

A Bucket Elimination Approach for Determining Strong Controllability of Temporal Plans with Uncontrollable Choices

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-D224, Cambridge, MA 02139, +1 (617) 253-3447
psantana@mit.edu, williams@mit.edu

Abstract

This work presents a new algorithm based on the Bucket Elimination framework that efficiently determines strong controllability of temporal plans formulated as Labeled Simple Temporal Networks with Uncertainty (LSTNU) with controllable and uncontrollable plan branches (choices).

Introduction

This work’s motivation stems from the study of robust task execution in uncertain environments, grounded in the application of robotic manufacturing. Among its many challenges, it is important to highlight the key role played by uncontrollable choices. Besides modeling runtime “exceptions”, uncontrollable choices are also very useful for representing conditional branches in a temporal plan which depend on future sensor readings, as introduced by the Conditional Temporal Plan (CTP) formalism (Tsamardinos, Vidal, and Pollack 2003). One way of guaranteeing that a plan with uncertainty will be able to satisfy all required constraints is by verifying its strong controllability, i.e., checking whether there is a precomputable assignment to the controllable variables that is robust to all possible outcomes of uncontrollable events. However, the algorithm for determining strong controllability of CTPs requires a search through the full space of uncontrollable plan branches, which grows exponentially with the number of uncontrollable choices and might become impractical in situations where sensing actions, and consequently uncontrollable choices, are ubiquitous, such as in our manufacturing scenario. In addition, CTPs do not handle uncertainty regarding activity durations. In this work, we leverage the temporal plan representations in (Conrad and Williams 2011; Effinger et al. 2009) in order to define Labeled Simple Temporal Network with Uncertainty (LSTNU) in the next section, where labels are formed by both controllable and uncontrollable choices and constraints can be of any type, including uncontrollable temporal constraints.

In this work, we propose a novel approach to the problem of checking strong controllability of temporal plans with uncontrollable choices based on a modified version of the Directional Resolution (DR) algorithm introduced

within the Bucket Elimination framework (Dechter 1999). By extending the latter with several lemmas developed in this work, we are able to eliminate the dependency of labeled temporal constraints on uncontrollable choices while maintaining their logical consistency without performing any search on the space of plan branches. Therefore, we can readily refute strong controllability or rewrite the problem as a function of only decision variables that are controlled by the plan executive. For additional motivation, a step-by-step numerical example involving our LSTNU representation, and proofs for the lemmas presented in the next sections, please refer to <http://people.csail.mit.edu/psantana/public/SantanaAAAI12.pdf>.

Problem Formulation

This section briefly defines strong controllability and the elements of an LSTNU, which are useful for understanding the main contributions in the next section.

Definition 1. (Label variable) A label variable (LV) X is an injection $X:D \rightarrow \mathbb{N}$ between the variable’s domain D and the set \mathbb{N} and is used to represent all possible outcomes of a plan branch (choice), either controllable or not. The symbol X_i represents the assignment $X=i$.

Definition 2. (Label) A label E is a logical expression written in Disjunctive Normal Form (DNF), i.e., $E = \alpha_1 \vee \dots \vee \alpha_n$, where α_i is the i -th conjunction of assignments to LVs.

Definition 3. (Constraint) A constraint is a pair $C = \langle S, R \rangle$, where S represents the constraint’s scope (the set of variables involved in the constraint) and R is a relation between the variables in S . If R holds, then we say that the constraint is satisfied. Otherwise, we say that it is violated.

Considering the STN model presented in (Dechter, Meiri, and Pearl 1991), we can write $S = \{T_i, T_j\}$ and $R : l \leq T_i - T_j \leq u$, where T_i and T_j are the time instants when two events occur and u and l correspond, respectively, to upper and lower bounds on their temporal distance.

Definition 4. (Labeled constraint) A labeled constraint (LC) $L_C = \langle E, C \rangle$ consists of a label E and a constraint C such that

$$E \Rightarrow C. \quad (1)$$

If E is TRUE, we say that L_C is active. Otherwise, L_C is said to be inactive.

It should be noticed that (1) is general enough to represent DTPs. In order to write a disjunctive constraint as (1), it suffices to represent each constraint disjunct as a different assignment to an LV. Depending on whether the plan executive or the environment picks which constraint should be satisfied, the LV can represent either a controllable or uncontrollable choice. In addition to the strong controllability definitions of (Morris, Muscettola, and Vidal 2001; Tsamardinis, Vidal, and Pollack 2003), we use the following in our results.

Definition 5. (Strong controllability) Let $E_i \Rightarrow C_i$, $i \in \{1, \dots, n\}$, be the set of all LCs that involve the LV X . In addition, define $L_X := \bigwedge_{i=1}^n (E_i \Rightarrow C_i)$. Then, we say that the planning problem is strongly controllable with respect to X if L_X always holds.

Determining strong controllability

The original DR algorithm is meant to perform variable resolution in logical expressions written in Conjunctive Normal Form (CNF). However, due to the disjunctive nature of choices, the labels in (1) are written in DNF. In addition, eliminating uncontrollable LVs has a direct impact on the constraints that the resulting LCs imply, which is also not handled by DR. In order to overcome these issues, we present the following lemmas, enunciated without proof.

Lemma 1. (Label separation) Let E_1 and E_2 be two possibly distinct labels and C a constraint such that $E_1 \vee E_2 \Rightarrow C$. Then the following holds

$$(E_1 \vee E_2 \Rightarrow C) \Leftrightarrow (E_1 \Rightarrow C) \wedge (E_2 \Rightarrow C).$$

Lemma 2. (Resolution in DNF) Let $\alpha_1, \dots, \alpha_n$, be a set of possibly distinct conjunctions of literals and X_1, \dots, X_n , a set of predicates representing all possible n values of an LV X . Then the following holds

$$(\alpha_1 \wedge X_1) \vee \dots \vee (\alpha_n \wedge X_n) \Rightarrow \alpha_1 \vee \dots \vee \alpha_n.$$

Lemma 3. (Strong controllability) Let E_1, \dots, E_n be a set of possibly distinct labels, each one of them consisting of exactly one conjunction of literals. In addition, assume that E_i contains the literal X_i , the i -th assignment to a choice variable with n possible values. Assuming that E_i labels the constraint C_i , the following holds

$$(E_1 \vee \dots \vee E_n \Rightarrow C_1 \wedge \dots \wedge C_n) \Rightarrow \bigwedge_{i=1}^n (E_i \Rightarrow C_i).$$

Lemma 3 provides a sufficient condition for strong controllability explored in Algorithm 2. A method for checking strong controllability of an LSTNU is given in Algorithm 1.

Conclusions and Future Work

Algorithms 1 and 2 have been successfully implemented and tested on example LSTNUs using (1) and consisting of both controllable and uncontrollable choices. Future work will investigate similar algorithms for dynamic and weak controllability of LSTNUs, since they will be readily applicable to robust task execution in uncertain environments and planning with sensing actions.

Algorithm 1 Strong Controllability of LSTNU

- Run Algorithm 2 in order to eliminate the dependency of LCs on uncontrollable LVs.
 - **IF** (the output of Algorithm 2 is non-null)
 - **FORALL** (full assignments to controllable LVs)
 - **IF** (The set of activated LCs is consistent) **THEN**
 - **RETURN** The problem is strongly controllable and the corresponding LCs
 - **RETURN** The problem is NOT strongly controllable
 - **RETURN** The problem is NOT strongly controllable
-

Algorithm 2 Modified DR

- Rewrite the LCs' labels as single conjunctions of assignments to LVs (Lemma 1).
 - Generate an ordered partition $Bucket_1, \dots, Bucket_m$, where $Bucket_i$ contains all the LCs whose highest LV is i . Create also a special bucket, $Bucket_0$, which will hold all LCs that do not depend on any LVs being eliminated.
 - **FOR** ($i = 1$ to m)
 - $C_{conj} \leftarrow$ Conjunction of all constraints within the bucket (Lemma 3)
 - **IF** ($C_{conj} \equiv FALSE$) **THEN RETURN NULL**
 - **ELSE**
 - **FORALL** (LC within the bucket)
 - Eliminate the assignment to i from its label (Lemma 2) and replace the constraint by C_{conj}
 - Move it to the next appropriate bucket
 - **RETURN** Content of $Bucket_0$
-

Acknowledgments

Thanks to Cheng Fang, Steven Levine, Masahiro Ono, and Andrew Wang for their invaluable comments and to Michael Kerstetter, Scott Smith, Ronald Provine, and the Boeing Company for their support. This research is funded by The Boeing Company grant MIT-BA-GTA-1.

References

- Conrad, P., and Williams, B. 2011. Drake: An Efficient Executive for Temporal Plans with Choice. *Journal of Artificial Intelligence Research* 42:607–659.
- Dechter, R.; Meiri, I.; and Pearl, J. 1991. Temporal constraint networks. *Artificial Intelligence* 49:61–95.
- Dechter, R. 1999. Bucket elimination: A unifying framework for probabilistic inference. *Learning in Graphical Models* 75–104.
- Effinger, R.; Williams, B.; Kelly, G.; and Sheehy, M. 2009. Dynamic Controllability of Temporally-flexible Reactive Programs. In *Proc. of ICAPS 09*.
- Morris, P.; Muscettola, N.; and Vidal, T. 2001. Dynamic control of plans with temporal uncertainty. In *Proc. of IJCAI 01*, volume 17, 494–502.
- Tsamardinis, I.; Vidal, T.; and Pollack, M. 2003. CTP: A new constraint-based formalism for conditional, temporal planning. *Constraints* 8(4):365–388.