Chance-Constrained Strong Controllability of Temporal Plan Networks with Uncertainty

Pedro H.R.Q.A. Santana and **Brian C. Williams**
Massachusetts Institute of Technology, Computer Science and Artificial Intelligence Laboratory, MERS
32 Vassar St. Room 32-224, Cambridge, MA 02139, +1 (617) 253-3447, {psantana,williams}@mit.edu

Abstract

This works presents a novel approach for determining chance-constrained strong controllability of Temporal Plan Networks with Uncertainty (TPNU) by framing it as an Optimal Satisfiability Problem (OpSAT).

Introduction

Providing autonomous systems with a keen sensitivity to risk and the ability to deal with uncertainty is required to allow them to be trusted in real-world situations. Inspired by airplane manufacturing, we envision a setting where humans and robots work together in a dynamic environment while completing a series of temporally-constrained tasks. In this context, robots not only have to rely on sensing information, but they must also reason about how their actions and random events might jeopardize plan success. This research focuses on the problem of chance-constrained planning over temporal actions, where uncertainty arises from the dependence of plan constraints on hidden quantities that we only have probabilistic information about.

Chance-constrained planning under uncertainty is a well-studied topic in the conditional and stochastic CSP literature, where state-of-the-art algorithms usually rely on a combination of chronological (depth-first) search and inference in order to quickly find satisficing solutions (Gelle and Sabin 2006; Fargier, Lang, and Schiex 1996; Tarim, Manandhar, and Walsh 2006). To the best of our knowledge, however, none of them has leveraged conflict-directed search (De Kleer and Williams 1987; De Kleer, Mackworth, and Reiter 1992; Williams and Ragno 2007) in order to improve their search performance. The contribution of this work is determining chance-constrained strong controllability of Temporal Plan Networks with Uncertainty (TPNU) (Effinger et al. 2009) by framing it as an instance of an Optimal Satisfiability Problem (OpSAT), which can be efficiently solved by Conflict-Directed A* (CDA*) (Williams and Ragno 2007). Prior literature on conditional and disjunctive temporal planning (Stergiou and Koubarakis 2000; Kim, Williams, and Abramson 2001; Tsamardinos, Vidal, and Pollack 2003; Effinger et al. 2009) has not focused on the problem of chance-constrained temporal planning. Compared to chronological search (CS) techniques used in state-of-the-art methods for solving similar problems, our most important contribution is in terms of the solution's scalability to larger problems sizes.

Problem statement and approach

Consider a TPNU (see Figure 1) conditioned on both controllable and uncontrollable (random) events, henceforth referred to as choices. Our goal is to find a set of assignments to the controllable choices (CC), plus a simple temporal network (STN) (Dechter, Meiri, and Pearl 1991) for the scheduling of activities, that is guaranteed to be temporally feasible with probability of at least $1-\Delta$, where Δ is a user-specified risk bound. The probabilistic component of temporal feasibility for TPNUs comes from the fact that uncontrollable choices (UC) are random variables, so the temporal constraints that they activate cannot be controlled or known beforehand.

A TPNU is said to be strongly controllable if there exists a set of off-line assignment to the CCs that ensures temporal feasibility no matter the outcome of UCs. Strong controllability criteria tend to be too conservative and hard to meet, which affects their usefulness. Also, it is very hard to find a realistic planning problem where one can be robust to any uncontrollable outcome. Hence, here we use the more useful notion of chance-constrained strong controllability, which stands for a set of off-line assignments to CCs that guarantees temporal feasibility with risk no greater than Δ.

Figure 1 shows an example TPNU representing a simple temporally-constrained manufacturing task. Circles represent the start and end events of an episode, the latter being denoted by ovals. An episode consists of a task that should be executed (written inside the oval), plus a simple temporal constraint in the form of an arc connecting the temporal events. Double solid circles denote CCs, while UCs have dashed lines. In this example, a robot has to choose between routes *A* (R_A) and *B* (R_B) in order to get to a manufacturing location, where it should ask a human (H) or a robotic manipulator (W) for assistance. There are four uncontrollable events that influence the plan: B_A and B_B are true when routes *A* and *B* are congested, respectively; G_U is true if the human gives up on the task of assisting the robot; and D_O is true if the robotic manipulator breaks down. We also assume that we have access to the probability distributions of these uncontrollable events.

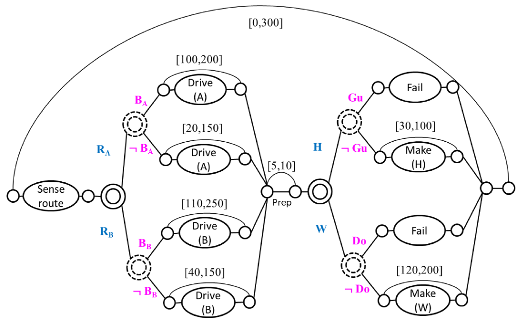

Figure 1: Example TPNU for a simple manufacturing task.

Conversion to OpSAT

An optimal constraint satisfaction problem (OpSAT) (Williams and Ragno 2007) extends the well-known concept of a CSP by quantifying how good a given solution is in terms of an objective function. The first step towards our OpSAT formulation is converting the TPNU in Figure 1 to the labeled constraint (LC) representation used in (Santana and Williams 2012). For example, the top branch in Figure 1 involving activity *Drive (A)* would be converted into the LC

$$R_A \wedge B_A \Rightarrow \text{Drive(A)} \in [100, 200]. \qquad (1)$$

An LC $L \Rightarrow C$ consists of a logical label L and a constraint C over the variables of the CSP. For the particular case of a TPNU, C represents simple temporal constraints (STC) between events. An LC is said to be active if $L \equiv$ True, forcing the CSP to satisfy C. If $L \equiv$ False, then the LC is said to be inactive and C can be excluded from the CSP. In the case of a TPNU, a full assignment to the CCs separates the LCs into three sets: *Necessarily Active* (*NA*), composed of all LCs where $L \equiv$ True; *Necessarily Inactive* (*NI*), composed of all LCs where $L \equiv$ False; and *Potentially Active* (*PA*), composed of all LCs whose labels may become True or False depending on assignments to UCs. It is easy to see that the set of active temporal constraints in the plan will always be a subset of $NA \cup PA$. Therefore, the goal of our algorithm is to find a subset $S \subseteq NA \cup PA$ that satisfies two conditions: I) S forms a consistent STN; II) the probability of any LC $l \in R = (NA \cup PA) \setminus S$ being active is no greater than Δ. All elements of R are said to be relaxed. The OpSAT problem is framed as:

1. **Decision variables**: $r_i \in \{0, 1\}$, $i \in \{1, \ldots, N\}$, where $r_i = 1$ if $l_i \in NA \cup PA$ is in R.
2. **Objective function**: $\min \Pr(\bigcup L_i)$, where L_i is the label of $l_i \in R$. Labels consist only of assignments to UCs (all CCs have been previously assigned) that activate an LC.
3. **Constraints**: S is a consistent STN and $\Pr(\bigcup L_i) < \Delta$.

Results

Figure 2 compares the average performance of CS and our OpSAT approach on a set of randomly generated temporal

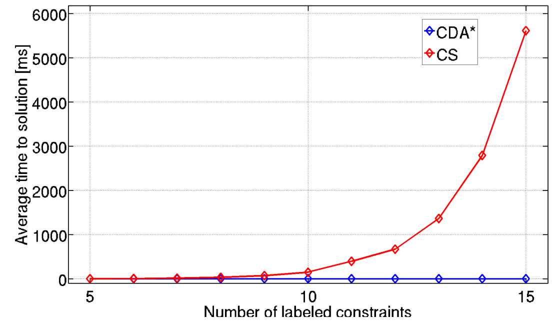

Figure 2: CDA* *versus* CS.

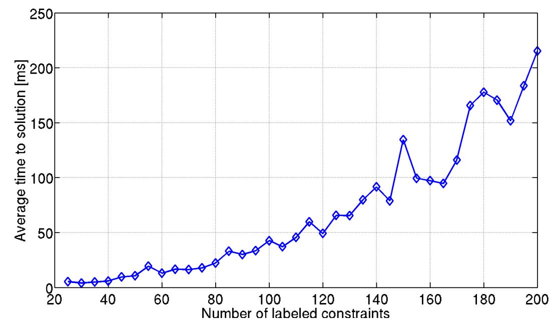

Figure 3: CDA* growth.

constraint networks. A problem with N LCs creates a search space of size 2^N. Both algorithms were ran until the first candidate satisfying the risk bound Δ was found or no more solutions were available. Whenever more than one solution existed, CDA* returned the one incurring minimum risk. For relatively small plans with no more than 10 LCs, we see that CS and CDA* showed very similar performances. However, if one increases the size of the problem by a few more constraints, we see a strong exponential growth in the time required by CS to either find a solution or return that no solution exists. Our approach using CDA*, on the other hand, kept its performance virtually unchanged. Despite the exponential trend in Figure 3 for CDA*, we see that it happens at a much smaller rate than for CS-based methods. Therefore, we hope that our approach will be suitable for reactive online planning even for moderately-sized plans composed of a few hundreds of LCs, for which CS-based methods cannot be adequately applied.

Conclusions and future work

This work presented an efficient approach for determining chance-constrained strong controllability of a plan conditioned on the outcome of uncontrollable events. A performance comparison with chronological search is provided, but future work will involve comparisons with other search strategies. We intend to apply this algorithm to dynamic execution of temporal constraints with active sensing actions in partially observable environments.

Acknowledgments

This research is funded by AFOSR grant 6926079.

References

De Kleer, J., and Williams, B. C. 1987. Diagnosing multiple faults. *Artificial intelligence* 32(1):97–130.

De Kleer, J.; Mackworth, A. K.; and Reiter, R. 1992. Characterizing diagnoses and systems. *Artificial Intelligence* 56(2):197–222.

Dechter, R.; Meiri, I.; and Pearl, J. 1991. Temporal constraint networks. *Artificial intelligence* 49(1):61–95.

Effinger, R.; Williams, B.; Kelly, G.; and Sheehy, M. 2009. Dynamic Controllability of Temporally-flexible Reactive Programs. In *Proceddings of the 19th International Conference on Automated Planning and Scheduling (ICAPS 09)*.

Fargier, H.; Lang, J.; and Schiex, T. 1996. Mixed constraint satisfaction: A framework for decision problems under incomplete knowledge. In *Proceedings of the National Conference on Artificial Intelligence*, 175–180.

Gelle, E., and Sabin, M. 2006. Solver framework for conditional constraint satisfaction problems. In *Proceeding of European Conference on Artificial Intelligence (ECAI-06) Workshop on Configuration*, 14–19.

Kim, P.; Williams, B. C.; and Abramson, M. 2001. Executing reactive, model-based programs through graph-based temporal planning. In *IJCAI*, volume 17, 487–493.

Santana, P., and Williams, B. 2012. A bucket elimination approach for determining strong controllability of temporal plans with uncontrollable choices. In *26th AAAI Conference on Artificial Intelligence*.

Stergiou, K., and Koubarakis, M. 2000. Backtracking algorithms for disjunctions of temporal constraints. *Artificial Intelligence* 120(1):81–117.

Tarim, S. A.; Manandhar, S.; and Walsh, T. 2006. Stochastic constraint programming: A scenario-based approach. *Constraints* 11(1):53–80.

Tsamardinos, I.; Vidal, T.; and Pollack, M. 2003. CTP: A new constraint-based formalism for conditional, temporal planning. *Constraints* 8(4):365–388.

Williams, B., and Ragno, R. 2007. Conflict-directed A * and Its Role in Model-based Embedded Systems. *Discrete Applied Mathematics* 155(12):1562–1595.