

# The 2014 International Planning Competition: Progress and Trends

*Mauro Vallati, Lukáš Chrpa, Marek Grześ,  
Thomas L. McCluskey, Mark Roberts, Scott Sanner*

■ *This article reviews the 2014 International Planning Competition (IPC-2014), the eighth in a series of competitions starting in 1998. IPC-2014 was held in three separate parts to assess the state of the art in three prominent areas of planning research: the deterministic (classical) part (IPCD), the learning part (IPCL), and the probabilistic part (IPPC). Each part evaluated planning systems in ways that pushed the edge of existing planner performance by introducing new challenges, novel tasks, or both. The competition surpassed again the number of competitors that participated in its predecessor, highlighting the competition's central role in shaping the landscape of ongoing developments in evaluating planning systems.*

Automated planning studies the problem of reasoning about actions to achieve goals or to maximize a reward. Actions are usually expressed in terms of preconditions and effects. Preconditions indicate the requirements that must hold to apply the action, while effects are the consequence (including the cost) of applying the action to the state of the world. Automated planning has been applied to diverse, real-world application areas such as space exploration, manufacturing, machine tool calibration, and road traffic management.

In order to foster the development and comparison of planning approaches, to assess the state of the art in planning, and to coordinate new challenging benchmarks, the International Planning Competition (IPC) has been organized every two or three years since 1998.

This article summarizes the eighth International Planning Competition held in 2014 (IPC-2014), which focused on

aspects that have a significant impact on planning and, in general, on the AI community: both identifying the emerging planning trends and describing the newly introduced domains, which have been designed for investigating the applicability of planning techniques to a range of real-world applications. More information about the competition, including complete results, source code of planning systems, and domain models, can be found on the IPC portal for all years and editions of the competition.<sup>1</sup> A summary of the history behind these parts can be found at in the article about the prior competition, IPC-2011 (Coles et al. 2012). Similar to IPC-2011, the competition was held in three distinct parts. The deterministic part (IPCD) has been running since IPC-1998 and is focused on fully observable environments where actions are atomic with deterministic effects, and planning is episodic. The learning part (IPCL) relaxes the episodic assumption to allow planners to learn from prior experience with the domain model or search process. The probabilistic part (IPPC) introduces episodic, cost-optimal problems with stochastic transitions, and (optionally) partial observability.

### Deterministic Part (IPCD-2014)

IPCD is the longest running part of the competition, and its evaluation tracks have evolved as the research community focused on new challenges. IPCD-2014 introduced three innovations: the agile track, a protocol for problem selection, and a unified planner submission system called DES.

The agile track evaluates how quickly planners solve challenging problems. The cutoff CPU time limit is 5 minutes compared to the usual limit of 30 minutes. This has been done because, in many planning applications, it might not be possible to wait a significant amount of time for having a plan to use; satisficing plans are required as soon as possible and, eventually, they can be optimized.

IPCD-2014 introduced a reproducible, unbiased and general protocol for efficient selection of testing problem instances.<sup>2</sup> Clearly, such a selection is an extremely critical step of every competition, since it is strongly related to performance and outcomes of planning engines. This approach relies on (1) generation of a large set of benchmarks; (2) execution of anonymized planning systems on generated instances; and (3) selection of suitable benchmarks. In particular, the third step aims to remove trivial and too complex planning problems.

Also, a new technique for submitting planners has been designed and exploited: the DES system. Such a system allowed participants to configure, compile, test, and submit their planning engines directly on the competition premises. This minimized the number of last-minute bugs, which are mainly because of compiling issues, and let teams understand the actu-

al performance of planners on the competition machines, which can significantly change on different hardware or software configurations (Howe and Dahlman 1993).

IPCD-2014 followed the evaluation formats of IPCD-2008 and IPCD-2011. The modeling language was the same as aforementioned editions of the competitions, but new core features required to be supported by planners were introduced: negative preconditions and conditional effects. This was done to foster features support and promoting planners that can be more appropriate to exploit in real-world planning applications. As in previous editions of the deterministic track, the VAL (Howey, Long, and Fox 2004) tool was used for validating the plans provided by planners. In all but the agile track the evaluation metrics of IPC-2008 and IPC-2011, which favor quality and coverage over problem-solving speed, were maintained. Briefly, each planner receives a score between 0 and 1 for each planning task. The score is the ratio between the quality of the solution found, if any (0 if no solution is provided), and the quality of the best solution found by any competitor. The score is summed across all problems for a given planner, and the winner is the planner with the highest score. Scores are not aggregated across tracks. In the agile track, a similar metric is used, but scores are given by evaluating run time. In this case, each planner receives a score between 0 and 1 for each planning task. The score is the ratio between the CPU time needed for solving the problem, 0 if no solution is provided, and the CPU time needed by the fastest competitor. In order to evaluate progress made in the planning area, results included a comparison to the winner of the last competition.

IPCD-2014 received a record number of submissions — 67 planners took part in the deterministic track alone. In the following sections we will discuss trends of the planning area that emerged from the competition, and the new domain models introduced.

### Results and Trends

IPCD-2014 held five tracks: agile (15 participants), multicore (9), optimal (17), satisficing (20) and temporal satisficing (6). Two tracks, the temporal optimal and preferences tracks, were cancelled due to too few competitors. We remark that few planners can deal with preferences or temporal models, which limits applicability of these systems in many real-world applications. We also note the number of entrants of the 2014 temporal satisficing track was lower than the corresponding 2011 track. It should be remarked that multicore solvers are not yet as well engineered as classical planners, a fact also observed at IPCD-2011 (Coles et al. 2012), though this may be due to the multicore track being recently added.

In the sequential satisficing track, IBaCoP2 (Cenamor, de la Rosa, and Fernández), which is a portfo-

lio-based approach combining a set of state-of-the-art planning engines, was declared as the winner and Mercury (Katz and Hoffmann), which uses best first search with partial delete-relaxation heuristics, was declared as the runner-up. In the sequential optimal track, SymBA\*-2 (Torralba, Alcázar, Borrajo, Kissmann, and Edelkamp), which uses bidirectional blind search with perimeter abstraction heuristics, was declared as the winner, and cGamer (Torralba, Alcázar, Kissmann, and Edelkamp), an extension of the Gamer planner (Kissmann and Edelkamp 2011) that won the IPC-2008, which uses bidirectional symbolic search, was declared as the runner-up. In the sequential agile track, YAHSP3 (Vidal), which performs a search embedding delete-relaxed heuristics, was declared as the winner, and Madagascar-pC (Rintanen), which translates planning problems into SAT, was declared as the runner-up. In the sequential multicore track, ArvandHerd (Valenzano, Nakhost, Müller, Schaeffer, and Sturtevant), which is a portfolio-based approach combining random-walk and best-first search-based planning, was declared as the winner, and IBaCoP (Cenamor, de la Rosa, and Fernández), a variant of IBaCoP2, was declared as the runner-up. In the sequential temporal track, YAHSP3-MT (Vidal), a variant of YAHSP3, was declared the winner, and Temporal Fast Downward (Eyerich, Keller, Aldinger, and Dornhege), an extension of Fast Downward (Helmert 2006), a well known heuristic search-based planner, was declared the runner-up.

As confirmed by the record number of participants and the good overall performance, many high-performance sequential planners have been developed. We believe this to be the result of the availability of well documented and supported planning platforms, such as FF (Hoffmann and Nebel 2001; Hoffmann 2003), Fast-Downward (Helmert 2006), and the very recent LAPKT,<sup>3</sup> which makes testing and developing new techniques easier. On the other hand, this leads to a large number of planning systems that are similar, and therefore share most of the weaknesses and strengths; for instance, 29 systems out of 67 are built on top of Fast-Downward (Helmert 2006).

To encourage the development of innovative planning techniques, and avoid the proliferation of too-similar planning systems, two special jury awards were given for the most innovative planners. Mercury (Katz and Hoffmann) is based on the red-black heuristic, which relaxes only some state variables in order to balance between taking advantage of delete-relaxation and mitigating its drawbacks (Katz, Hoffmann, and Domshlak 2013). RPT (Alcázar, Fernández, Borrajo, and Veloso) exploits rapidly exploring random trees (RRTs), which is a well known technique in path planning, in order to decompose planning tasks into much smaller subtasks that connect randomly generated (nonspurious) states (Alcázar, Veloso, and Borrajo 2011).

The newly introduced agile track evaluated plan-

ners on how quickly they found their first solution (solution quality was not considered). The speed of planners was not assessed in previous competitions, and the rationale behind introducing agile track was to encourage development of fast planners, since many application areas need solutions quickly. We found that planners usually performed very well in 2 or 3 domains, however, in other domains their performance was poor. On the other hand, most of the problems were solved by at least one of the planners.

Portfolio-based planning relies on combining different planning techniques for solving an instance; 29 out of 67 systems can be broadly classified as a portfolio planner. In IPCD-2011 portfolios were mainly static that is, they were configured once on a set of training instances and exploited on testing problems. In contrast, IPCD-2014 saw a significant number of portfolio-based entrants that exploit different approaches for either online or offline configuration. Portfolio-based planners performed very well in the satisficing track and the multicore track participants were all portfolio-based. On the other hand, they tended to underperform in the optimal and agile tracks, where combining techniques usually increases coverage at the expense of run-time performance.

In terms of progress made by planning engines, LAMA-11 (Richter and Westphal 2010),<sup>4</sup> the winner of the 2011 competition, would have placed 12th in the IPCD-2014 quality rankings, which reveals a strong trend of progress in the state of the art, larger than the trend we observed between IPCD-2011 and IPCD-2008 (Coles et al. 2012). To evaluate progress on run time, we considered LPG-td (Gerevini, Saetti, and Serina 2003; 2010) and Metric-FF (Hoffmann and Nebel 2001, Hoffmann 2003); they would have ranked respectively the 13th and the 17th position (out of 17) of the agile track. Interestingly, it is worth noting that the winner of the sequential temporal track, YAHSP3-MT, is not able to address concurrency, and it won due to its very good performance on a small number of domains.

## Domain Models Introduced

Since one of the aims of the competition is to evaluate the domain-independent performance of planners, and given the fact that planning techniques can be finely tuned on existing and known domains, it is of crucial importance that new challenging domains and problems be introduced at every edition. Moreover, competition is a good occasion for evaluating how state-of-the-art planners perform in potential application domains. In 2014, we introduced 9 new domains, out of 23 that have been used among all the tracks. Some of them were specifically designed for testing planners' ability in handling the required PDDL features, and had not directly lead to applications. However, several domains deal with relevant real-world problems. It is the case of traffic control,

which is modeled in three domain models: RTAM (Shah et al. 2013), in which planners have to deal with accident management, that is, planning actions to be taken by police, ambulances, and so on, and CityCar and MapAnalyser (Vallati and Chrapa), where planners are required to shape a map, in terms of road connections between junctions, for maximizing the flow of cars between different areas of the city. Other newly introduced domains are cave diving (Robinson, Muise, and Gretton), where a group of divers has to be organized in order to visit a number of underwater caves; maintenance (Rintanen), which schedules maintenance of aircrafts in airports; child's snack (de la Rosa and Fuentetaja), which creates sandwiches and serves them to a child; Tetris (Vallati), which models the well known *Tetris* game; hiking (McCluskey), which organizes hiking trips; and, finally, the genome edit distances domain (Haslum 2011), which encodes the problem of finding a minimum cost sequence of operations that transforms one genome (signed permutation of genes) into another.

It is worth noting that, for various reasons, a number of new domains were not suitable to be used in the competition. Domains that were submitted but not used in the competition included airport (Westerman and McCluskey), where automated planning is exploited for performing ground traffic control operations in airports; NRP (Piip and Ernits), which models the nurse scheduling problem in a hospital; pizza (de la Rosa and R. Fuentetaja), which cuts pizzas into slices and then serves them to customers; crisp (Koller and Hoffmann 2010), which describes the problem of sentence generation in natural languages; and calibration (Parkinson et al. 2014), which optimizes the calibration process of machine tools.

## Learning Part (IPCL-2014)

IPCL-2014 built upon the two previous learning parts to include a quality subtrack that employed the plan quality evaluation from the deterministic track. This track is designed to evaluate learning versus non-learning planners in the context of problems similar to the deterministic track. An integrated execution subtrack was proposed but did not have enough competitors to run.

The quality subtrack featured three awards: overall, best learner, and basic solver. The overall award included any approach fitting within the competition framework (CPU, memory, disk space), so there were few, if any, restrictions for this award. The hope in keeping the format similar to the IPCD is to facilitate direct comparison to IPCD results. The best learner award identifies the planner that most improved the difference in quality (that is, the learning delta) when learning was applied over when it was not. In numerous conversations leading up to the competition, there was no clear consensus on

how to assess the learning delta, and some felt the Pareto-optimal metric from IPCL-2011 confounded the two metrics and was too easily manipulated (that is, a competitor can artificially inflate the learning delta by failing during the no-knowledge runs). Thus, metrics were evaluated separately with an elimination round to ensure competitors could not artificially inflate the learning delta.

The basic solver award was introduced in IPCL-2014. While any definition of a basic solver can be perceived as somewhat arbitrary, the intent was to make a succinct definition available before the competition and take one small step toward encouraging fundamental research in basic solvers without limiting any approach. A basic solver is defined as any single (meta) algorithm that does not leverage more than one general-purpose solver at its core. This definition includes as a basic solver any meta-algorithmic approach (for example, iterated local search, iterated WA\*, managing calls to one SAT solver, randomized restarting A\*) provided the parameterized variants use the same core solver. To make this concrete we will use some examples: A core solver cannot itself be an ensemble of heterogeneous solvers. A fallback strategy that applies a different core solver is considered more than one core solver and is excluded as a basic solver. Simultaneously searching with different heuristics (with possibly distinct open lists) is no different in spirit than making iterated calls of WA\* with different weights and is included as a basic solver. A randomized restart algorithm that adjusts its restart strategy is included; iterated solvers that select (or adjust) parameters for a single base algorithm are similar in spirit to other iterated meta-algorithmic approaches. A planner that reencodes the task for a single core solver (while possibly learning to select among distinct encodings) is considered a basic solver. Competitors certified their planners as a basic solver subject to a final decision by the organizers.

Competitors were judged on six domains chosen from previous competitions: elevators, floortile, nomystery, parking, spanner, and transport. No new domains were submitted for this track, so the domains from the 2008 and 2011 deterministic and learning tracks were used. In order to use domains from these competitions, the language for the learning part of IPC-2014 was a restricted subset of PDDL 3.1. All solutions were validated with VAL (Howey, Long, and Fox 2004). A challenge in the learning competition is to generate problem instances that extend beyond nonlearning approaches, lie within reach of the anticipated performance of learning approaches, extend beyond the most capable learning planners, and draw the testing set from a similar distribution from the training set while avoiding results affected by overfitting. To mitigate the bias in selecting domains and in selecting the instance distributions, the domains were chosen to balance the

approaches that did well on them in previous competitions and instances were selected randomly from a range.

At the start of a six-week learning stage, competitors were provided generators for these domains, a representative set of validation problems, and guideline ranges for the evaluation distributions. Errors in the domains and generators were corrected at this point. After the learning stage was complete, competitors were provided runs from selected validation problems to ensure that their planner was performing as expected. Problems found in those runs were corrected before collecting the final results. For the final evaluation, 5 problems from each domain were randomly generated from the distributions, resulting in 30 problem instances. Competitors did not know these evaluation problems until after the results were released. The planners were run on the EC2 cloud compute platform with the support of a generous grant from Amazon Web Services; each compute platform had a compute equivalent of 2 cores and 3.75 GB memory and ran Ubuntu 12.04 LTS. To account for variations in the actual computing resources on the cloud platform, each planning system was run 30 times with and without domain knowledge on each problem instance. There are three categories of awards, each with a first, second, and third place.

Of the 14 planners from 8 teams that expressed an interest in competing for the quality track, 11 planners from 7 teams competed in the final evaluation. Five planners were registered as basic solvers that used only one algorithm to solve problems.

## Results and Trends

The overall best quality award compares planners on the quality of the best plan they produced for each problem. The awards for best overall quality (out of a possible score of 30) go to MIPlan (first place, quality: 21.88) from team members Nuñez, Borrajo, López; Fast Downward Cedalion (second place, quality: 19.98) from team members Seipp, Sievers, Hutter; and Fast Downward SMAC (third place, quality: 17.45) from team members Seipp, Sievers, and Hutter.

The best learner award compares planners on the learning delta between their overall improvement on plan quality when knowledge was applied over when it was not. To ensure that the baseline performance without knowledge was fair, any problem solved by seven or more (that is, half or more) planners was removed from this evaluation, resulting in 24 problem instances. No planners were eliminated from consideration, but the ranking did change slightly due to this adjustment. The awards for best learner go to Fast Downward Cedalion (first place, adjusted quality delta: 10.40), Eroller (second place, adjusted quality delta: 9.97) from team members de la Rosa and Fuentetaja; and Fast Downward SMAC (third place, adjusted quality delta: 9.18). The basic solver

award compares the quality of the best plan while using only a single core algorithm. The best core solver awards go to Fast Downward SMAC (first place, quality: 17.45); LLama (second place, quality: 14.30) from team members Virseda, Alcázar; and Eroller (third place, quality: 12.51).

Assessing the benefit of learning for classical planning remains a challenge. The elimination round seems to have partially resolved one issue in assessing a learning delta. However, there remains no consistent measurement of the computational effort invested during the learning stage, suggesting that teams with access to greater computational resources may be at an advantage. Further, it remains possible to win the overall award with a strong base system and a modest performance increase with learning. It seems worthwhile to continue discussions on how to reward learning in future competitions in spite of the challenges associated with defining such a metric.

Learning planners have advanced considerably since IPCL-2008 and IPCL-2011. The problem distributions extended the range of the previous competitions, and the most capable planners solved many of these harder problems with learning where they failed without learning. However, the lack of new domains for IPCL-2014 was unfortunate, and it will be illuminating to evaluate these systems on the new problems from IPCD-2014. It may also be worth considering in future competitions how to leverage new domains in both IPCD and IPCL so that the results are based on the same problems and can be more directly comparable.

Learning alone does not account for the best overall performance, although learning was important for ranking well in the overall award. For example, IBaCoP performed very well in IPCD-2014 and its companion learning planner, LIBaCoP, similarly placed well in the ranking before learning, but it was surpassed in the overall ranking by systems that had larger learning deltas. The best overall and best learner rankings differ, and the rankings changed substantially (sometimes for the worse) when learning was applied for all but the winning overall planner, MIPlan. For the 24 problems used to assess learning delta, less than half (5 of 11) of the systems improved the quality score by an average of more than 5 problems; 3 of those were basic solvers.

Basic solvers deserve greater attention. The best three basic solvers scored overall rankings of 3rd, 7th, and 8th while scoring 2nd, 3rd, and 4th in the best learner ranking. This ranking difference underscores a performance gap between ensembles and basic solvers, reveals that basic solvers still remain viable for improved learning, and strongly suggests it is worthwhile to highlight basic solvers in future competitions. Basic solvers, which usually excel on specific problem instances, are often the atomic building blocks of ensembles, which usually push the performance envelope across problem sets. Both

are necessary to advance planning and search; it is prudent to consider how we can encourage advancements in both. More broadly, separately awarding basic solvers (rather than penalising or eliminating ensemble systems) may also prove useful in other competitions facing concern over how to balance performance differences between ensemble systems and basic solvers.

## The Probabilistic Part (IPPC-2014)

IPPC-2014 continued with the competition format initiated in IPPC-2011 that provided tracks for both discrete MDP and POMDP problem specifications described in the Relational Dynamic influence Diagram Language (RDDL) language.<sup>5</sup> Continuous tracks of IPPC-2014 were also planned but not run due to a lack of competitors.

While probabilistic tracks dating back to the IPPC-2004 used a probabilistic extension of PDDL known as PPDDL (Younes et al. 2005), RDDL was introduced in IPPC-2011 as an alternative specification language to PPDDL to provide joint modeling of stochasticity, concurrency, and complex reward and transition structure not possible in (lifted) PPDDL. The use of RDDL in the IPPC-2011 marked the first time that complex stochastic domains such as traffic control and elevator control could be modeled. A slight extension of RDDL for the IPPC-2014 allowed the further modeling of real-world domains taken from the ecology literature. As in the IPPC-2011, IPPC-2014 provided translations of RDDL domains and problem instances to various alternative formats (for example, factored MDPs and POMDPs and grounded PPDDL) to facilitate participation by a wide range of competitors.

IPPC-2014 used the purely reward-based evaluation metric introduced in IPPC-2011 — for each of 10 problem instances and for each of eight problem domains (80 problem instances altogether), a planner was assigned a normalized  $[0, 1]$  score with the lower bound determined by the maximum average performance of a noop and random policy and the upper bound determined by the best competitor; any planner not competing or underperforming the lower bound was assigned a score of 0 and all normalized  $[0, 1]$  instance scores were averaged to arrive at a single final score for each planner. In contrast to the IPPC-2011, which only used a single time limit for completion of the entire competition (leading to highly varying amounts of time allocated to different problem instances), IPPC-2014 enforced a strict per-instance time limit of 18 minutes to complete all 30 trials. This timing change was made to ensure fair comparison of planner performance on a per instance basis.

Participation in IPPC-2014 offered a rematch of the top two competitors from the respective MDP and POMDP tracks of IPPC-2011 using variants of

Monte Carlo tree search (MCTS) and online value iteration. In addition, two new competitors joined the MDP track IPPC-2014. Overall results showed that the three year head-start of the previous competitors paid off in this competition with the top two places in each track unchanged from IPPC-2011; nonetheless, results on four of the domains common between IPPC-2011 and IPPC-2014 show that planners competing in both have demonstrated significant performance improvements since 2011. Later on in this article we will elaborate on general planning trends that can be inferred from these results along with a discussion of new domains that were introduced in IPPC-2014 to provide novel technical challenges and real-world applications.

## Results and Trends

With the change from PPDDL used in IPPCs 2004, 2006, and 2008 to RDDL used in IPPCs 2011 and 2014, the field of competitors and winning methods have shifted substantially. Given the derivation of PPDDL from its deterministic PDDL subset, high-performance planners for PPDDL in IPPCs 2004-2008 often relied on deterministic replanning methods that used an underlying PDDL planner — the first and quite successful of these approaches being FF-Replan (Yoon, Fern, and Givan 2007), which replanned on unexpected outcomes in a deterministic translation of PPDDL to PDDL. With the change to RDDL in IPPC-2011 that made heavy use of exogenous stochasticity, concurrency, and general reward (cost-optimal) objectives in a finite horizon setting, the winning approaches in IPPC-2011 shifted to variants of Monte Carlo tree search and in some cases, online value iteration.

The dominance of planners using Monte Carlo tree search continued in IPPC-2014 with revised entries from the same winning teams and runners-up as in 2011. For the MDP track, the winner was PROST-2014 (Keller and Geisser), which used a modified UCT approach (Keller and Helmert 2013); the runner-up was G-Pack (Kolobov, Mausam, and Weld), which used an iterative deepening version of labeled RTDP (Bonet and Geffner 2003) with sampled Bellman backups (Kolobov, Mausam, and Weld 2012). For the POMDP track, the winner was POMD-PX-NUS (Ye, Wu, Zhang, Hsu and Lee), which used a combination of heuristic search on a sampled sparse belief tree (Somani et al. 2013) along with UCT for POMDPs known as POMCP (Silver and Veness 2010); the runner-up was KAIST-AIPR-Lab (Han, Nam, Lee, and Kim), which also used POMCP in combination with symbolic heuristic search value iteration (Sim et al. 2008). All these planners showed improvement on their previous versions on the four domains common to both the 2011 and 2014 competitions; in some domains, the 2014 versions of planners halved the average cost of their 2011 versions, indicating substantial progress in the MDP and POMDP tracks

since IPC 2011. An innovation award was not offered in IPPC-2014 as has been done for other tracks, because the number of competitors is still small and diverse and does not (yet) suffer from a proliferation of planners based on a common code base.

To compare planners from IPPC-2008 to IPPC-2014, a variant of the triangle tireworld domain (Little and Thiébaux 2007) used in IPPC-2008 was translated to RDDDL for use in IPPC-2014. An analysis of results from both competitions suggests that IPPC-2008 replanners could solve larger problems, albeit somewhat suboptimally, while the IPPC-2014 Monte Carlo planners were closer to optimal on the problems they could solve, but could not scale to the largest problems. This inherently reflects the strengths and weaknesses of the two approaches. Triangle tireworld was intentionally designed to prove difficult for replanners since it required careful reasoning about the probabilities of different paths, hence the potential suboptimality of determinizing approaches that discard probabilities. On the other hand, Monte Carlo tree search approaches reason directly about expected costs, but in doing so must reason about a tree of contingency paths with varying probabilities in contrast to probability-free deterministic replanning. This observation suggests that a merger of Monte Carlo tree search and replanning techniques might provide more optimal planners for longer horizons; notably techniques such as RFF (Teichteil-Königsbuch, Kuter, and Infantes 2010), which won the fully observable uncertainty track of IPPC-2008, do offer some potential guarantees for replanning by incorporating Monte Carlo simulations, but so far RFF has only been evaluated in the goal-oriented setting of IPC 2008.

One final trend observed in 2014 was that an unprecedented half of the planners entered in IPPC-2014 used symbolic (decision diagram) representations. While these decision diagram-based planners did not take the top places, one of them — KAIST-AIPR-Lab — did achieve runner-up in the POMDP track. It remains to be determined whether any particular type of structure for MDP or POMDP domains can be identified where symbolic planners can definitively outperform nonsymbolic planners.

### Domain Models Introduced

A principle guiding the switch from PPDDL to RDDDL in IPPC-2011 was to enable the representation of rich planning domains with concurrency and exogenous stochastic events such as traffic control and elevator control that could not be represented in PPDDL. IPPC-2014 sought to continue this trend by introducing three new contributed domains of interest to the broader scientific community. The first two new domains were from ecology and respectively represent the problems of wildfire management (Karafyllidis and Thanailakis 1997) (that is, how to allocate firefighting resources to protect assets), contributed

by Zhenyu Yu; and invasive species management (Muneepeerakul et al. 2007) (that is, how to allocate personnel resources to limit the spread of the Tamarisk invasive plants in stream systems), as modeled by Dieterich, Taleghan, and Crowley (2013). A slight extension of RDDDL for IPPC-2014 allowed transition distribution parameters to be composed from exponentials and other elementary functions, which enabled the modeling of these two domains that used empirically derived transition models. A third domain, academic advising, contributed by Libby Ferland, represented an academic advisor's task of recommending courses for students to help them graduate as quickly as possible.

## Concluding Remarks

All tracks in IPC-2014 have shown significant advances in planning technology in the last three years, and in some cases, the techniques used by winning planners in particular tracks indicate a fundamental shift in the approach that appears to be most effective. Hence the IPC-2014 has served its purpose of both encouraging and tracking progress in the planning community by comparing state-of-the-art planning technologies in a controlled evaluation setting on domains of common interest to the community. As the AI and planning communities move forward to consider future IPCs and other similarly motivated competitions in AI in general, we remark on two imperatives that we believe future organizers should bear in mind.

First, we reinforce a remark from the IPC 2011 (Coles et al. 2012) that the competition languages and domains have profoundly influenced the direction of planning research dating back to the first IPC in 1998. Hence we believe it is imperative that future competition domains are chosen so as to maintain relevance not only to trends in the planning community but also to potential end users of planning technology. In many cases, modeling domains of interest to the broader research community may simply require elicitation from domain experts and formalization within the language constraints of existing tracks. In other cases this may require using more expressivity already available in existing languages or even novel language extensions to include, for example, object fluents (supported by both PDDL 3.1 and RDDDL but not used in existing competitions), constraints and timelines (supported by ANML [Smith, Frank, and Cushing 2008]), hierarchical decomposition (Erol, Hendler, and Nau 1994; Shivashankar et al. 2013), or more holistic views of the planning system (Ghallab, Nau, and Traverso 2014; Pollack and Horty 1999).

Second, we believe the adage of “what you measure is what improves” is critical to keep in mind for future competitions. Future organizers should thus consider whether the current evaluation metrics

encourage planning research that finds wider adoption in the AI community and beyond. For instance, organizers might consider different or richer quality metrics that explicitly trade off time and objective quality or perhaps even a change of setting, for example, to focus on real-time planning scenarios that limit online deliberation and require effective offline planning for a variety of scenarios that may be encountered at execution time. Furthermore, it would be useful if planner computational effort could be measured independently of its implementation in order to better compare planning paradigms, but no such measurement has been proposed to date.

In conclusion, we believe that as long as future IPCs make a concerted effort to remain relevant — to model domains of interest for real-world applications and to measure what matters — future IPCs can continue to drive forward rapid progress in planning research and its practical application.

### Acknowledgements

The organizers of IPCD-2014 would like to acknowledge the use of the University of Huddersfield Queensgate Grid in carrying out this work as well as Ibad Kureshi and John Brennan for their invaluable support with the cluster. The organizers of IPCL-2014 thank Colorado State University and the Naval Research Laboratory for providing compute resources; Mark Roberts was funded partially by the Office of Strategic Defense. The IPCL and IPPC were supported by a generous grant from Amazon Web Services allowing the competitions to run on the Elastic Compute Cloud.

### Notes

1. [pc.icaps-conference.org/](http://pc.icaps-conference.org/).
2. [helios.hud.ac.uk/scommv/IPC-14/selection.html](http://helios.hud.ac.uk/scommv/IPC-14/selection.html).
3. Available at [lapkt.org](http://lapkt.org).
4. See also LAMA 2008 and 2011 by S. Richter, M. Westphal, and M. Helmert, which appeared in a booklet distributed at the 7th International Planning Competition.
5. See Relational Dynamic Influence Diagram Language (RDDL): Language Description by S. Sanner ([users.cecs.anu.edu.au/~ssanner/IPPC2011/RDDL.pdf](http://users.cecs.anu.edu.au/~ssanner/IPPC2011/RDDL.pdf)).

### References

- Alcázar, V.; Veloso, M. M.; and Borrajo, D. 2011. Adapting a Rapidly-Exploring Random Tree for Automated Planning. In *Proceedings, the Fourth International Symposium on Combinatorial Search*, ed. D. Borrajo, M. Likhachev, and C. Linares Lopez. Palo Alto, CA: AAAI Press.
- Bonet, B., and Geffner, H. 2003. Labeled RTDP: Improving the Convergence of Real-Time Dynamic Programming. In *Proceedings of the 13th International Conference on Automated Planning and Scheduling (ICAPS-03)*, 12–21. Menlo Park, CA: AAAI Press.
- Coles, A. J.; Coles, A.; Olaya, A. G.; Celorrio, S. J.; López, C. L.; Sanner, S.; and Yoon, S. 2012. A Survey of the Seventh International Planning Competition. *AI Magazine* 33(1): 83–88.
- Dietterich, T. G.; Taleghan, M. A.; and Crowley, M. 2013. Pac Optimal Planning for Invasive Species Management: Improved Exploration for Reinforcement Learning from Simulator-Defined MDPS. In *Proceedings of the 27th National Conference on Artificial Intelligence (AAAI-13)*. Palo Alto, CA: AAAI Press.
- Erol, K.; Hendler, J. A.; and Nau, D. S. 1994. HTN planning: Complexity and expressivity. In *Proceedings of the 12th National Conference on Artificial Intelligence (AAAI)*, 1123–1128. Palo Alto, CA: AAAI Press.
- Gerevini, A.; Saetti, A.; and Serina, I. 2003. Planning through Stochastic Local Search and Temporal Action Graphs. *Journal of Artificial Intelligence Research (JAIR)* 20: 239–290.
- Gerevini, A.; Saetti, A.; and Serina, I. 2010. An Empirical Analysis of Some Heuristic Features for Planning Through Local Search and Action Graphs. *Fundamenta Informaticae* 105(2010): 1–31 doi: 10.3233/FI-1010-350.
- Ghallab, M.; Nau, D.; and Traverso, P. 2014. The Actor's View of Automated Planning and Acting: A Position Paper. *Artificial Intelligence* 208 (March): 1–17.
- Haslum, P. 2011. Computing Genome Edit Distances Using Domain-Independent Planning. Paper presented at the ICAPS 2011 Scheduling and Planning Applications Workshop (SPARK), 13 June, Freiburg, Germany.
- Helmert, M. 2006. The Fast Downward Planning System. *Journal of Artificial Intelligence Research (JAIR)* 26: 191–246.
- Hoffmann, J. 2003. The Metric-FF Planning System: Translating Ignoring Delete Lists to Numeric State Variables. *Journal of Artificial Intelligence Research (JAIR)* 20: 291–341.
- Hoffmann, J., and Nebel, B. 2001. The FF Planning System: Fast Plan Generation Through Heuristic Search. *Journal of Artificial Intelligence Research (JAIR)* 14: 253–302.
- Howe, A., and Dahlman, E. 1993. A Critical Assessment of Benchmark Comparison In Planning. *Journal of Artificial Intelligence Research (JAIR)* 1:1–15.
- Howey, R.; Long, D.; and Fox, M. 2004. Automatic Plan Validation, Continuous Effects and Mixed Initiative Planning Using PDDL. In *Proceedings of the Sixteenth IEEE International Conference on Tools with Artificial Intelligence*, 294–301. Piscataway, NJ: Institute for Electrical and Electronics Engineers.
- Karafyllidis, I., and Thanailakis, A. 1997. A Model for Predicting Forest Fire Spreading Using Gridular Automata. *Ecological Modelling* 99(1): 87–97.
- Katz, M.; Hoffmann, J.; and Domshlak, C. 2013. Red-Black Relaxed Plan Heuristics. In *Proceedings of the 27th AAAI Conference of the American Association for Artificial Intelligence (AAAI13)*, 489–495. Palo Alto, CA: AAAI Press.
- Keller, T., and Helmert, M. 2013. Trial-Based Heuristic Tree Search for Finite Horizon MDPs. In *Proceedings of the 23rd International Conference on Automated Planning and Scheduling (ICAPS-13)*, 135–143. Palo Alto, CA: AAAI Press.
- Kissmann, P., and Edelkamp, S. 2011. Improving Cost-Optimal Domain-Independent Symbolic Planning. In *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence (AAAI 2011)*. Palo Alto, CA: AAAI Press.
- Koller, A., and Hoffmann, J. 2010. Waking Up a Sleeping Rabbit: On Natural-Language Sentence Generation with FF. In *Proceedings of the 20th International Conference on Auto-*

*mated Planning and Scheduling* (ICAPS10), 238–241. Palo Alto, CA: AAAI Press.

Kolobov, A.; Mausam; and Weld, D. 2012. LRTDp Versus UCT for Online Probabilistic Planning. In *Proceedings of the 26th National Conference on Artificial Intelligence* (AAAI-12). Palo Alto, CA: AAAI Press.

Little, I., and Thiébaux, S. 2007. Probabilistic Planning Versus Replanning. Paper presented at the 2007 ICAPS Workshop International Planning Competition: Past, Present and Future. 23 September, Providence, Rhode Island.

Muneepeerakul, R.; Weitz, J. S.; Levin, S.; Rinaldo, A.; and Rodriguez-Iturbe, I. 2007. A Neutral Metapopulation Model of Biodiversity in River Networks. *Journal of Theoretical Biology* 245(2): 35–63.

Parkinson, S.; Longstaff, A. P.; Crampton, A.; and Gregory, P. 2014. Automated Planning for Multiobjective Machine Tool Calibration: Optimising Makespan and Measurement Uncertainty. In *Proceedings of the 24th International Conference on Automated Planning and Scheduling* (ICAPS14). Palo Alto, CA: AAAI Press

Pollack, M. E., and Horty, J. F. 1999. There's More to Life Than Making Plans: Plan Management in Dynamic, Multi-agent Environments. *AI Magazine* 20(4): 71.

Richter, S., and Westphal, M. 2010. The LAMA Planner: Guiding Cost-Based Anytime Planning with Landmarks. *Journal of Artificial Intelligence Research* (JAIR) 39: 127–177.

Shah, M. M. S.; Chrupa, L.; Kitchin, D. E.; McCluskey, T. L.; and Vallati, M. 2013. Exploring Knowledge Engineering Strategies in Designing and Modeling a Road Traffic Accident Management Domain. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence*. Palo Alto, CA: AAAI Press.

Shivashankar, V.; Alford, R.; Kuter, U.; and Nau, D. 2013. The GoDeL Planning System: A More Perfect Union of Domain-Independent and Hierarchical Planning. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, 2380–2386. Palo Alto, CA: AAAI Press.

Silver, D., and Veness, J. 2010. Monte-Carlo Planning in Large POMDPs. In *Proceedings of 24th Conference on Neural Information Processing Systems* (NIPS-10), 2164–2172. Red Hook, NY: Curran Associates, Inc.

Sim, H. S.; Kim, K.-E.; Kim, J. H.; Chang, D.-S.; and Koo, M.-W. 2008. Symbolic Heuristic Search Value Iteration for Factored POMDPs. In *Proceedings of the 23rd National Conference on Artificial Intelligence* (AAAI-08), 1088–1093. Menlo Park, CA: AAAI Press.

Smith, D. E.; Frank, J.; and Cushing, W. 2008. The ANML language. Paper presented at the ICAPS 2008 Workshop on Knowledge Engineering for Planning and Scheduling (KEPS). 14 September, Sydney, Australia.

Somani, A.; Ye, N.; Hsu, D.; and Lee, W. S. 2013. Despot: Online POMDP Planning with Regularization. Paper presented at 27th Conference on Neural Information Processing Systems (NIPS-13), 5–8 December, Lake Tahoe, Nevada.

Teichteil-Königsbuch, F.; Kuter, U.; and Infantes, G. 2010. Incremental Plan Aggregation for Generating Policies in MDPs. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems* (AAMAS-10): Volume 1, 1231–1238. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems.

Yoon, S.; Fern, A.; and Givan, R. 2007. FF-Replan: A Baseline for Probabilistic Planning. In *Proceedings of the 17th International Conference on Automated Planning and Scheduling*

(ICAPS-07), 352–359. Palo Alto, CA: AAAI Press.

Younes, H. L. S.; Littman, M. L.; Weissman, D.; and Asmuth, J. 2005. The First Probabilistic Track of the International Planning Competition. *Journal of Artificial Intelligence Research* (JAIR) 24: 851–887.

**Mauro Vallati** is a research fellow at the PARK research group of the University of Huddersfield, UK. His main research interest is learning for planning. He was co-organizer of IPCD-2014.

**Lukáš Chrupa** is a research fellow at the PARK research group of the University of Huddersfield, UK. His research interests mainly consist of applying machine learning and knowledge engineering techniques in AI planning. He was co-organizer of IPCD-2014.

**Marek Grześ** is a lecturer at the University of Kent at Canterbury, UK. His research interests include artificial intelligence, decision-theoretic planning, probabilistic reasoning, and applications thereof. He was a co-organizer of IPPC-2014.

**Thomas L. McCluskey** is a coleader of the PARK research group of the University of Huddersfield, UK. His research interests include automated plan generation, domain modeling, information extraction, knowledge engineering, machine learning, data mining, requirements validation, and formal specification. He cofounded and helps run the Knowledge Engineering for Planning and Scheduling competitions that are run biennially at ICAPS and that complement the IPC series. He was co-organizer of IPCD-2014.

**Mark Roberts** is a postdoctoral fellow for the National Research Council at the Naval Research Laboratory. He was a co-organizer of IPCL-2014.

**Scott Sanner** is a principal researcher at NICTA and an adjunct research fellow at the Australian National University. He was a co-organizer of both IPPC-2011 and IPPC-2014.