

Dynamics of Declarative Goals in Agent Programming

M. Birna van Riemsdijk Mehdi Dastani Frank Dignum
John-Jules Ch. Meyer

Institute of Information and Computing Sciences
Utrecht University
The Netherlands
{birna, mehdi, dignum, jj}@cs.uu.nl

Abstract. In this paper, the notion of declarative goals as used in agent programming is central. Declarative goals describe desirable states and are updated during the execution of an agent. These goal dynamics are analyzed by distinguishing and formalizing various notions of goal dropping and goal adoption. Furthermore, possible motivations for an agent to drop or adopt goals are identified. Based on these motivations, we define specific mechanisms for implementing dropping and adoption. We show how these mechanisms are related to the general definitions of dropping and adoption.

1 Introduction

An important concept in agent theory, agent logics and agent programming is the concept of a *goal*. In agent theory, goals are introduced to explain and specify an agent's (proactive) behavior. In this view, agents are assumed to have their own objectives, for the achievement of which they initiate behavior [21,14,3,7]. Various logics have been introduced to formalize the concept of goals and reasoning about goals [17,2]. In these logics, a goal is formalized as a set of states. What is important in these logics, is which conclusions can be drawn from the existence of a certain goal set, i.e. which other goals can and cannot be inferred, etc.

Many agent programming languages have been proposed to *implement* (represent and process) an agent's goals [19,8,7,16,1]. The way in which goals are dealt with varies from language to language. For example, different languages propose programming constructs that capture different aspects of the concept of a goal. Also, in some programming languages goals are interpreted in a procedural way as processes that need to be performed while in other programming languages goals are interpreted in a declarative way as states to be reached. In this paper, we are interested in this *declarative* interpretation of goals. Declarative goals have a number of advantages in agent programming. They for example provide for the possibility to decouple plan execution and goal achievement [20]. If a plan fails, the goal that was to be achieved by the plan remains in the goal

base of the agent. The agent can then for example select a different plan or wait for the circumstances to change for the better. Furthermore, agents can be implemented such that they can communicate about their goals [13]. Also, a representation of goals in agents enables reasoning about goal interaction [18].

Reasoning with goals is essential in agent logics as well as in agent programming. In agent logics, reasoning with mental attitudes such as beliefs and goals is usually (formally) done through the use of epistemic (or doxastic) logics [17,10]. The logics for beliefs are well developed and give many properties to the beliefs [12]. By contrast, the logical axioms used for reasoning about goals are only the D-axiom (taking care that goals are consistent) and the K-axiom (which makes it possible to combine goals).

The considerations above lead to the fact that reasoning about goals in the sense of making derivations, can be kept very limited and the logic to represent it simple. The biggest role of goals in agents is thus not the ability to reason about them, but their motivational power of generating behavior. This means that we are interested in the relations between goals on the one hand and beliefs and behavior of the agent on the other hand. A certain plan or behavior is generated *because* of a goal. It might disappear again when the goal disappears or maybe the goal disappears when there is no feasible plan to reach the goal. In this light, we are more interested in the *dynamics of goals* than in the reasoning about goals. In particular, we are concerned with questions such as when does an agent adopt or drop a goal, how long does the goal persist, etc.

This paper aims to analyze these dynamics of declarative goals in the context of agent programming. We will do this by distinguishing and formalizing various notions of goal dropping (section 3) and goal adoption (section 4). In these sections, also possible motivations for an agent to drop or adopt goals are identified. Based on these motivations, we define specific mechanisms for capturing dropping and adoption in agent programming languages. Furthermore, we show how these mechanisms are related to the general definitions of dropping and adoption. Finally, in section 5, we conclude the paper and discuss some future research.

2 Preliminaries

In order to facilitate discussion, we give a number of definitions. In the sequel, a language defined by inclusion shall be the smallest language containing the specified elements.

First, we define the notion of an agent configuration. An agent configuration consists of a belief base, a goal base, a plan and a set of rules as defined below.

Definition 1. (*agent configuration*) Let \mathcal{L} with typical element ϕ be a propositional language with negation and conjunction, let \mathbf{Plan} be a language of plans and let R be a set of rules¹. An agent configuration, typically denoted by c , then

¹ Agents will in general have multiple sets of rules of various types, such as rules to select or revise plans and rules to specify goal dynamics. In this paper, we will

is a tuple $\langle \sigma, \gamma, \pi, R \rangle$ where $\sigma \subseteq \mathcal{L}$ is the belief base, $\gamma \subseteq \mathcal{L}$ is the goal base, $\pi \in \text{Plan}$ is the plan² of the agent and R is a set of rules.

In the sequel, we will use σ_c , γ_c , π_c and R_c to denote respectively the belief base, the goal base, the plan and the set of rules of an agent configuration c .

This paper is based on the idea that an agent consists of data structures representing the agent's mental attitudes such as beliefs, goals and rules. Agents from the 3APL language family [11,19,8] are for example defined based on this view, but the ideas that are presented in this paper apply to any type of cognitive agent with similar mental attitudes.

During the execution of an agent, the mental attitudes of the agent can change through for example plan execution and rule application. It will often be the case that e.g. multiple rules are applicable in a certain configuration. The decision of which rule to apply, can then be made by the agent interpreter or so called deliberation cycle [6], for example based on a certain ordering of the rules.

Given an agent configuration, we are interested in the question whether the agent has certain beliefs and goals. For this reason, we introduce a belief and a goal language.

Definition 2. (*belief and goal formulas*) The belief formulas \mathcal{L}_B with typical element β and the goal formulas \mathcal{L}_G with typical element κ are defined as follows.

- if $\phi \in \mathcal{L}$, then $\mathbf{B}\phi \in \mathcal{L}_B$ and $\mathbf{G}\phi \in \mathcal{L}_G$,
- if $\beta, \beta' \in \mathcal{L}_B$ and $\kappa, \kappa' \in \mathcal{L}_G$, then $\neg\beta, \beta \wedge \beta' \in \mathcal{L}_B$ and $\neg\kappa, \kappa \wedge \kappa' \in \mathcal{L}_G$.

Below, we define a semantics for the belief and goal formulas, that we call the “initial” semantics. In the sequel, we will introduce various other semantics.

Definition 3. (*initial semantics of belief and goal formulas*) Let $\models_{\mathcal{L}}$ be an entailment relation defined for \mathcal{L} as usual, let $\phi \in \mathcal{L}$ and let $\langle \sigma, \gamma, \pi, R \rangle$ be an agent configuration. Let $\varphi \in \mathcal{L}_B \cup \mathcal{L}_G$. The initial semantics \models_0 of the belief and goal formulas is then as defined below.

$$\begin{aligned}
\langle \sigma, \gamma, \pi, R \rangle \models_0 \mathbf{B}\phi &\Leftrightarrow \sigma \models_{\mathcal{L}} \phi \\
\langle \sigma, \gamma, \pi, R \rangle \models_0 \mathbf{G}\phi &\Leftrightarrow \gamma \models_{\mathcal{L}} \phi \\
\langle \sigma, \gamma, \pi, R \rangle \models_0 \neg\varphi &\Leftrightarrow \langle \sigma, \gamma, \pi, R \rangle \not\models_0 \varphi \\
\langle \sigma, \gamma, \pi, R \rangle \models_0 \varphi_1 \wedge \varphi_2 &\Leftrightarrow \langle \sigma, \gamma, \pi, R \rangle \models_0 \varphi_1 \text{ and } \langle \sigma, \gamma, \pi, R \rangle \models_0 \varphi_2
\end{aligned}$$

In this paper, we assume the semantics of agent programming languages are defined in terms of a transition system [15]. A transition system is a set of derivation rules for deriving transitions. A transition is a transformation of one agent configuration into another and it corresponds to a single computation step.

however consider only one type of rule at the time, which is why it suffices to have only one set of rules in the agent configuration.

² For the purpose of this paper, an agent configuration could be defined without a plan component, as it will not be used in the definitions. We however include it for ease of possible extensions of the paper.

In the sequel, we use $c \rightarrow c'$ to indicate a transition from agent configuration c to c' . It will sometimes be useful to add a label, denoting the kind of transition, e.g. $c \rightarrow_l c'$.

The following definitions will be used in the sequel and are introduced for notational convenience. The first definition below specifies what we mean by an expansion or contraction of the beliefs of an agent with a certain formula. The second definition specifies two notions of a formula ϕ being a goal in a goal base γ , the first defined as membership of a set (modulo equivalence) and the second as entailment.

Definition 4. (*expansion and contraction of beliefs*) Let c, c' be agent configurations. Let $\phi \in \mathcal{L}$ and $\beta \in \mathcal{L}_B$. Then, we define respectively the notion of expanding the beliefs with ϕ or β , and contraction of the beliefs with ϕ or β over the transition $c \rightarrow c'$ as follows.

$$\begin{aligned} \text{expansion}_{\mathbf{B}}(\phi, c \rightarrow c') &\Leftrightarrow c \not\models \mathbf{B}\phi \text{ and } c' \models \mathbf{B}\phi \\ \text{expansion}_{\mathbf{B}}(\beta, c \rightarrow c') &\Leftrightarrow c \not\models \beta \text{ and } c' \models \beta \\ \text{contraction}_{\mathbf{B}}(\phi, c \rightarrow c') &\Leftrightarrow c \models \mathbf{B}\phi \text{ and } c' \not\models \mathbf{B}\phi \\ \text{contraction}_{\mathbf{B}}(\beta, c \rightarrow c') &\Leftrightarrow c \models \beta \text{ and } c' \not\models \beta \end{aligned}$$

Definition 5. (*ϕ is a goal in γ*) Let γ be a goal base and let $\phi \in \mathcal{L}$. We then define the following notions specifying when ϕ is a goal in γ : $\text{goal}_{\text{set}}(\phi, \gamma) \Leftrightarrow \exists \phi' \in \gamma : \phi' \equiv \phi$ and $\text{goal}_{\text{ent}}(\phi, \gamma) \Leftrightarrow \gamma \models_{\mathcal{L}} \phi$. Note that $\text{goal}_{\text{set}}(\phi, \gamma)$ implies $\text{goal}_{\text{ent}}(\phi, \gamma)$.

3 Goal Dropping

In this section, we consider possible reasons or motivations for an agent to drop a goal. The notion of goal dropping can be related to the level of commitment an agent has towards a goal. If the agent is not committed at all, it might for example drop its goals right after they are adopted. If the agent is very committed or even fanatic, it will not at all be inclined to abandon its goals. These various levels of commitment or the way in which a certain agent deals with goal abandonment, is often referred to as a *commitment strategy* for that agent [17]. Although in principle one could consider any level of commitment for agents, the common commitment strategies require some level of *persistence* of goals [20]. In sections 3.1, 3.2 and 3.3, we will describe two widely used strategies in some detail and discuss a few more possibilities (together with associated problems). Before we can go into a discussion on various commitment strategies however, we will first define the notion of goal dropping in general.

As we explained in section 2, the execution or semantics of an agent can be described in terms of transitions. The phenomenon of dropping a goal naturally involves a configuration change of some sort and goal dropping can thus be defined as a property of these transitions. Informally, a goal ϕ is dropped over a transition $c \rightarrow c'$, if ϕ is a goal in c , but not in c' . In order to be more precise about what we mean when we say that a goal is dropped, we first need to specify what it means that “ ϕ is a goal in a configuration”.

We distinguish two different notions of what we can consider to be a goal in an agent configuration. Firstly, a formula ϕ can be viewed as a goal in a configuration c if ϕ is in the goal base, i.e. $\phi \in \gamma_c$ ³. Secondly, a formula ϕ can be considered as a goal in c if the formula $\mathbf{G}\phi$ holds, i.e. $c \models \mathbf{G}\phi$ where \models is an entailment relation defined for \mathcal{L}_G . If $\mathbf{G}\phi$ is defined such that it holds if and only if $\phi \in \gamma_c$, these notions coincide. As we will however see in the sequel, this is usually not the case. Based on these two views on the goals of an agent, we now distinguish two perspectives on dropping, i.e. a so called *deletion perspective* and a *satisfaction perspective*. The first is based on the deletion of a goal from the goal base, whereas the second is based on the satisfaction of a formula $\mathbf{G}\phi$.

Definition 6. (*dropping, deletion perspective*) Let c, c' be agent configurations and let $c \rightarrow c'$ be a transition. Let $\phi \in \mathcal{L}$. Then, we define the notion of the goal ϕ being dropped over the transition $c \rightarrow c'$, denoted by $dropped_{del}(\phi, c \rightarrow c')$, as follows: $dropped_{del}(\phi, c \rightarrow c') \Leftrightarrow goal_{set}(\phi, \gamma_c)$ and $\neg goal_{set}(\phi, \gamma_{c'})$.

Definition 7. (*dropping, satisfaction perspective*) Let c, c' be agent configurations and let $c \rightarrow c'$ be a transition. Let \models be an entailment relation defined for \mathcal{L}_G and let $\phi \in \mathcal{L}$. Then, we define the notion of the goal ϕ being dropped over the transition $c \rightarrow c'$, denoted by $dropped_{sat}(\phi, c \rightarrow c')$, as follows: $dropped_{sat}(\phi, c \rightarrow c') \Leftrightarrow c \models \mathbf{G}\phi$ and $c' \not\models \mathbf{G}\phi$.

In the definition of dropping from a satisfaction perspective above, we assume an entailment relation \models , defined for \mathcal{L}_G . One such entailment relation is specified in definition 3 and in the sequel we will also define other entailment relations. However, in the definition of dropping from a satisfaction perspective, we want to abstract from these specific entailment relations and assume a relation \models .

3.1 Blind Commitment

An often mentioned and very intuitive reason for dropping a goal is, that the agent *believes to have achieved* the goal [17,5]. In [17], an agent that only drops its goals if believed to have achieved them, is called a blindly committed agent. An agent that also drops its goals if believed to be unachievable, is called a single minded agent.

A blindly committed agent should drop a goal ϕ if it comes to believe ϕ . An implementation of a blindly committed agent should thus be such that it drops a goal ϕ as soon as it comes to believe ϕ . This dropping can be approached from the two perspectives discussed above, i.e. we can specify the dropping of ϕ as deletion or as satisfaction. The dropping from a deletion perspective can be defined as a general constraint on the transition systems that can be specified for blindly committed agents.

Definition 8. (*blind commitment, deletion perspective*) Let c, c' be agent configurations and let $\phi \in \mathcal{L}$. An agent is then blindly committed iff $\forall c \rightarrow c' : [(\exists \phi : expansion_{\mathbf{B}}(\phi, c \rightarrow c')) \Rightarrow (\gamma_{c'} = \gamma_c \setminus \{\phi \mid \sigma_c \models_{\mathcal{L}} \phi\})]$ where $c \rightarrow c'$ is a transition that can be derived in the transition system for the agent.

³ Possibly modulo equivalence: ϕ is a goal in γ_c iff $goal_{set}(\phi, \gamma_c)$, i.e. $\exists \phi' \in \gamma_c : \phi' \equiv \phi$.

The following proposition relates the definition of a blindly committed agent above, to the general definition of dropping from a deletion perspective.

Proposition 1. (*Goals are dropped from a deletion perspective once the agent believes they are achieved.*) If, for a blindly committed agent as specified in definition 8, an expansion with ϕ takes place over a transition $c \rightarrow c'$ and ϕ is a goal in γ_c , then ϕ is dropped over this transition from a deletion perspective, i.e.: if $expansion_{\mathbf{B}}(\phi, c \rightarrow c')$ and $goal_{set}(\phi, \gamma_c)$ then $dropped_{del}(\phi, c \rightarrow c')$.

Besides taking the deletion perspective on blind commitment, we can also approach this issue from a satisfaction perspective. In order to do this, we extend the semantics for belief and goal formulas of definition 3, specifying that $\mathbf{G}\phi$ holds if and only if ϕ follows from the goal base *and* ϕ does not follow from the belief base.

Definition 9. (*blind commitment, satisfaction perspective*) Let $\phi \in \mathcal{L}$ and let $\langle \sigma, \gamma, \pi, R \rangle$ be an agent configuration. The semantics \models_s of the belief and goal formulas for a blindly committed agent is then as defined below⁴.

$$\langle \sigma, \gamma, \pi, R \rangle \models_s \mathbf{G}\phi \Leftrightarrow \gamma \models_{\mathcal{L}} \phi \text{ and } \sigma \not\models_{\mathcal{L}} \phi$$

From the definition above, we can derive that $\models_s \mathbf{B}\phi \rightarrow \neg\mathbf{G}\phi$ is a validity, i.e. $\mathbf{G}\phi$ cannot hold if ϕ is believed. This implies, that if an agent comes to believe ϕ over a transition, a goal ϕ is dropped from a satisfaction perspective (assuming that ϕ was a goal before the transition). This is formulated in the following proposition.

Proposition 2. (*Goals are dropped from a satisfaction perspective once the agent believes they are achieved.*) If the semantics of belief and goal formulas of an agent is as specified in definition 9 and an expansion with ϕ takes place over a transition $c \rightarrow c'$ and ϕ is a goal in γ_c , then ϕ is dropped over this transition from a satisfaction perspective, i.e.: if $expansion_{\mathbf{B}}(\phi, c \rightarrow c')$ and $goal_{ent}(\phi, \gamma_c)$ then $dropped_{sat}(\phi, c \rightarrow c')$.

We can conclude that blindly committed agents can relatively easily be specified in terms of goals and beliefs of the agents. However, the strategy seems very limited and not very realistic. In the literature often agent commitment strategies are discussed that are a bit looser on the commitment, which means that an agent could also drop its goal if it *believes that it is unachievable* [17,5]. We will discuss this strategy at the end of this section.

3.2 Failure Condition

The conditions for dropping a goal can be seen as a kind of *failure condition* on the goal achievement. For blindly committed agents, the failure condition is that

⁴ The clauses for belief formulas, negation and conjunction are as in definition 3, but we do not repeat them here or in definitions in the sequel, for reasons of space.

the agent already believes the goal is true. In [20], Winikoff et al. also consider the specification of more specific failure conditions for goals. The idea is, that this condition specifies an explicit reason for the agent to drop the goal, i.e. if the failure condition becomes true, the agent drops its goal. This failure condition is thus specific to a certain goal.

The authors do not elaborate on the intuitions behind this failure condition, but one could imagine specifying a condition which, once true, will never become false again and which falsehood is necessary for the agent to be able to achieve the goal. Suppose for example that agent A has a goal to have a certain egg sunny side up and suppose A comes to believe that the egg is scrambled, then this would be reason for A to drop its goal, as a scrambled egg can never be prepared sunny side up. The failure condition for a goal should thus correspond to a situation from which the agent will never be able to achieve the goal. This situation is however specified by the designer of the agent. The designer for example knows that a scrambled egg cannot be transformed into one that is prepared sunny side up. The reasoning is thus done at design time by the agent developer instead of leaving it up to the agent itself.

In order to implement this idea of specifying a failure condition for a goal, we propose a so called failure rule. This is a rule with a condition on beliefs as the head and a goal (being a propositional formula) as the body. The informal reading is, that the goal in the body can be dropped if the condition in the head holds.

Definition 10. (*failure rule*) The set of failure rules \mathcal{R}_f is defined as follows: $\mathcal{R}_f = \{\beta \Rightarrow_{\mathbf{G}} \phi \mid \beta \in \mathcal{L}_B, \phi \in \mathcal{L}\}$.

The interpretation of failure rules can be approached from the two perspectives on goal dropping we identified. We first define the semantics of this rule from a deletion perspective, resulting in the deletion of a goal from the goal base if the rule is applied⁵.

Definition 11. (*failure rule semantics, deletion perspective*) Let \mathcal{R}_f be the set of failure rules of definition 10 and let $R_f \subseteq \mathcal{R}_f$. Let $f = (\beta \Rightarrow_{\mathbf{G}} \phi) \in R_f$ and let \models be an entailment relation defined for \mathcal{L}_B . The semantics of applying this rule is then as follows, where $\gamma' = \gamma \setminus \{\phi' \mid \phi' \equiv \phi\}$:

$$\frac{\langle \sigma, \gamma, \pi, R_f \rangle \models \beta \text{ and } goal_{set}(\phi, \gamma)}{\langle \sigma, \gamma, \pi, R_f \rangle \rightarrow_{apply(f)} \langle \sigma, \gamma', \pi, R_f \rangle}$$

The following proposition relates the semantics of failure rule application above, to the general definition of dropping from a deletion perspective.

Proposition 3. (*Applying a failure rule results in dropping from a deletion perspective.*) If $c \rightarrow_{apply(f)} c'$ where $f = (\beta \Rightarrow_{\mathbf{G}} \phi)$ is a transition derived using the transition rule of definition 11, then $dropped_{del}(\phi, c \rightarrow_{apply(f)} c')$ holds.

⁵ Note that a blindly committed agent could be specified in terms of failure rules of the form $\mathbf{B}\phi \Rightarrow_{\mathbf{G}} \phi$.

The semantics of failure rule application that is defined above, takes an operational view on failure rules. Another option is using these rules to define, in a declarative way, the goals of an agent as the satisfaction of a formula $\mathbf{G}\phi$ in a configuration. This is done in the following definition that extends definition 9, specifying that $\mathbf{G}\phi$ holds if and only if ϕ follows from the goal base, ϕ is not believed *and* there cannot be a rule which head holds and which body is equivalent to ϕ .

Definition 12. (*failure rule semantics, satisfaction perspective*) Let \mathcal{R}_f be the set of failure rules of definition 10 and let $R_f \subseteq \mathcal{R}_f$. Let $\phi \in \mathcal{L}$ and let $\langle \sigma, \gamma, \pi, R_f \rangle$ be an agent configuration. The semantics \models_f of the belief and goal formulas in the presence of failure rules is then as defined below.

$$\langle \sigma, \gamma, \pi, R_f \rangle \models_f \mathbf{G}\phi \Leftrightarrow \gamma \models_{\mathcal{L}} \phi \text{ and } \sigma \not\models_{\mathcal{L}} \phi \text{ and} \\ \neg \exists f \in R_f : (f = (\beta \Rightarrow_{\mathbf{G}} \phi') \text{ and } \langle \sigma, \gamma, \pi, R_f \rangle \models_f \beta \\ \text{and } \phi' \equiv \phi)$$

From the definition above, we can conclude that $\mathbf{G}\phi$ cannot hold in a configuration if there is a rule $\beta \Rightarrow_{\mathbf{G}} \phi'$ in this configuration such that $\phi' \equiv \phi$ and such that β holds. This implies, that if an agent comes to believe β over a transition, i.e. if the rule is “activated” over this transition, the goal ϕ is dropped from a satisfaction perspective (assuming that ϕ was a goal before the transition). We formulate this in the proposition below, after first defining the notion of rule activation⁶.

Definition 13. (*rule activation*) Let $f = (\beta \Rightarrow_{\mathbf{G}} \phi) \in R_f$ be a failure rule, let c, c' be configurations with ruleset R_f and let $c \rightarrow c'$ be a transition. The rule f is activated over the transition, denoted by $activated(f, c \rightarrow c')$, iff $expansion_{\mathbf{B}}(\beta, c \rightarrow c')$, i.e. if the rule’s head is false in c and true in c' .

Proposition 4. (*If a failure rule is activated over a transition, the goal associated with that rule is dropped from a satisfaction perspective.*) If the semantics of belief and goal formulas of an agent is as specified in definition 12 and a failure rule $f = (\beta \Rightarrow_{\mathbf{G}} \phi)$ is activated over a transition $c \rightarrow c'$ and ϕ is a goal in γ_c , then ϕ is dropped from a satisfaction perspective over this transition, i.e.: if $activated(f, c \rightarrow c')$ and $goal_{set}(\phi, \gamma_c)$ then $dropped_{sat}(\phi, c \rightarrow c')$.

3.3 Other Strategies

In the previous two sections we discussed two widely used strategies for dropping goals. Both strategies can be implemented in a rather straightforward way. Theoretically, one can of course have far more commitment strategies. We already mentioned the single minded commitment strategy. However, implementing a

⁶ Note that we use the term “activation” of a rule over a transition to indicate that the antecedent becomes true over this transition. The rule is not activated from the outside.

single minded agent is much more difficult. The condition stating that the agent does not believe a goal ϕ to be achievable, could be specified using CTL temporal logic [4] by the following formula: $\mathbf{B}(\neg\mathbf{EF}\ \phi)$, i.e. the agent believes that there is no possible course of future events in which ϕ is eventually true. In order to evaluate this formula however, the agent would have to reason about its possible future execution traces. In general it is very difficult to check this formula, but one could approximate it in several ways, e.g. by only considering future traces up to a certain length, or by considering only traces generated by possible plans of the agent. In whichever way the strategy is approximated though, the agent needs a mechanism to reason with temporal aspects, thus complicating the implementation considerably.

A last commitment strategy to be mentioned here is the open minded strategy. This strategy states that a goal is dropped whenever the motivation for having that goal has gone. This is directly related to the issue of goal adoption. To implement this strategy, we should keep track of why a goal is adopted, i.e. which are the conditions for adopting a goal. Whenever these conditions are no longer true, the goal will be dropped, e.g. if a goal is adopted to go to New York in order to attend an AAMAS workshop and the workshop is cancelled, we can drop the goal to go to New York (even though we might still believe it is possible to go there and we are not there yet). We will briefly get back to this in section 4.1.

4 Goal Adoption

The issue of goal adoption can be subdivided into the questions of *when* to start considering to adopt goals and *which* goals are to be adopted. Regarding the first question, a possible motivation for an agent to start adopting goals could for example be the lack of goals or the lack of appropriate plans for the goals it has. If we assume that agents generate behavior because they have goals, situations like these would call for goal adoption to prevent an agent from being idle. The decision of when to start adopting goals could be specified in the interpreter or deliberation cycle of the agent (see section 2). In this paper, we will focus on the second question.

As for goal dropping, we also distinguish two perspectives on goal adoption, i.e. an *addition perspective* and a *satisfaction perspective*. The first is based on the addition of a goal to the goal base, whereas the second is again based on the satisfaction of a formula $\mathbf{G}\phi$.

Definition 14. (*adoption, addition perspective*) Let c, c' be agent configurations and let $c \rightarrow c'$ be a transition. Let $\phi \in \mathcal{L}$. Then, we define the notion of the goal ϕ being adopted over the transition $c \rightarrow c'$, denoted by $adopted_{add}(\phi, c \rightarrow c')$, as follows: $adopted_{add}(\phi, c \rightarrow c') \Leftrightarrow \neg goal_{set}(\phi, \gamma_c)$ and $goal_{set}(\phi, \gamma_{c'})$.

Definition 15. (*adoption, satisfaction perspective*) Let c, c' be agent configurations and let $c \rightarrow c'$ be a transition. Let \models be an entailment relation defined for \mathcal{L}_G and let $\phi \in \mathcal{L}$. Then, we define the notion of the goal ϕ being

adopted over the transition $c \rightarrow c'$, denoted by $adopted_{sat}(\phi, c \rightarrow c')$, as follows: $adopted_{sat}(\phi, c \rightarrow c') \Leftrightarrow c \not\models \mathbf{G}\phi$ and $c' \models \mathbf{G}\phi$.

In this section, we discuss important motivations for goal adoption that have been identified in the literature. We distinguish reasons for adoption based on *motivational attitudes* such as desires and norms (section 4.1), and reasons based on the notion of *subgoals* (section 4.2). Based on this analysis, we sketch mechanisms for dealing with goal adoption, such as explicit goal adoption rules. We believe it is important to analyze possible motivations for goal adoption, as different motivations may lead to different kinds of rules or other goal adoption mechanisms.

Goal adoption rules have been proposed before in for example research on 3APL [8] and BOID [7]. However, in each of these languages the focus is on one type of interpretation of the rules. 3APL for example interprets rules from an addition perspective, whereas BOID takes the satisfaction point of view. We believe that the observation that there are different interpretations of rules is important, in order to be able to identify conditions under which these perspectives are equivalent or differ. Although we do not provide this kind of analysis of similarities and differences in this paper, we take a first step towards this by identifying and defining the different perspectives.

4.1 Internal and External Motivations for Goal Adoption

In this section, we distinguish important internal and external motivations for goal adoption. As internal motivations, we will discuss so called abstract goals and desires, and as external motivations we will discuss obligations, norms and communication. After a general discussion on these motivations, we will propose a goal adoption rule to implement these ideas.

Motivations In [9], Dignum and Conte discuss the *generation of concrete goals from built-in abstract goals* as an internal motivation for adopting goals. As Dignum and Conte put it, these abstract goals are often not really achievable but can be approximated through concrete goals. An abstract goal could for example be to be social or to be a law abiding agent. The concrete goal of not driving above the speed limit, would then for example contribute to being a law abiding agent.

Other important sources that may cause the generation of new goals for an agent are *desires, norms and obligations* of the agent. In general, desires are considered as agents' internal motivational attitude while norms and obligations are classified as external motivational attitudes. An agent's desires represent its preferences, wants and urges. They may be produced by emotional or affective processes or even by biological survival mechanisms. For example, if an agent is without food for some period, this might produce an acute desire for food. Desire may also be long-term preferences or wants such as being rich. Such long term preferences can be triggered by an observation, belief, or communication

through which they are turned into goals, i.e. desires can be viewed as goals that are conditionalized by beliefs, etc.

The norms and obligations represent the social nature of agents or what agents have to adhere to. One might have very dutiful agents that generate a goal for any obligation they incur. In general, the norms that an agent wants to adhere to are rules of conduct that pertain in the society in which the agent operates. These could be represented through abstract goals that state that the agent tries to satisfy an obligation or adhere to a norm.

Agents usually operate in a multi-agent environment and have the ability to *communicate* with other agents. They do not only communicate knowledge or belief about the world, but they can also communicate requests for achieving goals. If an agent decides to comply with a request to achieve a goal, the request triggers the generation of a goal.

Formalisation In order to implement these reasons for goal adoption, we propose a goal adoption rule. This is a rule with a condition on abstract goals, beliefs and/or communicated formulas as the head, and a goal (being a propositional formula) as the body. The informal reading is, that the goal in the body can be adopted if the condition in the head holds. In order to define the semantics of these rules, we need to extend agent configurations, adding an abstract goal set and a set of communicated formulas.

Definition 16. (*extended agent configuration*) Let \mathcal{A} be a set of abstract goals consisting of abstract goal names and let \mathcal{L}_C be a set of communication formulas. Let $\langle \sigma, \gamma_{concr}, \pi, R \rangle$ be an agent configuration. An extended agent configuration is then a tuple $\langle \sigma, \gamma, \pi, R \rangle$ where γ is a tuple $\langle \alpha, \gamma_{concr}, \gamma_{comm} \rangle$ with $\alpha \subseteq \mathcal{A}$ is the abstract goal base and $\gamma_{comm} \subseteq \mathcal{L}_C$ are the communicated formulas.

Definition 17. (*goal adoption rules*) We assume a set of abstract goals \mathcal{A} consisting of abstract goal names and we assume a set of communication formulas \mathcal{L}_C . The set of goal adoption rules \mathcal{R}_a is then defined as follows:

$$\mathcal{R}_a = \{h \Rightarrow_{\mathbf{G}}^+ \phi \mid h = h_1, \dots, h_n \text{ with } h_i \in (\mathcal{A} \cup \mathcal{L}_B \cup \mathcal{L}_C)\}.$$

Definition 18. (*semantics of goal adoption rule head*) Let $e = \langle \sigma, \gamma, \pi, R \rangle$ be an extended agent configuration with $\gamma = \langle \alpha, \gamma_{concr}, \gamma_{comm} \rangle$ and let $a \in \mathcal{A}$. We then define an entailment relation for abstract goals as follows: $e \models_{\mathcal{A}} a \Leftrightarrow a \in \alpha$. We furthermore assume an entailment relation $\models_{\mathcal{L}_C}$ for the language of communication formulas. The entailment relation for the set of formulas $\mathcal{A} \cup \mathcal{L}_B \cup \mathcal{L}_C$ is then denoted as $\models_{\mathcal{A} \cup \mathcal{L}_B \cup \mathcal{L}_C}$. Let h_1, \dots, h_n be the head of a goal adoption rule. The entailment relation \models_H for rule heads is then as follows.

$$\begin{aligned} \langle \sigma, \gamma, \pi, R \rangle \models_H h_1, \dots, h_n \Leftrightarrow & \quad \langle \sigma, \gamma, \pi, R \rangle \models_{\mathcal{A} \cup \mathcal{L}_B \cup \mathcal{L}_C} h_1 \text{ and} \\ & \quad \vdots \\ & \quad \text{and } \langle \sigma, \gamma, \pi, R \rangle \models_{\mathcal{A} \cup \mathcal{L}_B \cup \mathcal{L}_C} h_n \end{aligned}$$

As for failure rules, we define an operational as well as a declarative semantics of the goal adoption rule. This results in semantics from an addition and a satisfaction perspective as also indicated by the propositions below.

Definition 19. (*goal adoption rule semantics, addition perspective*) Let $R_a \subseteq \mathcal{R}_a$ be a set of goal adoption rules. Let $a = (h \Rightarrow_{\mathbf{G}}^+ \phi) \in R_a$. The semantics of applying this rule is then as follows, where $\gamma' = \gamma \cup \{\phi\}$:

$$\frac{\langle \sigma, \gamma, \pi, R_a \rangle \models_H h}{\langle \sigma, \gamma, \pi, R_a \rangle \rightarrow_{\text{apply}(a)} \langle \sigma, \gamma', \pi, R_a \rangle}$$

Proposition 5. (*Applying a goal adoption rule results in adoption from an addition perspective.*) If $c \rightarrow_{\text{apply}(a)} c'$ where $a = (h \Rightarrow_{\mathbf{G}}^+ \phi)$ is a transition derived using the transition rule of definition 19 and ϕ is not a goal in γ_c , i.e. $\neg \text{goal}_{\text{set}}(\phi, \gamma_c)$, then $\text{adopted}_{\text{add}}(\phi, c \rightarrow_{\text{apply}(a)} c')$ holds.

Definition 20. (*goal adoption rule semantics, satisfaction perspective*) Let \mathcal{R}_a be the set of goal adoption rules and let $R_a \subseteq \mathcal{R}_a$. The semantics \models_a for belief and goal formulas in the presence of goal adoption rules is then as follows.

$$\langle \sigma, \gamma, \pi, R_a \rangle \models_a \mathbf{G}\phi \Leftrightarrow (\gamma \models_{\mathcal{L}} \phi \text{ or } \exists a \in R_a : (a = (h \Rightarrow_{\mathbf{G}}^+ \phi') \text{ and } \langle \sigma, \gamma, \pi, R_a \rangle \models_H h \text{ and } \phi' \equiv \phi)) \text{ and } \sigma \not\models_{\mathcal{L}} \phi$$

Proposition 6. (*If a goal adoption rule is activated over a transition, the goal associated with that rule is adopted from a satisfaction perspective.*) If the semantics of belief and goal formulas of an agent is as specified in definition 20 and a goal adoption rule $a = (h \Rightarrow_{\mathbf{G}}^+ \phi)$ is activated over a transition $c \rightarrow c'$ and ϕ is not a goal in c , then ϕ is adopted from a satisfaction perspective over this transition, i.e.: if $\text{activated}(a, c \rightarrow c')$ and $c \not\models_a \mathbf{G}\phi$ then $\text{adopted}_{\text{sat}}(\phi, c \rightarrow c')$.

Note that if a goal adoption rule is deactivated over a transition, the goal in the consequent *could* be dropped over this transition due to this deactivation, provided that no other adoption rule has this goal as its consequent. This phenomenon could thus be considered an implementation of the open minded commitment strategy (section 3.3).

4.2 Subgoal Adoption

A goal can be viewed as a *subgoal* if its achievement brings the agent “closer” to its topgoal. This notion of “closeness” to a topgoal is rather vague. One could argue that the achievement of a concrete goal contributing to an abstract goal, brings the agent closer to this abstract goal. A concrete goal can thus be viewed as a subgoal of an abstract goal. In this section, we distinguish two other views on subgoals, i.e. subgoals as being the “parts” of which a topgoal is composed and subgoals as landmarks or states that should be achieved on the road to achieving a topgoal. As we see it, these different kinds of subgoals can lead to different goal adoption mechanisms.

Goal Decomposition A decomposition of a goal into subgoals should be such, that the achievement of all subgoals at the same time implies achievement of the topgoal. The goal $p \wedge q$ could for example be decomposed into the subgoals p and q . Achievement of both p and q at the same time, now implies achievement of $p \wedge q$.

Goal decomposition is most naturally reached through defining the semantics of goal formulas like was done in definition 9, i.e. such that $\mathbf{G}\phi$ holds if ϕ is a logical consequence of the goal base. In this way, if for example $p \wedge q$ is a goal in the goal base, $\mathbf{G}p$ will hold and $\mathbf{G}q$ will hold (assuming both p and q are not believed). We define the notion of a goal being a subgoal of another goal as follows: a goal ϕ' is a subgoal of ϕ , iff $\phi \models_{\mathcal{L}} \phi'$ but $\phi' \not\models_{\mathcal{L}} \phi$, which we will denote by $subgoal(\phi', \phi)$.

In the following proposition, we state that under the semantics of belief and goal formulas of definition 9, we can get subgoal adoption over a transition if the subgoal was achieved before the transition, but not anymore after the transition (assuming that the topgoal remains in the goal base).

Proposition 7. (*Subgoals are adopted from a satisfaction perspective once the agent believes they are not achieved anymore.*) If the semantics of belief and goal formulas of an agent is as specified in definition 9 and ϕ' is a subgoal of ϕ and contraction with ϕ' takes place over a transition $c \rightarrow c'$ and ϕ is a goal in γ_c as well as in $\gamma_{c'}$, then the subgoal ϕ' is adopted from a satisfaction perspective over this transition, i.e.: if $subgoal(\phi', \phi)$ and $contraction_{\mathbf{B}}(\phi', c \rightarrow c')$ and $goal_{set}(\phi, \gamma_c)$ and $goal_{set}(\phi, \gamma_{c'})$ then $adopted_{sat}(\phi', c \rightarrow c')$.

Landmarks The second view on subgoals we discuss in this section, is as landmarks. If an agent for example believes that it is in Utrecht and has the topgoal to be in New York (and has a ticket for a flight to New York etc.), then a subgoal would be to be at Schiphol airport. This subgoal does not contribute to the topgoal in the sense that concrete goals contribute to abstract goals. Achievement of the subgoal neither implies in some way achievement of the topgoal (together with achievement of other subgoals for example) and it is thus different from subgoals generated through decomposition.

It is important for an agent to be able to adopt landmark goals, because it can be the case that the agent only has plans to get from landmark to landmark. It can for example be the case that the agent has a plan in its library to get from Utrecht to Schiphol and that it has another plan to get from Schiphol to New York, i.e. the second plan is only applicable if the agent is at Schiphol. If the agent now believes that it is in Utrecht and it has the goal to be in New York, it does not have an applicable plan to execute. If however the agent can adopt the goal to be at Schiphol from the goal to be in New York and the knowledge that it has a plan to get to New York from Schiphol and possibly the belief to be in Utrecht, it *can* execute an applicable plan.

The adoption of landmark subgoals could be implemented in various ways. One possibility is the introduction of a goal adoption rule as below, through

which a goal can be adopted on the basis of beliefs and other goals. The semantics can be defined analogously to that of the adoption rule of definition 17.

Definition 21. (*landmark adoption rule*) The set of landmark adoption rules \mathcal{R}_l is defined as follows: $\mathcal{R}_l = \{\beta, \kappa \Rightarrow_{\mathbf{G}}^+ \phi \mid \beta \in \mathcal{L}_B, \kappa \in \mathcal{L}_G, \phi \in \mathcal{L}\}$.

Note that this formalisation does not record any structure or order among the landmarks that are adopted.

We will mention two other ways to adopt landmark goals. Due to space limitations however, we cannot elaborate on these. A first possibility could be to use plan specifications, indicating the preconditions under which the plan could be executed and the desired or expected postconditions. If the agent then has the postcondition of a plan as a goal and does not believe the precondition to be the case, it could adopt the precondition as a goal. If it then achieves this goal or precondition, it can execute the plan and reach its initial goal.

Secondly, one could consider the definition of a goal adoption statement in an agent's plans, similar to achievement goals in AgentSpeak(L) [16]. The goal in the goal adoption statement can be viewed as a subgoal of the plan at hand and the goal can be adopted if the statement is executed. Another possible interpretation of such a goal achievement statement could be, that this goal state should be achieved before proceeding with the rest of the plan. A plan will have to be selected for the specified goal. Plans with these kinds of statements could thus be viewed as partial plans, the goal achievement statements of which will need to be refined into plans.

5 Conclusion and Future Research

In agent programming languages, goals are often considered in a procedural way. In most agent specification logics on the other hand, goals are employed in a declarative way. We maintain that declarative goals are interesting and useful not only in agent specification, but also in agent programming. In this paper we have particularly explored the issue of the dynamics of declarative goals in the context of agent programming. That is to say, we have analyzed several motivations and mechanisms for dropping and adopting declarative goals in a fairly general setting. We believe this distinction between dropping and adoption and also the distinction between the different perspectives on these phenomena are important in order to get a better understanding of declarative goal dynamics. We have thus provided a basis for analyzing this phenomenon, but many issues were not addressed and remain for future research.

Most importantly, we did not discuss the relation between the two perspectives on dropping and adoption we defined. It will need to be investigated under what circumstances these notions are equivalent or yield similar agent behavior with respect to goal dynamics. Under most entailment relations for goal formulas, it will for example be the case that if a goal ϕ is adopted from an addition perspective, ϕ is also adopted from a satisfaction perspective (assuming a belief expansion with ϕ does not take place and assuming that ϕ does not follow

from the goal base before the adoption). Also, it is important to establish the advantages and disadvantages of both approaches and investigate whether they can or should be combined. A possible disadvantage for example concerns the interpretation of goal adoption rules from a satisfaction perspective, as this interpretation could diminish goal persistency: these rules can be activated and deactivated again over a series of transitions. This could result in the repeated adoption and dropping of a certain goal, which could be considered undesirable.

Another issue for future research has to do with the semantics of goal formulas in the presence of dropping or adoption rules. We took a rather conservative approach, defining that only formulas equivalent to the goal in the body of the rules can be dropped or adopted (definitions 12 and 20). One could also consider for example dropping logical consequences of the goal in the body of the failure rule, or combining applicable adoption rules by defining that logical consequences of the set of goals in the bodies of applicable rules can be adopted. Moreover, we did not discuss interactions between rules for dropping and adoption.

Furthermore, we did not discuss goal consistency. Goals are often assumed or required to be consistent [20] as it is argued that it is not rational for an agent to pursue conflicting objectives. This requirement has implications for goal adoption, as goals could become inconsistent through adoption. The issue could for example be dealt with like is done in BOID [7]. In this framework, the rules are interpreted as default rules from which (consistent) extensions or goal sets can be calculated. In the language GOAL [11], individual goals in the goal base are required to be consistent, rather than the entire goal base. This has implications for the definition of the semantics of goal formulas, as it will need to be defined in terms of individual goals rather than in terms of the goal base as a whole.

Finally, we mention goal revision. It seems natural that goal revision can be characterized in terms of dropping and adoption. One could however imagine that motivations for goal revision are different from those for dropping and adoption, possibly calling for a separate treatment of this issue. Also the relation with belief revision should be investigated in order to identify whether results from this field can be applied to goal revision.

References

1. F. Bellifemine, A. Poggi, G. Rimassa, and P. Turci. An object oriented framework to realize agent systems. In *Proceedings of WOA 2000 Workshop*, pages 52–57. WOA, may 2000.
2. C. Boutilier. Toward a logic for qualitative decision theory. In *Proceedings of the KR'94*, pages 75–86, 1994.
3. J. Broersen, M. Dastani, J. Hulstijn, and L. van der Torre. Goal generation in the BOID architecture. *Cognitive Science Quarterly*, 2(3-4):428–447, 2002.
4. E. M. Clarke and E. A. Emerson. Design and synthesis of synchronization skeletons using branching-time temporal logic. In *Logic of Programs, Workshop*, pages 52–71. Springer-Verlag, 1982.
5. P. R. Cohen and H. J. Levesque. Intention is choice with commitment. *Artificial Intelligence*, 42:213–261, 1990.

6. M. Dastani, F. S. de Boer, F. Dignum, and J.-J. Ch. Meyer. Programming agent deliberation – an approach illustrated using the 3APL language. In *Proceedings of the second international joint conference on autonomous agents and multiagent systems (AAMAS'03)*, pages 97–104, Melbourne, 2003.
7. M. Dastani and L. van der Torre. Programming BOID-Plan agents: deliberating about conflicts among defeasible mental attitudes and plans. In *Proceedings of The Third Conference on Autonomous Agents and Multi-agent Systems (AAMAS'04)*, New York, USA, 2004. To appear.
8. M. Dastani, M. B. van Riemsdijk, F. Dignum, and J.-J. Ch. Meyer. A programming language for cognitive agents: goal directed 3APL. In *First Workshop on Programming Multiagent Systems (ProMAS'03)*. 2003.
9. F. Dignum and R. Conte. Intentional agents and goal formation. In *Agent Theories, Architectures, and Languages*, pages 231–243, 1997.
10. F. Dignum, D. Kinny, and L. Sonenberg. From desires, obligations and norms to goals. *Cognitive Science Quarterly*, 2(3-4):407–430, 2002.
11. K. V. Hindriks, F. S. de Boer, W. van der Hoek, and J.-J. Ch. Meyer. Agent programming with declarative goals. In *Intelligent Agents VI - Proceedings of the 7th International Workshop on Agent Theories, Architectures, and Languages (ATAL'2000)*, Lecture Notes in AI. Springer, Berlin, 2001.
12. J.-J. C. Meyer and W. van der Hoek. *Epistemic logic for AI and computer science*. Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, Cambridge, 1995.
13. A. F. Moreira, R. Vieira, and R. H. Bordini. Extending the operational semantics of a BDI agent-oriented programming language for introducing speech-act based communication. In *First International Workshop on Declarative Agent Languages and Technologies (DALT03)*, pages 129–145, 2003.
14. A. Newell. The knowledge level. *Artificial Intelligence*, 18(1):87–127, 1982.
15. G. Plotkin. A structural approach to operational semantics. Technical report, Aarhus University, Computer Science Department, 1981.
16. A. S. Rao. AgentSpeak(L): BDI agents speak out in a logical computable language. In W. van der Velde and J. Perram, editors, *Agents Breaking Away (LNAI 1038)*, pages 42–55. Springer-Verlag, 1996.
17. A. S. Rao and M. P. Georgeff. Modeling rational agents within a BDI-architecture. In J. Allen, R. Fikes, and E. Sandewall, editors, *Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning (KR'91)*, pages 473–484. Morgan Kaufmann, 1991.
18. J. Thangarajah, L. Padgham, and M. Winikoff. Detecting and exploiting positive goal interaction in intelligent agents. In *Proceedings of the second international joint conference on autonomous agents and multiagent systems (AAMAS'03)*, pages 401–408, Melbourne, 2003.
19. M. B. van Riemsdijk, W. van der Hoek, and J.-J. Ch. Meyer. Agent programming in Dribble: from beliefs to goals using plans. In *Proceedings of the second international joint conference on autonomous agents and multiagent systems (AAMAS'03)*, pages 393–400, Melbourne, 2003.
20. M. Winikoff, L. Padgham, J. Harland, and J. Thangarajah. Declarative and procedural goals in intelligent agent systems. In *Proceedings of the eighth international conference on principles of knowledge representation and reasoning (KR2002)*, Toulouse, 2002.
21. M. Wooldridge. *An introduction to multiagent systems*. John Wiley and Sons, LTD, West Sussex, 2002.