Perspectives on plans with loops in stochastic domains

Paper XXX

ABSTRACT

A plan with control structures such as branches and loops can serve as a general solution when there is inherent uncertainty about the initial parameters of a planning problem. The correctness of such plans, however, is non-trivial to define. Recently, it has been suggested that we review correctness epistemically: if the same structure terminates after enabling a goal in every world (that is, plan instance), then we are done. This is quite different from being an epistemic plan, however; actions advised across worlds may differ, so what is the robot to do? That subtlety notwithstanding, the applicability of these formal theories to many real-world domains is suspect to serious criticism: there is little or no mention of effector and sensor noise, and the underlying theory of knowledge has no apparatus to incorporate probabilistic information.

In this work, we revisit loopy plans and its epistemic perspective thereof in a stochastic setting. In particular, we logically characterize the interactions between plan correctness, noise and degrees of belief in a rich theory of action that allows for discrete and continuous probability distributions.

CCS Concepts

•Computing methodologies → Reasoning about belief and knowledge; Planning with abstraction and generalization; •Theory of computation → Modal and temporal logics;

Keywords

Reasoning about Knowledge and Action, Generalized Planning, Degrees of Belief, The Situation Calculus

1. INTRODUCTION

Much of the work in symbolic AI is concerned with the actions taken by a putative agent, such a robot or a web service, towards a goal. While automated planning has turned into a major endeavor [18], most efforts deal with *sequential planning*, that of generating a sequence of actions that enable the goal. Sensing actions, however, necessitate *conditional plans* that branch on observations. In more contrived situations, where there is inherent uncertainty about the initial parameters of a planning problem, an important body of work on *iterative planning* has emerged, often represented by loopy plans, which are powerful memoryless controllers that can express unbounded iteration.

Consider the problem of chopping a tree of thickness $n \in \mathbb{N}$. If a *chop* action can be expected to cut the thickness by one, Figure 1 is a sequential plan to bring down the tree. However, if the agent

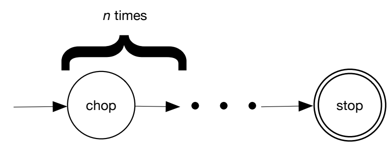

Figure 1: controller for chopping a tree of thickness $n \in \mathbb{N}$

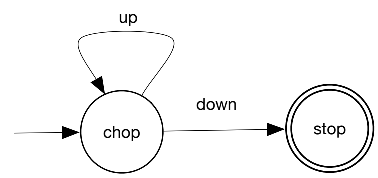

Figure 2: controller for chopping a tree of arbitrary thickness

is equipped with a sensor to tell it if the tree is still standing, Figure 2 is an elegant and compact controller to get the job done. The attractiveness of this controller (to a theoretician) is this: it works for arbitrary n! Synthesizing these plans is an ongoing effort with many successes [8, 42, 29, 20], complemented by a long and distinguished history [5, 44, 31, 16].

Such appealing plan structures raise the obvious question: why do we think they are correct? Levesque [29] suggested that we look at this *epistemically*: let the ignorance about initial plan parameters (*e.g.* tree thickness) be seen as possible worlds. If we can then argue that at each world, the plan terminates (reaches the distinguished state labeled *stop*) by making the goal true, we are done.

Interestingly, observe that this epistemic perspective is really a proxy for multiple problem instances. It is *not* an epistemic plan in the sense that the appropriate course of action advised by the controller in the possible worlds may differ; so what is the agent to do? The property of an agent *knowing how* to execute a plan, which roughly amounts to all worlds either agreeing on termination or which action to perform at the next step, needs a characterization of *ability* [46, 27]. In a sense, Levesque's account is a *world-based* specification, whereas demanding the agent to know how to execute plans is understandably a *belief-based* one. An analysis of the interactions between these specifications is, needless to say, critical to the epistemic and algorithmic foundations of iterative planning [34, 14, 40].

That subtlety notwithstanding, the applicability of these formal theories of action to many real-world robotics domains is suspect to serious criticism: there is little or no mention of effector and sensor noise, and the underlying theory of knowledge has no apparatus to incorporate probabilistic information. Noise models in most machine learning and physical robotics applications, moreover, is

Appears in: *Proceedings of the 16th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2017), S. Das, E. Durfee, K. Larson, M. Winikoff (eds.), May 8–12, 2017, São Paulo, Brazil.*

continuous [45].

In this work, we revisit loopy plans and its epistemic perspective thereof in a stochastic setting. We will be interested in structures like Figure 2, but we will understand that the action advised by the controller may lead to (possibly uncountably) many successors, and the observations obtained by the agent may be noisy. Beginning with Levesque's account on the execution semantics of loopy plans in noise-free domains, we will motivate a new execution semantics in the stochastic setting. This semantics is embedded in a rich theory of action for degrees of belief in the situation calculus [1, 3] that admits both discrete and continuous probability distributions. For our purposes, the theory is carefully extended to deal with the intricacies of reactive plan execution, and with that, we will be able to logically characterize various flavors of correctness that emerge in the presence of nondeterminism [2]. As mentioned above, however, Levesque's account is essentially a a world-based specification and uses knowledge as a proxy for multiple problem instances. So, next, we develop a belief-based specification for loopy plan correctness. Here, we will not appeal to an explicit characterization of ability (something we leave for the future), but instead look at plan execution for belief states. Most significantly, we will show that this account correctly handles changes to the state of the world as a result of noisy actions, as well as the changes to the beliefs of an agent after noisy sensing.

In sum, we characterize, for the first time, the interactions between loopy plan correctness, noise and degrees of belief in a rich theory of action allowing for discrete and continuous probability distributions.

Organization is as follows. We first recap the non-probabilistic epistemic situation calculus for noise-free actions and the execution semantics for loopy plans in this setting. We then introduce an existing account for degrees of belief in the situation calculus, and explore results in the order described above.

We reiterate that the technical thrust of this paper is purely limited to formal characterizations. At no point will we concern ourselves with algorithmic ideas or plan heuristics. The penultimate section on related work provides a number of references on computational methodologies for generalized planning.

2. A THEORY OF ACTION

The language $\mathcal{L}$ of the situation calculus [32] is a many-sorted dialect of predicate calculus, with sorts for *actions*, *situations* and *objects* (for everything else, and includes the set of reals $\mathbb{R}$ as a subsort). A situation represents a world history as a sequence of actions. A set of initial situations correspond to the ways the world might be initially. Successor situations are the result of doing actions, where the term $do(a, s)$ denotes the unique situation obtained on doing a in s. The term $do(\alpha, s)$, where α is the sequence $[a_1, \ldots, a_n]$ abbreviates $do(a_n, do(\ldots, do(a_1, s) \ldots))$. Initial situations are defined as those without a predecessor:

$$Init(s) \doteq \neg\exists a, s'.\ s = do(a, s').$$

We let the constant S_0 denote the actual initial situation, and here, we use the variable ι to range over initial situations only.

The picture that emerges from the above is a set of trees, each rooted at an initial situation and whose edges are actions. In general, we want the values of predicate and functions to vary from situation to situation. For this purpose, $\mathcal{L}$ includes *fluents* whose last argument is always a situation.[1]

[1]We sometimes suppress the situation term in a formula ϕ, or use a special variable *now*, with the understanding that $\phi(s)$ restores the situation term.

Basic action theory

Following [39], we model dynamic domains in $\mathcal{L}$ by means of a *basic action theory* $\mathcal{D}$, which consists of,[2]

1. axioms $\mathcal{D}_0$ that describe what is true in the initial states, including S_0;
2. precondition and successor state axioms that describe the conditions under which actions are executable and the changes to fluents on executing actions respectively;
3. domain-independent *foundational* axioms, the details of which need not concern us here. See [39].

We obtain many advantages by axiomatizing this way, including a simple solution to the frame problem [39]. An agent reasons about actions by means of the entailments of $\mathcal{D}$.

Example 1. *For the tree chop problem in Figure 1, the basic action theory includes:*

$Poss(chop, s) \equiv thickness(s) > 0.$

$$thickness(do(a, s)) = u \equiv (a = chop \land thickness(s) = u + 1) \lor (a \neq chop \land thickness(s) = u).$$

The initial theory is the following sentence: $thickness(S_0) = n$.

Reasoning about knowledge

The situation calculus was first extended for knowledge by Moore [33], and later refined to incorporate Reiter's solution to the frame problem in [41]. Mirroring modal logic [11], the idea is allow possible states of affairs relative to a situation using a distinguished predicate K. The agent's ignorance initially can be stated for the tree chop problem as follows:

$$K(\iota, S_0) \equiv 1 \leq thickness(\iota) \leq 10 \tag{1}$$

which can be read as saying that any world where the tree's thickness is a number between 1 and 10 is epistemically possible when the agent is at the (real) world S_0. (Unlike standard modal logic, worlds are reified in the language and the order of accessibility is reversed.) As usual, knowledge is defined as truth at accessible worlds:

$$Know(\phi, s) \doteq \forall s'.\ K(s', s) \supset \phi[s'].$$

The crux of the extension is a successor state axiom for K that determines how the accessibility relation changes after actions:

$$K(s', do(a, s)) \equiv \exists s''[K(s'', s) \land s' = do(a, s'') \land Poss(a, s'') \land (SF(a, s'') = SF(a, s))].$$

Here, SF is a distinguished function that, for example, in the tree chop problem works as follows:

$$SF(a, s) = \begin{cases} down & a = chop \land thickness(s) = 0 \\ up & a = chop \land thickness(s) \neq 0 \\ 0 & a \neq chop \end{cases} \tag{2}$$

So if $K(s'', s)$ and $a \neq chop$, $K(do(a, s''), do(a, s))$ holds as well, provided a is executable at s''. However, if *chop* is executed, s'' is expected to conform with s on the status of the tree being down. This way, an account of knowledge expansion based on sensing is realized.[3]

[2]As usual, free variables in any of these axioms should be understood as universally quantified from the outside.

[3]Like in modal logic [15], constraints on K correspond to appropriate properties for *Know* in the truth theory [41]. For example,

Plans with loops

We will be interested in computable program-like plans that are simple to execute requiring no internal memory. In particular, we appeal to finite state controllers, which are fairly common in the literature [7, 20, 21]. Other choices are possible [27, 30], of course.

Definition 2. *Suppose $\mathcal{A}$ is a finite set of (parameterless) action terms, and O a finite set of objects, denoting observations. A finite memoryless plan X is a tuple $\langle Q, Q_0, Q_F, \gamma, \delta\rangle$:*

- *Q is a finite set of control states;*
- *$Q_0 \in Q$ is the initial control state, and $Q_F \in Q$ the final one;*
- *$\gamma \in [Q^- \to \mathcal{A}]$ is a labeling function for $Q^- = Q - \{Q_F\}$;*
- *$\delta \in [Q^- \times O \to Q]$ is a transition function.*

We assume terms such as $chop \in \mathcal{A}$ and $\{up, down\} \in O$.

Informally, we may think of applying these plans in an environment as follows: starting from Q_0 that advises the action $\gamma(Q_0)$, the environment executes that action and changes externally to return $o \in O$. Then, internally, we reach the control state $\delta(Q_0, o)$ and so on, until Q_F. To reason about these plans in an environment enabled in the situation calculus, we need to encode the plan structure as a $\mathcal{L}$-sentence, and axiomatize the execution semantics over situations, for which we follow [21] and define:

Definition 3. *Let Σ be the union of the following:*

1. *domain closure axiom for control states:*
$$(\forall q)\ \{q = Q_0 \lor q = Q_1 \lor \ldots \lor q = Q_n \lor q = Q_F\};$$
2. *unique name axioms for control states: $Q_i \neq Q_j$ for $i \neq j$;*
3. *action association axioms $\forall Q \in Q^-$ of the form $\gamma(Q) = a$;*
4. *transition axioms $\forall Q \in Q^-$ of the form $\delta(Q, o) = Q'$.*

Definition 4. *We use $T^*(q, s, q', s')$ as abbreviation for $\forall T[\ldots \supset T(q, s, q', s')]$, where the ellipsis is the conjunction of the universal closure of:*

- *$T(q, s, q, s)$*
- *$T(q, s, q'', s'') \land T(q'', s'', q', s') \supset T(q, s, q', s')$*
- *$\gamma(q) = a \land Poss(a, s) \land SF(a, s) = o \land \delta(q, o) = q' \supset T(q, s, q', do(a, s))$.*

In English: T^* is the reflexive transitive closure of the one-step transitions in the plan.

We are now prepared to reason about plan correctness:

Definition 5. *For any goal formula $\phi \in \mathcal{L}$, basic action theory $\mathcal{D}$, plan X and its encoding Σ, we say X is correct for ϕ iff*
$$\mathcal{D} \cup \Sigma \models \forall s.\ K(s, S_0) \supset \exists s'[T^*(Q_0, s, Q_F, s') \land \phi(s')].$$

Example 6. *Let $\mathcal{D}_{dyn}$ denote the axioms from Example 1, and then it is easy to see that $\mathcal{D}_{dyn} \cup \{(1), (2)\} \cup \Sigma$, where Σ represents the encoding of Figure 2, is correct for the goal $thickness = 0$.*

transitivity $\forall s, s', s'' K(s, s') \land K(s', s'') \supset K(s, s'')$ enables positive introspection for *Know*. We assume, as is standard, that *Know* has the properties of **S5**, by consequence of a stipulation that K is an equivalence relation.

Basically, in every initial possible world, that is, where $thickness \in \{1, \ldots, 10\}$, the plan brings the tree down. But, of course, the number of chop actions executed in each world differs, which reiterates a consideration from Section 1 on the formulation not being an epistemic plan. Put differently, the example can be entirely captured without the K fluent using the following correctness criteria:
$$\mathcal{D} \cup \Sigma \models \exists s[T^*(Q_0, S_0, q, s) \land \phi(s)] \qquad (\dagger)$$
with the understanding that $\mathcal{D}$ models ignorance using connectives.

Example 7. *Consider the following analogue of* (1)*:*
$$thickness(S_0) = 1 \lor \ldots \lor thickness(S_0) = 10. \qquad (3)$$
Then, $\mathcal{D}_{dyn} \cup \{(3), (2)\} \cup \Sigma$, where Σ is as before, is correct for the goal $thickness = 0$ in the sense of $(\dagger)$.

Basically, in every model of the action theory, *thickness* takes a value $\in \{1, \ldots, 10\}$, but the plan brings the tree down regardless.

The correspondence between disjunctions and knowledge is straightforward [28, 39], but the above definitions explicate this correspondence for the correctness of loopy plans. In fact, Definition 5 is adopted from Levesque's account on the correctness of *robot programs* defined epistemically [29], but where robot programs are dropped for memoryless plans, whose correctness is often not characterized using K [21]. For our purposes, we continue to work with the epistemic formulation.

3. STOCHASTIC ENVIRONMENTS

Our objective now is to generalize the above well-understood framework on knowledge and loopy plans to a stochastic setting. We will attempt to lift existing accounts as much as possible, although the subtlety of a plan over noisy actions and its interaction with an external environment will require us to revisit some basic notions, and carefully extend them.

Let us review the setting informally. We begin with the understanding that the agent is ignorant about some aspects of the world; this uncertainty may or may not be quantifiable, and so the agent is assumed to have both categorical and probabilistic knowledge. (An important feature of building our account on the prior work of [1, 3] is precisely the amenability for categorical and probabilistic assertions to co-exist in the same logical language, which means that the modeler is not required to provide a unique prior, nor place strict assumptions on how variables are affected after actions.) To make this concrete, imagine a robot attempting to grasp an object at an unknown distance d from its manipulator, as seen in Figure 3. By using its (noisy) light sensor via the action *getd*, the robot is able to get an estimate about the value of d. However, due to poor lighting conditions, it is unable to determine if the object is a ball or a cube. Finally, the robot may have an action to move its arm towards the object, but this movement is noisy: intending to move by a unit may cause the arm to actually move by 1.5 units or .9 units. The purpose of a plan, of course, is to achieve the goal wrt the background knowledge and noise. As far as the plan structure goes, we will not want it to be any different than Definition 2; however, we will need to revisit Definition 4 to internalize the noisy aspects of acting.

To prepare for that, we will begin with the logical language. We recap the bare essentials of an account for reasoning about degrees of belief in the situation calculus [1] that admits both discrete and continuous probability distributions [3, 4]. The account is based on 3 distinguished fluents: p, l and alt. The p fluent here is a numeric

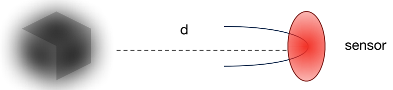

Figure 3: grasping an object of unknown type

analogue to K in that $p(s', s)$ denotes the weight (or density) accorded to s' when the agent is at s. Initial constraints about what is known before actions can be provided as usual:

$$p(\iota, S_0) = \begin{cases} 1 & \text{if } (5 \leq d(\iota) \leq 6) \wedge (type(\iota) = cube \vee type(\iota) = ball) \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

says that any initial situation where d is in the real-interval $[5, 6]$ and where the object's type is cube or ball is accorded a density of 1, and 0 otherwise. In probabilistic terms, d is a uniform random variable on $[5, 6]$, whereas ignorance about the object's type is specified with a logical connective. (So, a constraint without a unique prior.) Often, we write $K(s' s) \doteq p(s', s) > 0$.

The l fluent is used to express the likelihoods of outcomes. For example, if $getd(z)$ denotes that the value z was observed on executing the sensing action, then

$$l(getd(z), s) = \mathcal{N}(z; d(s), .25) \quad (5)$$

says that observed values are normally distributed around true values, with a variance of .25.

To handle noisy actions, we will need to allow action types to have a second set of arguments: the idea is that if $mv(x)$ represents a movement by the robot's manipulator by x units towards the object in Figure 3, we now assume a new action type $mv(x, y)$ in that x is taken as the intended argument, and y is taken as the actual outcome. These action types are retrofitted in successor state axioms in an obvious fashion; for example, suppose

$$d(do(a, s)) = u \equiv (a = mv(x) \wedge u = d(s) - x) \vee (a \neq mv(x) \wedge u = d(s)) \quad (6)$$

is the dynamic law in the noise-free case, then the one allowing for intended and actual outcomes would work as follows:

$$\begin{aligned} d(do(a, s)) = u \equiv\ & (a = mv(x, y) \wedge u = d(s) - y) \vee \\ & (a \neq mv(x, y) \wedge u = d(s)) \end{aligned} \quad (7)$$

Of course, the agent is assumed to not control y, and that is what makes this simple extension a device for enabling nondeterminism. The alt fluent allows the modeler to determine the space of actual outcomes. For example,

$$alt(mv(x, y), a', z) \equiv a' = mv(x, z) \wedge |z - x| \leq 5 \quad (8)$$

says that $mv(x, y)$ is alt-related to $mv(x, z)$ iff the actual value z and the intended value are at most 5 units apart. Likelihoods for noisy actions can be defined in much the same way as noisy sensors:

$$l(mv(x, y), s) = \mathcal{N}(y; x, 1) \quad (9)$$

says that the actual value is normally distributed around the intended value.

Likelihoods and alt-axioms determine the probability of successors, enabled by the following successor state axiom for p:

$$\begin{aligned} & p(s', do(a, s)) = u \ \equiv \\ & \quad \exists a', z, s'' \ [alt(a, a', z) \wedge s' = do(a', s'') \wedge Poss(a', s'') \wedge \\ & \qquad\qquad u = p(s'', s) \times l(a', s'')] \\ & \vee \neg\exists a', z, s'' \ [alt(a, a', z) \wedge s' = do(a', s'') \wedge Poss(a', s'') \wedge u = 0] \end{aligned}$$

That is, $p(do(a', s''), do(a, s))$ is the p-value of s' relative to s multiplied by the likelihood of a', which is alt-related to a. Observe that $p(s', s)$ is 0 if s' and s do not share the same history of actions (but allowing for alternatives, of course).

Putting all this together, the degree of belief in ϕ at s is defined as the weight of worlds where ϕ is true:

$$Bel(\phi, s) \doteq \sum_{\{s' : \phi(s')\}} p(s', s) \Big/ \sum_{s'} p(s', s)$$

Bel can be extended to handle both discrete and continuous probability distributions by stipulating that there are only n nullary real-valued functional fluents $f_1, \ldots, f_n$ in $\mathcal{L}$ and that there is precisely one initial situation for each possible assignment for these terms:

$$[\forall \vec{x} \exists \iota (\bigwedge f_i(\iota) = x_i)] \wedge [\forall \iota, \iota' \ (\bigwedge f_i(\iota) = f_i(\iota') \supset \iota = \iota')] \quad (10)$$

We refer readers to [3] to details, and do not provide the general expansion for Bel here. The main takeaway is that we will not need to make any sort of discrete approximations to axioms like (9). The definition for Bel is also shown to exhibit Bayesian conditioning in [3]. Often, we write $Know(\phi, s) \doteq Bel(\phi, s) = 1$.

Plans with loops revisited

Before we consider the execution semantics of plans in stochastic environments, we will need to address two logical issues.

Runtime sensing outcomes

The internal execution of a plan is determined by sensing results, enabled by the special $\mathcal{L}$-function SF in Section 2. The complication of realizing this feature in a stochastic setting is two-fold. One the one hand, the account of Bel above makes no mention of sensing functions, and in this sense, the language is only geared for projection. That is, we can infer the value of $Bel(d \leq 4, do(getd(3), S_0))$ where we explicitly provide the sensor reading, but we cannot easily characterize belief properties for runtime sensor readings. On the other hand, the external feedback (*e.g.* number reported on a light sensor) is only an estimate about the true property because of the noisy sensor. This is quite unlike Section 2 where we could axiomatize the behavior of SF categorically based on the current situation.

One simple way to address this in $\mathcal{L}$ is to think of a *runtime sensing outcome function* π that takes as input n real numbers, and a sequence of action terms to return an object, that is, $\pi(x_1, \ldots, x_n, a_1, \ldots, a_k) \in O$. To see why this helps, let us reconsider SF, whose value at some ground situation term can be defined using π as follows:

$$\begin{aligned} & SF(a, do(a_1 \cdots a_k, \iota)) = u \doteq \exists x_1, \ldots, x_n. \\ & \quad [\bigwedge f_i(\iota) = x_i \wedge \pi(x_1, \ldots, x_n, a_1, \ldots, a_k, a) = u]. \end{aligned}$$

(The property of having a unique initial situation is made possible because of (10).) The idea is that we may use SF in K's successor state axiom as usual, but instead of axioms like (2), we would now include axioms for the mappings of π. The disadvantage is that axioms like (2) are intuitive for a modeler, whereas π is abstract, and so nothing is gained (and some is lost) in domains with exact sensing. In domains with noisy sensing, however, since π can be arbitrary,[4] the modeler can set π to the observed runtime values.

[4]Indeed, with noisy sensors, it is possible that even if an action does not physically change any property, the sensing function may return different values on repeated execution. For example, if $d = 4$, the light sensor in Figure 3 might provide a reading of 4.1, but then a second reading immediately afterwards might yield 3.97.

With this machinery, we can now introduce parameterless sensing actions like *getd* whose likelihood is now defined to mimic (5) as follows:

$$l(getd, s) = \mathcal{N}(z; d(s), 1) \wedge z = SF(getd, s). \tag{11}$$

Analogously, to model that a *chop* action returns $\{up, down\}$ but in a noisy manner, we define:

$$l(chop, s) = \begin{cases} .9 & \text{if } SF(chop, s) = up \wedge thickness(s) \neq 0 \\ .1 & \text{if } SF(chop, s) = up \wedge thickness(s) = 0 \\ .5 & \text{if } SF(chop, s) = down \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

This says that if we read that the tree is up, it is highly likely that the tree is actually up in the real world. However, if we read that the tree is down, with equal probability, the tree is up or down.

Actions for control states

We will want to use plan structures like Figure 2, with the understanding that the actions advised are noisy. Fortunately, this is easily done. Let the set $\mathcal{A}$ considered in plan structures be parameterless action terms as before, and we will interpret them as ground $\mathcal{L}$-actions. For example, suppose $mv \in \mathcal{A}$, then we take mv as an abbreviation for $mv(x, y)$ with some fixed x, say 1, and an arbitrary y, say 0.[5] Similarly, suppose $chop(x, y)$ is a version of the action used in Example 1 in that the agent intends to do x chops, but y chops actually happen. Then let $chop \in \mathcal{A}$ mean the $\mathcal{L}$-action $chop(1, y)$, and say, let $chopchop \in \mathcal{A}$ mean $chop(2, y)$, and so on.

Since we understand the symbols in $\mathcal{A}$ as abbreviations for standard $\mathcal{L}$-actions, noisy or otherwise, no other change is necessary to an action theory. Likelihood axioms and *alt*-axioms are specified as before.

Plan execution

We are now prepared to define the execution semantics of plans in stochastic domains.

Definition 8. *We use* $U^*(q, s, q', s')$ *as abbreviation for* $\forall U[\ldots \supset U(q, s, q', s')]$, *where the ellipsis is the conjunction of the universal closure of:*

- $U(q, s, q, s)$
- $U(q, s, q'', s'') \wedge U(q'', s'', q', s') \supset U(q, s, q', s')$
- $\gamma(q) = a \wedge \exists b, z\ (alt(a, b, z) \wedge Poss(b, s) \wedge SF(b, s) = o \wedge \delta(q, o) = q') \supset U(q, s, q', do(b, s))$.

The main new ingredient here over T^*, of course, is how control states transition from a situation to a successor. The reflexive transitive closure of U basically says that if the controller advises a and b is any action that is *alt*-related to a, we consider the transition wrt the executability and the sensing outcome of the action b. The idea, then, is allow U^* to capture the least set that accounts for all the successors of a situation where a noisy action is performed.

Given the many successors after actions, none of which are in the agent's control, our criteria for correctness in Definition 5 seems defective as it only captures a single goal-satisfying path. Indeed, a single path of this sort is referred to as a *weak plan* in [12]. In recent work [2], a number of other criteria are identified, and it is suggested that we call a plan θ-adequate if the criteria θ, defined in terms of plan execution and goal satisfaction, is satisfied. Following that suggestion, we define:

Definition 9. *For any goal formula* $\phi \in \mathcal{L}$, *basic action theory* $\mathcal{D}$, *plan* $\mathcal{X}$ *and its encoding* Σ, *we say* $\mathcal{X}$ *is* θ*-adequate for* ϕ *iff*

$$\mathcal{D} \cup \Sigma \models \forall s.\ K(s, S_0) \supset \ldots$$

where

- *if* $\theta =$ ONE, *then the ellipsis stands for*
$$\exists s'[U^*(Q_0, s, Q_F, s') \wedge \phi(s')] \tag{13}$$
- *if* $\theta =$ PC, *then it stands for*
$$\forall s'\ [U^*(Q_0, s, Q_F, s') \supset \phi(s')] \tag{14}$$
- *if* $\theta =$ TER, *then it stands for*
$$\forall s'\ [U^*(Q_0, s, q, s') \supset \exists s''\ (U^*(q, s', Q_F, s''))] \tag{15}$$

(There are other criteria identified in [2], but these will suffice for our purposes.) Here, ONE, like Definition 5, characterizes a single terminating execution that also satisfies the goal; PC says that every terminating execution satisfies the goal state and finally TER says that from every partial execution, there will be one terminating execution. (The goal is not mentioned in TER, but we will often use it in conjunction with PC.)

We have two objectives for formalizing the criteria from [2]: first, we would like to explicate how precisely the logical account developed in this paper conforms to the abstract model in [2], and second, we would like to further argue that our plan execution semantics U^* is downward compatible with T^*.

It is shown in [2] in that ONE and {PC,TER} are essentially equivalent in noise-free domains.[6] Formally, let us call a basic action theory $\mathcal{D}$ *noise-free* iff its *alt* and *l*-axioms are trivial: $\forall z(alt(a, a', z) \equiv a = a')$ and $l(a, s) = 1$.

Theorem 10. *Suppose* $\mathcal{D}$ *is any noise-free action theory,* $\mathcal{X}$ *and* Σ *as before, and* ϕ *does not mention* K. *Then*

$$\mathcal{D} \cup \Sigma \models \forall s.\ K(s, S_0) \supset ((13) \equiv (14) \wedge (15)).$$

Proof: The proof uses the same arguments from [2]. The key observation is that in noise-free domains, if $U^*(q, s, q', do(a, s))$ then $\neg\exists b \neq a$ such that $U^*(q, s, q', do(b, s))$. That is, the execution path is unique.

For the proof itself, consider any s such that $K(s, S_0)$. Note that the reverse direction is trivial. If (14) $\wedge$ (15), there is a s^* such that $U^*(Q_0, s, Q_F, s^*)$ and $\phi(s^*)$, and so (13).

For the forward direction, suppose (13). Then, by assumption, there is a $k, a_1, \ldots, a_k$ such that $s^* = do(a_1 \cdots a_k, s)$, $U^*(Q_0, s, Q_F, s^*)$ and $\phi(s^*)$. By the uniqueness of execution paths, $U^*(Q_0, s, Q_F, s')$ holds only for $s' = s^*$, and by assumption $\phi(s')$, and so (14). Now, suppose $U^*(Q_0, s, q', s')$ for some q', s'. But then $U^*(q', s', Q_F, s^*)$ because of the uniqueness of paths. So (15). ∎

As mentioned above, ONE is precisely the correctness criteria of Definition 5 except over a different execution semantics. In fact, for noise-free action theories, the new account is downward compatible with Definition 5:

[5] We will not be too pedantic here and formalize the choosing of y. Moreover, the second argument affects the real world but is irrelevant for the agent's beliefs.

[6] This claim and the one on downward compatibility are only meaningful when we restrict our attention to "objective formulas," that is, formulas not mentioning the p fluent, including abbreviations like *Bel*. Because we define $K(s', s) \doteq p(s', s) > 0$, it suffices to say that K is not mentioned.

Theorem 11. *Suppose $\mathcal{D}$ is a noise-free action theory, $\mathcal{X}$ and Σ are as above, and ϕ is any situation-suppressed formula not mentioning the fluent K. Then, $\mathcal{X}$ is correct for ϕ in the sense of Definition 5 iff $\mathcal{X}$ is* ONE*-adequate for ϕ.*

Proof: U^* differs from T^* is only one aspect, that of *alt*-related actions governing the transition to a successor situation. By assumption, $\forall z(alt(a, a', z) \equiv a = a')$; so T^* and Definition 5 coincide with U^* and ONE. ∎

Example 12. *Let us consider a version of the tree chop problem from Example 1 but with noisy acting but noise-free sensing. So, let the sensing outcome function π work as in* (2)*, let $chop \in \mathcal{A}$ in the plan structure correspond to the $\mathcal{L}$-action $chop(1, 0)$, let $alt(chop(x, y), a', z) \equiv a = chop(x, z)$ and*

$$l(chop(x, y), s) = \begin{cases} .9 & \text{if } x = y \\ .1 & \text{if } y = 0 \\ 0 & \text{otherwise} \end{cases} \quad (16)$$

That is, the chop action is very likely to be successful, but with a small probability it fails in that the tree does not get hit. The successor state axiom from Example 1 be retrofitted for $chop(x, y)$ in the obvious way. Finally, let the initial theory be a p-based axiom for (1)*.*

For the above action theory together with axioms for Figure 2, it follows that:

- *The plan is* ONE*-adequate for the goal $thickness = 0$ because in each initial world, there is clearly one execution path, that of chops always succeeding, which terminates after enabling the goal (owing to exact sensing).*
- *The plan is* TER*-adequate because for any s such that $K(s, S_0)$, for every q', s' such that $U^*(Q_0, s, q', s')$, there is clearly one execution path, that of the appropriate number of chops always succeeding, which terminates after enabling the goal. For example, s' may be the result of only failed chops, or perhaps the result of alternations between failed and successful ones. Whatever may the case, the required number of successful chops are sufficient. Of course, we cannot place any a prior bound on how long these paths may be.*
- *Because of exact sensing, if $U^*(Q_0, s, Q_F, s')$, $thickness(s') = 0$ must hold, and so the plan is also* PC*-adequate.*

A world-based specification is appropriate for reasoning about multiple problem instances, and in this regard, observe that while Definition 9 looks at every non-zero initial world, there is no reason to restrict ourselves to this picture. We could instead look at worlds with weights greater than some fixed κ:

$$\mathcal{D} \cup \Sigma \models \forall s.\ p(s, S_0) > \kappa \supset \forall s'[U^*(Q_0, s, Q_F, s') \supset \phi(s')] \quad (\ddagger)$$

for PC-adequacy (and similarly, for other criteria).

Example 13. *Imagine a tree chop problem with two types of trees, wood ones and metal ones. Suppose our initial theory assumes a few wooden trees, as in* (3)*, but one metal tree. Suppose further the chop actions only work if the type is wood. (A simple modification to Example 1 that incorporates a contextual constraint on the tree's type can axiomatize this sort of domain.) Fortunately, imagine the weight on worlds where the type is metal is only .1, whereas the others have a weight of .9. Then, say, for $\kappa = .5$, the plan from Figure 2 can be shown to be* {PC, TER}*-adequate, but in the categorical case of Definition 9 (in other words: $\kappa = 0$), clearly the plan is not* ONE*-adequate.*

While this looks at individual worlds, we can instead consider the set of all worlds to ascertain a certain degree of belief. More precisely, observe that Definition 9 can be equivalently written as:

$$\mathcal{D} \cup \Sigma \models Know(\forall s'[U^*(Q_0, now, Q_F, s') \supset \phi(s')], S_0)$$

for PC-adequacy (and similarly, for other criteria); so we define a weaker form using:

$$\mathcal{D} \cup \Sigma \models Bel(\forall s'[U^*(Q_0, now, Q_F, s') \supset \phi(s')], S_0) > \kappa \quad (\sharp)$$

which can be read as saying that the sum (or integral) of initial worlds where PC holds is greater than κ.

Example 14. *Imagine a domain like Example 13, but limited to 3 worlds: the first with a wooden tree of thickness 1 and weight .3, the second with a wooden tree of thickness 2 and weight .3, and the third with a metal tree of arbitrary non-zero thickness with weight .4. There is no value for κ such that* {PC, TER}*-adequacy holds in the sense of* ($\ddagger$)*, but for $\kappa = .5$,* {PC, TER}*-adequacy holds in the sense of* ($\sharp$)*.*

This is admittedly a useful application of *Bel* to world-based accounts.

From the perspective of epistemic planning, however, world-based accounts are problematic because the course of action advised by the plan may differ at each world, and so the agent may not know what it should do at the next step. World-based accounts also do not provide any means for reasoning with noisy sensors:

Example 15. *Let us imagine the very same action theory and plan structure as in Example 12, but with* (2) *replaced by* (12)*.*

We note that the plan structure is not PC*-adequate for $thickness = 0$. Imagine an initial situation where $thickness = 2$. After a single chop action, suppose that the plan structure receives down, a false positive from a noisy sensor. The execution then terminates, but clearly $thickness \neq 0$.*

Motivated by these two concerns, we now investigate a belief-based specification.

4. BELIEFS AND LOOPY PLANS

Belief-based planning is very popular in automated planning with incomplete information, see, for example, [6, 37, 35]. The usual approach is to think of planning problems with belief states as their search space: actions change belief states and sensing outcomes (nondeterministically) yield states where beliefs are held or not held [40].

To be concrete, consider the tree chop problem with a sensing function like (2). Here, from a world viewpoint, executing a chop action leads to another world whether the tree is either down or not. From a belief viewpoint, this can be seen as a nondeterministic transition to two possible belief states, one where the tree is down and another where it is not. The idea then is to synthesize a sequence of actions leading to a belief state where the goal holds.

Rather than simply reason about the set of reachable belief states through possible transitions, like the rest of the rest of the paper, we will be interested in the transitions of belief states against reactive memoryless plans like Figure 2. To review this setting informally, consider a noise-free tree chop problem instantiated by (1). While the agent believes that $thickness \in \{1, \ldots, 10\}$, in the real world S_0, the tree has a fixed thickness, say 1. In this case, a chop action will be executed and in each of the possible worlds, including S_0 (because K is assumed to be an equivalence relation), the thickness goes down by 1. At this point, if a noise-free sensor returns *down*,

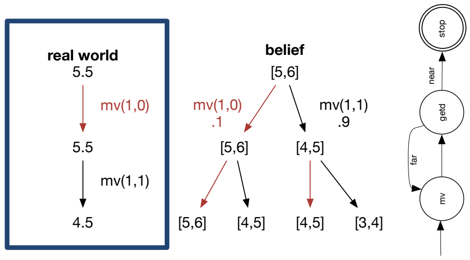

Figure 4: execution path and controller for getting close to the object (with exact sensors)

the agent will come to believe that the tree is down. Implicitly, all the worlds other than S_0 considered epistemically possible initially will be dropped. (But, of course, we will want to allow for noisy sensors too.)

To formalize this intuition, we will want to think of execution paths over belief states, but while referencing the real world to test for sensing outcomes. Before that, however, to define execution paths over belief states, we introduce the following notation. For any ground situation term s, we introduce a new term $\overline{s}$ to be used with formulas in that we write $\phi(\overline{s})$ to mean $\forall s'.\ K(s', s) \supset \phi(s')$. Then, we define:

Definition 16. *We use* $V^*(q, \overline{s}, q', \overline{s'})$ *as abbreviation for* $\forall V[\ldots \supset V(q, \overline{s}, q', \overline{s'})]$, *where the ellipsis is the conjunction of the universal closure of:*

- $V(q, \overline{s}, q, \overline{s})$
- $V(q, \overline{s}, q'', \overline{s''}) \wedge V(q'', \overline{s''}, q', \overline{s'}) \supset V(q, \overline{s}, q', \overline{s'})$
- $\gamma(q) = a \wedge Poss(a, \overline{s}) \wedge SF(a, s) = o \wedge \delta(q, o) = q' \supset V(q, \overline{s}, q', \overline{do(a, s)})$.

The one-step transition is based on the controller advising a, this action being executable at all epistemically accessible worlds, and the sensing function returning o for a at s, which is taken to be the real world. It is worth remarking that $\phi(\overline{do(a, s)})$, by means of p's successor state axiom, would implicitly account for all the *alt*-related actions to a.

With this, we are prepared to reason about correctness:

Definition 17. *For any goal formula* $\phi \in \mathcal{L}$, $\mathcal{D}$, $\mathcal{X}$ *and its encoding* Σ, *we say* $\mathcal{X}$ *is epistemically correct for* ϕ *iff*

$$\mathcal{D} \cup \Sigma \models \forall s.\ K(s, S_0) \supset \exists s'\ [V^*(Q_0, \overline{s}, Q_F, \overline{s'}) \wedge \phi(\overline{s'})]$$

Example 18. *Let us consider the grasping problem from Figure 3 with a noisy effector and a noise-free sensor. (A noise-free sensor will provide intuition on making progress towards the goal in the real world [40].) So let* $\mathcal{D}$ *be an action theory built from* (4), (8), *and the following likelihood axiom for* $mv(x, y)$:

$$l(mv(x, y), s) = \begin{cases} .9 & \textit{if } x = y \\ .1 & \textit{if } y = 0 \\ 0 & \textit{otherwise} \end{cases} \tag{17}$$

That is, with a very high likelihood it succeeds, and otherwise, the arm does not move. The light sensor returns $\{near, far\}$ *as follows:*

$$l(getd, s) = \begin{cases} 1 & SF(getd, s) = near \wedge d \leq 5 \\ 1 & SF(getd, s) = far \wedge d > 5 \\ 0 & \textit{otherwise} \end{cases} \tag{18}$$

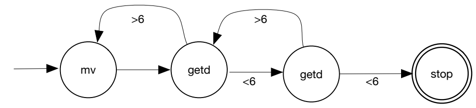

Figure 5: a controller for getting near to the object

It is noise-free and considers $d \leq 5$ *to be near. Finally, suppose our goal is* $Bel(d \leq 5, now) > .8$.

A plan that is epistemically correct for this goal is given in Figure 4. (We only argue for the initial world S_0.) *It also depicts a possible execution path of the plan using* V^*. *Let us imagine* $d(S_0) = 5.5$. *Then the action advised is mv, and suppose the action instantiates as* $mv(1, 0)$. *Then* $d(do(mv(1, 0), S_0)) = 5.5$, *but the agent accords a belief of .9 to* $[4, 5]$ *that corresponds to a successful move, and a belief of .1 to its failure. The noise-free sensor now returns far. The plan advises mv again, and suppose this time, the action instantiates as* $mv(1, 1)$. *The sensor returns near, and incidentally, the degree of belief in* $d \leq 5$ *is .99. Regardless, the plan terminates at* $do(\alpha, S_0)$ *for* $\alpha = [mv(1, 0) \cdot getd \cdot mv(1, 1)]$ *and indeed* $Know(Bel(d \leq 5, now) > .8, do(\alpha, S_0))$.

Example 19. *Let us consider the grasp problem from Figure 3 with noisy sensors and effectors. Suppose the initial theory and alt-axioms are* (4) *and* (8) *as before, but the likelihoods for the effector is given by* (9) *and that for the sensor is given by* (5). *Finally, suppose our goal is* $Bel(d \leq 5, now) > .8$ *as before.*

A plan that is epistemically correct for the goal is given in Figure 5 wrt observed values of 5.5, 4.5 and 3.9. (We only argue for S_0; *also, assume that the set of observations used by the plan maps these real-valued sensor readings to appropriate branches.) The controller advises mv, after which the belief in* $\psi = d \leq 5$ *is* $> .5$. *(The values can be obtained by expanding the definition of Bel, and solving the expression numerically or by sampling.) Suppose now the sensed value is 5.5. The belief in* ψ *drops to* $< .5$. *The controller advises another mv, at which point the belief in* ψ *increases to* $> .5$. *Suppose we observe a reading of 4.5 followed by 3.9. The controller terminates. On termination, the robot knows that the degree of belief in* ψ *is* $> .8$.

Having a plan that is epistemically correct in hand does mean that our world-based specification holds in a noise-free setting.

Theorem 20. *Suppose* $\mathcal{D}$ *is a noise-free action theory,* $\mathcal{X}, \Sigma$ *as above, and* ϕ *is any formula not mentioning K. If* $\mathcal{X}$ *is epistemically correct for* ϕ, *then it is correct for* ϕ *in the sense of Definition 5.*

Proof: Suppose $\mathcal{X}$ is epistemically correct but not correct. Then there is some s such that $K(s, S_0)$ and $\neg\exists s'' T^*(Q_0, s, Q_F, s'') \wedge \phi(s'')$. By assumption, $V^*(Q_0, \overline{s}, Q_F, \overline{s'}) \wedge \phi(\overline{s'})$ for some s'. Thinking of $\langle$control states, situations$\rangle$ are "nodes" in an execution path, the definition of V^* is the least set of pairs of nodes containing: $\langle Q_0, t\rangle$ for all situation terms t such that $K(t, s)$, which includes s because K is assumed to be an equivalence relation; $\langle q, do(a_1, t)\rangle$ for all situation terms t such that $K(t, s)$ provided Q_0 advises a_1, it is executable at every t and the sensing function returns o for a_1 at s and $\delta(Q_0, o) = q$; and so on. By assumption, V^* contains as node $\langle Q_F, s'\rangle$ for $s' = do(a_1 \cdots a_k, s)$. But, by the definition of T^*, it follows that $T^*(Q_0, s, Q_F, s')$. Moreover, since $\phi(\overline{s'})$ and K is reflexive, $\phi(s')$. Contradiction. ∎

However, epistemically correct plans may not entail reasonable world-based properties in the noisy setting:

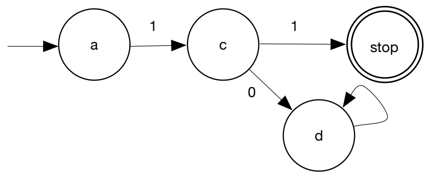

Figure 6: a problematic controller for relating belief and world-based correctness

THEOREM 21. *Suppose $\mathcal{D}$ is any action theory $\mathcal{X}, \Sigma$ as above, and ϕ is any formula not mentioning K. If $\mathcal{X}$ is epistemically correct for ϕ, then it does not follow that it is* TER-*adequate.*

Proof: Imagine a plan like in Figure 6, and let the control states with the label x be denoted by q_x. So q_a is the initial control state, and q_{stop} the final one. Suppose there is only a single initial world S_0, and $\forall a, s(Poss(a, s) \equiv true)$. The controller advises a and suppose b is *alt*-related to a, leading to two successor situations: $do(a, S_0)$ and $do(b, S_0)$. Suppose $SF(a, S_0) = 1$ and with that, we have $V^*(q_a, \overline{S_0}, q_c, \overline{do(a, S_0)})$, $U^*(q_a, S_0, q_c, do(a, S_0))$ and $U^*(q_a, S_0, q_c, do(b, S_0))$. Suppose now $SF(c, do(a, S_0)) = 1$ and so we have $V^*(q_a, \overline{S_0}, q_{stop}, \overline{do(a \cdot c, S_0)})$, and let us assume $\phi(\overline{do(a \cdot c, S_0)})$. However, suppose $SF(c, do(b, S_0)) = 0$. Then we have $U^*(q_a, S_0, q_d, do(b \cdot c, S_0))$, which will not terminate. ▮

The intuitive reason is that the adequacy conditions defined over U^* respond to sensing results along every execution path, whereas V^* only looks at the sensing outcomes for the path of advised actions from an initial world.

5. RELATED WORK

Generalizing plans, in fact, has been on the agenda since the early days of planning [16]. Algorithmic proposals to synthesize plans that generalize (in one way or another) vary widely in methodology, ranging from interactive theorem proving [44] to learning from examples [47]. The convergence of these approaches to loopy plans designed to solve multiple planning problem instances is a recent effort [8, 42, 20]. We refer interested readers to [42, 2] for a comprehensive list of references, and [43] for recent algorithmic work with nondeterministic effects.

Let us now review a few strands of work that have direct bearing to our efforts for this paper. The correctness of program-like plans was considered by Levesque [29], which is related to "strong planning" in the sense of [12]. This criteria of correctness was later scrutinized for stochastic domains in [2], where the notion of θ-adequacy was proposed. However, they considered an abstract (world-based) planning model (so, no logical framework to reason about actions) and most significantly, sensors were assumed to be exact, which means that problems like Examples 12 and 19 were out of reach. In other words, the current paper can be seen as pushing our understanding on the correctness of loopy plans further from an epistemic viewpoint: a language with an explicit notion of belief makes it possible to compare (♯) and Definition 17, for example.

Belief-based planning and programming are very popular in the literature [6, 22, 39, 26, 4], and the usual approach is formulate a nondeterministic (conformant) planning problem over belief states. However, [40] argued that an execution semantics that does not track progress in the real world is problematic. Our definition of V^* can be seen as reacting to sensing outcomes at the real world, as seen in Example 18. Finally, a major concern in agent design is to ensure that the agent has the sufficient knowledge to execute a plan or a program, especially in the context of task delegation, for which an account of *ability* is essential [46, 27]. How ability should be formalized in stochastic domains is still to be investigated, however.

There is a notable glaring omission from this discussion: the most influential planning framework in stochastic domains are (partially observable) Markov decision processes (PO-/MDPs), a natural model for sequential decision making under Markovian assumptions on probabilistic transitions [9]. In brief, a solution to a PO-/MDP is a mapping from (belief) states to actions that maximizes the expected reward over an infinite history, where a discount factor favors current rewards over future ones. (See [24], for example, for a synthesis framework, and [10] on rewards over infinite runs.) A number of recent approaches compute loopy plans for MDPs [38, 19, 36, 25], but the mathematical emphasis is very different from ours. The most significant difference is that there is no explicit stopping condition (equivalently, no analysis of termination), and the focus is instead on the accumulation of (possibly discounted) rewards over infinite execution paths. Thus, the style of analysis that we inherit from [12, 29] brings a new kind of clarity to such plan structures, and is similar in spirit to program analysis [13]. There is also a superficial difference, of course, that of goal states (in ours) versus rewards (in PO-/MDPs), but there are ways to bridge these linguistic particulars [23, 17]. What is more interesting, however, is think of goal formulas being the expectation of terms – see [4] for a formalization of expectation in $\mathcal{L}$ – and use that to further relate our ideas to PO-/MDPs objectives.

6. CONCLUSIONS

This paper introduced a logical account to reason about the correctness of loopy plans in partially observable stochastic domains. Beginning with an existing execution semantics T^* for noise-free loopy plans, we considered some extensions to an account on probabilistic beliefs in the situation calculus and introduced U^*. The idea is to use U^* in cases where belief is really a proxy for multiple problem instances, and we showed how a recent study of adequacy can be formalized here. Finally, we turned to loopy plan execution with belief states, and developed the execution semantics V^*. Here, V^* was shown to correctly handle noisy sensing and acting, but it can be seen to diverge from the intentions of U^*, for example.

The specification does provide us with a generic deduction-based planning procedure:

input: ϕ, $E^* \in \{T^*, U^*, V^*\}$, Δ is a correctness criteria

repeat with $\mathcal{X} \in$ FINITE STATE CONTROLLERS

if $\mathcal{D} \cup \Sigma \models \forall s.\ K(s, S_0) \supset \Delta(E^*, \phi, s)$ **then return** $\mathcal{X}$

Of course, we do not expect planners to actually use such a procedure. We believe existing algorithms for noise-free domains, *e.g.* bounded AND/OR searches [20], can be adapted for stochastic settings, and we leave such a investigation for the future. The two other pressing directions for the future include further clarification on the epistemic foundations of generalized plans in stochastic domains via an account of ability, and formalizing the notions of adequacy in terms of expectations of accumulated rewards.

REFERENCES

[1] F. Bacchus, J. Y. Halpern, and H. J. Levesque. Reasoning about noisy sensors and effectors in the situation calculus. *Artificial Intelligence*, 111(1–2):171 – 208, 1999.

[2] V. Belle and H. Levesque. Foundations for generalized planning in unbounded stochastic domains. In *KR*, 2016.

[3] V. Belle and H. J. Levesque. Reasoning about continuous uncertainty in the situation calculus. In *Proc. IJCAI*, 2013.

[4] V. Belle and H. J. Levesque. Allegro: Belief-based programming in stochastic dynamical domains. In *IJCAI*, 2015.

[5] S. Biundo. Present-day deductive planning. In *Current Trends in AI Planning. EWSP*, pages 1–5, 1994.

[6] B. Bonet and H. Geffner. Planning with incomplete information as heuristic search in belief space. In *AIPS*, pages 52–61, 2000.

[7] B. Bonet, H. Palacios, and H. Geffner. Automatic derivation of memoryless policies and finite-state controllers using classical planners. In *ICAPS*, 2009.

[8] B. Bonet, H. Palacios, and H. Geffner. Automatic derivation of finite-state machines for behavior control. In *AAAI*, 2010.

[9] C. Boutilier, T. Dean, and S. Hanks. Decision-theoretic planning: Structural assumptions and computational leverage. *Journal of Artificial Intelligence Research*, 11(1):94, 1999.

[10] K. Chatterjee and M. Chmelík. Pomdps under probabilistic semantics. *Artificial Intelligence*, 221:46–72, 2015.

[11] B. Chellas. *Modal logic*. Cambridge University Press, 1980.

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

[13] B. Cook. Automatically proving program termination. In *Computer Aided Verification, 19th International Conference, CAV 2007, Berlin, Germany, July 3-7, 2007, Proceedings*, page 1, 2007.

[14] E. Davis. Knowledge preconditions for plans. *Journal of Logic and Computation*, 4(5):721–766, 1994.

[15] R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. *Reasoning About Knowledge*. MIT Press, 1995.

[16] R. Fikes, P. Hart, and N. Nilsson. Learning and executing generalized robot plans. *Artificial intelligence*, 3:251–288, 1972.

[17] H. Geffner and B. Bonet. *A Concise Introduction to Models and Methods for Automated Planning*. Morgan and Claypool Publishers, 2013.

[18] M. Ghallab, D. Nau, and P. Traverso. *Automated Planning: theory and practice*. Morgan Kaufmann Publishers, 2004.

[19] C. Guestrin, D. Koller, C. Gearhart, and N. Kanodia. Generalizing plans to new environments in relational MDPs. In *IJCAI*, 2003.

[20] Y. Hu and G. De Giacomo. A generic technique for synthesizing bounded finite-state controllers. In *ICAPS*, 2013.

[21] Y. Hu and H. J. Levesque. A correctness result for reasoning about one-dimensional planning problems. In *IJCAI*, pages 2638–2643, 2011.

[22] L. P. Kaelbling and T. Lozano-Pérez. Integrated task and motion planning in belief space. *I. J. Robotic Res.*, 32(9-10):1194–1227, 2013.

[23] S. Koenig. Optimal probabilistic and decision-theoretic planning using markovian decision theory. Technical Report UCB/CSD-92-685, EECS Department, University of California, Berkeley, May 1992.

[24] A. Kučera and O. Stražovský. On the controller synthesis for finite-state markov decision processes. *Fundamenta Informaticae*, 82(1-2):141–153, 2008.

[25] A. Kumar and S. Zilberstein. History-based controller design and optimization for partially observable mdps. In *ICAPS*, 2015.

[26] J. Lang and B. Zanuttini. Knowledge-based programs as plans - the complexity of plan verification. In *Proc. ECAI*, pages 504–509, 2012.

[27] Y. Lespérance, H. J. Levesque, F. Lin, and R. B. Scherl. Ability and knowing how in the situation calculus. *Studia Logica*, 66(1):165–186, 2000.

[28] H. Levesque and G. Lakemeyer. *The logic of knowledge bases*. The MIT Press, 2001.

[29] H. J. Levesque. What is planning in the presence of sensing? In *Proc. AAAI / IAAI*, pages 1139–1146, 1996.

[30] F. Lin and H. J. Levesque. What robots can do: Robot programs and effective achievability. *Artif. Intell.*, 101(1-2):201–226, 1998.

[31] Z. Manna and R. J. Waldinger. A deductive approach to program synthesis. *ACM Trans. Program. Lang. Syst.*, 2(1):90–121, 1980.

[32] J. McCarthy and P. J. Hayes. Some philosophical problems from the standpoint of artificial intelligence. In *Machine Intelligence*, pages 463–502, 1969.

[33] R. C. Moore. A Formal Theory of Knowledge and Action. In *Formal Theories of the Commonsense World*, pages 319–358. Ablex, Norwood, NJ, 1985.

[34] L. Morgenstern. Knowledge preconditions for actions and plans. In *IJCAI*, pages 867–874, 1987.

[35] C. Muise, V. Belle, P. Felli, S. McIlraith, T. Miller, A. Pearce, and L. Sonenberg. Planning over multi-agent epistemic states: A classical planning approach. In *Proc. AAAI*, 2015.

[36] J. K. Pajarinen and J. Peltonen. Periodic finite state controllers for efficient pomdp and dec-pomdp planning. In *NIPS*, pages 2636–2644, 2011.

[37] R. Petrick and F. Bacchus. Extending the knowledge-based approach to planning with incomplete information and sensing. In *Proc. ICAPS*, pages 2–11, 2004.

[38] P. Poupart and C. Boutilier. Bounded finite state controllers. In *NIPS*, pages 823–830, 2004.

[39] R. Reiter. *Knowledge in action: logical foundations for specifying and implementing dynamical systems*. MIT Press, 2001.

[40] S. Sardiña, G. De Giacomo, Y. Lespérance, and H. J. Levesque. On the limits of planning over belief states under strict uncertainty. In *KR*, pages 463–471, 2006.

[41] R. B. Scherl and H. J. Levesque. Knowledge, action, and the frame problem. *Artificial Intelligence*, 144(1-2):1–39, 2003.

[42] S. Srivastava. *Foundations and Applications of Generalized Planning*. PhD thesis, Department of Computer Science, University of Massachusetts Amherst, 2010.

[43] S. Srivastava, S. Zilberstein, A. Gupta, P. Abbeel, and S. Russell. Tractability of planning with loops. In *AAAI*, 2015.

[44] W. Stephan and S. Biundo. Deduction-based refinement planning. In *AIPS*, pages 213–220, 1996.

[45] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. MIT Press, 2005.

[46] W. van der Hoek, B. van Linder, and J. C. Meyer. A logic of capabilities. In *LFCS*, pages 366–378, 1994.

[47] E. Winner and M. M. Veloso. LoopDISTILL: Learning domain-specific planners from example plans. In *Workshop on AI Planning and Learning, ICAPS*, 2007.