# Goal Specification, Non-determinism and Quantifying over Policies

**Chitta Baral** and **Jicheng Zhao**
Department of Computer Science and Engineering
Arizona State University
Tempe, Arizona 85281
{*chitta, jicheng*}*@asu.edu*

**Abstract**

One important aspect in directing cognitive robots or agents is to formally specify what is expected of them. This is often referred to as goal specification. Temporal logics such as LTL, and CTL$^*$ have been used to specify goals of cognitive robots and agents when their actions have deterministic consequences. It has been suggested that in domains where actions have non-deterministic effects, temporal logics may not be able to express many intuitive and useful goals. In this paper we first show that this is indeed true with respect to existing temporal logics such as LTL, CTL$^*$, and $\pi$-CTL$^*$. We then propose the language, P-CTL$^*$, which includes the quantifiers, exist a policy and for all policies. We show that this language allows for the specification of richer goals, including many intuitive and useful goals mentioned in the literature which cannot be expressed in existing temporal languages. We generalize our approach of showing the limitations of $\pi$-CTL$^*$ to develop a framework to compare expressiveness of goal languages.

## Introduction and motivation

To specify goals of an autonomous agent, a cognitive robot or a planner, one often needs to go beyond just stating conditions that a final state should satisfy. The desired goal may be such that there is no final state (such as in many maintenance goals), and even if there is a final state, the desired goal may also include restrictions on how a final state is reached. To express such goals some of the existing temporal logics such as LTL, and CTL$^*$ (Emerson & Clarke 1982) have been used (Bacchus & Kabanza 1998; Niyogi & Sarkar 2000; Pistore & Traverso 2001; Baral, Kreinovich, & Trejo 2001). Most of these papers – except (Pistore & Traverso 2001), only consider the case when actions are deterministic. In (Dal Lago, Pistore, & Traverso 2002), a question was raised regarding whether the existing temporal logics are adequate to specify many intuitive goals, especially in domains where actions have non-deterministic effects.

In this paper, we first show that in the case that actions have non-deterministic effects, many intuitive and useful goals cannot be expressed in existing temporal logics such as LTL, CTL$^*$ or $\pi$-CTL$^*$ (Baral & Zhao 2004). We then argue that for certain goals, we need two higher level quantifiers beyond the quantifiers already in CTL$^*$ and $\pi$-CTL$^*$. We show that our proposed temporal language with the two new quantifiers can indeed express many intuitive and useful goals that cannot be expressed in LTL, CTL$^*$ or $\pi$-CTL$^*$. We now start with a couple of motivating examples.

### Motivating examples

In a domain where actions have non-deterministic effects, plans are often policies (mapping from states to actions) instead of simple action sequences. Even then, in many domains an agent with a goal to reach a state where a certain fluent is true may not find a policy that can guarantee this. In that case, the agent may be willing to settle for less, such as having a strong cyclic policy (Cimatti *et al.* 2003), that always has a path to a desired state from any state in the policy, or even less, such as having a weak policy, that has a path from the initial state to a desired state. But the agent may want to choose among such different options based on their availability. The following example[1] illustrates such a case.

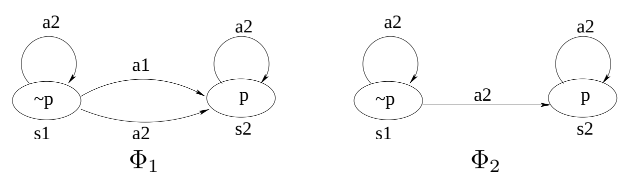


Figure 1: Two Transitions

Consider the two transition diagrams $\Phi_1$ and $\Phi_2$ of Figure 1, which may correspond to two distinct domains. In each state of the diagrams, there is always an action $nop$[2] that keeps the agent in the same state. The two diagrams have states $s_1$ and $s_2$, and actions $a_1$ and $a_2$. In the state $s_1$ the fluent $p$ is false, while $p$ is true in the state $s_2$. In both transition diagrams $a_2$ is a non-deterministic action which when executed in state $s_1$ may result in the transition to state

[1] Although in our examples, to save space, we use state space diagrams. These diagrams can easily be grounded on action descriptions. For an example see (Dal Lago, Pistore, & Traverso 2002).

[2] We have this assumption throughout the paper.

$s_2$ or may stay in $s_1$, and when executed in $s_2$ stays in $s_2$. On the other hand, the action $a_1$ is only present in the transition diagram $\Phi_1$ and if it is executed in $s_1$ then it causes the transition to $s_2$. Now suppose our agent, which is in the state $s_1$ (where $p$ is false), wants to get to $s_2$ where $p$ is true. Aware of the fact that actions could be non-deterministic and there may not always exist policies that can guarantee that our agent reaches $p$, our agent and its handlers are willing to settle for less, such as a strong cyclic policy, when the better option is not available. Thus the goal is 'guaranteeing to reach $p$ if that is possible and if not then making sure that $p$ is always reachable'.

For the domain corresponding to transition diagram $\Phi_2$, the policy $\pi$ which does action $a_2$ in $s_1$, is an acceptable policy. But it is not an acceptable policy for the domain corresponding to transition diagram $\Phi_1$, as there is a better option available there. In $\Phi_1$ if one were to execute $a_1$ in $s_1$ then one is guaranteed to reach $s_2$ where $p$ is true. Thus executing $a_2$ in $s_1$ is no longer acceptable. Hence, with respect to $\Phi_1$ only the policy ($\pi'$) that dictates that $a_1$ should be executed in $s_1$ is an acceptable policy.

We will show that the above discussed goal cannot be expressed using $\pi$-CTL$^*$, and CTL$^*$. To further elaborate on the kind of goals that cannot be expressed using these temporal logics, let us consider the following example in expressing various nuances of the goal of reaching a state:

**Example 1** *There are five different states:* $s_1$, $s_2$, $s_3$, $s_4$, *and* $s_5$. *The proposition* $p$ *is only true in state* $s_4$. *The other states are distinguishable based on fluents which we do not elaborate here. Suppose the only possible actions (besides* $nop$ *actions) and their consequences are as given below in Figure 2.*

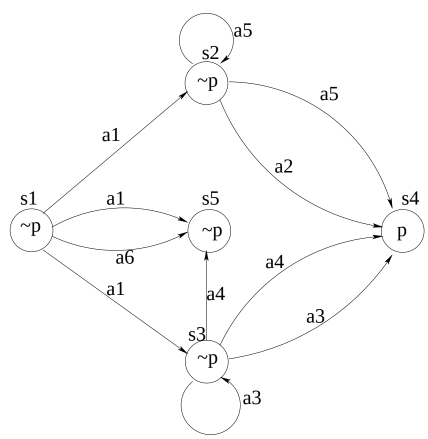


Figure 2: Transition between the locations

As before let us consider that the agent would like to try its best[3] to get to a state where $p$ is true. But 'Try your best' would then have a different meaning depending on where the agent is: In state $s_1$ doing $a_1$ is better than doing $a_6$, although neither guarantee that $s_4$ will be reached. $a_6$ makes the goal impossible. Similarly, at $s_2$, doing $a_2$ is better than doing $a_5$, and in $s_3$ doing $a_3$ is better than doing $a_4$. So the best policy seems to be to do $a_1$ in $s_1$, $a_2$ in $s_2$ and $a_3$ in $s_3$. But sometimes one may try to weaken this notion of trying its best by allowing some other actions besides the best choice in some states. Or the goal of the agent may not be to find the best policy but to get a policy with some compromised properties. To analyze this further, let us consider the following policies:

1. Policy $\pi_1 = \{(s_1, a_1), (s_2, a_2), (s_3, a_3)\}$
2. Policy $\pi_2 = \{(s_1, a_1), (s_2, a_2), (s_3, a_4)\}$
3. Policy $\pi_3 = \{(s_1, a_1), (s_2, a_5), (s_3, a_3)\}$
4. Policy $\pi_4 = \{(s_1, a_1), (s_2, a_5), (s_3, a_4)\}$
5. Policy $\pi_5 = \{(s_1, a_6)\}$

Figure 3 shows the relation between the five policies in terms of which one is preferable to the other with respect to the goal of trying ones best to get to a state where $p$ is true. A directed edge from $\pi_i$ to $\pi_j$ means $\pi_i$ is preferable to $\pi_j$ and this preference relation is transitive.

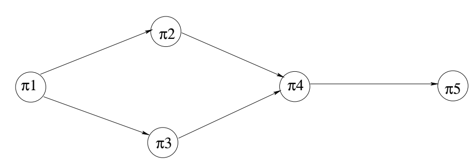


Figure 3: The preference relation between the policies

Using $\pi$-CTL$^*$ we are able to express a goal which when considered from the starting state $s_1$, considers $\pi_5$ to be an unacceptable policy, but considers the rest in $\{\pi_1, \ldots, \pi_5\}$ to be acceptable. We will argue that there is no specification in $\pi$-CTL$^*$ which only accepts $\pi_1$, and show how arbitrary partitions of $\{\pi_1, \ldots, \pi_5\}$ can be expressed when we have an enhanced language that allows quantification over policies.

By quantifying over policies, the agent may alter its expectation in the process of executing. For example, in terms of the goal of trying the best in reaching $p$, initially, the agent may not guarantee to reach $p$ due to the non-deterministic property of the domain. However, in the process of executing, it may be lucky enough to reach a state that $p$ can be guaranteed to reach. In finding the best policy, we may require the agent have to reach $p$ from then on. In (Pistore & Traverso 2001; Dal Lago, Pistore, & Traverso 2002), the authors also tried to capture the intuition of modifying the plan during the execution, but their method is insufficient in doing so (Baral & Zhao 2004).

Quantifying over policies also takes the difficulties of the domain into account when we specify the goal. Again, consider the goal of trying the best to reach $p$. Even if the agent give up with the answer that "$p$ is not reachable" since the domain is indeed impossible to reach $p$, we may still regard

[3] Note that special cases of 'try your best' are the well-studied (in AI) notions of strong planing, strong cyclic planning, and weak planning (Cimatti *et al.* 2003), and **TryReach p** of (Dal Lago, Pistore, & Traverso 2002).

the agent satisfies its goal since it has already "tried its best" in reaching $p$. That is to say, whether a policy satisfying a goal depends on the current domain. Thus goal specifications are adaptive to domains.

Overall, our main contributions in this paper are:

- Extending temporal logics for goal specification in non-deterministic domains by quantifying over policies;
- Proposing mechanisms and using them in formally comparing expressiveness of goal specification languages.

## Background: $\pi$-CTL$^*$

To show the limitations of the expressibility of $\pi$-CTL$^*$ we now give an overview of $\pi$-CTL$^*$.

### Syntax of $\pi$-CTL$^*$

The syntax of state and path formulas in $\pi$-CTL$^*$ is as follows. Let $\langle p \rangle$ denote an atomic proposition, $\langle sf \rangle$ denote a state formula, and $\langle pf \rangle$ denote a path formula.

$$\langle sf \rangle ::= \langle p \rangle \mid \langle sf \rangle \wedge \langle sf \rangle \mid \langle sf \rangle \vee \langle sf \rangle \mid \neg \langle sf \rangle \mid \mathsf{E}\langle pf \rangle \mid \mathsf{A}\langle pf \rangle \mid \mathsf{E}_\pi \langle pf \rangle \mid \mathsf{A}_\pi \langle pf \rangle$$

$$\langle pf \rangle ::= \langle sf \rangle \mid \langle pf \rangle \vee \langle pf \rangle \mid \neg \langle pf \rangle \mid \langle pf \rangle \wedge \langle pf \rangle \mid \langle pf \rangle \,\mathsf{U}\, \langle pf \rangle \mid \bigcirc \langle pf \rangle \mid \Diamond \langle pf \rangle \mid \Box \langle pf \rangle$$

As in CTL$^*$, the symbol $\mathsf{A}$ means 'for all paths', the symbol $\mathsf{E}$ means 'exists a path', and the linear temporal logic symbols $\Box$, $\Diamond$, $\bigcirc$ and $\mathsf{U}$ stands for '$always$', '$eventually$', '$next$', and '$until$' respectively. The symbols $\mathsf{A}_\pi$ and $\mathsf{E}_\pi$ are the branching time operators meaning 'for all paths that agree with the policy that is being executed' and 'there exists a path that agrees with the policy that is being executed' respectively.

### Formal semantics of $\pi$-CTL$^*$

In $\pi$-CTL$^*$, policies are mappings from the set of states to the set of actions. The semantics of $\pi$-CTL$^*$ is similar to that of CTL$^*$. Here we present a simplification (but equivalent) of the characterization of $\pi$-CTL$^*$ given in (Baral & Zhao 2004). Since our actions may have non-deterministic effects we consider mappings, $\Phi$, from states and actions to sets of states. Next we need the following definitions.

**Definition 1 (Paths in $\Phi$ starting from $s$)**
(i) A path in $\Phi$ starting from a state $s$ is an infinite trajectory $s = s_0, s_1, \cdots$ such that $s_{i+1} \in \Phi(s_i, a_i)$, $0 \leq i$, for some action $a_i$.
(ii) A path in $\Phi$ starting from a state $s$ consistent with a policy $\pi$ is an infinite trajectory $s = s_0, s_1, \cdots$ such that $s_{i+1} \in \Phi(s_i, \pi(s_i))$, $0 \leq i$. $\Box$

**Definition 2 (Truth of state formulas in $\pi$-CTL$^*$)** The truth of state formulas are defined with respect to a triple $(s_j, \Phi, \pi)$ where $s_j$ is a state, $\Phi$ is the transition function, and $\pi$ is the policy.

- $(s_j, \Phi, \pi) \models p$ iff $p$ is true in $s_j$.
- $(s_j, \Phi, \pi) \models \neg sf$ iff $(s_j, \Phi, \pi) \not\models sf$.
- $(s_j, \Phi, \pi) \models sf_1 \wedge sf_2$ iff $(s_j, \Phi, \pi) \models sf_1$ and $(s_j, \Phi, \pi) \models sf_2$.
- $(s_j, \Phi, \pi) \models sf_1 \vee sf_2$ iff $(s_j, \Phi, \pi) \models sf_1$ or $(s_j, \Phi, \pi) \models sf_2$.
- $(s_j, \Phi, \pi) \models \mathsf{E}\, pf$ iff there exists a path $\sigma$ in $\Phi$ starting from $s_j$ such that $(s_j, \Phi, \pi, \sigma) \models pf$.
- $(s_j, \Phi, \pi) \models \mathsf{A}\, pf$ iff for all paths $\sigma$ in $\Phi$ starting from $s_j$ we have that $(s_j, \Phi, \pi, \sigma) \models pf$.
- $(s_j, \Phi, \pi) \models \mathsf{E}_\pi\, pf$ iff there exists a path $\sigma$ in $\Phi$ starting from $s_j$ consistent with the policy $\pi$ such that $(s_j, \Phi, \pi, \sigma) \models pf$.
- $(s_j, \Phi, \pi) \models \mathsf{A}_\pi\, pf$ iff for all paths $\sigma$ in $\Phi$ starting from $s_j$ consistent with the policy $\pi$ we have that $(s_j, \Phi, \pi, \sigma) \models pf$. $\Box$

**Definition 3 (Truth of path formulas in $\pi$-CTL$^*$)** The truth of path formulas are now defined with respect to the quadruplet $(s_j, \Phi, \pi, \sigma)$, where $s_j$, $\Phi$, and $\pi$ are as before and $\sigma$ is an infinite sequence of states $s_j, s_{j+1}, \ldots$, called a path.

- $(s_j, \Phi, \pi, \sigma) \models sf$ iff $(s_j, \Phi, \pi) \models sf$.
- $(s_j, \Phi, \pi, \sigma) \models \neg pf$ iff $(s_j, \Phi, \pi, \sigma) \not\models pf$
- $(s_j, \Phi, \pi, \sigma) \models pf_1 \wedge pf_2$ iff $(s_j, \Phi, \pi, \sigma) \models pf_1$ and $(s_j, \Phi, \pi, \sigma) \models pf_2$.
- $(s_j, \Phi, \pi, \sigma) \models pf_1 \vee pf_2$ iff $(s_j, \Phi, \pi, \sigma) \models pf_1$ or $(s_j, \Phi, \pi, \sigma) \models pf_2$.
- $(s_j, \Phi, \pi, \sigma) \models \bigcirc pf$ iff $(s_{j+1}, \Phi, \pi, \sigma) \models pf$.
- $(s_j, \Phi, \pi, \sigma) \models \Box pf$ iff $(s_k, \Phi, \pi, \sigma) \models pf$, for all $k \geq j$.
- $(s_j, \Phi, \pi, \sigma) \models \Diamond pf$ iff $(s_k, \Phi, \pi, \sigma) \models pf$, for some $k \geq j$.
- $(s_j, \Phi, \pi, \sigma) \models pf_1 \,\mathsf{U}\, pf_2$ iff there exists $k \geq j$ such that $(s_k, \Phi, \pi, \sigma) \models pf_2$, and for all $i$, $j \leq i < k$, $(s_i, \Phi, \pi, \sigma) \models pf_1$. $\Box$

Note that in the above definition, $\sigma$ is not required to be consistent with $\pi$.

### Policies for $\pi$-CTL$^*$ goals

We now define the notion of a policy w.r.t. a $\pi$-CTL$^*$ goal $G$, an initial state $s_0$, and a transition function $\Phi$.

**Definition 4 (Policy for a goal from an initial state)** Given an initial state $s_0$, a policy $\pi$, a transition function $\Phi$, and a goal $G$ we say that $\pi$ is a policy for $G$ from $s_0$, iff $(s_0, \Phi, \pi) \models G$. $\Box$

From the above definition, goals in $\pi$-CTL$^*$ are state formulas. It can be shown that under comparable notions of goals and policies, $\pi$-CTL$^*$ is syntactically a proper superset of CTL$^*$ and is strictly more expressive.

## Expressiveness limitations of $\pi$-CTL$^*$

We consider a goal expressed in natural languages as an intuitive goal. In this section, we prove that some intuitive goals cannot be expressed in $\pi$-CTL$^*$. For that we need the following lemma about the transition diagrams $\Phi_1$ and $\Phi_2$ of Figure 1.

**Lemma 1** *Consider $\Phi_1$, $\Phi_2$ in Figure 1, and $\pi$ as $\pi(s_1) = \pi(s_2) = a_2$.*

*(i) For any state formula $\varphi$ in $\pi$-CTL$^*$, $(s_1, \Phi_1, \pi) \models \varphi$ iff $(s_1, \Phi_2, \pi) \models \varphi$.*

*(ii) For any path formula $\psi$ in $\pi$-CTL$^*$ and any path $\sigma$ in $\Phi_1$ (or $\Phi_2$) $(s_1, \Phi_1, \pi, \sigma) \models \psi$ iff $(s_1, \Phi_2, \pi, \sigma) \models \psi$.*

The proof is done by induction on the depth of the formula. An atomic propositions has depth 1 and addition of any connectives increases the depth.

**Proposition 1** There exists a goal which cannot be expressed in $\pi$-CTL$^*$.

**Proof:** (sketch) Consider the following intuitive goal $G$:

"*All along your trajectory*
*<u>if</u> from any state $p$ can be achieved for sure*
*<u>then</u> the policy being executed must achieve $p$,*
*<u>else</u> the policy must make $p$ reachable from any state in the trajectory.*"

Let us assume that $G$ can be expressed in $\pi$-CTL$^*$ and let $\varphi_G$ be its encoding in $\pi$-CTL$^*$. From our intuitive understanding of $G$, $(s_1, \Phi_2, \pi)$ satisfies the goal $\varphi_G$ while $(s_1, \Phi_1, \pi)$ does not. This contradicts with Lemma 1, and hence our assumption is wrong.

□

# P-CTL$^*$: need for higher level quantifiers

Let us further analyze the goal $G$ from the previous section. While the <u>then</u> and <u>else</u> part of $G$ can be expressed in $\pi$-CTL$^*$, the <u>if</u> part can be further elaborated as "*there exists a policy which guarantees that $p$ can be achieved for sure*", and to express that, one needs to quantify over policies. Thus we introduce a new existence quantifier $\mathcal{EP}$ and its dual $\mathcal{AP}$, meaning 'there exists a policy starting from the state' and 'for all policies starting from the state' respectively.

## Syntax of P-CTL$^*$

We extend the syntax of $\pi$-CTL$^*$ to incorporate the above mentioned two new quantifiers. Let $\langle p \rangle$ denote an atomic proposition, $\langle sf \rangle$ denote a state formula, and $\langle pf \rangle$ denote a path formula. Intuitively, state formulas are properties of states, path formulas are properties of paths. With that the syntax of state and path formulas in P-CTL$^*$ is as follows.

$$\langle sf \rangle ::= \langle p \rangle \mid \langle sf \rangle \wedge \langle sf \rangle \mid \langle sf \rangle \vee \langle sf \rangle \mid \neg \langle sf \rangle \mid \mathsf{E}\langle pf \rangle \mid \mathsf{A}\langle pf \rangle \mid \mathsf{E}_\pi \langle pf \rangle \mid \mathsf{A}_\pi \langle pf \rangle \mid \mathcal{EP}\langle sf \rangle \mid \mathcal{AP}\langle sf \rangle$$

$$\langle pf \rangle ::= \langle sf \rangle \mid \langle pf \rangle \vee \langle pf \rangle \mid \neg \langle pf \rangle \mid \langle pf \rangle \wedge \langle pf \rangle \mid \langle pf \rangle \,\mathsf{U}\, \langle pf \rangle \mid \bigcirc \langle pf \rangle \mid \Diamond \langle pf \rangle \mid \Box \langle pf \rangle$$

Note that in the above definition we have $\mathcal{EP}\langle sf \rangle$ as a state formula. That is because once the policy part of $\mathcal{EP}$ is instantiated, the reminder of the formula is still a property of a state. The only difference is that a policy has been instantiated and that policy needs to be followed for the rest of the formula. We now define the semantics of P-CTL$^*$.

## Semantics of P-CTL$^*$

The semantics of P-CTL$^*$ is very similar to the semantics of $\pi$-CTL$^*$. For brevity we only show the part where they differ. We first need the following notion: We say that a policy $\pi$ is consistent with respect to a transition function $\Phi$ if for all states $s$, $\Phi(s, \pi(s))$ is a non-empty set.

**Definition 5 (Truth of state formulas in $\pi$-CTL$^*$)** The truth of state formulas are defined with respect to a triple $(s_j, \Phi, \pi)$ where $s_j$ is a state, $\Phi$ is the transition function, and $\pi$ is a policy.

- $(s_j, \Phi, \pi) \models p$, $(s_j, \Phi, \pi) \models \neg sf$, $(s_j, \Phi, \pi) \models sf_1 \wedge sf_2$, $(s_j, \Phi, \pi) \models sf_1 \vee sf_2$, $(s_j, \Phi, \pi) \models \mathsf{E}\, pf$, $(s_j, \Phi, \pi) \models \mathsf{A}\, pf$, $(s_j, \Phi, \pi) \models \mathsf{E}_\pi\, pf$, and $(s_j, \Phi, \pi) \models \mathsf{A}_\pi\, pf$ are defined as in $\pi$-CTL$^*$.
- $(s_j, \Phi, \pi) \models \mathcal{EP}\ sf$ iff there exists a policy $\pi'$ consistent with $\Phi$ such that $(s_j, \Phi, \pi') \models sf$.
- $(s_j, \Phi, \pi) \models \mathcal{AP}\ sf$ iff for all policies $\pi'$ consistent with $\Phi$, we have $(s_j, \Phi, \pi') \models sf$. □

**Definition 6 (Truth of Path Formulas)** The truth of path formulas are now defined with respect to the quadruple $(s_j, \Phi, \pi, \sigma)$, where $s_j, \Phi$ and $\pi$ are as before and $\sigma$ is an infinite sequence of states $s_j, s_{j+1}, \ldots$, called a path.

- $(s_j, \Phi, \pi, \sigma) \models sf$, $(s_j, \Phi, \pi, \sigma) \models \neg pf$, $(s_j, \Phi, \pi, \sigma) \models pf_1 \wedge pf_2$, $(s_j, \Phi, \pi, \sigma) \models pf_1 \vee pf_2$, $(s_j, \Phi, \pi, \sigma) \models \bigcirc pf$, $(s_j, \Phi, \pi, \sigma) \models \Box pf$, $(s_j, \Phi, \pi, \sigma) \models \Diamond pf$, and $(s_j, \Phi, \pi, \sigma) \models pf_1 \,\mathsf{U}\, pf_2$ are defined as in $\pi$-CTL$^*$. □

## Policies for P-CTL$^*$ goals

We now define when a mapping $\pi$ from states to actions is a policy with respect to a P-CTL$^*$ goal $G$, an initial state $s_0$, and a transition function $\Phi$.

**Definition 7 (Policy for a goal from an initial state)** Given an initial state $s_0$, a policy $\pi$, a transition function $\Phi$, and a goal $G$ we say that $\pi$ is a policy for $G$ from $s_0$, if $(s_0, \Phi, \pi) \models G$. □

# Goal representation in P-CTL$^*$

In this section, we illustrate several goal examples that can be expressed in P-CTL$^*$ while cannot be expressed in $\pi$-CTL$^*$ or CTL$^*$. To start with, the overall goal of a planning problem is best expressed as a state formula. But if the goal starts with a quantifier over policies then it is not quite suitable to test the validity $(s_0, \Phi, \pi) \models G$ of a given policy $\pi$, as then the $\pi$ will be ignored. Therefore most meaningful goal formulas, denoted by $gf$ are given by the following:

$$\langle gf \rangle ::= \langle gf \rangle \wedge \langle gf \rangle \mid \langle gf \rangle \vee \langle gf \rangle \mid \neg \langle gf \rangle \mid \mathsf{E}_\pi \langle pf \rangle \mid \mathsf{A}_\pi \langle pf \rangle$$

Many of the goals we will be presenting in this section will be with respect to Example 1 in the introduction section. But first we start with some building blocks that can be expressed in $\pi$-CTL$^*$.

• $G^\pi_w = \mathsf{E}_\pi \Diamond p$: This goal specifies that from the initial state, a state where $p$ is true may be reached by following the given policy. This is referred to as weak planning.

• $G^\pi_s = \mathsf{A}_\pi \Diamond p$: This goal specifies that from the initial state, a state where $p$ is true, will be reached by following the given policy. This is referred to as strong planning.

• $G^\pi_m = \mathsf{A}_\pi \Box q$: This goal specifies that from the initial state, $q$ is true all along the trajectory.

• $G^\pi_{sc} = \mathsf{A}_\pi \Box(\mathsf{E}_\pi \Diamond p)$: This goal specifies that all along the trajectory – following the given policy – there is always a possible path to a state where $p$ is true. This is referred to as strong cyclic planning.

Now we use the new quantifiers in P-CTL$^*$ to express conditionals similar to the one mentioned in the beginning of the previous Section.

• $C_w = \mathcal{EP}\mathsf{E}_\pi \Diamond p$: This is a state formula, which characterizes states with respect to which (i.e., if that state is considered as an initial state) there is a policy. If one were to follow that policy then one can, but not guaranteed to, reach a state where $p$ is true.

• $C_s = \mathcal{EP}\mathsf{A}_\pi \Diamond p$: This is a state formula, which characterizes states with respect to which there is a policy such that if one were to follow that policy then one is guaranteed to reach a state where $p$ is true.

• $C_{sc} = \mathcal{EP}\mathsf{A}_\pi \Box(\mathsf{E}_\pi \Diamond p)$: This is a state formula, which characterizes states with respect to which there is a policy such that if one were to follow that policy then all along the trajectory there is always a possible path to a state where $p$ is true.

The above three formulas are not expressible in $\pi$-CTL$^*$, and are state formulas of P-CTL$^*$. But, by themselves they, or a conjunction, disjunction or negation of them, are not meaningful goal formulas with respect to which one would try to develop policies (or plan) for. Indeed, they do not obey the syntax of meaningful goal formulas, $\langle gf \rangle$, given earlier in this section. Nevertheless, they are very useful building blocks.

Recall goal $G$ in the proof of Proposition 1. It can be expressed in P-CTL$^*$ as $G^P_{p,q} = \mathsf{A}_\pi \Box((\mathcal{EP}\mathsf{A}_\pi \Diamond p \Rightarrow \mathsf{A}_\pi \Diamond p) \wedge (\neg \mathcal{EP}\mathsf{A}_\pi \Diamond p \Rightarrow \mathsf{A}_\pi \Box(\mathsf{E}_\pi \Diamond p)))$. In Figure 1, policy $\pi'_1 = \{(s_1, a_2), (s_2, a_2)\}$ achieves the goal $G^P_{p,q}$ with respect to $\Phi_2$, but not with respect to $\Phi_1$, while policy $\pi'_2 = \{(s_1, a_1), (s_2, a_2)\}$ achieves the goal $G^P_{p,q}$ with respect to $\Phi_1$. The reason $\pi'_1$ does not satisfy the goal with respect to $\Phi_1$ is that $\mathcal{EP}\mathsf{A}_\pi \Diamond p$ is true with respect to $s_1$ (in $\Phi_1$), but the policy $\pi'_1$ does not satisfy $\mathsf{A}_\pi \Diamond p$.

## Goals corresponding to Example 1

We now use the conditionals $C_s$, $C_w$ and $C_{sc}$ and the $\pi$-CTL$^*$ formulas $G^\pi_s$, $G^\pi_c$, $G^\pi_{sc}$, and $G^\pi_m$ to express various goals with respect to Example 1.

- $G^P_w = \mathsf{A}_\pi \Box(\mathcal{EP}\mathsf{E}_\pi \Diamond p \Rightarrow \mathsf{E}_\pi \Diamond p)$: This goal specifies that all along the trajectory following the given policy, if there is a policy that makes $p$ reachable then the given policy makes $p$ reachable. The policies $\pi_1$, $\pi_2$, $\pi_3$ and $\pi_4$ satisfy this goal while $\pi_5$ does not.
- $G^P_s = \mathsf{A}_\pi \Box(\mathcal{EP}\mathsf{A}_\pi \Diamond p \Rightarrow \mathsf{A}_\pi \Diamond p)$: This goal specifies that all along the trajectory following the given policy, if there is a policy that can always reach $p$ no matter the non-deterministic actions, then in the policy chosen by the agent, $p$ must be reached. The policies $\pi_1$, $\pi_2$ and $\pi_5$ satisfy this goal while $\pi_3$ and $\pi_4$ do not.
- $G^P_{sc} = \mathsf{A}_\pi \Box(\mathcal{EP}\mathsf{A}_\pi \Box(\mathsf{E}_\pi \Diamond p) \Rightarrow \mathsf{A}_\pi \Box(\mathsf{E}_\pi \Diamond p))$: This goal specifies that all along the trajectory following the given policy, if there is a policy that is a strong cyclic policy for $p$, then the policy chosen by the agent is a strong cyclic policy for $p$. The policies $\pi_1$, $\pi_3$, and $\pi_5$ satisfy this goal while policies $\pi_2$ and $\pi_4$ do not.
- $G^P_4 = G^P_s \wedge G^P_{sc} \wedge G^P_w$: This goal specifies that all along the trajectory following the given policy, if there is a policy that guarantees that $p$ will be reached, then the agent's policy must guarantee to reach $p$; else-if there is a strong cyclic policy for $p$, then the policy chosen by the agent must be a strong cyclic policy; and else-if there is a policy that makes $p$ reachable then the policy makes $p$ reachable. This can be considered as formal specification of the goal of "trying ones best to reach $p$". Only $\pi_1$, among $\pi_1 - \pi_5$ satisfies this goal.

| Goal | Satisfiable policies |
|---|---|
| $G^\pi_w, G^P_w$ | $\pi_1$, $\pi_2$, $\pi_3$, $\pi_4$ |
| $G^P_{sc}$ | $\pi_1$, $\pi_3$, $\pi_5$ |
| $G^P_s$ | $\pi_1$, $\pi_2$, $\pi_5$ |
| $G^P_w \wedge G^P_s$ | $\pi_1$, $\pi_2$ |
| $G^P_w \wedge G^P_{sc}$ | $\pi_1$, $\pi_3$ |
| $G^P_w \wedge G^P_s \wedge G^P_{sc}$ | $\pi_1$ |
| $G^P_s \wedge \neg G^P_{sc}$ | $\pi_2$ |
| $G^P_{sc} \wedge \neg G^P_s$ | $\pi_3$ |
| $G^P_w \wedge \neg G^P_{sc} \wedge \neg G^P_s$ | $\pi_4$ |
| $G^P_s \wedge \neg G^P_w$ | $\pi_5$ |
| $G^\pi_s$ | $\emptyset$ |

Table 1: Different P-CTL$^*$ and $\pi$-CTL$^*$ goal specifications and policies satisfied

Based on these formulations, we may have various specifications. Some of these specifications and the subset of the policies $\pi_1 - \pi_5$ that satisfy these goals are summarized in Table 1. In this example, we have arbitrary partition of $\{\pi_1, \cdots, \pi_5\}$, while most of these partitions cannot be done in existing languages. Language P-CTL$^*$ has more power in expressing our intention of comparing among policies.

## Some more goals specified in P-CTL$^*$

We illustrate the P-CTL$^*$ specification of some goals involving two propositions, $p$ and $q$. In particular, the additional expressive power is not just for expressing the "if-then" type of conditions discussed earlier.

- Suppose there is an agent that would like to reach $q$ but wants to make sure that all along the path if necessary it can make a new (contingent) policy that can guarantee that $p$ will be reached. Here, $q$ may be a destination of a robot and $p$ may be the property of locations that have recharging stations. This goal can be expressed in P-CTL* as $\mathsf{A}_\pi \Box((\mathcal{EP}\mathsf{A}_\pi \Diamond p)\mathsf{U} q)$. Alternative specifications in CTL* or $\pi$-CTL* do not quite capture this goal.
- Consider an agent that would like to reach either $p$ or $q$, but because of non-determinism the agent is satisfied if all along its path at least one of them is reachable, but at any point if there is a policy that guarantees that $p$ will be reached then from that point onwards the agent should make sure that $p$ is reached, otherwise, if at any point if there is a policy that guarantees that $q$ will be reached then from that point onwards the agent should make sure that $q$ is reached. This can be expressed in P-CTL* as $\mathsf{A}_\pi \Box(\mathsf{E}_\pi(p \vee q) \wedge (\mathcal{EP}\mathsf{A}_\pi \Diamond p \Rightarrow \mathsf{A}_\pi \Diamond p) \wedge ((\neg \mathcal{EP}\mathsf{A}_\pi \Diamond p \wedge \mathcal{EP}\mathsf{A}_\pi \Diamond q) \Rightarrow \mathsf{A}_\pi \Diamond q))$.
- Consider an agent whose goal is to maintain $p$ true and if that is not possible for sure then it must maintain $q$ true until $p$ becomes true. This can be expressed in P-CTL* as $\mathsf{A}_\pi \Box((\mathcal{AP}\mathsf{E}_\pi \neg \Box p \Rightarrow \mathsf{A}_\pi(q \mathsf{U} p)) \wedge (\mathcal{EP}\mathsf{A}_\pi \Box p \Rightarrow \mathsf{A}_\pi \Box p))$.

## Framework for comparing goal languages

In this section, we present a general notion for comparing expressiveness of goal specification languages. Let $L$ be a goal specification language. Let $g$ be a goal formula in $L$, $\Phi$ be a transition function, $\models_L$ be the entailment relation in language $L$, and $s_0$ be an initial state. As defined in Definition 7, a policy $\pi$ is a plan for the goal $g$ from state $s_0$ if $(s_0, \pi, \Phi) \models_L g$. We use $Pset(g, \Phi, s_0, \models_L)$ to denote the set $\{\pi : (s_0, \pi, \Phi) \models_L g\}$ as the set of policies satisfying $g$ in $L$. By $Gset(\pi, \Phi, s_0, \models_L)$, we denote the set $\{g : (s_0, \pi, \Phi) \models_L g\}$ as the set of goal formulas satisfied by policy $\pi$ in $L$. Let $G_L$ be all goal formulas in language $L$. Let $P_L(\Phi)$ be all policies in $\Phi$ corresponding to language $L$.

**Definition 8** *An intuitive goal $g$ is* not expressible *in a goal specification language $L$ if there are $\Phi_1$, $\Phi_2$, $s_0^1$, $s_0^2$ such that*

1. *For any goal specification $g_1$ in $G_L$,*
   $Pset(g_1, \Phi_1, s_0^1, \models_L) \cap P_L(\Phi_2) = Pset(g_1, \Phi_2, s_0^2, \models_L) \cap P_L(\Phi_1)$*;*
2. *There is a policy $\pi_1 \in P_L(\Phi_1) \cap P_L(\Phi_2)$ such that intuitively $\pi_1$ is a policy for the goal $g$ w.r.t. $(\Phi_1, s_0^1, \models_L)$ but not w.r.t. $(\Phi_2, s_0^2, \models_L)$.*

Note that the proof of Proposition 1 uses a similar notion. There and as well as above, we need to appeal to intuition. To make it formal we need to specify what policies "intuitively" satisfy a goal $g$ with respect to a given $\Phi$ and an initial state $s_0$. Alternatively, we can compare two formally defined goal languages. The following definition allows us to do that.

**Definition 9** *Consider two languages $L_1$ and $L_2$.* $L_1$ is more expressive (in a conservative sense) than $L_2$ *if*

1. $G_{L_2} \subseteq G_{L_1}$*;*
2. $\forall \Phi$, $P_{L_2}(\Phi) \subseteq P_{L_1}(\Phi)$*;*
3. $\forall g \in G_{L_2}$, $\forall \Phi$, $\forall s_0$*:*
   $Pset(g, \Phi, s_0, \models_{L_2}) = Pset(g, \Phi, s_0, \models_{L_1}) \cap P_{L_2}(\Phi)$ *;*
4. $\forall \Phi, \forall \pi \in P_{L_2}(\Phi)$, $\forall s_0$*:*
   $Gset(\pi, \Phi, s_0, \models_{L_2}) = Gset(\pi, \Phi, s_0, \models_{L_1}) \cap G_{L_2}$.

**Proposition 2** *If language $L_1$ is more expressive than $L_2$, and for all $\Phi$, $P_{L_1}(\Phi) = P_{L_2}(\Phi)$, then any intuitive goal that can be expressed in $L_2$ can be expressed in $L_1$.*

With respect to the above definition, we now compare the languages $P$-CTL*, $\pi$-CTL*, $P_t$-CTL* and $\pi_t$-CTL*, where the last two languages extend the former two by allowing policies to be mappings from trajectories of states to actions.

**Proposition 3** *1. $P$-CTL* is more expressive than $\pi$-CTL*.*

2. *There exists an intuitive goal than can be expressed in $P$-CTL* but not in $\pi$-CTL*.*
3. *$P_t$-CTL* is more expressive than $\pi_t$-CTL*.*
4. *There exists an intuitive goal than can be expressed in $P_t$-CTL* but not in $\pi_t$-CTL*.*
5. *$\pi_t$-CTL* is more expressive than $\pi$-CTL*.*
6. *$P_t$-CTL* is not more expressive than $P$-CTL*.*

## Conclusions

Systematic design of semi-autonomous agents involves specifying (i) the domain description: the actions the agent can do, its impact, the environment, etc.; (ii) directives for the agent; and (iii) the control execution of the agent. While there has been a lot of research on (i) and (iii), there has been relatively less work on (ii). In this paper we made amends and explored the expressive power of existing temporal logic based goal specification languages. We showed that in presence of actions with non-deterministic effects many interesting goals cannot be expressed using existing temporal logics such as CTL* and $\pi$-CTL*. We gave a formal proof of this. We then illustrated the necessity of having new quantifiers which we call "exists policy" and "for all policies" and developed the language P-CTL* which builds up on $\pi$-CTL* and has the above mentioned new quantifiers. We showed how many of the goals that cannot be specified in $\pi$-CTL* can be specified in P-CTL*.

In terms of closely related work, we discovered that quantification over policies was proposed in the context of games in the langauge ATL (Alur, Henzinger, & Kupferman 2002). Recently, an extension of that called CATL (van der Hoek, Jamroga, & Wooldridge 2005) has also been proposed. However in both of those languages the focus is on games and single transitions are deterministic. In our case we have a single agent and the transitions could be non-deterministic. It is not obvious that one can have a 1-1 correspondence between those formalisms and ours. In particular, their exact definitions on game structures require each state to have an action for each agent. This makes the obvious translation from their formalism (two person games with deterministic transitions) to our formalism (one person, but with non-deterministic transitions) not equivalent. Moreover we still allow constructs such as $\mathsf{E}\Diamond p$ which can no longer be expressed in game structures.

An interesting aspect of our work is that it illustrates the difference between program specification and goal specification. Temporal logics were developed in the context of program specification, where the program statements are deterministic and there are no goals of the kind “trying ones best”. (Its unheard of to require that a program try its best to sort.) In cognitive robotics actions have non-deterministic effects and sometimes one keeps trying until one succeeds, and similar attempts to try ones best. The proposed language P-CTL$^*$ allows the specification of such goals. P-CTL$^*$ has the ability of letting the agent to compare and analysis policies and “adjust” its current domain accordingly. As a consequence, it is useful for agent to plan in a non-deterministic or dynamic domains in which current states are unpredictable.

## Acknowledgments

We thank Vladimir Lifschitz for the proof of Proposition 1. We also acknowledge support of the NSF grant 0412000 and the support from the ARDA AQUAINT program.

## References

Alur, R.; Henzinger, T. A.; and Kupferman, O. 2002. Alternating-time temporal logic. *Journal of the ACM* 49:672–713.

Bacchus, F., and Kabanza, F. 1998. Planning for temporally extended goals. *Annals of Math and AI* 22:5–27.

Baral, C., and Zhao, J. 2004. Goal specification in presence of non-deterministic actions. In *Proceedings of ECAI’04*, 273–277.

Baral, C.; Kreinovich, V.; and Trejo, R. 2001. Computational complexity of planning with temporal goals. In *IJCAI-01*, 509–514.

Cimatti, A.; Pistore, M.; Roveri, M.; and Travrso, P. 2003. Weak, strong, and strong cyclic planning via symbolic model checking. *Artificial Intelligence* 147(1-2):35–84.

Dal Lago, U.; Pistore, M.; and Traverso, P. 2002. Planning with a language for extended goals. In *AAAI’02*, 447–454.

Emerson, E. A., and Clarke, E. 1982. Using branching time temporal logic to synthesize synchronization skeletons. In *Science of Computer programming, vol 2*. 241–266.

Niyogi, R., and Sarkar, S. 2000. Logical specification of goals. In *Proc. of 3rd international conference on Information Technology*, 77–82.

Pistore, M., and Traverso, P. 2001. Planning as model checking for extended goals in non-deterministic domains. In *IJCAI’01*.

van der Hoek, W.; Jamroga, W.; and Wooldridge, M. 2005. A logic for strategic reasoning. In *AAMAS-05*, 157–164.