

Changing Commitments Based on Reasons and Assumptions

Diana Marosin^{1,2}(✉) and Leendert van der Torre³

¹ CRP Henri Tudor, Luxembourg, Luxembourg

² Radboud University Nijmegen, Nijmegen, The Netherlands

³ University of Luxembourg, Luxembourg, Luxembourg

diana.marosin@tudor.lu, leon.vandertorre@uni.lu

Abstract. Intention reconsideration is a central challenge in BDI (Belief-Desire-Intention) theory. The intention reconsideration models of the early nineties focus on *when* a commitment may be reconsidered, for example when it has been achieved, when it is no longer achievable, or when the associated goal has been dropped. In this paper, we create an abstract framework in which we add the “reasons” and the “assumptions” that form the base of those commitments. Reasons are an abstract category of cognitive attitudes that motivate something else, including goals, principles, norms, intentions, and actions. To make our mechanism widely applicable, we introduce an abstract approach and 10 requirements from both early and more recent approaches, such as the model of Icard *et al.*, focusing on the interaction between beliefs and changing commitments. Moreover, we do not focus only on an intention’s reconsideration, but also on the *effects*. To that end, we introduce an intention reconsideration mechanism with algorithms satisfying the requirements, using only the dichotomy between assumptions and reasons.

Keywords: Group planning agreements · Distributed decision making · Coordination · Intention and commitment · Logics for agreement

1 Introduction

Intention reconsideration is a central challenge in BDI theory. The models of the early nineties developed by Cohen and Levesque [8] and by Rao and Georgeff [20] focus on *when* an intention may be reconsidered. We are interested in intentions that may be reconsidered when the reasons for the intention are no longer valid, or when the associated assumptions are violated. Previous work [3, 8, 18–20, 24] in the field of belief revision is lacking the relation between multiple intentions.

In human behaviour we see that a goal, a norm, an intention or action can lead to another intention, while in formal models of intention reconsideration we talk only about goals that can generate commitments. Therefore our scope for this paper is to investigate the link among the reasons of intentions. Also, in social sciences and real life scenarios we see people forming commitments based

on assumptions formulated a priori. One presumes certain background information (e.g. domain knowledge, environment, stakeholders) in order to define the requirements of a system under discussion. This information can be referred to as “assumptions” or “rationales”. An assumption is a notion related to “beliefs”, defined as “an act or statement (as a proposition, axiom, postulate, or notion) taken for granted” [2,6]. There is a distinction between assumptions and constraints [1]. Constraints are “items that will limit the developer’s options” (e.g. regulatory policies, reliability requirements, criticality of the applications safety and security considerations), while assumptions are “factors that affect the requirements stated” (e.g. one makes assumptions about the fact that a specific operating system will be available on the hardware designated for the software product).

These features are missing in the existing intention reconsideration models. Therefore, our main research question is:

How to change commitments based on reasons and assumptions?

There has already been a lot of work on intention reconsideration, using a variety of formalisms. For example, the early approaches use modal temporal logics, such as modal predicate logic [8] or BDI_{CTL} [20]. A more recent approach, by Icard *et al.*, uses belief revision [14]. We take as starting point those formalisms and in order to make our model generally applicable, we introduce an abstract approach, accommodating reasons and assumptions. The research questions are answered in three parts:

1. What are requirements for intention reconsideration based on reasons and assumptions?
2. How to define an abstract formal framework to accommodate reasons and assumptions?
3. How to define algorithms for changing commitments, based on reasons and assumptions?

Our approach is based on three ideas. First, if an assumption is violated, then we have to reconsider all intentions based on the assumption. Second, if an intention is retracted, then we have to find new intentions to satisfy the reasons. However, in general, when intentions have to be reconsidered, there can be many reasons for this change. In order to be able to change the assumptions and intentions we introduce the notion of an explained event. Third, an explained event does not contain only the assumptions which are violated and the intentions which are reconsidered, but also the reasons for the violations and reconsiderations.

We explain the model with an extended scenario of the house robot Willie, from Cohen and Levesque [8]. We do not fully automate the change of intentions. Instead, our logical abstract framework for intention reconsideration provides a setting to define actual procedures (definitions and algorithms).

This paper is structured after the research questions. In Sect. 2 we survey existing intention reconsideration mechanisms. We define ten requirements for

an intention reconsideration mechanism in Sect. 3. We introduce our running example in Sect. 4. In Sect. 5 we define the abstract formal framework, give examples and properties, and explain how our framework satisfies four requirements. In Sect. 6 we introduce two intention reconsideration algorithms based on assumptions and reasons and show how the algorithms satisfy the other six requirements. Related work is described in Sect. 7. Furthermore, we apply our framework on a case study from the field of enterprise architecture in Sect. 8. In the last section we conclude our work and we present our future research focus.

2 Intention Reconsideration

The BDI approach is one of the major approaches for building agents and multiagent systems. It was inspired by philosophy (theory of mind) and folk psychology, and as the name implies, the key here is to build agents using symbolic representations of their beliefs, desires, and intentions. The main idea is that an autonomous agent should act on its intentions, not in spite of them, adopt intentions it believes are feasible and forget those believed to be infeasible, keep or commit to intentions, but not forever, discharge those intentions believed to be satisfied, alter intentions when relevant beliefs change and adopt subsidiary intentions during plan formation.

To specify what it means for an agent to have an intention, one needs to describe how that intention affects the agent's web of beliefs, commitments to future actions or other independent intentions [8]. Cohen and Levesque define intentions in terms of temporal sequences of an agent's beliefs and goals, using the operators *BEL*, *GOAL* and *INTEND*. The agent fanatically commits to its intentions and will maintain its goals until either they are believed to be fulfilled or to be impossible to achieve. Rao and Gerogeoff [20] identify three commitment strategies. A *blindly* committed agent maintains its intentions until it actually believes that they were achieved. If an agent intends that ϕ be eventually true, then the agent will inevitably maintain its intentions until it believes ϕ . A blind-commitment strategy is very strong, as the agent will eventually come to believe it achieved its intentions or keep them forever. A weakening of the requirements leads us to defining a *single-minded* commitment, in which the agent maintains its intentions as long as it believes they are still options. As long as an agent believes its intentions are still achievable, a *single-minded* agent will not drop the intentions (and its committed goals). This requirement can be relaxed further, as an *open-minded* commitment. In this case, the agent maintains its intentions as long as these intentions are still its goals.

An alternative perspective is given by Shoham's database approach [14, 22]. In addition to atomic facts, the agent has beliefs about what the preconditions and postconditions of actions are and about which sequences of actions might be possible. From the perspective of a planner, the postconditions of intended actions are justifiable beliefs merely by the fact that the agent has committed to completing the action. In this way, these beliefs are contingent on the success of the agent's plans. The preconditions, on the other hand, are believed even if they

are not directly justified by any future intended action. These kinds of beliefs might also be called “optimistic” beliefs, since the agent assumes the success of the action without ensuring the preconditions hold.

We believe that Shoham’s approach is suitable to describe plans and in general the decision making process. Each commitment to a goal or action has an associated belief about the world, and has the role to support that commitment. We call those beliefs “assumptions.” Assumptions by definition do not necessarily need to be true, but the agent makes commitments assuming they are. This is what makes the preconditions in Shoham’s model similar to assumptions in our model and such assumptions are the key starting point of our framework.

3 Intention Reconsideration Requirements

In this section we present and motivate ten requirements for a mechanism of intention reconsideration.

The first requirement derives from the fact that existing models use a variety of formalisms, either to describe an intention reconsideration mechanism, or properties of such mechanisms, such as temporal logic methods (Cohen and Levesque [8], Rao and Georgeff [20]), or belief revision methods (Icard *et al.* [14]). The first requirement implies that our mechanism has to be applicable more generally, independently of the used formalism.

Requirement 1. The intention reconsideration mechanism should be defined in an abstract model covering existing models, in particular both the BDI logic approach, and the belief revision based approach.

The following requirements state that our mechanism should be able to model key features of existing models. Existing models distinguish blindly committed, simple - minded commitment and open - minded commitment. This says *when* an intention may be reconsidered, for example when it has been achieved, when it is no longer achievable, or when the associated goal has been dropped.

Requirement 2. The mechanism must be able to represent that intentions are reconsidered when the agent believes that they are no longer achievable.

Requirement 3. The mechanism must be able to represent that intentions are reconsidered when the agent believes that the associated goal is dropped.

Icard *et al.* [14] study the relation between belief revision and intention revision. An important relation between the two is that belief revision may trigger intention revision, but not vice versa. If intention revision would trigger belief revision we might come across inconsistent results. By imposing that intention revision does not trigger belief revision we avoid *wishful thinking* [5]. For example, Bob intends to drive his car to work, based on the belief that the car works. What if a colleague offers to drive Bob to work and he drops the intention of driving himself? This should not generate a revision process concerning the belief that Bob’s car is functional.

Requirement 4. Belief revision may trigger intention revision, but not vice versa.

The first extension we consider is that intentions are based on assumptions, and when the assumptions turn out false, then the intention is reconsidered. Also, there can be many reasons to form an intention. In classical models [8, 14, 20], the only reasons considered are goals. But, intentions can be based for example on norms, such that if norms are no longer in force, then the intention is reconsidered.

Requirement 5. The model of the mechanism associates assumptions with intentions, such that if belief revision leads to a violation of assumptions, then the related intentions are reconsidered.

Requirement 6. The model of the mechanism associates reasons such as goals and norms with intentions, such that if the reason disappears, then the intention is reconsidered.

From an architectural point of view, intentions need to be translated in a way such that their impact on the system can be described directly, in order to be incorporated into the system. We adopt Grossi's [13] abstract and concrete norms and apply them to intentions. Abstract intentions can be decomposed into several more concrete intentions. For example, the intention to "go to the cinema" can be decomposed into the intentions to "finish work early", "buy a ticket", and "travel to the cinema."

Requirement 7. The model of the mechanism associates new intentions with existing ones, such that if the latter are reconsidered, also the former must be reconsidered.

Existing models seem to focus on when an intention can be reconsidered. However, it is less often discussed how the reconsideration of an intention can affect other intentions and how to elaborate alternatives in case is not longer possible to commit to an intention.

Considering that an agent commits based on its assumptions about the current state of the world, we formulate the following requirement:

Requirement 8. The intention reconsideration mechanism should be such that if an intention is reconsidered because an assumption is violated, then other intentions based on the same assumption must also be reconsidered.

Given the human nature of the world, we argue that agents may commit to something based on previously made commitments. For example, Bob commits to "read a book", goes to the store in order to "buy a book" and commits also to "pay the book" based on a previously made commitment of "respecting the law." Considering Bob does not have his wallet with him, he should not just drop his goal of "reading" or his commitment to "respect the law", but he should find an alternative (e.g. "borrow a book" or "return home for money").

Requirement 9. The intention reconsideration mechanism should be such that if an intention is reconsidered while the reason¹ of this intention is still valid, then, if possible, another intention should be created to address this reason.

The description of the mechanism has to focus on the intention revision, and should not go into details less relevant for the mechanism.

Requirement 10. The model of the mechanism should be as simple as possible, in the sense that it does not introduce more concepts than necessary.

4 Running Example

We have the scenario by Cohen and Levesque as a starting point, describing Willie [8], the household robot. We explain how introducing reasons and assumptions can fix Willie’s attitude problems.

We represent Willie’s goal to provide beer to his owner (and *enable the owner to drink the beer*) and the plan he follows (deprecated in intentions and actions). He is committing to his goal, and two other commitments follow (*get the beer* and *bring a bottle opener*). Bringing the beer raises the option of getting it *from the fridge* or *from the table*. We mark the “and” relation between two nodes with an arc, while for the “or” relations we use two unconnected arrows. In Fig. 1 we also associate each intention/commitment/goal (square nodes) with the underlying assumptions (rounded nodes). For example, the intention to get the beer from the table is made based on the assumption that *there is beer on the table* and the robot *can reach the table*.

For this paper we use a simple example in order to illustrate our abstract framework and reconsideration mechanisms. The mechanism can be applied to more complicated structures; reasons can incorporate norms, goals, other intentions, commitments, actions. In this example we use all distinct elements of our framework. The example can be expanded further to any level or abstraction, or applied to different domains of interest.

5 Formal Framework

In this section we introduce a formal abstract framework for intention reconsideration based on reasons and assumptions. In Fig. 2 we abstractly represent the plan of robot Willie.

A reason is valid at a time moment if and only if all reasons that influence it are valid at that time. We say a reason holds at a time moment if it has not been invalidated, either by a false assumption or an influence reason. We say that if an upper node has his children in an “and” relation, all of the children need to be valid in order for the parent to be valid. If children are in an “or” relation, the parent needs at least one valid child in order to be valid.

¹ Note that we call norms, goals, other intentions, commitments or actions “reasons for a commitment”. Our driving hypothesis is that “everything is a reason for something else.”

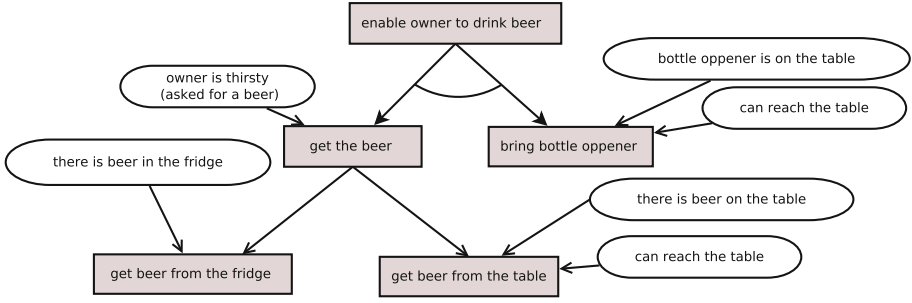


Fig. 1. Robot Willie’s plan extended with reasons and assumptions

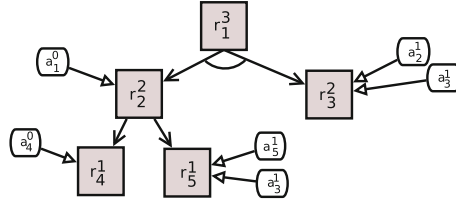


Fig. 2. Abstract representation of Robot Willie’s plan

5.1 Definitions

A standard distinction in temporal models is the distinction between *validity time* and *reference time*. If we assume today that it will rain tomorrow, then today is the validity time and tomorrow the reference time. We write a^t or (a, t) for an assumption a with reference time t , and we write \mathcal{A}_v for all assumptions with validity time v . We also write \mathcal{A}_v for all possible untimed assumptions at validity time v , and \mathcal{AT}_v for all possible timed assumptions at validity time v . We assume that the set of assumptions can increase over time, as new concepts may be introduced, and we thus have $\mathcal{A}_v \subseteq \mathcal{A}_w$ and thus $\mathcal{AT}_v \subseteq \mathcal{AT}_w$ if $v \leq w$.

Definition 1 (Assumptions). Let $\mathcal{T} \subset \mathcal{N}$ be a set of natural numbers expressing time moments, and let \mathcal{A} be the set of all possible assumptions. The set of all possible timed assumptions $\mathcal{AT} \subseteq \mathcal{A} \times \mathcal{T} \times \mathcal{T}$ is a set of triples of assumptions and two moments in time, the reference time and the validity time, such that $(a, t, v) \in \mathcal{AT}$ implies $(a, u, w) \in \mathcal{AT}$ for $v \leq w$.

We write $\mathcal{A}_v = \{a \mid (a, t, v) \in \mathcal{AT}\}$ and $\mathcal{AT}_v = \{(a, t) \mid (a, t, v) \in \mathcal{AT}\}$ for the projections of the possible assumptions at validity time v . For committed assumptions $A \subseteq \mathcal{AT}$, we write $A_v = \{(a, r) \mid (a, r, v) \in A\}$ for the projection of all assumptions holding at validity time v , and we write a^r for (a, r) , the assumption with reference time r .

The following example illustrates the notation. Note that assumption a_2^1 means assumption a_2 at reference time 1, in other words, the 2 is an index of the assumption and not a temporal reference.

Example 1. Consider the planning of robot Willie, represented abstractly in Fig. 2. $\mathcal{A}_0 = \{a_1, a_2, a_3, a_4, a_5\}$ represents the set of all possible assumptions in our model at time $t = 0$. $\mathcal{AT}_0 = \{a_1^0, a_2^0, \dots, a_1^1, a_2^1, \dots, a_1^3, a_2^3, \dots\}$ represents the set of all possible assumptions at a moment in time, made at time $t = 0$. The subset $A_0 = \{a_1^0, a_2^1, a_3^1, a_4^0, a_5^1\}$ represents the set of assumptions committed to at validity time 0. The fact that Willie assumes at the moment of planning 0 that the bottle opener is on the table at reference time 1 is represented by the presence in the set A_0 of the element a_2^1 .

Reasons are defined in precisely the same way as assumptions.

Definition 2 (Reasons). *Let \mathcal{R} be the set of all possible reasons. The set of all possible timed reasons $\mathcal{RT} \subseteq \mathcal{R} \times \mathcal{T} \times \mathcal{T}$ is a set of triples of reasons and two moments in time, the reference time and the validity time, such that $(r, t, v) \in \mathcal{AT}$ implies $(r, u, w) \in \mathcal{AT}$ for $v \leq w$.*

We write $\mathcal{R}_v = \{r \mid (r, t, v) \in \mathcal{RT}\}$ and $\mathcal{RT}_v = \{(r, t) \mid (r, t, v) \in \mathcal{RT}\}$ for the projections of the possible reasons at validity time v . For committed reasons $R \subseteq \mathcal{RT}$, we write $R_v = \{(r, t) \mid (r, t, v) \in R\}$ for the projection of all reasons holding at validity time v , and we write r^t for (r, t) , the reason with reference time t .

Reasons can be norms, goals, principles, or plans. We say that a reason is satisfied if and only if the norm is fulfilled, the goal is achieved, the principle is satisfied, or the plan is committed to.

Example 2. Continued from Example 1, we represent the set of reasons similar to the assumptions. For the abstract representation of Willie’s plan we define the following sets: $\mathcal{R}_t = \{r_0, r_1, r_2, r_3, r_4, r_5\}$, $\mathcal{RT}_t = \{r_1^0, r_2^0, \dots, r_5^0, \dots, r_1^3, r_2^3, \dots, r_5^3\}$, $R = \{r_1^3, r_2^2, r_3^2, r_4^1, r_5^1\}$

The elements of R_t which do not have parents in the graph are called **root** reasons, and the elements of R_t which do not contain a child in the graph are called **leaf** reasons.

Definition 3 (Assumptions dependences). *We define assumptions dependences as a function $AoR : R \rightarrow 2^A$ that maps a reason to its underlying assumptions.*

Example 3. We illustrate assumptions dependences from Example 1 as follows: $AoR_0(r_2^0) = \{a_1^0\}$, equivalently written $AoR_0 = \{(r_2^0, a_1^0)\}$, maps at validity time 0 the assumption “owner is thirsty” with Willie’s commitment to “get the beer.” The reason has reference time 2, meaning it is expected to hold at time moment 2.

Definition 4 (Reasons dependences). *We define reasons dependences as a function $RoR : R \rightarrow 2^{2^R}$ that maps a reason to a subset of influenced reasons.*

For each reason there can be several sets of other reasons depending on it, such that committing to one of these sets of reasons is sufficient to satisfy the reason. This has the property that if S and T depend on the same reason, then S cannot be a strict subset of T. This property can be seen in the way we represent the “or” relation between reasons.

Example 4. Continued from Example 1, for the validity time moment 0 we define the function RoR as follows: $RoR_0(r_1^3) = \{\{r_2^2, r_3^2\}\}$. In our example, r_1^3 is the root. Notice that r_2^2 and r_3^2 are in an “and” relation in respect with the parent intention r_1^3 . $RoR_0(r_2^2) = \{\{r_4^1\}, \{r_5^1\}\}$. In this case the commitments r_4^1 and r_5^1 are placed in an “or” relation in respect to their parent. For instance, using set notation, the “and” relation for the children of the root can be written as $\{(r_1^3, \{r_2^2, r_3^2\})\}$. The “or” relation for reason r_2^2 can be written equivalently $\{(r_2^2, r_4^1), (r_2^2, r_5^1)\}$. $RoR_0(r_4^1) = RoR_0(r_5^1) = RoR_0(r_3^2) = \{\}$. All three reasons are leaf nodes.

Definition 5 (Alternatives under discussion). *The alternatives under discussion are a tuple $Alt_t = \langle A_t, R_t, AoR_t, RoR_t \rangle$, where: $t \in \mathcal{T}$ is the validity time, A_t is a set of assumptions, R_t is a set of reasons, $AoR_t \subset A_t$ is the set of assumptions of each reason $r \in R_t$, $RoR_t \subset 2^R$ is a set of set of reasons of each reason $r \in R_t$, such that RoR_t is acyclic, connected and the sets of reasons in RoR_t does not contain strict subsets.*

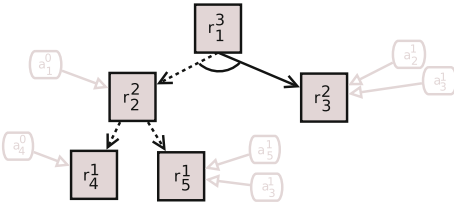


Fig. 3. Alternatives under discussion

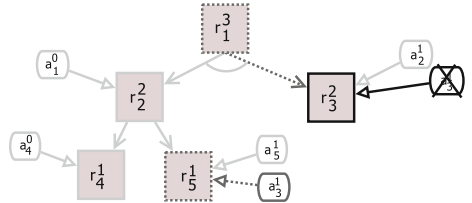


Fig. 4. An explained event

Example 5. The following example illustrates the alternatives under discussion, as presented in Example 1. Considering the time moment 0, we say that the entire plan is part of the alternatives under discussion, Willie having both options being equally valid, being able to commit to either “get the beer from the fridge” or to “get the beer from the table.” On the other hand, at time moment 2 it has to make a choice so the plan is divided in two independent alternatives, left and right side of the tree. The alternatives of the robot are represented abstractly in Fig 3.

Property 1 (Acyclic graph). We say that RoR is acyclic iff the graph $\{(x, y) \mid (Y) \in RoR(x), y \in Y\}$ is acyclic. A graph is acyclic if there is no path from a

node to itself. We do not consider discussions with cyclic dependencies among reasons.

Property 2 (Connected graph). A graph is connected if there is a path from each node to each other one if we add the inverse to the graph, i.e. inverse of G is $\{(x, y) \mid (y, x) \in G\}$. We consider only single issue discussions, that is, in which the graph of reasons in the alternatives under discussion is connected.

Definition 6 (Agreement). *An agreement at moment t is a tuple $AG_t = \langle C_t, RoR_t(r) \rangle$, where: $C_t \subseteq R_t$ is set of reasons committed to, such that $RoR_t(r) \cap C_t \times C_t$ is connected.*

Example 6. Continued from Example 1, an agreement is a subset of the alternatives under discussion. At time moment 1 we can represent the choice of “bring the beer from the fridge” as follows: $C_1 = \{r_1^3, r_2^2, r_4^1, r_3^2\}$, meaning that Willie committed to “get the beer from the fridge” to “bring a bottle opener.”

The reasons dependencies are defined for each reason in C_1 . An agreement made at time moment 1 is a tuple $AG_1 = \langle C_1, RoR_1(r) \rangle$.

Property 3 (Complete agreement). An agreement is complete iff C_t contains all root reasons and for every reason r which is not a leaf, C_t contains all reasons of one of the elements of $RoR_t(r)$.

Property 4 (Minimal agreement). An agreement is minimal iff it is minimal for set inclusion among the complete agreements.

When the agents decommit from an intention, they have to explain their decommitment by agreeing about the reason of the decommitment, and which other reasons are affected. In this discussion, the alternatives under discussion may be extended with new assumptions, reasons, and dependencies among them (hidden assumptions and reasons, i.e. hidden agenda is made explicit). We do not introduce new names for alternatives under discussion, we assume that from now on A_t , R_t , AoR_t and RoR_t refer to the expanded sets.

Definition 7 (Explained event). *An explained event E_t is a tuple $\langle D_t, V_t \rangle$, where: $D_t \subseteq C_t$ is a set of reasons the agents decommit from and $V_t \subseteq A_t$ is a set of assumptions which are violated such that $\{r \mid \exists a \in AoR_t(r) \cup V_t\} \subseteq D_t$.*

Example 7. Continued from Example 1, in Fig. 4 we illustrate an explain event (plain black lines), composed by the assumption that Willie “can reach the table” and the reason “bring bottle opener.” With dotted lines we marked the other reasons that are influenced by the failure of this assumption. More details follow in the algorithms presented in Sect. 6.

There can be two explanations of a decommitment: either an assumption is violated, or the agents decommitted from the reasons for the reason. Note that there can be several reasons why the agents committed to a reason, and therefore an explanation has to decommit from all these reasons.

Property 5 (Complete event). An explained event is complete iff for every $r \in D_t$, either there is an $(a \in V_t \cup AoR_t(r))$ or $(\{r' \mid \exists R \text{ s.t. } r \in R \in RoR(r')\} \subseteq D_t)$.

This leads to new alternatives under discussion Alt_{t+1} . Violated assumptions and decommitted reasons are removed from the alternatives under discussion, and new assumptions and reasons may be added to it.

Finally we consider the new agreement. We assume that the agents stay committed to their reasons, i.e. they are persistent.

Definition 8 (Persistent decisions). *The agents decisions are persistent iff $C_t \cup R_{t+1} \subseteq C_{t+1}$.*

5.2 Intention Reconsideration Requirements, Part 1

In this section we discuss how our model satisfies 4 of the 10 requirements for the mechanism of intention reconsideration presented in Sect. 3.

Requirement 1. Inspired by Dung’s abstract theory of argumentation [11], providing a graph based abstraction for non-monotonicity logics, our mechanism is expressed on a graph based representation of reasons and intentions. We can also instantiate our abstract model with logical formulas, along the lines of the aspic+ model [4].

Requirement 2. We can represent that an agent is not blindly committed, but drops its intentions once it believes that the intention is no longer achievable, by representing the belief as an assumption.

Requirement 3. Goals are a kind of reasons. The goal of an intention can be represented as a reason for the intention. Once the goal is dropped, the agent decommits from the reason and therefore also the intention is dropped. This is detailed in the algorithms presented in the next section.

Requirement 4. Beliefs revision leads to violation of assumptions, and consequently to decommitment of intentions. However, decommitment of intentions (or more generally, reasons) does not lead to violation of assumptions.

6 Algorithms and Requirements

6.1 Reconsideration Algorithms

In this section we introduce two revision algorithms: first, if the assumptions are violated we generate a reason revision; second, when we drop a commitment we generate a reason revision based on reasons.

Algorithm 1. Reason reconsideration based on assumptions

```

1: function REVISEBASEDONASSUMPTIONS( $A'', R, AoR, RoR$ )
2:   define  $R''$ , uncommitted reasons
3:   for all  $\bar{a} \in A''$  do
4:     for all  $r \in R$  do
5:       if  $(r, a) \in AoR$  then
6:          $r \leftarrow \bar{r}$ 
7:         add  $r$  to  $R''$ 
8:         del  $(r, X) \in AoR$ 
9:         del  $(r, \{X\}) \in RoR$ 
10:      end if
11:    end for
12:  end for
13:  reviseReasons( $R'', R, RoR$ )
14: end function

```

Algorithm 2. Reasons reconsideration based on reasons

```

1: function REVISEREASONS( $R'', R, RoR$ )
2:   while  $R''$  is not empty do
3:     for all  $r_c \in R''$  do
4:       for all  $r_p$  s.t  $r_p$  parent of  $r_c$  do
5:          $rel_1 \leftarrow (r_p, \{X\})$  s.t.
           (COND1)
6:          $rel_2 \leftarrow (r_p, \{Y\}) \in RoR$ 
7:         if  $rel_2 - rel_1 ==$  empty then
8:            $r_p \leftarrow \bar{r}_p$ 
9:           add  $r_p$  in  $R''$ 
10:        end if
11:         $RoR \leftarrow RoR - rel_1$ 
12:      end for
13:      remove  $r_c$  from  $R''$ 
14:    end for
15:  end while
16: end function

```

where *COND1* : $(r_c \in X \text{ or } X \equiv r_c) \text{ and } ((r_p, \{X\}) \in RoR)$.

The first algorithm receives as its parameters the invalidated assumptions, together with the reasons and relations among them (all sets defined in previous sections). It constructs a set of all invalidated reasons (R'') given the assumptions that failed (lines 2, 6 and 7). It also revises the relations between reasons and assumptions (lines 8 and 9). For the set of invalid reasons it calls the function for revision based on reasons (line 13).

The second algorithm receives as its parameter the set of reasons that turned out to be invalid. For each reason r_c it iterates in the original tree of reasons and builds the relation function between reasons (rel_1). By taking the original function RoR and making a difference between the elements ($rel_2 - rel_1$) we basically check if the parent node has more valid children or has to be invalidated also (lines 7–9). *COND1* is checking that each element of RoR containing an invalid reason is added to the set rel_1 . In the end, we update the relation RoR and remove the current invalid reason from the set R'' . It repeats the same operations for each reason in R'' , until the set becomes empty.

Example 8. Consider that Willie’s assumption that he “can reach the table” fails. Using the first algorithm we iterate through the set of reasons and select all those affected (“get beer from the table” and “bring bottle opener”). We also remove all pairs of reasons and assumptions from the list of assumption dependencies (AoR). The first algorithm calls also the revision based on reasons. The second algorithm receives as its parameters the invalid reasons (“get beer from the table” and “bring bottle opener”). For each reason it iterates in the tree in order to find, respectively check the validity of the parent. In the case of reason “get beer from the table” the parent “get the beer” remains invalid, because there is one element left in the reasons dependencies. This is not the case for the reason “bring bottle opener”, which also invalidates the parent “enable owner to drink beer.”

6.2 Intention Reconsideration Requirements, Part 2

In this section we discuss how our model satisfies the last 6 requirements for the mechanism of intention reconsideration, presented in Sect. 3.

Requirement 5. The *AoR* function associates assumptions with reasons. Moreover, Algorithm 1 shows how intentions are reconsidered, when the assumptions do not become reality.

Requirement 6. Both goals and intentions are represented as reasons, such that the *RoR* function can associate goals as well as other concepts like norms, intentions and actions with intentions. Moreover, the Algorithm 2 shows how intentions are reconsidered, when goals are dropped, actions are impossible to perform or norms are no longer in force.

Requirement 7. Since intentions are reasons, the *RoR* function can also represent that intentions depend on other intentions. For example, abstract intentions can be decomposed into several more concrete intentions. As for Requirement 6, the same method ensures that if intentions are decommitted, also intentions depending on it are decommitted.

Requirement 8. Algorithm 1 illustrates how the invalidation/reconsideration of an assumption can affect intentions.

Requirement 9. Algorithm 2 illustrates how the reconsideration of an intention can affect other intentions.

Requirement 10. The model only introduces assumptions and reasons. Many concepts have been unified, such a goals, norms and intentions into a single class called reasons. The fact that even intentions are called reasons, is that an intention itself can be a reason for another intention in an extension of the model (see Requirement 7.) We cannot further unify assumptions and reasons, because they have to be treated differently following Requirement 4. Finally, we show in the previous section that the algorithms can be applied on these two abstract classes, without for example having to know whether a reason is actually a goal, norm or intention. The algorithms distinguish only assumptions from reasons, and do not have to distinguish types of reasons.

7 Related Work

Decisions are treated as plans, in which the assumptions about the world are represented in a variety of ways, depending on the nature of the assumptions. When a plan is executed in a real environment it can encounter differences between the expected and actual context of execution. Those differences can manifest as divergences between the expected and observed states of the world, or changes in goals to be achieved. In both cases, the old plan must be replaced with a new one [23]. Classical planning techniques are often not sufficient, and they have therefore been extended with the theory of intentions.

Wooldridge and Parsons [19,24] develop a simple formal model and investigate the behaviour of this model in different types of task environment. An agent’s internal state is characterised by a set of beliefs and a set of intentions. In addition, an agent has a deliberation function, which allows it to reconsider and if necessary modify its intentions, and an action function, which allows it to act towards its current intentions. Shoham suggest viewing the plan as specifying an “intelligent database” [14,22] capturing the current beliefs of the agent while ensuring that the beliefs remain consistent at all times. Each action has associated an associated pre- and post- condition, with the property that if the preconditions are absent or invalid, the action can not be taken, but in case the action was taken then the postcondition hold. What is lacking in those BDI approaches are reasons for an intention, which may be another intention. We introduced an abstract framework that allows us to reason on relations between multiple goals, principles, actions (all called “reasons” for simplicity) and their assumptions. We developed a model of intention reconsideration inbetween the “single-minded” and “open-minded” revision, as described by Cohen and Levesque [8] or Rao and Georgeff [20]. We call this paradigm “assumption-minded” revision.

Mavromichalis and Vouros [18] propose a BDI approach for plan elaboration and reconsideration based on reasons for intentions (recorded as previous user inputs). In the case of a conflictual situation the user asks for the collaborative agent’s help. The agent recognizes the cause of failure and initiate collaboration for an alternative action, that is defined based both on the erroneous action and an action that can resolve the conflict. Therefore, the agent communicates the actions that have to be performed and motivates their performance. We take a similar approach, formally defining an *explained event*, that contains both the set of violated assumptions and the reasons the agents decommits from (see Definition 7 and Example 7).

An alternative approach to belief revision for *updating existing information* is the use of Truth Maintenance Systems, as described by Doyle [10]. Both try to solve the same problem, but TMS can be seen as a way of storing proofs, while our approach is an abstract framework for intention reconsideration, therefore there is not a straight forward link to TMS.

We consider the work of Castelfranchi and Paglieri [7] complementary to ours. The authors created a constitutive theory of intentions and a taxonomy of beliefs, we, on the other hand, are not concerned on how intentions are formed, but rather on how they are triggered to change. The author’s claim that “goals have to be supported by beliefs” is completely integrated in our framework by the *AoR* relation (see Definition 3). We go one step further and investigate also the regulative/supporting role of intentions over other intentions (see Definition 4).

Another important difference between our formalism and others mentioned above is the way we represent time. Rao and Georgeff use a CTL (Computation Tree Logic), meaning that the model of time is a tree-like structure in which the future is not determined; there are different paths in the future, any one of which might be an actual path that is realised. Cohen and Levesque define intentions as temporal sequences of the agents’ beliefs and goals. We represent

time modalities using LTL (Linear Temporal Logic), such that we can encode information about the future (a condition will eventually be true, a condition will be true until another fact becomes true). In TMSs time is not represented explicitly. Time steps can be deduced by the sequential tagging rules that feed the system.

Also, we mention that our framework is abstract, not instantiated, while Rao and Georgeff or Shoham use propositional logics. Some of TMSs use propositional logic, others are designed for predicate logics, others for only monotonic or non-monotonic logics [21].

8 “ArchiSurance” – Case Study

We illustrate our intention reconsideration model with an example from enterprise architecture driven by the fact that decisions (together with their associated commitments) in enterprise architecture typically change various times during their life span. Enterprises are in a constant state of change given by, e.g. the economic climate, companies merging or acquisitions, new technologies. Business performance depends on a balanced and integrated design of the organization, involving people, competences, structure, business processes, IT, finance, products, and services [12]. Our framework allows us to revise commitments, rather than creating them from scratch. The driving hypothesis of our work is that the resulting traceability between decisions and their underlying assumptions can enable a better underpinning of architectures, while at the same time triggering advanced impact analysis when confronted with changes. We are not interested in how decisions are taken in the first place, what cultural issues are involved (like norms, trust, organisation...), as we do not follow the contractual aspects. Instead, we focus on planning and on changes on the level of intentions, triggered by the change of assumptions or other intentions.

8.1 Description of the Case Study

In this section we briefly present the *ArchiSurance* case study. This case is inspired by a paper on the economic functions of insurance intermediaries [9], and is the running case used to illustrate the ArchiMate language specifications [15]. More details about the application of the current framework on the case study can be found in our related work [16, 17].

ArchiSurance is the result of a merger of three previously independent insurance companies: *Home and Away*, specializing in home owner’s insurance and travel insurance, *PRO-FIT*, specializing in auto insurance and *Legally Yours*, specializing in legal expense insurance. The company now consists of three divisions with the same names and headquarters as their independent predecessors.

The board’s main driver (goal) is to increase its “Profit”. Drivers motivate the development of specific business goals, as shown below in Fig. 5. Sub-goals such as “cost reduction” can be partitioned into the “reduction of maintenance costs” and the “reduction of personnel costs”.

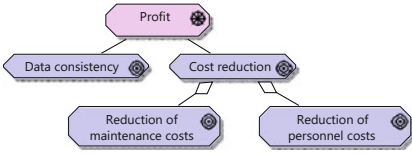


Fig. 5. “ArchiSurance” - business goals associated with “Profit” [15]

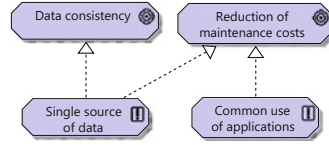


Fig. 6. “ArchiSurance” - refinement of business goals [15]

Business goals can be further refined, as presented in Fig. 6. The company needs to commit to realise a “single data source” in order to fulfil “data consistency”, and commit either to “single data source” or “create common use application” to ensure the realization of the goal “reduction of maintenance costs”.

8.2 Application of the Framework

In the original case study time is not present explicitly but implicitly, due the influence relations between commitments made. For example, it is pointless to commit to testing an application if chronologically you did not commit to creating the application beforehand. We use common sense reasoning and attach time points to the goals, principles, actions and assumptions. We describe below the reasons, assumptions and relations between them.

We consider that the board decides to commit to the strategic goal “profit” (r_1^5). This commitment appears at $t = 0$, the moment of initial planning and it means that at moment $t = 5$ the goal “profit” will be fulfilled. In order for this to happen, the board expects that the strategic principles “data consistency” (r_2^4) and “cost reduction” (r_3^4) will be fulfilled at time moment 4. Data consistency can be achieved by the acquisition of a new server (r_4^2) and/or merging of the databases (r_3^3). Notice that those actions have to be completed at an earlier time that the moment we expect data consistency to hold, here time moments 2 and 3. In order to validate the “cost reduction”, the board commits to “creating a common use application” (r_6^4). This leads to a testing phase (r_7^4).

Notice that the time point associated with each commitment show a logical ordering of actions. For example we cannot plan at time moment 4 a testing phase (r_7^4) before committing to creating an application (r_6^4) at an earlier time point, here 3.

In a real world environment all commitments are made based on assumptions about the world. In our example the assumptions that were made are as follows: buying a new server was generated by the availability of the technology on the market (a_2^1); common application and merging of databases require the hiring of a new developer (a_4^2) and the acquisition of software licenses (a_5^1); testing phase is planned on the assumption that bugs might be introduced (a_7^3).

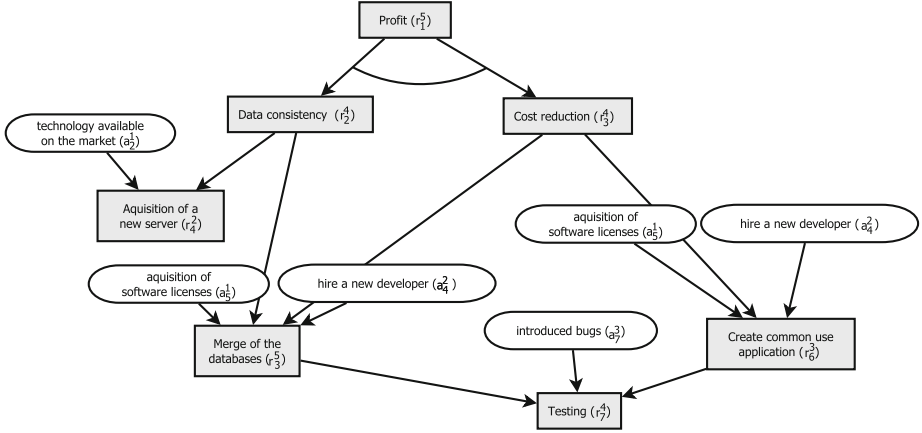


Fig. 7. Initial planning for the company “ArchiSurance”

Example 9. The planning of the merger of the three companies is presented in Fig. 7. The time point of this planning is $t = 0$. We notice that the initial planning contains also alternatives. We see that in order to obtain “profit” we need both “data consistency” and “cost reduction”, but for example “data consistency” can be fulfilled with either a “merge of databases” (\mathcal{AG}_2) or “acquisition of a new server” (\mathcal{AG}_1). The company can very well plan in the beginning to obtain both, even if one is already sufficient:

$$\mathcal{AG}_1 = \langle \{r_1^5, r_2^4, r_3^3, r_4^4, r_6^3, r_7^3\}, \{r_1^5, \{r_2^4, r_3^3\}\}, (r_2^4, \{r_3^3\}), (r_3^3, \{r_6^3\}), (r_6^3, \{r_7^3\}) \rangle$$

$$\mathcal{AG}_2 = \langle \{r_1^5, r_2^4, r_4^4, r_3^3, r_6^3, r_7^3\}, \{r_1^5, \{r_2^4, r_4^4\}\}, (r_2^4, \{r_4^4\}), (r_4^4, \{r_6^3\}), (r_6^3, \{r_7^3\}) \rangle$$

Note that each time there is a “fork” in the plan the board has to get to a new agreement.

9 Conclusions and Future Work

In this paper we introduced a mechanism of intention reconsideration based on reasons and assumptions. Intention reconsideration is a central challenge in BDI theory, but the models of the early nineties (e.g. Cohen and Levesque, Rao and Georgeff) focus on *when* an intention may be reconsidered, for example when it has been achieved, when it is no longer achievable, or when the associated goal has been dropped. The first contribution of the paper is a list of ten requirements for intention reconsideration mechanisms.

The first set of requirements concern the nature of the model, which must be expressive enough to incorporate key concepts of existing models, for example to express commitment strategies. The models of the early nineties focus on *when* an intention may be reconsidered, for example when it has been achieved, when it is no longer achievable, or when the associated goal has been dropped. In addition, we require that intentions may be reconsidered too when the associated assumptions are violated, or when the reasons for the intention are no longer valid.

The second set of requirements concerns the intention reconsideration mechanism, which not only states when to reconsider an intention, but also the *effects* of reconsidering an intention. The mechanism has to consist not only of a model, but also of algorithms. The dichotomy between assumptions and reasons greatly simplifies our algorithms for changing agreements: if an assumption is violated, then we have to reconsider all intentions based on the assumption, and if an intention is retracted, then we have to find new intentions to satisfy the reasons. The reasons explain not only why the agents commit to the intentions, but they are in particular used to explain how intentions are reconsidered. Besides the reasons, also assumptions of intentions are represented. The relations between intentions, assumptions and reasons lead to a graph based representation, and the intention reconsideration therefore corresponds to change of these graphs.

The third set of requirements concerns the applicability of an intention reconsideration mechanism. Despite the popularity of existing models and formalisms, Shoham observes that due to their philosophical nature, they also tend to become relatively complicated. To make our mechanism widely applicable, we require an abstract approach. Here we are inspired by Dung's theory in the field of argumentation, which achieved a much wider public after existing work on non-monotonic logic and logic programming was abstracted to simple graph based notions. We would like to make an additional comparison with Dung's theory of abstract argumentation. The success of this formalism is partly due to the possibility to instantiate the abstract arguments with logical proofs. We intend to study the possibility of instantiating our abstract model with logical languages to relate it to logical models of intention reconsideration, such as Shoham's model which we used to represent assumptions as preconditions of actions.

We applied our framework of intention reconsideration with a case study from the field of enterprise architecture. Even if it presents a simplification of reality, it is a step further in assessing both the utility and the applicability of the framework. In practice it is often the case that we can not commit to more goals or actions at the same time. Another topic for future research is investigating the consequences of conflating intentions, norms and goals. It is easy for an agent to drop goals and intentions unilaterally (assuming no external commitment), but norms cannot be unilaterally changed (although one could choose to ignore them). We envision refining our model by distinguishing between norms and goals, but this is out of the scope of this paper. Furthermore, it is often the case that we can not commit to more goals or actions at the same time. It happens due to the lack of resources or due to the conflicting goals, to be forced to chose "exactly one" commitment. In order to describe those situations, we should introduce a relation of the type "xor" between reasons. We also believe that further investigations should be done on concurrent commitments and on the issues raised.

In this paper, we kept the formal details at a minimum, to make the paper as accessible as possible. We focused on the ideas and motivations rather than technical details. In the foreseen extension of this paper, these technical details

will be developed further, and the mechanism will be illustrated with a larger case study.

We intend to continue with time manipulation in the structure. For example, at time moment 2 we realize the impossibility to fulfill a commitment. After revision the newly generated alternative should follow at a later time point. All commitments that depended on the invalid reason, their children, children of children, etc., should shift their time point.

References

1. IEEE guide for developing system requirements specifications. Technical report (1998)
2. Merriam Webster Dictionary, July 2013
3. Alchourrón, C.E., Gärdenfors, P., Makinson, D.: On the logic of theory change: partial meet contractions and revision functions. *J. Symbolic Logic* **50**(2), 510–530 (1985)
4. Bex, F., Prakken, H., Reed, C.: A formal analysis of the AIF in terms of the ASPIC framework. In: *COMMA*, pp. 99–110 (2010)
5. Boella, G., da Costa Pereira, C., Pigozzi, G., Tettamanzi, A., van der Torre, L.: The role of goals in belief selection. *Logic J. IGPL* **18**(4), 559–578 (2010)
6. Bulandran, S.: An exploration of assumptions in requirements engineering. Ph.D. thesis, The University of Western Australia (2012)
7. Castelfranchi, C., Paglieri, F.: The role of beliefs in goal dynamics: prolegomena to a constructive theory of intentions. *Synthese* **155**(2), 237–263 (2007)
8. Cohen, P.R., Levesque, H.J.: Intention is choice with commitment. *Artif. Intell.* **42**(2–3), 213–261 (1990)
9. Cummins, J., Doherty, N.: The economics of insurance intermediaries. *J. Risk Insur.* **73**(3), 359–396 (2006)
10. Doyle, J.: A truth maintenance system. *Artif. Intell.* **12**(3), 231–272 (1979)
11. Dung, P.M.: On the acceptability of arguments and its fundamental role in non-monotonic reasoning, logic programming and n-person games. *Artif. Intell.* **77**(2), 321–358 (1995)
12. Greefhorst, D., Proper, H.: *Architecture Principles - The Cornerstones of Enterprise Architecture*. Enterprise Engineering Series. Springer, Berlin (2011)
13. Grossi, D.: *Designing invisible handcuffs - formal investigations in institutions and organizations for multi-agent systems*. Ph.D. thesis, University of Utrecht (2007)
14. Icard, T.F., Pacuit, E., Shoham, Y.: Joint revision of beliefs and intention. In: *KR* (2010)
15. Jonkers, H., Band, I., Quartel, D.: *The ArchiSurance Case Study*. White paper, The Open Group, Spring (2012)
16. Marosin, D.: *Changing enterprise architectures: abstract framework, revision procedures and algorithms*. Master's thesis, University of Luxembourg (2012)
17. Marosin, D., Proper, H.A., van der Torre, L.: Changing agreements: intention reconsideration based on assumptions and reasons. In: Ossowski, S., Toni, F., Vouros, G.A. (eds.) *AT*. *CEUR Workshop Proceedings*, vol. 918, pp. 262–263. CEUR-WS.org. (2012)
18. Mavromichalis, V.K., Vouros, G.: Building intelligent collaborative interface agents with the icagent development framework. *Auton. Agents Multi-Agent Syst.* **13**(2), 155–195 (2006)

19. Parsons, S., Pettersson, O., Saffiotti, A., Wooldridge, M.: Intention reconsideration in theory and practice. In: ECAI, pp. 378–382 (2000)
20. Rao, A.S., Georgeff, M.P.: Intentions and rational commitment. In: Proceedings of the First Pacific Rim Conference on Artificial Intelligence (PRICAI-90) (1993)
21. Shapiro, S.C.: Belief revision and truth maintenance systems: an overview and a proposal. Technical report (1998)
22. Shoham, Y.: Logical theories of intention and the database perspective. *J. Philos. Logic* **38**, 633–647 (2009)
23. van der Krogt, R., de Weerdt, M.: Plan repair as an extension of planning. In: ICAPS, pp. 161–170 (2005)
24. Wooldridge, M., Parsons, S.: Intention reconsideration reconsidered. In: Papadimitriou, Ch., Singh, M.P., Müller, J.P. (eds.) ATAL 1998. LNCS (LNAI), vol. 1555, pp. 63–79. Springer, Heidelberg (1999)