How to Spice up your Planning under Uncertainty Research Life

Scott Sanner
Statistical Machine Learning Group
NICTA, Australian National University
Canberra, Australia
ssanner@nicta.com.au

Abstract

Does planning under uncertainty have you down? Is your "state-of-the-art" uncertain planner slugglish in comparison to that peppy deterministic replanner that you wish would just go away? Do you feel like the world would just be better off without uncertain planners? Before you take drastic measures (and switch research topics), you should take a moment to read this short position paper.

Planning under uncertainty is a field rife with unexplored possibilities. Current benchmarks and planning competitions have only begun to scratch the surface of the types of problems that can be solved *and* the level of excitement to be had by exploring the research issues that these problems pose. Here we discuss just a few of the myriad extensions of planning under uncertainty that promise to spice up your uncertain planning research life.

Spicing up your Uncertain Planning Research

Like any activity, research in planning under uncertainty can seem dull and boring if occasional efforts are not made to explore interesting alternatives. Furthermore, by lack of exploration, we risk creating the misconception that problems currently being solved are the only problems of interest to the uncertain planning research community. In the following, we show how VIAGRA[1] may add some spice to your uncertain planning research life:

- *Ele***V***ators*: *Aerosmith aside, never underestimate the level of sheer planning excitement that is possible when elevators and planning under uncertainty are combined.*

  Of course, the key concept here is not the elevator itself, but rather the notion of multiple *concurrent* actions with uncertain outcomes. Such research has already been addressed in factored planning models (Guestrin, Koller, & Parr 2001b) where joint transition functions are factored according to individual concurrent actions. In such a model, the number of joint actions is generally exponential in the number of concurrent actions. Thus, for large numbers of concurrent actions, there simply is not an option to explore all possible actions or outcomes in a deterministic replanning framework, e.g., (Yoon, Fern, & Givan 2007). Even if all joint actions could be evaluated and only the most likely outcome[2] was used for replanning, it is not clear that this would lead to a useful model — even though an outcome may be most likely, it may still only occur a negligible fraction of the time given an exponential number of outcomes. Clearly, there may be some gain by exploiting the uncertainty directly in a true probabilistic planning approach by (efficiently) calculating the expected value of a course of action.

- *Cont***I***nuous State and Action Spaces*: *Going outside of your comfort zone makes things more interesting.*

  Sure, discrete state and action spaces are nice, but the world is generally not discrete. The Bellman equations still hold for uncertain planning problems that can be formalized as *continuous state and action Markov decision processes* (MDPs) (Puterman 1994). With this generalization, complex continous transition distributions and states consisting of continuous time and resources can be encoded, thus leading to more accurate models of real-world problems, e.g., Mars Rovers (Bresina *et al.* 2002). Of course, solving such continuous state and action MDPs is another story; it is not clear that deterministic replanning methods that rely on a most likely outcome (from an infinite set) would work well with multi-modal transition distributions. In this case, a proper expectation as computed by the optimal Bellman equations is likely to give a more robust solution.

- *Multiple* **A***gents*: *Sometimes it takes more than one agent to adequately spice up your planning research.*

  It is well-known in tha AI literature that multiagent problems can be formalized as a Markov game (Littman 1994) and that a simple minimax reformulation of the Bellman backup suffices to provide an optimal finite-horizon solution to this adversarial planning problem. With this generalization, one can model strict uncertainty in the transition model when it is affected by the actions

[1] Your results may vary. Please discontinue use if VIAGRA leads to an overwhelming feeling of unease or discomfort.

[2] Using variable elimination (Zhang & Poole 1996) to efficiently compute the max or expectation in the factored model.

of other self-interested agents or when transition probabilitites are not well-specified and the agent must plan for the worst case. While stochastic strategies may be required for optimality in this setting, it is not immediately clear how to generalize current deterministic replanners to cope with this paradigm and produce (approximately) optimal stochastic policies.

- *No* **G***oals*: *Sometimes its good not to have a predefined conception of exactly what you intend to achieve during planning under uncertainty.*

  A lot of uncertain planning research focuses on problems with clearly defined absorbing goal states. However, there is a more general class of problems that do not always have clearly defined goals but rather the more general task of optimizing *expected (infinite-horizon) discounted or average reward* (Boutilier, Dean, & Hanks 1999). Take for instance a mail-delivery robot: as different packages arrive due to exogenous events (see below), the robot must continuously optimize its delivery schedule to maximize reward over an infinite-horizon; note that there are no absorbing goal-states to be reached in this problem.

  In goal-oriented problems, it is already known that deterministic replanners may have difficulties with domains with *avoidable dead-ends* (Little & Thiebaux 2007) (although such problems may be partially resolved through dead-end analysis in the underlying domain). However, avoidable dead-ends are just the tip of the iceberg w.r.t. the ways in which the performance of optimal deterministic replanners may differ from the performance of optimal uncertain planners. For the more general class of MDPs with expected utility maximization objectives, the problem for deterministic replanners may be generalized to that of *avoidable low expected value states*. While deterministic planning may be generalized to cope with reasoning in expectation, doing so will start to blur the distinction between deterministic replanners and (approximately) optimal uncertain planners.

- **R***eal Problems*: *We cannot expect to maximize our planning under uncertainty experience if we play with toys instead of focusing on reality.*

  From *dialogue management* in natural language processing to *robotics* to real-time *program optimization*, many real-world problems inherently involve making sequential decisions that should be optimized for best performance. Most of these problems are *partially observable*, which speaks to the need for such model expressivity in practical uncertain planning research.

- *Exogenous* **A***ctions and Events*: *When unexpected things happen, uncertain planning can get interesting.*

  Most usage of the planning domain description language PPDDL (Younes *et al.* 2005) makes a strong frame assumption that only allows relational fluents directly referenced by an action's parameterization to change as a result of that action. More realistic probabilistic planning problems may also include multiple *exogenous* actions that occur independently, e.g., independent random failures of computers in the SysAdmin domain (Guestrin, Koller, & Parr 2001a), or arrival of packages in a mail-delivery robot domain. Planning in these non-inertial models can be very difficult and adds an extra dimension over simple stochastic versions of standard deterministic planning models (which inherently make a strong frame assumption). In Appendix B, we present a PPDDL variant of the SysAdmin problem with exogenous events that cause computers not directly affected by an action to crash. Problems such as this one may benefit from direct reasoning about uncertain exogenous events.

Summary

In brief, if you find your current uncertain planning research life to be lackluster, you may want to consider spicing it up with VIAGRA: *Ele***V***ators*, *Cont***I***nuous State and Transitions*, *Multiple* **A***gents*, *No* **G***oals*, **R***eal Problems*, *Exogenous* **A***ctions and Events*. While VIAGRA is certainly not the only answer, it offers the hope of opening up many new dimensions in uncertain planning that are likely to increase your general level of research interest and excitement.

References

Boutilier, C.; Dean, T.; and Hanks, S. 1999. Decision-theoretic planning: Structural assumptions and computational leverage. *Journal of Artificial Intelligence Research (JAIR)* 11:1–94.

Bresina, J.; Dearden, R.; Meuleau, N.; Ramakrishnan, S.; and Washington, R. 2002. Planning under continuous time and resource uncertainty: a challenge for ai. In *19th Conference on Uncertainty in AI (UAI-02)*.

Guestrin, C.; Koller, D.; and Parr, R. 2001a. Max-norm projections for factored MDPs. In *17th International Joint Conference on Artificial Intelligence (IJCAI-2001)*, 673–680.

Guestrin, C.; Koller, D.; and Parr, R. 2001b. Multiagent planning with factored MDPs. In *Advances in Neural Information Processing Systems 14 (NIPS-2001)*, 1523–1530.

Hoffmann, J., and Nebel, B. 2001. The FF planning system: Fast plan generation through heuristic search. *Journal of Artificial Intellence Research (JAIR)* 14:253–302.

Little, I., and Thiebaux, S. 2007. Probabilistic planning vs replanning. In *Workshop on International Planning Competition: Past, Present and Future (ICAPS-07)*.

Littman, M. L. 1994. Markov games as a framework for multi-agent reinforcement learning. In *International Conference on Machine Learning (ICML-94)*, 157–163.

Puterman, M. L. 1994. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. New York: Wiley.

Yoon, S.; Fern, A.; and Givan, R. 2007. FF-Replan: A baseline for probabilistic planning. In *17th International Conference on Automated Planning and Scheduling (ICAPS-07)*, 352–359.

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.

Zhang, N. L., and Poole, D. 1996. Exploiting causal independence in bayesian network inference. *Journal of Artificial Intelligence Research (JAIR)* 5:301–328.

Acknowledgements

Craig Boutilier fundamentally shaped my thinking on these issues. Kristian Kersting, Bob Givan, Brian Milch, Luke Zettlemoyer, and Robert Holte further contributed to these ideas during our discussions at the Dagstuhl seminar on "Probabilistic, Logical and Relational Learning - A Further Synthesis". Finally, the anonymous reviewers helped sharpen some of the general arguments. None of the above should be held accountable for my presentation.

Thanks also to Dan Bryce for pointing out how to encode the exogenous effects for SysAdmin in Appendix B and to Dan and Olivier Buffet for debugging the PPDDL specification in Appendix B.

Appendix A: Additional Responses to Workshop Provocations

The Role of the Planning Competition

Uncertain planning competitions are good. Common benchmarks for comparing plan length, running time, and success rate are crucial for fair comparisons of different planning approaches. However, uncertain planning competitions must necessarily appeal to a common denominator of planner capability in order to encourage widespread participation. As such, it is important not to assume that this common denominator is representative of the entire field of uncertain planning, nor is it appropriate to over-generalize results on the highly restricted planning competition benchmarks to be indicative of planner performance on more expressive (and difficult) uncertain planning domains like those discussed previously.

The Role of (Approximate) Optimality

To claim that (approximately) optimal probabilistic planning should not be the goal of uncertain planning research is curious. It is not clear what the role of uncertain planning research would be otherwise — (approximately) suboptimal planning? To the extent that uncertain planning *is* the task of sequential decision making in pursuit of some objective, the notion of (approximate) optimality is inextricable from the notion of uncertain planning.

The fact that deterministic replanners are good at solving current uncertain planning benchmark problems indicates that this approach *is* approximately optimal for this *subset* of problems. We should not take this lesson lightly; for many goal-oriented stochastic shortest path problems, deterministic replanning as in (Yoon, Fern, & Givan 2007) may be an excellent choice, especially given that FF-style heuristics (Hoffmann & Nebel 2001) prove to be highly effective for directing search in these problems.

On the other hand, we already know of some exceptions where deterministic replanners may perform poorly on problems with avoidable dead-ends (Little & Thiebaux 2007) and generalizations of these ideas to avoidable low expected value states discussed previously. While modifications to deterministic replanners may overcome these and other obstacles, as pointed out previously, it may require techniques that blur the distinction between deterministic replanners and (approximately) optimal uncertain planners. Such hybrid alternatives may represent a fair tradeoff between the efficiency of deterministic search-based heuristic approaches and potentially less efficient, but provably approximately optimal approaches.

Thus, when we must focus on heuristic solutions to uncertain planning for practical efficiency reasons, we should focus our effort on explaining why these heuristic techniques perform well (e.g., properties of problems where these heuristic techniques guarantee some degree of approximate optimality) and examine cases where they breakdown. Overall, such understanding will help with the construction of hybrid planners that are *efficient* and *robust* across a wide variety of problem domains.

Appendix B: Exogenous Effects in PPDDL

```
;; System Administrator Problem, variant of (Guestrin, Koller, Parr; IJCAI-2001)
;; Encoded in PPDDL by Scott Sanner with assistance of Dan Bryce & Olivier Buffet.
;;
;; Note: The original SysAdmin is discounted infinite horizon, with an additive
;;       reward function, and a transition function probability that scales
;;       according to the number of connected computers that are "up".
;;       The latter two additive aspects cannot be encoded in a lifted manner
;;       in PPDDL.
;;
;;       Here, a computer may fails if at least one of its upstream
;;       connections has failed, so it is important to reboot the computers
;;       with the highest downstream impact first.

(define (domain sysadmin)

 (:requirements :typing :equality :disjunctive-preconditions
                :probabilistic-effects :existential-preconditions
                :conditional-effects :negative-preconditions
                :universal-preconditions :rewards)
 (:types comp)

 (:predicates (up ?c)
              (conn ?c ?d))

;; Don't need for finite horizon problems
;;(:action noop
;; )

(:action reboot
  :parameters (?x - comp)
  :effect (and (decrease (reward) 1)
               (probabilistic 0.9 (up ?x))
               (forall (?d - comp)
                   (probabilistic
                      0.6 (when (exists (?c - comp) (and (conn ?c ?d)
                                                          (not (up ?c))
                                                          (not (= ?x ?d))))
                                (not (up ?d))
                          ))))
)

(define
 (problem sysadmin-5)
  (:domain sysadmin)
  (:objects comp0 - comp
            comp1 - comp
            comp2 - comp
            comp3 - comp
            comp4 - comp
  )
  (:init (conn comp0 comp1)
         (conn comp1 comp2)
         (conn comp2 comp3)
         (conn comp3 comp4)
         (conn comp4 comp0)
         (conn comp3 comp2)
         (conn comp0 comp4)
  )
  (:goal (forall (?c - comp)
                 (up ?c)))
  (:goal-reward 500)
)
```