Reasoning about Actions in Fuzzy Environment

Emad Saad[1] Shaimaa A. Elmorsy[2] Mahmoud M. H. Gabr[2] Yasser F. M. Hassan[2]

1. Department of Computer Science, Gulf University for Science and Technology
West Mishref, Kuwait
2. Department of Mathematics and Computer Science, Faculty of Science, Alexandria University
Alexandria, Egypt
Email: saad.e@gust.edu.kw, shaimaa_elmorsy@hotmail.com, mahgabr@yahoo.com, y.fouad@sci.alex.edu.eg

Abstract— *Reasoning in the presence of imprecision and vagueness is inevitable in many real-world applications including those in robotics and intelligent agents. Although, reasoning about actions is a major component in these real-world applications, current actions languages for reasoning about actions lack the ability to represent and reason about actions in the presence of imprecision and vagueness that stem from effects of actions in these real-world applications. In this paper we present a new action language called fuzzy action language, $\mathcal{A}_F$, that allows the representation and reasoning about actions with vague (fuzzy) effects. In addition we define the notions of fuzzy planning and fuzzy plan in the fuzzy action language $\mathcal{A}_F$. Furthermore, we describe a fuzzy planner based on the fuzzy action language A_F that is developed by translating a fuzzy action theory, $\mathbf{FT}$, in A_F into a normal logic program with answer set semantics, Π, where trajectories in $\mathbf{FT}$ are equivalent to the answer sets of Π. In addition, we formally prove the correctness of our translation. Furthermore, we show that fuzzy planning problems can be encoded as a SAT problem.*

Keywords— Answer set programming, fuzzy planning, reasoning about actions with fuzzy effects, reasoning under uncertainty.

1 Introduction

Reasoning about the properties of actions is fundamental in many real-world applications. Therefore, many action languages that allow representing and reasoning about actions have been developed which include [2, 5, 9, 10, 11, 12, 18, 21, 22, 24]. Furthermore, uncertainty is a main issue in representing and reasoning about actions in these real-world applications. Uncertainty in these real-world applications stems from different sources including incompleteness, erroneous, or imprecision (vagueness). For reasoning under uncertainty, probability theory is used to reason in the presence of incompleteness or erroneous, however, fuzzy set theory is used to reason in the presence of imprecision or vagueness. To deal with probabilistic uncertainty in reasoning about actions, different proposals have been presented. These proposals include [2, 5, 12, 21, 22]. Although, these proposals deal with uncertainty probabilistically, they are inappropriate in situations where actions have imprecise or vague effects, which need a different formalism from the one required in reasoning about actions with probabilistic effects as in [2, 5, 12, 21, 22]. This is because the underlying formalism in reasoning about actions with probabilistic effects is the probability theory, however, the formalism required for reasoning about actions with imprecise effects is the fuzzy set theory. Actions with imprecise effects appear in many domains including robotics and intelligent agents. Consider the following example adapted from [23]. Consider a planner that controls an autonomous car that is moving in a highway with the goal to keep the car close to the middle of a lane. The autonomous car planner's uses the action navigate whose effect is to make the car close to the middle of the lane. The effect of the action navigate is neither probabilistically nor precisely defined. Therefore, action languages, for example [9, 10, 11, 18, 24], that reason about actions whose effects are precisely defined (either entirely true or entirely false), and action languages, for example [2, 5, 12, 21, 22], that reason about actions whose effects are probabilistically defined (either true or false to a probabilistic degree), can not be used to correctly represent and reason about these kind of actions whose effects are imprecisely or vaguely defined.

In this paper we present a new action language called fuzzy action language, $\mathcal{A}_F$, that allows the representation and reasoning about actions with vague (fuzzy) effects. In addition, we define the notions of fuzzy planning and fuzzy plan in the fuzzy action language $\mathcal{A}_F$. Furthermore, we describe a fuzzy planner based on the fuzzy action language $\mathcal{A}_F$ that is developed by translating a fuzzy action theory, $\mathbf{FT}$, in $\mathcal{A}_F$ into a normal logic program with answer set semantics, Π, where trajectories in $\mathbf{FT}$ are equivalent to the answer sets of Π. In addition, we formally prove the correctness of our translation. Furthermore, we show that fuzzy planning problems can be encoded as SAT problems.

This paper is organized as follows. Section 2 reviews the answer set semantics of normal logic programs and the fundamental notions of the fuzzy set theory. The syntax and semantics of the fuzzy action language, $\mathcal{A}_F$, is introduced in section 3. In section 4, a translation from a fuzzy planning problem in the fuzzy action language $\mathcal{A}_F$ into a normal logic program with answer set semantics is presented. The correctness of the translation is introduced in section 5. Finally, conclusion and related work is presented in section 6.

2 Preliminaries

In this section we review the answer set semantics of normal logic programs [7] and the basic notions of fuzzy set theory [27].

2.1 Normal Logic Programs with Answer Set Semantics

Let $\mathcal{L}$ be a first-order language with finitely many predicate symbols, function symbols, constants, and infinitely many variables. The Herbrand base of $\mathcal{L}$ is denoted by $\mathcal{B}$. A Her-

ISBN: 978-989-95079-6-8

brand interpretation is a subset of the Herbrand base $\mathcal{B}$. A normal logic program is a finite set of rules of the form

$$a \leftarrow a_1, \ldots, a_n, not\ b_1, \ldots, not\ b_m$$

where a, $a_1, \ldots, a_n, b_1, \ldots, b_m$ are atoms and not is the negation-as-failure. Intuitively, the meaning of the above rule is that if it is believable (provable) that a_i is true and it is not believable that b_i is true then it is believable that a is true. A normal logic program is ground if no variables appear in any of its rules.

Let Π be a ground normal logic program, S be a Herbrand interpretation, and r be a rule as above. Then, we say that

- S satisfies a_i (denoted by $S \models a_i$) iff $a_i \in S$.
- S satisfies $not\ b_j$ (denoted by $S \models not\ b_j$) iff $b_j \notin S$.
- S satisfies $(a_1, \ldots, a_n, not\ b_1, \ldots, not\ b_m)$ (denoted by $S \models (a_1, \ldots, a_n, not\ b_1, \ldots, not\ b_m)$) iff $\{a_1, \ldots, a_n\} \subseteq S$ and $\{b_1, \ldots, b_m\} \nsubseteq S$.
- S satisfies $a \leftarrow a_1, \ldots, a_n, not\ b_1, \ldots, not\ b_m$ iff $S \models a$ or $S \not\models (a_1, \ldots, a_n, not\ b_1, \ldots, not\ b_m)$.

A Herbrand interpretation is said to satisfy a normal logic program Π iff it satisfies every rule in Π. A Herbrand model of a normal logic program Π is a Herbrand interpretation of Π that satisfies Π. A Herbrand interpretation S of a normal logic program Π is said to be an answer set of Π if S is the minimal Herbrand model (with respect to the set inclusion) of the reduct, denoted by Π^S, of Π w.r.t. S, where Π^S is a set of rules of the form

$$a \leftarrow a_1, \ldots, a_n$$

such that

$$a \leftarrow a_1, \ldots, a_n, not\ b_1, \ldots, not\ b_m \in \Pi$$

and $\{b_1, \ldots, b_m\} \nsubseteq S$

Intuitively, for any $not\ b_j$ in the body of a rule in Π with $b_j \notin S$ is simply satisfied by S, and $not\ b_j$ is safely removed from the body of the rule. If $b_j \in S$ then the body of the rule is not satisfied and the rule is trivially ignored.

2.2 Fuzzy Sets

In this section we review the basic notions of fuzzy sets as presented in [27]. Let U be a set of objects. A fuzzy set, F, in U is defined by the grade membership function $\mu_F : U \rightarrow [0, 1]$, where for each element $x \in U$, μ_F assigns to x a value $\mu_F(x)$ in $[0, 1]$. The support for F denotes the set of all objects x in U for which the grade membership of x in F is a non-zero value. Formally, $support(F) = \{x \in U \mid \mu_F(x) > 0\}$. The intersection (conjunction) of two fuzzy sets F and F' in U, denoted by $F \wedge_f F'$ is a fuzzy set G in U where the grade membership function of G is $\mu_G(x) = \min(\mu_F(x), \mu_{F'}(x))$ for all $x \in U$. However, the union (disjunction) of two fuzzy sets F and F' in U, denoted by $F \vee_f F'$ is a fuzzy set G in U where the grade membership function of G is $\mu_G(x) = \max(\mu_F(x), \mu_{F'}(x))$ for all $x \in U$. The complement (negation) of a fuzzy set F in U is a fuzzy set in U denoted by $\overline{F}$ where the grade membership function of $\overline{F}$ is $\mu_{\overline{F}}(x) = 1 - \mu_F(x)$ for all $x \in U$. A fuzzy set F in U is said to be contained in another fuzzy set G in U if and only if $\mu_F(x) \leq \mu_G(x)$ for all $x \in U$. Notice that we use the notations $\wedge_f$ and $\vee_f$ to denote fuzzy conjunction and fuzzy disjunction respectively to distinguish them from $\wedge$ and $\vee$ for propositional conjunction and disjunction respectively. Furthermore, other function characterizations for the fuzzy conjunction and fuzzy disjunction operators can be used. However, we will stick with the min and max function characterizations for the fuzzy conjunction and fuzzy disjunction as originally proposed in [27].

3 Fuzzy Action Language $\mathcal{A}_F$

In this section we develop a novel action language, called *fuzzy action language*, $\mathcal{A}_F$, that allows the representation of actions with vague (fuzzy) effects. An action theory in $\mathcal{A}_F$ is capable of representing the grade membership distribution of the possible initial states, the executability conditions of actions, and the grade membership distribution of the fuzzy effects of actions. The semantics of $\mathcal{A}_F$ is based on a transition function that maps an action and a set of states to a set of states. The fuzzy action language $\mathcal{A}_F$ is inspired by the action languages in [21, 22].

3.1 Language syntax

A predicate that describes a property of the environment, which may contain variables, is called a fluent. Let $\mathcal{A}$ be a set of action names that can contain variables and $\mathcal{F}$ be a set of fluents. A fluent $f \in \mathcal{F}$ or $\neg f$, the negation of f is called a fluent literal. A conjunction of fluent literals of the form $l_1 \wedge \ldots \wedge l_n$ is called a conjunctive fluent formula, where $l_1, \ldots, l_n$ are fluent literals. Sometimes we abuse the notation and refer to a conjunctive fluent formula as a set of fluent literals ($\emptyset$ denotes $true$).

A fuzzy action theory, $\mathbf{FT}$, in $\mathcal{A}_F$ is a tuple of the form $\mathbf{FT} = \langle S_0, \mathcal{AD} \rangle$, where S_0 is a proposition of the form (1), $\mathcal{AD}$ is a set of propositions from (2-3) as follows:

$$\textbf{initially} \begin{cases} \psi_1 \ : \ v_1 \\ \psi_2 \ : \ v_2 \\ \quad \ldots \\ \psi_n \ : \ v_n \end{cases} \tag{1}$$

$$\textbf{executable}\ a\ \textbf{if}\ \psi \tag{2}$$

$$a\ \textbf{causes} \begin{cases} \phi_1 \ : \ v_1\ \textbf{if}\ \psi_1 \\ \phi_2 \ : \ v_2\ \textbf{if}\ \psi_2 \\ \quad \ldots \\ \phi_n \ : \ v_n\ \textbf{if}\ \psi_n \end{cases} \tag{3}$$

where $\psi, \psi_1, \ldots, \psi_n, \phi_1, \ldots, \phi_n$ are conjunctive fluent formulae, $a \in \mathcal{A}$ is an action, and for all $1 \leq i \leq n$, we have $v_i \in [0, 1]$. The set of all ψ_i must be mutually exclusive. Let $\mathcal{S}$ be the set of all states formed from the fluents in $\mathcal{F}$.

Proposition (1) is a fuzzy set in the set of all states $\mathcal{S}$. It represents the grade membership of the possible initial states. We consider that only ψ_i with grade membership that is non-zero is listed in (1). Proposition (1) says that the grade membership of the possible initial state ψ_i is v_i for all $1 \leq i \leq n$. Proposition (2) represents the *executability condition* of actions, where each variable that appears in a also appears in ψ. It states that an action a is executable in any state in which ψ holds. A proposition of the form (3) represents the fuzzy (vague) effects resulting from executing an action a in the

ISBN: 978-989-95079-6-8

states in which a is executable. For each $1 \leq i \leq n$, all variables that appear in ϕ_i also appear in a and ψ_i. It describes that a causes ϕ_i to hold with grade membership v_i in a successor state to a state in which a is executed and ψ_i holds for all $1 \leq i \leq n$. For each $1 \leq i \leq n$, ψ_i is called a precondition of an action a that corresponds to an effect ϕ_i, ϕ_i is called an effect of a, v_i is the grade membership that ϕ_i holds given that ψ_i holds, where v_i is a non-zero value in $[0, 1]$. For any proposition of the form (3), the set of ground preconditions ψ_i are mutually exclusive and exhaustive.

It will be more convenient for the subsequent results to represent an action a as a set of the form $a = \{a_1, \ldots, a_n\}$, where each a_i corresponds to ϕ_i, v_i, and ψ_i. Therefore, alternatively, for each $1 \leq i \leq n$, proposition (3) can be represented as

$$a_i \textbf{ causes } \phi_i \ : \ v_i \textbf{ if } \psi_i$$

A fuzzy action theory is ground if it does not contain any variables.

3.2 Semantics

A consistent set of ground literals ϕ is a set of literals that does not contain a pair of complementary literals, i.e., l and $\neg l \notin \phi$. If a literal $l \in \phi$, then we say l is true (holds) in ϕ (denoted by $\phi \models l$), and l is false (does not hold) in ϕ if $\neg l$ is in ϕ (denoted by $\phi \models \neg l$). If a set of literals σ is contained in ϕ then we say σ is true (holds) in ϕ (denoted by $\phi \models \sigma$), otherwise, σ is false (does not hold) in ϕ (denoted by $\phi \not\models \sigma$). A state s is a complete and consistent set of literals that describes the world at a certain time point.

Definition 1 *Let* $\mathbf{FT} = \langle S_0, \mathcal{AD} \rangle$ *be a ground fuzzy action theory,* $\mathcal{S}$ *is the set of all states in* $\mathbf{FT}$*,* s *be a state whose grade membership* $\mu_{\mathcal{S}}(s)$ *is* v*,* a_i **causes** ϕ_i : v_i : **if** ψ_i *(*$1 \leq i \leq n$*) be a proposition in* $\mathcal{AD}$*, and* $a = \{a_1, \ldots, a_n\}$ *be an action, where each* a_i *corresponds to* ϕ_i, v_i*, and* ψ_i *for* $1 \leq i \leq n$*. Then,* $s' = \Phi(a_i, s)$ *is the state resulting from executing* a *in* s*, given that* a *is executable in* s*, where:*

- $l \in \Phi(a_i, s)$ *and* $\neg\, l \notin \Phi(a_i, s)$ *if* $l \in \phi_i$ *and the precondition* ψ_i *holds in* s*.*
- $\neg\, l \in \Phi(a_i, s)$ *and* $l \notin \Phi(a_i, s)$ *if* $\neg\, l \in \phi_i$ *and the precondition* ψ_i *holds in* s*.*
- *Otherwise,* $l \in \Phi(a_i, s)$ *iff* $l \in s$ *and* $\neg\, l \in \Phi(a_i, s)$ *iff* $\neg\, l \in s$*.*

where the grade membership of the resulting state $s' = \Phi(a_i, s)$ *is* $\mu_{\mathcal{S}}(s') = \min(\mu_{\mathcal{S}}(s), v_i) = \min(v, v_i)$*. We call* Φ *a fuzzy transition function.*

Example 1 *Consider the following fuzzy planning task adapted from [14]. An arm of a robot is grasping a block from a table, where the* pickup *action the robot performs has effects that are imprecisely defined. The arm is able to tightly hold a block (hb) with a grade membership* 0.9 *after executing the* pickup *action in the state in which the gripper is dry (gd), and the arm tightly cannot hold the block (¬hb), after executing the* pickup *action in the same state, with grade membership value* 0.4*. On the other hand, when executing the* pickup *action in the state while the gripper is not dry (¬ gd) causes the block to be tightly held (hb) with grade of membership equal to* 0.3 *and tightly not held (¬ hb) with grade membership* 0.6*. We assume the initial grade membership distribution of the initial states of the world is given by the grade membership function* $\mu_{\mathcal{S}}$ *such that* $\mu_{\mathcal{S}}(s_1) = 0.9$ *and* $\mu_{\mathcal{S}}(s_2) = 0.5$*, where* $s_1 = \{gd, \neg hb\}$ *and* $s_2 = \{\neg gd, \neg hb\}$*, with the understanding that the grade membership of the other states of the world is* 0*. This fuzzy planning domain can be represented in the fuzzy action language* $\mathcal{A}_F$ *as the fuzzy action theory* $\mathbf{FT} = \langle S_0, \mathcal{AD} \rangle$ *where*

$$S_0 = \textbf{initially} \left\{ \begin{array}{lcl} \{gd, \neg hb\} & : & 0.9 \\ \{\neg gd, \neg hb\} & : & 0.5 \end{array} \right.$$

and $\mathcal{AD}$ *consists of:*

$$\textbf{executable } pickup \textbf{ if } \emptyset$$

$$pickup \textbf{ causes} \left\{ \begin{array}{lclll} \{hb\} & : & 0.9 & \textbf{if} & \{gd\} \\ \{\neg hb\} & : & 0.4 & \textbf{if} & \{gd\} \\ \{hb\} & : & 0.3 & \textbf{if} & \{\neg\, gd\} \\ \{\neg\, hb\} & : & 0.6 & \textbf{if} & \{\neg\, gd\} \end{array} \right.$$

The $pickup$ action can be represented as the set $pickup = \{pickup_1, pickup_2, pickup_3, pickup_4\}$, where

$$\begin{array}{llllll} pickup_1 & \textbf{causes} & \{hb\} & : & 0.9 & \textbf{if} \ \{gd\} \\ pickup_2 & \textbf{causes} & \{\neg hb\} & : & 0.4 & \textbf{if} \ \{gd\} \\ pickup_3 & \textbf{causes} & \{hb\} & : & 0.3 & \textbf{if} \ \{\neg\, gd\} \\ pickup_4 & \textbf{causes} & \{\neg\, hb\} & : & 0.6 & \textbf{if} \ \{\neg\, gd\} \end{array}$$

The grade membership distribution resulting from executing the $pickup$ action in the initial states s_1 and s_2 is given by:

- $s_1^{'} = \{gd, hb\} = \Phi(pickup_1, \{gd, \neg hb\})$, where $\mu_S(s_1^{'}) = \min(\mu_S(s_1), 0.9) = \min(0.9, 0.9) = 0.9$.
- $s_2^{'} = \{gd, \neg hb\} = \Phi(pickup_2, \{gd, \neg hb\})$, where $\mu_S(s_2^{'}) = \min(\mu_S(s_1), 0.4) = \min(0.9, 0.4) = 0.4$.
- $s_3^{'} = \{\neg gd, hb\} = \Phi(pickup_3, \{\neg gd, \neg hb\})$, where $\mu_S(s_3^{'}) = \min(\mu_S(s_2), 0.3) = \min(0.5, 0.3) = 0.3$.
- $s_4^{'} = \{\neg gd, \neg hb\} = \Phi(pickup_4, \{\neg gd, \neg hb\})$, where $\mu_S(s_4^{'}) = \min(\mu_S(s_2), 0.6) = \min(0.5, 0.6) = 0.5$.

Definition 2 (Fuzzy Plan) *A sequence of actions* $\langle a_0, a_1, \ldots, a_{n-1} \rangle$ *is called a fuzzy plan, where each* $(0 \leq i \leq n-1)$ a_i *is an action with fuzzy effects.*

The grade membership that a fuzzy plan P satisfies a conjunctive fluent formula $\mathcal{G}$ after executing P in a given state s is given by the following definition.

Definition 3 *Let* $\mathbf{FT}$ *be a ground fuzzy action theory,* s, s' *be states,* s_I *be a variable ranging over the possible initial states,* $\mathcal{G}$ *be a conjunctive fluent formula, and* $\langle a_0, a_1, \ldots, a_{n-1} \rangle$ *be a fuzzy plan. Then the grade membership that* $\mathcal{G}$ *is true in a state* s' *that results after executing* $\langle a_0, a_1, \ldots, a_{n-1} \rangle$ *in the possible initial states* s_I *is given by*

$$\mu'_{\mathcal{P}}(s' | s_I, \langle a_0, a_1, \ldots, a_{n-1} \rangle) =$$
$$\max_{s''}(\min_{s_I = s}(\mu'_{\mathcal{P}}(s'' | s, a_0),\ \mu'_{\mathcal{P}}(s' | s'', \langle a_1, \ldots, a_{n-1} \rangle)))$$

where $\mathcal{P}$ *is the set of all plans in* $\mathbf{FT}$*, and* $\mu'_{\mathcal{P}}(s'' | s, a_0) = \mu_{\mathcal{S}}(s'') = \min(\mu_{\mathcal{S}}(s), v)$ *such that* $s'' = \Phi(a_0, s)$*. In general, for any action* $a \in \mathcal{A}$*,* $\mu'_{\mathcal{P}}(s'' | s, a) = \mu_{\mathcal{S}}(s'') = \min(\mu_{\mathcal{S}}(s), v)$ *such that* $s'' = \Phi(a, s)$*.*

ISBN: 978-989-95079-6-8

Definition 4 *A fuzzy planning problem is a 4-tuple* $\mathbf{FP} = \langle S_0, \mathcal{AD}, \mathcal{G}, \mathcal{T}\rangle$*, where* S_0 *is a fuzzy set in the set of all states* S *that represents the initial agent knowledge about the world states at the time of execution (the initial grade membership distribution over states),* $\mathcal{AD}$ *is a fuzzy action description,* $\mathcal{G}$ *is conjunctive fluent formula represents the goal to be satisfied, and* $0 \leq \mathcal{T} \leq 1$ *is the fuzzy threshold for the goal* $\mathcal{G}$ *to be achieved. We say* $\langle a_0, \ldots, a_{n-1}\rangle$ *is a fuzzy plan for* $\mathbf{FP}$ *iff each* a_i *appears in* $\mathcal{AD}$ *and* $\mu'_{\mathcal{P}}(s'|s_I, \langle a_0, a_1, \ldots, a_{n-1}\rangle) \geq \mathcal{T}$*, where* s_I *is a variable ranging over the possible initial states, and* $\mathcal{G}$ *is true in* s'*.*

4 Fuzzy Planning Using Answer Set Programming

This section presents a translation from a fuzzy planning problem $\mathbf{FP} = \langle S_0, \mathcal{AD}, \mathcal{G}, \mathcal{T}\rangle$ into a normal logic program with answer set semantics, $\Pi_{\mathbf{FP}}$, where the rules in $\Pi_{\mathbf{FP}}$ encode (1) the initial grade membership distribution S_0, (2) the fuzzy transition function Φ, (3) the fuzzy action description $\mathcal{AD}$, (4) and the goal $\mathcal{G}$. The answer sets of $\Pi_{\mathbf{FP}}$ correspond to valid trajectories in $\mathbf{FP}$. The normal logic program translation of a fuzzy planning problem $\mathbf{FP}$ is mainly adapted from [25]. We assume that the length of the fuzzy plan that we are looking for is known. We use the predicates $holds(L, T)$ to represent the fact that a literal L holds at time moment T and $occ(AC, T)$ to describe that an action AC executes at time moment T. We use lower case letters to represent constants and upper case letters to represent variables.

Let $\Pi_{\mathbf{FP}}$ be the normal logic program translation of a fuzzy planning problem $\mathbf{FP} = \langle S_0, \mathcal{AD}, \mathcal{G}, \mathcal{T}\rangle$, where $\Pi_{\mathbf{FP}}$ is the set of rules described as follows. In addition, given p is a predicate and $\psi = \{l_1, \ldots, l_n\}$, we use $p(\psi)$ to denote $p(l_1), \ldots, p(l_n)$.

- For each action $a = \{a_1, \ldots, a_n\} \in \mathcal{A}$, we add to $\Pi_{\mathbf{FP}}$ the set of facts

$$action(a_i) \leftarrow \tag{4}$$

for each $1 \leq i \leq n$. States of the world are described by literals that are encoded in $\Pi_{\mathbf{FP}}$ by the rules

$$literal(A) \leftarrow atom(A) \tag{5}$$
$$literal(\neg A) \leftarrow atom(A) \tag{6}$$

where $atom(A)$ is a set of facts that describe the properties of the world. To present that A and $\neg A$ are contrary literals, the following rules are added to $\Pi_{\mathbf{FP}}$.

$$contrary(A, \neg A) \leftarrow atom(A) \tag{7}$$
$$contrary(\neg A, A) \leftarrow atom(A) \tag{8}$$

- The initial grade membership distribution **initially** $\psi_i : v_i$ for $1 \leq i \leq n$ is represented in $\Pi_{\mathbf{FP}}$ as follows. Consider that $s_1, s_2, \ldots, s_n$ form the set of possible initial states, where for each $1 \leq i \leq n$, $s_i = \{l^i_1, \ldots, l^i_m\}$, and the grade membership of s_i is $\mu_{\mathcal{S}}(s_i) = v_i$. Moreover, let $s = s_1 \cup s_2 \cup \ldots \cup s_n$, $s' = s_1 \cap s_2 \cap \ldots \cap s_n$, $\widehat{s} = s - s'$, and $s'' = \{l \,|\, l \in \widehat{s} \vee \neg l \in \widehat{s}\}$. To generate the set of all possible initial states, the following set of rules are added to $\Pi_{\mathbf{FP}}$. For each literal $l \in s'$, the fact

$$holds(l, 0) \leftarrow \tag{9}$$

is in $\Pi_{\mathbf{FP}}$. This fact presents that the literal l holds at time moment 0. This set of facts specifies the set of literals that hold in every possible initial state. Moreover, for each literal $l \in s''$, we add to $\Pi_{\mathbf{FP}}$ the rules

$$holds(l, 0) \leftarrow not\ holds(\neg l, 0) \tag{10}$$
$$holds(\neg l, 0) \leftarrow not\ holds(l, 0) \tag{11}$$

The above rules say that the literal l (similarly $\neg l$) holds at time moment 0, if $\neg l$ (similarly l) does not hold at the time moment 0.

- Each executability condition proposition of an action $a = \{a_1, \ldots, a_n\}$ of the form (2) is encoded in $\Pi_{\mathbf{FP}}$ for each $1 \leq i \leq n$ as

$$exec(a_i, T) \leftarrow holds(\psi, T) \tag{12}$$

- For each proposition of the form a_i **causes** $\phi_i : v_i$ **if** ψ_i ($1 \leq i \leq n$), in $\mathcal{AD}$, we proceed as follows. Let $\phi_i = \{l^1_i, \ldots, l^m_i\}$. Then, $\forall$ $(1 \leq i \leq n)$, we have for each $1 \leq j \leq m$,

$$holds(l^j_i, T+1) \leftarrow occ(a_i, T), exec(a_i, T), holds(\psi_i, T) \tag{13}$$

belongs to $\Pi_{\mathbf{FP}}$. This rule states that if the action a occurs at time moment T and the precondition ψ_i holds at the same time moment, then the literal l^j_i holds at the time moment $T + 1$.

- The frame axioms are presented in $\Pi_{\mathbf{FP}}$ as below. For any literal L we have the rule

$$holds(L, T+1) \leftarrow holds(L, T), not\ holds(L', T+1), contrary(L, L') \tag{14}$$

in $\Pi_{\mathbf{FP}}$. The above rule states that L holds at the time moment $T + 1$ if it holds at the time moment T and its contrary does not hold at the time moment $T + 1$.

- To encode the fact that a literal A and its negation $\neg A$ cannot hold at the same time, we add the following rule to $\Pi_{\mathbf{FP}}$

$$\leftarrow holds(A, T), holds(\neg A, T) \tag{15}$$

- Action generation rules are described by

$$occ(AC^i, T) \leftarrow action(AC^i), not\ abocc(AC^i, T) \tag{16}$$
$$abocc(AC^i, T) \leftarrow action(AC^i), action(AC^j), occ(AC^j, T), AC^i \neq AC^j \tag{17}$$

The above rules generate action occurrences once at a time, where AC^i and AC^j are variables representing actions.

- Let $\mathcal{G} = g_1 \wedge \ldots \wedge g_m$ be a goal expression, then $\mathcal{G}$ is encoded in $\Pi_{\mathbf{FP}}$ as

$$goal \leftarrow holds(g_1, T), \ldots, holds(g_m, T) \tag{18}$$

ISBN: 978-989-95079-6-8

Example 2 *The normal logic program translation, $\Pi_{\mathbf{FP}}$, of the fuzzy planning problem $\mathbf{FP} = \langle S_0, \mathcal{AD}, \mathcal{G}, \mathcal{T} \rangle$, described in Example 1 proceeds as follows, where S_0 and $\mathcal{AD}$ are as presented in Example 1, $\mathcal{G} = \{hb\}$, and $\mathcal{T} \in [0, 1]$. In addition to the rules (5), (6), (7), (8), (14), (15), (16), and (17), $\Pi_{\mathbf{FP}}$ contains the rules*

$$action(pickup_1) \leftarrow$$
$$action(pickup_2) \leftarrow$$
$$action(pickup_3) \leftarrow$$
$$action(pickup_4) \leftarrow$$

where pickup is in $\mathcal{A}$. Properties of the world are described by the atoms gd (gripper dry) and hb (holding block), which are encoded in $\Pi_{\mathbf{FP}}$ by the rules

$$atom(gd) \leftarrow$$
$$atom(hb) \leftarrow$$

Executability conditions of action pickup are encoded in $\Pi_{\mathbf{FP}}$ by the rules

$$exec(pickup_1, t) \leftarrow$$
$$exec(pickup_2, t) \leftarrow$$
$$exec(pickup_3, t) \leftarrow$$
$$exec(pickup_4, t) \leftarrow$$

where $0 \leq t \leq n$. The set of possible initial states are encoded in $\Pi_{\mathbf{FP}}$ by the rules:

$$\begin{array}{lll} holds(\neg hb, 0) & \leftarrow & \\ holds(gd, 0) & \leftarrow & not\ holds(\neg gd, 0) \\ holds(\neg gd, 0) & \leftarrow & not\ holds(gd, 0) \end{array}$$

Effects of the pickup action are encoded in $\Pi_{\mathbf{FP}}$ by the rules

$$\begin{array}{lll} holds(hb, T+1) & \leftarrow & occ(pickup_1, T), \\ & & exec(pickup_1, T), holds(gd, T) \\ holds(\neg hb, T+1) & \leftarrow & occ(pickup_2, T), \\ & & exec(pickup_2, T), holds(gd, T) \\ holds(hb, T+1) & \leftarrow & occ(pickup_3, T), \\ & & exec(pickup_3, T), holds(\neg gd, T) \\ holds(\neg hb, T+1) & \leftarrow & occ(pickup_4, T), \\ & & exec(pickup_4, T), holds(\neg gd, T) \end{array}$$

The goal is encoded in $\Pi_{\mathbf{FP}}$ by the rule

$$goal \quad \leftarrow \quad holds(hb, T)$$

5 Correctness

In this section we prove the correctness of our translation. We show that the answer sets of the normal logic program translation, $\Pi_{\mathbf{FP}}$, of a fuzzy planning problem, $\mathbf{FP}$, correspond to trajectories in $\mathbf{FP}$. Consider that the domain of T is $\{0, \ldots, n\}$. Assume that Φ is the fuzzy transition function associated with $\mathbf{FP}$, s_0 is a possible initial state, and $a_0, \ldots, a_{n-1}$ be a set of actions in $\mathcal{A}$. Any action a_i can be represented as a set where $a_i = \{a_{1_i}, \ldots, a_{m_i}\}$. Therefore, a trajectory in $\mathbf{FP}$ is $s_0\ a_{j_0}\ s_1 \ldots a_{j_{n-1}}\ s_n$ for $(1 \leq j \leq m)$ and $(0 \leq i \leq n)$, such that $\forall (0 \leq i \leq n)$, s_i is a state, a_i is an action, $a_{j_i} \in a_i = \{a_{1_i}, \ldots, a_{m_i}\}$, and $s_i = \Phi(a_{j_{i-1}}, s_{i-1})$.

Theorem 1 *Let $\mathbf{FP} = \langle S_0, \mathcal{AD}, \mathcal{G}, \mathcal{T} \rangle$ be a fuzzy planning problem, P be a fuzzy plan in $\mathbf{FP}$, and T_P be the set of all trajectories in P. Then, $s_0\ a_{j_0}\ s_1 \ldots a_{j_{n-1}}\ s_n$ is a trajectory in T_P iff $occ(a_{j_0}, 0), \ldots, occ(a_{j_{n-1}}, n-1)$ is true in an answer set of $\Pi_{\mathbf{FP}}$.*

Theorem 1 presents that any fuzzy planning problem, $\mathbf{FP}$, can be translated into a normal logic program with answer set semantics, $\Pi_{\mathbf{FP}}$, such that a trajectory in $\mathbf{FP}$ is equivalent to an answer set of $\Pi_{\mathbf{FP}}$. Theorem 1 shows that normal logic programs with answer set semantics can be used to find fuzzy plans for fuzzy planning problems in two steps. The first step is to translate a fuzzy planning problem, $\mathbf{FP}$, into a normal logic program whose answer sets correspond to valid trajectories in $\mathbf{FP}$. From the answer sets of the normal logic program translation of $\mathbf{FP}$, the set of trajectories T_P that correspond to a fuzzy plan P in $\mathbf{FP}$ is determined. The second step is to calculate the grade membership of the fuzzy plan P using the formula

$$\max_{s_0\ a_{j_0}\ s_1 \ldots a_{j_{n-1}}\ s_n \in T_P} \left(\min_{0 \leq i \leq n-1} (\mu_{\mathcal{S}}(s_i), \mu_{\mathcal{S}}(s_{i+1})) \right)$$

Furthermore, we show that any fuzzy planning problem can be encoded as a SAT formula. Hence, state-of-the-art SAT solvers can be used to find fuzzy plans for fuzzy planning problems. Any normal logic program, Π, can be translated into a SAT formula, $\mathcal{S}$, where the models of $\mathcal{S}$ are equivalent to the answer sets of Π [19]. Therefore, the normal logic program translation of a fuzzy planning problem $\mathbf{FP}$ can be encoded into an equivalent SAT formula, where the models of $\mathcal{S}$ correspond to valid trajectories in $\mathbf{FP}$.

Theorem 2 *Let $\mathbf{FP}$ be a fuzzy planning problem and $\Pi_{\mathbf{FP}}$ be the normal logic program encoding of $\mathbf{FP}$. Then, the models of the SAT encoding of $\Pi_{\mathbf{FP}}$ are equivalent to valid trajectories in $\mathbf{FP}$.*

6 Conclusions and Related Work

We described a novel action language called fuzzy action language, $\mathcal{A}_F$, that allows the representation and reasoning about actions with fuzzy effects. In addition we introduced the notions of fuzzy planning and fuzzy plan in the fuzzy action language $\mathcal{A}_F$. Furthermore, we described a fuzzy planner based on the fuzzy action language A_F that is developed by translating a fuzzy planning problem, $\mathbf{FP}$, in $\mathcal{A}_F$ into a normal logic program with answer set semantics, $\Pi_{\mathbf{FP}}$, where trajectories in $\mathbf{FP}$ are equivalent to the answer sets of $\Pi_{\mathbf{FP}}$. In addition, we formally proved the correctness of our planner. Furthermore, we showed that a fuzzy planning problem can be encoded as a SAT problem.

The literature is rich with action languages that are capable of representing and reasoning about actions in the presence of probabilistic uncertainty, which include [2, 3, 5, 12, 14, 21, 22]. In [22], a probabilistic action language $\mathcal{P}$ is described that allows the representation of imperfect sensing actions (with probabilistic outcomes), non-sensing actions with probabilistic effects, the initial probability distribution over the possible initial states, and the indirect effects of actions. The action language $\mathcal{E}^+$ [12] allows sensing actions under the assumption that the agent's sensors are perfect, actions with probabilistic effects, and actions with non-deterministic effects. In

ISBN: 978-989-95079-6-8

addition, the semantics of $\mathcal{E}^+$ is based on description logic. Other high level probabilistic action description languages are described in [2, 5]. These languages are similar to $\mathcal{E}^+$ in the sense that they represent and reason about actions with probabilistic effects, except that they do not allow actions with non-deterministic effects or sensing actions. In [21], a high level action language called $\mathcal{A}_{MD}$ is presented that allows the factored representation and reasoning about Markov Decision Processes for reinforcement learning. In addition to the fact that $\mathcal{A}_F$ is a high level language, the major difference between $\mathcal{A}_F$ and these languages is that $\mathcal{A}_F$ represent and reason about actions under the presence of fuzzy uncertainty or imprecision. This is achieved in $\mathcal{A}_F$ by allowing description of actions with fuzzy effects, the executability conditions of actions, and the initial grade membership distribution of the initial states.

References

[1] C. Baral. *Knowledge representation, reasoning, and declarative problem solving*. Cambridge University Press, 2003.

[2] C. Baral, N. Tran, L. C. Tuan. *Reasoning about actions in a probabilistic setting*. In *AAAI*, 2002.

[3] C. Boutilier, T. Dean, and S. Hanks. Decision-theoretic planning: structural assumptions and computational leverage. *Journal of AI Research*, 11:1–94, 1999.

[4] C. da Costa Pereira and A. Tettamanzi. Reasoning about actions with imprecise and incomplete state descriptions. *Fuzzy Sets and Systems*, 160(10):13831401, 2009.

[5] T. Eiter and T. Lukasiewicz. Probabilistic reasoning about actions in nonmonotonic causal theories. In *19th Conference on Uncertainty in Artificial Intelligence*, 2003.

[6] T. Eiter et al. Declarative problem solving in dlv. In *Logic Based Artificial Intelligence*, 2000.

[7] M. Gelfond and V. Lifschitz. The stable model semantics for logic programming. *ICSLP*, 1988, MIT Press.

[8] M. Gelfond and V. Lifschitz. Classical negation in logic programs and disjunctive databases. *New Generation Computing*, 9(3-4):363-385, 1991.

[9] M. Gelfond and V. Lifschitz Representing action and change by logic programs. *Journal of Logic Programming*, 17:301–321, 1993.

[10] M. Gelfond and V. Lifschitz. Action languages. In *Electronic Transactions on AI*, 3(16), 1998.

[11] E. Giunchiglia and V. Lifschitz. An action language based on causal explanation: preliminary report. In *AAAI*, 1998.

[12] L. Iocchi, T. Lukasiewicz, D. Nardi, and R. Rosati. Reasoning about actions with sensing under qualitative and probabilistic uncertainty. In *16th European Conference on Artificial Intelligence*, 2004.

[13] H. Kautz and B. Selman. Pushing the envelope: planning, propositional logic, and stochastic search. In *13th National Conference on Artificial Intelligence*, 1996.

[14] N. Kushmerick, S. Hanks, and D. Weld. An algorithm for probabilistic planning. *Artificial Intelligence*, 76(1-2):239-286, 1995.

[15] J. Lee and V. Lifschitz. Describing additive fluents in action language C+. In *International Joint Conference in AI*, 2003.

[16] V. Lifschitz. Answer set planning. In *ICLP*, 1999.

[17] V. Lifschitz. Two components of an action language. *Annals of Mathematics and Artificial Intelligence*, 21:305–320, 1997.

[18] V. Lifschitz and W. Ren. A modular action description language. In *AAAI*, 2006.

[19] F. Lin and Y. Zhao. ASSAT: Computing answer sets of a logic program by SAT solvers. *Artificial Intelligence*, 157(1-2):115-137, 2004.

[20] I. Niemela and P. Simons. Efficient implementation of the well-founded and stable model semantics. *In Joint International Conference and Symposium on Logic Programming*, 289-303, 1996.

[21] E. Saad. A logical framework to reinforcement learning using hybrid probabilistic logic programs. In *Second International Conference on Scalable Uncertainty Management*, 2008.

[22] E. Saad. Probabilistic planning with imperfect sensing actions using hybrid probabilistic logic programs. In *Ninth International Workshop on Computational Logic in Multi-Agent Systems (CLIMA'08)*, 2008.

[23] A. Saffiotti, K. Konolige, and E. Ruspini. A Multivalued logic approach to integrating planning and control. *Artificial Intelligence*, 76(1-2): 481–526, 1995.

[24] T. Son and C. Baral. Formalizing sensing actions - a transition function based approach. *Artificial Intelligence*, 125:(1-2): 19 - 91, 2001.

[25] T. Son, C. Baral, T. Nam, and S. McIlraith. Domain-dependent knowledge in answer set planning. *ACM Transactions on Computational Logic*, 7(4):613–657, 2006.

[26] V. S. Subrahmanian and C. Zaniolo. Relating stable models and AI planning domains. *In International Conference of Logic Programming*, 233-247, 1995.

[27] L. Zadeh. Fuzzy Sets. *Information and Control*, 8(3): 338 –353, 1965.

[28] L. Zadeh. Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Trans. on Systems, Man, and Cybernetics*, SMC-3:28–44, 1973.

ISBN: 978-989-95079-6-8