Local Search Topology: Implications for Planner Performance

Mark Roberts and Adele Howe
Computer Science Dept., Colorado State University
Fort Collins, Colorado 80523
mroberts, howe@cs.colostate.edu
http://www.cs.colostate.edu/meps

Abstract

Hoffmann's topological analysis of the h^+ and h^{FF}state spaces explained why Fast Forward dominated the performance on early planning domains (Hoffmann 2004). His taxonomy segmented domains according to the presence/size of local minima and dead-end class. Surprisingly, the taxonomy has not been used to explain the performance of other planners that use similar heuristics. In this paper, we extend this analysis in several ways: 1) We apply the taxonomy to 10 heuristic search (including FF-2.3) and 17 non-heuristic search planners to determine the extent to which it explains planner performance. 2) We test model scalability by examining a subset of challenging problems to determine the extent to which the topology explains performance. We conclude that topological analysis is a valuable tool in explaining the conditions under which h^+ is favored. Similar analyses of newer domains and planners could benefit the community and may lead to more informed application, extensions, or simplifications of either the heuristics or the planners that use them.

Introduction

Linking search topology analysis with algorithm performance has contributed to researchers' understanding of search algorithms and heuristics. It leads to insights such as *the search space for an oversubscribed scheduling application is dominated by large plateaus* (Barbulescu *et al.* 2006), *the search space of job shop scheduling is dominated by many small local minima* (Watson, Howe, & Whitley 2003), *the search space of SAT depends on the problem instance* (Frank, Cheeseman, & Stutz 1997) , and in domain-independent planning, *the search spaces are often trivial* (Hoffmann 2004). Researchers have used these insights to both construct simpler algorithms and explain algorithm behavior. In some cases, the explanations reveal how these algorithms yield such dramatic improvements while still guaranteeing bounded computation (Chen, Gomes, & Selman 2001; Hoos & Stutzle 2004).

Hoffmann analyzed *whether* the success of Fast Forward (FF) could be explained in terms of local search topology. Hoffmann's (Hoffmann 2004) partitions 20 domains from the first two International Planning Competitions (IPCs) along two axes: the presence of local minima and the type of dead-end; dead-ends are states from which the goal cannot be reached. He provided both empirical (Hoffmann 2001a) and theoretical (Hoffmann 2001b) analyses showing that many common benchmarks were easily solvable (some even linearly) by heuristic search planners because either they lacked significant minima or the heuristic easily addressed them (Hoffmann 2004). In more recent work, Hoffmann extended these results to the newer problems from IPC3 and IPC4 (Hoffmann 2005), but we focus on the earlier work as our starting point.

Yet, little is known about when particular heuristics are useful for *other* planners – especially in the context of newer problems – and we believe the taxonomy has not been sufficiently exploited as a tool for explaining planner performance beyond FF. Hoffmann's analysis is convincing in its findings for FF; there is good reason to believe it might extend to other planners. In this paper, we take preliminary steps to show that Hoffmann's model explains planner performance of other heuristic search planners that employ h^+ approximations. We address the following questions:

1. On problems too large to enumerate, is the performance of FF sensitive to the taxonomy axes?
2. Is the performance of h^+ and non-h^+ planners distinct regardless of the taxonomy?
3. Is the performance of h^+ planners (as a group) sensitive to the taxonomy axes?
4. Is the performance of non-h^+ planners insensitive to the taxonomy axes?

We explore these questions using a hypothesis driven experiment design. We also use two data sets that are intended to give us some information about whether the taxonomy is also sensitive to problem difficulty.

Experiment Design

We apply Hoffmann's two-dimensional taxonomy to additional planners on the original 20 domains. Table 1 shows the 27 planners we included for this study; the heuristic search planners in bold are those that directly use an approximation of the h^+ heuristic for state-space search. Our planners represent the variety of planning technologies present from the International Planning Competitions – SAT-based, POCL-based, Heuristic Search, and model checking.

We have reproduced Hoffmann's taxonomy as a flat table in Table 2 (right-most two columns). Domains either

altalt-1.0	blackbox-4.2	cpt-1.0
ff-2.3	**hsp-2.0**	**hsp-2.0-b-h1plus**
hsp-2.0-b-H2Max	ipp-4.0	ipp-4.1
lpg-1.1	**lpg-1.2**	**lpg-td-1.0**
metric-ff_2002	mips-3	optop-1.6.19
prodigy-4.0	r-1_1	sapa-2.200406
satplan-2006	sgp-1.0b	sgp-1.0h
sgplan-2006	simplanner-2.0	snlp-1.0
stan-3	ucpop-4.1	vhpop-2.2

Table 1: The 27 publicly available planners we ran in our experiments; planners in bold use some version of the h^+ heuristic.

Domain	Problem Sets			Labels	
(Symbol)	H	*noH*	*mO1*	mType	dType
Assembly (AS)	135	-	-	M1	DU
Blks-3op (B3)	150	-	-	M0	DC
Blks-4op (B4)	150	103	78	M1	DC
Briefcase(BC)	115	3	-	M0	DC
Ferry (FY)	120	3	-	MX	DC
Freecell (FC)	110	100	98	M1	DU
Fridge (FR)	12	4	-	M0	DC
Grid (GD)	139	5	5	M0	DH
Gripper (GP)	7	20	17	MX	DC
Hanoi (HN)	3	-	2	M0	DC
Logistic (LG)	101	73	29	MX	DC
Mic-ADL (MA)	107	150	-	M1	DC
Mic-SIM (MC)	113	-	-	MX	DH
Mic-Str (MS)	101	150	-	MX	DH
Movie (MV)	30	31	-	MX	DH
Mystery (MY)	112	72	38	M1	DU
M-prime (MP)	107	35	23	M1	DU
Schedule (SD)	191	150	-	M1	DR
Tyreworld(TY)	1	11	-	MX	DH
TSP (TS)	8	-	-	MX	DH
TOTAL	1814	910	290		

Table 2: A flattened representation of the domains and problems we examine. The rightmost columns show the placement of each domain into the appropriate ordered mType $\{$MX, M0, M1$\}$ and dType $\{$DC,DH,DR,DU$\}$. The second through fourth columns list number of problems each domain contributes to the problem sets. Note that *mO1* is a subset of *noH*.

have local minima (M1) or do not (M0) and some domains that lack minima also have benches with a median exit distance less than a constant (MX). Along the dead-end axis the topology divides domains into the presence of dead-ends: If dead-ends do not exist the transition graph is either undirected (DC) or directed but harmless (DH). When dead ends exist they are heuristically recognized (DR) or heuristically unrecognized (DU). It is important that the ordering of the taxonomy categories listed in the caption implies that problems in the DU:M1 pairing are among the most challenging while those in the lowest pairing DC:MX are among the most simple.

Planners and Problems

We collected all the publicly available problems we could for these 20 domains; a total of 2724 problems came from the IPC1,IPC2, IPC3, and UCPOP benchmarks as well as Hoffmann's study (Hoffmann 2004). These problems are split into three subsets shown in Figure 2. Hoffmann's original problems are in the 'H' column and we exclude them for any analysis in this paper because Hoffmann previously analyzed them. The *noH* problems exclude all 'H' problems but include any other problem instance for the domain we could locate. The median over one second data in the *mO1* column attempts to focus our attention on the challenging problem instances: an instance is included if it took more than half the 27 planners at least a second to complete (fail or solve)[1]. To collect our data, we run each planner on each problem instance and save runtime and success.

Method

To answer our questions, we statistically measure the effect of both topological axes in terms of ratio of success and running time. We judge a test significant if $p < 0.05$ and highly significant if $p \ll 0.001$. All analyses are performed using R statistical package (R Development Core Team 2006) plus a custom G-test package[2]. We will explain these tests as we apply them in the following section.

Empirical Analyses

Is FF-2.3's performance dependent on the taxonomy axes for larger problems?

Hoffmann's analysis required exhaustive enumeration of the state spaces and so was restricted to small problem instances. We used the domain taxonomy constructed to assess whether larger instances conform to the expectations derived from the smaller.

To test whether there is an impact for success ratio (the dependent variable), we produce a contingency table of successes and failures grouped by the category of interest (independent variable) then perform a G-test[3]. A significant G-test indicates the likely presence of an effect other than chance and can be interpreted that taxonomy category effectively predicts success. We report both the G-value and p-value for each test.

We begin with instances from *noH*; a summary of the groupings appears:

G-test and Contingency Table of FF-2.3 for *noH*

mType				dType		
$G = 0.58, p = 0.748$			A	$G = 55.81, p \ll 0.001$		
	Succeed	Fail			Succeed	Fail
MX	249	39		DC	177	29
M0	10	2		DH	159	38
M1	516	94		DR	150	0
				DU	289	68

The top row of the contingency table shows the G-test results for each axis type on all problems (signified by the 'A' in the middle column). Thus, for the *noH* data success is highly dependent on dType ($p \ll 0.001$) but not on mType

[1] Upon archived publication, we intend to provide public access to our data sets and analysis code.

[2] We used Peter Hurd's g.test code available at http://www.psych.ualberta.ca/˜phurd/cruft/

[3] The G-test is the exact version of the more familiar approximation called the χ^2 test; the χ^2 was designed to overcome (now outdated) limitations of hand calculating the log-likelihood.

($p = 0.75$). This result suggests that performance on the *noH* problems is impacted by the dead-end class much more so than the presence of local minima.

We also examined the runtime of FF-2.3 for these problem instances by grouping the data by each type and performing several tests. The results for runtime analysis of FF-2.3 on *noH* are summarized in the table:

ANOVA and Pairwise comparisons on FF-2.3 for *noH*

mType					dType				
$F = 11.43, p \ll 0.001$				TTC	$F = 20.47, p \ll 0.001$				
$F = 1.98, p = 0.139$				TTS	$F = 4.96, p = 0.002$				
$F = 13.36, p \ll 0.001$				TTF	$F = 26.518, p \ll 0.001$				
	MX	M0	M1			DC	DH	DR	DU
MX	–	cs	s		DC	–	cs	cs	
M0	cf	–	cs		DH	cf	–	cs	
M1		cf	–		DR	c	c	–	
					DU				–

We start with a one-way analysis-of-variance (ANOVA) test on a linear model of the data (this is a common and fairly robust assumption). For this test, we report the F-statistic and the p-value as seen in the top three rows where we have split the data into time-to-completion (TTC), time-to-success (TTS), and time-to-failure (TTF); space limitations prohibit us from including the full ANOVA table for each of the six entries. A significant ANOVA implies a true difference over *all* groupings and justifies a pair-wise comparison.

For pair-wise comparisons, we use the TukeyHSD test because it is conservative in assigning grouping similarities and because it is standard to our statistical software; this test is reported as p-value adjusted to control the experiment-wise error at $\alpha = 0.05$. An insignificant TukeyHSD pair-wise comparison indicates that the two groupings are significantly *similar*. We show a summary of the pairwise comparisons of the groups for TTS ('s') in the upper triangle of each sub-figure, TTF ('f') in the lower triangle, and TTC ('c') in both. A letter for a pairing indicates that TukeyHSD **did not** find a significant difference between them – it means they are statistically similar. In the cases where the ANOVA does not yield a significant difference, we completely fill in the appropriate part of the table for all pairwise comparisons; as seen in the upper triangle of the mType box for TTS.

From the table, we see that all ANOVA results are highly significant except mType on TTS. In the mType, there is little to distinguish between the groupings for TTC, TTS, and TTF. Only one pair, MX:M1, is distinct for TTF (lower triangle). For TTS on both types (upper triangle), we note that neither taxonomy axis significantly explains performance with one key exception: DU is always viewed as a distinct group from the others. This finding validates Hoffmann's conjecture (Hoffmann 2004) that problems in the DU grouping are more challenging for FF. It also suggests that the presence of local minima has less impact than the dead-end class on FF-2.3 performance for these problems.

The more challenging problems from *mO1* produce different results. The results are constructed:

G-test and Contingency Table of FF-2.3 for *mO1*

mType				dType		
$G = 24.92, p \ll 0.001$			A	$G = 2.38, p = 0.30$		
	Succeed	Fail			Succeed	Fail
MX	46	0		DC	100	26
M0	7	0		DH	5	0
M1	182	55		DU	130	29

The effect that each axis has on success ratio using a G-test shows that the mType has a highly significant effect ($p \ll 0.001$), while the dType does not have a significant effect ($p = 0.30$). The result reverses what we saw for the *noH* problems above and could suggest that the predictive power of the taxonomy is dependent upon problem size. The ANOVA on TTS was insignificant for both dType ($p = 0.08$) and mType ($p = 0.55$); there was no data for failures as seen in the contingency table.

These results suggest that the taxonomy is useful for predicting success of *mO1* but not for predicting runtime. The results also suggest that the presence of local minima has more significant impact for challenging problems but that the dead-end class has more impact on performance for simpler problems. The finding is not terribly groundbreaking since we know that FF uses enforced hill-climbing with restarts and that its heuristic is an approximation of h^+. After failing to find a solution using enforced hill-climbing, FF-2.3 switches to best first search. We may need to consider the problems on which FF switches to best-first search to understand if switching confounds the findings.

In summary, we demonstrated our methodology and showed that the actual performance of FF-2.3 corroborates the model with one caveat: it appears that the effect of the mType or dType may depend upon problem difficulty. We showed that challenging problems are sensitive to the presence of local minima, while less challenging problems are sensitive to the dead-end class.

Is the performance of h^+ planners and non-h^+ planners distinct?

Before we examine if the taxonomy explains the groups of h^+ and non-h^+ planners, we need to examine whether these groups are distinct. To assess this question, we compare the 10 h^+ planners against non-h^+ planners on the *noH* problems. For success ratio, the G-test comparing for the same grouping of the 10 h^+ planners against the non-h^+ control was highly significant ($G = 3655.92, p \ll 0.001$), which simply validates the results of the recent competitions. We perform an ANOVA of the runtime of the 10 planners against a control group consisting of the other 17 non-h^+ planners. The ANOVA is significant ($F = 81.57, p \ll 0.0001$) justifying a pair-wise comparison. In comparing each h^+ planner to the non-h^+ control, we note that only HSP-2.0r using the H2Max heuristic ($p = 0.51$) had runtimes that were not significantly different from the control.

Viewing the runtime data in terms of TTS or TTF yields a slightly different picture. For TTS, the ANOVA is still significant ($F = 13.69, p \ll 0.0001$) justifying a pair-wise comparison. Only four planners **did** perform significantly different from the non-h^+ control: FF-2.3 ($p = 0.005$) HSP-2.0 ($p = 0$), LPG-1.1 ($p = 0$), MetricFF ($p = 0.015$). For TTF, the ANOVA is significant ($F = 115.43, p \ll 0.0001$) and three planners **did not** perform significantly differently from the non-h^+ control: FF-2.3 ($p = 0.999$), HSP-2.0r using H2Max heuristic ($p = 1.0$), and LPG-1.2 ($p = 0.998$).

In summary, for TTC, the results support the hypothesis of a significant difference between the h^+ planners and the non-h^+ planners for runtime and success ratio. We see differences when we look only at TTF or TTS; primarily the data seem to support the hypothesis that the h^+ planners have successful runtimes that appear very similar to non-h^+ planners but have failure runtimes that are different. The result on successes could be a floor effect in that many problems are solvable regardless of the planning architecture.

Are the h^+ planners distinguished from each other? If all the h^+ planners perform like one another in contrast to non-h^+ planners, it seems possible that there is no significant performance difference between the h^+ planners. We examine this question using the same method as the previous section but without the non-h^+ control. For success ratio, the G-test is highly significant ($G = 2729.414, p \ll 0.001$). For runtime on TTC, the ANOVA is significant ($F = 76.32, p \ll 0.001$); a pair-wise comparison shows the following pairs of planners are not significantly different from each other:

Planner	Planner	p
hsp-2.0-b-H2Max	hsp-2.0	0.133
sgplan-2006	metric-ff(2002)	0.286
sgplan-2006	lpg-td-1.0	0.336
lpg-1.2	hsp-2.0-b-h1plus	0.531
lpg-td-1.0	hsp-2.0-b-h1plus	0.685
metric-ff(2002)	altalt-1.0	0.772
lpg-1.1	hsp-2.0-b-h1plus	0.871
metric-ff(2002)	ff-2.3	0.908
sgplan-2006	ff-2.3	0.992
lpg-1.2	lpg-1.1	0.999

Some of the similarities make sense in light of planner subfamilies. For example, the FF, LPG, HSP planners would each comprise a subfamily. Other similarities are likely the cause of similar underlying engine code. For example, in the case of SGPlan-2006, the performance similarity to the FF subfamily is to be expected since SGPlan-2006 uses Metric-FF as its primary search engine. This only remaining pairs are the similarity between LPG and other planners We plan to examine this similarity further. We also plan to detail how much the planners overlap on specific problems.

For TTS, the ANOVA was significant ($F = 17.226, p \ll 0.001$) and 17 of the pairs were significantly different from each other; TTF had a significant ANOVA ($F = 50.19, p \ll 0.001$) with 15 pairs being similar.

We have shown that many of the 10 h^+ planners on all data have distinct performance from one another with minor surprises. When viewed by success or failure the differences start to disappear, though still more than half of the pairwise comparisons remain significant.

Do the dimensions in the taxonomy explain performance of the h^+ planners as a group?

To assess this hypothesis, we focus on the h^+ planners and measure the impact that the topology has on performance. We group all problems attempted by the ten planners according to the dType and mType then perform our statistical analysis.

G-test and Contingency Table of h^+ planners for *noH*

mType				dType		
$G = 379.48, p \ll 0.001$			A	$G = 577.94, p \ll 0.001$		
	Succeed	Fail			Succeed	Fail
MX	1536	1344		DC	1190	870
M0	73	47		DH	878	1092
M1	1975	4125		DR	322	1178
				DU	1194	2376

ANOVA and Pairwise comparisons on h^+ planners for *noH*

mType					dType				
$F = 199.38, p \ll 0.001$				TTC	$F = 126.35, p \ll 0.001$				
$F = 47.63, p < 0.001$				TTS	$F = 19.38, p \ll 0.001$				
$F = 112.97, p \ll 0.001$				TTF	$F = 172.84, p \ll 0.001$				
	MX	M0	M1			DC	DH	DR	DU
MX	–	c	s		DC	–		s	c
M0	cf	–			DH		–	s	
M1		f	–		DR			–	
					DU	c			–

As can be seen above, the G-test comparing success ratio was highly significant for both mType and dType. All ANOVA results for runtime are highly significant regardless of the data subset. In the mType, we note that the M0 and MX groups are similar for TTC and TTF data. Data subsets in both types show that the most distant pairs (MX:M1 and DC:DU) are similar; TTF further supports a distinction of the all dType pairs and the MX:M1 pair. These results suggest that for mType the taxonomy is less predictive (at the extremes) for TTS but still predictive for TTF. For dType the taxonomy is quite useful for predicting TTF and many pairs for TTS and TTC but not very useful for predicting the extremes on TTC.

The more challenging problems from *mO1* show a significant G-test for both types. All ANOVAs were significant except for TTS on dType, which suggests that predicting successful performance does not depend on dType. Pairwise comparisons for mType show that the taxonomy extremes are still predictive.

G-test and Contingency Table of h^+ planners for *mO1*

mType				dType		
$G = 148.81, p \ll 0.001$			A	$G = 23.04, p \ll 0.001$		
	Succeed	Fail			Succeed	Fail
MX	313	147		DC	585	675
M0	47	23		DH	36	14
M1	921	1449		DR	–	–
				DU	660	930

ANOVA and Pairwise comparisons on h^+ planners for *mO1*

mType					dType				
$F = 80.49, p \ll 0.001$				TTC	$F = 17.39, p \ll 0.001$				
$F = 18.22, p \ll 0.001$				TTS	$F = 1.158, p = 0.3144$				
$F = 27.94, p \ll 0.001$				TTF	$F = 10.54, p \ll 0.001$				
	MX	M0	M1			DC	DH	DR	DU
MX	–	cs			DC	–	cs	-	s
M0	cf	–	s		DH	cf	–	-	s
M1		f	–		DR	-	-	–	s
					DU		f		–

We have seen that at least some portions of the taxonomy remain useful for predicting TTS, TTF, and TTC. For the *noH* problems, both dType and mType are predictive of TTF and it appeared that mType was still slightly more predictive than dType. For the *mO1* problems, dType was not predictive at all for TTS and marginally predictive for TTF, while mType was still able to distinguish performance at the extremes. In comparison to the above results for FF-2.3, these results are certainly more mixed. Although all the h^+ planners use a similar heuristic, each planner searches in a distinct way. It may be possible to account for more variance by grouping the planners according to subfamilies based on search algorithm or other planner features.

Do the dimensions in topology explain performance of the non-h^+ planners as a group?

Given the previous results, we do not expect that the taxonomy will be informative for explaining the performance of non-h^+ planners. To confirm this intuition, we group the non-h^+ planners to determine if performance is sensitive to the taxonomy. The results for *noH* are summarized:

G-test and Contingency Table of non-h^+ planners for *noH*

mType				dType		
$G = 636.64, p \ll 0.001$			A	$G = 904.34, p \ll 0.001$		
	Succeed	Fail			Succeed	Fail
MX	1484	3412		DC	859	2643
M0	58	146		DH	1061	2288
M1	1347	9023		DR	132	2418
				DU	837	5232

ANOVA and Pairwise comparisons on non-h^+ planners for *noH*

mType					dType				
$F = 14.59, p \ll 0.001$				TTC	$F = 110.00, p \ll 0.001$				
$F = 0.56, p = 0.57$				TTS	$F = 2.56, p = 0.053$				
$F = 76.96, p \ll 0.001$				TTF	$F = 195.47, p \ll 0.001$				
	MX	M0	M1			DC	DH	DR	DU
MX	–	cs	s		DC	–	s	s	s
M0	cf	–	cs		DH		–	s	cs
M1		cf	–		DR			–	s
					DU		a		–

The G-test is significant for both dType and mType; note that both of the most challenging categories (M1 and DU) are the least likely to show success. On runtime for dType, the ANOVA is significant ($p \ll 0.001$) on TTC. TTF reveals a highly significant ANOVA ($p \ll 0.001$) with all pairwise comparisons being significantly different. TTS had an insignificant ANOVA ($p = 0.05$) so pair-wise comparison is unjustified. These results suggest that the difference in performance for successful runs is not significantly impacted by the dType model but that the failures can be explained by the dType. But the results are mitigated by the fact that some of these planners do in fact use the h^+ heuristic to control search (for example, VHPOP and some versions of MIPS) or construct a problem representation based on the relaxed plangraph (for example, SATPlan).

For mType, we see that the extremes MX and M1 lead toward explaining performance; except for success. On TTC the ANOVA is significant ($p \ll 0.001$) with two insignificant comparisons: M1:M0 and MX:M0. TTF also had a significant ANOVA ($p \ll 0.001$) and the same two insignificant comparisons. TTS had an insignificant ANOVA ($p = 0.57$). These results suggest – counter to the intuition – that the mType has impact on performance for failures suggesting the taxonomy is useful for predicting TTF but not very useful otherwise.

On the more challenging problems in *mO1*, neither G-test was significant, which suggests that the taxonomy is useless for predicting success on these problems. The remaining results for TTF, TTS, and TTC look very similar to those for *noH* so we skip detailed presentation of the results.

In summary, we can see that the taxonomy was very useful for predicting TTF for both problem sets but that it is not very useful in other ways. Even in comparison to the mixed results we saw for the group of h^+ planners our results are not a very compelling for extensions of this work to non-h^+ planners with the exception of predicting TTF. We may need to examine more closely planner subfamilies in this group to get a better sense of whether the observation is valid for individual non-h^+ planners.

Limitations

The limitations of this work center on the planners, problems, and methodology and cause us to view our results as suggestive rather than definitive.

In the case of the planners, it is clear that the planning systems are complex and not designed with large-scale comparisons in mind. Some planners try one approach for a specified time then switch approaches. Such switching algorithms could disperse the performance one might see from a pure implementation of a single algorithm and single heuristic. We also did not control for a difference between optimal and satisficing planners.

With respect to the problems, there are several mitigating factors that could confound our findings. Intra-domain problem difficulty is hard to assess, but there may be an effect due to the existence of simple (or challenging) problems in one or more domains (for example, from a single grouping) that led to success or failure. This was partly what the *mO1* data set was designed to help alleviate. But much more work could be done in examining specific problem instances.

We did control for at least one simple kind of error: We examined the problems for which there was a failure and found that all problems were solvable by at least one planner and that these failures resulted from an actual failure to solve the instance rather than a syntax or other error.

Another kind of limitation derives from our methodology. We began our analysis at a coarse level, preferring to answer questions about larger groupings of planners. This limits the inference one can draw for specific planners, though our findings do suggest it will be worthwhile to examine individual planners from each grouping.

In some places, low cell counts limited our inference. For example, in the first contingency table for FF-2.3, there were only 5 entries for M0 and 7 entries for DH and the failure entry was zero in MX. The missing/sparse data limits our inferences but still allows us to ask whether the axes impact performance on the larger extremes of the taxonomy.

Summary

Hoffmann studied problem instances from 20 domains to determine whether performance of FF was sensitive to a two-dimensional taxonomy: the presence of local minima and the dead-end classification. The scope of this early work was on *solvable* problems for which the computation of the NP-Complete h^+ heuristic was *tractable*; the problems therefore limit the findings. His research showed that the performance of FF was sensitive to the taxonomy axes; in particular, problems in the upper right (the DU:M1 pair) were among the most difficult.

We extended the work on the same 20 domains studied by Hoffmann in several ways. First, we applied the taxonomy to performance results for another 9 heuristic search planners. Second, we examined larger problems from the domains; these problems may have had a solution but remained unsolved by the planner in the allotted 30 minutes. Third, we segmented the problems into subsets based on problem difficulty. The *noH* problems consisted of any instance from the 20 domains that was not in the original set studied by Hoffmann. These problems were further split into the *mO1* problems, which were the set of problems for which at least half the planners took over one second to complete. With these two problem sets, we examined four questions.

For the question of FF-2.3's sensitivity to the taxonomy, we found that the performance depended on the taxonomy, but that the degree varied with problem difficulty. For the *noH* problems, FF-2.3 was sensitive to dType more so than mType, while for the *mO1* problems FF-2.3 was more sensitive to mType than dType.

In comparing the h^+ and non-h^+ planners, we showed that the performance of the individual h^+ planners is distinguished from the grouped non-h^+ planners regardless of the taxonomy. We also showed that, for the most part, the performance of h^+ planners differed from one another. Common similarities appeared to involve families or series of h^+ planners, though LPG was closely related to several other heuristic search planners.

With respect to sensitivity of the h^+ and non-h^+ planners to the taxonomy axes, we made several discoveries. For non-h^+ on both problem sets, we saw that taxonomy does not sufficiently predict performance with the exception of time-to-failure. For h^+, we saw mixed results that indicated that the taxonomy was useful for predicting particular categories on *noH* and *mO1*. In particular, we saw that TTF seemed somewhat easy to predict regardless of taxonomy type and we saw that mType was useful for separating the extreme categories. We conjectured that one reason we saw the mixed results was due to lumping the planners together.

Given these results we conclude that Hoffmann's taxonomy appears to be a useful tool for explaining the performance of other heuristic search planners using approximations of the h^+ heuristic. The taxonomy appears sensitive to problem difficulty and still shows some unaccounted for variance, both of which suggest that more work could be done to refine the taxonomy to control for these hidden factors. Not too surprisingly, the taxonomy does not appear compelling as a tool for explaining the successful performance of non-h^+ planners. In closing, we see some potential areas for extending this work:

1. Perform a similar analysis of the newer IPC problems analyzed in Hoffmann's recent article (Hoffmann 2005).
2. Further explore intra-domain differences (and problem difficulty) in the problem instance distributions. This is the most critical area to address in our future work.
3. Add more control for differences in planner technology; for example, optimal versus satisficing or planner family.
4. Characterize the performance on even more challenging problems by using the problem generators for these domains; we have already taken steps to incorporate these larger problems into our study.
5. Link this research with recent domain specific complexity analysis (Helmert 2006). We conjecture that doing so may account for additional variance within subgroups of these problems. Two-way ANOVA analysis may lead to further refinement of the taxonomy as it relates to the theoretical properties of domains.
6. Further characterize the behavior of new/existing h^+ planners and/or heuristics with respect to this taxonomy in the spirit of advancing our knowledge of when and why it is appropriate to select a particular planner or heuristic for a new problem.

Acknowledgments We wish to thank the anonymous reviewers for insightful and thought provoking discussion of this work as well as for suggestions for its presentation.

References

Barbulescu, L.; Howe, A.; Whitley, L.; and Roberts, M. 2006. Understanding algorithm performance on an oversubscribed scheduling application. *JAIR* 27:577–615.

Chen, H.; Gomes, C.; and Selman, B. 2001. Formal models of heavy-tailed behavior in combinatorial search. *Lecture Notes in Computer Science* 2239:408+.

Frank, J.; Cheeseman, P.; and Stutz, J. 1997. When gravity fails: Local search topology. *JAIR* 7:249–281.

Helmert, M. 2006. New complexity results for classical planning benchmarks. In *ICAPS 2006*, 52–61.

Hoffmann, J. 2001a. Local search topology in planning benchmarks: An empirical analysis. In *Proc. of 17th IJCAI*, 453–458.

Hoffmann, J. 2001b. Local search topology in planning benchmarks: A theorectical analysis. Technical Report 165, Albert Ludwigs University, Freiburg, Germany.

Hoffmann, J. 2004. *Utilizing Problem Structure in Planning: A local Search Approach*. Berlin, New York: Springer-Verlag.

Hoffmann, J. 2005. Where ignoring delete lists works: Local search topology in planning benchmarks. *JAIR* 24:685 – 758.

Hoos, H., and Stutzle, T. 2004. *Stochastic Local Search*. San Francisco, CA: Morgan Kaufmann.

R Development Core Team. 2006. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Watson, J.; Howe, A.; and Whitley, L. 2003. An analysis of iterated local search for job-shop scheduling. In *Proceedings of MIC-2003)*.