# Localized Temporal Reasoning: A State-Based Approach

**Shieu-Hong Lin** and **Thomas Dean**
Department of Computer Science
Brown University, Providence, RI 02912

## Abstract

We are concerned with temporal reasoning problems where there is uncertainty about the order in which events occur. The task of temporal reasoning is to derive an event sequence consistent with a given set of ordering constraints immediately following which one or more conditions have specified statuses. Previous research shows that the associated decision problems are hard even for very restricted cases. In this paper, we present a framework of localized temporal reasoning which use subgoals and abstraction to exploit structure in temporal ordering and causal interaction. We investigate (1) locality regarding ordering constraints that group events hierarchically into sets called *regions*, and (2) locality regarding causal interactions among regions, which is characterized by subsets of the set of conditions. The complexity for an instance of temporal reasoning is quantified by the sizes of the characteristic subsets of conditions and the numbers of child regions of individual regions in the region hierarchy. This demonstrates the potential for substantial improvements in performance.

## 1 Introduction

We are interested in reasoning about a dynamical system whose state is modeled by a set of conditions $\mathcal{P}$. Each condition has a status, true or false, at a given point in time and the *evolution of the system* corresponds to a sequence of state changes. The evolution of the system depends on the interactions among a set of events $\mathcal{E}$. An event $e$ in $\mathcal{E}$ changes the system state according to a set of *causal rules* associated with $e$. A rule associated with an event $e$ is a STRIPS-like operator describing the causal effects of $e$ when the preconditions of the rule are satisfied. In this way, each event specifies a state-transition function on the state space determined by $\mathcal{P}$ [10]. The ordering of the events is uncertain. The possible event sequences are determined by a given set of ordering constraints on the events. Generally, many event orderings are possible, each of which may result in a different evolution of the system.

Given the initial state and a set of goal conditions $\mathcal{G}$, the task of *temporal reachability* [10] is to (1) detect the existence of a possible event sequence immediately following which the conditions in $\mathcal{G}$ have specified statuses, and (2) generate one such event sequence if one exists. In other words, the task of temporal reachability is to predict the possibility of reaching the goal states where the conditions in $\mathcal{G}$ have the specified statuses, and report one possible event sequence which ends in the goal states. For example, consider a partial plan where the ordering of the components is constrained by some partial order. Using temporal reachability, we can coordinate the component events to reach the goal states in a way consistent with the inherent ordering constraints. On the other hand, we can also apply temporal reachability to validate the partial plan by (1) specifying the undesirable states as goal states where the set of goal conditions have the specified statuses, (2) detecting the possibility of reaching undesirable states, and (3) reporting such a possible event sequence as evidence. This research extends the earlier work of Dean and Boddy [5] on reasoning about partially ordered events.

In special cases, temporal reasoning with uncertainty about the ordering of events is harder than the corresponding planning problem [11]. This happens because in the corresponding planning problem, the planner is free to select an arbitrary set of event instances from a fixed set of event types with no restrictions on the ordering of event instances. In most cases, if the events are not under the planner's control, however, the problems are computationally equivalent. Previous research shows that the associated decision problems for temporal reachability [10] and planning are hard even for very restricted cases [1] [2] [3] [5] [6].

We have been trying to understand why temporal reachability and planning are so difficult and what, if any, structure might be extracted from such problems to expedite decision making and inference [10]. In this paper, we present a framework of localized temporal reasoning. Using this framework, temporal reachability is viewed as search in the global search spaces associated with individual problem instances. The sizes of the global search spaces are determined by (1) the total number of the conditions in $\mathcal{P}$, and (2) the total number of the events in $\mathcal{E}$ . Our investigations have focused on the notions of locality in time and the structure of the search spaces. Locality in time in a particular problem instance is modeled by a hierarchy of *regions*. Each region is composed of an encapsulated event subset. The events outside a region $R$ must occur either before or after the events in $R$. We allows the child regions of a region to be partially ordered. Given a hierarchy of regions, we can induce a set of *coupling conditions* and a set of *abstract conditions* for each individual region, which reflect locality in the associated search space, and enable us to construct local search spaces. The coupling conditions of a region $R$ are the media for inter-regional interactions between the events in different child regions of $R$. The abstract conditions of a region $R$ are the media for the interactions between the events in region $R$ and the events not in $R$. Instead of search in the global search space, we can search the local search spaces and propagate the effects of local search. The sizes of the sets of coupling conditions and the numbers of child regions of individual regions determine the complexity for an instance of temporal reachability.

This research utilizes the notions of localized planning [9], subgoals [8], and abstraction [4] [7] [8] [12] [13] to expedite temporal reasoning. The use of event-based localized reasoning in planning has been advocated by Lansky in GEM [9]. In GEM, locality regarding domain behavior is also modeled by "regions". "Regions" in GEM are composed of sets of interrelated events whose behaviors are delimited by event-based temporal logic constraints. The GEMPLAN planner utilizes localized planning to exploit locality by subdividing the global search space into regional search spaces. In this paper, we identify the corresponding notion of regions in the temporal reachability problem, which models locality in time. In addition, we study the dependencies between conditions and regions, and identify the coupling conditions and the abstract conditions associated with individual regions. These structures model locality in search spaces. The state-based approach allows us to use knowledge regarding subgoals, abstract events, and local search spaces to conduct and analyze localized reasoning in a clear and elegant way. The usefulness of subgoals, macro-operators, and abstraction in reducing the search effort in planning has been investigated by Korf [8]. We identify subgoals and abstract events as useful knowledge for temporal reachability. We develop techniques to derive this critical knowledge from individual temporal reachability instances. Our techniques are different from previous work on abstraction [4] [7] [13] in that we transform encapsulated event subsets into abstract events, instead of transforming individual operators into abstract operators.

## 2 The Temporal Reachability Problem

An instance of the temporal reachability problem is defined by $\langle \mathcal{P}, \mathcal{I}, \mathcal{G}, \mathcal{E}, \mathcal{O} \rangle$.

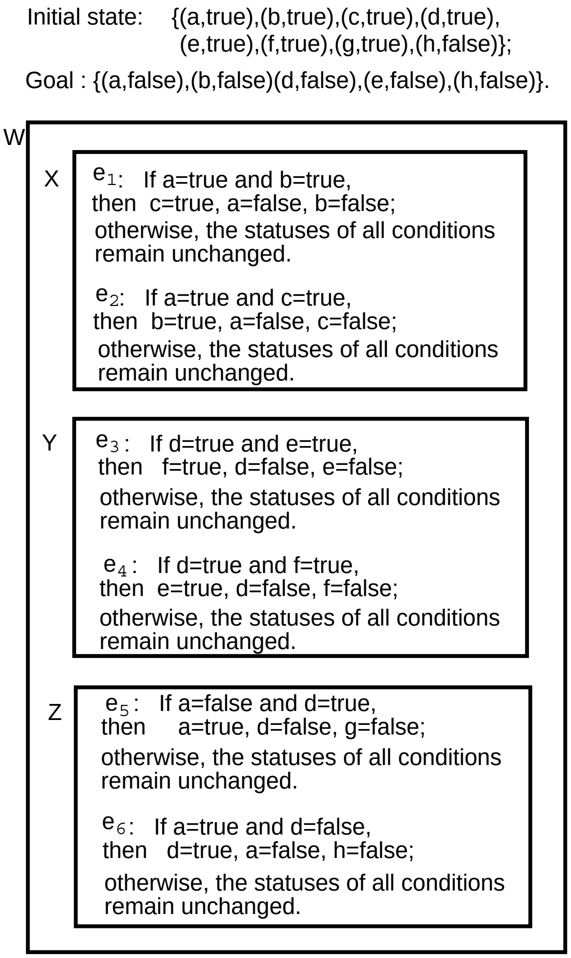
- $\mathcal{P}$ is a set of conditions modeling the world state. Each condition in $\mathcal{P}$ has a status, true or false, at a given point in time.
- $\mathcal{I}$ is a set of condition/status pairs, that specifies the initial statuses of each condition in $\mathcal{P}$.
- $\mathcal{G}$ is a set of condition/status pairs, that specifies the goal statuses of all or a subset of the conditions in $\mathcal{P}$.
- $\mathcal{E}$ is a set of events, each of whose causal effects are represented by a set of causal rules. A causal rule $r$ is a STRIPS-like operator, which describes the status changes of a subset of the conditions in $\mathcal{P}$ (i.e. the consequent effects) when the conditions in another subset of $\mathcal{P}$ have the statues specified by $r$ (i.e. the antecedent requirements). In this way, each event determines a state-transition function on the state space modeled by $\mathcal{P}$.
- $\mathcal{O}$ is a set of arbitrary constraints on $\mathcal{E}$.

The task of temporal reachability is to (1) detect the existence of event sequences of size $|\mathcal{E}|$ consistent with $\mathcal{O}$ such that the goal specified by $\mathcal{G}$ is achieved immediately following the event sequences, and (2) generate one such event sequence if one exists.

Figure 1 describes an example problem instance.

- The world state is modeled by a set of conditions $\mathcal{P} = \{a, b, c, d, e, f, g, h\}$.
- The initial state is described by $\mathcal{I}$. All the conditions are initially true except that $h$ is false.
- $\mathcal{E} = \{e_1, e_2, e_3, e_4, e_5, e_6\}$. Each event in $\mathcal{E}$ is associated with a causal rule while in general an event can be associated with multiple causal rules.
- Figure 2 depicts the ordering constraints on $\mathcal{E}$. (For notational convenience, we use $W$ as an alias name for $\mathcal{E}$.) The events in the three event subsets $X$, $Y$, and $Z$ must occur as three atomic groups, where $X = \{e_1, e_2\}$, $Y = \{e_3, e_4\}$,$Z = \{e_5, e_6\}$. The events in $Y$ must occur before the events in $X$. Event $e_1$ must occur before event $e_2$.

Our task is to (1) determine the existence of event sequences consistent with $\mathcal{O}$ such that the goal $\mathcal{G} = \{(a, false), (b, false), (d, false), (e, false), (h, false)\}$ is achieved immediately following the event sequences, and (2) generate one such event sequence if one exists.



Initial state: {(a,true),(b,true),(c,true),(d,true), (e,true),(f,true),(g,true),(h,false)};

Goal : {(a,false),(b,false)(d,false),(e,false),(h,false)}.

W

X

$e_1$: If a=true and b=true, then c=true, a=false, b=false; otherwise, the statuses of all conditions remain unchanged.

$e_2$: If a=true and c=true, then b=true, a=false, c=false; otherwise, the statuses of all conditions remain unchanged.

Y

$e_3$: If d=true and e=true, then f=true, d=false, e=false; otherwise, the statuses of all conditions remain unchanged.

$e_4$: If d=true and f=true, then e=true, d=false, f=false; otherwise, the statuses of all conditions remain unchanged.

Z

$e_5$: If a=false and d=true, then a=true, d=false, g=false; otherwise, the statuses of all conditions remain unchanged.

$e_6$: If a=true and d=false, then d=true, a=false, h=false; otherwise, the statuses of all conditions remain unchanged.

Figure 1: An instance of the temporal reachability problem

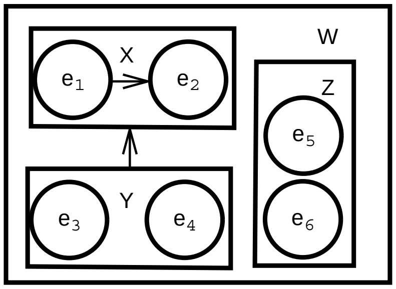


Figure 2: Ordering constraints on events

## 3 Locality in Temporal Ordering

Temporal reachability turns out to be NP-Complete even if events are totally unordered (i.e. $\mathcal{O} = \emptyset$) and the associated state space is polynomial in the size of the event set (i.e. $|\mathcal{P}| = O(\log |\mathcal{E}|)$) [10]. This complexity result motivates our effort to exploit inherent locality regarding event ordering and the dependencies among conditions and events.

When totally unordered, events can occur in arbitrary order. However, there is locality in event ordering if the occurrences of events or subsets of events closely relate to one another. In this paper, we consider locality in event ordering regarding *hierarchical task networks*. We define a hierarchical task network as a partial plan such that

- the plan is organized as a hierarchy of tasks, where an individual task may be composed of subtasks that are also tasks in the hierarchy,
- the subtasks of an individual task may be constrained by some partial order, and
- the leaf tasks of the task hierarchy are events, each of which changes the world states according to a set of associated causal rules. Each causal rule is a STRIPS-like operator.

For example, we may be working on several programming tasks. Each task is composed of a set of subtasks, including prototype design, prototype testing, problem reformulation, etc., such that each subtask involves a set of events that we wish to occur as a group. We allow ourselves to switch between programming tasks, but, once we begin a subtask in a given programming task, we commit to completing all of the events associated with that subtask before switching to any other subtasks.

In the following, we define the concepts of *regions*, *child regions*, *region hierarchy*, and *hierarchical ordering constraints*. Given a hierarchical task network, locality in event ordering can be abstractly modeled by a hierarchy of regions, and a set of hierarchical ordering constraints on the regions.

**Definition 1.** *Given a set of events $\mathcal{E}$, an event subset $X$ of $\mathcal{E}$ is a region in $\mathcal{E}$ if the events in $X$ occur as an atomic group with the events outside the region occurring either before or after the events inside the region. $\mathcal{E}$ and each individual event in $\mathcal{E}$ are regions by themselves.*

In a hierarchical task network, a region models the set of descendant events (i.e. leaf tasks) of a task, which always occur as an atomic group. In other words, each task is associated with a region.

**Definition 2.** *Given a set of events $\mathcal{E}$, the regions in $\mathcal{E}$ form a region hierarchy if, for any two regions in $\mathcal{E}$, either they are disjoint or one is a proper subset of the other.*

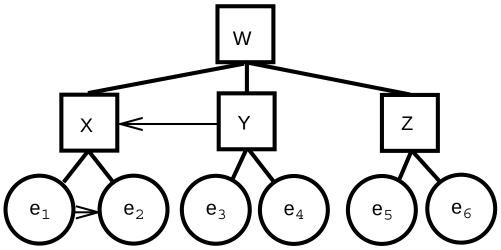


Figure 3: A region hierarchy

The events in a hierarchical task network form a region hierarchy, since they have the property described in Definition 2.

**Definition 3.** *In a region hierarchy, region $X$ is a child region of another region $Y$ if $Y$ is the smallest region containing $X$.*
For a task $k$, the regions associated with $k$'s subtasks are the child regions of the region associated with $k$.

**Definition 4.** *A hierarchical ordering constraint is a 3-tuple $\langle R, \sigma, \prec \rangle$, where $R$ is a region, $\sigma = \{R_1, R_2, \ldots\}$ is the set of child regions of $R$, and $\prec$ is a partial order on $\sigma$. The events in $R_i$ must occur before the events in $R_j$ if $R_i \prec R_j$.*
A hierarchical ordering constraint models a partial order on the subtasks of a task.

In our example problem instance, $X = \{e_1, e_2\}$, $Y = \{e_3, e_4\}$, $Z = \{e_5, e_6\}$, $W = X \bigcup Y \bigcup Z$ and every individual event are regions. Figure 3 depicts the corresponding region hierarchy. The possible event sequences regarding our example problem instance are determined by the following hierarchical ordering constraints: $\langle X, \{e_1, e_2\}, \{e_1 \prec e_2\} \rangle$, $\langle Y, \{e_3, e_4\}, \emptyset \rangle$, $\langle Z, \{e_5, e_6\}, \emptyset \rangle$, and $\langle W, \{X, Y, Z\}, \{Y \prec X\} \rangle$.

## 4 Locality in Causal Interactions

In this section, we derive useful knowledge regarding the causal interactions among regions. This knowledge allows us to describe *subgoals* and *abstractions* for individual regions, which yields a localized reasoning algorithm in next section. Our investigation focuses on the causal dependencies among regions and subsets of conditions. Events that affect all or even most conditions are rare. An event tends to affect or be affected by a small subset of the set of conditions $\mathcal{P}$. We say that a condition $p$ is *dependent* on an event $e$ if (1) event $e$ may change the truth value of condition $p$ immediately following $e$ or (2) the truth value of condition $p$ prior to event $e$ may affect the truth values of some other conditions immediately following $e$; otherwise, we say that condition $p$ is *independent* of event $e$. Similarly, a condition $p$ is dependent on region $R$ if $p$ is dependent on at least one of the events encapsulated in $R$; otherwise $p$ is independent of $R$. If $p$ is independent of $R$, we do not need to concern ourselves with $p$ in reasoning about the changes caused by $R$. In Figure 1, the conditions $a$, $b$, $c$ are dependent on events $e_1$, $e_2$ and region $X$ while the conditions $d$, $e$, $f$, $g$, $h$ are independent of events $e_1$, $e_2$ and region $X$.

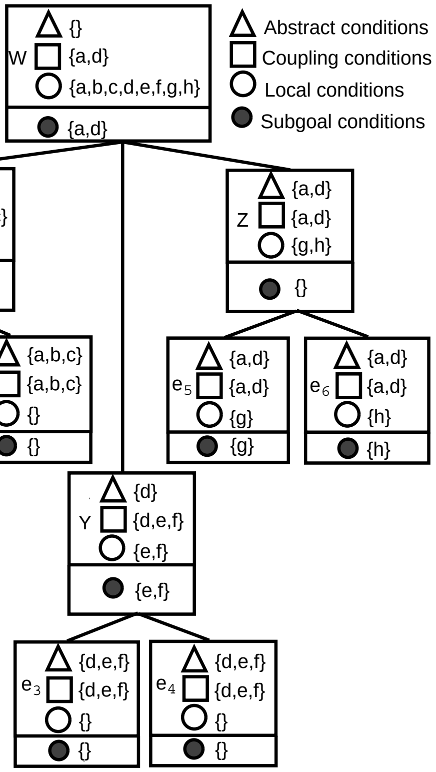


Figure 4: Characteristic condition subsets

### 4.1 Characteristic Condition Subsets

In the following, we generalize this notion of dependency to characterize subsets of conditions for each individual region $R$ in a given problem instance, namely (1) the set of *local conditions* of $R$, (2) the set of *subgoal conditions* of $R$, (3) the set of *abstract conditions* of $R$, and (4) the set of *coupling conditions* of $R$. These subsets of conditions characterize the causal interactions among regions. Figure 4 shows such knowledge derived from the example problem instance in Figure 1.

**Local conditions:** A condition $p$ is a local condition of region $R$ if $p$ is dependent on $R$ but not dependent on any events outside $R$. The statuses of the local conditions of $R$ can only affect and be affected by the events in $R$. The initial statuses of the local conditions of $R$ will not be changed until the events in $R$ occur and change their statuses.

**Subgoal conditions:** *The subgoal conditions of $R$ describe the regional subgoal in $R$.* A condition $p$ is a subgoal condition of region $R$ if (1) $p$ is a local condition of $R$, *and* (2) $p$ is dependent on more than one child region of $R$ if $R$ has more than one child region. As soon as all the events in $R$ occur, the statuses of the subgoal conditions of $R$ will not be changed any more. At that time, the statuses of the subgoal conditions must have the specified conditions given by the goal of the problem instance. This is the subgoal to be achieved by the events in $R$. For example, in Figure 1 the status of the subgoal condition $d$ in region $Y$ must be false immediately after region $Y$ is finished.

**Abstract conditions:** *We use the knowledge of the abstract conditions of a region $R$ in deriving the abstraction of $R$.* A condition $p$ is an abstract condition of region $R$ if $p$ is (1) dependent on at least one event in $R$, *and* also (2) dependent on at least one event not in $R$. Both the events in $R$ and the events outside $R$ can affect and be affected by the statuses of the abstract conditions of $R$. Therefore the set of abstract conditions of $R$ is the media for inter-regional interactions between the events in $R$ and the events outside $R$. For example, in Figure 1 the events $e_1$, $e_2$ in region $X$ interact with the other events through the abstract condition $a$ of $X$.

**Coupling conditions:** *In next section, we use the knowledge of the coupling conditions of a region $R$ in constructing the local search space for $R$.*

A condition $p$ is a coupling condition of region $R$ if (1) $p$ is an abstract condition of region $R$, *or* (2) $p$ is an abstract condition of one or more child regions of $R$. The coupling conditions of $R$ are the media for (1) the inter-regional interactions between the events in $R$ and the events outside $R$, and (2) intra-regional interactions among different child regions of $R$. For example, in Figure 1 the three child regions $X$, $Y$, and $Z$ of region $W$ interact with one another through the coupling conditions $a$, $d$ of $W$.

## 4.2 Subgoals and Abstractions

Based on the knowledge of the sets of subgoal conditions and abstract conditions of individual regions, we define the concepts of regional subgoals and the abstract events of individual regions, which yields the localized reasoning algorithm in Section 5.

### 4.2.1 Regional subgoals

The statuses of the subgoal conditions of $R$ must have the specified statuses described by the goal as soon as the events in $R$ have all occurred, and the statuses of the subgoal conditions of $R$ will not be changed after that time.

**Definition 5.** *The regional subgoal $\mathcal{G}_R$ is a subset of the goal $\mathcal{G}$. $\mathcal{G}_R$ is composed of the condition/status pairs in $\mathcal{G}$ whose condition component is a subgoal condition of region $R$.*

Figure 5 depicts the regional subgoals regarding the example problem instance in Figure 1. By achieving the subgoal in every individual region, the goal is achieved. This describes a problem decomposition for temporal reachability.

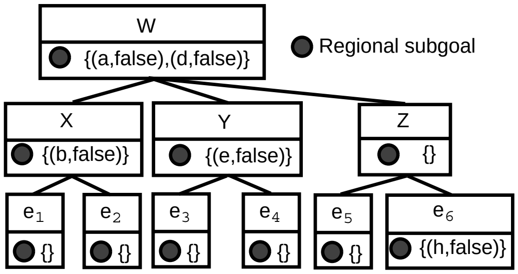


Figure 5: Subgoals in individual regions

Our task in an individual region $R$ is to (1) achieve the subgoals in the children regions of $R$ recursively, (2) determine the ordering of the child regions of $R$ to achieve the regional subgoal of $R$.

For example, the subgoal for $Y$ is $\{(e, false)\}$, which indicates that the status of condition $e$ must be false immediately after the events in $Y$ have all occurred. $\langle e_3, e_4\rangle$ is the only ordering of $e_3$, $e_4$ to achieve this subgoal.

### 4.2.2 The abstractions of individual regions

The effects caused by the events in a region $R$ are determined by

(1) the initial statuses of the local conditions of $R$,

(2) the statuses of the abstract conditions of $R$ immediately before the events in $R$ occur, and

(3) the ordering of the events in $R$, which can be recursively described by the orderings on the child regions of $R$ and $R$'s descendant regions.

In general, there are many possible orderings on the child regions of $R$ to achieve the subgoal in an individual region $R$. In order to achieve the subgoals in region $R$,

(1) the set of abstract conditions of $R$ must have appropriate statuses immediately before any events in $R$ occur, and

(2) the child regions in $R$ must be properly ordered.

In other words, (1) whether or not an ordering on $R$'s child regions may achieve the subgoal in $R$ depends on the statuses of the abstract conditions of $R$ immediately before the events in $R$ occur, and (2) in turn, that ordering also affects the statuses of the abstract conditions of $R$ after achieving the subgoal in $R$. An *abstract event $e_R$* of a region $R$ provides a causal abstraction of $R$ regarding how to achieve the subgoal in $R$ using the events in $R$.

**Definition 6.** *The abstract event $e_R$ of a region $R$ is represented by a set of causal rules. Each causal rule of $e_R$ encodes one possibility regarding (1) the statuses of the abstract conditions of $R$ immediately before any events in $R$ occur, which represent the antecedent requirements to achieve the subgoal, and (2)*

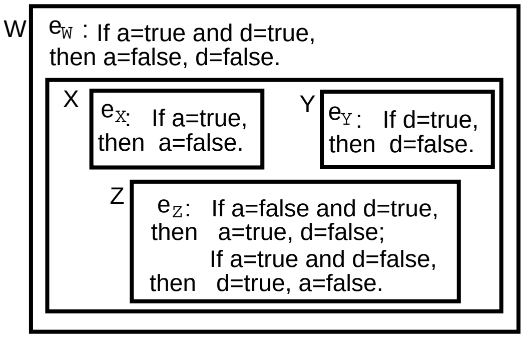


Figure 6: Abstract events for individual regions

*the statuses of the abstract conditions immediately after the regional subgoal is achieved by the events in $R$, which represent the consequent effects.*

The abstract event $e_R$ is what we need to reason about the inter-regional interactions between the events in $R$ and the events outside $R$. The derivation of abstract events for individual regions is investigated in Section 5. Figure 6 depicts the abstract events of the individual regions in our example problem instance. For example, the abstract event $e_Y$ of region $Y$ is associated with a single rule. This is because (1) the subgoal for $Y$ is $\{(e, false)\}$, (2) when the condition $d$ is true immediately before $e_3$ and $e_4$ occur, the event ordering $\langle e_3, e_4\rangle$ can achieve the regional subgoal, and (3) when the condition $d$ is false immediately before $e_3$ and $e_4$ occur, the subgoal cannot be achieved.

# 5 Localized Temporal Reasoning Using Subgoals and Abstractions

In this section, we first present temporal reachability as search in global search spaces. We then investigate the use of local reasoning to expedite temporal reasoning by exploiting the structure of a region hierarchy. Finally, we demonstrate the potential of substantial improvements in performance, which are quantified by the numbers of child regions and the sizes of the sets of coupling conditions of individual regions.

## 5.1 Temporal Reachability as Search

Given an instance of the temporal reachability problem $\langle \mathcal{P}, \mathcal{I}, \mathcal{G}, \mathcal{E}, \mathcal{O}\rangle$, we describe the global search space as a directed graph $G = (V, A)$ where
A node in $V$ indicates (1) the state of the conditions in $\mathcal{P}$, i.e., the status of $p$ for each $p$ in $\mathcal{P}$, and and (2) for each event $e$ in $\mathcal{E}$, whether $e$ has occurred.
An edge $(u, v)$ in $A$ models the occurrence of an event $e$ where
(1) $e$ can transform the statuses of the conditions in $\mathcal{P}$ from the state indicated by $u$ to the state indicated by $v$,
(2) node $u$ indicates that $e$ has not yet occurred while node $v$ indicates that $e$ has occurred, and
(3) $e$ can occur immediately after those events that are marked as occurred at node $u$ without violating the ordering constraints in $\mathcal{O}$.

The task of temporal reachability can be viewed as search for a path from a root node $u_0$ to any goal node $t$ where
**(1)** $u_0$ is the node in which all events are marked as not yet occurred, and the conditions in $\mathcal{P}$ have the specified initial statuses given by $\mathcal{I}$, and
**(2)** a node $t$ is a goal node if all events are indicated as occurred and the conditions have the statuses specified by the goal $\mathcal{G}$.
Since an edge in $G = (V, A)$ models the occurrence of an event, such a path corresponds to a possible event sequence immediately following which the conditions have the statuses specified in $\mathcal{G}$.

## 5.2 Local Reasoning in Local Search Spaces

In the following, we consider the kind of problem instances ($\langle \mathcal{P}, \mathcal{I}, \mathcal{G}, \mathcal{E}, \mathcal{O}\rangle$) where the events in $\mathcal{E}$ form a region hierarchy according to a set of hierarchical ordering constraint $\mathcal{O}$. Instead of reasoning about (1) the whole set of events and (2) the whole set of conditions to achieve the goal, we can conduct local reasoning in each individual region about (1) the abstract events of their child regions and (2) the set of coupling conditions to achieve the regional subgoal. The goal is attained by incrementally achieving the regional subgoals. This yields the following localized temporal reasoning algorithms.

### 5.2.1 Constructing Local Search Spaces

For a region $R$, the local search space of $R$ embeds the information regarding (1) how the child regions of $R$ interact with one another through $R$'s coupling conditions, and (2) the ordering constraints on the child regions of $R$.

Given (1) the abstract events of region $R$'s child regions and (2) the set of coupling conditions of $R$, we can construct $R$'s local search space as a directed graph $G_R = (V_R, A_R)$ in the following way.

- Construct a set of nodes $V_R$ such that each node in $V_R$ encodes one possibility regarding (1) the statuses of the set of coupling conditions, (2) for each child region $R'$ in $R$, whether the events in $R'$ as a whole have occurred.
- Construct an edge $(u, v)$ in $A_R$ if there exists a child region $R'$ of $R$ such that
(1) $e_{R'}$, the abstract event of $R'$, can transform the statuses of the coupling conditions indicated by node $u$ to the statuses indicated by node $v$ while achieving the subgoal in $R'$,
(2) node $u$ indicates that $R'$ has not yet occurred while node $v$ indicates that $R'$ has occurred, and

(3) $R'$ can occur immediately after those child regions that are marked as occurred at node $u$ without violating the ordering constraints on the child regions of $R$.
(*Note that edge $(u, v)$ models the occurrence of the events in the child region $R'$ as a whole.*)

### 5.2.2 Deriving Abstract Events

Given the local search space $G_R = (V_R, A_R)$ of a region $R$, we define the following two types of nodes in $G_R$.
**Type-I nodes:** A node $v$ in $V_R$ is a Type-I node if all child regions of $R$ are marked as not yet occurred at $v$.
**Type-I nodes:** A node $v$ in $V_R$ is a Type-II node if (1) at $v$, all child regions of $R$ are marked as occurred, and (2) at $v$, the subgoal conditions of $R$ have the statuses specified by the regional subgoal of $R$.
Type-I nodes represent the possible statuses of the coupling conditions of $R$ immediately before any child region of $R$ occurs.
Type-II nodes represent the possible statuses of the coupling conditions of $R$ immediately after all child regions of $R$ occur and the regional subgoal of $R$ is attained.

**Property 1.** *The local search space $G_R$ of a region $R$ is a directed acyclic graph. A path in $G_R$ from a Type-I node to a Type-II node represents an ordering of the child regions of $R$ to achieve the regional subgoal in $R$.*

**Procedure Region-Abstraction**
**Input:** (1) the abstract events of the child regions of $R$, (2) the regional subgoal of $R$.
**Output:** if the regional subgoal of $R$ can be achieved, report the abstract event $e_R$; otherwise, stop and report failure in achieving the subgoal.
**1.** Construct the local search space $G_R = (V_R, A_R)$. Derive the reachability information between Type-I nodes and Type-II nodes by searching $G_R$.
**2.** If no Type-I nodes can reach Type-II nodes, stop and report failure.
**3.** If a Type-I node $u$ can reach a Type-II node $v$, we encode a rule $r$ associated with the abstract event $e_R$ such that
(i) the antecedent requirement of $r$ is that the abstract conditions of region $R$ must have the statuses indicated at node $u$, and
(ii) the consequent effect of rule $R$ is that the abstract conditions of region $R$ must have the statuses indicated at node $v$.

### 5.2.3 Achieving Regional Subgoals

For a region $R$, the following procedure generates a sequence of the events in $R$ to achieve all the subgoals of $R$ and $R$'s descendant regions.

**Procedure Generate-Sequence**
**Input:** (1) the statuses of $R$'s coupling conditions before the events in $R$ occur, (2) the abstract events and the subgoals of $R$ and $R$'s descendant regions.
**Output:** a sequence of the events in $R$ to achieve the subgoals of $R$ and $R$'s descendant regions.
**1.** Search the local search space $G_R$ for a path from $u$ to $v$ where (i) $u$ is a Type-I node in which $R$'s coupling conditions have the specified initial statuses and, (ii) $v$ is a Type-II node. **2.** According to the derived path (by Property 1), derive (i) an ordering of the child regions of $R$, (ii) for each child region $R'$ of $R$, the statuses of the coupling conditions of $R'$ immediately before the events in $R'$ occur.
**3.** For each child region $R'$ of $R$, recursively calls procedure **Generate-Sequence** to generate a sequence of the events in $R'$ to achieves the regional subgoals of $R'$ and the child regions of $R'$. According to the derived ordering of $R$'s child regions, concatenate these sequences to generate a sequence of the events in $R$.

### 5.2.4 A Localized Temporal Reasoning Algorithm

In the following, we present a localized temporal reasoning algorithm for temporal reachability.
**Procedure Localized-Reasoning**
**input:** a problem instance $\langle \mathcal{P}, \mathcal{I}, \mathcal{G}, \mathcal{E}, \mathcal{O} \rangle$ where the events in $\mathcal{E}$ form a region hierarchy.
**output:** if there exist possible event sequences to achieve the goal, report one such sequence; otherwise, report failure in finding such a sequence.
**1.** Derive the knowledge regarding the sets of abstract conditions, coupling conditions, and subgoals of individual regions.
**2.** Starting from the bottom level of the region hierarchy, we conduct local reasoning described in Step 3 for the regions at the same level respectively, and then proceed in the same way level by level until we finish the local reasoning in the root region of the region hierarchy.
**3.** For each individual region $R$, call procedure **Region-Abstraction** to derive the abstract event $e_R$ to achieve the regional subgoal of $R$. If the regional subgoal can not be achieved, stop and report failure; otherwise, propagate the knowledge of $e_R$ to $R$'s parent region.
**4.** In the root region, according to the initial statuses of the coupling conditions call procedure **Generate-Sequence** to generate a solution event sequence.

### 5.2.5 Quantifying the Computation Efficiency

**Theorem 1.** *The time complexity of the localized temporal reasoning algorithm is $O(\sum_R 2^{2(B_R+C_R)})$ where $B_R$ and $C_R$ are the number of the child regions and the number of the coupling conditions of an individual region $R$.*
**Proof (Sketch).** The local search space $G_R =$

$(V_R, A_R)$ is a directed acyclic graph of size $O(2^{B_R+C_R})$. To search the graph, record and process the reachability information, it takes $O(2^{2(B_R+C_R)})$ time and space. □

The localized reasoning algorithm is a polynomial-time algorithm if $B_R$ and $C_R$ are of $O(\log|\mathcal{E}|)$ size for each individual region $R$. Since temporal reachability is NP-Complete [10], this demonstrates the potential for dramatic improvements in performance by exploiting inherent locality, which happens when the numbers of child regions ($B_R$'s) and the sizes of the sets of coupling conditions ($C_R$'s) in individual regions are small with respect to the total number of events $|\mathcal{E}|$ and the total number of conditions $|\mathcal{P}|$.

The performance degrades to be exponential in $|\mathcal{E}|$ when some $B_R$ or $C_R$ is of $O(|\mathcal{E}|)$ size. This is under our expectation, since (1) a set of totally unordered events corresponds to a single region containing the individual events as child regions, and (2) temporal reachability regarding totally unordered events is NP-complete [10].

### 5.2.6 An example.

We illustrate the use of the localized reasoning algorithm in solving our example problem instance in Figure 1.

First, we derive causal knowledge and the subgoals for individual regions depicted in Figure 4 and Figure 5 respectively. At the bottom level, we conduct local reasoning regarding the six events respectively, which are regions by themselves. The subgoal in $e_6$ is that $h$ must be false after $e_6$ occurs, while we have null subgoals for the other five events. The subgoal in $e_6$ is always achieved, since $e_6$ can only make $h$ false and $h$ is false initially.

At the second level of the region hierarchy, we conduct local reasoning regarding the regions $X$, $Y$ and $Z$ respectively. The subgoal in region $X$ ($Y$) is that $b$ ($e$) must be true (false) after all the events in region $X$ ($Y$) occur, while we have null subgoal in region $Z$. We call procedure **Region-Abstraction** to determine the abstract event $e_X$ ($e_Y$ and $e_Z$ respectively).

At the top level, the subgoal in *regionW* is to make the conditions $a$ and $d$ true after all events occur. Here the child regions $X$, $Y$ and $Z$ are treated as the abstract events $e_X$, $e_Y$, $e_Z$ respectively. We call procedure **Region-Abstraction** to determine $e_W$.

Finally, we call procedure **Generate-Sequence** to derive a solution sequence, given that the coupling conditions $a$ and $d$ of region $W$ are initial true. Procedure **Generate-Sequence** recursively generates the ordering $\langle e_Z, e_Y, e_X\rangle$, $\langle e_5, e_6\rangle$, $\langle e_1, e_2\rangle$, and $\langle e_3, e_4\rangle$ for the regions $W$, $Z$, $Y$, $X$ respectively. We replace $e_Z$,$e_Y$ and $e_X$ in $\langle e_Z, e_Y, e_X\rangle$ with $\langle e_5, e_6\rangle$, $\langle e_1, e_2\rangle$, $\langle e_3, e_4\rangle$ respectively, and derive $\langle e_5, e_6, e_1, e_2, e_3, e_4\rangle$ as a sequence to achieve the goal.

## 6 Conclusion

We investigate the locality in (1) the ordering constraints that group events hierarchically into sets called *regions*, and (2) the dependencies among conditions and regions. This enables us to describe subgoals and abstractions for individual regions. We develop a localized temporal reasoning algorithm to exploit locality and demonstrate the potential for dramatic improvements in performance.

## References

[1] Bäckström, C. and Klein, I., Parallel Non-Binary Planning in Polynomial Time, *Proceedings IJCAI 12, Sydney, Australia*, IJCAII, 1991, 268–273.

[2] Bylander, Tom, Complexity Results for Planning, *Proceedings IJCAI 12, Sydney, Australia*, IJCAII, 1991, 274–279.

[3] Chapman, David, Planning for Conjunctive Goals, *Artificial Intelligence*, **32** (1987) 333–377.

[4] Christensen, J., A Hierarchical Planner that Generates its own Abstraction Hierarchies, *Proceedings AAAI-90*, AAAI, 1990, 1004–1009.

[5] Dean, Thomas and Boddy, Mark, Reasoning About Partially Ordered Events, *Artificial Intelligence*, **36**(3) (1988) 375–399.

[6] Gupta, Naresh and Nau, Dana S., Complexity Results for Blocks-World Planning, *Proceedings AAAI-91, Anaheim, California*, AAAI, 1991, 629–633.

[7] Knoblock, Craig A., Search Reduction in Hierarchical Problem Solving, *Proceedings AAAI-91, Anaheim, California*, AAAI, 1991, 686–691.

[8] Korf, Richard, Planning as Search: A Quantitative Approach, *Artificial Intelligence*, **33**(1) (1987) 65–88.

[9] Lansky, Amy L., Localized Event-Based Reasoning for Multiagent Domains, *Computational Intelligence*, **4**(4) (1988).

[10] Lin, Shieu-Hong and Dean, Thomas, Exploiting Locality in Temporal Reasoning, *Proceedings of the Second European Workshop on Planning, Vadstena, Sweden*, 1993.

[11] Nebel, Bernhard and Bäckström, Christer, On the Computational Complexity of Temporal Projection, Planning, and Plan Validation, *Artificial Intelligence*, (1993), To appear.

[12] Sacerdoti, Earl, Planning in a Hierarchy of Abstraction Spaces, *Artificial Intelligence*, **7** (1974) 231–272.

[13] Yang, Qiang and Tenenberg, Josh D., Abtweak: Abstracting a Nonlinear, Least Commitment Planner, *Proceedings AAAI-90*, AAAI, 1990, 204–209.