Intention Reconsideration Reconsidered

Michael Wooldridge and Simon Parsons

Department of Electronic Engineering
Queen Mary and Westfield College
University of London
London E1 4NS, United Kingdom
{M.J.Wooldridge, S.D.Parsons}@elec.qmw.ac.uk

Abstract. In this paper, we consider the issue of designing agents that successfully balance the amount of time spent in reconsidering their intentions against the amount of time spent acting to achieve them. Following a brief review of the various ways in which this problem has previously been analysed, we motivate and introduce a simple formal model of agents, which is closely related to the well-known belief-desire-intention model. In this model, an agent is explicitly equipped with mechanisms for deliberation and action selection, as well as a meta-level control function, which allows the agent to choose between deliberation and action. Using the formal model, we define what it means for an agent to be optimal with respect to a task environment, and explore how various properties of an agent's task environment can impose certain requirements on its deliberation and meta-level control components. We then show how the model can capture a number of interesting practical reasoning scenarios, and illustrate how our notion of meta-level control can easily be extended to encompass higher-order meta-level reasoning. We conclude with a discussion and pointers to future work.

1 Introduction

Much of the research activity from the intelligent agent community in the mid-to-late 1980s was focussed around the problem of designing agents that could achieve an effective balance between *deliberation* (the process of deciding *what to do*) and *means-ends reasoning* (the process of deciding *how to do it*) [2]. One particularly successful approach that emerged at this time was the *belief-desire-intention* (BDI) paradigm [5, 2, 10, 13]. The development of the BDI paradigm was to a great extent driven by Bratman's theory of (human) practical reasoning [1], in which *intentions* play a central role. Put crudely, since an agent cannot deliberate indefinitely about what courses of action to pursue, the idea is it should eventually *commit* to achieving certain states of affairs, and then devote resources to achieving them. These chosen states of affairs are intentions, and once adopted, they play a central role in future practical reasoning [1, 3].

A major issue in the design of agents that are based upon models of intention is that of when to *reconsider* intentions. An agent cannot simply maintain an intention, once adopted, without ever stopping to reconsider it. From time-to-time, it will be necessary to check, (for example), whether the intention has been achieved, or whether it is believed to be no longer achievable [3]. In such situations, it is necessary for an agent to deliberate over its intentions, and, if necessary, to *change focus* by dropping existing intentions and adopting new ones. Kinny and Georgeff undertook an experimental study of different intention reconsideration strategies [6]. They found that *dynamic* environments — environments in which the rate of world change was high — tend to favour *cautious* intention reconsideration strategies, i.e., strategies which frequently stop to reconsider intentions. Intuitively, this is because although such agents incur the costs of deliberation, they do not waste effort attempting to achieve intentions that are no longer viable, and are able to exploit new opportunities as they arise. In contrast, *static* environments — in which the rate of world change is low — tend to favour *bold* intention reconsideration strategies, which only infrequently pause to reconsider intentions.

Our aim in this paper is to consider the question of when to deliberate (i.e., to reconsider intentions) *versus* when to act from a formal point of view, in contrast to the experimental standpoint of Kinny and Georgeff [6]. We develop a simple formal model of practical reasoning agents, and investigate the behaviour of this model in different types of task environment. In this agent model, (which is very closely related to the BDI model [5, 2, 10]) an agent's internal state is characterised by a set of beliefs (information that the agent has about its environment) and a set of intentions (commitments the agent has made about what states of the world to try and achieve). In addition, an agent has a deliberation function, which allows it to reconsider and if necessary modify its intentions, and an action function, which allow it to act towards its current intentions. These functions are mediated by a *meta-level control* function. The purpose of the meta-level control function is simply to choose between deliberation and action. The meta-level control function thus acts somewhat like the interpreter in the PRS [5], but more closely resembles the meta-plans that are used to manage an agent's intention structures in the PRS.

The remainder of this paper is structured as follows. In section 2 we present our formal model of agents, and we define what it means for an agent to be optimal with respect to a particular *task environment*. In section 3, we investigate what it means for a task environment to be *real time*, and discuss the relationships that must hold between an agent's meta-level control and deliberation components in order for an agent to act optimally in such task environments. In particular, we define notions of soundness and completeness for meta-level control and deliberation strategies, and show that an optimal meta-level control function must be sound and complete with respect to a deliberation function in an important class of real-time task environments. In section 4, we show how our formal framework can capture a number of typical practical reasoning scenarios (taken from [2]). In section 5, we generalise our model of meta-level control to capture *higher-order* meta-level reasoning strategies (intuitively, strategies to determine what sort of meta-level reasoning function to use), and we integrate these with our agent model. Finally, in section 6, we present some conclusions and issues for future work.

2 Agents and Environments

In this section, we formalise a simple model of practical reasoning agents and the environments they occupy, and define what we mean by a *run* or *history* of an agent in an

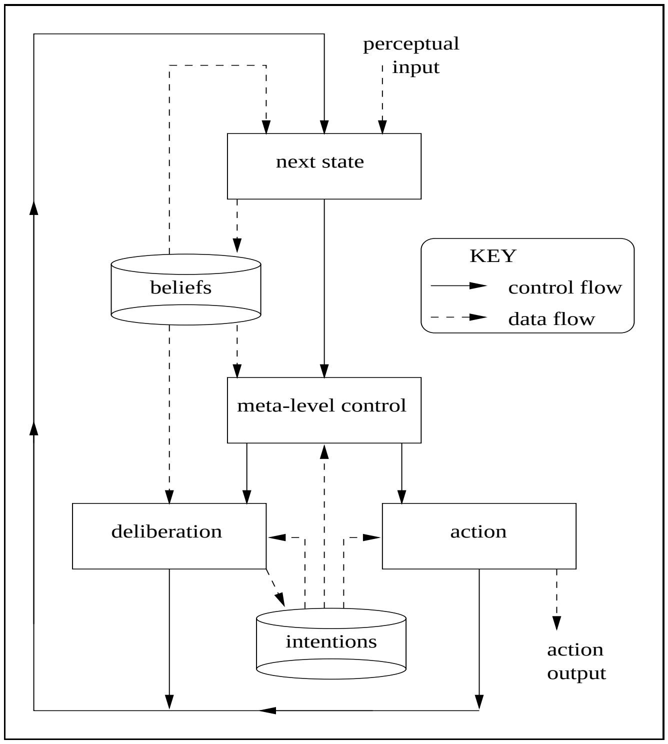

Fig. 1. Meta-level control, deliberation, and action in an architecture for practical reasoning agents.

environment. An overview of our agent model is given in Figure 1.

Before discussing this model in detail, it is important to make several points clear. First, the architecture is emphatically *not* intended to be a proposal for a new implementable agent architecture in the sense of the PRS, INTERRAP, and so on [15]. Rather, it is intended to be an *abstraction* of the key functional components of the BDI architecture, which we find to be useful for analysis purposes. Second, note that although the architecture is closely related to the BDI model of agency, it also has some key differences. Perhaps most importantly, the reader will note that *desires* are missing. Desires in a BDI agent are essentially “options” or “possibilities” available to the agent. The agent chooses and commits to a subset of its desires, which then become intentions. Desires are thus used by an agent during the process of intention formation, and in particular, they are not a key component of the intention reconsideration process, which is our primary object of study in this paper. Hence they are subsumed within the deliberation component of our architecture. If one were to actually *implement* the deliberation component of our architecture, then it might well be useful to employ desires — but at our analysis level, they do not play any useful role.

Returning to Figure 1, our agents have two main data structures: a *belief set* and an *intention set*. An agent's beliefs represent information that the agent has about its environment. In implemented agent systems (such as PRS [5]), beliefs are often represented symbolically, as PROLOG-like facts, but they may simply be variables of a PASCAL-like programming language. However they are represented, beliefs correspond to an agent's *information state*. Let B be the set of all beliefs. For the most part, the contents of B will not be of concern to us here. However, it is often useful to suppose that B contains formulae of some logic, so that, for example, it is possible to determine whether two beliefs are mutually consistent or not. An agent's actions at any given moment are guided by its *intention set*, which represents its *focus*: the "direction" of its activities. Intentions may be thought of as states of affairs that an agent has committed to bringing about. Formally, let I be the set of all intentions. Again, we are not concerned here with the contents of I. As with beliefs, however, it is often useful to assume that intentions are expressed in some sort of logical language. An agent's *local state* will then be a pair (b, i), where $b \subseteq B$ is a set of beliefs, and $i \subseteq I$ is a set of intentions. The local state of an agent is its internal state: a snapshot of its information and focus at any given instant. Let $L = \wp(B) \times \wp(I)$ be the set of all internal states. We use l (with annotations: $l', l_1, \ldots$) to stand for members of L. If $l = (b, i)$, then we denote the belief component of l by b_l, and the intention component of l by i_l.

Agents do not operate isolation: they are situated in *environments*; we can think of an agent's environment as being everything external to the agent. (This external component may, of course, include other agents; we leave the exploration of this possibility to future work.) We assume that the environment external to the agent may be in any of a set $E = \{e, e', \ldots\}$ of states.

Together, an agent and its environment make up a *system*. The *global* state of a system at any time is thus a pair containing the state of the agent and the state of the environment. Formally, let $G = E \times L$ be the set of all such global states. We use g (with annotations: $g, g', \ldots$) to stand for members of G.

2.1 Choice, Deliberation, and Action

As Figure 1 illustrates, our agents have four main components, which together generate their behaviour: a *next-state function*, a *meta-level control function*, a *deliberation function*, and an *action function*. The *next state* function can be thought of as a *belief revision function*. On the basis of the agent's current state and the state of the environment, it determines a new set of beliefs for the agent, which will include any new information that the agent has perceived. An agent's next-state function thus realises whatever *perception* the agent is capable of. Formally, a next-state function is a mapping $\mathcal{N} : E \times \wp(B) \to \wp(B)$.

The next component in our agent architecture is meta-level control. The idea here is that at any given instant, an agent has two choices available to it. It can either *deliberate* (that is, it can expend computational resources deciding whether to change its focus), or else it can *act* (that is, it can expend resources attempting to actually achieve its current intentions). Note that we assume the only way an agent can *change* its focus (i.e., modify its intentions) is through explicit deliberation. To represent the choices (deliberation versus action) available to an agent, we will assume a set $C = \{d, a\}$, where d denotes deliberation, and a denotes action. The purpose of an agent's *meta-level control function* it to choose between deliberation and action. If it chooses to deliberate, then the agent subsequently deliberates; if its chooses to act, then the agent subsequently acts. Formally, we can represent such strategies as functions $\mathcal{M} : L \to C$, which on the basis of the agent's internal state, decides whether to deliberate (d) or act (a).

The *deliberation* process of an agent is represented by a function that, on the basis of an agent's internal state (i.e., its beliefs and intentions), determines a new set of intentions. Formally, we can represent this deliberative process via a function $\mathcal{D} : L \to \wp(I)$.

If an agent decides to act, rather than deliberate, then it is acting to achieve its intentions. To do so, it must decide *which* action to perform. The action selection component of an agent is essentially a function that, on the basis of the agent's current state, returns an action, which represents that which the agent has chosen to perform. Let $Ac = \{\alpha, \alpha', \ldots\}$ be the set of actions. Formally, an action selection function is a mapping $\mathcal{A} : L \to Ac$.

Finally, we define an agent to be a 5-tuple $(\mathcal{M}, \mathcal{D}, \mathcal{A}, \mathcal{N}, l_0)$, where $\mathcal{M}$ is a meta-level control function, $\mathcal{D}$ is a deliberation function, $\mathcal{A}$ is an action selection function, $\mathcal{N}$ is a next-state function, and $l_0 \in L$ is an *initial state*.

Before proceeding any further, we state some assumptions upon which later results depend. First, note that although we choose to abstractly model the components of an agent as functions, they will be ultimately be implemented by programs of some kind. If f is a program, then we write $cost_f$ for the *time cost* of f. The idea is that if f has time cost $O(g(n))$ and f' has time cost $O(h(n))$, where $O(g(n)) > O(h(n))$, then $cost_f > cost_{f'}$. We assume that the cost of deliberation is approximately equal to the cost of acting (i.e., $cost_{\mathcal{D}} \simeq cost_{\mathcal{A}}$). Second, we assume the cost of meta-level control is very much smaller than the cost of deliberation (i.e., $cost_{\mathcal{M}} \ll cost_{\mathcal{D}}$).

2.2 Runs

Recall that an agent is situated in an environment, and that such an environment may be in any of a set E of states. In order to represent the effect that an agent's actions have on an environment, we introduce a *state transformer* function, τ (cf. [4, p154]). The idea is that τ takes as input an environment state $e \in E$ and an action $\alpha \in Ac$, and returns the environment state that would result from performing α in e. Thus $\tau : E \times Ac \to E$. We are implicitly assuming that environments are *deterministic*: there is no uncertainty about the result of performing an action in some state [11, p46]. In addition, we assume that the only way an environment state can change is through the performance of an action on the part of an agent (i.e., the environment is *static* [11, p46]). Dropping these assumptions is not particularly problematic and does not alter any of our results, although it does make the formalism somewhat more convoluted. We leave the reader to make the required modifications. Formally, we define an environment *Env* to be a triple (E, τ, e_o), where E is a set of environment states as above, τ is a state transformer function, and $e_0 \in E$ is the initial state of *Env*.

A *run* of an agent/environment system can be thought of as an infinite sequence:

$$r : g_0 \xrightarrow{c_0} g_1 \xrightarrow{c_1} g_2 \xrightarrow{c_2} g_3 \xrightarrow{c_3} \cdots \xrightarrow{c_{u-1}} g_u \xrightarrow{c_u} \cdots$$

In such a run, g_0 is the initial state of the system (comprised of the initial state of the environment and the initial state of the agent) and $c_0 \in C$ is the *choice* dictated by the agent's meta-level control function on the basis of it's initial state. The state $g_1 = (e_1, l_1)$ is that which results after the agent has made its choice c_0. If the agent chose to *act* (that is, if $c_0 = a$), then $e_1 = \tau(e_0, \mathcal{A}(l_0))$ and $l_1 = (\mathcal{N}(e_0, b_{l_0}), i_{l_0})$, that is, the environment state e_1 is that which results from the agent performing its chosen action in the initial state, and the internal state l_1 is that which results from the agent updating its beliefs via its belief revision function and not changing its intentions (since it did not deliberate).

If, however, the agent chose to *deliberate* at time 0 (i.e., if $c_0 = d$) then $e_1 = e_0$ (i.e., the environment remains unchanged, since the agent did not act), and $l_1 = (\mathcal{N}(e_0, b_{l_0}), \mathcal{D}(l_0))$ (i.e., the agent's beliefs are updated as in the previous case, and the agent's intentions are updated through its deliberation function $\mathcal{D}$.

Formally, an infinite sequence $(g_0, g_1, g_2, \ldots)$ over G represents a run of an agent $Ag = (\mathcal{M}, \mathcal{D}, \mathcal{A}, \mathcal{N}, l_0)$ in an environment $Env = (E, \tau, e_0)$ iff $g_0 = (e_0, l_0)$ and $\forall u \in I\!N$, we have

$$g_{u+1} = \begin{cases} (e_u, (\mathcal{N}(e_u, b_{l_u}), \mathcal{D}(l_u))) & \text{if } \mathcal{M}(l_u) = d \\ (\tau(e_u, \mathcal{A}(i_{l_u})), (\mathcal{N}(e_u, b_{l_u}), i_{l_u})) & \text{if } \mathcal{M}(l_u) = a. \end{cases}$$

We will denote by $r(Ag, Env)$ the run of agent Ag in environment Env, and let Run be the set of all possible runs.

2.3 Optimal Behaviour

In order to express the *value*, or *utility* of a run, we introduce a function $V : Run \to I\!R$, which assigns real numbers indicating "payoffs" to runs. Thus V essentially captures a standard decision-theoretic notion of utility. We will assume that there is some upper bound to the utility that V assigns to a run, so there will always be one or more "optimal" runs. The function V represents a *performance measure* against which an agent will be measured.

A *task environment* is defined to be a pair (Env, V), where Env is an environment, and $V : Run \to I\!R$ is a utility function. We say an agent Ag is *optimal* with respect to a task environment (Env, V) if there is no agent Ag' such that $V(r(Ag', Env)) > V(r(Ag, Env))$. Again, this is in essence the by-now standard notion of an optimal agent (cf. [12, p583]).

Viewed at its most abstract, an agent is simply an action selection or decision-making function, which maps perceptual inputs to actions [11, p34]. The architectural components of an agent — its meta-level control function, deliberation, action, and next-state function — are there *in the service* of this decision making. An obvious question is therefore whether or not we can define what it means for such a component to be optimal. Let us consider the case of the meta-level control. Suppose that in some situation, the meta-level control function chose to deliberate rather than act, and as a consequence, the agent lost some utility. (Imagine that the agent was about to be hit by a speeding car, and instead of choosing to jump, chose to deliberate about which way to jump.) Then clearly, the meta-level control function was sub-optimal in this case — it would have done better by choosing differently. This leads us to the following definition: a meta-level control function $\mathcal{M}$ is *sub-optimal* if there is some other meta-level control function $\mathcal{M}'$ such that if the agent used $\mathcal{M}'$ instead of $\mathcal{M}$, it would obtain a higher utility. Formally, if $(\mathcal{M}, \mathcal{D}, \mathcal{A}, \mathcal{N}, l_0)$ is an agent, then $\mathcal{M}$ if optimal (with respect to (Env, V), $\mathcal{D}$, $\mathcal{A}$, and $\mathcal{N}$) if there is no $\mathcal{M}'$ such that $V(r(\mathcal{M}', \mathcal{D}, \mathcal{A}, \mathcal{N}, l_0), Env) > V(r(\mathcal{M}, \mathcal{D}, \mathcal{A}, \mathcal{N}, l_0), Env)$. In a similar way, we can define optimality for $\mathcal{D}$, $\mathcal{A}$, and $\mathcal{N}$ — the details are left to the reader. Notice that optimality of a component is defined not only with respect to a task environment, but also with respect to the other components of an agent. The following theorem captures the relationship between optimality for an agent and the optimality of its components.

Theorem 1 *If an agent is optimal with respect to some task environment, then the components of that agent are mutually optimal.*

Proof. Suppose $Ag = (\mathcal{M}, \mathcal{D}, \mathcal{A}, \mathcal{N}, l_0)$ is globally optimal with respect to (Env, V), but that one component is sub-optimal. Assume this component is $\mathcal{M}$ (the cases for $\mathcal{D}$, $\mathcal{A}$, or $\mathcal{N}$ are identical). Then $V(r(\mathcal{M}', \mathcal{D}, \mathcal{A}, \mathcal{N}, l_0), Env) > V(r(\mathcal{M}, \mathcal{D}, \mathcal{A}, \mathcal{N}, l_0), Env)$ for some $\mathcal{M}'$ such that $\mathcal{M}' \neq \mathcal{M}$. But in this case, Ag is not optimal with respect to (Env, V), which is a contradiction.

Notice that the implication in this theorem cannot be strengthened to a biconditional: the fact that the components of an agent are mutually optimal does imply that the agent is itself optimal. We can think of agents that have mutually optimal components but that are globally sub-optimal as having achieved a kind of local maxima: an optimality of sorts, but not the best that could be achieved.

To make the concept of a valuation function and task environment concrete, we consider the Tileworld scenario, introduced by Pollack and Ringuette [9], and used by Kinny and Georgeff in their investigation into agent commitment strategies [6]. In this scenario, the environment is a two-dimensional "grid world". The agent is situated in this grid world, and can move around it in single steps. The grid world is also occupied by a number of randomly distributed blocks, and holes into which an agent can shove blocks. An agent does this by moving around the world, pushing blocks ahead of it. The "optimal" agent is the one that, on average, maximises the number of blocks shoved into holes. The valuation function V_{TW} for the Tileworld can simply be defined as $V_{TW}(r) = blocks(r)/unsuccessful(r)$ where $blocks(r)$ denotes the number of blocks that were successfully shoved into holes during r and $unsuccessful(r)$ denotes the number of time steps on r during which a block was not shoved into a hole. Note that the valuation function V_{TW} ranges from 0 (the agent failed to shove any block into a hole), to 1 (a block was shoved into a hole at every time step).

An agent entering such a Tileworld could, in principle, compute an optimal plan for shoving blocks into holes, (although as a variant of the travelling salesman problem, the computation of such a plan would be NP-complete). However, decision making in the Tileworld is complicated by the fact that blocks themselves appear and disappear at random. The agent has no way of knowing in advance where holes will appear or disappear, and if it is to operate effectively, it must monitor such environmental changes, and, where appropriate, modify its course of action. We will return to the Tileworld and comment further on this issue in the following section.

For the remainder of this paper, we will be particularly concerned with the relationship between just two of the components of an agent: its meta-level control function and deliberation component. We shall therefore assume from here on that an agent's next-state function and action function are fixed and optimal.

3 Real-Time Task Environments

It should be clear that the performance of an agent is very much dependent on the nature of the task environment in which it is situated. An agent that performs badly in one task environment may do well in one that has different properties [11, p46]. An understanding of the relationship between agents and the task environments they occupy is therefore likely to be of great benefit when we come to build agents that will operate in real environments.

Arguably the most important everyday class of task environments are those that come under the banner of *real-time*. Put at its most abstract, a real-time task environment is simply one in which time plays a part in the evaluation of an agent's performance [12, p585]. It is possible to identify several different sorts of real-time task environments, for example:

– those in which the agent must bring about some state of affairs as quickly as possible — the sooner it achieves this state of affairs, the higher its payoff;
– one in which an agent is required to repeat some task, with the optimal agent being the one that repeats the task as often as possible.

Real-time task environments are problematic because, in general, if an agent is to operate successfully in such an environment, then it must successfully trade-off the amount of time it spends deliberating against the amount of time it spends acting. For if an agent deliberates indefinitely, then it will typically never achieve anything (cf. the notion of reactivity in [15])[1].

Formally defining what it means for a task environment to be real-time is not simple, since, as the examples above indicate, the concept of real-time actually encompasses a number of related properties. Rather than attempt to present such a general definition, we define a class of task environments in which *wasted effort is penalised*. We argue that this concept captures many aspects of real-time task environments.

How might an agent waste effort? There are essentially two possibilities. First, an agent is wasting effort if it is expending resources attempting to achieve the "wrong" intentions. Consider the Tileworld, discussed in the preceding section. Suppose an agent has observed some block, and has formed an intention to shove that block into a particular hole. Now if the agent is attempting to achieve this intention even when that hole

[1] It is easy to construct providential task environments, in which an optimal agent is one that always chooses to deliberate or always chooses to act. However, we argue that such task environments do not correspond to many interesting real-world situations.

has vanished, then it is in some sense wasting effort. It would do better to reconsider its intentions. A similar waste of effort occurs if an agent fails to exploit a serendipitous situation (for example when a hole appears to the side of an agent, making it possible to obtain additional utility).

A second type of wasted effort occurs if an agent has "correct" intentions, but is not acting on them — in such a situation, an agent is engaging in unnecessary deliberation. For example, suppose an agent in the Tileworld has an intention of shoving some particular block into a hole, and stops to deliberate. After deliberation, the agent's intentions are unchanged, and it continues to push the same block to the same hole. In this case, all other things being equal, the utility accorded to the agent would be less than it would have obtained by not deliberating at all (since the value $\mathit{unsuccessful}(r)$ has increased). The agent would thus have done better by simply acting instead.

In order to formally define what we mean for an agent to waste effort, we must first define what it means for an agent to have *optimal intentions*. Intuitively, an agent has optimal intentions if there is no good reason for changing them — if, given the information available to the agent, an optimal deliberation function would not choose to change them. Formally, if $(\mathcal{M}, \mathcal{D}, \mathcal{A}, \mathcal{N}, l_0)$ is an agent that is currently in state (b, i), and that is situated in task environment (Env, V), then its intention set i is optimal for $\mathcal{M}, \mathcal{A}, \mathcal{N}, l_0$ iff $\mathcal{D}((b, i)) = i$. Note that the notion of an optimal intention set is inherently *relative* to a specific agent. An intention set that is optimal for one agent may well not be optimal for another. An agent $Ag = (\mathcal{M}, \mathcal{D}, \mathcal{A}, \mathcal{N}, l_0)$ in task environment (Env, V) is then said to waste effort iff $r(Ag, \mathit{Env}) = (g_0, \ldots)$ and for some $u \in \mathbb{N}$ we have either i_{l_u} is optimal for $\mathcal{M}, \mathcal{A}, \mathcal{N}, l_0$ and $c_u = d$ or else i_{l_u} is not optimal for $\mathcal{M}, \mathcal{A}, \mathcal{N}, l_0$ and $c_u = a$. Finally, a task environment is said to penalise wasted effort iff any optimal agent for this task environment does not waste effort.

Let us now turn to the relationship between meta-level control and deliberation for task environments that penalise wasted effort. The possible interactions between meta-level control and deliberation in such task environments are summarised in Table 1 (adapted and extended from [2, p353]). Consider situation (1). In this situation, the agent does not have optimal intentions, and hence would do well to deliberate. However, it does not choose to deliberate and hence the meta-level reasoning function that chose to act was sub-optimal. In situation (2), the agent again has sub-optimal intentions, but this time chooses to deliberate, rather than act. Unfortunately, the agent's deliberation function $\mathcal{D}$ does not change focus, and is thus sub-optimal. Situation (3) is essentially the same as situation (2), but this time, the deliberation function *does* change focus. While it is clear that the meta-level reasoning function is optimal in this situation, it is not certain that the deliberation function is optimal, since we do not know what the old intentions were replaced with. However, it would certainly be sub-optimal *not* to change intentions.

In situation (4), the agent has optimal intentions, and does not choose to deliberate. Since the intentions are optimal, the meta-level control function is obviously correct not to deliberate in this situation, and is hence optimal. In situation (5), the agent has optimal intentions, but this time chooses to deliberate; the deliberation function, however, does not change focus. Hence while the meta-level control function is clearly sub-optimal, the deliberation function is optimal. Situation (6) is as situation (5), except

Situation number	Optimal intentions?	Chose to deliberate?	Changed focus?	$\mathcal{M}$ optimal?	$\mathcal{D}$ optimal?
1	No	No	—	No	—
2	No	Yes	No	Yes	No
3	No	Yes	Yes	Yes	Maybe
4	Yes	No	—	Yes	—
5	Yes	Yes	No	No	Yes
6	Yes	Yes	Yes	No	No

Table 1. Practical Reasoning Situations (cf. [2])

that this time, the deliberation function changes focus. In this case, both the meta-level control and deliberation components must be sub-optimal, since the agent wasted time deliberating, and then modified its intentions despite the fact that there is no reason to do so.

From the discussion above, we can extract the following simple principle: for task environments that penalise wasted effort, a meta-level control function should choose to deliberate *only* when its corresponding deliberation function would change focus. We will say a meta-level control function $\mathcal{M}$ is *sound* with respect to an optimal deliberation function $\mathcal{D}$ iff whenever $\mathcal{M}$ chooses to deliberate, $\mathcal{D}$ chooses to change focus (i.e., if $M(l) = d$ implies $\mathcal{D}(l) \neq i_l$). Similarly, we say $\mathcal{M}$ is *complete* with respect to $\mathcal{D}$ iff whenever $\mathcal{D}$ *would* change focus, $\mathcal{M}$ chooses to deliberate (i.e., if $\mathcal{D}(l) \neq i_l$ implies $M(l) = d$). We can easily establish the following result, which relates sound and complete meta-level control strategies to task environments that penalise wasted effort.

Theorem 2 *For task environments that penalise wasted effort, an optimal agent has a meta-level control function that is sound and complete with respect to its deliberation function.*

Proof. Assume an arbitrary agent $(\mathcal{M}, \mathcal{D}, \mathcal{A}, \mathcal{N}, l_0)$ is optimal with respect to some task environment that penalises wasted effort. We need to show that $\mathcal{M}$ is sound and complete with respect to $\mathcal{D}$. For soundness, start by assuming that $\mathcal{M}$ is not sound with respect to $\mathcal{D}$. Then for some $l \in L$, we have $\mathcal{M}(l) = d$ (the meta-level control function says deliberate) but that $\mathcal{D}(l) = i_l$ (the deliberation function does not choose to change focus). But by definition, this is a waste of effort, hence $(\mathcal{M}, \mathcal{D}, \mathcal{A}, \mathcal{N}, l_0)$ cannot be optimal, which is a contradiction, so $\mathcal{M}$ is sound. For completeness, start by assuming that $\mathcal{M}$ is not complete with respect to $\mathcal{D}$. Hence for some $l \in L$, we have $\mathcal{D}(l) \neq i_l$ but that $\mathcal{M}(l) = a$. But this is a waste of effort, hence $(\mathcal{M}, \mathcal{D}, \mathcal{A}, \mathcal{N}, l_0)$ cannot be optimal, which is a contradiction, so $\mathcal{M}$ is complete.

An optimal meta-level control function for task environments that penalise wasted effort thus has a kind of oracle for its corresponding deliberation function. One might therefore wonder what is the point of having both meta-level control *and* deliberation components, as an optimal meta-level control function need only run the deliberation function as a subroutine to see if it would change focus, and choose to deliberate just in case the deliberation function *does* change focus. Formally, such a meta-level control function would be defined as follows:

$$\mathcal{M}(l) = \begin{cases} a & \text{if } \mathcal{D}(l) = i_l \\ d & \text{otherwise.} \end{cases}$$

This would indeed be a successful strategy if the cost of the meta-level control function was approximately equal to the cost of deliberation (i.e., if $cost_{\mathcal{M}} \simeq cost_{\mathcal{D}}$). However, as we pointed out earlier, we require that the cost of meta-level control be *significantly less* than that of deliberation ($cost_{\mathcal{M}} \ll cost_{\mathcal{D}}$). Under this assumption, running the deliberation component in order to decide whether to deliberate is not an option.

4 An Example

In the previous section, we discussed the notion of a real-time task environment, and investigated the relationship between meta-level control and deliberation in such task environments. In this section, we show how four illustrative practical reasoning scenarios (introduced in [2]) can be represented within our framework. (More accurately, Bratman and colleagues give six scenarios, since there are two variants each of scenarios one and four. However, these variants are meaningless in our framework.)

4.1 Scenario One

All four scenarios are based on the following basic story: Rosie is an agent that has been assigned the task of repairing a malfunctioning VDU. As a result of some task analysis, she has decided that this might best be done by replacing the CRT (which she believes is burnt out), and so she has adopted the intentions of going to the VDU armed with a replacement CRT, and then using this new tube to fix the VDU. In the first scenario, Rosie arrives at the VDU to find that the CRT is not burnt out: the contrast has just been turned way down. She therefore has the option of fixing the VDU by adjusting the contrast. This information is sufficient for her meta-level control function to decide that it is worth deliberating, and in so doing, Rosie finds that adjusting the contrast is cheaper than replacing the CRT. She thus adopts the new intention of adjusting the contrast. She then acts, adjusting the contrast and completes her initial task.

In this, and all other scenarios, we represent Rosie's world as a set of propositions. The propositions of interest to us are summarised in Table 2. While the intended interpretation for most of these is self-evident, some require additional explanation: s is intended to capture the presence of the additional CRT in scenarios three and four; b_1 is intended to capture the fact that Rosie knows that if it is possible to fix the VDU by just adjusting the contrast then this is a better option than using the CRT she carries with her; b_2 is intended to capture the fact that rewiring the faulty CRT is the best option, and b_3 is intended to capture the fact that an additional CRT in scenarios three and four is superior to the CRT she carries with her.

In addition, we will also represent Rosie's possible intentions as propositions: see Table 2. Again, most of these are self-explanatory, but i_v is needed to capture Rosie's initial progress from wherever she picks up the first CRT to wherever the broken VDU is.

Beliefs	
w	VDU working
c	CRT burnt out
d	Contrast turned down
b_1	Adjust contrast is better
r	CRT can be fixed by re-wiring
b_2	Re-wiring is better
s	Spare VDU
b_3	Spare VDU is better

Intentions	
i_o	Fix VDU using original CRT
i_c	Fix VDU by adjusting contrast
i_r	Fix VDU by re-wiring
i_a	Fix VDU by using alternative CRT
i_v	Go to VDU

Table 2. Rosie's Possible Beliefs and Intentions

For simplicity we will assume that each of these intentions can be achieved by a single action (though each of these could equally well be a series of actions). Thus the action to achieve intention i_r is α_r, the action to achieve intention i_v is α_v, and so on.

We can now formalise Rosie's reasoning. Initially the state of the world is $e_0 = \{\neg w, \neg c, d\}$ (the VDU is not working, the CRT is not burnt out, and the contrast is turned down). Rosie's initial internal state l_0 is thus: $(\{\neg w, c, \neg d, b_1\}, \{i_v, i_o\})$. She thus begins scenario one with false beliefs, since she wrongly believes that the CRT is burned out. Note that Rosie's beliefs also include the preference information b_1. She initially has two intentions: to fix the VDU using the original CRT, and to go to the VDU.

The first part of Rosie's operation is to decide whether to deliberate or act. She chooses to act, and executes the action α_v that achieves her intention i_v, and thus arrives at the VDU. At this point she deliberates, and removes the now-achieved intention of moving to the VDU from her intention set, so that the previously adopted intention of fixing the VDU using the CRT she brought with her becomes the main focus. At this point she can identify the real state of the world, and her next-state function $\mathcal{N}$ updates her beliefs to reflect this. Her internal state becomes: $l_1 = (\{\neg w, \neg c, d, b_1\}, \{i_o\})$. The state of the external world is unchanged: $e_1 = e_0$.

Rosie again applies her meta-level control function:

$$\mathcal{M}(l) = \begin{cases} d \text{ if } \{\neg c, d, b_1\} \subseteq b_l \text{ or } \{\neg c, r, b_2\} \subseteq b_l \text{ or } \{c, s, b_3\} \subseteq b_l \\ a \text{ otherwise.} \end{cases}$$

Thus there are three situations in which she will choose to deliberate, all of which can be glossed as "there is now some reason to suspect that there is a better alternative to repair the VDU". Clearly this is just an illustrative fragment of the complete meta-level control function which is appropriate to this example. Since Rosie now believes $\neg c$, she chooses to deliberate. That is, $\mathcal{M}(l_1) = d$ since the CRT is known to not be burnt out, the contrast is known to be turned down, and it is known that adjusting the contrast gives a better means of fixing the VDU than replacing the CRT. To find the result of deliberation, we need to define $\mathcal{D}$. We have:

$$\mathcal{D}(l) = \begin{cases} \{i_c\} & \text{if } \{\neg c, d, b_1\} \subseteq b_l \\ \{i_r\} & \text{if } \{\neg c, r, b_2\} \subseteq b_l \\ \{i_a\} & \text{if } \{c, s, b_3\} \subseteq b_l \\ l_i & \text{otherwise.} \end{cases}$$

The deliberation function $\mathcal{D}$ thus decides to adjust the contrast: $\mathcal{D}(l_1) = \{i_c\}$. Note that $\mathcal{D}$ should really check that the agent has a means of adopting the intention before it decides to adopt it — if Rosie is unable to adjust the contrast (because she has the wrong kind of gripper for instance) then however good a solution this might be, there is no point in changing focus to try and achieve it. For our purposes, we can ignore this subtlety, however.

After deliberation, Rosie's internal state becomes: $l_2 = (\{\neg w, \neg c, d, b_1\}, \{i_c\})$, while the external world remains unchanged: $e_2 = e_1 = e_0$. This time $\mathcal{M}$ chooses to act, and since $\mathcal{A}(l_2) = \alpha_c$, the contrast is adjusted, which repairs the VDU. This change in the world causes Rosie to revise her beliefs about the state of the VDU and the contrast control. The final state of the environment is thus $e_3 = \{w, \neg c, \neg d\}$, while Rosie's internal state is $l_3 = (\{w, \neg c, \neg d, b_1\}, \emptyset)$.

The complete run for scenario one is thus:

$$r_1 : g_0 \xrightarrow{a_v} g_1 \xrightarrow{d} g_2 \xrightarrow{a_c} g_3$$

4.2 Scenario Two

In this scenario, Rosie arrives at the VDU to find that the CRT is not burnt out and can be fixed by re-wiring. However, this fix will only be short term, and the CRT will soon burn out anyway. This information is sufficient for Rosie's meta-level control function to decide it is not worth deliberating to see if she is able to fix the VDU by rewiring, and so she acts, replacing the CRT in line with her unchanged intention. The start this scenario is described by:

$$\begin{aligned} e_0 &= \{\neg w, \neg c, r\} \\ l_0 &= (\{\neg w, c, \neg r, \neg b_2\}, \{i_v\}) \end{aligned}$$

So, although the CRT is not burnt out and the VDU can be fixed by re-wiring (facts that Rosie initially does not know), Rosie *does* know that re-wiring is a worse option than replacing the CRT. After moving to the VDU, popping the intention stack, and revising beliefs, just as in the previous scenario, the environment state remains unchanged but Rosie's internal state is $l_1 = (\{\neg w, \neg c, r, \neg b_2\}, \{i_o\})$.

Rosie then applies her meta-level control function, and despite the fact that there is reason for her to suspect that deliberation might lead to an alternative means of repairing the VDU (a situation which is actually true), $\mathcal{M}$ returns *a* because Rosie also knows that fixing the CRT by re-wiring is a worse option than the one she has already. Thus she can reject the idea of changing her focus without going as far as establishing whether or not she can build a new plan in order to fix the VDU. Having decided to act, Rosie performs $\mathcal{A}(i_o) = \alpha_o$ and the situation becomes:

$$\begin{aligned} e_2 &= \{w, \neg c, r\} \\ l_2 &= (\{w, c, r, \neg b_2\}, \emptyset) \end{aligned}$$

The complete run for Scenario Two is thus:

$$r_2 : g_0 \xrightarrow{a_v} g_1 \xrightarrow{a_o} g_2$$

4.3 Scenario Three

In Scenario Three, Rosie arrives at the VDU to find a spare (and therefore free) CRT sitting by the terminal, but notes that the spare is inferior to the tube she brought with her. Her meta-level control mechanism therefore realises that there is no advantage to seeing if the new tube can be used, and so chooses to act. Rosie then replaces the CRT in line with her original intention. Scenario Three thus begins with the following state of affairs:

$$\begin{aligned} e_0 &= \{\neg w, c, s\} \\ l_0 &= (\{\neg w, c, \neg s, \neg b_3\}, \{i_v\}) \end{aligned}$$

As before, Rosie proceeds to the VDU and this time finds the spare tube. After belief revision, the environment state remains unchanged but Rosie's internal state becomes $l_1 = (\{\neg w, c, s, \neg b_3\}, i_0)$. This time $\mathcal{M}$ tells her to act, because the newly visible CRT is worse than the one she is carrying with her. She acts, $\mathcal{A}(l_1) = \alpha_o$ by replacing the CRT and the situation becomes:

$$\begin{aligned} e_2 &= \{\neg w, \neg c, s\} \\ l_2 &= (\{\neg w, \neg c, s, \neg b_3\}, \emptyset) \end{aligned}$$

The complete run for Scenario Three is thus:

$$r_3 : g_0 \xrightarrow{a_v} g_1 \xrightarrow{a_o} g_2$$

4.4 Scenario Four

In Scenario Four, Rosie arrives at the VDU to again find a spare CRT sitting by the terminal, and this time notes that the spare is superior to the tube she brought with her. Her meta-level control mechanism therefore realises that there is considerable advantage to seeing if the new tube can be used since the saving in the cost of the tube is greater than the cost of deliberation. So she chooses to deliberate. Deliberation results in the adoption of the intention to use the new tube, and Rosie then replaces the CRT in line with this new intention. This scenario is almost the same as the third, except that this time the "new" CRT is superior to the one that Rosie brings with her. Thus the initial situation is:

$$e_0 = \{\neg w, c, s\}$$
$$l_0 = (\{\neg w, c, \neg s, b_3\}, \{i_v\})$$

After moving to the VDU and revising beliefs, the environment is unchanged ($e_1 = e_0$) but Rosie's internal state is $l_1 = (\{\neg w, c, s, b_3\}, \{i_0\})$. This time $\mathcal{M}(l_1) = d$ and $\mathcal{D}(l_1) = \{i_a\}$. After this, the environment state again remains unchanged but Rosie's internal state is $l_2 = (\{\neg w, c, s, b_3\}, \{i_a\})$, and Rosie proceeds to act $\mathcal{A}(l_2) = \alpha_a$ giving the following global state:

$$e_3 = \{\neg w, \neg c, s\}$$
$$l_3 = (\{\neg w, \neg c, s, b_3\}, \emptyset)$$

The complete run for Scenario Four is thus:

$$r_1 : g_0 \xrightarrow{a_v} g_1 \xrightarrow{d} g_2 \xrightarrow{a_a} g_3$$

There are several points to note about this example. The first is that both $\mathcal{M}$ and $\mathcal{D}$ are optimal for the cases given. There is no set of actions which could be chosen to give a better result. The second is that it is easy to alter the example so that Rosie is not optimal. Consider what would happen in Scenario Four if she had no means of using the additional CRT (which would mean that there was no intention i_a, or, worse no action α_a for achieving i_a). $\mathcal{M}$ would choose to deliberate since the CRT is superior, but either this deliberation would not change the intentions (if there was no i_a), or when Rosie came to act on the changed intention, she would be unable to achieve that intention and would have to revert to i_o. The final point to note is that it is this consideration of intentions and actions which justifies our assumption that the time cost of $\mathcal{M}$ is less than that of $\mathcal{D}$. Deliberation will typically involve an expensive activity such as building and evaluating the quality of plans to achieve some set of alternative intentions. Although that activity might be as simple as looking to see if there is some alternative intention which can be adopted, as here, it is still an overhead.

5 Generalised Meta-Level Reasoning

In this section, we will sketch out how an agent might use *higher-order* meta-level control strategies in its architecture, and what role such strategies might play. What do we mean by a higher-order meta-level control function? Let us refer to the meta-level control strategies as described above as *first-order* meta-level strategies. Such strategies merely choose whether to deliberate or to act. A *second-order* meta-level control function can be thought of as *selecting* which first-order meta-level control function to use. For example, a second-order meta-level control function might examine the agent's beliefs to see how dynamic the agent's environment is. If it determines that the environment is highly dynamic (i.e., the rate of world change is high [6]), then it might select a cautious first-order meta-level control function — one which frequently causes the agent to deliberate. If, in contrast, the environment is relatively static (the rate of world change is low), then it might select a *bold* meta-level control function (one that favours action over deliberation).

It is easy to imagine an agent with a "tower" of such meta-level control strategies, with nth-order function selecting which function to use at level $n-1$. The idea is very similar to the use of meta-language hierarchies in meta-logic [8, 14].

We can incorporate such higher-order meta-level reasoning into our formal model with ease. First, let $MLC_1 = L \to C$ be the set of all *first-order* meta-level control strategies. These are the meta-level control strategies that we discussed above. Then define $MLC_u = L \to MLC_{u-1}$, for all $u \in I\!N$ such that $u > 1$. Thus MLC_2 is the set of all second-order meta-level control strategies, MLC_3 is the set of all third-order meta-level control strategies, and so on. An agent becomes a 5-tuple, $(\mathcal{M}_n, \mathcal{D}, \mathcal{A}, \mathcal{N}, l_0)$, where $\mathcal{M}_n$ is an nth order meta-level control function and the agent's other components are as before. Given this, we can redefine what it means for a run to represent a history of an agent in an environment. Formally, an infinite sequence $(g_0, g_1, g_2, \ldots)$ over G represents a run of an agent $Ag = (\mathcal{M}_n, \mathcal{D}, \mathcal{A}, \mathcal{N}, l_0)$ in an environment $Env = (E, \tau, e_0)$ iff $g_0 = (e_0, l_0)$ and $\forall u \in I\!N$, we have

$$g_{u+1} = \begin{cases} (e_u, (\mathcal{N}(e_u, b_{l_u}), \mathcal{D}(l_u))) & \text{if } \mathcal{M}_n(l_u)\overbrace{(l_u)\cdots(l_u)}^{n-1 \text{ times}} = d \\ (\tau(e_u, \mathcal{A}(i_{l_u})), (\mathcal{N}(e_u, b_{l_u}), i_{l_u})) & \text{if } \mathcal{M}_n(l_u)\underbrace{(l_u)\cdots(l_u)}_{n-1 \text{ times}} = a. \end{cases}$$

Notice that agents which make use of higher-order meta-level control are strictly speaking no more powerful than "ordinary" agents, as defined earlier. For every higher-order agent there is an "ordinary" agent that behaves in exactly the same way. The point is that from the point of view of an agent designer, it may make sense to divide the functionality of the agent up into different levels of meta-reasoning.

6 Conclusions

In this paper, we have investigated the relationship between the deliberation, action, and meta-level control components of a practical reasoning architecture. While this relationship has previously been investigated from an experimental perspective (particularly by Kinny [6]), we have in contrast attempted a formal analysis. We have demonstrated how it is possible to construct a simple but, we argue, realistic model of practical reasoning agents of the type investigated by Kinny and Georgeff, and we have established some basic properties of such agents when placed in different types of task environment. We have focussed in particular on real-time task environments, since these are, we believe, the most common class of real-world task environment that one encounters. Our work, which attempts an (admittedly preliminary) formal analysis of the relationship between agent and environment, is similar in spirit to that of [7].

This work was originally instigated in an attempt to relate the work of Russell and Subramanian on bounded-optimal agents (i.e., agents that perform as well as any agent can do under certain architectural constraints [12]) to the increasingly large literature on BDI agents [5, 2, 10, 13]. While this initial investigation led us into some areas we had not initially anticipated visiting, we believe that investigating the implications of bounded-optimal agents for BDI model will be an interesting research issue, and one that we hope to investigate in future work. Another issue that we hope to consider is the moving from individual agents to multi-agent systems.

Acknowledgments. This paper has benefited enormously from comments by Rogier van Eijk, Koen Hindriks, Jörg Müller, Tim Norman, Barney Pell, and Martha Pollack.

References

1. M. E. Bratman. *Intentions, Plans, and Practical Reason.* Harvard University Press: Cambridge, MA, 1987.
2. M. E. Bratman, D. J. Israel, and M. E. Pollack. Plans and resource-bounded practical reasoning. *Computational Intelligence*, 4:349–355, 1988.
3. P. R. Cohen and H. J. Levesque. Intention is choice with commitment. *Artificial Intelligence*, 42:213–261, 1990.
4. R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. *Reasoning About Knowledge.* The MIT Press: Cambridge, MA, 1995.
5. M. P. Georgeff and A. L. Lansky. Reactive reasoning and planning. In *Proceedings of the Sixth National Conference on Artificial Intelligence (AAAI-87)*, pages 677–682, Seattle, WA, 1987.
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. J. P. Müller. The right agent (architecture) to do the right thing. In J. P. Müller, M. P. Singh, and A. S. Rao, editors, *Intelligent Agents V — Proceedings of the Fifth International Workshop on Agent Theories, Architectures, and Languages (ATAL-98)*, Lecture Notes in Artificial Intelligence. Springer-Verlag, Heidelberg, 1999. In this volume.
8. D. Perlis. Meta in logic. In P. Maes and D. Nardi, editors, *Meta-Level Architectures and Reflection*, pages 37–49. Elsevier Science Publishers B.V.: Amsterdam, The Netherlands, 1988.
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. Georgeff. Decision procedures of BDI logics. *Journal of Logic and Computation*, 8(3):293–344, 1998.
11. S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach.* Prentice-Hall, 1995.
12. S. Russell and D. Subramanian. Provably bounded-optimal agents. *Journal of AI Research*, 2:575–609, 1995.
13. K. Schild. On the relationship between BDI logics and standard logics of concurrency. In J. P. Müller, M. P. Singh, and A. S. Rao, editors, *Intelligent Agents V — Proceedings of the Fifth International Workshop on Agent Theories, Architectures, and Languages (ATAL-98)*, Lecture Notes in Artificial Intelligence. Springer-Verlag, Heidelberg, 1999. In this volume.
14. R. Turner. *Truth and Modality for Knowledge Representation.* Pitman Publishing: London, 1990.
15. M. Wooldridge and N. R. Jennings. Intelligent agents: Theory and practice. *The Knowledge Engineering Review*, 10(2):115–152, 1995.