Planning for Temporally Extended Goals based on alpha-CTL

Extended Abstract

Viviane Bonadia dos Santos
Universidade de São Paulo
São Paulo, Brasil
vbonadia@ime.usp.br

Leliane Nunes de Barros
Universidade de São Paulo
São Paulo, Brasil
leliane@ime.usp.br

Maria Viviane de Menezes
Universidade Federal do Ceará
Quixadá, Brasil

Silvio do Lago Pereira
Faculdade de Tecnologia de São Paulo
São Paulo, Brasil

ABSTRACT

We present the PACTL-XR planner for FOND planning, a planner based on symbolic model checking and α-CTL logic. The experiments show that our planner can efficiently find policies for complex planning goals, such extended reachability goals, and outperform the results of state-of-the-art planners in some domains.

KEYWORDS

FOND planning, Symbolic Model Checking, complex goals

ACM Reference Format:

Viviane Bonadia dos Santos, Leliane Nunes de Barros, Maria Viviane de Menezes, and Silvio do Lago Pereira. 2025. Planning for Temporally Extended Goals based on alpha-CTL: Extended Abstract. In *Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Detroit, Michigan, USA, May 19 – 23, 2025*, IFAAMAS, 3 pages.

1 INTRODUCTION

Automated planning is a long-standing field of AI that relies on a high-level description language to define the agent's planning task and reason over such definition to automatically generate a plan. A *Fully Observable Non-Deterministic* (FOND) planning problem assumes uncertainty over the action effects. The objective is to automatically synthesize a *policy*, a mapping between states and actions, that can lead the agent toward the goal, possibly with a loop from which the agent can eventually exit. In a real scenario the agent can either reach the goal or a *dead-end*, i.e. a state from which the agent can no longer achieve its goal. An R-Goal is a *simple reachability goal* specified by a formula that must be satisfied in a set G of terminal states. A more complex goal is an XR-Goal, *extended reachability goal*, that specifies an extra property which must hold in all states along the path to $s_g \in G$.

In this work, we build upon a previously proposed FOND planner, based on temporal logic, to design a new FOND planner called PACTL-XR, which can reason about temporally extended goals and is based on model checking and α-CTL logic, a branching time temporal logical that considers actions in its semantics. We also implemented a symbolic version called PACTL-XR-Sym. The experiments show that our planner can efficiently find policies for complex planning goals, such XR-Goal. We also show that when solving an XR-Goal that includes knowledge about dead-ends, we can simplify the strong-cyclic solution and outperform the results of state-of-the-art FOND planners in some domains.

Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Y. Vorobeychik, S. Das, A. Nowé (eds.), May 19 – 23, 2025, Detroit, Michigan, USA. © 2025 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

2 FOND PLANNING FOUNDATION

A planning domain describes the environment dynamics and a task defines the initial state and the goal. Formally, a FOND *planning domain* is a tuple $\mathcal{D} = \langle \mathcal{S}, \mathcal{L}, \mathcal{T} \rangle$ over a set of propositional atoms $\mathbb{P}$ and a set of actions $\mathbb{A}$, such that: $\mathcal{S}$ is a finite set of states; $\mathcal{L} : \mathcal{S} \mapsto 2^{\mathbb{P}}$ is a state labeling function; and $\mathcal{T} : \mathcal{S} \times \mathbb{A} \to 2^{\mathcal{S}}$ is a non-deterministic state transition function such that, given a state $s \in S$ and an action $a \in \mathbb{A}$, $\mathcal{T}$ returns a set of possible next states.

A FOND *planning task* is a tuple $\mathcal{P} = \langle \mathcal{D}, s_0, \varphi \rangle$, where $\mathcal{D}$ is the planning domain, $s_0 \in \mathcal{S}$ is the initial state and φ is a logical goal formula. For an R-Goal, φ is a propositional formula that must be satisfied in a set of goal states G, i.e. $s_g \models \varphi$ if $s_g \in G$. For an XR-Goal, φ is given by a pair of logical formulae i.e. $\varphi = (\varphi_1, \varphi_2)$, where φ_2 defines the property that must be satisfied in the set of goal states, and φ_1 defines the property that must hold in all states visited along the path to s_g. We call φ_2 the *target-goal* and φ_1 the *path-goal*. Although φ_1 can be any temporal formula, here we assume it is a propositional formula. Note that if $\varphi_1 = True$, the XR-Goal is equivalent to an R-Goal, i.e., $\varphi = (True, \varphi_2)$. A terminal state is a state with no applicable actions or where the only applicable action is a self-loop action, which can be a goal state or a dead-end state.

One can formally specify which type of policy quality best suits the agent [4]. Considering an R-Goal task, by following a *weak policy*, the agent can eventually reach the target-goal, but since this is not guaranteed, the agent may reach a dead-end state. With a *strong policy*, the agent should always reach the target-goal, despite non-determinism. Finally, with a *strong-cyclic policy*, the agent should always achieve the target-goal, under the *fairness assumption* the execution will eventually exit all existing cycles.

The α-CTL Logic. α-CTL [8] is a branching time temporal logic whose semantics is defined over transition-labeled Kripke structures. The formulae of α-CTL are composed by atomic propositions, logical connectives ($\neg$, $\wedge$ and $\vee$), path quantifiers ($\exists$ and $\forall$), and the following temporal operators: $\odot$ (*next*), $\boxdot$ (*invariantly*), $\diamond$ (*finally*) and $\sqcup$ (*until*) [8]. Intuitively, the α-CTL formula $\forall \odot \varphi$ holds on a state $s \in \mathcal{D}$ if and only if there *exists* an action α, whose execution in s *necessarily* leads to an immediate successor s' satisfying φ. The modality $\odot$ represents the set of α-*successors* of s, for *some* particular action α, and the quantifier $\forall$ requires that every state in this set can satisfy φ. The semantics of other modal operators are derived from the semantics of the operators $\exists\odot\varphi$ and $\forall\odot\varphi$, using minimum and maximum fixed-point operations. The formal definition of the α-CTL's semantics is based on the concept of *preimage*.

Definition 1 (Weak Preimage in α-CTL with transitions). *Let $X \subseteq \mathcal{S}$ be a set of states. The weak preimage of X is the set $\{(s,a) : s \in \mathcal{S}, a \in \mathcal{A}$ and $\mathcal{T}(s,a) \cap X \neq \varnothing\}$.*

Definition 2 (Strong Preimage in α-CTL with transitions). *Let $X \subseteq \mathcal{S}$ be a set of states. The strong preimage of X is the set $\{(s,a) : s \in \mathcal{S}, a \in \mathcal{A}$ and $\varnothing \nsubseteq \mathcal{T}(s,a) \subseteq X\}$.*

Given a FOND planning problem $\mathcal{P} = \langle \mathcal{D}, s_0, \varphi \rangle$, where φ is an XR-Goal, we can express in α-CTL a new complex goal, called $\bar{\varphi}$, that includes the desired policy quality as follows: $\bar{\varphi} = \exists(\varphi_1 \sqcup \varphi_2)$ to specify a ***weak policy***; $\bar{\varphi} = \forall(\varphi_1 \sqcup \varphi_2)$ to specify a ***strong policy***; and $\bar{\varphi} = \forall \boxdot \exists(\varphi_1 \sqcup \varphi_2)$ for ***strong-cyclic policy***.

3 PACTL-XR PLANNER

Given a planning problem $\mathcal{P} = \langle \mathcal{D}, s_0, \bar{\varphi} \rangle$, PACTL-XR has algorithms for each of the policy types and XR-Goal.

Strong policy for XR-Goal. To find a strong policy we first compute the set of states G_1 that satisfies φ_1 and the set of states G_2 that satisfies φ_2 (target-goal state set). Then, the submodel defined by G_2 is expanded through strong preimage (Def. 2) operations until reaching a fixed point. At each preimage step, state-action pairs that are not in G_1 are pruned. If the expanded submodel contains the initial state s_0, then a strong policy can be extracted from it.

Weak policy for XR-Goal. Similar to Strong Policy but it uses the weak preimage operation (Def. 1) to compute a weak submodel.

Strong-cyclic policy for XR-Goal. This algorithm starts by computing the sets G_1 and G_2. Then, it computes a submodel by alternating between two Phases: **1)** Generation of a weak submodel M_1 by the preimage computation from $G2$ that intersects $G1$; and **2)** A new model M_2 is computed as a set of state-action pairs that necessarily reaches some state in M_1 (the removed states are dead-ends). To synthesize M_2, the algorithm applies strong preimage operations until reaching a fixed point. These two phases alternate until the algorithm reaches a fixed point. It is important to highlight that once Phase 2 is computed, we return to Phase 1 to compute the preimage ignoring the eliminated state-action pairs.

Strong-cyclic policy for XR-Goal: a simplified version. Computing a strong-cyclic policy for an XR-Goal is the most costy algorithm since it involves two phases. Thus, if the path goal φ_1 is a condition representing knowledge about dead-ends, the strong-cyclic policy algorithm can be simplified by removing state-action pairs that leads to states that do not satisfy φ_1 in the first Phase, eliminating the need for the second Phase of the algorithm.

The complexity of computing a fixed point is $O(|\mathcal{S}|)$. Note that the weak, strong, and simplified strong-cyclic policies have one fixed point, while the strong-cyclic policy requires at least three.

PACTL-XR-Sym: symbolic planning. We implemented a symbolic version of the PACTL-XR planner using Binary Decision Diagrams (BDDs) [2]. This approach consists of representing sets of states and actions using Quantified Boolean Formulae (QBF) logic [3] (formulae $\xi(.)$) which allows preimage operations to be performed on sets of states rather than on individual state. The symbol $\exists$ is a quantifier of QBF logic, where $\exists x\varphi \equiv \varphi[0/x] \vee \varphi[1/x]$.

In our symbolic version, a state s is represented by the formula: $\xi(s) = \bigwedge_{p \in \mathcal{L}(s)} p \ \wedge \ \bigwedge_{q \in \mathbb{P} \setminus \mathcal{L}(s)} \neg q$. A set of states $X \subseteq \mathcal{S}$ can be represented by a disjunction of every state $s \in X$. A precondition of an action a, denoted by $prec(a)$, is given by a conjunction of $\forall p \in prec(a)$. The effects representing the changes in s after the execution of a, are respectively represented in QBF as following: $\xi(\mathit{eff}(a, e_i)) = \left(\bigwedge_{q \in \mathit{eff}^+(a,e_i)} q \quad \wedge \bigwedge_{r \in \mathit{eff}^-(a,e_i)} \neg r \right), e_i \in \mathit{eff}(a)$.

Definition 3 (Symbolic Weak Regression). *The weak regression of a set of states X by a non-deterministic action a, computes the set of states from which a state in X is reached by some effect of a, i.e.: $\xi(prec(a)) \wedge \left(\bigvee_{e_i \in \{1\ldots n\}} \exists changes(a, e_i).(\xi(\mathit{eff}(a, e_i)) \wedge \xi(X)) \right)$.*

Definition 4 (Symbolic Strong Regression). *The strong regression of a set of states X computes the states from which all the non-deterministic effects of a reach a state in X, i.e.: $\xi(prec(a)) \wedge \left(\bigwedge_{e_i \in \{1\ldots n\}} \exists changes(a, e_i).(\xi(\mathit{eff}(a, e_i)) \wedge \xi(X)) \right)$.*

4 EMPIRICAL ANALYSIS

Domain	total score				% Solve			
	PRP	PR2	PACTL-XR-Sym		PRP	PR2	PACTL-XR-Sym	
	R-Goal	R-Goal	R-Goal	XR-Goal	R-Goal	R-Goal	R-Goal	XR-Goal
Gripper	10.51	15	13.58	14.57	100	100	100	100
Triangle	10	10	9.92	9.99	100	100	100	100
Island	16.69	30	27.46	28.97	66.66	100	100	100
Travel	10	10	9.93	9.99	100	100	100	100

Table 1: Statistics to find strong-cyclic policies; higher total scores means better performance; %solve indicates the planner's coverage.

We run experiments to find strong-cyclic policies for (i) R-Goal tasks and (ii) XR-Goal tasks, with φ_1 formula containing knowledge about dead-ends. We compare the results with PRP [7] and PR2 [6] only for R-Goal tasks, since they cannot deal with XR-Goal tasks, even when knowledge about dead-ends is given. We analized four benchmark planning domains: Gripper (IPC), Triangle-Tire (IPC), Island [5] and Traveling (a variation of Triangle-Tire).

Table 1 shows the total score and the percentage of instances solved (normalized coverage) by the planners for each analyzed domain. The task-score is 1 if a task is solved within 1 second, 0 if unsolved, and $1 - \log(T)/\log(\text{MAXTIME})$ for $1 \leq time \leq \text{MAXTIME}$ (we set MAXTIME to 1800 seconds). The task-score rewards planners for being faster. The total score is the sum of all task-scores in a domain. With the exception of PRP, which couldn't solve all the Island domain tasks, all planners solved all the tasks. PR2 achieved the best total score results. For the Gripper and Island domains, PACTL-XR-Sym achieved a score close to PR2 (either for R-Goal and XR-Goal tasks) and outperformed PRP. In the TriangleTire and Travelling domains, PACTL-XR-Sym scored similarly to other planners. PACTL-XR-Sym performed better when solving XR-Goal tasks in the same domain; this is due to the simplified version of the algorithm. Finally, our experiments show that PACTL-XR-Sym delivers competitive results when compared to the other planners.

ACKNOWLEDGMENTS

We thank CNPq (148935/2021-4) and FAPESP (19/07665-4) (Brazil) for the financial support.

REFERENCES

[1] Viviane Bonadia, Leliane Barros, , and Maria Viviane Menezes. 2019. Symbolic Planning for Strong-Cyclic Policies. In *2019 8th Brazilian Conference on Intelligent Systems (BRACIS)*. IEEE, 168–173.

[2] Randal E. Bryant. 1992. Symbolic Boolean Manipulation with Ordered Binary-decision Diagrams. *ACM Comput. Surv.* 24 (1992), 293–318.

[3] Hans Kleine Büning and Uwe Bubeck. 2009. Theory of Quantified Boolean Formulas. In *Handbook of Satisfiability*. IOS Press, 735–760.

[4] A. Cimatti, M. Pistore, M. Roveri, and P. Traverso. 2003. Weak, Strong, and Strong Cyclic Planning via Symbolic Model Checking. *Artificial Intelligence Journal* 147 (2003), 35–84.

[5] Tomas Geffner and Hector Geffner. 2018. Compact policies for fully observable non-deterministic planning as SAT. In *Proceedings of the International Conference on Automated Planning and Scheduling*, Vol. 28. 88–96.

[6] Christian Muise, Sheila A McIlraith, and J Christopher Beck. 2024. PRP Rebooted: Advancing the State of the Art in FOND Planning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 20212–20221.

[7] Christian J Muise, Sheila A McIlraith, and J Christopher Beck. 2012. Improved Non-Deterministic Planning by Exploiting State Relevance.. In *ICAPS*.

[8] Silvio Lago Pereira and Leliane Nunes Barros. 2008. A logic-based agent that plans for extended reachability goals. *AAMAS Journal* 16 (2008), 327–344.

[9] Viviane Bonadia dos Santos, Leliane Nunes de Barros, Silvio do Lago Pereira, and Maria Viviane de Menezes. 2022. Symbolic fond planning for temporally extended goals. In *Workshop on Knowledge Engineering for Planning and Scheduling*.