# Principles of Intention Reconsideration

Martijn Schut and Michael Wooldridge

Department of Computer Science
University of Liverpool
Liverpool L69 7ZF, U.K.

{m.c.schut, m.j.wooldridge}@csc.liv.ac.uk

## ABSTRACT

We present a framework that enables a belief-desire-intention (BDI) agent to dynamically choose its intention reconsideration policy in order to perform optimally in accordance with the current state of the environment. Our framework integrates an abstract BDI agent architecture with the decision theoretic model for discrete deliberation scheduling of Russell and Wefald. As intention reconsideration determines an agent's commitment to its plans, this work increases the level of autonomy in agents, as it pushes the choice of commitment level from design-time to run-time. This makes it possible for an agent to operate effectively in dynamic and open environments, whose behaviour is not known at design time. Following a precise formal definition of the framework, we present an empirical analysis that evaluates the run-time policy in comparison with design-time policies. We show that an agent utilising our framework outperforms agents with fixed policies.

## 1. INTRODUCTION

One of the key problems in the design of belief-desire-intention (BDI) agents is the selection of an *intention reconsideration policy* [6, 12, 15]. Such a policy defines the circumstances under which a BDI agent will expend computational resources deliberating over its intentions. Wasted effort — deliberating over intentions unnecessarily — is undesirable, as is not deliberating when such deliberation would have been fruitful. There is currently no consensus on exactly how or when an agent should reconsider its intentions. Current approaches to this problem simply dictate the *commitment level* of the agent, ranging from *cautious* (agents that reconsider their intentions at every possible opportunity) to *bold* (agents that do not reconsider until they have fully executed their current plan). Kinny and Georgeff investigated the effectiveness of these two policies in several types of environments [6]; this work has been extended by other researchers [12]. However, in all previous work, the intention reconsideration policy is selected at *design time*, and hardwired into the agent. There is no opportunity for modifying the policy at run time. This is clearly not a practical solution for agents that must operate in dynamic and open environments.


*AGENTS'01*, May 28-June 1, 2001, Montréal, Quebec, Canada.


In this paper, we propose to let an agent choose for itself which policy to adopt depending on the current state of the environment. To this end, we adopt a *meta-level decision theoretic approach*: instead of figuring out *what* to decide, the agent must know *how* to decide. The key idea is that an intention reconsideration policy can be seen as a kind of meta-level control (cf. [16]), which selects between deliberating and acting. In this frame of reference, relevant research has been carried out by Russell and Wefald [11] on the role of meta-reasoning in a decision-theoretic reasoning model. Their model aims to control reasoning by making an explicit choice between action and computation. In this paper, we adapt and apply Russell and Wefald's model to the problem of intention reconsideration in BDI agents. The aim is to develop a rigorous framework for intention reconsideration, that can be applied and used by agents in practical circumstances. Meta-level reasoning has of course been a major topic of research in AI for some time, (see e.g., [7]), but to the best of our knowledge, we are the first to apply a theoretical model of meta-reasoning and meta-control to the problem of optimal intention reconsideration.

The remainder of this paper is structured as follows. The following subsection provides some basic background information and introduces the two models — the BDI architecture and Russell and Wefald's model of decision-theoretic meta-reasoning — upon which our framework builds. Section 2 formalises these models, and shows how intention reconsideration can be understood through the medium of Russell and Wefald's model. We then show how the analysis we develop can be applied to the TILEWORLD domain [9]. In section 3, we present an empirical evaluation of our framework in the TILEWORLD domain, in section 4 we discuss related work and finally, in section 5 we present some conclusions and discuss possible future work.

## Background

Ideally, an autonomous agent reasons about its decision making process and chooses a decision mechanism in accordance with the current state of the environment. If the agent has bounded resources, then these reasoning processes must be carried out efficiently. These are the long term goals of the research described in this paper: we want an agent that is flexible and autonomous with respect to open and unpredictable environments. Therefore we must first of all acknowledge the fact that the behaviour of the agent might in some cases not be optimal in the broadest sense: optimality must be considered with respect to the constraints that an environment puts upon the agent. In this way, optimality is a trade-off between the capabilities of the agent and the structure of the environment. The characteristics of this trade-off have been empirically investigated in [6] and [12].

A popular approach to autonomous agent design is the belief-

```
Algorithm:  BDI Agent Control Loop
1.
2.   B ← B0;
3.   I ← I0;
4.   π ← null;
5.   while (true) do
6.        get next percept ρ;
7.        update B on the basis of ρ;
8.        if (reconsider(B, I)) then
9.             D ← options(B, I);
10.            I ← filter(B, D, I);
11.            if (not sound(π, I, B)) then
12.                 π ← plan(B, I);
13.            end-if
14.       end-if
15.       if (not empty(π)) then
16.            α ← hd(π);
17.            execute(α);
18.            π ← tail(π);
19.       end-if
20.  end-while
```

**Figure 1: The abstract BDI agent control loop. The loop consists of continuous observation, deliberation, planning and execution. To perform optimally, the *reconsider*(...) function decides whether deliberation and planning is necessary.**

desire-intention (BDI) model of agency. The control loop of a BDI agent is shown in figure 1, which is based on the BDI agent control loop presented in [10] and [15, p38]. The idea is that an agent has *beliefs B* about the world, *intentions I* to achieve and a *plan* $\pi$ to achieve intentions. In lines 2–4, the beliefs, intentions and plan are initialised. The main control loop is then in lines 5–20. In lines 6–7, the agent perceives and updates its beliefs; in line 8, it decides whether to reconsider or not; in lines 9–13 the agent deliberates, by generating new options and deliberating over these; in line 12, the agent generates a plan for achieving its intentions; and in lines 15–18 an action of the current plan is executed. Because the purpose of the functions used in this loop can be easily derived from their names, we omit the actual formalisations here for reasons of space, but direct the reader to [15, ch2].

It is necessary for a BDI agent to reconsider its intentions from time to time. One of the key properties of intentions is that they enable the agent to be goal-driven rather than event-driven, i.e., by committing to intentions the agent can pursue long-term goals. But when circumstances have changed and, for example, an intention cannot be achieved anymore, the agent would do well to drop that intention. Similarly, when opportunities arise that enable intentions that the agent currently has not adopted, the agent should reconsider. However, because reconsideration is itself a potentially costly computational process, one would not want the agent to reconsider its intentions at every possible moment, but merely when it is necessary to reconsider, i.e., when, after reconsideration, the set of intentions has changed. The purpose of the *reconsider*(...) function as shown in figure 1 is precisely this: to deliberate when it pays off to deliberate, (i.e., when deliberation will lead to a change in intentions), and otherwise not to deliberate, but to act.

This gives us insight into the desired behaviour of an intention reconsideration policy, but it does not say how to *implement* it. Our framework, which we introduce in the next section, is to be used for this implementation. This model is based on the decision theoretic model of Russell and Wefald that we discuss in the remainder of this section.

In [11], Russell and Wefald describe how an agent should schedule deliberation and action to achieve efficient behaviour. Their framework is known as *discrete deliberation scheduling*[1]. The key idea is that *deliberations are treated as if they were actions*. Decision theory gives us various models of how to determine the best possible action, of which the maximum expected utility model is perhaps the best known. Viewing deliberations as actions allows us to compute the utility of a deliberation action, and so makes it possible to apply the expected utility model as the meta-level reasoning component over all possible actions and deliberations. However, it is not difficult to see that this can be computationally hard. Russell and Wefald propose the following strategy in order to overcome this problem. Assume that at any moment in time the agent has some default action it can perform. The agent can either execute this action or deliberate, where deliberation can lead to a better action than the current default action. Their control algorithm then states that as long as there exist deliberations with a positive value, perform the deliberation with the highest value; otherwise, execute the default action.

This paper discusses the integration of the decision theoretic model for deliberation scheduling from Russell and Wefald and the BDI agent architecture. In the section to follow we lay out the initial formalisation of the model.

## 2. THE FORMAL MODEL

In this section, we present our formal model. We introduce all necessary basic elements, present the control algorithm which uses these elements, and suggest some additional assumptions required to make the algorithm computationally attractive. Finally, we show how the model can be applied to an example scenario — the TILEWORLD — which illustrates the theory and which serves as the application domain for our experiments.

The most important issue we are concerned with relates to the set of available actions $A$ of the agent: we distinguish between *external* actions $A_{ext} = \{a, a', a'', \ldots\}$, affecting the agent's environment, and *internal* actions $A_{int} = \{d, d', \ldots\}$, affecting the internal state of the agent. We let $A = A_{ext} \cup A_{int}$ and assume $A_{int} \cap A_{ext} = \emptyset$. We assume the agent's environment, (i.e., everything external to the agent), may be in any of a set $E = \{e, e', e'', \ldots\}$ of environment states. We let *utility* be defined over environment states: $U_{\mathbf{e}} : E \rightarrow \mathbb{R}$. If the agent uses maximum expected utility theory (MEU) as a decision strategy, it chooses an action $a_{meu} \in A_{ext}$ for which the utility of the outcome state is maximal:

$$a_{meu} = \arg\max_{a \in A_{ext}} \sum_{e \in E} P(e \mid a) U_{\mathbf{e}}(e) \qquad (1)$$

where $P(e \mid a)$ denotes the probability of state $e$ occurring, given that the agent chooses to perform external action $a$.

However intuitive this notion of decision making is, many problems arise when MEU is used in the real world. It assumes $U_{\mathbf{e}}(E)$ is known before deciding, that enough time is available to obtain $a_{meu}$, and it does not extend to *sequential* decision making. Russell and Wefald offer an alternative [11]. The idea underlying their model is that the agent chooses between: (1) a default external action $a_{def}$, and (2) an internal action from the set of internal actions — at any moment, the agent selects an action from $\{a_{def}, d, d', \ldots\}$. The only purpose of an internal action is to revise the default external

[1]This contrasts with the *continuous* deliberation scheduling framework, which is the term mainly used to cover work such as anytime algorithms (see e.g. [1]).

action, presumably to a better one. This algorithm does not ensure an optimal choice, but computationally it can be a lot more attractive than MEU. To choose between actions, we need to represent the preferences for those actions, which we do via a function $U_{\mathbf{a}} : A \to \mathbb{R}$, which represents the *net value* of an action (either external or internal). Note that now we have two kinds of utilities in the model: for environment states and for actions respectively. We relate these by letting the utility of an action $a$ be the weighted sum of the environment states the action may lead to:

$$U_{\mathbf{a}}(a) = \sum_{e \in E} P(e \mid a) U_{\mathbf{e}}(e) \tag{2}$$

where $P(e \mid a)$ is the probability of outcome $e$ given that the agent chooses to perform external action $a$, and $U_{\mathbf{e}}(e)$ is the utility of $e$.

Now, the *best* possible internal action for the agent to undertake, referred to as the *optimal deliberation*, is the deliberation with maximum utility:

$$d_{opt} = \arg\max_{d \in A_{int}} U_{\mathbf{a}}(d). \tag{3}$$

Russell and Wefald's decision control algorithm (DCA) then lets the agent deliberate when there exists a deliberation with a positive net value, and act when this is not the case.

While DCA reduces the search space of actions to deliberate over (it limits the actions for which utilities must be computed to the default action), it is still not applicable in real-time, because the computation of $d_{opt}$ is generally very costly (see e.g., [14]). We are not so concerned with this intractability here, since we only consider a single internal action: the deliberation that leads to new intentions. The key is to determine the utility of this action as opposed to the external actions available to the agent.

In order to represent the behaviour of the environment, we use an *external state transition function*, $\mathcal{N} : E \times A^* \to E$, which maps a state of the environment and a sequence of actions to some new environment state. Notice that the environment is here implicitly assumed to be deterministic.

Thus far, we have presented Russell and Wefald's decision algorithm and the BDI agent architecture. We now formalise Russell and Wefald's model of meta-reasoning and show that it can be used for implementing the *reconsider*$(\ldots)$ function in BDI: we show how to compute the utility of external and internal actions, explain how to estimate the utilities of internal actions and then concern ourselves with representing temporal constraints on the utilities of internal actions.

First, we redefine the notion of MEU using these richer semantics. An agent chooses the *optimal external action* — the action that maximises expected utility:

$$E(U_{\mathbf{e}}(\mathcal{N}(e_{now}, [a]))) = \sum_{e_i \in E} P(e_i) U_{\mathbf{e}}(\mathcal{N}(e_i, [a])) \tag{4}$$

where $e_{now}, e_i \in E$ and $a \in A_{ext}$; $\mathcal{N}(e_{now}, [a])$ is the result of executing action $a$ in the current environment state; $P(e_i)$ is the probability that the current environment state is $e_i$; and $\mathcal{N}(e_i, [a])$ is the result of executing action $a$ in environment state $e_i$. Note that this definition takes only external actions into account. We define the value of an internal action initially as the difference between the utility of executing the default action $a_{def}$ and an internal action $d$:

$$U_{\mathbf{a}}(d) = U_{\mathbf{e}}(\mathcal{N}(e_{now}, [d; a_d])) - U_{\mathbf{e}}(\mathcal{N}(e_{now}, [a_{def}])) \tag{5}$$

where $a_d \in A_{ext}$ denotes the external action resulting from $d$ and $\mathcal{N}(e_{now}, [d; a_d])$ is the environment state resulting from first executing $d$ and then executing $a_d$. Note the following two assumptions in this definition: $d$ immediately results in an external action $a_d$ and executing $d$ does not cost anything computationally. The first assumption excludes series of internal actions: it might be the case that $d$ will not result immediately in an external action. Russell and Wefald refer to an internal action that immediately results in an external action as a *complete computation*, and to one that does not necessarily do so as a *partial computation*[2] — the set of complete computations is a subset of the set of partial computations. In [11], the emphasis is mainly on complete computations. Here too, we are only concerned with complete computations and leave the issue of partial computations as an interesting theoretical extension of the framework for further work.

The equations presented so far assume that the agent has immediate access to its utility function. In reality, however, this is hardly the case for people when they make decisions. Instead, they have to estimate utilities of environment states before deciding and indeed so will our agent need to estimate its utilities. In the equations we replace the utility $U$ by a utility estimate $\hat{U}_{\mathbf{e}\sigma}$, where $\sigma \in A^*$ is a sequence of actions. Then $\hat{U}_{\mathbf{e}\sigma}$ denotes the estimation of a state utility after executing the specified course of action $\sigma$. Consequently, we replace the value of an action $U_{\mathbf{a}}$ by the estimate value $\hat{U}_{\mathbf{a}}$. In this way, equation (5) becomes:

$$\begin{aligned} \hat{U}_{\mathbf{a}}(d) \quad = \quad & \hat{U}_{\mathbf{e}[S;d]}(\mathcal{N}(e_{now}, [d])) \\ & - \hat{U}_{\mathbf{e}[S;d]}(\mathcal{N}(e_{now}, [a_{def}])) \end{aligned} \tag{6}$$

where $[S; d]$ denotes a sequence of computations $S$ followed by computation $d$, and $\hat{U}_{\mathbf{e}[S;d]}(E)$ denotes the utility estimate of the environment state based on $[S; d]$ — in the equation resulting from executing $d$ or $a_{def}$ respectively.

Now, estimates are by default random at initialisation, i.e., before $d$ is executed. In order to be able to utilise knowledge of, for example, statistical knowledge of the distribution of $\hat{U}_{\mathbf{a}}$ from past situations, we need to use the expectations of these estimates. Consequently, we replace (6) by:

$$\begin{aligned} E(\hat{U}_{\mathbf{a}}(d)) \quad = \quad & E(\hat{U}_{\mathbf{e}[S;d]}(\mathcal{N}(e_{now}, [d])) \\ & - \hat{U}_{\mathbf{e}[S;d]}(\mathcal{N}(e_{now}, [a_{def}]))). \end{aligned} \tag{7}$$

Russell and Wefald show that the value of $E(\hat{U}_{\mathbf{a}}(d))$ depends on the probability distribution for future utility estimates for external actions. After $d$ has been executed, the agent has at its disposal a joint distribution for the probability that external actions $\{a, a', a'', \ldots\}$ obtain new utility estimates $\{u, u', u'', \ldots\}$, respectively. Then the external action resulting from $d$ is the action with corresponding maximum estimated utility, weighted by the probability distribution for this action. The utility of the current best external action — the default action — is the estimated utility of it, weighted by its probability distribution — that is, the projection of the joint probability distribution for this particular action. For the formalisation of this, we direct the interested reader to Russell and Wefald's paper [11].

Until now, we have not taken into account the fact that our agent is situated in a real-time environment. We represent this dependence by a *cost of time*: we distinguish between *intrinsic* utility

[2] In section 4 we refer to the fact that our work is closely related to the research of Markov Decision Processes (MDP's) [2]: we can relate complete and partial computations as used here to finite and infinite horizons in MDP's.

$\hat{U}_I(E)$ — a time-independent utility, and *total* utility — the intrinsic utility corrected with a temporal discount factor. Until now we have been only concerned with the intrinsic utility. A cost function $C : A_{int} \rightarrow \mathbb{R}$ denotes the difference between the intrinsic and total utility. Assuming the existence of some implementation of $C$, the estimated utility of an action $a_d$ after some internal action $d$ is then

$$\hat{U}_{\mathbf{e}}(\mathcal{N}(e_{now}, [d; a_d])) = \hat{U}_I(\mathcal{N}(e_{now}, [a_d])) - C(d) \qquad (8)$$

which expresses that the utility of $c$ is its intrinsic utility minus the cost of performing $d$; this thus corresponds with the total utility of $a_d$. Internal actions only affect the agent's internal state and therefore $C(d)$ only depends on its own length $\mid d \mid$. A function $TC : \mathbb{R} \rightarrow \mathbb{R}$ then expresses the time cost of an internal action, taking as input the length of an action and outputting the time cost of it. Then (8) can be rewritten as

$$\hat{U}_{\mathbf{e}}(\mathcal{N}(e_{now}, [d; a_d])) = \hat{U}_I(\mathcal{N}(e_{now}, [a_d])) - TC(\mid d \mid). \qquad (9)$$

Intuitively, we can define the value of an internal action $d$ as the difference between the *benefit* — the utility of the external action $a_d$ as resulting from $d$ — minus its *cost* — the time it takes to perform $d$. Russell and Wefald show this can be formalised by rewriting (9) as follows:

$$\hat{U}_{\mathbf{a}}(d) = \Delta(d) - TC(\mid d \mid), \qquad (10)$$

where $\Delta(d)$ expresses the estimated benefit of $d$:

$$\begin{aligned} \Delta(d) \quad = \quad & \hat{U}_{I[S;d]}(\mathcal{N}(e_{now}, [a_d])) \\ & - \hat{U}_{I[S;d]}(\mathcal{N}(e_{now}, [a_{def}])). \end{aligned}$$

Is is clear that in this model it is still not feasible in practice to assess the expected value of all continuations of a computation, because computations can be arbitrarily long. Russell and Wefald make two simplifying myopic assumptions, through which some major difficulties concerning the tractability of the model are avoided. The first assumption is that the algorithms used are *meta-greedy*, in that they consider single primitive steps, estimate their ultimate effect and choose the step appearing to have the highest immediate benefit. The second assumption is the *single-step assumption*: a computation value as a complete computation is a useful approximation to its true value as a possibly partial computation.

Having now defined both the BDI model and discrete deliberation scheduling, we discuss how the models can be integrated. The agent's control loop of our framework is the BDI agent control loop as shown in figure 1. As mentioned above, integrating the frameworks comes down to implementing the *reconsider*(...) function in this control loop. This implementation is shown in figure 2; it is based on Russell and Wefald's meta-reasoning model. The function *computeUtility*(...) computes the estimated utility of deliberation, by applying equations (7) to (10). The argument of this function is the agent's set of beliefs. These beliefs typically include the values of the necessary distributions for computing the estimates, e.g., the dynamism of the environment.

Because we use the BDI model, we treat deliberation on a very abstract level: we merely recognise deliberation as a way to alter the set of intentions. Therefore, we are only concerned with a single internal action: deliberation itself. The *reconsider*(...) function then decides whether to deliberate (indicated by *reconsider*(...) evaluating to "true"), or act (*reconsider*(...) evaluates to "false").

```
Function:  boolean reconsider(B, I)
1.
2.  get current plan π from I;
3.  a_def ← π[0];
4.
5.  U_a(a_def) ← Σ_{e∈E} P(e | a_def) U_e(e);
6.  Û_a(d) ← computeUtility(B);
7.
8.  if ((Û_a(d) − U_a(a_def)) > 0) then
9.      return true;
10. end-if
11. return false;
```

**Figure 2: The *reconsider*(...) function in the BDI agent control loop. It computes and compares the utilities of acting and deliberating, and decides, based on the outcome of this comparison, whether to deliberate or not.**

We can regard choosing to act as the default action $a_{def}$ and choosing to deliberate as the single internal action. It is clear that this relates Russell and Wefald's model to the BDI model. We are left with two questions: what should the default action $a_{def}$ be and how do we compute the utilities of choosing to deliberate versus choosing to act? We deal with these questions subsequently.

Let $\Pi$ be the set of all plans. A plan is a recipe for achieving an intention; $\pi \in \Pi$ represents a plan, consisting of actions $\pi[0]$ through $\pi[n]$, where $\pi[i] \in A_{ext}$ and $n$ denotes the length of the plan. The agent's means-ends reasoning is represented by the function $plan : \wp(B) \times \wp(I) \rightarrow \Pi$, used on line 12 in figure 1. At any moment in time, we let the default action $a_{def}$ be $\pi[0]$, where the computation of the utility of $a_{def}$ is done through equation (4). This answers the first question.

The computation of the utility of deliberation is done using Russell and Wefald's model: we estimate the utility of deliberation, based on distributions which determine how the environment changes. These distributions are necessary knowledge because the optimality of intention reconsideration depends *only* on events that happen in the environment. For now, we assume that the agent knows these distributions and that they are *static* (they do not change throughout the existence of the environment) and *quantitative*. (Because these assumptions may be considered very demanding, we explain in section 5 how we plan to adjust our model in future work to drop them.) Using these distributions and equation (6), we estimate the utility of deliberation as the difference between the utility of the outcome of the deliberation (i.e., a revised $\pi[0]$), and $a_{def}$ (i.e., the current $\pi[0]$). Situated in a real-time environment, the agent will discount the estimated utility of deliberation, based on the length of deliberating, using equation (10). The decision control algorithm DCA then prescribes to deliberate and execute the revised $\pi[0]$ if this estimate is positive, and to act — execute the current $\pi[0]$ — otherwise.

This results in a meta level control function *reconsider*(...) which enables the agent at any time to compute the utility of $\pi[0]$ and also to estimate the utility of deliberating over its intentions, and then, according to these utilities, acts (by executing $\pi[0]$) or deliberates (by reconsidering its intentions). Next, we illustrate the theory with a simple exemplar scenario.

## The Tileworld

Our exemplar domain is a simplified TILEWORLD [9], which involves a grid environment on which there are agents and holes.

```
Function:  boolean reconsider(B, I)
1.
2.  get dist_IH from B;
3.  get avedist from B;
4.  get newholes from B;
5.  get current plan π from I;
6.  a_def ← π[0];
7.
8.  U_a(a_def) ← n(dist_IH);
9.  Û_a(d) ← n(avedist/newholes);
10.
11. if ((Û_a(d) − U_a(a_def)) > 0) then
12.     return true;
13. end-if
14. return false;
```

**Figure 3: The *reconsider*(...) function for the TILEWORLD. Deliberation is considered necessary when it is expected that since the last deliberation, the current goal has disappeared or that new goals have appeared.**

Let $H$ represent the set of possible holes; an environment state is an element from the set $E = \wp(H)$ with members $e, e', e'', \ldots$. An agent can move up, down, left, right and diagonally. Holes have to be visited by the agent in order for it to gain rewards. The TILEWORLD starts in some randomly generated world state and changes over time with the appearance and disappearance of holes according to some fixed distributions — thus $H$ changes over time. The agent moves about the grid one step at a time. We let $A_{ext} = \{noop, ne, e, se, s, sw, w, nw, n\}$, where each action denotes the direction to move next and the *noop* is a null action, by executing which the agent stays still. The agent's only internal action is to deliberate, thus $A_{int} = \{d\}$. At any given time, if holes exist in the world, an agent has a single intended hole `IH` — the hole it is heading for — over which it is deliberating. If no holes exist, the agent stays still. Let $\texttt{dist}_h$ denote the distance between the agent and hole $h \in H$. Then $\texttt{mindist} = \min\{dist_h \mid h \in H\}$ denotes the distance to the hole closest to the agent. The agent's deliberation function *d* selects `IH`, based on `mindist`; the means-ends reasoning function *plan* selects a plan $\pi$ to get from the agent's current location to `IH`. For example, if the agent is currently at location $(2, 0)$ and `IH` is at $(1, 3)$, then $\pi = [s; s; sw]$. We assume that *d* and *plan* are optimal, in that *d* selects the closest hole and *plan* selects the fastest route.

According to our model, the agent must at any time choose between executing action $\pi[0]$ and deliberating. Based on the utilities of these actions, the *reconsider*(...) function decides whether to act or to deliberate. Let the utility of an environment state be the inverse of the distance from the agent to its intended hole, $n(\texttt{dist}_{\texttt{IH}})$, where $n$ is an order-reversing mapping[3]. Equation (2) then defines the utility of an external action. While in this domain, the utility of an external action is immediately known, the utility of internal actions is not immediately known, and must be estimated. In accordance with our model, we use pre-defined distributions here: the utility of an internal action is estimated using knowledge of the distribution of the appearance and disappearance of holes[4].

[3]The TILEWORLD is a domain in which it is easier to express costs (in terms of distances) rather than utilities. With an order-reversing mapping from costs to utilities, we can continue to use utilities, which fits our model better.

[4]Note that we do *not* let the agent know when or where holes appear, we merely give it some measure of how fast the environment changes.

| Parameter | Value/Range |
|---|---|
| world dimension | 20 |
| hole score | 10 |
| hole life-expectancy | [240,960] |
| hole gestation time | [60,240] |
| dynamism ($\gamma$) | (1,80) |
| accessibility | 20 |
| determinism | 100 |
| number of time-steps | 15,000 |
| number of trials | 25 |
| planning cost ($p$) | 0, 1, 2, or 4 |

**Table 1: Overview of the experiment parameters**

The reason for this is that the appearance and the disappearance of holes are events that cause the agent to change its intentions. For example, when the set of holes $H$ does not change while executing a plan, there is no need to deliberate; but when $H$ does change, this might mean that `IH` has disappeared or that a closer hole has appeared: reconsideration is necessary. Let `avedist` be the average distance from the agent to every location on the grid; this is a trivial computation. Let `newholes` be the estimated number of holes that have appeared since the last deliberation; this is calculated using the dynamism of the world and the gestation period of holes — the gestation period is the elapsed time in between two successively appearing holes. We deem `avedist/newholes` an appropriate estimate for the utility of deliberation.

The *reconsider*(...) function for TILEWORLD agents is shown in figure 3. We let the belief set of the agent at least consist of

$$B = \{\texttt{dist}_{\texttt{IH}}, \texttt{avedist}, \texttt{newholes}\}$$

and let the intention set be

$$I = \{\texttt{IH}\}.$$

The *reconsider*(...) function computes the utility of executing $\pi[0]$ and estimates the utility of deliberating: if

$$\texttt{dist}_{\texttt{IH}} < \frac{\texttt{avedist}}{\texttt{newholes}}$$

the agent acts, and if not, it deliberates.

As mentioned above, this does not guarantee optimal behaviour, but it enables the agent to determine its commitment to a plan autonomously. We empirically evaluate our framework in the next section, and demonstrate an agent using such an intention reconsideration scheme performs better than when a level of commitment is hardwired into the agent.

## 3. EXPERIMENTAL RESULTS

In this section we present a series of simulations in which we utilise a TILEWORLD environment — as described above — inhabited by a single agent. The experiments are based on the methodology described in [12]. (We repeated the experiments described in [12] to ensure that our results were consistent; these experiments yielded identical results, which are omitted here for reasons of space.)

In [12], the performance of a range of intention reconsideration policies were investigated in environments of different structure.

Because we use similar experimental parameters here, we briefly summarise the parameters of the [12] experiments. Environments were varied to the degree of *dynamism*, denoted by $\gamma$ — the rate of change of the environment independent of the activities of the agent, *accessibility* — the extent to which an agent has access to the state of the environment, and *determinism* — the degree of predictability of the system behaviour for identical system inputs. Here, we assume the environment is fully accessible and deterministic. With respect to agent properties, the *planning cost p* and *reconsideration strategy* were varied. The planning cost represents the time cost of planning, i.e., the number of time-steps required to form a plan, and took values 0, 1, 2 and 4. Two reconsideration strategies were investigated: a *bold* agent never replans while executing a plan, and a *cautious* agent replans every time before executing an action. For these experiments, we introduce an *adaptive* agent, which figures out for itself how committed to its plans it should be. The decision mechanism of this agent is based on the theory as described in section 2.

We measured three dependent variables: the *effectiveness* $\epsilon$ of an agent is the ratio of the actual score achieved by the agent to the score that could in principle have been achieved; *commitment* $\beta$ is expressed as how many actions of a plan are executed before the agent replans[5]; the *cost of acting c* is the total number of actions the agent executes[6].

In table 1 we summarise the values of the experimental parameters ($[x, y]$ denotes a uniform distribution from $x$ to $y$ and $(x, y)$ denotes the range from $x$ to $y$).

## Results

The experiments for dynamism resulted in the graphs shown in figure 4. In figure 5.a we plotted commitment $\beta$ of an adaptive agent, varying dynamism, with a planning cost $p$ of 0, 1, 2 and 4, respectively[7]. The commitment of a cautious and bold agent are of course constantly 0 and 1 respectively. In figure 5.b, the cost of acting $c$ is plotted for the three agents for $p = 4$. The cost of acting represents the number of time steps that the agent performed an action. We refer to a plot of effectiveness $\epsilon$ as in figure 4 as an *effectiveness curve*. We continue this section with an analysis of these results.

## Analysis

For the bold and cautious agent, we obtained the same results as from the series of experiments as described in [12]. When planning is free ($p = 0$) as in graph 4.a, it was shown in the experiments in [12] that a bold agent outperforms a cautious agent. This out-performance was, however, negligible in a very dynamic environment. In these experiments, it is very clear that in a static world (where dynamism is low), a bold agent indeed outperforms a cautious agent. But from some point onwards (dynamism is approximately 28), a cautious agent outperforms a bold one. This observation agrees with the natural intuition that it is better to stick with a plan as long as possible if the environment is not very likely to change much, and to drop it quickly if the environment changes frequently. More importantly, when planning is free, the adaptive agent outperforms the other two agents, independent of the dynamism of the world. This means that adaptive agents indeed outsmart bold and cautious agents when planning is free.

As planning cost increases, the adaptive agent's effectiveness gets very close to the bold agent's effectiveness. However, there is more to this: when we take the cost of acting into account, we observe that the adaptive agent's acting cost is much lower. Considering these costs, we can safely state that the adaptive agent keeps outperforming the bold agent. When planning is expensive ($p = 4$) as in graph 4.d, the cautious agent suffers the most from this increase in planning cost. This is because it only executes one step of its current plan and after that, it immediately plans again. It thus constructs plans the most often of our types of agents. We also observe that the bold agent and adaptive agent achieve a similar effectiveness. But again, as shown in 4.d, the adaptive agent's acting costs are much lower.

We included the level of commitment for an adaptive agent, as shown in figure 4.d, to demonstrate how commitment is related to the dynamism of the world. Some interesting observations can be made here. Firstly, we see that planning cost has a negative influence on commitment — as planning cost increases, the level of commitment decreases. The reason for this is that the cost of planning is the time it takes to plan; as this value increases, more events can take place in the world during the planning period, and it becomes more attractive to replan earlier rather than later. Secondly, we see that if dynamism increases, the level of commitment decreases. This can be easily explained from the intuition, as described above, that in a very fast changing world, it is better to reconsider more often in order to be effective.

# 4. RELATED WORK

The notion of commitment has been widely studied in the agent literature. Two different fields of research can be easily distinguished: a *single agent* and *multi agent* case. In both fields, only recently investigation has been initiated on run-time decision making. Until now, the majority of previous work presupposed the problem as a design-time one. Whereas in the single agent field, commitment is mostly referred to as *a deliberation and action trade-off*, in the multi agent field it is a *"pledge" to undertake a specified course of action* (from [4]) and, obviously, more related to the social property of agents.

Our work originates in the research on the role of intentions in the deliberation process of practical reasoning agents, which was initiated by Bratman et al. [3]. Since then, Pollack has investigated the issue of commitment in single practical reasoning agent systems by means of *overloading intentions* [8]. The idea behind overloading is closely related to the filter override mechanism in the initial BDI agent model as described in [3]: the agent makes use of opportunities that arise in the world, based on the intentions it has already adopted. This research is more focused on the optimal usage of the current set of intentions, rather than the actual process of deliberating over intentions.

More recently, Veloso et al. [13] used a *rationale based monitoring* (RBM) method to control of reasoning in intentional systems. The idea behind RBM is that plan dependent features of the world are monitored during plan execution; if a feature changes value, this is reason to replan. It must be noted here that the determination of such monitors is a very domain-dependent task and this might hinder the way to a more general domain-independent theory of control of reasoning.

[5] Commitment for a plan $\pi$ with length $n$ is $(k-1)/(n-1)$, where $k$ is the number of executed actions. Observe that commitment defines a spectrum from a cautious agent ($\beta = 0$, because $k = 1$) to a bold one ($\beta = 1$, because $k = n$).

[6] Whereas cost of acting can easily be factored into the agent's effectiveness, we decided to measure it separately in order to maintain clear comparability with previous results.

[7] The collected data was smoothed using a Bezier curve in order to get these commitment graphs, because the commitment data showed heavy variation resulting from the way dynamism is implemented. Dynamism represents the acting ratio between the world and the agent; this ratio oscillates with the random distribution for hole appearances, on which the adaptive agent bases its commitment.

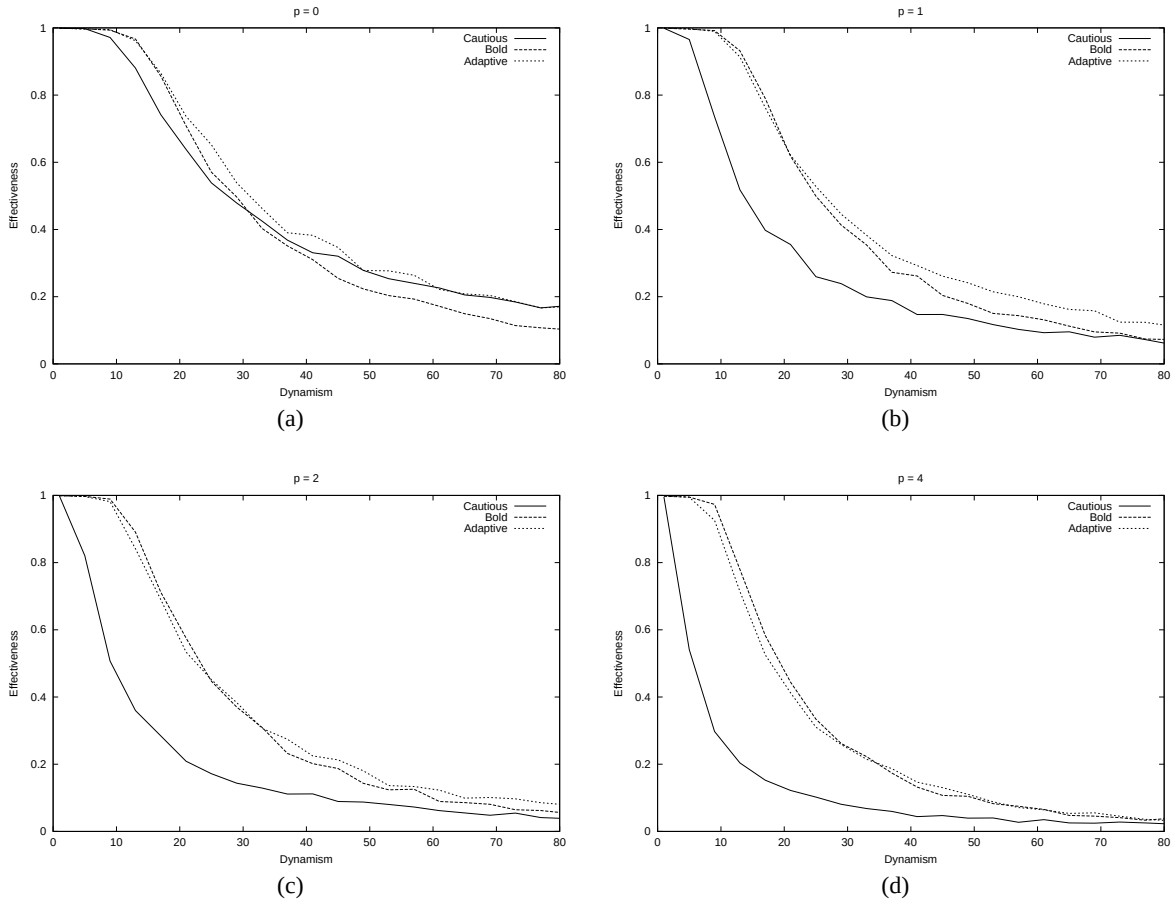


**Figure 4: Performance of a cautious, bold and adaptive agent. Effectiveness is measured as a result of a varying degree of dynamism of the world. The four panels represent the effectiveness at different planning costs (denoted by *p*), ranging from 0 to 4.**

Finally, research on Partially Observable Markov Decision Processes (POMDP's) by Kaelbling et al. [5] is relevant to our work, since it offers a formal framework which can be used in conjunction with our model in order to generalise it to a more inclusive model, applicable to a wider range of decision problems.

## 5. DISCUSSION

In this paper we presented a preliminary formal model for BDI agents that are able to determine their own intention reconsideration strategy based on future goals and arising opportunities. By applying the model in a simple TILEWORLD scenario, we have shown that an agent using the model yields better results than agents with fixed strategies. This empirical evaluation demonstrates the benefit of flexibility in reasoning for agents situated in dynamic and open environments.

While the BDI model enables the agent to direct its future deliberations and actions by adopting certain intentions, it is crucial for the agent to determine for itself how committed it is to these intentions. This has to be done autonomously, because commitment changes depending on how the environment changes. Our agent chooses a level of commitment according to the current state of the environment, and bases this choice on estimates from distributions of how the environment changes. An example of such a distribution in the TILEWORLD is the frequency with which holes appear and disappear during the existence of the world. Currently, the system is limited in the way that these distributions are given to the agent and they are assumed to be static. Future work will include research on these issues: we propose research in which the agent obtains the distributions itself using reinforcement learning, and we have initiated empirical research which will demonstrate how the level commitment changes under various kinds of distributions.

The empirical investigation we conducted showed interesting results. Firstly, an agent's effectiveness increases as its reasoning mechanism is more flexible. Secondly, when the environment's rate of change increases, the level of commitment decreases. This corresponds to the intuition that intentions are more liable to reconsideration when the environment changes fast. Finally, the experiments showed that as planning takes longer, the level of commitment decreases. This can be explained as follows: when it takes longer to plan, the probability that the environment changes during planning increases. In order to cope with this, one needs to replan sooner rather than later.

This work is part of research that aims to determine efficient mechanisms for the control of reasoning in environments of different structure. In future work we hope to extend the framework to cover richer environments in terms of realism and structure: we intend to deliver an agent that is flexible and autonomous with respect to open and unpredictable environments.

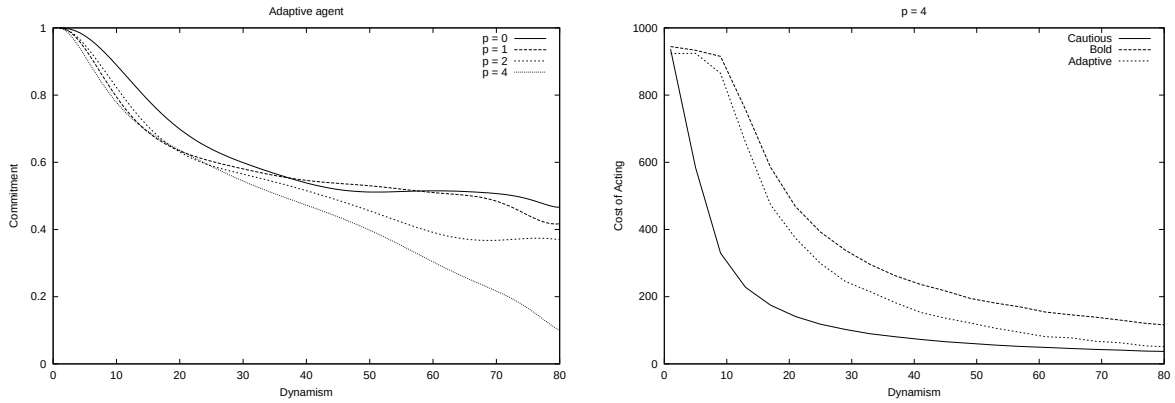


**Figure 5: Commitment for an adaptive agent and Cost of Acting for a cautious, bold and adaptive agent. In (a), the commitment level is plotted as a function of the dynamism of the world for an adaptive agent with planning cost (denoted by *p*) of 0, 1, 2 and 4. In (b), the cost of acting — the number of time steps that the agent moves – is plotted as a function of the dynamism of the world for a cautious, bold and adaptive agent with a planning cost of 4.**

# 6. REFERENCES


[1] M. Boddy and T. Dean. Solving time-dependent planning problems. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence (IJCAI-89)*, pages 979–984, Detroit, MI, 1989.

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

[3] M. E. Bratman, D. J. Israel, and M. E. Pollack. Plans and resource-bounded practical reasoning. *Computational Intelligence*, 4:349–355, 1988.

[4] N. R. Jennings. Commitments and conventions: The foundation of coordination in multi-agent systems. *The Knowledge Engineering Review*, 8(3):223–250, 1993.

[5] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101:99–134, 1998.

[6] D. Kinny and M. Georgeff. Commitment and effectiveness of situated agents. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI-91)*, pages 82–88, Sydney, Australia, 1991.

[7] P. Maes and D. Nardi, editors. *Meta-Level Architectures and Reflection*. Elsevier Science Publishers B.V.: Amsterdam, The Netherlands, 1988.

[8] M. E. Pollack. Overloading intentions for efficient practical reasoning. *Noûs*, 25(4):513–536, 1991.

[9] M. E. Pollack and M. Ringuette. Introducing the Tileworld: Experimentally evaluating agent architectures. In *Proceedings of the Eighth National Conference on Artificial Intelligence (AAAI-90)*, pages 183–189, Boston, MA, 1990.

[10] A. S. Rao and M. P. Georgeff. An abstract architecture for rational agents. In C. Rich, W. Swartout, and B. Nebel, editors, *Proceedings of Knowledge Representation and Reasoning (KR&R-92)*, pages 439–449, 1992.

[11] S. Russell and E. Wefald. Principles of metareasoning. *Artificial Intelligence*, 49(1-3):361–395, 1991.

[12] M. C. Schut and M. Wooldridge. Intention reconsideration in complex environments. In M. Gini and J. Rosenschein, editors, *Proceedings of the Fourth International Conference on Autonomous Agents (Agents 2000)*, pages 209–216, Barcelona, Spain, 2000.

[13] M. Veloso, M. Pollack, and M. Cox. Rationale-based monitoring for planning in dynamic environments. In R. Simmons, M. Veloso, and S. Smith, editors, *Proceedings of the Fourth International Conference on Artificial Intelligence Planning Systems (AIPS 1998)*. AAAI Press, 1998.

[14] M. Wooldridge. The computational complexity of agent design problems. In E. Durfee, editor, *Proceedings of the Fourth International Conference on Multi-Agent Systems (ICMAS 2000)*. IEEE Press, 2000.

[15] M. Wooldridge. *Reasoning about Rational Agents*. The MIT Press: Cambridge, MA, 2000.

[16] M. Wooldridge and S. D. Parsons. Intention reconsideration reconsidered. In J. P. Müller, M. P. Singh, and A. S. Rao, editors, *Intelligent Agents V (LNAI Volume 1555)*, pages 63–80. Springer-Verlag: Berlin, Germany, 1999.