Desires and Defaults: A Framework for Planning with Inferred Goals

Richmond H. Thomason
AI Laboratory
University of Michigan
Ann Arbor, MI 48109–2210
USA

rich@thomason.org
http://www.eecs.umich.edu/~rthomaso/

November 16, 2010

Working Version of: November 16, 2010

Do not cite: completed version will be found at

http://www.eecs.umich.edu/~rthomaso/pr/practical-reason.html

Abstract

In this paper, I will describe a formalism designed to integrate reasoning about desires with planning. One motive for such a formalism is the need to create a framework for reasoning about actions and change that provides for flexible reasoning about goals.

The ideas rely on a crucial distinction between *prima facie* and all-things-considered attitudes. I model *prima facie* beliefs and desires as defaults, using the approach to non-monotonic logic known as Default Logic. All-things-considered beliefs (representing actual epistemic commitments) and all-things-considered desires (representing goals) are selected by choosing an extension that is allowed by the logic.

I show in the paper how to integrate these ideas with a planning formalism. The resulting logic, BDP, is capable of modeling a wide range of common-sense practical arguments, and can serve as a more general and flexible model for agent architectures.

1. Introduction

In this paper, I will describe a formalism designed to integrate reasoning about desires with planning. You can motivate the ideas from either direction: (1) by reflecting on the need to extend planning formalisms to allow infererred goals, or (2) by explaining the need to extend a bare logic of belief and desire to a true system of practical reasoning by adding the capability to reason about actions. I actually followed the second path, but the first is probably the best way to begin explaining things, since the starting point is more familiar.

In AI, the paradigm for deliberation about what to do conceives of planning as a search for appropriate action sequences, given certain beliefs and desires—beliefs about the initial state of affairs, and desires about the outcome state. The focus is on action dynamics, on the way actions transform states. In particular, there is no provision for reasoning about goals. Goals are given as part of a planning problem; they are never, for instance, judged to be unworthy and discarded in the course of planning.

Commonsense planning is less constrained; goals are frequently discarded in the course of many commonsense planning problems. Suppose, for instance, that I get down to planning a long-anticipated vacation. Prior to this stage I have formed specific desires about my itinerary. These are the things that I definitely would like to do, and these are the things that would become the goals of an AI-style planning solution to the problem. But as I work through the financial and scheduling details of the commonsense problem I discover that the travel cost is excessive, and the proportion of travel time is too high. The reasonable thing to do in that case is to adjust the itinerary by visiting fewer places, and that is what I do.

Goals come from desires, and desires can be impractical; they can also conflict with one another. (See, for instance, [2].) To arrive at practical recommendations, we may need to suspend some desires; but, as the travel example illustrates, we may not be able to judge which desires to discard without integrating the process of goal selection into the planning process.

To sum up: there are compelling reasons for supplementing traditional planning formalisms with a mechanism for reasoning about desires. Also, there are similarly compelling reasons for supplementing a nonmonotonic logic of beliefs and desires with a mechanism for planning. (See Section 2.5, below.) Both motives lead to the same sort of formalism: one that supplements planning with general-purpose nonmonotonic reasoning about beliefs and desires. The logic BDP presented below in Section 3 is intended as a system of this kind. I have purposely kept it as simple and generic as possible, in the hope that it can serve as a common starting point for developing more powerful and sophisticated formalisms for practical reasoning.

2. A logic of belief and desire

2.1. *Prima facie* and all-things-considered beliefs

In [8], a distinction is introduced between *skeptical* and *credulous* approaches to nonmonotonic reasoning. In general, formalisms that provide for defaults will allow sets of premises in which these defaults conflict; these cases are characterized by *multiple extensions*, theories representing different conclusion sets that could be reached from these premisses. In many reasoning applications, it is better to extract as much information from the premisses as possible, even at the risk of reaching some false conclusions. In these applications, a credulous strategy may be appropriate, in which the reasoner chooses one of many extensions.

In this paper, I will use Raymond Reiter's default logic (see [14]) as the framework for nonmonotonic resaoning, In that framework, defaults are formalized as rules. I will only consider normal defaults, rules having the form $A \hookrightarrow C$, where A and C are formulas of first order logic. In this framework, an axiomatization will consist of (1) a set M of formulas of first order logic (the monotonic axioms), and (2) a set N of defaults (the nonmonotonic part of the the axiomatization, i.e. the default rules). A *theory* for an axiomatization $\langle M, N \rangle$ is a triple $\langle M, N, E \rangle$, where E is an extension of $\langle M, N \rangle$.[1] An extension E is a first order theory—it is closed under logical consequence. But, unlike the set of consequences of a set of monotonic axioms, it needn't be unique; a single axiomatization $\langle M, N \rangle$ can have many extensions.

The three components of a default theory can be mapped in a fairly natural way to attitudes of an ideal agent. A formula in M corresponds to an immediate belief, one that—at least, in the reasoning context under consideration—can't be retracted. A default in N corresponds to a *prima facie* belief, one that carries some conviction, but can be suspended in certain cases. A conclusion in E (at least, a conclusion that is not logically implied by M) corresponds to an *all-things-considered belief*, the outcome of a process of contradiction resolution and selection of competing defaults.

The distinction between *prima facie* and all-things-considered beliefs makes good intuitive sense. Take the following example, for instance.

Example 2.1. *Beliefs about the porch light.*

(i) I have a reason to believe the porch light is off, because I asked my daughter to turn it off.

(ii) I have a reason to believe the porch light is on, because the last time I saw it, it was on.

(iii) All things considered, I believe the porch light is off, because my daughter is pretty reliable.

In this example, *Prima facie* beliefs (i) and (ii) conflict with each other. The conflict is resolved by discarding (ii) and retaining (i) in reaching the all-things-considered belief (iii). Note that, like an intention, the all-things-considered belief that the light is off acts as a constraint on future deliberation; I will not form a plan to turn it off if I believe, all things considered, that it is already off.

2.2. Wishes and wants

There are systematic similarities between beliefs and desires, which are reflected in the language used to describe them and in commonsense reasoning. Of special importance for

[1] We will not repeat Reiter's definition of an extension here; a generalization of the definition, for systems with defaults for both beliefs and desires, is given in Section 2, below.

motivating the formalism that I am aiming for here is an analogy between *prima facie* and all-things-considered beliefs on the one hand, and *wishes* and *wants* on the other.

Commonsense practical reasoning is concerned with the practicalization of desires. Immediate desires needn't be feasible, and typically will conflict with other immediate desires. We do not expect all of these wishes to survive as practical goals. The ones that do survive I will call *wants*.

This distinction seems to correspond to one important difference between the way 'wish' and 'would like' on the one hand and 'want' on the other are typically used. The following example shows that wishes can conflict with beliefs.

Example 2.2. *An infeasible wish.*

I'd like to take a long vacation.
I'd need to get time off from work to take a long vacation.
But: I can't get time off from work.

Also, wishes can conflict with each other, in light of background beliefs.

Example 2.3. *Conflicting wishes.*

I'd like to take a long vacation.
But: I'd like to save more money this year.
And: I can't save more money this year and take a long vacation.

Finally, wishes can conflict with intentions, or more generally with adopted plans. This point is made by Michael Bratman, David Israel, and Martha Pollack. See [4, 3].

Example 2.4. *Conflicting wishes.*

I'd like to take a nap.
But: I intend to catch a plane.
So: I can't take a nap.

2.3. Modeling belief and desires with defaults[2]

Exploiting the analogy developed in Section 2.2, I propose to use default rules to formalize both beliefs and desires. However, as we will see, it will be important to mark the difference between belief-based and desire-based defaults. I will use

$$A \overset{\mathrm{B}}{\hookrightarrow} C$$

for the first sort of default rule, and

$$A \overset{\mathrm{D}}{\hookrightarrow} C$$

for the second sort of default rule.

An axiomatization in the logic BD, or a *BD-basis,* will now consist of (1) a set M of formulas of first order logic (the nonmonotonic axioms), (2) a set NB of belief defaults, and

[2]These ideas have been presented in several conference papers. See [17].

(3) a set ND of desire defaults. A theory *induced by* a BD-basis $\langle M, NB, ND\rangle$ is a triple $\langle M, N, E\rangle$, where E is an extension of $\langle M, N\rangle$. Before defining the crucial notion of an extension, we will consider some motivating examples.

Example 2.5. *The Reasoning in Natural Language.*

1. Coffee is available.
2. I'd like to have decaf coffee if decaf coffee is available.
3. Decaf coffee must be available if coffee is available.
4. To have decaf coffee, I'll need to order decaf coffee.
5. *So:* I'll order decaf coffee.

Example 2.5, continued. *Formalizing the Premises*

1. $\top \overset{\mathrm{B}}{\hookrightarrow} \texttt{Coffee-Available}$
2. $\texttt{Coffee-Available} \overset{\mathrm{D}}{\hookrightarrow} \texttt{Have-Decaf}$
3. $\texttt{Coffee-Available} \overset{\mathrm{B}}{\hookrightarrow} \texttt{Decaf-Available}$
4. $\texttt{Have-Decaf} \overset{\mathrm{B}}{\hookrightarrow} \texttt{Order-Decaf}$

Example 2.5, continued. *Notes on the Formalization of Example 2.5*

1. A premise of the form $\top \overset{\mathrm{B}}{\hookrightarrow} C$ is a default with a vacuously true antecedent. It represents a reason to believe C, which may or may not result in an all-things-considered belief in C.
2. C, on the other hand, is a monotonic belief, which has to be part of any extension.
3. Thus, Premise 1, above, is a default, representing a guess. If Premise 1 had read `Coffee-Available`, it would have represented an immediate, unretractable belief.
4. Intuitively, the premisses recorded in Example 2.5 should have only one reasonable conclusion. (Recall that conclusions are generated by a combination of beliefs and desires.) I should assume that coffee is available and that decaf coffee is available. I should want to have decaf, and I should order decaf. So the formal theory should deliver only one extension in this case, the one which is generated by the following choices:
   $\{\texttt{Coffee-Available}, \texttt{Decaf-Available}, \texttt{Have-Decaf}, \texttt{Order-Decaf}\}$
5. The above extension is the only one that is generated by Reiter's definition. (I am assuming here that the definition is applied in the simplest way, treating both sorts of defaults similarly.)
6. Temporal reasoning is suppressed in the formalization. Explicit temporal information will be represented in the extended formalism of Section 3.

7. Premise 4, above, is a fairly crude way of compiling information that would be more appropriately be inferred as part of a planning process. The formalism of Section 3 will incorporate explicit end-means reasoning.

In the next example, imagine a hiking scenario.

Example 2.6. *The Reasoning in Natural Language.*

1. I think it's going to rain.
2. If it rains, I'll get wet.
3. (Even) if it rains, I wouldn't like getting wet.
4. *So:* I'll get wet.

Example 2.6, continued. *Formalizing the Premises*

1. $\top \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Rain}$
2. $\mathtt{Rain} \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Wet}$
3. $\mathtt{Rain} \overset{\mathrm{B}}{\hookrightarrow} \neg\mathtt{Wet}$

In the presence of Premise 1, there is a direct conflict between the defaults in Premises 2 and 3. The former premise represents a belief, the latter a desire.

In this case, there should be only one extension, which is generated by the following choices:

$$\{\mathtt{Rain}, \mathtt{Wet}.\}$$

If I genuinely believe that it will rain, and that I will get wet if it rains, I should believe that I will get wet, regardless of my preferences or likings. To do otherwise would be to indulge in *wishful thinking.*

Note that wishful thinking is incorrect, regardless of whether the beliefs are defeasible. To take another example, suppose that I am not at home now, that I believe my umbrella is home now, and that I would like to have my umbrella in an hour. If wishful thinking were not prohibited, and waiting is available as an action, nothing would prevent a plan in which at I simply wait, achieving my goal by wish fulfillment.

To put it another way, if practical reasoning is to be practical, wishes can't be fulfilled simply because they are wishes, but have to be achieved by feasible actions. It is belief that determines feasibility, not desire. So in a formalism that allows desire-based defaults, belief-based defaults have to take precedence over desire-based defaults in cases when there is any conflict between the two.

In the context of the formalism I am proposing, prioritization of belief over desire can be achieved by using a more or less standard account of prioritized defaults. See, for instance, [5]. Technical details are provided in Section 2.4, below.

With belief defaults prioritized over desire defaults, we obtain a single extension in Example 2.6, the one generated by the following choices.

$$\{\mathtt{Rain}, \mathtt{Wet}\}$$

The next example elaborates the hiking scenario of Example 2.6.

Example 2.7. *The Reasoning in Natural Language.*

1. I think it will rain.
2. If it rains, I'll get wet.
3. (Even) if it rains, I wouldn't like to get wet.
4. If I get wet, I'd like to change into dry clothes.
5. If I change into dry clothes, I'll have to walk home.
6. If I walk home, I'll have to walk an extra two hours.
7. I wouldn't like to walk an extra two hours.

Example 2.7, continued. *Formalizing the Premises*

1. $\top \overset{\mathrm{D}}{\hookrightarrow} \mathtt{Rain}$
2. $\mathtt{Rain} \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Wet}$
3. $\mathtt{Rain} \overset{\mathrm{D}}{\hookrightarrow} \neg\mathtt{Wet}$
4. $\mathtt{Wet} \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Change\text{-}Clothes}$
5. $\mathtt{Change\text{-}Clothes} \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Home}$
6. $\mathtt{Home} \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Walk\text{-}Two\text{-}Hours}$
7. $\top \overset{\mathrm{B}}{\hookrightarrow} \neg\mathtt{Walk\text{-}Two\text{-}Hours}$

In this case, there should be two extensions. The first is generated by the following choices:

$$\{\mathtt{Rain}, \mathtt{Wet}, \mathtt{Change\text{-}Clothes}, \mathtt{Home}, \mathtt{Walk\text{-}Two\text{-}Hours}\}$$

The second is generated by the following choices:

$$\{\mathtt{Rain}, \mathtt{Wet}, \mathtt{Change\text{-}Clothes}, \neg\mathtt{Walk\text{-}Two\text{-}Hours}\}$$

These two extensions represent the two decision alternatives that the scenario of Example 2.7 makes available; on the one hand getting dry, but walking two hours, and on the other staying wet, but avoiding the extra walk.

In a purely epistemic version of default logic, multiple extensions represent equally reasonable alternatives, and the logic itself provides no way to choose between them. (This remains true, even if the number of extensions is reduced by prioritizing defaults; whatever multiple extensions remain will be equally reasonable as far as the logic is concerned.) With desire-based defaults added to the mix, multiple extensions are still equally reasonable as far as the logic is concerned, but in many cases an agent will see some of these extensions as obviously preferable, and will have no difficulty in choosing among them. An agent who strongly dislikes being wet, but who enjoys walking, for instance, will prefer the second extension in Example 2.6. These are the sorts of choices that numerical utilities are designed to resolve. I do not think that the logic should be expected to do more than to make the choices apparent. See Section 3.7 for further discussion of this issue.

2.4. Characterizing the extensions of a BD-basis

Here is the idea behind the formal definition to be given below. A BD-extension E of a BD-basis $\langle M, NB, ND\rangle$ is a mininal first order theory that is closed under all the defaults that are applicable to it. Prioritization of B-defaults to D-defaults is ensured by allowing a D-default to be applicable to E only if there is no set of conflicting B-defaults.

In the following definitions, $\mathcal{S} = \langle M, NB, ND\rangle$, and $\vdash$ is the consequence relation of first order logic; $\mathrm{Th}_{\mathrm{FOL}}(T) = \{A : T \vdash A\}$. Following Reiter, we define applicability relative to a set T of premises, and a "conjectured extension" T^* that is used to test consistency in applying rules.

Definition 2.1. *Applicability for B-defaults.*

A default rule $A \overset{\mathrm{B}}{\hookrightarrow} C$ is applicable to T relative to T^*, where T and T^* are sets of formulas, iff (1) $T \vdash A$ and $T^* \nvdash \neg C$. $A \overset{\mathrm{B}}{\hookrightarrow} C$ is *vacuously* applicable to T relative to T^* if it is applicable to T relative to T^* and $C \in T$.

Definition 2.2. *B-conflictedness for D-defaults.*

$A \overset{\mathrm{D}}{\hookrightarrow} C$ is B-conflicted for T with respect to T^*, $\mathcal{S}$ iff for some $A_1 \overset{\mathrm{B}}{\hookrightarrow} C_1, \ldots, A_n \overset{\mathrm{B}}{\hookrightarrow} C_n \in NB$, $T \vdash A_i$ for all i, $1 \leq i \leq n$ and $T \cup \{C_1, \ldots, C_n\} \vdash \neg C$.

Definition 2.3. *Applicability for D-defaults.*

A default rule $A \overset{\mathrm{D}}{\hookrightarrow} C$ of $\langle M, NB, ND\rangle$ is applicable to T, relative to T^* and $\mathcal{S}$, if (1) $T \vdash A$ and $T^* \nvdash \neg C$, and (2) $A \overset{\mathrm{D}}{\hookrightarrow} C$ is not B-conflicted for T with respect to T^*. $A \overset{\mathrm{D}}{\hookrightarrow} C$ is *vacuously applicable* to T relative to T^* if it is applicable to T relative to T^* and $C \in T$.

Definition 2.4. *BD-closure.*

T is BD-closed, relative to $\mathcal{S}$, T^*, iff (1) $T = \mathrm{Th}_{\mathrm{FOL}}(T)$, (2) $M \subseteq T$, (3) for all $A \overset{\mathrm{B}}{\hookrightarrow} C \in NB$, $C \in T$ if $A \overset{\mathrm{B}}{\hookrightarrow} C$ is applicable to T relative to T^*, and (4) for all $A \overset{\mathrm{D}}{\hookrightarrow} C \in ND$, $C \in T$ if $A \overset{\mathrm{D}}{\hookrightarrow} C$ is applicable to T relative to T^*.

Definition 2.5. *BD-extension.*

E is a BD-extension of a BD-basis $\mathcal{S}$ iff (1) E is BD-closed, relative to $\mathcal{S}$, E, and (2) for all E' such that E' is BD-closed, relative to $\mathcal{S}$, E', we have $E' = E$ if $E' \subseteq E$.

BD-extensions can also be characterized in a more constructive way, by conjecturing an extension (a set T^*) and using this set for consistency checks in a proof-like process that applies defaults $\mathcal{S}$ successively to stages that begin with M; such a process yields a BD-extension if it produces T^* as its limit. The following definition invokes an "alphabetical" ordering of $NB \cup ND$. Any function from ω onto $NB \cup ND$ will serve the purpose.

Definition 2.6. *BD-proof process.*

$\mathcal{P}(\mathcal{S}, T^*)$ is the sequence $\{T_i^{\mathcal{P}(\mathcal{S},T^*)} : i \in \omega\}$ defined as follows:

1) $T_0^{\mathcal{P}(\mathcal{S},T^*)} = M$.
2) $T_{i+1}^{\mathcal{P}(\mathcal{S},T^*)} = \mathrm{Th}_{\mathrm{FOL}}(T_i^{\mathcal{P}(\mathcal{S},T^*)} \cup \{C\})$ if there is a default in $NB \cup ND$ that is nonvacuously applicable to $T_i^{\mathcal{P}(\mathcal{S},T^*)}$ relative to $\mathcal{S}$, T^*, where the alphabetically first such default has the form $A \stackrel{\mathrm{B}}{\hookrightarrow} C$ or $A \stackrel{\mathrm{D}}{\hookrightarrow} C$.
3) $T_{i+1}^{\mathcal{P}(\mathcal{S},T^*)} = T_i^{\mathcal{P}(\mathcal{S},T^*)}$ if no default in NB or ND is nonvacuously applicable to $T_i^{\mathcal{P}(\mathcal{S},T^*)}$ relative to $\mathcal{S}$, T^*.

Definition 2.7. *Lim*($\mathcal{P}(\mathcal{S}, T^*)$)
$\mathrm{Lim}(\mathcal{P}(\mathcal{S}, T^*)) = \bigcup\{T_i^{\mathcal{P}(\mathcal{S},T^*)} : i \in \omega\}$.

In this version, I will simply state the following theorems without proof.

Theorem 2.1. Let E be a BD-extension of $\mathcal{S} = \langle M, NB, ND\rangle$. Then $E = \mathrm{Lim}(\mathcal{P}(\mathcal{S}, E))$.

Theorem 2.2. Let $E = \mathrm{Lim}(\mathcal{P}(\mathcal{S}, E))$, where $\mathcal{S} = \langle M, NB, ND\rangle$. Then E is a BD-extension of $\mathcal{S}$.

In view of Theorem 2.2, we can show that T is a BD-extension by (1) using T for consistency checks in a default reasoning process from $\langle M, NB, ND\rangle$, (2) taking the limit T' of this process, and (3) verifying that in fact $T' = T$.

In Examples 2.5–2.6, some instances of informal reasoning involving beliefs and desires were formalized, along with remarks about the extensions that the examples seemed to require. The characterization of BD-extension provided by Definition 2.5 matches the requirements of these examples.

In particular, consider Example 2.6. This corresponds to the BD-basis $\langle M, NB, ND\rangle$, where:

$$M = \emptyset;$$
$$NB = \{\top \stackrel{\mathrm{B}}{\hookrightarrow} \mathtt{Rain}, \mathtt{Rain} \stackrel{\mathrm{B}}{\hookrightarrow} \mathtt{Wet}\};$$
$$ND = \{\mathtt{Rain} \stackrel{\mathrm{B}}{\hookrightarrow} \neg\mathtt{Wet}\}.$$

It is straightforward to use Theorem 2.2 to show that $\{\mathtt{Rain}, \mathtt{Wet}\}$ is a BD-extension. The proof process generated by using $\mathrm{Th}_{\mathrm{FOL}}(\{\mathtt{Rain}, \mathtt{Wet}\})$ as a conjectured extension produces $\mathrm{Th}_{\mathrm{FOL}}(\{\mathtt{Rain}\})$ at the first step, $\mathrm{Th}_{\mathrm{FOL}}(\{\mathtt{Rain}, \mathtt{Wet}\})$ at the second step, and remains constant thereafter. Theorem 2.2 can also be used to show that the unwanted "wishful thinking" conclusion set, $\mathrm{Th}_{\mathrm{FOL}}(\neg\mathtt{Wet})$, is *not* a BD-extension. The proof process generated by using $\mathrm{Th}_{\mathrm{FOL}}(\neg\mathtt{Wet})$ as a conjectured extension produces $\mathrm{Th}_{\mathrm{FOL}}(\{\mathtt{Rain}\})$ at the first step, and remains constant thereafter. The B-default $\mathtt{Rain} \stackrel{\mathrm{B}}{\hookrightarrow} \mathtt{Wet}$ cannot be applied, because the conclusion conflicts with the conjectured extension. The D-default $\mathtt{Rain} \stackrel{\mathrm{D}}{\hookrightarrow} \neg\mathtt{Wet}$ cannot be applied, because it is B-conflicted.

In Section 3, we will need the following definition.

Definition 2.8. *B-closure.*

T is B-closed, relative to $\mathcal{S} = \langle M, NB \rangle$, T^*, iff (1) $T = \mathrm{Th}_{\mathrm{FOL}}(T)$, (2) $M \subseteq T$, and (3) for all $A \overset{\mathrm{B}}{\hookrightarrow} C \in NB$, $C \in T$ if $A \overset{\mathrm{B}}{\hookrightarrow} C$ is applicable to T relative to T^*.

Definition 2.9. *B-extension.*

E is a B-extension of $\mathcal{S} = \langle M, NB \rangle$ iff (1) E is B-closed, relative to $\mathcal{S}$, E, and (2) for all E' such that E' is B-closed, relative to $\mathcal{S}$, E', we have $E' = E$ if $E' \subseteq E$.

A B-extension makes use only of D-defaults. The definitions, then, are equivalent to those of [14].

2.5. A problem

Of course, the fact that the logic BD satisfies the intuitive requirements of a number of examples is no guarantee that it is complete or even sound. In cases like this, it would be desirable to have a formal criterion of soundness and completeness that is intuitively satisfactory, and substantially different from the rather proof-theoretic formulations of Section 2. I do not think it will be easy to devise a semantics of this kind for BD, and believe that this is a side-effect of trying to formalize practical reasoning. I do not propose to throw semantics to the wind; providing a useful model theoretic semantics for logics like BD is certainly an appropriate long-range goal. But for the time being, I will continue to rely on specific example-driven intuitions, and on the general logical intuitions that derive from the close relationship of the logic BD to a familiar formalism for nonmonotonic reasoning.

Pursuing this method reveals a residual problem with BD. Depending on how you look at it, it is either inadequate, or highly incomplete as a system of practical reasoning.

Recall the train of thought that disclosed the need to prioritize defaults in the logic BD. (1) I decided at the outset to model practical reasoning by allowing both beliefs and desires to act as defaults. (2) Then I noticed that this model induced intuitively invalid cases of invalid "wishful thinking" in which desires were not properly constrained by beliefs. (3) To solve this problem, I prioritized B-defaults over D-defaults.

There is a good reason to allow desires to license default conclusions—this provides a natural way of modeling how goals are introduced into practical arguments. However, we learned that the way in which desires can enter into practical arguments has to be limited.

The following example shows that we may need further limitations of this kind.

Example 2.8. *The Reasoning in Natural Language.*

1. I'd like to have decaf coffee.
2. I can only have decaf coffee if decaf coffee is available.
3. *So:* Decaf coffee must be available.

Example 2.8, continued. *Formalizing the Premises.*

1. $\top \overset{\mathrm{D}}{\hookrightarrow}$ `Have-Decaf`
2. `Have-Decaf` $\overset{\mathrm{B}}{\hookrightarrow}$ `Decaf-Available`

This example yields just one BD-extension, the one that is generated by the following choices.

$\{\texttt{Have-Decaf}, \texttt{Decaf-Available}\}$

There are no prior beliefs in this example concerning the availability of decaf. The default desire $\top \overset{D}{\hookrightarrow} \texttt{Have-Decaf}$ is not conflicted, and so produces the conclusion `Have-Decaf`. Since I believe that `Decaf-Available` is a necessary condition for `Decaf-Available`, the additional conclusion is produced.

This reasoning is clearly a case of unsound, wishful thinking, but given what has been said so far, there is no evident criterion for separating the apparently sound reasoning of Examples 2.5–2.7 from the fallacious reasoning of Example 2.8.

You might conclude from this exercise that the system BD is unsound. I think it is more accurate to say that it is sound, as far as the examples that have been discussed here are concerned, and also sound as far as I know. *But*, without a mechanism for formalizing action and change, the system can only deliver an account of what extensions maximize desires, without contradicting any beliefs. There is nothing wrong with the conclusion reached in Example 2.8, as long as we think of it as representing an outcome that is maximally desirable, within the limits of what is believed. The fact that this is not very useful information—it does little good to know what outcomes are preferred, if we have no information about how to achieve these outcomes—simply shows that BD is expressively incomplete as a system of practical reasoning.

Planning formalisms of the sort that have been developed by AI-minded logicians—systems for reasoning about action and change—provide exactly what is needed to remedy this defect. In the next section, I will show how natural constraints that can be formulated in the extended logic BDP for belief, desire, and planning can eliminate extensions of the sort that BD produces in Example 2.8.

3. A formalism for belief, desire, and planning

In this section, I correct the deficiencies of BD by adding mechanisms for temporal reasoning and plan formation.

The general motives for this project were presented in Section 1; two of these are worth repeating at this point. (1) The primary goal is to produce a formalism for practical reasoning that can deal with inferred desires as well as beliefs. (2) A secondary goal is to keep the system as simple and generic as possible, so that there will be a starting point for work in this area that many people can accept and work with.

Although the system BD introduced in Section BD provides for inferred desires, it doesn't really represent a fully developed system for *practical* reasoning, because it makes no provision for a conclusion that tells the reasoner how to do something. To do that, we need to add to BD the capability for reasoning about feasible action; i.e., we need to extend BD to include a planning formalism of the sort discussed in [15, 16]. Because of the secondary goal, the simpler and more familiar the planning formalism, the better. For that reason, I will choose a simplified version of the *Situation Calculus*, [11, 9, 10].

3.1. Representing change and plans in BDP

BDP is an extension of BD incorporating a specialized first-order language containing an apparatus for reasoning about actions and change; it will also use a refined extension definition. The Situation Calculus uses a predicate `Holds` to formalize this sort of reasoning: `Holds` denotes a relation between fluents (i.e., dynamic properties) and situations; a functional constant `result` denotes a function from actions and situations to situations. Suppose that s_0 denotes an initial situation $\mathbf{s}_0$, that a_1 and a_2 denote actions $\mathbf{a}_1$ and $\mathbf{a}_2$, and that f denotes the fluent $\mathbf{f}$. Then

(3.1) $\mathtt{Holds}(f, \mathtt{result}(a_2, \mathtt{result}(a_1, s_0)))$

is the standard way of expressing in the formalism that performing $\mathbf{a}_1$ and $\mathbf{a}_2$ in $\mathbf{s}_0$ will yield a situation in which $\mathbf{f}$ holds. There is no reasoning about goal selection in the Situation Calculus, and no way of distinguishing the fluents that are goals from the others. And there is no way of distinguishing adopted plans from the others; one can only say what goals a plan will achieve, in a given initial situation.

BDP is intended to formalize reasoning that results in the formation of plans. Since extension-construction is the sole reasoning mechanism of BDP, we will allow extensions to determine not only all-things-considered beliefs and desires, but sequences of actions, or plans. In choosing an extension, an agent commits not only to beliefs and desires, but to actions. Therefore, I will have to modify the Situation Calculus to provide for explicit commitment to plans. There are three ingredients to the modification. (1) BDP has a family $\mathtt{step}_1, \mathtt{step}_2, \ldots$ of individual constants; $\mathtt{step}_i$ denotes ith step of the selected plan. (2) There is a sequence $\mathtt{Planlength}_i$, $i \in \omega$, of designated propositional constants; $\mathtt{Planlength}_i$ means that the designated plan has i steps. There is a special constant `null`; $\mathtt{step}_i = \mathtt{null}$ means that the ith step of the selected plan is undefined. (3) BDP has a family $\mathtt{Holds}_0$, $\mathtt{Holds}_1 \ldots$ of 1-place predicates, rather than a single `Holds` relation.[3] With this apparatus, we can make a further simplification; there is no need to represent situations explicitly. The initial situation is represented implicitly, by $\mathtt{Holds}_0$; the situation that results after the performance of i steps of the designated plan is represented implicitly by $\mathtt{Holds}_i$.

We will need to formulate desires about the future. Realistic desires are seldom associated with a specific time; for purposes of this paper I will introduce a predicate `Eventually`. For practical purposes, this predicate can be defined in terms `Planlength` and `Holds`.

The resulting formalism is expressive enough to allow the formalism of many examples. But it is easy to think of natural examples that could not be formalized without enriching the language. For instance, a "standing desire" will involve a universal quantifier over future times; there is no way to express such desires in BDP. Other desires may involve temporal contraints on events; for instance I may want to be home before my guests arrive. This can't be expressed in BDP. Many other natural, BDP-inexpressible desires can be imagined. Enriching the temporal expressivity is one very natural way to extend the initial version of BDP that I present here.

Actions are treated as individuals in BDP. There is a set of designated action constants and action functions. A *closed action term* is a term having the form t, where t either is an action constant, or is $f(t_1, \ldots, t_n)$, where $t_1, \ldots, t_n$ are terms containing no individual variables and f is an action function letter. AT is the set of closed action terms.[4]

[3] There are, of course, more direct ways to represent plans; we could add a type of sequences, or add a theory of arithmetic or a theory of actions and events. I chose this method because it seemed to me to be about the simplest way to explicitly represent commitment to plans. Remember, I am trying to construct a generic theory, that can be refined in a number if ways.

[4] It may be useful to impose type restrictions on the terms $t_1, \ldots, t_n$.

I will illustrate how BDP works with two sorts of examples: (1) blocks-world examples, and (2) reformalizations of the reasoning examples from Section 2. The former examples are unrealistic, but their simplicity and familiarity makes it relatively easy for me to present detailed formalizations. When I come to the latter examples, I will not try to formalize the domains entirely, but will rely on common intuitions.

To illustrate how plans are formed in BDP, take a blocks world with just two blocks, **a** and **b**, denoted respectively by `a` and `b`; there is a table, denoted by a constant `table`. Actions are denoted by terms of the form $\mathtt{move}(t_1, t_2)$. Suppose that in the initial situation **b** is on **a**. Then a theory selecting the simplest plan to get **a** on **b** will contain (among other formulas):

Example 3.1. *A blocks-world plan extension.*

$\mathtt{step}_1 = \mathtt{move}(\mathtt{b}, \mathtt{table})$
$\mathtt{step}_2 = \mathtt{move}(\mathtt{a}, \mathtt{b})$
$\mathtt{Planlength}_2$
$\mathtt{Holds}_0(\mathtt{on}(\mathtt{b}, \mathtt{a}))$, $\mathtt{Holds}_0(\mathtt{on}(\mathtt{a}, \mathtt{table}))$
$\mathtt{Holds}_1(\mathtt{on}(\mathtt{b}, \mathtt{table}))$, $\mathtt{Holds}_0(\mathtt{on}(\mathtt{a}, \mathtt{table}))$
$\mathtt{Holds}_2(\mathtt{on}(\mathtt{a}, \mathtt{b}))$, $\mathtt{Holds}_2(\mathtt{on}(\mathtt{b}, \mathtt{table}))$
$\neg\mathtt{Planlength}_3$
$\mathtt{step}_3 = \mathtt{null}$

3.2. Formalizing action and change in BDP

It is straightforward to translate the usual policies for formalizing action and change in the Situation Calculus to BDP. This can be illustrated by elaborating Example 3.1, making the axiomatization of the fluent dynamics explicit.

The axioms for a BDP domain are divided into five groups: (1) plan axioms, (2) general domain axioms, (3) causal axioms for actions, and (4) frame axioms. Information about initial conditions is also needed, and in familiar planning formalisms it is provided by axioms concerning the initial situation. In BDP, this information could be given by monotonic axioms, or by belief defaults, or by a combination of the two.

The plan axioms are common to all BDP-bases. They apply, for instance, to Example 3.1, so that this extension contains, for instance, $\mathtt{Eventually}(\mathtt{on}(\mathtt{a}, \mathtt{b}))$ and $\mathtt{step}_3 = \mathtt{null}$.

The remaining axioms given below are for a blocks world with three blocks. I believe they are adequate for generating a collection of plans that is neither too large nor too small; I have not tried to produce a complete axiomatization, one that would enable all formulas that intuitively hold in the domain to be proved.

Some axioms are meant to hold at every step of any plan. Since I am using a stripped-down formalism that does not allow quantification over situations, such axioms will have to be presented as schemes. $\mathbf{Ax3}_i$ is a scheme in which a $\mathtt{Holds}_i$ predicate figures; there is an instance of the scheme for each $i \geq 0$. Similarly, there is an instance of $\mathbf{Ax1}_{i,j}$ for each $i, j \geq 0$. Properties (like `Block`) that are change-invariant are represented using first-order predicates rather than fluents.

Useful definitions

$$
\begin{array}{lll}
\text{‘}\mathtt{Object}(t)\text{’} & =_{\mathrm{df}} & \text{‘}\mathtt{Block}(t) \vee t = \mathtt{table}\text{’} \\
\text{‘}\mathtt{Clear}_i(t)\text{’} & =_{\mathrm{df}} & \text{‘}\mathtt{Block}(t) \wedge \forall x \neg Holds_i(\mathtt{on}(x,t))\text{’}
\end{array}
$$

Plan axioms

$$
\begin{array}{ll}
\mathbf{Ax1}_{i,j}\text{:} & \mathtt{Planlength}_i \rightarrow \neg\mathtt{Planlength}_j, \text{ for } i \neq j \\
\mathbf{Ax2}_{i,j}\text{:} & \mathtt{Planlength}_i \rightarrow \mathtt{step}_j = \mathtt{null}, \text{ for } i < j \\
\mathbf{Ax3}_{i}\text{:} & \mathtt{Planlength}_i \rightarrow \forall x[\mathtt{Eventually}(x) \leftrightarrow \mathtt{Holds}_i(x)]
\end{array}
$$

General domain axioms

$$
\begin{array}{ll}
\mathbf{Ax4}\text{:} & \mathtt{Block}(\mathtt{table}) \\
\mathbf{Ax5}\text{:} & \forall x[\mathtt{Block}(x) \leftrightarrow [x = \mathtt{a} \vee x = \mathtt{b} \vee x = \mathtt{c}]] \\
\mathbf{Ax6}\text{:} & \mathtt{a} \neq \mathtt{b} \wedge \mathtt{b} \neq \mathtt{c} \wedge \mathtt{c} \neq \mathtt{a} \\
\mathbf{Ax7}_i\text{:} & \forall x \forall y[\mathtt{Holds}_i(\mathtt{on}(x,y)) \rightarrow [\mathtt{Block}(x) \wedge \mathtt{Object}(y)]] \\
\mathbf{Ax8}_i\text{:} & \forall x \forall y \forall z[[\mathtt{Holds}_i(\mathtt{on}(x,y)) \wedge \mathtt{Holds}_i(\mathtt{on}(x,z))] \rightarrow z = y] \\
\mathbf{Ax9}_i\text{:} & \forall x \forall y \forall z[[\mathtt{Holds}_i(\mathtt{on}(x,y)) \wedge \mathtt{Holds}_i(\mathtt{on}(z,y))] \rightarrow z = x] \\
\mathbf{Ax10}_i\text{:} & \forall x \forall y[\mathtt{Holds}_i(\mathtt{on}(x,y)) \rightarrow \mathtt{Holds}_i(\mathtt{on}(y,x))] \\
\mathbf{Ax11}_i\text{:} & \forall x \exists y[\mathtt{Holds}_i(\mathtt{on}(x,y))
\end{array}
$$

Causal axioms

$$
\begin{array}{ll}
\mathbf{Ax12}_i\text{:} & \forall x \forall y \forall z[[\mathtt{step}_{i+1} = \mathtt{move}(x,y) \wedge \mathtt{Clear}_i(x) \wedge \mathtt{Holds}_i(\mathtt{on}(x,z)) \\
& \quad \wedge\, y \neq z \wedge [y = \mathtt{table} \vee \mathtt{Clear}_i(y)]] \\
& \quad \rightarrow [\mathtt{Holds}_{i+1}(\mathtt{on}(x,y))) \wedge \neg\mathtt{Holds}_{i+1}(\mathtt{on}(x,z)))]
\end{array}
$$

Frame axioms

To keep things simple, I will use monotonic frame axioms. Of course, familiar formalisms for reasoning about action and change will typically incorporate a nonmonotonic solution to the frame problem; but in these formalisms the nonmonotonic apparatus is usually confined to this one application. In typical applications of BDP, several components will be nonmonotonic, and it will be necessary to think carefully about interactions between these components. I will discuss these matters briefly in Section 3.6, below.

The frame axioms are as follows.

$$
\begin{array}{ll}
\mathbf{Ax13}_i\text{:} & \forall x \forall y \forall z[[\mathtt{step}_{i+1} = \mathtt{move}(x,y) \wedge \mathtt{Holds}_i(\mathtt{on}(u,v) \wedge u \neq x] \\
& \quad \rightarrow \mathtt{Holds}_{i+1}(\mathtt{on}(u,v))]
\end{array}
$$

$$
\begin{array}{ll}
\mathbf{Ax14}_i\text{:} & \forall x \forall y \forall z[[\mathtt{step}_{i+1} = \mathtt{move}(x,y) \wedge \neg\mathtt{Holds}_i(\mathtt{on}(u,v) \wedge u \neq x] \\
& \quad \rightarrow \neg\mathtt{Holds}_{i+1}(\mathtt{on}(u,v))]
\end{array}
$$

3.3. Informal description of BDP extensions

As in BD, a basis will consist not only of monotonic axioms, but of belief and desire defaults. As before, belief defaults have the form $A \overset{\mathrm{B}}{\hookrightarrow} B$. Any desire default of the form $A \overset{\mathrm{D}}{\hookrightarrow} B$ is allowed, but in realistic cases we are interested in desires that are future-directed. As

I explained above, in the context of the present paper, that means desires of the form $A \overset{\mathrm{D}}{\hookrightarrow} \texttt{Eventually}(f)$.

We wish extensions to choose *feasible* options—ones that are not only desirable and compatible with beliefs, but that can be secured by acting on a plan. We ensure this by first constructing BD extensions. Assuming that the beliefs and desires have to do only with world states (i.e., with whether or not fluents hold), and not with plan length or selection of actions, a BD extension will present a picture of things as the planning agent would like them to be, but without any specific information about actions that would realize it.

I do not exclude *prima facie* beliefs such as $\top \overset{\mathrm{B}}{\hookrightarrow} \texttt{Holds}_3(f)$, or *prima facie* desires such as $\top \overset{\mathrm{D}}{\hookrightarrow} \texttt{Planlength}_5$ or $\top \overset{\mathrm{D}}{\hookrightarrow} \texttt{step}_1 = \texttt{move}(\texttt{a}, \texttt{table})$. In the simplest case, though, all belief defaults have conclusions of the form $\texttt{Holds}_0(f)$, and all desire defaults have conclusions of the form $\texttt{Eventually}(f)$. In this case (assuming that the monotonic theory M provides only general domain information) a BD extension will provide initial conditions and goals—precisely the sort of input that a classical planning algorithm would need.

A BDP plan-description over a set of closed action terms AT is a set of formulas consisting of a choice of a plan length and a selection of plan steps up to the plan length. For any BDP plan-description PG, then, there will be an n and a set of terms $\{t_1, \ldots, t_n\} \subseteq AT$, such that:

$$PG = \{\texttt{Planlength}_n, \texttt{step}_1 = t_1, \ldots, \texttt{step}_n = t_n\}.$$

A BDP proto-plan over a BDP-basis $\mathcal{S} = \langle M, NB, ND, AT\rangle$ is a consistent set of the form $\mathrm{Th}_{\mathrm{FOL}}(E_1 \cup PG)$, where E_1 is a BD extension of $\mathcal{S}$ and PG is a plan-description over PG. A solved planning problem will involve a *means*, a series of actions calculated to achieve certain goals; and, of course, it will involve the goals or ends themselves. BDP proto-plans determine a means, in the form of a plan-description; they determine ends, in the form of the desires activated in a BD-extension.

But a proto-plan needn't correspond to a feasible plan; it may involve wishful thinking. There is nothing to ensure that the means of a proto-plan cause the ends; a goal may belong to a proto-plan simply because it is the conclusion of a default desire in ND.

Therefore, we filter out the proto-plans that involve wishful thinking. A BDP extension of a BD-basis $\mathcal{S} = \langle M, NB, ND\rangle$ is a BDP proto-plan E over $\mathcal{S}$ with plan-description PG such that E is a B extension of $\langle M \cup PG, NB\rangle$. Thus, every formula in E—including the formulas that represent desires or ends—follows merely from beliefs, given the steps of its plan-description.

The technical definitions are given below, in Section 3.5; before turning to these, I will illustrate the ideas with examples.

3.4. Examples of BDP reasoning and extensions

Blocks-world examples

All blocks-world examples use the blocks-world axioms given in Section 3.2. In the first example, the agent has unconflicted *prima facie* beliefs about the initial conditions, and an unconflicted *prima facie* desire to have block **a** on block **b**. In BDP, this case works like a traditional planning problem in which the default beliefs give the initial conditions, and the desire gives the goal.

Example 3.2. *Simple, unconflicted beliefs and desires.*
Defaults:

$$\begin{aligned} NB = \{ & \top \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Holds}_0(\mathtt{on}(\mathtt{a}, \mathtt{table})), \\ & \top \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Holds}_0(\mathtt{on}(\mathtt{b}, \mathtt{table})), \\ & \top \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Holds}_0(\mathtt{on}(\mathtt{c}, \mathtt{table}))\} \\ ND = \{ & \top \overset{\mathrm{D}}{\hookrightarrow} \mathtt{Eventually}(\mathtt{on}(\mathtt{a}, \mathtt{b}))\} \end{aligned}$$

Example 3.2, continued. *BDP extensions.*

There are in fact many BDP plans, corresponding to the various sequences of moves that will get **a** on **b**. The shortest such plan is generated by the following formulas:

(Pl_1): $\{\mathtt{Planlength}_1, \mathtt{step}_1 = \mathtt{move}(\mathtt{a}, \mathtt{b})\}$

In this case, default beliefs act like initial conditions, the desire acts like a goal. Extensions correspond to plans that produce the desired goal.

To go into the matter in more detail, why exactly does (Pl_1) generate a BDP plan? First, note that

$$\begin{aligned} (\mathit{BD\text{-}Ext}_1)\text{:} \quad & \{\mathtt{Holds}_0(\mathtt{on}(\mathtt{a}, \mathtt{table})), 4 \\ & \mathtt{Holds}_0(\mathtt{on}(\mathtt{b}, \mathtt{table})), \\ & \mathtt{Holds}_0(\mathtt{on}(\mathtt{c}, \mathtt{table})), \\ & \mathtt{Eventually}(\mathtt{on}(\mathtt{a}, \mathtt{b}))\} \end{aligned}$$

generates a BD-extension in this example. (In fact, this is the only BD-extension.)

Second, note that Pl_1 is a BDP plan-description. Since $\mathit{BD\text{-}Ext}_1 \cup Pl_1$ is consistent, its logical closure, PG_1, is a BDP proto-plan. It is easy to check that PG_1 is a B-extension of the information given in the example; all that is involved is verifying that $\mathtt{Holds}_1(\mathtt{on}(a, b))$ follows from the initial conditions and $\mathbf{Ax12}_0$, the appropriate causal axiom for `move`. Therefore, PG_1 is a BDP extension.

In the next example, the agent has the same unconflicted *prima facie* beliefs about the initial conditions, and conflicted *prima facie* desires: a desire to have block **a** on block **c** and a desire to have block **b** on block **c**. This yields two sorts of BDP extensions: those corresponding to plans for putting **a** on **c** and those corresponding to plans for putting **b** on **c**. Since these goals are incompatible, there are no BDP extensions in which both goals are satisfied.

Example 3.3. *Conflicted desires, unconflicted beliefs.*

Defaults: like Example 3.2, except that:

$$ND = \{\top \overset{\mathrm{D}}{\hookrightarrow} \mathtt{Eventually(on(a,c))}, \top \overset{\mathrm{D}}{\hookrightarrow} \mathtt{Eventually(on(b,c))}\}$$

Example 3.3, continued. *BDP Extensions.*

The simplest BDP extensions (corresponding to the shortest plans) are generated by the following choices:

$$\begin{array}{ll} \textit{BDP-Ext}3.3_1\text{:} & \{\mathtt{Planlength}_1, \mathtt{step}_1 = \mathtt{move(a,c)}\} \\ \textit{BDP-Ext}3.3_2\text{:} & \{\mathtt{Planlength}_1, \mathtt{step}_1 = \mathtt{move(b,c)}\} \end{array}$$

In this sort of case, the best way to choose among the alternative BDP extensions would be to use plan evaluation methods that somehow combine criteria having to do with the simplicity of the plan and with the utilities of the outcomes.

In the next example, the agent has contradictory *prima facie* beliefs about the initial conditions: that block **a** is on block **c** and that block **b** is on block **c**. There is one *prima facie* desire: to have both **c** clear. Here, there are two sorts of extensions: those corresponding to plans for getting **a** on the table, assuming that **a** is on **b**, and those corresponding to plans for getting **b** on the table, assuming that **b** is on **a**.

Example 3.4. *Unconflicted desires, conflicted beliefs.*
Defaults:

$$\begin{aligned} NB = \{&\top \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Holds}_0\mathtt{(on(a,c))}, \\ &\top \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Holds}_0\mathtt{(on(b,c))}, \\ &\top \overset{\mathrm{B}}{\hookrightarrow} \neg\mathtt{Holds}_0\mathtt{(on(a,b))}, \\ &\top \overset{\mathrm{B}}{\hookrightarrow} \neg\mathtt{Holds}_0\mathtt{(on(b,a))}, \\ &\top \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Holds}_0\mathtt{(on(c,table))}\} \\ ND = \{&\top \overset{\mathrm{D}}{\hookrightarrow} \mathtt{Eventually(Clear(c))}\} \end{aligned}$$

Example 3.4, continued. *BDP extensions.*

The simplest extensions (corresponding to the shortest plans) are generated by the following choices:

$$\begin{array}{ll} \textit{BDP-Ext}3.4_1\text{:} & \{\mathtt{Planlength}_1, \mathtt{step}_1 = \mathtt{move(a,table)}\} \\ \textit{BDP-Ext}3.4_2\text{:} & \{\mathtt{Planlength}_1, \mathtt{step}_1 = \mathtt{move(b,table)}\} \end{array}$$

Remember, this is a case in which the agent has a reason to believe that **a** is on **c** and a reason to believe that **b** is on **c**. It would not be at all appropriate to choose between the two extensions according to utilities, since the plan with the higher utility could rest on a false assumption and be infeasible.

Nor is this quite the same as a case of inadequate information, as in Example 3.7. When beliefs conflict, the information is *overdetermined*, by a clash between opposed reasons for belief. In cases of inadequate information, information is *underdetermined*. The two cases are easy to confuse because in practice, we may switch between them. In cases when we lack information about a crucial alternative, we may reason by cases, considering the consequences of pretending we believe each alternative. And in cases where our beliefs conflict, we may suspend belief concerning the outcome. Nevertheless the two cases are different. One way to see the difference is to notice that the cure for inadequate information is to acquire more beliefs, whereas the cure for belief overdetermination is to *remove* beliefs.

In a model of practical reasoning according to which agents act on all-things-considered beliefs, the role of *prima facie* beliefs is to enable action by precipitating all-things-considered beliefs. Since the reasoning process is defeasible, the resulting plans may fail, so an agent has to be prepared to react appropriately when plans fail because they are based on false all-things-considered beliefs. As a corollary, an agent has to avoid acting on *prima facie* beliefs when the adverse consequences of acting on a false belief would be disastrous.

When no adverse consequences attach to trying and failing, and choosing and acting on one BDP extension does not remove the opportunity for acting on other BDP extensions, the best solution may be to choose one arbitrarily and act on it. The other choices can be remembered, and invoked as fallback procedures in case of failure.

In Example 3.4, for instance, an agent could arbitrarily choose $\mathit{BDP\text{-}Ext}3.4_1$. This would entail assuming that **a** is on **b**, and acting on a plan to move **a** to the table.

In case the assumption that **a** is on **b** is incorrect, the agent will *attempt* to move **a** to the table, and the attempt will fail. If this failed attempt leaves the situation unchanged, $\mathit{BDP\text{-}Ext}3.4_2$ can be invoked, and (assuming that at least one of the *prima facie* beliefs in the example is correct), this plan will now succeed.

In order to assess the risks of following this strategy, the agent will need to reason about the consequences of *failed attempts* to perform an action. This sort of reasoning, which seems to be critical in planning that is based on fallible beliefs, has—as far as I know—been neglected in the literature on planning formalisms. The standard formalisms concentrate on the effects of actions when the preconditions for the performance of these actions are met, but say nothing about the effects of attempted actions which the fail because their preconditions are not met. To formalize strategic planning, this deficiency will need to be addressed.

Example 3.5. *Conflicted desires, conflicted beliefs.*

Defaults: like Example 3.4, except that:

$ND = \{\top \overset{\mathrm{D}}{\hookrightarrow} \mathtt{Eventually}(\mathtt{on}(\mathtt{a}, \mathtt{b})), \top \overset{\mathrm{D}}{\hookrightarrow} \mathtt{Eventually}(\mathtt{on}(\mathtt{b}, \mathtt{a})), \}$

Example 3.5, continued. *BDP extensions.*

The simplest extensions are generated by the following choices:

BDP-Ext3.5_1: $\{\mathtt{Holds}_0(\mathtt{on}(\mathtt{a}, \mathtt{c})), \mathtt{Planlength}_1, \mathtt{step}_1 = \mathtt{move}(\mathtt{a}, \mathtt{b})\}$
BDP-Ext3.5_2: $\{\mathtt{Holds}_0(\mathtt{on}(\mathtt{a}, \mathtt{c})), \mathtt{Planlength}_2, \mathtt{step}_1 = \mathtt{move}(a, \mathtt{table}), \mathtt{step}_1 = \mathtt{move}(\mathtt{b}, \mathtt{a})\}$
BDP-Ext3.5_3: $\{\mathtt{Holds}_0(\mathtt{on}(\mathtt{b}, \mathtt{c})), \mathtt{Planlength}_2, \mathtt{step}_1 = \mathtt{move}(b, \mathtt{table}), \mathtt{step}_2 = \mathtt{move}(\mathtt{a}, \mathtt{b})\}$
BDP-Ext3.5_4: $\{\mathtt{Holds}_0(\mathtt{on}(\mathtt{b}, \mathtt{c})), \mathtt{Planlength}_1, \mathtt{step}_1 = \mathtt{move}(\mathtt{b}, \mathtt{a})\}$

In this case, the extensions are a sort of cross-product of the alternatives presented by the conflicting beliefs and those presented by the conflicting desires. In the absence of any opportunity to gather information, the best way to attack this would be to first choose the most plausible or likely beliefs, and then to choose the plans that, within these, BDP extensions, maximize utility.

Aristotle noted a constraint on practical deliberation; it only concerns what is in an agent's power.[5] It is worth asking whether the apparatus of BDP extensions enforces this constraint on practical deliberation. We can't ask this question directly, since the BDP formalism doesn't represent ability explicitly; but we can look into special cases. One of the most interesting of these cases is deliberation about the present. We would hope to see the constraint enforced as follows: any desire regarding the present (i.e., any default having the form $A \overset{\mathrm{D}}{\hookrightarrow} \mathtt{Holds}_0(f)$) will have no effect whatever on plan formation in BDP.

In cases where a BD extension E_1 provides complete information about the initial situation, $\mathtt{Holds}_0(f)$ will either be a logical consequence of E_1 or will be inconsistent with it, so the inefficacy of $A \overset{\mathrm{D}}{\hookrightarrow} \mathtt{Holds}_0(f)$ is trivial. The following example illustrates a case where there is incomplete information.

Example 3.6. *Deliberation about the present.*
Defaults:

$$NB = \{\top \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Holds}_0(\mathtt{on}(\mathtt{b}, \mathtt{table}))\},$$
$$\top \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Holds}_0(\mathtt{on}(\mathtt{c}, \mathtt{table}))\}$$
$$ND = \{\top \overset{\mathrm{D}}{\hookrightarrow} \mathtt{Holds}_0(\mathtt{on}(\mathtt{a}, \mathtt{b}))\}$$

Example 3.6, continued. *BDP extensions.*

[5] See [1], 1112^a18ff.

To understand this example, it is useful to consider what happens when an agent has no desires, i.e., when $ND = \emptyset$. In that case, any plan that is consistent with the agent's beliefs will precipitate a BDP extension. This makes good sense; desires act as constraints on plans, so the more desires, the fewer the plans. In the limiting case where there are no desires, any feasible plan is allowed.

The only desire in the example is $\top \overset{\mathrm{D}}{\hookrightarrow} \mathtt{Holds}_0(\mathtt{on}(\mathtt{a}, \mathtt{b}))$. And the only B extension is the logical closure E_1 of the axioms and

$$\{\mathtt{Holds}_0(\mathtt{on}(\mathtt{b}, \mathtt{table}))\}, \mathtt{Holds}_0(\mathtt{on}(\mathtt{c}, \mathtt{table}))\}.$$

Now, we can show by induction on n that adding a plan-description of length n to E_1 will have a model satisfying $\neg\mathtt{Holds}_0(\mathtt{on}(\mathtt{a}, \mathtt{table}))$. Therefore, $\mathtt{Holds}_0(\mathtt{on}(\mathtt{a}, \mathtt{table}))$ is in no BDP extension. So, in this case, the BDP extensions include all the possible plan-descriptions—in fact, the set of BDP extensions is exactly the same as the set that would be induced if *NB* were empty. Desires about the present have no effect on planning.

It would be desirable to generalize Example 3.6. But there is no very easy way to do that. The fact that proving in general that desires about the present are inefficacious is a challenge is, I think, one of the more interesting features of BDP. Like the Yale Shooting Problem and the suite of similar challenges that evolved from work on the frame problem, it seems to me—at a very early stage in thinking about the issues—that progress on the problem is possible, and that it could yield new insights into commonsense reasoning about causes and actions.

There are anomalous BDP-bases that will allow desires about the present to play a role in plan construction. Suppose, for instance, that we add the following axiom scheme to the ones listed in Section 3.2.

Ax99$_i$: $\mathtt{step}_{i+1} = \mathtt{move}(\mathtt{a}, \mathtt{table}) \rightarrow \neg\mathtt{Holds}_i(\mathtt{on}(\mathtt{b}, \mathtt{a}))$

Note that any instance of this axiom scheme is true in the intended domain. (I am assuming that $\mathtt{step}_i = \mathtt{move}(\mathtt{a}, \mathtt{table})$ means that the action of moving **a** to the table actually occurs, rather than an attempt to perform the action.) However, if we let $NB = \emptyset$ and $ND = \top \rightarrow \neg\mathtt{Holds}_0(\mathtt{on}(\mathtt{b}, \mathtt{a}))$, there will be a BDP plan, there will be a BDP extension in which the plan-description

$$\{\mathtt{Planlength}_1, \mathtt{step}_1 = \mathtt{move}(\mathtt{a}, \mathtt{table})\}$$

is adopted, in order to achieve the goal $\mathtt{Holds}_0(\mathtt{on}(\mathtt{b}, \mathtt{a}))$.

A crude solution to the problem would disallow axioms of the sort that create the problem. Probably, there is a way of restricting axioms about actions to ones resembling the causal axioms in Section 3.2 that would eliminate anomalies of this kind. But this approach, and any approach that disallows true axioms for purely syntactic reasons, strikes me as *ad hoc* and unenlightening.

It seems to me that the intuitive reason that Ax99$_i$ doesn't support planning is that this axiom is not causal. A solution to the problem that relies on an explicit representation of causality is more likely to shed light on practical reasoning and on the role of causality in commonsense reasoning. Independent work on the foundations of reasoning about action and change has also led at least some people to explicit representations of causality, and it seems to me that one of the major successes of this research program has been the light it has shed on commonsense causality. Perhaps BDP could shed light on these matters from another angle.

Example 3.7. *Inadequate information.*
Defaults:

$$NB = \{\top \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Holds_0(on(a,table))}\}$$
$$ND = \{\top \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Eventually(on(a,b))}\}$$

Example 3.7, continued. *BDP extensions.*

There are no BDP extensions in this case.

There is only one BD extension of $\langle M, NB, ND, AT\rangle$, where M is the set of the blocks-world axioms given in Section AC. This is the logical closure E_1 of

$$M \cup \{\top \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Holds_0(on(a,table))}, \top \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Eventually(on(a,b))}\}.$$

There is only one B extension of $\langle M \cup PG,\ \ NB\rangle$, where PG is a plan-description over AT. This is the logical closure $E_2(PG)$ of

$$M \cup PG \cup \{\top \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Holds_0(on(a,table))}\}.$$

It can be shown by induction on plan length that for every plan-description PG over AT, $E_2(PG)$ has a model that satisfies

$$\mathtt{Eventually(on(a,table))} \text{ and } \mathtt{Eventually(on(b,c))}$$

and a model that satisfies

$$\mathtt{Eventually(on(a,table))} \text{ and } \mathtt{Eventually(on(c,b))}.$$

It follows from this that for every plan-description PG over AT, $E_2(PG) \nvdash$ $\mathtt{Eventually(on(a,table))}$.

This feature depends on the fact that the axioms make the performance of actions whose preconditions fail entirely unconstrained. So the example is a bit unnatural. But it does indicate that in cases of incomplete information, we are liable to wind up with few plans, or perhaps with no plans at all. In this case, BDP works in much the same way as classical planning formalisms.

This inadequacy can be addressed in various ways. (1) The experimental approach mentioned in connection with Example 3.4, above. But experimentation is not of much use unless you can gather some information—at least, enough information to be able to tell whether the goal has been achieved. So this approach should be combined with the capability to reason about information-gathering events. (2) This leads naturally to an extension of BDP to deal with sensory actions, an idea that has been explored in connection with classical planning formalisms. (See, for instance, [7].) (3) One could also try to integrate some probabilistic form of reasoning with BDP. These are all open-ended research projects.

Reformalization of commonsense reasoning examples

We now turn to the examples of commonsense reasoning given in Section 2. I will not try to axiomatize the domains, but in each case will try to give enough of the formalization to create a sense that BDP could deliver a reasonable account of the reasoning.

Example 3.8. *Reformalizing Example 2.5.*

1. $NB = \{\top \overset{\text{B}}{\hookrightarrow} \texttt{Holds}_0(\texttt{Coffee-Available}),$
   $\texttt{Holds}_0(\texttt{coffee-available}) \overset{\text{B}}{\hookrightarrow} \texttt{Holds}_0(\texttt{decaf-available})\}$
2. $ND = \{\texttt{Holds}_0(\texttt{coffee-available}) \overset{\text{D}}{\hookrightarrow} \texttt{Eventually}(mthave - decaf)\}$
3. There is one action, `order-decaf`. Its only precondition is `decaf-available`, its only effect is `have-decaf`.

The BDP extensions correspond to plans that use the action `order-decaf` to secure the goal `have-decaf`. The simplest such extension, of course, describes the following one-step plan.

$\texttt{Planlength}_1$
$\texttt{step}_1 = \texttt{order-decaf}$

Example 3.9. *Reformalizing Example 2.7.*

1. $NB = \{\top \overset{\text{B}}{\hookrightarrow} \texttt{Holds}_1(\texttt{rain}), \texttt{Holds}_1(\texttt{rain}) \overset{\text{B}}{\hookrightarrow} \texttt{Holds}_1(\texttt{wet})\}$
2. $ND = \{\texttt{Holds}_1(\texttt{rain}) \overset{\text{D}}{\hookrightarrow} \neg\texttt{Holds}_1(\texttt{wet}),$
   $\texttt{Holds}_1(\texttt{wet}) \overset{\text{D}}{\hookrightarrow} \texttt{Eventually}(\texttt{change-clothes}),$
   $\top \overset{\text{D}}{\hookrightarrow} \neg\texttt{Eventually}(\texttt{two-hours-walking})\}$
3. There are three actions:
   (a) `walk-home` has no preconditions. It has `at-home` and `two-hours-walking` as its effects.
   (b) `change-clothes` has `at-home` as a precondition and has `wet` as a negative effect.
   (c) `hike-on` has no preconditions and no effects.

Here, there are two kinds of BDP extensions: the ones that satisfy the desire to not be wet by performing a `walk-home`, `change-clothes` sequence, but frustrate the desire to avoid two hours of walking, and the ones that perform `hike-on` and so satisfy the desire to avoid walking two hours but frustrate the desire to not be wet.

3.5. Formal description of BDP extensions

Definition 3.10. *BDP-basis*
A BDP-basis is a package $\mathcal{S} = \langle M, NB, ND, AT\rangle$, where $\langle M, NB, ND\rangle$ is a BD-basis and AT is a set of action terms.

Definition 3.11. *BDP plan-description.*
PG is a BDP plan-description of length n over AT if for some $t_1, \ldots, t_n \in AT$:

(1) $\mathtt{Planlength}_n \in PG$,

(2) $t_1, \ldots, t_n \subseteq PG$.

Definition 3.12. *BDP proto-plan.*
T is a BDP proto-plan over a BDP-basis $\langle M, NB, ND, AT\rangle$ iff for some BD extension E_1 of $\langle M, NB, ND\rangle$ and plan-description PG over AT, $T = E_1 \cup PG$

Definition 3.13. *BDP extension.*
E is a BDP extension of a BDP basis $\mathcal{S} = \langle M, NB, ND, AT\rangle$ iff (1) E is a proto-plan over $\mathcal{S}$ with plan-description PG, and (2) E is a B extension of $\langle M \cup PG, NB\rangle$.

3.6. Interacting defaults

In applications of nonmonotonic logic that are at all complex, one has to be on the lookout for interactions between defaults that may call for prioritization. In this case, I have not come up with any obvious examples of that sort, even in more complicated domains where the nonmonotonic apparatus is used to solve the frame problem.

If we were to have inertial defaults, of the form

$$\mathtt{Holds}_i(f) \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Holds}_{i+i}(f)$$

and

$$\neg\mathtt{Holds}_i(f) \overset{\mathrm{B}}{\hookrightarrow} \neg\mathtt{Holds}_{i+i}(f),$$

these defaults could certainly conflict with other *prima facie* beliefs.

> **Example 3.10.** *Conflicts between* prima facie *beliefs and inertial defaults.*
> Defaults:
>
> $$\begin{aligned} NB = &\{\mathtt{Holds}_{i,f}(f) \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Holds}_{i+i}(f) \;:\; i \in \omega\, f \in FT\} \\ &\cup \{\neg\mathtt{Holds}_{i,f}(f) \overset{\mathrm{B}}{\hookrightarrow} \neg\mathtt{Holds}_{i+i}(f) \;:\; i \in \omega\, f \in FT\} \\ &\cup \{\top \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Holds}_0(\mathtt{on}(\mathtt{a}, b)), \\ &\quad \top \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Holds}_0(\mathtt{on}(\mathtt{b}, \mathtt{table})), \\ &\quad \top \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Holds}_0(\mathtt{on}(\mathtt{c}, \mathtt{table})), \\ &\quad \mathtt{step}_1 = \mathtt{move}(\mathtt{a}, \mathtt{table}) \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Holds}_1(\mathtt{on}(\mathtt{c}, \mathtt{b}))\} \end{aligned}$$

In evaluating the B extensions of these defaults, with respect to a basis that includes $\mathtt{Planlength}_1$ and $\mathtt{step}_1 = \mathtt{move}(\mathtt{a}, \mathtt{table}) \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Holds}_1(\mathtt{on}(\mathtt{c}, \mathtt{b}))$, there is a conflict between the *prima facie* belief $\mathtt{step}_1 = \mathtt{move}(\mathtt{a}, \mathtt{table}) \overset{\mathrm{B}}{\hookrightarrow} \mathtt{Holds}_1(\mathtt{on}(\mathtt{c}, \mathtt{b}))$ and the inertial default $\neg\mathtt{Holds}_0(\mathtt{on}(\mathtt{c}, \mathtt{b}) \overset{\mathrm{B}}{\hookrightarrow} \neg\mathtt{Holds}_1(\mathtt{on}(\mathtt{c}, \mathtt{b}))$.

This is an artificial case, so maybe there are no clear intuitions. But there are natural cases in which inertial defaults conflict with *prima facie* beliefs, and there seems to be no simple way to adjudicate the conflicts. One example is Example 2.1, the case of the porch light. Another is a case in which I park my car by a fire hydrant. Inertial defaults will suggest that my car is where I left it. The rule that cars parked by fire hydrants are ticketed and towed suggests that my car is not where I left it.

Without any logical criterion for separating such conflicts, I will assume that in the general case there are no logical priorities among B defaults. Further reflection may disclose general priorities, and special priorities may arise in special domains.

3.7. Extension evaluation

BDP leaves conflicts between desires unresolved if they are not removed by feasibility considerations. This is illustrated by the two extensions generated in Example 3.9; the reasoning provided by BDP does not solve the hiker's dilemma. Although it does focus the practical problem by providing two alternatives, it provides no mechanism for choosing between (i) remaining wet and (ii) getting dry but having a longer walk. To resolve dilemmas like this, we will have to make a direct comparison between the costs of the two alternatives.

In the hiking example, this means that—if we wish to resolve the problem generally—we will need to find some way to balance the discomfort and inconvenience of wearing wet clothes against the discomfort and inconvenience of extra walking. The appropriate analytical methods for such problems are those of multiattribute utility theory (see, for instance, [6]). These methods work best on cases where there are recurrent, similar decision problems, where there are appropriate scales for measuring the attributes, and where the tradeoffs under consideration are relatively context-independent. This means that analytical methods are unlikely to yield a satisfactory resolution of the hiker's dilemma. Nevertheless, utility analyses can be usefully applied to a broad range of phenomena, and the combination of qualitative reasoning of the sort provided by BDP seems to be a potentially useful one—BDP can eliminate some alternatives as excluded incompatible with feasible maximization of desires, and perhaps in some cases it could deliver a limited range of alternatives appropriate for utility analysis. It remains to be seen, of course, whether this sort of reasoning can be carried out efficiently in cases of practical interest.

In all of the cases considered so far, the element of risk is negligable. BDP does not rely on probabilistic reasoning; its nonmonotonic approach to belief simply excludes alternatives from practical consideration that are incompatible with beliefs. This approach is appropriate in cases where it is important to focus the reasoning, and the consequences of acting inappropriately on false assumptions are bearable. I haven't begun to think through the details of how to extend BDP to cases where risk needs to be reasoned about explicitly; I am sure, however, that this extension would have to involve an integration of some sort of probabilistic and default reasoning; it would certainly require a major reworking of the treatment of belief, and for the moment I think it is best to concentrate on the case in which risk can be neglected.

4. Agent Architectures

Ideas from the AI planning paradigm, together with the popularity of agent design, have inspired the development of *BDI architectures*: frameworks for designing agents that take the attitudes of belief, desire, and intention to be central.[6]

A BDI agent is practical; it performs actions, in the real world or a simulated one. These actions are determined by plans, which are formed on the basis of beliefs (beliefs about the initial state, and about the preconditions and effects of actions) and goals. If we identify adopted plans with intentions and goals with desires, we have the three elements of the BID trio.

Although planning is the central reasoning process that gave rise to the BDI architecture, any agent worth the name must do more than plan. Intentions have to be maintained; at the very least, they have to be removed from the agent's agenda once they have been achieved. And in a more flexible agent, they may need to be revised or abandoned in the light of experience. Goals have to be maintained; like intentions, they need to be dropped when they have been satisfied. And new goals have to come from somewhere. Finally, beliefs have to be maintained; the agent will need to modify beliefs in light of experience, and these modifications may at times involve retractions of existing beliefs.

These key functions will have to be performed by a BDI agent, but it is not clear how the procedures that perform them fit into a model of reasoning that is narrowly based on planning. In an implementation of a BDI agent they would either need to be realized by *ad hoc* procedures, or auxiliary procedures could in some cases be based on independent theories of the reasoning tasks. (The most promising opportunity for such a development is belief revision; there is an extensive literature on this particular reasoning process.)

But to the extent that we want to respect the idea of an agent *architecture*, it is preferable to integrate core functions into the central reasoning processes on which the architecture is based. And there are more practical considerations that make it desirable, in the case of a planning agent, to integrate reasoning functions into the planning process. Here is a brief list of such considerations.

1. Belief revision only becomes nontrivial when the reasoning that produces beliefs is nonmonotonic; typically, beliefs that need to be discarded in the course of belief revision will be ones that are created by fallible reasoning processes, and a belief revision policy that fails to take these processes into account will neglect crucial information.[7] This suggests that nonmonotonic reasoning should be an important component of planning. However, although nonmonotonic logics are often integrated into inertial constraints on state dynamics in planning formalisms (i.e., they are often used to address the frame problem), nonmonotonic domain reasoning is generally ignored

[6] See [4, 13, 12]. The ideas behind BDI architectures also have a philosophical dimension; see [3].

[7] Much of the literature on belief revision ignores the relationship to nonmonotonic domain reasoning, so this claim has to be regarded as controversial.

in planning formalisms. The BDP formalism takes nonmonotonic domain reasoning into account explicitly.

2. For replanning in light of revised beliefs, it could be useful to record the dependencies of plans on beliefs.
3. These plan dependencies could also be invoked in qualitative assessments of risk. A plan can be tested for riskiness by comparing the value of the expected outcome with the value of the outcomes that ensue if various fallible beliefs on which the plan depends fail. These comparisons would be facilitated by a nonmonotonic logic of belief that provided some qualitative assessment of the reliability of beliefs.
4. In a further development of this line of thought, risk assessment could be integrated into planning by seeking to avoid risky plans.
5. Goals come from desires, and—as I pointed out and illustrated in Section 2, desires can be mutually inconsistent in light of beliefs. This provides strong motives for the commonsense planning strategy of planning with flexible goals, and discarding desires that prove to be too costly in light of feasibility considerations. (The vacation planning example from Section 1 was intended to illustrate this point.)
6. A similar point applies to intentions. Agents may need to cut their loses. It is unreasonable to cling to an intention when it becomes too costly to do so, even though the intention remains feasible in principle. By carrying out the planning process in such a way that dependencies on fallible beliefs are recorded, as well as correlations of these beliefs to cost factors, it may be possible to provide in advance for a flexible readjustment of intentions.

The logic BDI suggests an agent made up of attitudes that, after all, are not such a radical departure from the BDI model; you could call it a B^2D^2I agent. Intentions, as before, are the actions that make up adopted plans. Beliefs and desires are divided into two varieties: *prima facie* and all-things-considered. Practical reasoning, which is a generalization of AI-style planning, consists in converting *prima-facie* into all-things-considered attitudes, and in providing a means in terms of feasible actions of satisfying the all-things-considered desires.

5. Conclusion

I think it would be premature at this stage to write a real conclusion. In an AI article, you expect that concluding sections will discuss future work. At this point, things are a little too programmatic for that.

I can think of ways in which the logic BDP could usefully be developed. Except in simple, small-scale cases, the number of potentially relevant *prima facie* desires could be very large—certainly, too large to enter them all by hand. Here, we run up against a fundamental limitation of Default Logic; it treats defaults as rules, and provides no mechanism for inferring rules. If we want to infer desires, we will have to adopt a stronger nonmonotonic logic. But also, we will need to mobilize some intuitions about how desires are inferred, if they are inferred at all (in many cases, desires seem to be felt, rather than concluded). Results and ideas from several areas, including deontic logic, conditional logic, nonmonotonic logic, and the logic of preference, can be helpful here, but there is still much to be done.

To put it another way, the most obvious weakness in the program proposed here is the absence of good ideas about how to implement the formalism for practical reasoning that is proposed here. It would be misleading for me to characterize this as future work. What I hope to do at this point is to get comments on the theoretical ideas, to improve the formulations in this version, and to see whether there seems to be interest in the AI community in pursuing the program. If there is interest in doing that, it will be time to think about implementations; and I hope that I won't have to do that alone.

References

[1] Aristotle. Nicomachean ethics.

[2] David E. Bell, Ralph L. Keeney, and Howard Raiffa, editors. *Conflicting Objectives in Decisions*. John Wiley and Sons, New York, 1977.

[3] Michael Bratman. *Intentions, Plans and Practical Reason*. Harvard University Press, 1987.

[4] Michael Bratman, David Israel, and Martha Pollack. Plans and resource bounded practical reasoning. *Computational Intelligence*, 4:349–355, 1988.

[5] Gerhardt Brewka. Reasoning about priorities in default logic. In Barbara Hayes-Roth and Richard Korf, editors, *Proceedings of the Twelfth National Conference on Artificial Intelligence*, pages 940–945, Menlo Park, California, 1994. American Association for Artificial Intelligence, AAAI Press.

[6] Peter C. Fishburn. Independent preferences. In John Eatwell, Murray Milgate, and Peter Newman, editors, *The New Palgrave: Utility and Probability*, pages 121–127. Macmillan, New York, 1987.

[7] Keith Golden and Daniel Weld. Representing sensing actions: The middle ground revisited. In Luigia Carlucci Aiello, Jon Doyle, and Stuart Shapiro, editors, *KR'96: Principles of Knowledge Representation and Reasoning*, pages 174–185. Morgan Kaufmann, San Francisco, California, 1996.

[8] John F. Horty, Richmond Thomason, and David Touretzky. A skeptical theory of inheritance in nonmonotonic semantic nets. In Kenneth Forbus and Howard Shrobe, editors, *Proceedings of the Sixth National Conference on Artificial Intelligence*, pages 358–363, Menlo Park, California, 1987. American Association for Artificial Intelligence, AAAI Press.

[9] Vladimir Lifschitz. Formal theories of action. In Frank M. Brown, editor, *Proceedings of the 1987 Workshop on the Frame Problem*, pages 35–57, Los Altos, California, 1987. Morgan Kaufmann.

[10] Vladimir Lifschitz. Towards a metatheory of action. In James Allen, Richard Fikes, and Erik Sandewall, editors, *KR'91: Principles of Knowledge Representation and Reasoning*, pages 376–386. Morgan Kaufmann, San Mateo, California, 1991.

[11] John McCarthy and Patrick J. Hayes. Some philosophical problems from the standpoint of artificial intelligence. In B. Meltzer and D. Michie, editors, *Machine Intelligence 4*, pages 463–502. Edinburgh University Press, Edinburgh, 1969.

[12] Anand S. Rao and Michael Georgeff. An abstract architecture for rational agents. In Bernhard Nebel, Charles Rich, and William Swartout, editors, *KR'92. Principles of Knowledge Representation and Reasoning: Proceedings of the Third International Conference*, pages 439–449. Morgan Kaufmann, San Mateo, California, 1992.

[13] Anand S. Rao and Michael P. Georgeff. Modeling rational agents within a BDI-architecture. In James F. Allen, Richard Fikes, and Erik Sandewall, editors, *KR'91: Principles of Knowledge Representation and Reasoning*, pages 473–484. Morgan Kaufmann, San Mateo, California, 1991.

[14] Raymond Reiter. A logic for default reasoning. *Artificial Intelligence*, 13:81–32, 1980.

[15] Eric Sandewall. *Features and Fluents: A Systematic Approach to the Representation of Knowledge About Dynamical Systems*. Oxford University Press, Oxford, 1994.

[16] Murray Shanahan. *Solving the Frame Problem*. The MIT Press, Cambridge, Massachusetts, 1997.

[17] Richmond H. Thomason. Towards a logical theory of practical reasoning. In *Working Notes of the AAAI Spring Symposium on Reasoning about Mental States*, pages 133–142, Menlo Park, CA, 1993. American Association for Artificial Intelligence.