Assessing the Probability of Legal Execution of Plans with Temporal Uncertainty

Ioannis Tsamardinos
Department of Biomedical Informatics
Vanderbilt University
Nashville, TN 37232 USA
ioannis.tsamardinos@vanderbilt.edu

Martha E. Pollack
Computer Science and Engineering
University of Michigan
Ann Arbor, MI 48109 USA
pollackm@eecs.umich.edu

Sailesh Ramakrishnan
QSS Group Inc.
NASA Ames Research Center
Moffett Field, CA 94305 USA
sailesh@email.arc.nasa.gov

Abstract

Temporal uncertainty is a feature of many real-world planning problems. One of the most successful formalisms for dealing with temporal uncertainty is the Simple Temporal Problem with uncertainty (STP-u). A very attractive feature of STP-u's is that one can determine in polynomial time whether a given STP-u is dynamically controllable, i.e., whether there is a guaranteed means of execution such that all the constraints are respected, regardless of the exact timing of the uncertain events. Unfortunately, if the STP-u is not dynamically controllable, limitations of the formalism prevent further reasoning about the probability of legal execution. In this paper, we present an alternative formalism, called Probabilistic Simple Temporal Problems (PSTPs), which generalizes STP-u to allow for such reasoning. We show that while it is difficult to compute the exact probability of legal execution, there are methods for bounding the probability both from above and below, and we sketch alternative candidate algorithms for this purpose. Computing the probability of legal execution allows a temporal planner to decide, when uncertainty is present, whether to accept or reject candidate plans. In addition, lower bound computation has an important side-effect: it provides guidance as to how to execute an STP-u even when it is not dynamically controllable.

Introduction

Many real-world planning problems involve temporal constraints, and a number of planning formalisms and algorithms have been developed to deal with them. One of the most well-known is the Simple Temporal Problem (STP) formalism (Dechter, Meiri, & Pearl 1991), which allows the representation of temporal constraints of the form $X - Y \leq d$, where X and Y are the times of occurrence of two instantaneous events in the plan and d is a real number (or infinity). For example, if X and Y denote the start and end points of a single action, then the constraint specifies that the action takes no more than d time units.

The STP formalism, along with generalizations of it, such as the Disjunctive Temporal Problem (DTP) (Stergiou & Koubarakis 2000; Tsamardinos 2001) have been very fruitful, both for theoretical investigations of temporal planning (Smith, Frank, & Jónsson 2000) and for practical deployment, notably in NASA's Remote Agent (Muscettola *et al.* 1998). However, these formalisms do not allow any explicit representation of uncertainty. Yet in most interesting, real-world domains, there are many types of uncertainty. One type of uncertainty is associated with conditional execution of actions that depend on observations and the status of the world during execution. The Conditional Temporal Problem (Tsamardinos, Vidal, & Pollack 2003) is an extension of the STP that is able to encode and reason with quantitative temporal constraints and conditional branches.

Another type of uncertainty, that this paper addresses, is *temporal uncertainty*, i.e., uncertainty about the time at which particular events will occur. Such events are said to *uncontrollable*, to distinguish them from the events that are under the control of the agent executing the plan. Often, plans must include temporal constraints that involve uncontrollable events: for instance, it may be necessary to respond to an alarm within two minutes of its going off. The time of the alarm is not within the control of the execution agent, but the time of the responsive event is.

In order to model uncontrollable events, an extended formalism, called Simple Temporal Problems with Uncertainty (STP-u) was developed (Vidal & Ghallab 1996; Vidal & Fargier 1997; Morris, Muscettola, & Vidal 2001). With STP-u's, one can model plans that contain constraints involving uncontrollable events. Notice that with such plans, decisions about when to perform actions must often be deferred until execution time. For instance, one cannot decide in advance when to respond to an alarm: all one can do is wait until the alarm goes off and then respond accordingly.

A very attractive feature of STP-u's is that one can determine in polynomial time whether a given STP-u is *dynamically controllable*, i.e., whether there is a guaranteed means of execution such that all the constraints are respected, regardless of the exact timing of the uncertain events[1]. Not all STP-u's are dynamically controllable. As a simple example, consider an STP-u that includes a constraint requiring an agent to respond to an uncontrollable alarm exactly two minutes *before* it goes off. If the agent doesn't control the alarm–does not know when it will go off and does not have any means of making it go off–then clearly the agent cannot act in a way to satisfy this constraint.

A plan generation system can approach the task of plan-

[1]Here, and in the rest of the paper we assume the only source of uncertainty in the plan is the temporal uncertainty of the uncontrollable events.

ning under temporal uncertainty by generating a plan formulated as an STP-u and then checking to see whether it is dynamically controllable. If it is, the planner can declare success. Unfortunately, if it is not, limitations of the STP-u formalism preclude further reasoning both about the probability of legal execution, and about strategies to maximize that probability. Consequently, the planning system does not know whether to adopt the plan or to search for an alternate, and, if the former–if it does adopt the plan–it is unable to formulate a effective means of executing it.

In this paper, we present a new formalism, called Probabilistic Simple Temporal Problems (PSTPs), which generalizes STP-u's in a way that supports reasoning about the probability that a plan with temporal uncertainty will be legally executed. Although is is difficult to compute an exact probability, we show that there are methods for bounding the probability both from above and below. An upper bound can be used to reject a current candidate plan which falls below a given threshold, while a lower bound can be used to accept a candidate plan.

The remainder of our paper is organized as follows. In the Background section we review the background material on STPs, STP-u's, and dynamic controllability, and in the next section we introduce the Probabilistic Simple Temporal Problems (PSTP) formalism. In the following two sections we describe the technique for computing the upper bound and lower bound on the probability of correctly executing a PSTP, respectively. In particular, we explain how the problem of computing the lower bound can be addressed by first converting a PSTP to an STP-u and then tightening the bounds on that STP-u until it becomes dynamically controllable. The following three sections then sketch some approaches to this lower bound computation.

Finally, the Discussion section summarizes the main ideas, open questions, and future directions. We note there in particular that the lower bound computation has an important side-effect: *it results in the specification of an execution strategy that maximizes the probability of satisfying all the execution constraints*. From another perspective this means that it provides guidance as to how to execute an STP-u even when it is not dynamically controllable.

Background

The formalism used to represent temporal information and uncertainty is based on the Simple Temporal Problem (STP) defined below:

Definition 1. Simple Temporal Problem, STP Solution, Consistent STP. A Simple Temporal Problem (**STP**) is a pair $< V, E >$, where

- V is a set of variables (also called nodes, events, or time-points) taking real values representing the time of occurrence of instanteneous events.
- E a set of temporal constraints on the variables of the form $X - Y \leq b$, $X, Y \in V$, and $b \in \Re \cup \{-\infty, \infty\}$.

A **solution** to an STP is an assignment to the variables that satisfies the constraints. An STP is **consistent** if there exists at least one solution.

STPs have been used to represent temporal plans by using a variable for each action's start and end point. For example, if $start(A)$ and $end(A)$ are the events of starting and ending action A, then the constraints $5 \leq end(A) - start(A) \leq 10$ specify the duration of the action to be between 5 and 10 time units.

STP constraints are binary. In order to represent unary constraints on absolute execution time, e.g., $100 \leq start(A) \leq 200$, a special variable called time reference point TR is defined and is assigned time zero; then, the constraint above can be written as $100 \leq start(A) - TR \leq 200$.

By using an all-pairs shortest path algorithm one can discover the distance from Y to X denoted as d_{YX} and defined as follows:

Definition 2. The **distance** from variable Y to variable X denoted as d_{YX} is the smallest number for which the equation $X - Y \leq d_{YX}$ holds in all STP solutions.

STPs can be executed with minimal on-line constraint propagation as discussed in (Muscettola, Morris, & Tsamardinos 1998; Tsamardinos, Morris, & Muscettola 1988). An STP does not represent uncertainty information about the occurence of events. All variables are assumed to be under the direct control of the agent executing the represented plan and so, if there exists a solution, then the STP is executable in a way that satisfies its constraints. To address this representational limitation, the Simple Temporal Problem with Uncertainty (STP-u) formalism was developed (Vidal & Ghallab 1996; Vidal & Fargier 1997; Morris, Muscettola, & Vidal 2001).

An STP-u makes a distinction between controllable and uncontrollable variables. *Controllable* variables are the ones whose timing of execution is under the direct control of the agent. *Uncontrollable* variables are the ones whose timing of execution depends on Nature (i.e., exogenous factors). The only information represented regarding the exact timing of an uncontrollable X is that it will occur sometime within the interval $[l, u]$ after a controllable Y, called the *parent* of X. Thus, the STP-u specifies that Nature will respect the constraint $l \leq X - Y \leq u$. These constraints involving Nature are called *contingent links* and are distinct from the constraints the plan has to respect to be legally executed, called *requirement links*.

Definition 3. Simple Temporal Problem with Uncertainty. A Simple Temporal Problem with Uncertainty **STP-u** is a tuple $< V_C, E, V_U, C >$, where

- V_C and V_U are the set of controllable and uncontrollable variables, respectively, taking real values.
- E is a set of constraints (requirement links) of the form $X - Y \leq b$, $X, Y \in V_C \cup V_U$, and $b \in \Re \cup \{-\infty, \infty\}$.
- C is a set of contingent links of the form $l \leq X - Y \leq u$, $Y \in V_C$, $X \in V_U$, and $l, u \in \Re$.

Figure 4 shows an STP-u with two controllable variables A, B and an uncontrollable variable C. The edge $A \rightarrow B$ in the figure, annotated with the interval $[p, q]$ graphically expresses the constraints $p \leq B - A \leq q$, i.e., $A - B \leq -p$ and $B - A \leq q$. Similar constraints hold for the other edges. The edge $A \rightarrow C$ is a contingent link, i.e., a constraint Nature is expected to observe.

Contingent constraints are always between a controllable variable Y and an uncontrollable variable X. A contingent link $l \leq end(A) - start(A) \leq u$ may be used for example to specify that the expected duration of an action A is between l and u time units; however, this duration is not something that is determined by the agent.

An STP-u, like an STP, should be executed in such a way that all its constraints are satisfied. However, as we illustrated in the introduction with the alarm example, the existence of uncontrollable events means that decisions about the timing of controllable events may need to be deferred to execution time.

Definition 4. Legal Execution, Execution Strategy. A **legal execution** of a STP-u $< V_C, E, V_U, C >$ is a schedule (time assignment) of occurrences of the events (variables) in $V_C \cup V_U$ in a way that satisfies all the constraints in E. An **execution strategy** is an algorithm that decides when to execute the next controllable action given the execution constraints *and* the observed history of the uncontrollable events.

Definition 5. Dynamic Controllability. (Informal) An STP-u $< V_C, E, V_U, C >$ is **dynamically controllable** if there exists an execution strategy that results in a legal execution regardless of when the uncontrollable variables in V_U occur (provided they occur within the bounds specified by the contingent constraints in C).

For a formal definition of dynamic controllability, see (Morris, Muscettola, & Vidal 2001), which also provides a polynomial-time algorithm for checking whether a given STP-u is dynamically controllable.

How does a domain expert model temporal uncertainty when specifying the constraints for a planner? For any temporally uncertain event A, the expert must specify some bounds on the time of occurrence of A. In the STP-u formalism, this corresponds to setting the values of l and u in a contingent link $l \leq end(A) - start(A) \leq u$. The looser these bounds are set to be, the more likely they are to be observed by Nature, and hence, the more accurate the model is; in the extreme case, if they are set to positive and negative infinity, then the expert is certain that the event will occur within the specified bounds. On the other hand, as the bounds get looser, the likelihood decreases that the STP-u is dynamically controllable. And when the STP-u is not dynamically controllable, the formalism provides no guidance about when to perform the controllable events so as to increase the probability of observing the constraints. Currently, there are no principled procedures for deciding the bounds of the uncontrollables in a way that maximizes the probability that Nature will respect them and that the resulting STP-u will be dynamically controllable.

In fact, because STP-u's do not explcitly represent probability distributions of uncontrollable events, they lack the information needed for such decisions. Probabilistic Simple Temporal Problems (PSTPs), first presented in (Tsamardinos 2002), are an extension of STP-u that includes such information in the temporal plan.

Probabilistic Simple Temporal Problems

We begin by defining PSTPs.

Definition 6. A Probabilistic Simple Temporal Problem **PSTP** is a tuple $< V_C, E, V_U, Par, \mathcal{P} >$, where:

- V_C is the set of controllable variables with real values.
- V_U is a set of real random variables (uncontrollable variables).
- E a set of constraints of the form $X - Y \leq b$, $X, Y \in V_C \cup V_U$, and $b \in \Re \cup \{-\infty, \infty\}$.
- Par is a function $V_U \rightarrow V_C$ that specifies for each uncontrollable its controllable parent.
- $\mathcal{P}$ is a set of conditional probability density functions (pdf) $p_X(t)$ for each $X \in V_U$ providing the mass of probability of X occurring t time units *after* $Par(X)$.

It is worth noting that in PSTPs, each uncontrollable event has a single parent, which must be a controllable event; thus, function Par is well-defined. The probability functions in $\mathcal{P}$ deserve further comment. $\mathcal{P}$ is a set of probability distributions $p_X(t)$ summarizing expectations about the occurrence of each uncontrollable event X. More precisely, given $p_X(t)$, the probability of X occurring T time units or less after $Par(X)$ has occurred is given by $P_X(t \leq T) = \int_{-\infty}^{T} p_X(t)dt$.

Implicit in our definitions is that the probability distribution of occurence of X with time does not depend on absolute time but on relative time from the moment the parent of X is executed ($p_X(t)$ is time invariant).

As an example, suppose that we want to model the fact that an action A has duration normally distributed with mean duration of half an hour ($30'$) and $5'$ standard deviation σ. We define the beginning of the action as the controllable $Y = start(A)$, and the end of the action as the uncontrollable $X = end(A)$, for which $p_X(t)$ follows $Normal(30, 5)$ (normal with half an hour mean and 5 minutes standard deviation). Then we can find out the probability that the action will finish within in 40 minutes after Y (i.e., after we started the action): $P(t \leq 40) = \int_{-\infty}^{40} p_X(t)dt = \Phi(\frac{40-30}{5}) = 97.72\%$, where $\Phi(z)$ is the integral of the Normal(0,1) at point z.[2]

Let us compare modelling action A in a PSTP with modelling the same action in an STP-u. In an STP-u one has to come up with reasonable bounds l and u and specify that $l \leq X - Y \leq u$ is a contingent link to be included in the plan. In comparison, in an PSTP the same constraint is represented by specifying that the parent of X is Y and that X will occur t time units after Y where t follows a normal probability distribution with mean $30'$ and standard deviation $5'$.

Given a PSTP, our goal is to assess the probability that all its execution constraints E will be satisfied during execution. More specifically, we would like to calculate *the probability of the plan being legally executed under an optimal execution strategy*. The reason for doing this is to provide

[2] This integral cannot be solved analytically but is typically computed numerically or provided in a table form.

guidance to the planning process: we want to know whether the current plan is "good enough", i.e., likely enough to succeed, or whether, instead, further effort should be put into looking for a better plan.

(Tsamardinos 2002) shows how to find an optimal execution strategy for PSTPs under certain conditions. However, the basic approach there is a static one, i.e., one that corresponds to strong controllability in STP-u's. To compute the equivalent of a dynamic execution strategy, it is necessary to iterate the process of finding an optimal PSTP execution strategy whenever an observation of an uncontrollable occurs. Computing the exact probability of success of this overall dynamic strategy is difficult. However, we do know how to compute bounds on the probability of success, and that is what we focus on in the remainder of this paper. In the next section, we show how to compute an upper bound on the probability of success. This bound can be used by a system to reject a plan, if it is too low. In the following sections, we describe how to compute a lower bound, by converting a PSTP to an STP-u and then tighening the latter's bounds until it becomes dynamically controllable.

Bounding the Probability of Legal Execution from Above

Suppose that an uncontrollable variable X with parent Y occurs t time units after Y, i.e., $X - Y = t$. If there is no solution to the constraints in E that admits a value t for the difference $X - Y$ then the probability of completing the execution in a way that respects the constraints is zero.

As we mentioned earlier, the distance between Y and X is the minimum number d_{YX} for which $X - Y \leq d_{YX}$ holds in all solutions. Similarly, $Y - X \leq d_{XY}$ holds in all solutions. These inequalities together imply that $-d_{XY} \leq X - Y \leq d_{YX}$ in any legal execution. Therefore, with probability $p_X(t)$ for t outside the interval $[-d_{XY}, d_{YX}]$ a legal execution cannot be achieved. Equivalently, a legal execution can be achieved with probability density at most $p_X(t)$ for t within the interval $[-d_{XY}, d_{YX}]$.

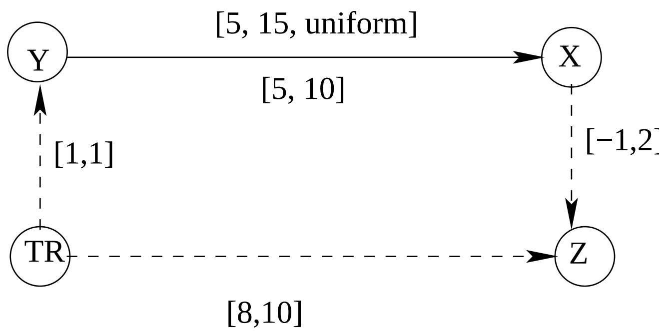
Assuming all uncontrollable events are independent of each other, and using $Success$ to denote the event of a legal execution occuring, then:

$$P(Success) \leq \prod_{X \in V_U, Y = Par(x)} P_X(t \in [-d_{XY}, d_{YX}])$$

$$= \prod_{X \in V_U, Y = Par(X)} P_X(-d_{XY} \leq t \leq d_{YX})$$

The distances d_{XY} can be determined with a polynomial all-pairs shortest path algorithm. The calculation of the probabilities in the product depends on the exact density functions. As an example, if $p_X(t)$ is uniform in the interval $[a, b]$, then $P_X(-d_{XY} \leq t \leq d_{YX}) = \frac{d_{XY} - (-d_{YX})}{b-a} = \frac{d_{YX} + d_{XY}}{b-a}$), assuming $[-d_{XY}, d_{YX}] \subseteq [a, b]$.

As an example consider the PSTP in Figure 1. The dotted edges represent temporal constraints. Specifically, each edge $A \rightarrow B$ annotated with the interval $[l, u]$ represents the



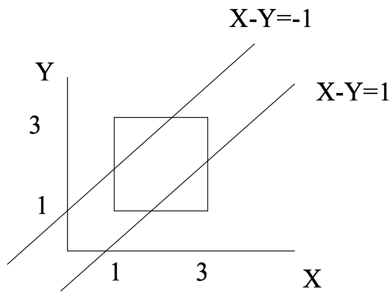


Figure 2: The polytope defined by the constraints provides a tighter upper bound.

two inequality constraints $l \leq B - A \leq u$. There are three such constraints (six single inequality constraints):

$$1 \leq Y - TR \leq 1$$

$$8 \leq Z - TR \leq 10$$

$$-1 \leq Z - X \leq 2$$

We can rewrite these as:

$$-1 \leq TR - Y \leq -1$$

$$8 \leq Z - TR \leq 10$$

$$-2 \leq X - Z \leq 1$$

By adding them up we infer that $5 \leq X - Y \leq 10$. This inferred constraint is depicted in the figure with the annotation $[5, 10]$ below the solid line. It is equivalent to calculating the distances between X and Y: $d_{XY} = -5$ and $d_{YX} = 10$.

The solid link denotes the fact that X is an uncontrollable variable with parent Y and the interval on top of this edge shows the form of this dependency: X will occur some time within the interval $[5, 15]$ after Y has been executed with uniform distribution. For example Y may be the beginning of an action with duration between 5 and 15 and X the end of this action.

As calculated, there is a solution to the constraints only if $5 \leq X - Y \leq 10$. Thus, with at most probability of $P_X(5 \leq t \leq 10) = \frac{10-5}{15-5} = \frac{1}{2}$ the PSTP can be executed successfully.

The upper bound calculated with the above method may not be tight in general. We now discuss ideas why this is the case and how to find tighter bounds. Consider a PSTP with two uncontrollables X and Y that both occur with uniform probability in $[1, 3]$ after the time reference point TR. Let us assume that the only constraints in this PSTP are $-1 \leq X - Y$ and $X - Y \leq 1$.

Figure 2 shows the space of legal executions. The x-axis is the time of occurence of X and similarly for Y. Since $TR = 0$ the set of legal execution is the area of the rectangle bounded by the lines $X - Y = -1$ and $X - Y = 1$. Calculating the bounds with the method that we provided yields: $P_X(-d_{X,TR} \leq t \leq d_{TR,X})P_Y(-d_{Y,TR} \leq t \leq d_{TR,Y}) = P_X(-\infty \leq t \leq \infty)P_Y(-\infty \leq t \leq \infty) = 1$.

A tighter bound on the probability in this example would be the area of the rectangle bounded by the two constraints. In general, a tighter bound could be obtained by calculating the mass of probability contained in the polytope defined by the constraints.

The mass of probability of this polytope is still only an upper bound on the probability of correct execution: even though for every point in this polytope (i.e., execution) the constraints are satisfied, that does not mean that an agent will dynamically be able to construct this solution, unless it is clairvoyant in the general case.

Bounding the Probability of Legal Execution from Below

Let $< V_C, E, V_U, Par, \mathcal{P} >$ be a PSTP and suppose that we are given intervals $[l_X, u_X]$ for each uncontrollable variable X with parent Y. The PSTP and the intervals can be seen as corresponding to an STP-u with the same controllable and uncontrollable variables, same constraints, and contingent links $l_X \leq X - y \leq u_X$ for each uncontrollable variable.

If this STP-u is dynamically controllable then there exists an execution strategy for legally executing the STP-u no matter when the uncontrollables occur *within these intervals*. Thus, in all such cases where the uncontrollables occur within these bounds an agent can execute the plan with probability one, provided it follows the execution strategy returned by the STP-u controllability algorithm. The probability of all such cases is thus a lower bound on the probability of a legal execution. Thus:

$$P(Success) \geq \prod_{X \in V_U} P_X(l_X \leq t \leq u_X)$$

Unlike the upper bound that we provided in the previous section, in the lower bound case the bounding intervals $[l_X, u_X]$ cannot be easily computed (because it is required that the corresponding STP-u is controllable). The looser these intervals the tighter the lower bound will be. To find the tightest bound possible one needs to solve the following optimization problem:

Definition 7. Lower Bound Problem.

Given PSTP $< V_C, E, V_U, Par, \mathcal{P} >$:

Maximize $\prod_{X \in V_U} P_X(l_X \leq t \leq u_X)$
subject to:
$< V_C, E, V_U, C >$ being dynamically controllable
where C is the set of contingent links
$\{l_X \leq X - Y \leq u_X \mid X \in V_U, Y = Par(X)\}$
and decision variables:
l_X, u_X, for each $X \in V_U$

Unfortunately, it is difficult to directly apply typical constraint optimization techniques such as gradient descent or Newton's Method on this problem. This is because such methods require expressing the feasible set as the decision variable vectors that satisfy a set of equality or inequality constraints. In the lower bound problem the feasible region is the set of decision variable vectors that satisfy the single constraint that the corresponding STP-u is dynamically controllable.

In the following two sections we sketch candidate algorithms that approximate the optimal solution to the lower bound problem. We then return to the formulation of the problem as an optimization problem and suggest ways to convert it to a form suitable for classical optimization techniques in a way that approximates the problem we are trying to solve.

Binary Search for Loosest Bounds

In our first algorithm, we perform binary search for the bounds on the uncontrollable intervals that are as loose as possible while still guaranteeing dynamic controllability. The basic algorithm is as follows:

1. Given PSTP $< V_C, E, V_U, Par, \mathcal{P} >$, construct a corresponding STP-u $S < V'_C, E', V'_U, C >$ where $V'_C = V_C$, $V'_U = V_U$, $E' = E$, and $C = l_X \leq X - Par(X) \leq u_X$ for each $X \in V_U$, where l_X and u_X are initialized to include most of the mass of probability of p_X. (For example, l_X might be the mean minus three standard deviations, while u_X might be the mean plus three standard deviations.)
2. Let ϵ be a small threshold value, and let $F = 1$.
3. While (S is not dynamically consistent) and ($F > \epsilon$)
4. Begin
5. If S is not dynamically controllable
6. $F = F/2$
7. Reduce all contingent edges in S by a factor of F
8. Else
9. $F = 3F/2$
10. Increase all contingent edges in S by a factor of F
11. End If
12. End While
13. Return S.

Note that this algorithm assumes that the underlying STP-u can eventually be made consistent by shrinking the bounds on the uncontrollable events far enough: i.e., if the time points of the uncontrollables could be pinned down, the network would be executable. Also, we have made an arbitary decision about the rate at which we modify the size of the intervals, reducing them by a half when the network is not dynamically controllable, and increasing them by a half when it is. To achieve faster convergence, we may want to vary these values.

This basic algorithm can be improved in several ways. First, when an STP-u is not dynamically uncontrollable, this

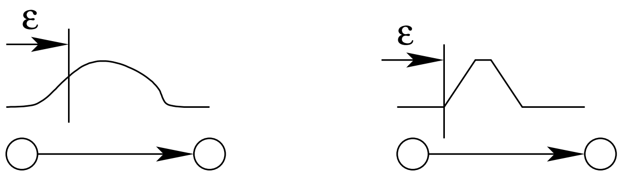


Figure 3: Two uncontrollable events with different distributions.

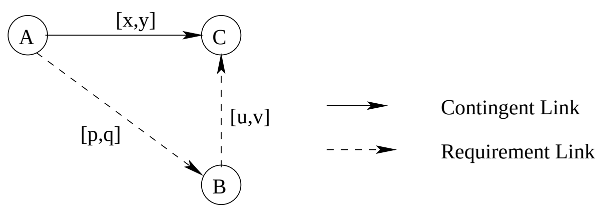


Figure 4: Triangular Networks (Morris, Muscettola, & Vidal 2001) .

may be due only to some, and not all, of the uncontrollable events. It may be possible to identify which uncontrollable events are to blame while running the STP-u controllability algorithm and then to modify the above algorithm so that only the edges incident upon culpable events are reduced. Second, the above algorithm does not take account of the fact that the PSTP explicitly models the probability distribution associated with each uncontrollable event. Instead, it reduces the time intervals associated with all events equally. An improved appropriate extension would be to reduce the bounds proportionally to the probability mass associated with each interval. For example, if one contingent interval has a distribution with very wide variance, while another has a much steeper distribution with narrower variance, we would prefer to place tighter bounds on the first– or, put otherwise, shrink the first interval more–than the second, because that would result in less reduction in the overall probability mass of the uncontrollable events modeled. (See Figure 3.)

Iterative Tightening

The Binary Search approach employs the dynamical controllability algorithm as a black box. The Iterative Tightening approach on the other hand modifies the dynamic controllability algorithm.

The Iterative Tightening first converts the PSTP to an STP-u by calculating loose bounds for the uncontrollables in a way that contain most (or all if possible) of the mass of probability (exactly as the Binary Search algorithm). Then, it runs a modified dynamic controllability algorithm: instead of stopping as soon as it is discovered that the STP-u is not controllable, the algorithm relaxes the problem (by tightening the bounds) and continues.

The dynamic controllability algorithm checks each triplet of variables A, B and C, where C is uncontrollable (as shown in Figure 4. The constraints (requirement links) $A \rightarrow B$ and $B \rightarrow C$ may be explicit or implicit constraints. The algorithm then imposes a set of additional constraints that ensure the existance of an execution strategy. If the propagation of these constraints does not result in a "squeeze" of the contingent link, the STP-u is controllable.

The Iterative Tightening algorithm employs the same strategy. It selects a triangle of variables to work on and imposes the constraints determined by the controllability algorithm. However, if the propagation of these constraints squeeze any other contingent link, instead of stopping, the algorithm tightens the contingent link to these new bounds. Obviously, this algorithm will not return an execution strategy that works for all possible occurences of the uncontrollables of the original bounds, but only for the final tightened bounds.

The algorithm is as follows:

1. Given PSTP $< V_C, E, V_U, Par, \mathcal{P} >$, construct a corresponding STP-u $S < V'_C, E', V'_U, C >$ where $V'_C = V_C$, $V'_U = V_U$, $E' = E$, and $C = l_X \leq X - Par(X) \leq u_X$ for each $X \in V_U$, where l_X and u_X are initialized to include most of the mass of probability of p_X.
2. Non-deterministically CHOOSE a triangle of variables A, B, C where C is uncontrollable.
3. Impose the constraints determined by the dynamic controllability algorithm.
4. Propagate the constraints as in the controllability algorithm, but allow requirement links to be squeezed.
5. Repeat until the lower bound that corresponds to the current STP-u is high enough, or the last constraint propagation did not change the STP-u.

In Iterative Tightening the order of consideration of each triangle matters. When a triangle A, B, C and a contingent link $A \rightarrow C$ is selected and appropriate constraints are imposed to ensure controllability, essentially the algorithm creates an execution strategy that works for all cases where C occurs within the current bounds given for $A \rightarrow C$. Propagation of these constraints may require that other uncontrollables occur within a tighter interval to allow for this strategy to work.

For example, suppose that there are two contingent links $Y \rightarrow X$ and $A \rightarrow B$. Selecting a triangle that involves the first one may cause the bounds on the second one to shrink considerably in order to allow the execution strategy to work with all possible occurences of X. If B's probability distribution has heavy tails, that means that a significant mass or probability will be excluded from the calculation of the lower bound. If instead the second triangle is selected first, its bounds will not be tightened but may cause the bounds on the first link to shrink. If however, the distribution of X has smaller variance, then shrinking the bounds will not exclude as much probability mass and the algorithm will return a tigher lower bound.

Possible variants of the algorithm include a backtracking search where different choices of triangles are made in a search for the STP-u that provides the tighest lower bound on the probability of successful execution.

Non-Linear Optimization Solutions

In this section, we explore the possibility of solving the lower bound problem with optimization methods. While the objective function is suitable for typical optimization methods, the constraints are not. We now attempt to cast the constraint of the resulting STP-u being dynamically controllable, as a set of inequalities, which would allow non-linear optimization techniques to be used.

Consider the triangle of variables and edges of Figure 4, where the edge $A \rightarrow C$ is a contingent link. According to (Morris, Muscettola, & Vidal 2001) the triangle is dynamically controllable if any of the following three conditions hold:

1. $v < 0$, and the triangle is pseudo-controllable.
2. $u \geq 0$, $B - A \subseteq [y - v, x - u]$, and the triangle is pseudo-controllable.
3. $v \geq 0$, $u < 0$, $y - v \leq x$, and the triangle is pseudo-controllable.

Pseudo-controllability denotes the fact that the bounds $[x, y]$ are not "squeezed" by the contraints of the triangle, or in other words that $[x, y] \subseteq [-d_{CA}, d_{AC}]$. Recall that the dynamic controllability algorithm determines whether there is a way to execute the plan no matter when the uncontrollables occur. The interval $[-d_{CA}, d_{AC}]$ is the interval dictated by the constraints in the plan: any time within that interval participates in at least one solution of the constraints. The interval $[x, y]$ is derived from the contingent link and is a constraint on Nature. Thus, if there are values of $[x, y]$ that do not participate in any solution of the constraints, Nature may select one of these values forbidding the completion of the plan in a way that satisfies the constraints.

Apart from the three cases above for when a triangle is dynamically controllable there is a fourth case. Thus, the above three cases together are sufficient but not necessary conditions. The fourth case involves accepting a ternary and disjunctive constraint that is called a wait, which we will ignore for the moment.

An STP-u is dynamically controllable if all such triangles in the network are dynamically controllable. These three cases direct the design of the our algorithm.

1. Given a PSTP $< V_C, E, V_U, Par, \mathcal{P} >$.
2. Define a non-linear optimization problem with decision variables x_i, y_i for every uncontrollable, objective function $\prod_i P_C(x_i \leq t \leq y_i)$, and inequality constraints S as defined below.
3. Initialize $S \leftarrow E$.
4. For each triple of variables A_i, B_i, C_i as in Figure 4, where C_i is uncontrollable:
   - If $v_i < 0$, then no extra constraint needs to be added.
   - If $u_i \geq 0$, then $S \leftarrow S \cup \{B_i - A_i \subseteq [y_i - v_i, x_i - u_i]\}$
   - Else, $S \leftarrow S \cup \{y_i - v_i \leq x_i\}$.
5. Solve the optimization problem.

The solution to the optimization problem will return a set of values for the decision variables for which all the constraints are satisfied. By construction, satisfying all these constraints implies that the corresponding STP-u is dynamically controllable. This is because each such triangle falls into one of the three cases above.

In addition, in any solution of the optimization problem the triangles are pseudo-controllable. This is because any bounds x, y selected by the optimization for a contingent link $A \rightarrow C$ are as squeezed as possible: $y \leq d_{AC}$ because if $y > d_{AC}$ then y is outside the feasible set imposed by the constraints of the optimization problem.

Intuitively, the algorithm is free to select any x, y bounds on contingent links and impose any constraints on Nature desired. Of course Nature may not observe these constraints but we can calculate the probability that she will and obtain the desired lower bound.

Notice that since the three cases are sufficient but not necessary it is conceivable that this is not the tightest lower bound on the probability that can be achieved using this kind of approach (i.e., by translating to an STP-u). Specifically, there is a fourth case that we omitted from consideration: it involves a disjunctive and ternary constraint called a wait on C. For example $wait < C, 5 >$ means that one should wait to execute B until 5 time units have passed after A has been executed or C has been observed. A non-linear optimization algorithm that takes into consideration this case may be able to further increase the bounds $[x, y]$ to include more mass of probability. It is currently unknown how significant is this case in practice and how much looser than optimal is a lower bound that is achieved by ignoring this case.

We now consider a specific class of probability density functions and the corresponding optimization problems they give rise to.

Optimization for Uniform Distributions

Let us denote with p_i the pdf of the ith uncontrollable variable and suppose that all probability distributions are uniform, that is $p_i(t) = \frac{1}{b_i - a_i}$, when $t \in [a_i, b_i]$ and zero outside this interval.

If $p(t)$ is uniform in $[a, b]$, then

$$P(x \leq t \leq y) = \frac{\min(b, y) - \max(a, x)}{b - a}$$

for $x \leq y$. In Figure 5 this is justified pictorially where the probability density of a uniform p_i within the bounds $[a, b]$ is shown. $P(x \leq t \leq y)$ is the area above the intersection of $[a, b]$ and $[x, y]$.

Instead of maximizing the actual probability of successful execution, we can maximize its logarithm.

$$\max \log \prod_i P_i(x_i \leq t \leq y_i)$$

which is equal to

$$\max \sum_i \log P_i(x_i \leq t \leq y_i)$$

By utilizing the fact that P_i's are uniform, this is equivalent to

$$\max \sum_i \log \frac{\xi_i - \sigma_i}{b_i - a_i},$$

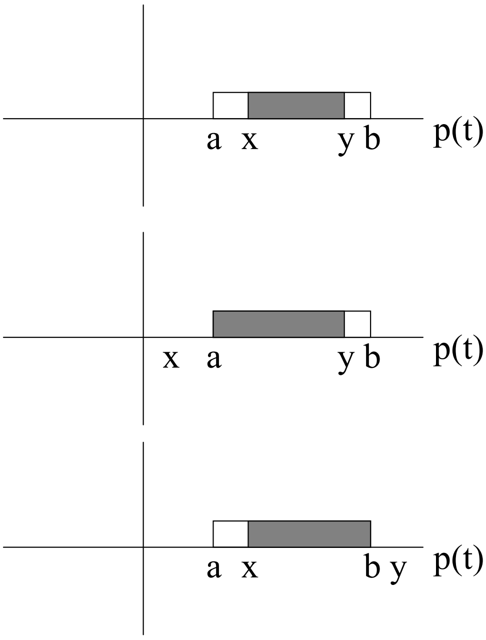


Figure 5: The mass of probability of a random variable uniformly distributed within $[x, y]$ occuring within $[a, b]$ is the area above the intersection of the intervals.

where $\sigma_i = \max(x_i, a)$ and $\xi_i = \min(y_i, b)$. In turn, this gives:

$$\max(\sum_i \log(\xi_i - \sigma_i) - \sum_i \log(b_i - a_i))$$

The last sum is a constant term and can be dropped from the objective function during optimization (but is required to compute the final bound on the probability). So the objective function becomes

$$\max \sum_i \log(\xi_i - \sigma_i),$$

or equivalently

$$\min \ -\sum_i \log(\xi_i - \sigma_i)$$

The constraints of this optimization problem are given by Steps 3 and 4 in the algorithm of the previous section. That is, they are the difference constraints among the PSTP variables (controllable and uncontrollable ones) union the constraints required to guarantee the resulting STP-u is dynamically controllable. In addition to these constraints however, we need to add that $\sigma_i = \max(x_i, a_i), \xi_i = \min(y_i, b_i)$, and that $x_i \leq y_i$.

The max and min functions present problems to most optimization algorithms. Fortunately, in this case we can substitute $\xi_i = \min(y_i, b_i)$ with the constraints $\xi_i \leq b_i, \xi_i \leq y_i$, and $\sigma_i = \max(x_i, a_i)$ with $\sigma_i \geq a_i$, and $\sigma_i \geq x_i$. This is because, in order to maximize the objective function the ξ_i should be as large as possible; so the optimization engine will increase the ξ_i until at least one of the equalities $\xi_i = b_i, \xi_i = y_i$ holds, in which case $\xi_i = \min(y_i, b)$. A similar argument holds for σ_i.

The feasible region defined by these inequality constraints is convex. Additionally, the objective function is twice differentiable everywhere except from the boundary where $\sigma_i = \xi_i$. So the objective function is twice differentiable in the interior of the feasible region.

Let us calculate the second derivative of each term in the sum. Define $f(\xi, \sigma) = -\log(\xi - \sigma)$. Then, $\nabla f(\xi, \sigma) = [-(\xi - \sigma)^{-1}, (\xi - \sigma)^{-1}] = [-a, a]$, for $a = (\xi - \sigma)^{-1}$. The Hessian is $H = \nabla^2 f(\xi, \sigma) = \begin{bmatrix} a^2 & -a^2 \\ -a^2 & a^2 \end{bmatrix}$. H is semidefinite positive because the eigenvalues are non-negative. The eigenvalues λ solve $\det(\lambda I - \nabla^2 f(\xi, \sigma)) = \lambda^2 - 2a^2\lambda = 0$, i.e., $\lambda = 0$ or $\lambda = 2a^2 > 0$ for $\sigma < \xi$. Thus, function f defined on a convex set (when $\sigma \leq \xi$) has a positive semidefinite $\nabla^2 f$ in the interior and thus is convex (provided the interior is non-empty). The objective function, as a sum of such convex functions is also convex.

Convex optimization problems have a unique minimum and in general, are relatively easily solved with modern optimization software. We are currently considering other families of probability distributions such as exponential or normal distributions.

Discussion

The recent literature on planning has shown a growing interest in handling more and more realistic problems, and along with that has come a concern with various types of uncertainty. In this paper, our focus has been on temporal uncertainty: uncertainty about the time at which certain exogenous, or "uncontrollable" events will occur. Significantly, domain experts typically know more about such events than just the interval of time during which they will occur–they often know a probability distribution over the interval of occurrence. Yet the most successful formalism for planning with temporal uncertainty, the STP-u's, don't allow one to exploit that knowledge. Instead, the domain expert must specify fixed bounds on the time during which each uncontrollable event must happen. If he sets the bounds too narrowly, he may produce a plan that is dynamically controllable, but that nonetheless fails, because the uncontrollable event occurs outside the modeled time. If he sets them too widely, he may produce a plan that is not dynamically controllable. And execution strategies only exist for dynamically controllable plans; if an STP-u is not dynamically controllable, there is no effective means of deciding when to execute the controllable actions in it.

This poses a real problem for the designer of a planner dealing with temporal uncertainty. It is difficult to know how to set the bounds on the uncontrollable events; and if the bounds are set too widely, it is impossible to assess the probability that the plan can nonetheless be legally executed, and thus, impossible to make a principled decision about whether to adopt this plan or whether to search further for an alternative.

What we would like to do is to enable the domain expert to specify bounds on the uncontrollable events that are “as wide as possible”: by this we mean that they maximize the probability that the events will in fact occur during the modeled bounds, subject to the constraint that the network is dynamically controllable. When the bounds are set in this fashion, there are two results: first, we have an evaluation of the probability that the plan can be legally executed, which can be used to decide whether to accept or reject it, and second, if it is accepted, then the network with the bounds thus set can serve as the basis of a legal execution strategy.

Because STP-u’s do not include information about the probability distribution of the timing of uncontrollable events, we presented a generalization of them, called Probabilistic Simple Temporal Problems (PSTPs). Unfortunately, given an PSTP, it is difficult to compute an exact probability of legal execution. What we can do, however, is bound the probability, both from above and below. The upper bound simply provides a way of rejecting a plan if it is not below a given threshold. The lower bound is arguably more interesting, as it is not only what gives a way of accepting a plan, when it is above a threshold, but is also what allows one to approximate the widest possible bounds.

We presented three alternative algorithms for approximating the widest possible bounds. The first performs binary search for a value v that represents the minimal proportion by which all the uncontrollable intervals need to be reduced to achieve a dynamic controllability. The second runs the dynamic controllability algorithm with the modification that it does not exit as soon as controllability is deemed impossible. Instead, it shrinks the intervals appropriately, relaxing the initial problem, until controllability is achieved. The third algorithm takes a very different approach, attempting to reduce the problem to one of non-linear optimization. It approximates the set of controllable STP-u’s with a set of inequality constraints. The next major step in this work it to implement these three algorithms and conduct both experimental analyses of their performance in terms of computational efficiency and quality of lower bounds returned. Additionally, it will be important to integrate work on temporal uncertainty of the kind described in this paper with work on causal uncertainty, such as that discussed in (Tsamardinos, Vidal, & Pollack 2003).

References

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

Morris, P.; Muscettola, N.; and Vidal, T. 2001. Dynamic control of plans with temporal uncertainty. In *Proceedings of the 17th International Joint Conference on A.I. (IJCAI-01)*. Seattle (WA, USA): Morgan Kaufmann, San Francisco, CA.

Muscettola, N.; Nayak, P. P.; Pell, B.; and Williams, B. C. 1998. Remote agent: To boldly go where no ai system has gone before. *Artificial Intellience* 103:5–47.

Muscettola, N.; Morris, P.; and Tsamardinos, I. 1998. Reformulating temporal plans for efficient execution. In *Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning (KR-98)*, 444–452.

Smith, D.; Frank, J.; and Jónsson, A. 2000. Bridging the gap between planning and scheduling. *Knowledge Engineering Review* 15(1).

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

Tsamardinos, I.; Morris, P.; and Muscettola, N. 1988. Fast transformation of temporal plans for efficient execution. In *Proceedings of the 15th National Conference on Artificial Intelligence*.

Tsamardinos, I.; Vidal, T.; and Pollack, M. E. 2003. CTP: A new constraint-based formalism for conditional, temporal planning. *Constraints (to appear), Special Issue on Planning* 8(4).

Tsamardinos, I. 2001. *Constrained-Based Temporal Reasoning Algorithms with Applications to Planning*. Ph.D. Dissertation, University of Pittsburgh.

Tsamardinos, I. 2002. A probabilistic approach to robust execution of temporal plans with uncertainty. In *Proc. of the 2nd Greek National Conference on Artificial Intelligence*.

Vidal, T., and Fargier, H. 1997. Contingent durations in temporal CSPs: from consistency to controllabilities. In *Proceedings of the 4th International Workshop on Temporal Representation and Reasoning (TIME-97)*, 78–85.

Vidal, T., and Ghallab, M. 1996. Dealing with uncertain durations in temporal constraint networks dedicated to planning. In *Proceedings of the 12th European Conference on Artificial Intelligence (ECAI-96)*, 48–52.