF=ff(cost_type=0)
--heuristic hLM=lmcount(lmg,admissible=true)
--search lazy(alt([single(sum([g(),weight(hFF,10)])),
                   single(sum([g(),weight(hFF,10)]),pref_only=true),
                   single(sum([g(),weight(hLM,10)])),
                   single(sum([g(),weight(hLM,10)]),pref_only=true)],
                  boost=500),
              preferred=[hFF],reopen_closed=false,cost_type=2)
```

- pipesworld-tankage:

```
--heuristic hFF=ff(cost_type=1)
--search lazy(alt([single(sum([g(),weight(hFF,7)])),
                   single(sum([g(),weight(hFF,7)]),pref_only=true)],
                  boost=5000),
              preferred=[hFF],reopen_closed=false,cost_type=1)
```

- psr-large:

```
--heuristic hAdd=add(cost_type=0)
--search lazy(alt([single(hAdd),single(hAdd,pref_only=true)],
                  boost=0),
              preferred=[hAdd],reopen_closed=true,cost_type=0)
```

- rovers:

```
--landmarks lmg=lm_hm(only_causal_landmarks=false,
                      disjunctive_landmarks=true,
                      conjunctive_landmarks=true,
                      no_orders=false,m=1,lm_cost_type=2,cost_type=0)
--heuristic hLM,hFF=lm_ff_syn(lmg,admissible=true)
--search lazy(alt([tiebreaking([sum([g(),weight(hLM,10)]),hLM]),
                   tiebreaking([sum([g(),weight(hLM,10)]),hLM],
                               pref_only=true),
                   tiebreaking([sum([g(),weight(hFF,10)]),hFF]),
                   tiebreaking([sum([g(),weight(hFF,10)]),hFF],
                               pref_only=true)],
                  boost=200),
              preferred=[hLM],reopen_closed=true,cost_type=2)
```

- satellite:

```
--heuristic hCg=cg(cost_type=2)
--search lazy(alt([single(hCg),single(hCg,pref_only=true)],
                  boost=0),
              preferred=[hCg],reopen_closed=true,cost_type=2)
```

- schedule:

```
--landmarks lmg=lm_hm(only_causal_landmarks=false,
                      disjunctive_landmarks=true,
                      conjunctive_landmarks=false,
                      no_orders=true,m=1,lm_cost_type=1,cost_type=0)
--heuristic hLM,hFF=lm_ff_syn(lmg,admissible=true)
--search lazy(alt([single(hLM),single(hLM,pref_only=true),
                   single(hFF),single(hFF,pref_only=true)],
                  boost=1000),
              preferred=[hLM,hFF],reopen_closed=false,cost_type=1)
```

- storage:

```
--landmarks lmg=lm_hm(only_causal_landmarks=false,
                      disjunctive_landmarks=true,
                      conjunctive_landmarks=false,
                      no_orders=true,m=1)
--heuristic hCg=cg(cost_type=1)
--heuristic hLM=lmcount(lmg,admissible=true)
--search lazy(alt([single(hLM),single(hLM,pref_only=true),
                   single(hCg),single(hCg,pref_only=true)],
                  boost=0),
              preferred=[hCg],reopen_closed=false,cost_type=1)
```

- tpp:

```
--landmarks lmg=lm_hm(only_causal_landmarks=false,
                      disjunctive_landmarks=true,
                      conjunctive_landmarks=false,
                      no_orders=true,m=1,lm_cost_type=0,cost_type=2)
--heuristic hLM,hFF=lm_ff_syn(lmg,admissible=true)
--search lazy(alt([single(sum([g(),weight(hLM,10)])),
                   single(sum([g(),weight(hLM,10)]),pref_only=true),
                   single(sum([g(),weight(hFF,10)])),
                   single(sum([g(),weight(hFF,10)]),pref_only=true)],
                  boost=500),
              preferred=[hLM],reopen_closed=false,cost_type=0)
```

- trucks:

  ```
  --heuristic hFF=ff(cost_type=1)
  --search lazy(alt([single(sum([weight(g(),2),weight(hFF,3)])),
                     single(sum([weight(g(),2),weight(hFF,3)]),
                            pref_only=true)],
                    boost=5000),
                preferred=[hFF],reopen_closed=true,cost_type=1)
  ```

- zenotravel:

  ```
  --heuristic hCg=cg(cost_type=1)
  --search lazy(tiebreaking([sum([g(),weight(hCg,2)]),hCg]),
                preferred=[],reopen_closed=true,cost_type=1)
  ```