On the Partial Observability of Temporal Uncertainty

Michael D. Moffitt
Department of Electrical Engineering and Computer Science
University of Michigan
Ann Arbor, MI 48109
mmoffitt@eecs.umich.edu

Abstract

We explore a means to both model and reason about partial observability within the scope of constraint-based temporal reasoning. Prior studies of uncertainty in Temporal CSPs have required the realization of all exogenous processes to be made entirely visible to the agent. We relax this assumption and propose an extension to the Simple Temporal Problem with Uncertainty (STPU), one in which the executing agent is made aware of the occurrence of only a subset of uncontrollable events. We argue that such a formalism is needed to encode those complex environments whose external phenomena share a common, hidden source of temporal causality. After characterizing the levels of controllability in the resulting *Partially Observable STPU* and various special cases, we generalize a known family of reduction rules to account for this relaxation, introducing the properties of *extended contingency* and *sufficient observability*. We demonstrate that these modifications enable a polynomial filtering algorithm capable of determining a local form of dynamic controllability; however, we also show that there do remain some instances whose global controllability cannot yet be correctly identified by existing inference rules, leaving the true computational complexity of dynamic controllability an open problem for future research.

Introduction

The problem of uncertainty has gained considerable attention in the field of constraint-based temporal reasoning. In this line of research, the time points of traditional temporal networks (Dechter, Meiri, & Pearl 1991) are divided into sets of *controllable events* (whose values are determined by the execution agent) and *uncontrollable events* (whose values are selected by an external force referred to as "Nature"), transforming the problem of consistency into one of *controllability*. Significant progress has been made in recent years toward increasing the efficiency of reasoning about temporal uncertainty, as well as simplifying its mathematical and structural foundations (Morris & Muscettola 2005; Morris 2006).

Despite these advances, prior literature has typically operated upon an implicit assumption about the means by which uncertainty is resolved dynamically (i.e., in an online setting). In particular, the vast majority of formalisms – including the Simple Temporal Problem with Uncertainty (STPU) (Vidal & Fargier 1999) and its variants – establish a direct correspondence between the *observation* of an event and its actual *execution*. Although early work acknowledged the limitations in a model of full observability (Muscettola *et al.* 1998), only recently have efforts been made to relax this requirement in formal constructions. For instance, the so-called Generalized STPU (Moffitt & Pollack 2007) models cases in which the agent becomes informed of the temporal locality of an event earlier than its exact realization. The knowledge obtained is incomplete, as the range of an uncertain duration may be only partially reduced to one of finitely-many subintervals (a process termed "partial shrinkage"). Such an extension is needed to enhance the agent's ability to react to the foreknowledge of uncontrollable events.

In this paper, we propose an alternate means to model and reason about partial observability within the context of temporal uncertainty, one that captures scenarios where the agent's ability to observe external phenomena is strictly diminished. We introduce an extension to the STPU called the *Partially Observable STPU*, in which the executing agent is made aware of the occurrence of only a subset of uncontrollable time points. We argue that such a formalism is necessary to model many complex environments whose unpredictable processes share a common, hidden source of temporal causality. We motivate the need for this richer expressive power by constructing practical, real-world examples that ultimately cannot be captured by existing representations. We then formally define the Partially Observable STPU and re-characterize the various levels of controllability in light of these extensions. We also note a special case where dynamic and strong controllability become semantically equivalent. We present a generalization to a family of reduction rules to account for the presence of unobservable events, relying on properties that we refer to as *extended contingency* and *sufficient observability*. Finally, we show that while these modifications enable a local guarantee of controllability, the enhanced inference remains too weak to globally determine dynamic controllability in all cases, leaving open its true computational complexity.

Background

Simple Temporal Problems

The *Simple Temporal Problem* (Dechter, Meiri, & Pearl 1991) is defined by a pair $\langle X, E\rangle$, where each X_i in X designates a time point and each E_{ij} in E is a constraint of the form

$$l_{ij} \leq x_i - x_j \leq u_{ij}$$

with x_i, $x_j \in X$ and l_{ij}, $u_{ij} \in \Re$. An STP is said to be *consistent* if there exists a *solution* $S : X \rightarrow \Re$ that satisfies all constraints in E.

Each STP has an associated graph-based encoding, which contains a node for each time point and a directed edge for each inequality having weight u_{ij} or $-l_{ij}$ (depending on its direction). For the STP to be consistent, it is both necessary and sufficient that its corresponding graph contain no negative cycles; this can be determined in polynomial time by computing its All Pairs Shortest Path matrix and examining the entries along the main diagonal.

Simple Temporal Problems with Uncertainty

The STP models situations in which the agent in charge of plan execution has full control over the values of all time points. The *Simple Temporal Problem with Uncertainty* (STPU) (Vidal & Fargier 1999) extends the STP by relaxing this basic assumption. Specifically, the STPU is defined as a tuple $\langle X_C, X_U, E, C\rangle$, where:

- X_C and X_U are sets of *controllable* and *uncontrollable* time points, respectively. Their union, $X_C \cup X_U$, forms an entire set X of time points.
- E is a set of *requirement links*, where each E_{ij} is of the form $l_{ij} \leq x_i - x_j \leq u_{ij}$ (written $x_i \stackrel{[l_{ij},u_{ij}]}{\longrightarrow} x_j$).
- C is a set of *contingent links*, where each C_{ij} is of the form $l_{ij} \leq x_i - x_j \leq u_{ij}$ and $x_i \in X_U$[1] (written $x_i \stackrel{[l_{ij},u_{ij}]}{\Longrightarrow} x_j$).

The contingent links in the STPU can be regarded as representing causal processes whose durations are uncertain, and thus their endpoints (the uncontrollable time points) are determined by some external force. The remaining time points are in the control of the agent, who is charged with the task of assigning them in such a way as to satisfy the requirement links.

It is often convenient to refer to a *projection* p of the STPU, which is simply an STP obtained by replacing the interval of each contingent link $[l, u]$ with a particular fixed bound $[b, b]$ where $l \leq b \leq u$. A *schedule* T is defined as a mapping

$$T : X \rightarrow \Re$$

where $T(x)$ is the *time* of time point x. A schedule is deemed *consistent* if it satisfies all links. The *prehistory* of a time point x with respect to a schedule T, denoted $T\{\prec x\}$, specifies the durations of all contingent links that finish prior to x; it can be determined directly from any schedule.

[1]As in prior work, we assume $0 < l_{ij} < u_{ij} < \infty$ for each contingent link, and that contingent links do not share endpoints.

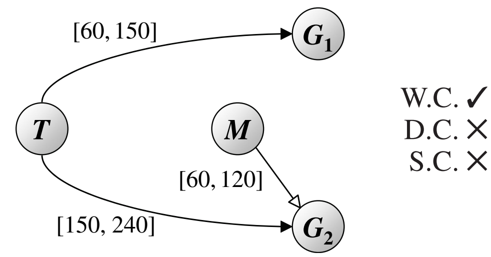

Figure 1: The network corresponding to Example 1.

Finally, we define an *execution strategy* S as a mapping:

$$S : \mathcal{P} \rightarrow \mathcal{T}$$

where $\mathcal{P}$ is the set of all projections, and $\mathcal{T}$ is the set of all schedules. An execution strategy S is *viable* if $S(p)$, henceforth written S_p, is consistent with p for each projection p.

Controllability of the STPU

With the addition of uncontrollable events and contingent links in the STPU, the previously defined notion of consistency for the STP is no longer sufficient. Instead, one must consider various flavors of *controllability*. For illustration, we put forth the following example taken from the context of Autominder (Pollack *et al.* 2003), an orthotic system designed to help persons with cognitive decline to perform routine activities:

Example 1: An elderly woman (Mrs. Smith) is expecting two family members from out of town to visit for dinner, and each will be arriving separately. The first has called to say that he will be arriving in 1 to $2\frac{1}{2}$ hours; the second guest is running late, and will not arrive until $2\frac{1}{2}$ to 4 hours from now. Mrs. Smith must take her medication between one and two hours prior to dinner (which will commence as soon as the second guest arrives). □

The STPU can model this scenario with the network depicted graphically in Figure 1. The event T is a temporal *reference point* representing the current wall-clock time. Events G_1 and G_2 represent the arrival times of the first and second guests, respectively, and each lies at the conclusion of a contingent link. Finally, event M represents the time at which medication must be taken. This time point is constrained to occur one to two hours before the second guest's arrival.

Weak Controllability An STPU is said to be *Weakly Controllable* if there is a viable execution strategy; in other words, for every possible projection, there must exist a consistent solution. Our example is indeed weakly controllable; if we happened to know when the second guest will arrive, we could obtain a consistent solution by setting M to $G_2 - x$ for any x between 60 and 120 minutes. Unfortunately, the property of weak controllability is not especially useful, as the agent cannot likely see into the future.

Strong Controllability An STPU is *Strongly Controllable* if there is a viable execution strategy S such that

$$S_{p1}(x) = S_{p2}(x)$$

for each controllable time point x and all projections $p1$ and $p2$, i.e., there exists a single consistent (conformant) solution that satisfies every possible projection. Our example is clearly not strongly controllable; for instance, there is no consistent assignment for M when $G_2 - T = 150$ that will also work for the case when $G_2 - T = 240$.

Dynamic Controllability The most interesting and useful type of controllability is that of dynamic controllability, a concept that exploits the temporal nature of plan execution. Specifically, an STPU is said to be *Dynamically Controllable* if there is a viable execution strategy S such that:

$$S_{p1}\{\prec x\} = S_{p2}\{\prec x\} \Rightarrow S_{p1}(x) = S_{p2}(x)$$

for each controllable time point x and all projections $p1$ and $p2$. In other words, there exists a strategy that depends on the outcomes of only those uncontrollable events that have occurred in the past (and not on those which have yet to occur). Our example is not dynamically controllable; without prior knowledge of the value of G_2, there is no way to set M in order to guarantee that medication is not taken too early or too late.

Of the three types of controllability, dynamic controllability has been the most extensively studied, and was recently shown to be computable in $O(N^4)$-time (Morris 2006).

Shared Temporal Causality

Observe that in Example 1, Nature is free to schedule the arrival of both guests independently; hence, the occurrence of event G_1 offers the agent no useful information regarding the eventual execution of G_2. However, under certain circumstances, it may be reasonable to expect that some combinations of delays will not occur; for instance, suppose that a portion of the uncertainty in the guests' travel time is due to unknown traffic conditions. If both guests are to travel along the same highway or interstate, they are likely to encounter similar traffic patterns that affect their cumulative delays. It is with such shared causality in mind that we propose a slight modification to our running example:

Example 2: Consider Example 1; however, we now know that the uncertainty in earlier estimates was due in part to variable traffic conditions. Taking into account routes that both guests will take, we can expect the second guest to arrive no earlier than $\frac{1}{2}$ hour (and no later than $2\frac{1}{2}$ hours) after the first. □

Note the difference between this and the original example, where arrivals could be separated from anywhere between 0 and 3 hours. One possible encoding of this new information is shown in Figure 2. Here, we model the co-dependence between the travelers' arrival times by factoring out a common contingent process whose endpoint is labelled C. Whatever duration Nature selects for $T \Rightarrow C$ will be shared by both guests. Appended to the end of this process is a portion of

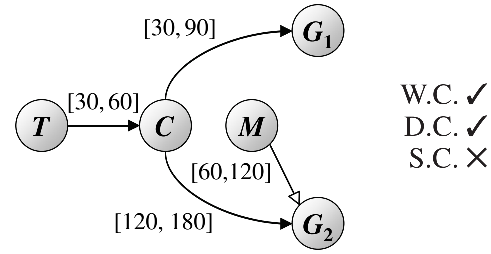

Figure 2: A common causal process has been factored into the contingent link $T \Rightarrow C$.

each of the original contingent links, which contribute separately to the individual delays. We encourage the reader to verify that in this new network, the respective total travel times of the first and second guests are still bounded by the intervals $[60, 150]$ and $[150, 240]$, and that the arrival of the second guest is now guaranteed to occur between $\frac{1}{2}$ and $2\frac{1}{2}$ hours after the first.

Upon examining Figure 2, we see that there is triangular subnetwork that involves the nodes C, M, and G_2. Since M is the only controllable event of interest, analysis of this subgraph will reveal insight into the problem's degree of controllability. Following the logic of (Morris, Muscettola, & Vidal 2001), we conclude that M must precede G_2 (the lower bound the requirement link $M \rightarrow G_2$ is positive). We can thus infer a lower and upper bound on $C \rightarrow M$: $[ub(G_2 - C) - ub(G_2 - M), lb(G_2 - C) - lb(G_2 - M)] = [180 - 120, 120 - 60] = [60, 60]$. Since this interval is nonempty, the network is safe and pseudo-controllable. No further reductions can be applied, and so we conclude that our example is indeed dynamically controllable with the strategy $M = C + 60$.

Hidden Temporal Causality

Unfortunately, there is a subtle fault in the previous line of reasoning, as our analysis did not address a key assumption made in the STPU: that the realization of each contingent link is required to be *observable*.[2] From Mrs. Smith's perspective, the earliest possible knowledge of either of the guests' travels is likely to be postponed until one of the guests actually arrives. In other words, our plan in reality cannot depend on C, since its corresponding causal process is effectively hidden from view.

One can easily imagine other temporal domains in which the execution of some exogenous processes cannot be observed. For instance, consider the Deep Space One (DSl) spacecraft controlled by the New Millennium Remote Agent (NMRA), one of the earliest applications of temporal reasoning with uncertainty (Muscettola *et al.* 1998). In NMRA, plans were shipped to an Executive that, in turn, issued direct commands to control software. The reactive nature of the Executive enabled it to respond in real-time to low-level sensor information. Supposing that some sensors aboard

[2] It is mentioned in (Muscettola *et al.* 1998) that an uncontrollable event may not be observable in practice; however, no formalism has yet been proposed to relax this assumption.

DS1 were to fail (as sensors often do), the agent would be incapable of observing and responding to many external events. Lossy information from distributed sensor networks also poses a significant complication to the Client Modeler in recent extensions to Autominder, preventing it from accurately monitoring the actions and activities of its client.

While the need to deal with partial observability has been well-recognized in the context of classical planning (Rintanen 2004; Bertoli *et al.* 2006; Nance, Vogel, & Amir 2006) and control theory (Cassandra, Kaelbling, & Littman 1994; Mausam & Weld 2006), it has remained all but neglected in literature on constraint-based temporal reasoning, driving us to seek a relaxation to the STPU that can accommodate such extensions.[3]

The Partially Observable STPU

We formally define the *Partially Observable STPU* as a tuple $\langle X_C, X_O, X_U, E, C_O, C_U \rangle$, where:

- X_C, X_O, and X_U are sets of *controllable*, *observable uncontrollable*, and *unobservable uncontrollable* time points, respectively.
- E is a set of *requirement links*, where each E_{ij} is of the form $l_{ij} \leq x_i - x_j \leq u_{ij}$ (written $x_i \overset{[l_{ij},u_{ij}]}{\longrightarrow} x_j$).
- C_O is a set of *observable contingent links*, where each C_{ij} is of the form $l_{ij} \leq x_i - x_j \leq u_{ij}$ and $x_i \in X_O$ (written $x_i \overset{[l_{ij},u_{ij}]}{\Longrightarrow} x_j$).
- C_U is a set of *unobservable contingent links*, where each C_{ij} is of the form $l_{ij} \leq x_i - x_j \leq u_{ij}$ and $x_i \in X_U$ (written $x_i \overset{[l_{ij},u_{ij}]}{\longmapsto} x_j$).

As in the STPU, we characterize each constraint as being either a *requirement* or *contingent* link; however, we also make a further distinction in the contingent case, classifying links as being either *observable* or *unobservable*. Uncontrollable time points are partitioned in a parallel fashion.

A proper encoding of Example 2 is now finally possible. We need only to label the edge between T and C as being an unobservable contingent link, with all other contingent links being observable. This relationship is shown as a dotted arrow in Figure 3.

Characterizing Controllability

Although the annotation of a contingent link as being unobservable reduces the agent's ability to react to Nature, it has no effect on those levels of controllability that do not depend on their mutual interaction. For instance, any conformant plan corresponding to a *strongly controllable* STPU remains conformant even if all constraints were to be tagged as unobservable. The same holds for any set of solutions that comprise a *weakly controllable* STPU. Thus, the definitions for both of these degrees of controllability are taken without change from their fully observable counterparts.

[3]Although a limited flavor of prior and partial observability is permitted by the Generalized STPU (Moffitt & Pollack 2007), immediate and full observability is still assumed to be simultaneous with the execution of an uncontrollable event.

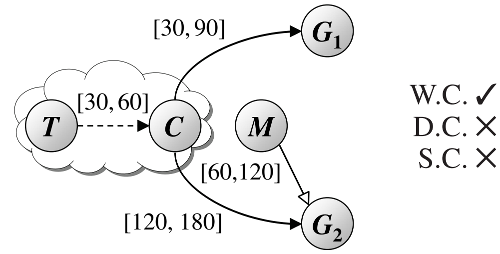

Figure 3: A dotted arrow signifies an unobservable contingent link $T \mapsto C$.

There is, however, a significant deviation in what it means for a Partially Observable STPU to be dynamically controllable. Recall that a dynamic strategy depends only on a schedule's *prehistory*: the set of durations for all contingent links that finish prior to a given event. Since the agent cannot rely on values that will not be observed, we must revise this definition: in particular, we say that the *prehistory* of a controllable[4] event x with respect to a schedule T, still denoted $T\{\prec x\}$, specifies the durations of all *observable* contingent links that finish prior to x; durations of unobservable contingent links are explicitly omitted from this set.

With this augmented definition of a prehistory, dynamic controllability is characterized as naturally as it was for the STPU. A Partially Observable STPU is said to be *Dynamically Controllable* if there is a viable execution strategy S such that:

$$S_{p1}\{\prec x\} = S_{p2}\{\prec x\} \Rightarrow S_{p1}(x) = S_{p2}(x)$$

for each executable time point x and all projections $p1$ and $p2$. In other words, there exists a strategy that depends on the outcomes of only those uncontrollable, observable events that have occurred in the past.

Special Cases

We note that the Partially Observable STPU admits some special cases that subsume previous formalisms and unify the various notions of controllability. For instance, it can be easily seen that if $C_U = \oslash$, the Partially Observable STPU reduces to a traditional STPU. Similarly, if we restrict $C_U = C_O = \oslash$, we obtain an ordinary STP.

A more interesting case occurs when we have only $C_O = \oslash$. Here, the agent is entirely incapable of observing any uncertain processes, and so any dynamic plan must be conformant by definition. Hence, we can establish dynamic controllability if and only if we can also establish strong controllability. The complexity of D.C. in this special case is thus equal to that of S.C. (i.e., strongly polynomial), and a conformant solution can be constructed using previously established algorithms. Our main focus, however, is to improve the power of inference in the more general case, where $C_U \neq \oslash, C_O \neq \oslash$.

[4]We presume that Nature may schedule its own (uncontrollable) time points with full observability of all events.

(UPPER-CASE REDUCTION)
$A \overset{B:x}{\longleftarrow} C \overset{y}{\longleftarrow} D$ adds $A \overset{B:(x+y)}{\longleftarrow} D$

(LOWER-CASE REDUCTION) **If $\mathbf{x < 0}$,**
$A \overset{x}{\longleftarrow} C \overset{c:y}{\longleftarrow} D$ adds $A \overset{x+y}{\longleftarrow} D$

(CROSS-CASE REDUCTION) **If $\mathbf{x < 0}$,** $B \neq C$,
$A \overset{B:x}{\longleftarrow} C \overset{c:y}{\longleftarrow} D$ adds $A \overset{B:(x+y)}{\longleftarrow} D$

(NO-CASE REDUCTION)
$A \overset{x}{\longleftarrow} C \overset{y}{\longleftarrow} D$ adds $A \overset{x+y}{\longleftarrow} D$

Figure 4: The original primary reduction rules for the STPU.

Algorithms for Dynamic Controllability

When the STPU was originally conceived, the computational complexity of dynamic controllability was speculated to be intractable. Nevertheless, work on improving the power of graph-based inference continued (Morris & Muscettola 1999; 2000), until it was shown that dynamic controllability for the STPU can be determined in pseudo-polynomial time (Morris, Muscettola, & Vidal 2001). More recently, an improved strongly polynomial-time algorithm has been introduced (Morris & Muscettola 2005) that applies a family of reduction rules to a variation on the traditional distance graph.

Original Reduction Rules

Just as the use of shortest path algorithms can determine the consistency of an STP, a graph-based formulation has also been characterized for the STPU, called the *labelled distance graph* (Morris & Muscettola 2005). As with the conventional distance graph, it contains a pair of edges $A \overset{y}{\longrightarrow} B$ and $A \overset{-x}{\longleftarrow} B$ for each requirement link $A \overset{[x,y]}{\longrightarrow} B$. For a contingent link $A \overset{[x,y]}{\Longrightarrow} B$, these same two edges are added, as well as an additional pair of so-called *labelled edges* of the form $A \overset{b:x}{\longrightarrow} B$ and $A \overset{B:-y}{\longleftarrow} B$.

Tightening of edges in the labelled distance is achieved using a family of reduction rules; the four primary rules are shown in Figure 4.[5] A strongly polynomial-time algorithm for dynamic controllability is obtained by repeatedly applying these rules until a certain cutoff (bounded quadratically by the number of events) is reached. To ensure consistency of the temporal network, each step is preceded by an AllMax projection that deletes all lower-case edges, removes labels from all upper-case edges, and applies an APSP calculation. The algorithm has total complexity of $O(N^5)$.

Augmented Reduction Rules

An unobservable contingent link is strictly less amenable to dynamic controllability than an equivalent observable contingent link; hence, if a contingent link's classification is ignored, the previous reduction rules are necessary (but not sufficient) to ensure dynamic controllability (i.e., the algorithm will always produce correct "No" answers, but not necessarily correct "Yes" answers). We must expand the range of cases where reductions can be safely applied if fewer problems are to incorrectly pass the test for D.C.

The assumption of full observability built into the original reduction rules takes a relatively subtle form, and stems from the clause "If $x < 0$" tagged onto the lower-case and cross-case reduction rules (which we have replicated in Figure 5 as the function **Must-Precede()**). If this condition is true, then A must necessarily precede C, implying that the value of A therefore cannot depend on C. This triggers the addition of a temporal constraint to decrease the reduced distance from D to A based on the contingent link that C concludes.

Of course, whether A must occur prior to C is entirely irrelevant; what truly matters is whether A must occur before the value of C will become known to the agent. Since the Partially Observable STPU no longer couples observation with execution, we must revise this condition to check whether A necessarily precedes the *observation* of C. Following this reasoning, it is tempting to insert the following line into **Must-Precede()**:

1.5. If ($C \in X_U$) *return true;*

After all, the value of C is never directly observed, and thus we may need to assign a value to A independently of C (as in the case of true precedence). However, recall that the veil of unobservability shields only the agent (and not Nature) from the occurrence of uncontrollables. Hence, it may still be possible to infer the value of C – or, more precisely, a sufficient range on its value – in hindsight, given a particular local topology of observability within the network.

For instance, consider the network in Figure 6, in which an unobservable contingent link $D \mapsto C$ is followed by an observable contingent link $C \Rightarrow O$. The agent becomes immediately aware of O at the conclusion of this process, and so it may at this time deduce enough information to satisfy the constraint on $C \rightarrow A$ (whose interval $[x', x]$ has possibly shrunken from its original length via reduction). We identify two conditions that collectively guarantee this ability:

- O must be sufficiently *punctual* ($z \leq x$): it must always be possible for A to occur during or after O, regardless of when O is scheduled by Nature. Otherwise, the occurrence of O may be too late for A to observe, and thus the agent will be unable to infer any information about C.
- O must be sufficiently *informative* ($z - z' \leq x - x'$): the width of the interval $[z', z]$ on $C \Rightarrow O$ must be no greater than the width of the interval $[x', x]$ on $C \rightarrow A$. Otherwise, the observation of O does not provide an adequate window of the true value of C.

If both of these conditions are met, we say that O makes C *sufficiently observable* to A.

Theorem: *If C is sufficiently observable to A via O, the subnetwork is locally controllable (i.e., we can dynamically determine a value for A following O that satisfies the requirement link $C \rightarrow A$).*

[5] As in (Morris 2006), we present a slight variation of the rules that allows instantaneous agent reaction to observed events.

Must-Precede($A \overset{x}{\leftarrow} C$) // *for STPU*

1. If $(x < 0)$ *return true;*
2. *return false;*

Figure 5: Returns *true* if C must execute after A (*false* otherwise).

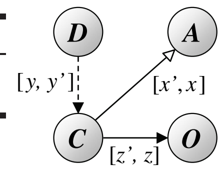

Figure 6

Must-Precede-Observation($A \overset{x}{\leftarrow} C$) // *for POSTPU*

1. If $(x < 0)$ *return true;*
2. If ($C \in X_U$ and $A \notin$ *Ripples*(C))
3. For each observable uncontrollable $O \in$ *Ripples*(C)
4. If $(x < z)$ *next* O;
5. If $(x - x' < z - z')$ *next* O;
6. *return false;*
7. *return true;*
8. *return false;*

Figure 8: Returns *true* if C is not sufficiently observable to A (*false* otherwise).

Proof: Suppose that O is observed at time t. From the bounds on $C \Rightarrow O$, we know that C must have occurred in the range $[t-z, t-z']$. To satisfy the earliest of these times, A can occur no later than $t - z + x$. To satisfy the latest of these times, A must occur no earlier than $t-z'+x'$. So, even having not observed C directly, we can establish a feasible window $[t - z' + x', t - z + x]$ on the execution of A. For this interval to be valid, we need $t - z' + x' \leq t - z + x$, or $z - z' \leq x - x'$ (ensured since sufficiently informative). For this to be executed dynamically (i.e., after O is observed), we need $t \leq t-z+x$, or $z \leq x$ (ensured by sufficient punctuality). Thus, from sufficient observability, a value for A can be determined dynamically from O to satisfy $C \rightarrow A$. □

In the case that multiple observable contingent links follow C, at least one of these must meet the above conditions to ensure sufficient observability of C. In fact, any observable time point that follows a cascade of contingent links from C has the potential to provide evidence; such a cascade can be considered a single unbroken chain of *extended contingency* executed by Nature without interaction on behalf of the agent. We formally refer to the set of all possible observation points of C as *Ripples*(C), analogous to the ripples observed in water after an object is dropped and disappears from view.

Definition: Given an unobservable, uncontrollable event C, we define *Ripples*(C) as the set of uncontrollables that lie at the conclusion of a contingent link that begins with either C or another unobservable uncontrollable in *Ripples*(C). □

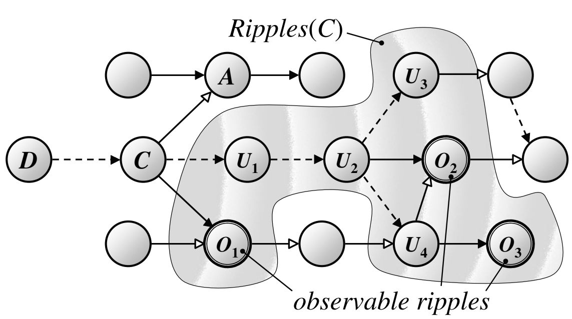

Figure 7: *Ripples*(C) includes those uncontrollable events that fall upon an extended path of contingency from C.

This recursive definition encompasses all events executed by Nature after C that do not require agent interaction; each of the observable uncontrollables in this set can be viewed as a delayed perception of C (see Figure 7 for an illustration).[6]

In Figure 8, we provide pseudocode for the function **Must-Precede-Observation()**, replacing the function **Must-Precede()** for the lower-case and cross-case reduction rules. If C is an unobservable uncontrollable and the event A will not be subsequently scheduled by Nature (line 2), we examine each observation point O in *Ripples*(C) (line 3) to check whether it is sufficiently punctual (line 4) and sufficiently informative (line 5). If both conditions are met, then sufficient observability is ensured, and the function returns *false* (line 6). Otherwise, no single observation point provides sufficient observability, and *true* is returned (line 7).

Incompleteness and an Open Problem

While the inference resulting from these augmented reduction rules is a sound improvement (and polynomial addition) over previous methods, it remains too weak to correctly dismiss global controllability in all cases. For instance, Figure 9 displays a network where (if $X = 3$) the unobservables C and C' are made sufficiently observable to A and A' via O and O', respectively. Repeated application of the reduction rules does not lead to a negative cycle, suggesting that the entire network may be dynamically controllable.

However, if *both* C and C' occur as late as possible, waiting for observables requires A' to execute no earlier than 4 + 2 + 4 + 2 = 12 units after D, violating the upper bound of 11. This inconsistency is not detected by the current algorithm, as the condition of sufficient punctuality is only checked locally, and not fully enforced or propagated. In contrast, an algorithm that blindly enforces precedence constraints between observation points and controllable events is not sound (e.g., if $X = 4$, dynamic controllability is established by ignoring sufficient observability and imposing no precedence of observables before controllables).

Hence, a complete decision procedure for handling partial observability appears to require some branching: the agent may either rely on observable events by enforcing the prece-

[6]If all requirement links are ignored, *Ripples*(C) is effectively the set of successors of C in the resulting Directed Acyclic Graph.

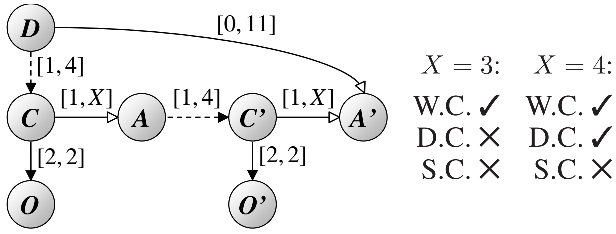

Figure 9: A network where sufficient observability does not necessarily guarantee global dynamic controllability.

dence of observation, or choose to ignore them by propagating an assumption of independence. Although this points to a combinatorial search, we leave open the possibility that stronger inference rules may exist to preserve the tractable complexity of dynamic controllability.

Future Work

Aside from resolving the complexity class of dynamic controllability, we envision other promising avenues for continued progress on the Partially Observable STPU. For instance, while our current algorithmic framework builds on the quadratic cutoff algorithm, there may be potential in extending the recently developed linear cutoff algorithm (Morris 2006). However, the normal form identified in that line of work cannot be applied to our POSTPU without complications; in particular, it necessitates the extraction of additional requirement links from all contingent links whose lower-bounds are non-zero. This imposes a discontinuity that our current formulation of extended contingency does not appear to overcome.

Conclusion

We have explored a means to both model and reason about the important yet largely neglected problem of partial observability within constraint-based temporal reasoning. Our approach has been primarily motivated by two key, interrelated questions: (1) how to adequately capture real-world environments whose visible but uncontrollable processes share a common source of temporal causality, and (2) how to determine the extent to which an agent can safely perform online execution when such underlying causes are entirely hidden from view.

Among our chief contributions are a new formalism (the *Partially Observable STPU*) that extends the STPU to include unobservable events, a re-characterization of its levels of controllability, an exposition of tractable special cases, and a sound extension to the reduction rules for maintaining the labelled distance graph. While the true complexity of dynamic controllability remains unresolved, our analysis of *extended contingency* and *sufficient observability* offers valuable insight into the enhanced inference needed to strengthen the power of graph-based algorithms for controllability.

Acknowledgements

This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. NBCHD030010 and the Air Force Office of Scientific Research under Contract No. FA9550-04-1-0043. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the view of DARPA, the Department of Interior-National Business Center, or the United States Air Force.

References

Bertoli, P.; Cimatti, A.; Roveri, M.; and Traverso, P. 2006. Strong planning under partial observability. *Artificial Intelligence* 170(4):337–384.

Cassandra, A. R.; Kaelbling, L. P.; and Littman, M. L. 1994. Acting optimally in partially observable stochastic domains. In *Proceedings of the 12th National Conference on Artificial Intelligence (AAAI-1994)*, 1023–1028.

Dechter, R.; Meiri, I.; and Pearl, J. 1991. Temporal constraint networks. *Artificial Intelligence* 49(1-3):61–95.

Mausam, and Weld, D. S. 2006. Probabilistic temporal planning with uncertain durations. In *Proceedings of the 21st National Conference on Artificial Intelligence (AAAI-2006)*, 880–887.

Moffitt, M. D., and Pollack, M. E. 2007. Generalizing temporal controllability. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI-2007)*, 1985–1990.

Morris, P. H., and Muscettola, N. 1999. Managing temporal uncertainty through waypoint controllability. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI-1999)*, 1253–1258.

Morris, P. H., and Muscettola, N. 2000. Execution of temporal plans with uncertainty. In *Proceedings of the 17th National Conference on Artificial Intelligence (AAAI-2000)*, 491–496.

Morris, P. H., and Muscettola, N. 2005. Temporal dynamic controllability revisited. In *Proceedings of the 20th National Conference on Artificial Intelligence (AAAI-2005)*, 1193–1198.

Morris, P. H.; Muscettola, N.; and Vidal, T. 2001. Dynamic control of plans with temporal uncertainty. In *Proceedings of the 17th International Joint Conference on Artificial Intelligece (IJCAI-2001)*, 494–502.

Morris, P. 2006. A structural characterization of temporal dynamic controllability. In *Proceedings of the 12th International Conference on Principles and Practice of Constraint Programming (CP-2006)*, 375–389.

Muscettola, N.; Morris, P. H.; Pell, B.; and Smith, B. D. 1998. Issues in temporal reasoning for autonomous control systems. In *Agents*, 362–368.

Nance, M.; Vogel, A.; and Amir, E. 2006. Reasoning about partially observed actions. In *Proceedings of the 21st National Conference on Artificial Intelligence (AAAI-2006)*, 888–893.

Pollack, M. E.; Brown, L.; Colbry, D.; McCarthy, C. E.; Orosz, C.; Peintner, B.; Ramakrishnan, S.; and Tsamardinos, I. 2003. Autominder: An intelligent cognitive orthotic system for people with memory impairment. *Robots and Autonomous Systems* 44(3-4):273–282.

Rintanen, J. 2004. Complexity of planning with partial observability. In *Proceedings of the 14th International Conference on Automated Planning and Scheduling (ICAPS-2004)*, 345–354.

Vidal, T., and Fargier, H. 1999. Handling contingency in temporal constraint networks: From consistency to controllabilities. *Journal of Experimental and Theoretical Artificial Intelligence* 11(1):23–45.