Abductive reasoning through Filtering

Chitta Baral

Department of Computer Science and Engg.
Arizona State University
Tempe, AZ, 85287, USA
chitta@asu.edu

Abstract

Abduction is an inference mechanism where given a knowledge base and some observations, the reasoner tries to find hypotheses which together with the knowledge base explain the observations. A reasoning based on such an inference mechanism is referred to as *abductive reasoning*. Given a theory and some observations, by *filtering* the theory with the observations, we mean selecting only those models of the theory that entail the observations. Entailment with respect to these selected models is referred to as filter entailment. In this paper we give necessary and sufficient conditions when abductive reasoning with respect to a theory and some observations is equivalent to the corresponding filter entailment. We then give sufficiency conditions for particular knowledge representation formalisms that guarantee that abductive reasoning can indeed be done through filtering and present examples from the knowledge representation literature where abductive reasoning is done through filtering. We extend the notions of abductive reasoning and filter entailment to allow preferences among explanations and models respectively and give conditions when they are equivalent. Finally, we give a weaker notion of abduction and abductive reasoning and show the later to be equivalent to filter entailment under less restrictive conditions.

1 Introduction and Motivation

Abduction is an inference mechanism where given a knowledge base and some observations, the reasoner tries to find hypotheses which together with the knowledge base explain the observations. These hypotheses are then referred to as explanations of the observations with respect to the knowledge base. Abduction was introduced by Peirce [Pei58,Pei92] in the beginning of the century and has been used in various AI applications [PMG98,Poo89], including: temporal explanations [Sha89,DMB92,Sha93]; diagnosis [Reg83,Rei87]; planning [Esh88,AKPT91,MD95]; natural language understanding [HSAM90];

default reasoning [PGA87,Poo88a,EK89,KKT93], belief revision and updates [BB95,Bou96]; and formulation of negation as failure [EK89,KKT93].

Although, abduction is often used as a 'backward reasoning' method where observations are explained, we can also do 'forward reasoning' using abduction. This happens when, certain *new conclusions* are entailed by each of the explanations [1] (or each of the preferred explanations) of an observation together with the knowledge base. By *new conclusions* we mean conclusions that are not entailed by the knowledge base without using abduction. Moreover, these conclusions may not be entailed by the theory obtained by simply adding the observations to the knowledge base. The following example makes this point clear.

Example 1 Consider the knowledge base T represented by the following logic program:

$fly(X) \leftarrow bird(X)$
$fly(X) \leftarrow aeroplane(X)$
$haswings(X) \leftarrow aeroplane(X)$
$haswings(X) \leftarrow bird(X)$

Now, suppose we observe $\{fly(tweety)\}$. If we would like to explain this observation in terms of the predicates *bird* and *aeroplane*, then this observation has three explanations:

$E_1 = \{bird(tweety)\}$,
$E_2 = \{aeroplane(tweety)\}$,
$E_3 = \{bird(tweety), aeroplane(tweety)\}$.

It is easy to see that for $i = 1, 2, 3$, $T \cup E_i$ entails the new conclusion $haswings(tweety)$, which is not entailed by T. However, the theory $T \cup \{fly(tweety)\}$ obtained by simply adding the observation to the theory, does not entail $haswings(tweety)$. $\square$

We refer to the entailment of $haswings(tweety)$ from the theory T and the observation $\{fly(tweety)\}$ as *abductive entailment* and reasoning with respect to this entailment relation as *abductive reasoning*. Such entailments were used in [DDS93,Dun93] while formalizing reasoning about actions using abductive logic programming and one such entailment was formally defined – using generalized stable models [KM90,EK88] – with respect to abductive logic programming in [BG94]. Shanahan in Chapter 17 of his book [Sha97] refers to such entailments as *'knowledge assimilation'*.

[1] Pearl [Peab,Peaa] refers to such reasoning from evidence to explanation, and then from explanation to predictions as *transduction*.

Several researchers [San89,Sha97,Lif95] have used a different approach to assimilate observations. In this approach observations are assimilated into a theory by selecting only those models of the theory that entail the observations. Note that since the theory could be in a non-monotonic language, this is not same as adding the observations to the theory. Sandewall, the first (to the best of our knowledge) to have used this in a non-probabilistic setting [San89] refers to this as *filtering*. (In a probabilistic setting the notion of Bayesian *conditioning* is very similar [Gef99].)

The goal of this paper is to study the relationship between these two approaches to assimilate observations. But since the filtering approach is comparatively new and less well-known in non-probabilistic settings, although quite pervasive in the recent research on reasoning about actions, we first give a brief overview of its use.

1.1 Use of filtering in knowledge representation: a brief overview

Three research groups that have extensively used filtering or an equivalent notion in their recent research are:

- Sandewall and his group at Linkoping University.
- Shanahan and his colleagues at Imperial College.
- The Texas action group at the University of Texas at Austin and the University of Texas at El Paso.

1.1.1 Use of filtering at Linkoping University

Filtering was first introduced by Sandewall in [San89], his IJCAI 89 paper titled 'Filter preferential entailment for the logic of action in almost continuous worlds'.

In that paper he defines a binary function Filter whose inputs are a non-empty set of formulas representing observations (Γ_0), and a set of models (S), and defines $Filter(\Gamma_0, S)$ as a subset of S consisting of those members of S which are also models of Γ_0.

The goal of that paper in Sandewall's own words are:

> *The topic of the present paper is now rephrased formally as follows:*
> *Let Γ_0 be a set of formulas representing observations (or goals, in a planning problem); let Γ be a set of formulas representing all the other given informa-*

tion; we look for a formula ϕ which characterizes the set of actions, or plan, that accounts for or obtains the observations. Often ϕ would be a disjunction of expressions each of which characterizes an alternative explanation or plan.

He lists two possible model set criteria for the simple case where there are observations but no actions.

- $M_1 = Min(\ll, Mod(\Gamma \cup \Gamma_0))$
- $M_2 = Filter(\Gamma_0, Min(\ll, Mod(\Gamma)))$

While comparing them, he first says:

> *The definition of M_2 however has an intuitive appeal: since $Min(\ll, Mod(\Gamma))$ is the set of all possible developments in the world regardless of any observations, it would make sense to take the whole set and "filter" it with the given observations.*

He then has a discussion and an example and concludes:

> *Based on this discussion and example, we suggest that the definition of M_2 is the one which should be used for identifying model sets in piecewise continuous worlds, and as the first step in the definition of semantic entailment there. The term* filter preferential entailment *is proposed for semantic entailment using the definition of M_2 as its model set criterium.*

He then considers the case where action occurrences are present, and uses an additional preference condition and proposes the following model set criterium:

$$Min(\prec, Filter(\Gamma_0, Min(\ll, Mod(\Gamma)))),$$

where $\ll$ only compares interpretations with the same action set, and $\prec$ compares interpretations according to the preference of their respective action sets.

The notion of filtering and filter preferential entailment has played a crucial role in subsequent works over the next decade by Sandewall and his colleagues and students at Linkoping University. In particular:

- In his book [San94] he has a subsection (9.4) titled 'Entailment methods that use filtering and their assessments'.
- In his recent survey article [San98] he has a subsection (4.1) on preferential entailment methods where he discusses the use of filter preferential entail-

ment in correctly dealing with postdiction scenarios, where there are also observations about non-initial time points. He also mentions that Lifschitz in [Lif95] having formulated filtering as *nested circumscription* and showing its use for nonmonotonic applications in several areas, including several approaches to the frame problem.

- In the recent survey article [DGKK98] Doherty et al. (also members of Sandewall's group at Linkoping University) discuss their language TAL, which is based on PMON [Doh94a] – a logic from [San89], which is defined using filtering. Thus, while filtering is used only in some of the entailment relations defined in [San94], it is part of the formulation of TAL and is directly or indirectly used in a large body of work (for example, [Doh94b,Doh97,GD96]) done in the context of TAL.

1.1.2 Forced separation

Shanahan in his book [Sha97] uses separation theorems, one of which gives conditions when $CIRC(\Sigma \wedge \Delta; \rho*; \sigma*)$ is equivalent to $CIRC(\Sigma; \rho*; \sigma*) \wedge \Delta$, where Σ and Δ are formulas and $\rho*$ and $\sigma*$ are tuples of predicate symbols, and $CIRC$ refers to circumscription.

In Chapter 16 of his book, he discusses the use of $CIRC(\Sigma; \rho*; \sigma*) \wedge \Delta$ and similar formulations [KL94,Lin95,CE92] where only part of the theories are circumscribed (as opposed to a circumscription of the theory as a whole) in solving the frame problem where there is no separation theorem backing its use. He refers to this as *forced separation.*

His notion of forced separation can be viewed as a special case of filtering where the models of $CIRC(\Sigma; \rho*; \sigma*)$ are filtered by Δ.

In that chapter Shanahan gives several objections against using forced separation, but nevertheless mentions many instances where forced separation has been used and how it can be used to produce elegant solutions to the frame problem. In his own words (Page 308, Section 16.3):

> *Those who are unimpressed by these objections to forced separation can exploit the idea to produce other elegant solutions to the frame problem. This section presents a version of the circumscriptive event calculus in which the forced separation approach is deployed. The result is a much simpler formulation than that of Chapters Fourteen and Fifteen, and is closer in some respects to the logic programming formulation in Chapter Thirteen.*

1.1.3 The Texas action group at Austin and El Paso

The idea of filtering has also been extensively used by the Texas action group [Lif95,KL94,MT95,Tur95,Tur97,MT97], and results in one of the three basic principles in the formulation by McCain and Turner in [MT97,Tur97]. The various papers by the Texas action group where the idea of filtering is used are:

– The notion of nested circumscription in [Lif95] is a generalization of the notion of filtering where multiple levels of filtering can be expressed. Nested circumscription is used in his works in [KL94] and also by others in [BGP98b,BGP98a] for formalizing reasoning about actions.

– Lifschitz in [Lif99] points out that writing observations as constraints – as opposed to facts that are directly added to the theory, as one of the three principles followed by McCain and Turner in their formulation [MT97,Tur97] using Reiter's default logic [Rei80]. Basically, the constraints are used to filter the theory. In case of default logic, they represent constraints by default rules with empty consequents (or $false$ as the consequent) and in case of logic programs constraints are represented using rules with empty head (or $false$ in the head).

Baral and Gelfond also represent observations as constraints in their logic programming formulation in [BG97] and Baral, Gabaldon and Provetti in [BGP98b,BGP98a] use nested circumscription to formulate narratives.

1.2 Relating filtering and abductive reasoning: Motivations

Besides the fact that both filtering and abductive reasoning are used for assimilating observations, our motivations to study the relationship between the two are as follows:

(i) Many of the papers that use filtering (that are mentioned in the previous section) use formal terms, such as 'explanation', from abductive reasoning, without giving any formal results linking their formulation using filtering to abductive reasoning. Our impression after reading these papers were that they were doing some form of abductive reasoning; but we could not find any formal result linking abductive reasoning to filtering. This motivated us to look for a formal connection.

(ii) We would like to know *when* abductive reasoning with respect to a theory and a set of observations is equivalent to using filtering of the models of the theory by the observations. Knowing these conditions is useful in many ways:

– Using these conditions, we would be able to know if indeed the use of filtering in the papers mentioned in the previous section, results in abductive reasoning.

– Filtering seems to have more efficient implementations.

  We are aware of two systems [ELM+98,NS96] that are extremely fast at generating (stable) models of logic programs with constraints, and often use the constraints in the generation process itself to eliminate a large number of models, that would have violated constraints. These systems can be (and have been) used for filtering with respect to logic programs. In particular, the smodels system has been used to do planning [DNK97] in times that are competitive to the best planners available. Although, we are aware of a few abductive systems [DD97,KM97], there has not been a systematic comparison of their performance with respect to other systems.

– Thus knowing when abductive reasoning can be done using filtering, we can formulate our theory appropriately so that we can do abductive reasoning and the corresponding knowledge assimilation using filtering. This provides us with an alternative way to do abductive reasoning.

(iii) When the conditions are satisfied and filtering is equivalent to abductive reasoning, we have additional justification for using filtering. Recall that we mentioned in the previous subsection about Shanahan's objections to forced separation, his term for filtering. We hope that equivalence to abductive reasoning will overcome some of his objections.

(iv) We can also study what kind of reasoning happens with filtering when the conditions are not satisfied. Does filtering under those circumstances still makes sense?

1.3 Organization of the rest of the paper: a summary of contributions

In Section 2 we formally define simple abductive entailment and filtering. In Section 3 we give generic conditions on theories that guarantee the equivalence of abductive reasoning and filter entailment. In Section 4 we develop specific *sufficiency* conditions that satisfy the generic conditions, for the equivalence when the theories are first-order theories, circumscriptive theories, disjunctive logic programs, and default theories. We show that the transformed theory obtained by Console et al. in [CDT91] satisfies our conditions, and hence filtering with respect to it (by just adding the observations, as it is a monotonic theory) is equivalent to abductive reasoning.

In Section 5 we show that the generic conditions of Section 3 are also *necessary* for the equivalence of abductive reasoning and filter entailment. In Section 6 we consider abductive entailment and filtering in presence of preference criteria and expand on the generic conditions to guarantee equivalence in this case. In Section 7 we define a weaker notion of abductive reasoning and show its equivalence with filtering under less restrictive conditions. In Section 8 we conclude and briefly discuss future directions; in particular, the consequences of removing some of the restrictions made in the paper while defining abductive reasoning.

2 Basic Definitions: Simple abduction and Filtering

The main notion in abductive reasoning is the notion of *explanations*. Given a theory T in a language with an entailment relation '$\models$' and an observation Obs, an explanation is normally (for example in Section 17.6 of [Sha97] and in Section 1 of [CDT91]) defined as a formula Δ, such that $T \cup \Delta$ is consistent (i.e., it has a model) and $T \cup \Delta \models Obs$. In addition, most often – including the above mentioned papers, there is a restriction placed on Δ that it be made of *abducible predicates*. Thereafter in many works, preference relations between explanation formulas are defined and used to define preferred or minimal explanations.

Instead of using abducible predicates (as in [Sha97,CDT91,KKT93], we make a slight generalization and use ground abducible literals, but make a restriction [2] that if an atom is abducible so is its negation. This set of ground abducible literals, which we will also refer to as abducibles, is denoted by Abd, and the atoms in this set are referred to by Abd_a.

In our formulation instead of defining explanations as formulas, we define them as (complete) interpretations of the abducibles. Since our ultimate goal is to define abductive entailment, not just to define explanations, this is not a limitation when the theory is in classical logic, possibly augmented with circumscription. Moreover, by having explanations as interpretations of the abducibles, it is easier to define sophisticated model theoretic preference ordering between explanations. This requirement of explanations being a complete subset of abducibles, could be a limitation when the theory is a logic program or a default theory. (For example, [EGL97] does not have such a restriction.) But even in this case our formulation is useful and non-trivial, as we will use

[2] Although we use a more general notion of abducibles than [Sha97,CDT91,KKT93], it would be interesting to extend the results in this paper for the more general case obtained by lifting the restriction that if an atom is abducible then so is its negation.

it (in Sections 4.2 and 4.3) to show the filter abducibility of several logic programming and default theory based formulations [Tur97,BG97,MT95,MT97] of reasoning about actions. In these formulations, each model corresponds to the evolution of the real state of the world, and the abducibles correspond to the value of fluents in the initial state of the world. Thus each potential explanation needs to be a complete interpretation of the abducibles.

Similar to abducibles, we will have a set of ground atoms denoted by Obs_a, which we will call observable atoms. Any formula constructed using atoms in Obs_a, and logical connectives will be referred to as an observable formula. By Obs, we will denote the set of all observable formulas. We will often refer to Obs as *the set of observables*, an element of Obs as an *observation*, and denote the set of ground literals in Obs as Obs_l.

In this simpler form, abduction can now be thought of as a method of reasoning which given a knowledge base T – whose language contains the atoms in Abd_a and Obs_a, and an observation Q, finds possible explanations of Q in terms of a complete subset [3] of the abducibles.

We would now like to formally define explanations and abductive entailment. In this paper T denotes a possibly non-monotonic theory. In particular we will be considering (i) extended and disjunctive logic programs, (ii) default theories, and (iii) theories in propositional and first-order logic possibly augmented with circumscription. For each of these theories, we will now list what is meant by a model, what observations are allowed and how entailment is defined.

- In case of a disjunctive logic program (or an extended logic program), by 'model' we will mean an *answer set* [GL91]. When T is a disjunctive logic program, observations (Q) are allowed to be a (possibly infinite) collection of ground formulas, constructed using atoms in Obs_a, and classical logic connectives. Q may be represented by a set of formulas Q_f with variables, where the variables serve as schema variables and are substituted with ground terms in the language to obtain Q. We say an answer set A entails a formula F if A entails F in the sense of classical logic. We say a disjunctive logic program T entails Q – denoted by $T \models Q$ – if all answer sets of T entail all formulas in Q.

  Given a set of literals L and a logic program T, by $T \cup L$ we mean the logic program $T \cup \{l \leftarrow :\ l \in L\}$.

- In case of a default theory, by 'model' we will mean consequences of an *extension*. The allowed observations (Q) is as in the previous case. We say

[3] We say a set S is a complete set w.r.t. Abd, iff for any atom a in Abd_a – the set of atoms in Abd, either a or $\neg a$ is in S. Often, we will just say S is complete.

an extension E entails a formula F if E entails F in the sense of classical logic. We say a default theory T entails Q – denoted by $T \models Q$ – if all extensions of T entail all formulas in Q.

– In case of propositional theory (or a first order theory) – possibly augmented with circumscription, by 'model' we will mean a classical model. The allowed observations (Q) are formulas in propositional logic (or first order logic) and entailment between T and Q is the classical entailment relation.

  Given a set of formulas Q and a classical theory $\mathcal{T} = \{T_1, \ldots, T_n\}$, where each T_i is a classical theory (possibly augmented with circumscription) by $\mathcal{T} \cup Q$ we will mean the theory $\{Q, T_1, \ldots, T_n\}$.

Definition 1 (Explanation) Let T be a (possibly nonmonotonic) theory with an entailment relation $\models$, and Q be an observation. A complete set of abducibles E (from *Abd*) is said to be an explanation of Q w.r.t. a theory T if $T \cup E \models Q$ and $T \cup E$ is consistent (i.e., it has a model). □

We would now like to define abductive entailment ($\models_{abd}$) with respect to a pair $\langle T, Q \rangle$, which we refer to as an *abductive theory.*

Definition 2 (Abductive Entailment) (i) M is a model of $\langle T, Q \rangle$ if there exists an explanation E of Q w.r.t. T such that M is a model of $T \cup E$.

(ii) For any formula f, $\langle T, Q \rangle \models_{abd} f$ if f is true in all models of $\langle T, Q \rangle$. □

Note that in this paper by *abductive reasoning* we refer to reasoning using the abductive entailment relation $\models_{abd}$ defined above.

Proposition 1 Abductive theories are monotonic with respect to addition of observations. □

Proof: Suppose we have $Q_1 \subseteq Q_2$. Then any explanation of Q_2 with respect to T is an explanation of Q_1 with respect to T. Thus models of $\langle T, Q_2 \rangle$ are models of $\langle T, Q_1 \rangle$ and hence, $\models_{\mathrm{abd}}$ is monotonic with respect to Q. □

Definition 3 Let T be a (possibly nonmonotonic) theory and Q be an observation. By $Filter(T, Q)$, we refer to the set of models of T which entail Q. □

Proposition 2 Entailment with respect to $Filter(T, Q)$ is monotonic with respect to Q. □

Proof: Follows directly from the definition of $Filter(T, Q)$. □

3 Abductive Reasoning through filtering – Main results

The main goal of this paper is to identify conditions on theories, abducibles and observables such that abductive reasoning can be done through filtering. We now formally define such triplets.

Definition 4 (Filter-abducible) A theory T, a set Abd, and a set Obs are said to be *filter-abducible* if for all possible observations $Q \in Obs$; $Filter(T, Q)$ is the set of models of $\langle T, Q \rangle$. $\Box$

Before we define conditions for filter-abducibility, we first show an example where $Filter(T, Q)$ is different than the set of models of $\langle T, Q \rangle$, and then show an example where they are the same.

Example 2 Consider the extended logic program [GL91] T_1:
$p \leftarrow a$
$p \leftarrow b$
Let $Abd = \{a, b, \neg a, \neg b\}$, and $Obs_a = \{p\}$.

Let $Q = \{p\}$. it is easy to see that the models of $\langle T_1, Q \rangle$ are $\{\{p, a, \neg b\}, \{p, b, \neg a\}, \{p, a, b\}\}$, while $Filter(T_1, Q) = \emptyset$.

Now consider T_1' to be the following extended logic program, where we have added four new rules to T_1.

$p \leftarrow a$
$p \leftarrow b$
$a \leftarrow \textbf{not } \neg a$
$\neg a \leftarrow \textbf{not } a$
$b \leftarrow \textbf{not } \neg b$
$\neg b \leftarrow \textbf{not } b$

It is easy to see that the set of models of $\langle T_1', Q \rangle$ is $\{\{p, a, \neg b\}, \{p, b, \neg a\}, \{p, a, b\}\}$ which is same as $Filter(T_1', Q)$.

Note that the set of models of $T_1' \cup Q$ is $\{\{p, a, \neg b\}, \{p, b, \neg a\}, \{p, a, b\}, \{p, \neg b, \neg a\}\}$ and is different from $Filter(T_1', Q)$. $\Box$

Now let us compare T_1 and T_1' and analyze the differences. Syntactically, the difference between them is the last four rules of T_1'. These four rules guarantee that T_1' has at least one model corresponding to each potential explanation (i.e., interpretation of the abducibles). Since unlike during abductive reasoning, during filtering there is no scope to try each potential explanation so as not to miss any explanation, the models of the theory should enumerate the potential explanations. This is missing in T_1 and therefore T_1 is not filter-abducible with respect to the above mentioned Abd and Obs. On the other hand, T_1' satisfies this criteria and it is filter-abducible with respect to the same Abd and Obs. In the following paragraphs, we precisely state the above mentioned property as condition B. We now discuss additional conditions that may be important.

For filtering to be equivalent to abductive reasoning, each one of the models obtained by filtering a theory T with an observation Q should contain an explanation of the observation. In that case the abducibles in those models consist of an explanation. For that to happen, the theory must be such that the abducibles uniquely determine the observables. If we want to avoid making any restrictions on the observables then the theory must be such that the abducibles uniquely determine the model. These two conditions are precisely stated below as Condition A' and Condition A, respectively.

We now formally state the above mentioned conditions.

Condition A
If M is a model of theory T then M is the unique model of the theory $T \cup (M \cap Abd)$.

Intuitively, Condition A means that the models of a theory T can be characterized by just the abducible literals in that model. It requires that if M is a model of a theory T then M should be the only model of the theory $T \cup (M \cap Abd)$. This is a strong condition, as in many cases $T \cup (M \cap Abd)$ may have multiple models. To take into account such cases, we can weaken condition A, by the following condition. But we will need to use Obs as part of our condition.

Condition A'
If M is a model of theory T then:
(i) M is a model of $T \cup (M \cap Abd)$;
(ii) all models of $T \cup (M \cap Abd)$ are also models of T; and
(iii) all models of $T \cup (M \cap Abd)$ agree on Obs; where two models M_1 and M_2 of a theory are said to agree on Obs if for all $Q \in Obs$, we have $M_1 \models Obs$ iff $M_2 \models Obs$.

Condition B
For any complete subset E of Abd if $T \cup E$ is consistent then there exists a model M of T such that $M \cap Abd = E$

Intuitively, Condition B means that the theory T has models corresponding to each possible interpretation of the abducibles. I.e., the models of the theory T enumerate the possible explanations.

Lemma 1 Let T be a theory satisfying Conditions A and B. Let E be any complete subset of Abd. If M is a model of $T \cup E$, then $M \cap Abd = E$. $\square$

Proof:
Let E be any complete subset of Abd.
From Condition B we have that there exists a model M' of T such that $M' \cap Abd = E$.
But From Condition A, we have that M' is the unique model of $T \cup E$.
Thus if M is a model of $T \cup E$, then $M = M'$ and thus $M \cap Abd = E$. $\square$

Theorem 1 If T, Obs and Abd satisfy conditions A' and B then they are filter-abducible; i.e., they satisfy the following:
"for all observations $Q \in Obs$, $Filter(T, Q)$ is the set of models of $\langle T, Q \rangle$." $\square$

Proof:
(a) We first show that if T, Obs and Abd satisfy conditions A' and B then all elements of $Filter(T, Q)$ are models of $\langle T, Q \rangle$.

Let M be an element of $Filter(T, Q)$
$\Rightarrow$ M is a model of T and M entails Q.
From A' we have $T \cup (M \cap Abd)$ is consistent and all models of $T \cup (M \cap Abd)$ are also models of T and agree on the observables.
Let $E = M \cap Abd$.
We then have $T \cup E \models Q$ and $T \cup E$ is consistent.
$\Rightarrow$ There exists an explanation E of Q w.r.t. T such that M is a model of $T \cup E$.
$\Rightarrow$ M is a model of $\langle T, Q \rangle$.

(b) We will now show that if T, Obs and Abd satisfy conditions A' and B then all models of $\langle T, Q \rangle$ are in $Filter(T, Q)$.

Let M be a model of $\langle T, Q \rangle$.
$\Rightarrow$ There exists an explanation E of Q w.r.t. T such that M is a model of $T \cup E$.
$\Rightarrow$ There exists a complete set of abducibles E such that $T \cup E \models Q$ and $T \cup E$ is consistent and M is a model of $T \cup E$.
From Condition B we have that there exists a model M' of T such that $M' \cap Abd = E$. But from Condition A' all models of $T \cup E$ are also models of T and agree on Q.
Thus M is a model of T. Since M is also a model of $T \cup E$ and $T \cup E \models Q$, we have M entails Q.
Thus $M \in Filter(T, Q)$. $\square$

Lemma 2 If T, Obs and Abd satisfy condition A then they also satisfies Condition A'. $\square$

Proof: Straight forward.

Corollary 1 If a theory T, and abducibles Abd satisfy conditions A and B then for any set of Obs_l in the language of T, the theory T, Abd and Obs are filter-abducible. □

Proof: Follows directly from Lemma 2 and Theorem 1.

The main significance of the above corollary is that by requiring the more restrictive condition A we have more flexibility with the observables.

We now give some examples where we can verify filter-abducibility by verifying the above mentioned conditions.

Example 3 The propositional theory $T_2 = p \iff a \vee b$ with $Abd = \{a, b, \neg a, \neg b\}$ and $Obs_l = \{p, \neg p\}$ satisfies Conditions A and B.

This is because the models of T_2 are $\{\{\neg a, \neg b, \neg p\}, \{\neg a, b, p\}, \{a, \neg b, p\}, \{a, b, p\}\}$ and it is easy to see that for each model M in this set, $T_2 \cup M$ has the only model M. □

Example 4 The following logic program T_2'

$q \vee r \leftarrow a$

$p \leftarrow a$

$p \leftarrow b$

$a \vee \neg a \leftarrow$

$b \vee \neg b \leftarrow$

with abducibles $\{a, b, \neg a, \neg b\}$ and $Obs_a = \{p\}$ satisfies Conditions A' and B.

This can be verified as follows. The program T_2' has six models, which are: $\{a, b, p, q\}, \{a, b, p, r\}, \{a, \neg b, p, q\}, \{a, \neg b, p, r\}, \{\neg a, b, p\}$, and $\{\neg a, \neg b\}$. Let us refer to them as M_1, M_2, M_3, M_4, M_5 and M_6 respectively. Consider the model M_1. Let us verify that it satisfies Condition A'. It is easy to see that M_1 is a model of $T_2' \cup (M_1 \cap Abd) = T_2' \cup \{a, b\}$, and all models of $T_2' \cup \{a, b\}$ are models of T_2' and they agree on Obs. We can similarly verify that the other models of T_2' satisfy the conditions of A'. (Note that if we include q or r in Obs_a, condition A' will no longer be satisfied.)

In this example there are four complete subsets of Abd. These are: $\{a, b\}, \{a, \neg b\}, \{\neg a, b\}, \{\neg a, \neg b\}$. Consider $E = \{a, b\}$. Since $T_2' \cup E$ is consistent, we need to verify that there exists a model M of T_2' such that $M \cap Abd = E$. M_1 is such a model of T_2'. We can similarly verify that the other complete subsets of Abd satisfy condition B. □

4 Filter-abducibility in monotonic and non-monotonic theories

Depending on whether a theory is a monotonic or a non-monotonic theory filtering can be achieved in different ways. We now list how filtering can be achieved in several different knowledge representation formalisms.

- If T is in a monotonic language to filter T by Q we just add Q to T. Otherwise, i.e. if T is in a non-monotonic language to filter T by Q we can not just add Q to T. This is because, as stated in Proposition 2, filtering has the monotonicity property.
- If T is a logic program to filter T by Q we can view Q as an integrity constraint. (Often integrity constraints are expressed through rules with empty head or with $false$ in its head. They can also be expressed using rules of the form $p \leftarrow c, \mathbf{not}\ p$, where p is a new atom, and c is the constraint.) Such a view is used in [Tur97,BG97] to assimilate observations to action theories expressed in a logic program. In [Llo87] an algorithm is given to translate general formulas to the restricted syntax of integrity constraints in logic programs.
- If T is a default theory to filter T by Q we can express Q as a default rule with an empty consequent or $false$ as its consequent. Such a view is used in [Tur97] to assimilate observations to action theories expressed in a logic program.
- If T is a nested abnormality theory (NAT) then we can filter T w.r.t. Q by the NAT consisting of two blocks; one of which is T and the other is Q. As a non-monotonic knowledge representation formalism NAT has the advantage that it makes it easy to have multiple levels of filtering.

Our goal in this section is to develop and describe specific sufficiency conditions that can be used to show filter-abducibility in particular knowledge representation formalisms. The ones we consider are: circumscriptive theories; first order theories; disjunctive logic programming; and Reiter's default logic. But first we define the notion of a *theory encoding a function*, which will play an important role when we define the specific sufficiency conditions for each of the knowledge representation formalism that we will be considering.

Definition 5 A theory T is said to encode a function (or is functional) from a set of literals, called *input*, to a set of literals called *output* if for any complete subset E of input such that $T \cup E$ is consistent, all models of $T \cup E$ agree on the literals from output. $\square$

The following proposition states conditions for the filter-abducibility of propositional and first-order theories.

Proposition 3 A propositional or first-order theory (possibly augmented with circumscription) T is filter-abducible with respect to abducibles Abd and observables Obs if T is functional from Abd to Obs_l.[4] $\square$

Proof:
The proof is based on showing that the if part of the proposition implies the conditions A' (i, ii, and iii) and B. Earlier in Theorem 1 we proved that A' and B guarantee filter-abducibility.

It is easy to see that, because of monotonicity of the theory T, conditions $A'(i)$ and $A'(ii)$ are true. Condition $A'(iii)$ is true due to the condition that T is functional from Abd to Obs. We now only need to show that Condition B holds. Let E be a complete subset of Abd such that $T \cup E$ is consistent. Let M be a model of $T \cup E$. Obviously, $E \subseteq M$. Since E is a complete subset of Abd we have that $M \cap Abd = E$. Thus Condition B holds. $\square$

We now discuss two examples whose filter-abducibility can be deduced using Proposition 3.

Example 5 Consider the following theory T_3:

$r(p) \leftarrow q(a)$
$r(p) \leftarrow q(b)$

and let $Obs_a = \{r(p)\}$ and $Abd = \{q(a), q(b), \neg q(a), \neg q(b)\}$.

The theory T_3 with Obs and Abd is not filter-abducible. Suppose our observation is $r(p)$. Then the models of T_3 are $\{q(a), q(b), r(p)\}$, $\{\neg q(a), q(b), r(p)\}$, $\{q(a), \neg q(b), r(p)\}$, $\{\neg q(a), \neg q(b), r(p)\}$, and $\{\neg q(a), \neg q(b), \neg r(p)\}$.

It is easy to see that the elements of $Filter(T_3, \{r(p)\})$ are $\{q(a), q(b), r(p)\}$, $\{\neg q(a), q(b), r(p)\}$, $\{q(a), \neg q(b), r(p)\}$, and $\{\neg q(a), \neg q(b), r(p)\}$, while the models of $\langle T_3, \{r(p)\}\rangle$ are $\{q(a), q(b), r(p)\}$, $\{\neg q(a), q(b), r(p)\}$, and $\{q(a), \neg q(b), r(p)\}$. Thus, T_3, Obs and Abd are not filter-abducible.

Since T_3 is a first order theory, using Proposition 3 we can conclude that T_3 does not encode a function from Abd to Obs. Indeed we can verify this by considering the input $E = \{\neg q(a), \neg q(b)\}$. The theory $T_3 \cup E$ has two

[4] We use Obs_l instead of Obs because functional is defined between sets of literals.

models $\{r(p), \neg q(a), \neg q(b)\}$ and $\{\neg r(p), \neg q(a), \neg q(b)\}$ with different values of the observable $r(p)$, and hence is not functional from Abd to Obs.

Let us now consider $\mathcal{T}_1 = CIRC(T_3; r; \emptyset)$, then $Filter(\mathcal{T}_1, \{r(p)\})$, and the set of models of $\langle \mathcal{T}_1, \{r(p)\} \rangle$ are the same. What happens is that by minimizing r in $\mathcal{T}_1$, the resulting theory is equivalent to $\mathcal{T}_2 = (r(p) \Leftrightarrow q(a) \vee q(b))$, which is functional from Abd to Obs. □

In the above example, $\mathcal{T}_2$ is obtained by doing completion [Cla78] of $\mathcal{T}_1$ with respect to the predicate r. In [CDT91], Console et al. describe more general results where they transform hierarchical propositional and first-order theories so that abductive reasoning with respect to the original theory is achieved by simply adding the observation – which in this case is same as filtering – to the transformed theory. We further discuss this in the next subsection.

4.1.1 Knowledge assimilation through completion

A hierarchical propositional theory is a collection of clauses of the form

$$L_1 \wedge \ldots \wedge L_n \rightarrow p$$

where each L_i is a literal and p is an atom, and where each atom in the theory is assigned a level (a natural number) such that in a clause of the above form, the level of p is greater than the level of L_i, for any $1 \leq i \leq n$. The set Abd_a of abducible atoms is the set of atoms that do not appear in the right hand side of any of the clauses.

Consider such a theory T with $p_1, \ldots, p_n$ as all the non-abducible atoms in it. The transformation T_c is the set of equivalences $\{p_i \leftrightarrow Q_{i1} \vee \ldots \vee Q_{im} \ : \ i = 1, \ldots, n\}$, where $\{Q_{ij} \rightarrow p_i, j = 1, \ldots, m\}$ are the set of clauses in T having p_i in their right hand side.

Console et al. [CDT91] show that abductive reasoning with respect to a hierarchical propositional theory T is equivalent to deduction with respect to the transformed theory T_c. They then generalize their result to the first-order case. The sort of completion used by Console et al. was first used by Poole in a more limited setting in [Poo88b]. Similar results were also independently developed by Konolige in [Kon92]. Such results are referred to as 'abduction through deduction' in the literature [Sha89,Sha97,KKT93] and Poole [Poo88b,Poo89] was perhaps the first one to pursue research along these lines in a non-probabilistic setting.

Coming back to T_c, let us assume that, as in [CDT91], observables are consistent conjunction of literals with no occurrence of abducible atoms. It is easy to see that T_c is functional from abducibles to observable literals. Thus using Proposition 3 we can conclude that T_c is filter-abducible with respect to the above mentioned abducibles and observables. Proposition 3 thus supports the result of Console et al. But it also has several additional implications:

- For example, Proposition 3 also explains the restrictions posed on the clauses.

  If we were to allow disjunctions in the right hand side of the clauses, and/or not require the clauses to be hierarchical, we can no longer guarantee that T_c will be functional from abducibles to observable literals. As a result we will not have a guarantee that T_c will be filter-abducible.

  For example, let T be $p \rightarrow a \vee b$. Then T_c will be $p \leftrightarrow a \vee b$. Suppose we observe a, then using deduction with respect to T_c we will conclude p as an explanation. But p is not an explanation of a with respect to T. The fact that T_c is not functional from $\{p, \neg p\}$ to $\{a, b, \neg a, \neg b\}$, warns us to not expect T_c to be filter-abducible with respect to them.

- Proposition 3 also suggests ways to extend the results of Console et al.

  If we can find a larger class of T, abducibles, or observables such that T_c will be functional from abducibles to observable literals then we will have a more general result.

  A simple generalization is the case, where we allow observables to include the abducibles. Since T_c remains functional from abducibles to (the expanded) observable literals the filter-abducibility result still holds.

We will now give an example from the domain of reasoning about actions, where we have filter-abducibility.

4.1.2 Filter-abducibility of action theories

Let us consider a subset of the action description language $\mathcal{A}$ [GL93] where a domain descriptions may have two kinds of propositions: *initial value propositions* and *effect propositions*. We will refer to this language as $\mathcal{A}_0$. (The alphabet of $\mathcal{A}_0$ consists of three disjoint nonempty sets of symbols called *fluents*, *actions*, and *situations*. We will also assume that S_0 is one of the situations in the language of $\mathcal{A}_0$ and by a *fluent literal* we will mean a fluent possibly preceded by $\neg$.)

An initial value proposition is a proposition of the form

initially F

where F is a fluent literal.

An effect proposition is a proposition of the form

A **causes** F **if** $P_1, \ldots P_n$

where $F, P_1, \ldots, P_n$ are fluent literals and A is an action. We refer to $\{P_1, \ldots, P_n\}$ as the precondition of the above effect proposition.

We will be restricting domain descriptions to collections of effect propositions and initial value propositions such that for any two effect propositions that describe the effect of the same action a on complementary f's, and have $P_1, \ldots, P_n$ and $Q_1, \ldots, Q_m$ as their preconditions we have that $\{P_1, \ldots, P_n\} \cap \{\overline{Q_1}, \ldots, \overline{Q_m}\} \neq \emptyset$. (This will guarantee that an action does not cause a fluent and its complement in the same situation.)

Following is a translation of domain descriptions in this language to a many-sorted nested abnormality theory [Lif95] in situation calculus notation. (The sorts in the theory are: *actions*, *fluents*, and *situations*. The variables for the three sorts will be denoted by possibly indexed letters a, f and s, respectively, unless otherwise stated. The language includes the actions and fluents in $\mathcal{A}$ as action constants, and fluent constants, and S_0 as one of the situation constants. In addition, the theory has the predicate constants: $Causes$ – with arguments of sort actions, fluents and situations, $Holds$ – with arguments of sort fluents and situations, and $Initially$ – with arguments of sort fluents, and the function constant Res from actions and situations to situations. The theory also has the standard Unique name axioms (UNA) [5] and Domain closure axioms (DCA) [6] (say as in [Lif95,KL94]). In the theory, we use the notation H as a short hand, where for a fluent F, $H(F, S)$ denotes $Holds(F, S)$ and $H(\neg F, S)$ denotes $\neg Holds(F, S)$.)

[5] If $A_1, \ldots, A_n$ are the only actions in the language, then $UNA[actions]$ stands for $A_1 \neq A_2, A_2 \neq A_3$, etc.

[6] If $A_1, \ldots, A_n$ are the only actions in the language, then $DCA[actions]$ stands for $(\forall a). a = A_1 \vee \ldots \vee a = A_n$

$$
SC_c(D_l) = \left\{
\begin{array}{l}
UNA[actions], UNA[fluents], UNA[situations] \\
DCA[actions], DCA[fluents], DCA[situations] \\
(1) \neg Causes^+(a,f,s) \wedge \neg Causes^-(a,f,s) \supset [Holds(f,s) \equiv Holds(f, Res(a,s))] \\
(2) Causes^+(a,f,s) \supset Holds(f, Res(a,s)) \\
(3) Causes^-(a,f,s) \supset \neg Holds(f, Res(a,s)) \\
(4) Initially(f) \equiv Holds(f, S_0) \\
(5) Initially(F) \qquad\qquad\qquad (\text{for each } \textbf{initially} \;\; F \in D) \\
(6) \neg Initially(F) \qquad\qquad\qquad (\text{for each } \textbf{initially} \;\; \neg F \in D) \\
\{ \text{ min } Causes^{+(-)} \;\; : \\
\quad (7) H(P_1, s) \wedge \ldots \wedge H(P_n, s) \supset Causes^{+(-)}(A, F, s) \\
\qquad\qquad\qquad\qquad (\text{for each } A \textbf{ causes } \;(\neg)F \textbf{ if } \; P_1, \ldots, P_n \in D) \\
\} \\
\end{array}
\right.
$$

In the above theory let us have Abd_a as the set of atoms using $Initially$ and Obs_a as the set of atoms using $Holds$. By showing that the above theory is functional from abducibles to observable literals, we can use Proposition 3 [7] to conclude that assimilation of observations with respect to the above theory can be done by simply adding the observation to the theory.

Let us now briefly argue why the above theory is functional from abducibles to observable literals. From axiom (4) it is clear that given a complete set of 'Initially' literals, we have a unique (and complete) set of $Holds$ literals at the situation s_0, in all models of the theory. We will now argue that given a complete set of $Holds$ literals at the situation s, for any action A, there is a unique (and complete) set of $Holds$ literals at the situation $Res(A, s)$, in all models of the theory. (A detailed proof of this is given in [BGP98a].)

The NAT block defining $Causes^{+(-)}$ guarantees that given a complete set of 'Holds' literals at a situation s, we have a unique (and complete) set of $Causes^{+(-)}$ literals at the situation s, w.r.t. any action A, and any fluent F. Because of our restrictions on domain descriptions, we will never have $Causes^+(a, f, s)$ and $Causes^-(a, f, s,)$ to be true in the same model. This fact together with the axioms (1) - (3) guarantee that given a complete set of

[7] Actually we need to extend Proposition 3 so as to be applicable to nested abnormality theories. This extension is straightforward.

$Holds$ literals at any situation s, and for any action A, there is a unique (and complete) set of $Holds$ literals at the situation $Res(A, s)$ in all models of the theory.

Thus SC_c is functional from its abducibles to its observable literals and hence it is filter abducible.

4.2 Filter abducibility of Disjunctive Logic Programs

In this subsection we will discuss the filter-abducibility of disjunctive logic programs. In particular, we will give some sufficiency conditions that guarantee that conditions A' and B holds. But first, for the sake of completeness, we will give the definitions and results related to the notion of splitting [LT94] which will be used in the sufficiency conditions.

A disjunctive logic program (DLP) is a collection of rules of the form

$$l_1 \;\vee\; \ldots \;\vee\; l_k \leftarrow l_{k+1}, \ldots l_m, \textbf{not } l_{m+1}, \ldots, \textbf{not } l_n \tag{1}$$

where $k \geq 0$, and each l_i is a literal, i.e., an atom possibly preceded by $\neg$, and **not** is the negation as failure operator. Expression on the left hand (right hand) side of $\leftarrow$ is called the *head* (the *body*) of the rule. Both, the head and the body of (1) can be empty. Unless otherwise stated, we assume that rules with variables are used as shorthand for the set consisting of all their ground instantiations. Intuitively the rule can be read as: if $l_{k+1}, \ldots, l_m$ are believed and it is not true that $l_{m+1}, \ldots, l_n$ are believed then at least one of $\{l_1, \ldots, l_k\}$ is believed. For a rule r of the form (1) the sets $\{l_1, \ldots, l_k\}$, $\{l_{k+1}, \ldots, l_m\}$ and $\{l_{m+1}, \ldots, l_n\}$ are referred to as $head(r)$, $pos(r)$ and $neg(r)$ respectively. $lit(r)$ stands for $head(r) \cup pos(r) \cup neg(r)$. For any DLP Π, $head(\Pi) = \bigcup_{r \in \Pi} head(r)$. For a set if predicates S, $Lit(S)$ denotes the set of literals with predicates from S. For a DLP Π, $Lit(\Pi)$ denotes the set of literals with predicates from the language of Π. When it is clear from the context we write Lit instead of $Lit(\Pi)$.

Definition 6 *(Splitting set) [LT94]*
A *splitting set* for a program Π is any set U of literals such that, for every rule $r \in \Pi$, if $head(r) \cap U \neq \emptyset$ then $lit(r) \subset U$. If U is a splitting set for Π, we also say that U splits Π. The set of rules $r \in \Pi$ such that $lit(r) \subset U$ is called the *bottom* of Π relative to the splitting set U and denoted by $bot_U(\Pi)$. The subprogram $\Pi \setminus bot_U(\Pi)$ is called *the top of* Π relative to U. $\square$

Definition 7 *(Partial evaluation) [LT94]*
The partial evaluation of a program Π with splitting set U w.r.t. a set of

literals X is the program $e_U(\Pi, X)$ defined as follows. For each rule $r \in \Pi$ such that:

$$(pos(r) \cap U) \subset X \ and \ (neg(r) \cap U) \cap X = \emptyset$$

put in $e_U(\Pi, X)$ all the rules $r^{'}$ that satisfy the following property:

$$head(r') = head(r),\ pos(r') = pos(r) \setminus U,\ neg(r') = neg(r) \setminus U$$

□

Definition 8 *(Solution) [LT94]*
Let U be a splitting set for a program Π. A solution to Π w.r.t. U is a pair $\langle X, Y \rangle$ of literals such that:

- X is an answer set for $bot_U(\Pi)$;
- Y is an answer set for $e_U(\Pi \setminus bot_U(\Pi), X)$;
- $X \cup Y$ is consistent. □

Lemma 3 *(Splitting Lemma) [LT94]*
Let U be a splitting set for a program Π. A set A of literals is a consistent answer set for Π if and only if $A = X \cup Y$ for some solution $\langle X, Y \rangle$ to Π w.r.t. U. □

We are now ready to give sufficiency conditions that guarantee filter-abducibility for disjunctive logic programs.

Proposition 4 An extended logic program T is filter-abducible with respect to abducibles Abd and observables Obs if
(i) T is functional from Abd to Obs_l,
(ii) for all $l, \neg l \in Abd$, $l \vee \neg l \leftarrow$ is in T, and
(iii) Abd is a splitting set for T. □

Proof:
We prove this by showing that the conditions (i) – (ii) above imply the conditions A' and B which in turn guarantee filter abducibility of a theory.

(a) **Showing** $A'(i)$
We now show that the conditions (i) – (ii) implies $A'(i)$.
When T is inconsistent this result trivially holds. Let us consider the case when T is consistent.
Let M be a consistent answer set of T
$\Rightarrow$ M is a minimal answer set of T^M.

$\Rightarrow M$ is a minimal answer set of $T^M \cup (M \cap Abd)$.
$\Rightarrow M$ is a minimal answer set of $(T \cup (M \cap Abd))^M$.
$\Rightarrow M$ is an answer set of $T \cup (M \cap Abd)$.
$\Rightarrow$ Condition $A'(i)$ holds.

(b) **Showing** $A'(ii)$
Let M be an answer set of T. It is clear that $M \cap Abd$ is a complete set of abducible literals and is an answer set of $bot_{Abd}(T)$. Thus by Lemma 3, all answer set of $T \cup (M \cap Abd)$ are answer sets of T. Thus condition $A'(ii)$ holds.

(c) **Showing** $A'(iii)$
Since T is functional from Abd to Obs and $M \cap Abd$ is a complete set of abducible literals, it is clear that condition $A'(iii)$ holds.

(d) **Showing B**
Let E be any arbitrary complete subset of Abd. From condition (ii) of the proposition, E is an answer set of $bot_{Abd}(T)$. Hence by Lemma 3, there exists an answer set M of T, such that $M \cap Abd = E$. Thus condition B is satisfied.
$\Box$

Let us again consider the domain description in the previous section and show how it can be formalized into a filter-abducible disjunctive logic program. Here we use the logic programming notation, with variables in capital letters, and constants in small letters.

1. *Inertia Axioms*:

$$\left.\begin{array}{l} (1a)\ holds(F, Res(A,S)) \leftarrow holds(F,S), \textbf{not}\ ab(A, \overline{F}, S), \\ (1b)\ \neg holds(F, Res(A,S)) \leftarrow \neg holds(F,S), \textbf{not}\ ab(A, F, S), \end{array}\right\} (1a) - (1b)$$

2. *Translating e–propositions*:

The translation of an e–proposition "a **causes** f **if** $p_1, \ldots, p_n$" consists of

$$\left.\begin{array}{l} (2a)\ h(f, res(a,S)) \leftarrow h(p_1,S), \ldots, h(p_n,S) \\ (2b)\ ab(a,f,S) \leftarrow h(p_1,S), \ldots, h(p_n,S) \end{array}\right\}$$

where for a fluent g, $h(g,s)$ denotes the literal $holds(g,s)$ and $h(\neg g, s)$ denotes the literal $\neg holds(g,s)$.

The effect axiom allows us to prove that f will hold after a, if the preconditions are satisfied.

3. *Initial value proposition*

An initial proposition " **initially** f" is translated as

$$(3)\ h(f, s_0) \leftarrow$$

4. *Full awareness about the initial situation rule*

$$(4)\ holds(F, s_0) \quad \vee \quad \neg holds(F, s_0) \leftarrow$$

For a given domain description D, the filter-abducibility of a program generated by the steps 1-4 can be easily verified by showing that the program satisfies the conditions of Proposition 4 when abducibles are literals about *holds* in the situation s_0 and observable literals are literals about *holds* in any situation. (The detailed analysis of an extension of this program is given in [BG97].)

To assimilate observations of the form "f **after** $a_1, \ldots, a_m$" we need to filter the above logic program. In logic programming filtering can be done using integrity constraints (or through clauses with empty heads). Thus in this case assimilation is done by adding integrity constraints of the form below to the logic program.

$$(5) \qquad \leftarrow \mathbf{not}\ h(f, [a_1, \ldots, a_m])$$

where $[a_1, \ldots, a_m]$ stands for the ground term $Res(a_m, Res(a_{m-1}, \ldots, Res(a_1, s_0) \ldots))$.

4.3 Filter abducibility of Default theories

Sufficiency conditions for the filter-abducibility of Reiter's default theory is very similar to the sufficiency conditions for disjunctive logic programs that we gave in the last subsection.

Proposition 5 A default theory T is filter abducible with respect to abducibles Abd and observables Obs if
(i) T is functional from Abd to Obs_l,
(ii) $\{\frac{:l}{l}, \frac{:\neg l}{\neg l}\ :\ l \in Abd\}$ is a subset of T, and
(iii) Abd is a splitting set [Tur96] for T. □

Proof:
Similar to the proof of Proposition 4. □

The logic program in the previous section can be written as a default theory satisfying the conditions in Proposition 5. The main trick is to write the 'full awareness about the initial situation rule' as defaults of the following form:

$$\frac{: holds(F, s_0)}{holds(F, s_0)} \qquad\qquad \frac{: \neg holds(F, s_0)}{\neg holds(F, s_0)}$$

Turner in [Tur97] uses such defaults.

Observations of the form "f **after** $a_1, \ldots, a_m$" can then be assimilated by adding

$$\frac{\neg holds(f, [a_1, \ldots a_m])}{false}$$

to the corresponding default theory. Such defaults with $false$ in their conclusion satisfy the role of filtering in default theories.

5 Are the conditions A' and B necessary?

In Section 3 we show that the conditions A' and B are sufficient for filter-abducibility. In this section we show that they are also necessary.

Theorem 2 *Let T be a theory, and Obs and Abd be observables and abducibles such that, for all $Q \in Obs$, $Filter(T, Q)$ is equivalent to the set of models of $\langle T, Q \rangle$. Then T, Obs and Abd satisfy the conditions B, $A'(i)$, $A'(ii)$ and $A'(iii)$.* □

Proof:

(i) Suppose T, Obs and Abd do not satisfy condition B. That means there exists an $E \subseteq Abd$, such that $T \cup E$ is consistent, but there does not exist a model M of T such that $M \cap Abd = E$.

Since $T \cup E$ is consistent it has at least one model. Let M^* be a model of $T \cup E$. Let Q be the conjunction of the literals in $M^* \cap Obs_l$. Obviously M^* is a model of $\langle T, Q \rangle$. Since M^* is a model of $T \cup E$, $M^* \cap Abd = E$. But then from our initial assumption, M^* can not be a model of T. Hence M^* is not in $Filter(T, Q)$. This contradicts the assumption in the lemma that $Filter(T, Q)$ is equivalent to the set of models of $\langle T, Q \rangle$. Hence T must satisfy condition B.

(ii) Suppose T, Obs and Abd do not satisfy condition $A'(i)$. That means there is a model M of T which is not a model of $T \cup (M \cap Abd)$. Let $Q = M \cap Abd$. Obviously, M is in $Filter(T, Q)$. We will now show that M is not a model of $\langle T, Q \rangle$. Suppose M is a model of $\langle T, Q \rangle$. That means there is an $E \subseteq Abd$, such that M is a model of $T \cup E$ and $M \models Q$. But then $M \cap Abd = E$, and this contradicts our initial assumption that M is not a model of $T \cup (M \cap Abd)$. Hence M is not a model of $\langle T, Q \rangle$. But this contradicts the assumption in the lemma that $Filter(T, Q)$ is equivalent to the set of models of $\langle T, Q \rangle$. Hence T must satisfy condition $A'(i)$.

(iii) Suppose T, Obs and Abd do not satisfy condition $A'(ii)$. That means there is a model M of T such that all models of $T \cup (M \cap Abd)$ are not models of T. Let $Q = M \cap Abd$. Let M' be a model of $T \cup (M \cap Abd)$ which is not a model of T. Obviously, M' is a model of $\langle T, Q \rangle$. But it is not an element of $Filter(T, Q)$. This contradicts the assumption in the lemma that $Filter(T, Q)$ is equivalent to the set of models of $\langle T, Q \rangle$. Hence T must satisfy condition $A'(ii)$.

(iv) Suppose T, Obs and Abd do not satisfy condition $A'(iii)$. That means there is a model M of T such that all models of $T \cup (M \cap Abd)$ do not agree on the observables. This means $Obs_l \setminus Abd \neq \emptyset$. Let $Q = (M \cap Obs_l) \cup (M \cap Abd)$. Obviously, M is in $Filter(T, Q)$. We will now show that M is not a model of $\langle T, Q \rangle$. Suppose M is a model of $\langle T, Q \rangle$. That means there is a complete subset E of Abd, such that M is a model of $T \cup E$ and $T \cup E \models Q$. Since E is a complete subset of Abd, $E = M \cap Abd$. Since all models of $T \cup (M \cap Abd)$ do not agree on observables $T \cup E \not\models Q$. This contradicts our assumption and hence M is not a model of $\langle T, Q \rangle$. But then, we have a contradiction to the assumption in the lemma that $Filter(T, Q)$ is equivalent to the set of models of $\langle T, Q \rangle$. Hence T must satisfy condition $A'(iii)$. $\square$

We would like to mention that the known ways to satisfy condition B in default theories and logic programs are to have rules of the form $l \vee \neg l$ (or have two rules of the form $l \leftarrow \textbf{not}\ \neg l; \neg l \leftarrow \textbf{not}\ l$) for all abducible atoms l in logic programs and defaults of of the form $\frac{:l}{l}$ and $\frac{:\neg l}{\neg l}$ for all abducible atoms l in default theories. (The former was first used in [Ino91] to relate semantics of abductive logic programs – based on the generalized stable models [KM90], and extended logic programs. The later was used in [Tur97,MT93].) Hence the necessity of condition B for filter-abducibility makes it necessary (to the best of our knowledge) to have such rules and defaults in filter-abducible logic programs and default theories, respectively.

6 Preferential abductive reasoning and Filtering

So far we have considered simple abductive reasoning and filtering. But often (as in [San89]) both abductive reasoning and filtering is accompanied by some preference criteria. In this section we extend the definition of abductive entailment to preferential abductive entailment and also extend the definition of filtering to preferential filtering. We then give conditions (sufficiency) when they are equivalent. We now define preferential abductive entailment and preferential filtering.

Definition 9 (Preferential abductive entailment) Suppose we have a partial ordering $\leq_e$ that encodes preferences between explanations. We say:

M is a model of $\langle T, Q, \leq_e \rangle$ if there exists an $\leq_e$-minimal explanation E of Q w.r.t. T such that M is a model of $T \cup E$.

$\langle T, Q, \leq_e \rangle \models_{abd} f$ if f is true in all models of $\langle T, Q, \leq_e \rangle$. □

Definition 10 Suppose we have a partial ordering $\leq_m$ that encodes preferences between models.

$Filter(T, Q, \leq_m)$ = The $\leq_m$-minimal models among the models of T that entail Q. □

For the equivalence of preferential abductive entailment and preferential filtering we need two additional conditions that relate the partial ordering $\leq_e$ between explanations and $\leq_m$ between models.

Condition C
If $E \leq_e E'$ then for any model M of $T \cup E$ and any model M' of $T \cup E'$ we have $M \leq_m M'$.

Condition D
If $M \leq_m M'$ then $M \cap Abd \leq_e M' \cap Abd$.

Definition 11 (Preferential-Filter-abducible) A theory T, observables Obs and abducibles Abd are said to be preferential filter abducible w.r.t. orderings $\leq_e$ and $\leq_m$ if for all possible observations Q; $Filter(T, Q, \leq_m)$ is equivalent to the set of models of $\langle T, Q, \leq_e \rangle$. □

Proposition 6 If T, Obs, Abd, $\leq_e$ and $\leq_m$ satisfy conditions A, B, C, and D then they are preferential filter-abducible. □

Proof:

(a) We show that all elements of $Filter(T, Q, \leq_m)$ are models of $\langle T, Q, \leq_e \rangle$.

Let M be an element of $Filter(T, Q, \leq_m)$. That means M is in $Filter(T, Q)$ and there does not exist an M' in $Filter(T, Q)$ such that $M' \leq_m M$. (1)

Since M is in $Filter(T, Q)$ it is clear that M is a model of $\langle T, Q \rangle$.
$\Rightarrow$ There exists E such that $T \cup E \models Q$, $T \cup E$ is consistent and M is a model of $T \cup E$.
We now need to show that *there does not exist* E' different from E such that $T \cup E' \models Q$, $T \cup E'$ is consistent, and $E' \leq_e E$.

Suppose such an E' exist.
Let M' be a model of $T \cup E'$.
$\Rightarrow$ M' is a model of $\langle T, Q \rangle$.
$\Rightarrow$ $M' \in Filter(T, Q)$.

From Condition C and the fact that M and M' are models of $T \cup E$ and $T \cup E'$ resp. and $E' \leq_e E$, we have $M' \leq_m M$. This contradicts (1).

Hence we have shown that *there does not exist* E' different from E such that $T \cup E' \models Q$, $T \cup E'$ is consistent, and $E' \leq_e E$. Hence M is a model of $\langle T, Q, \leq_e \rangle$.

(b) We will now show that all models of $\langle T, Q, \leq_e \rangle$ are also elements of $Filter(T, Q, \leq_m)$.

Let M be a model of $\langle T, Q, \leq_e \rangle$. That means there exists an E such that $T \cup E \models Q$, $T \cup E$ is consistent, M is a model of $T \cup E$ and there does not exist an E' such that $T \cup E' \models Q$, $T \cup E'$ is consistent and $E' \leq_e E$.

It is clear that M is a model of $\langle T, Q \rangle$, and hence is in $Filter(T, Q)$. We only need to show that there does not exist an M' such that M' is in $Filter(T, Q)$ and $M' \leq_m M$.

Let us assume to the contrary. I.e. there exists an M' such that M' is in $Filter(T, Q)$ and $M' \leq_m M$.

By Condition D, we have $M' \cap Abd \leq_e M \cap Abd$.

Since M is a model of $T \cup E$, from Lemma 1 we have $M \cap Abd = E$.

Now consider the fact that we have assumed M' to be in $Filter(T, Q)$. This means M' is a model of $\langle T, Q \rangle$. This means there exists an E^* such that M' is a model of $T \cup E^*$. But then using Lemma 1 we have $M' \cap Abd = E^*$.

Now we have $E^* \leq_e E$.

This contradicts our initial assumption that there does not exist an E' such that $T \cup E' \models Q$, $T \cup E'$ is consistent and $E' \leq_e E$.

Hence we can conclude that there does not exist an M' such that M' is in $Filter(T, Q)$ and $M' \leq_m M$. Therefore, M is in $Filter(T, Q, \leq_m)$. $\Box$

Example 6 Let us reconsider Example 3, where $T_2 = p \iff a \vee b$ with $Abd = \{a, b, \neg a, \neg b\}$ and $Obs = \{p, \neg p\}$. We showed that T_2, Abd and Obs satisfy conditions A and B.

Let us now define $\leq_e$ as $E_1 \subseteq E_2$ if $atoms(E_1) \subseteq atoms(E_2)$. The intuition behind the definition of $\leq_e$ is that we prefer explanations where less (based on the subset ordering) abducible atoms are true. The notion of minimal explanations commonly uses this ordering.

Now one simple ordering $\leq_m$ between models that will satisfy conditions C and D is as follows:
$M_1 \subseteq M_2$ if $atoms(M_1) \cap Abd \subseteq atoms(M_2) \cap Abd$. (Here, we are saying that we prefer models with less abducibles.)

It is easy to see that T_2, Obs, Abd, $\leq_e$, and $\leq_m$ satisfy conditions A, B, C and D. Thus they are preferential filter-abducible. Let us now verify this w.r.t. a particular observation $Q = \{p\}$.

The models of $\langle T_2, Q \rangle$ are $\{\{p, a, \neg b\}, \{p, b, \neg a\}, \{p, a, b\}\}$ which is same as $Filter(T_2, Q)$. Now using $\leq_m$ and $\leq_e$ we obtain that the models of $\langle T_2, Q, \leq_m \rangle$ are $\{\{p, a, \neg b\}, \{p, b, \neg a\}\}$ which is same as $Filter(T_2, Q, \leq_e)$. $\Box$

As mentioned earlier Condition A requires that if M is a model of a theory T then M should be the only model of the theory $T \cup (M \cap Abd)$. This is a strong condition, as in many cases $T \cup (M \cap Abd)$ may have multiple models. To take into account such cases, we weakened condition A to A'. But with preferential filtering and preferential abductive reasoning we also need to weaken condition D.

Condition D'
If $M \leq_m M'$ then for any E, and E' such that M is a model of $T \cup E$ and M' is a model of $T \cup E'$, $E \leq_e E'$.

Lemma 4 *If T, Obs, Abd, $\leq_e$ and $\leq_m$ satisfy conditions A, B, and D then they also satisfy Condition D'.* $\Box$

Proposition 7 If T, Obs, Abd, $\leq_e$ and $\leq_m$ satisfy conditions A', B, C, and D' then they are preferential filter-abducible. $\Box$

Proof:

(a) We show that all elements of $Filter(T, Q, \leq_m)$ are models of $\langle T, Q, \leq_e \rangle$.

The proof is exactly the same as the proof of the Proposition 6.

(b) We will now show that all models of $\langle T, Q, \leq_e \rangle$ are in $Filter(T, Q, \leq_m)$.

Let M be a model of $\langle T, Q, \leq_e \rangle$. That means there exists an E such that $T \cup E \models Q$, $T \cup E$ is consistent, M is a model of $T \cup E$ and there does not exist an E' such that $T \cup E' \models Q$, $T \cup E'$ is consistent and $E' \leq_e E$.

It is clear that M is a model of $\langle T, Q \rangle$, and hence is in $Filter(T, Q)$. We only need to show that there does not exist an M' such that M' is in $Filter(T, Q)$ and $M' \leq_m M$.

Let us assume to the contrary. I.e. there exists an M' such that M' is in $Filter(T, Q)$ and $M' \leq_m M$.

Now consider the fact that we have assumed M' to be in $Filter(T, Q)$. This means M' is a model of $\langle T, Q \rangle$. This means there exists an E^* such that M' is a model of $T \cup E^*$.

Since M is a model of $T \cup E$, M' is a model of $T \cup E*$ and $M' \leq_m M$, using Condition D' we have $E^* \leq_e E$.

This contradicts our initial assumption that there does not exist an E' such that $T \cup E' \models Q$, $T \cup E'$ is consistent and $E' \leq_e E$.

Hence we can conclude that there does not exist an M' such that M' is in $Filter(T, Q)$ and $M' \leq_m M$. Therefore, M is in $Filter(T, Q, \leq_m)$. □

7 Weak abductive reasoning vs filtering

Several instances of filtering used in the literature that define an intuitively meaningful entailment relation do not satisfy the conditions described earlier in this paper. In particular, when actions have non-deterministic effects (as in [Tur97]) filtering may still make intuitive sense, but our current definition of abductive reasoning is too strong to match the entailment defined through filtering. The following example illustrates our point.

Consider the disjunctive logic program: T

$a \vee b \leftarrow p$
$p \vee \neg p$

where $Abd = \{p, \neg p\}$, and $Obs_a = \{a, b\}$. Suppose we observe a. Using filtering we would be able to conclude – in this case, intuitively explain our observation by – p. (I.e., p will be true all models of $Filter(T, \{a\})$.) But the current definition of abductive reasoning is too strong to explain this observation by

p. (Note that the above theory will violate our condition $A'(iii)$.) This rigidity of abductive reasoning has been noticed earlier and several suggestions for weaker versions have been made; for example in [Gel90,Rei87,Sha97]. In this section we define a weaker notion of abductive reasoning and show that it is equivalent to filtering under less restrictive conditions than given in the earlier sections; in particular, we no longer need condition $A'(iii)$. As a result we can also weaken the sufficiency conditions in Propositions 3– 5. We now formally define *weak abductive entailment* and state theorems and propositions similar to the ones in the previous sections. (The proofs of these theorems and propositions are very similar to the earlier proofs and for brevity we omit them.)

Definition 12 (Weak abductive Entailment) Let T be a (possibly non-monotonic) theory with an entailment relation $\models$, and Q be an observation.

(i) M is a w-model (or weak-model) of $\langle T, Q\rangle$ if there exists a complete subset E of abducibles such that M is a model of $T \cup E$ and $M \models Q$.

(ii) For any formula f, $\langle T, Q\rangle \models_{wabd} f$ if f is true in all w-models of $\langle T, Q\rangle$. □

Definition 13 (Weak-filter-abducible) A theory T, a set Abd, and a set Obs are said to be *weak-filter-abducible* if for all possible observations $Q \in Obs$; $Filter(T, Q)$ is the set of w-models of $\langle T, Q\rangle$. □

Theorem 3 (Sufficiency) Let T be a theory, and Obs and Abd be observables. If T, Obs and Abd satisfy conditions $A'(i)$, $A'(ii)$ and B then they are weak-filter-abducible. □

Theorem 4 (Necessity) Let T be a theory, and Obs and Abd be observables and abducibles such that, for all $Q \in Obs$, $Filter(T, Q)$ is equivalent to the set of weak-models of $\langle T, Q\rangle$. Then T, Obs and Abd satisfy the conditions B, $A'(i), andA'(ii)$. □

Proposition 8 A propositional or first-order theory (possibly augmented with circumscription) T is weak-filter-abducible with respect to any abducibles Abd and observables Obs. □

Proposition 9 An extended logic program T is weak-filter-abducible with respect to abducibles Abd and observables Obs if
(i) for all $l, \neg l \in Abd$, $l \vee \neg l \leftarrow$ is in T, and
(ii) Abd is a splitting set for T. □

Proposition 10 A default theory T is weak-filter-abducible with respect to abducibles Abd and observables Obs if
(i) $\{\frac{:l}{l}, \frac{:\neg l}{\neg l} \ : \ l \in Abd\}$ is a subset of T, and

(ii) *Abd* is a splitting set [Tur96] for T. □

8 Conclusion and Future Work

In this paper we gave conditions on theories that guarantee the equivalence of particular kinds of abductive reasoning and filtering for assimilation of observations into theories. We also showed why these conditions are necessary. We gave some sufficiency conditions on theories expressed in several knowledge representation formalisms so as to guarantee the equivalence of filtering and abductive reasoning in those formalisms. Finally, we illustrated several examples from the knowledge representation literature where the theories that use filtering indeed satisfy the sufficiency conditions and thus can be shown to be encoding abductive reasoning.

One direction of future work is to explore more syntactic and easily verifiable sufficiency conditions for particular knowledge representation formalisms. It will be also interesting to examine additional formalizations in the literature where filtering is used and examine if indeed some form of abductive reasoning is encoded there.

Another important direction of future work would be to broaden the results in this paper, where we remove some of the restrictions imposed on the notion of abductive reasoning. (Note that the restrictions imposed are important in their own right as they lead us to show the relation between abductive reasoning and filtering in several formulations of reasoning about actions in the literature.) In particular, when the theory is a disjunctive logic program or a default theory, it would be interesting to broaden the notion of explanations, to allow incomplete subset of the abducibles. Such a notion of abductive reasoning in default theories is defined in [EGL97].

To do abductive reasoning using filtering in this case, the theory should be again such that its models enumerate the various possible explanations. Suppose $Abd = \{p, \neg p\}$. Now that we intend to allow explanations to be incomplete subset of the abducibles, the set of possible explanations will $\{\{p\}, \{\neg p\}, \{\}\}$. Since stable models are minimal sets, there does not exist a logic program whose models will be these three possible explanations. The minimality condition will eliminate the first two, in presence of the third. One way to overcome this would be to use a special fluent u_p (meaning *uncommitted about* p) to represent the third explanation. Then we can have disjunctive logic programming rules of the form:

$$p \ \vee \ \neg p \ \vee \ u\_p \leftarrow$$

or defaults of the form:

$$\frac{: p, \neg u\_p}{p}; \ \frac{: \neg p, \neg u\_p}{\neg p}; \ \frac{: p, \neg p}{u\_p}$$

In the sequel we plan to verify if indeed filtering using the above rules and/or defaults lead us to achieving a more general notion of abductive reasoning.

Acknowledgments

This work was done while the author was at the University of Texas at El Paso, and was revised while the author was visiting Linköping University.

References

[AKPT91] J. Allen, H. Kautz, R. Pelavin, and J. Tenenberg. *Reasoning about plans.* Morgan Kaufmann, 1991.

[BB95] C. Boutilier and V. Becher. Abduction as belief revision. *Artificial Intelligence*, 77:43–94, 1995.

[BG94] C. Baral and M. Gelfond. Logic programming and knowledge representation. *Journal of Logic Programming*, 19,20:73–148, 1994.

[BG97] C. Baral and M. Gelfond. Reasoning about effects of concurrent actions. *Journal of Logic Programming*, 31(1-3):85–117, May 1997.

[BGP98a] C. Baral, A. Gabaldon, and A. Provetti. Formalizing Narratives using nested circumscription. *Artificial Inteligence*, 104(1-2):107–164, Sept 1998.

[BGP98b] C. Baral, A. Gabaldon, and A. Provetti. Value minimization in circumscription. *Artificial Inteligence*, 102(2):163–186, July 1998.

[Bou96] C. Boutilier. Abduction to plausible causes: an event based model of belief update. *Artificial Intelligence*, 83:143–166, 1996.

[CDT91] L. Console, D. Dupre, and P. Torasso. On the relationship between abduction and deduction. *Journal of Logic and Computation*, 2(5), 1991.

[CE92] J. Crawford and D. Etherington. Formalizing reasoning about change: a qualitative reasoning approach. In *Proc. of AAAI*, pages 577–583, 1992.

[Cla78] K. Clark. Negation as failure. In Herve Gallaire and J. Minker, editors, *Logic and Data Bases*, pages 293–322. Plenum Press, New York, 1978.

[DD97] M. Denecker and D. De Schreye. SLDNFA: an abductive procedure for abductive logic programs. *Journal of Logic Programming*, 34(2):111–167, 1997.

[DDS93] M. Denecker and D. De Schreye. Representing incomplete knowledge in abductive logic programming. In *Proc. of ILPS 93, Vancouver*, pages 147–164, 1993.

[DGKK98] P. Doherty, J. Gustafsson, L. Karlsson, and J. Kvarnstrom. TAL: Temporal action logics – language specification and tutorial. *Electronic Transactions on Artificial Intelligence*, 3(15):(http://www.ep.liu.se/ej/etai/1998/015), 1998.

[DMB92] M. Denecker, L. Missiaen, and M. Bruynooghe. Temporal reasoning with abductive event calculus. In *Proc. of ECAI92*, 1992.

[DNK97] Y. Dimopoulos, B. Nebel, and J. Koehler. Encoding planning problems in non-monotonic logic programs. In *Proc. of European conference on Planning*, pages 169–181, 1997.

[Doh94a] P. Doherty. Notes on PMON circumscription. Technical Report LITH-IDA-94-43, Department of Computer and Information science, Linkoping University, Sweden, 1994.

[Doh94b] P. Doherty. Reasoning about action and change using occlusion. In *Proc. of ECAI 94*, pages 401–405, 1994.

[Doh97] P. Doherty. PMON+ : A fluent logic for action and change – formal specification. *Electronic Transactions on Artificial Intelligence*, 2(20):(http://www.ep.liu.se/ea/cis/1997/020), 1997.

[Dun93] P. Dung. Representing actions in logic programming and its application in database updates. In D. S. Warren, editor, *Proc. of ICLP-93*, pages 222–238, 1993.

[EGL97] T. Eiter, G. Gottlob, and N. Leone. Semantics and complexity of abduction from default theories. *Artificial Intelligence*, 90:177–223, 1997.

[EK88] K. Eshghi and R. Kowalski. Abduction through deduction, 1988. unpublished paper.

[EK89] K. Eshghi and R. Kowalski. Abduction compared with negation as failure. In Giorgio Levi and Maurizio Martelli, editors, *Logic Programming: Proc. of the Sixth Int'l Conf.*, pages 234–255, 1989.

[ELM+98] T. Eiter, N. Leone, C. Mateis, G. Pfeifer, and F. Scarcello. A deductive system for nonmonotonic reasoning. In *Proc. of KR 98*, pages 406–417, 1998.

[Esh88] K. Eshghi. Abductive planning with event calculus. In *Proc. of the Fith ICLP.*, pages 562–579, 1988.

[GD96] J. Gustafsson and P. Doherty. Embracing occlusion in specifying the indirect effects of actions. In *Proc. of KR 94*, 1996.

[Gef99] H. Geffner. Filtering and conditioning, 1999. personal communication.

[Gel90] M. Gelfond. Belief sets, deductive databases and explanation based reasoning. Technical report, University of Texas at El Paso, 1990.

[GL91] M. Gelfond and V. Lifschitz. Classical negation in logic programs and disjunctive databases. *New Generation Computing*, pages 365–387, 1991.

[GL93] M. Gelfond and V. Lifschitz. Representing actions and change by logic programs. *Journal of Logic Programming*, 17(2,3,4):301–323, 1993.

[HSAM90] J. Hobbs, M. Stickel, D. Appelt, and P. Martin. Interpretation as abductions. Technical report, SRI, Menlo Park, California, 1990.

[Ino91] K. Inoue. Extended logic programs with default assumptions. In *Proc. of ICLP91*, 1991.

[KKT93] A. Kakas, R. Kowalski, and F. Toni. Abductive logic programming. *Journal of Logic and Computation*, 2(6):719–771, 1993.

[KL94] G. Kartha and V. Lifschitz. Actions with indirect effects: Preliminary report. In *KR 94*, pages 341–350, 1994.

[KM90] A. Kakas and P. Mancarella. Generalized stable models: a semantics for abduction. In *Proc. of ECAI-90*, pages 385–391, 1990.

[KM97] A. Kakas and C. Mourlas. ACLP: flexible soultions to complex problems (http://www.cs.ucy.ac.cy/aclp/). In *Proc. of Logic programming and nonmonotonic reasoning*, 1997.

[Kon92] K. Konolige. Abduction versus closure in causal theories. *Artificial Intelligence*, 53:255–272, 1992.

[Lif95] V. Lifschitz. Nested abnormality theories. *Artificial Intelligence*, 74:351–365, 1995.

[Lif99] V. Lifschitz. Success of default logic, 1999. http://www.cs.utexas.edu/users/vl/mypapers/deflog.ps.

[Lin95] F. Lin. Embracing causality in specifying the indirect effects of actions. In *Proc. of IJCAI 95*, pages 1985–1993, 95.

[Llo87] J. Lloyd. *Foundations of logic programming.* 1987. Second, extended edition.

[LT94] V. Lifschitz and H. Turner. Splitting a logic program. In Pascal Van Hentenryck, editor, *Proc. of the Eleventh Int'l Conf. on Logic Programming*, pages 23–38, 1994.

[MD95] M. Missiaen, L.and Bruynooghe and M. Denecker. CHICA: A planning system based on event calculus. *Lournal of Logic and Computation*, 5(5):579–602, 1995.

[MT93] W. Marek and M. Truszczyński. *Nonmonotonic Logic: Context dependent reasoning.* Springer Verlag, 1993.

[MT95] N. McCain and H. Turner. A causal theory of ramifications and qualifications. In *Proc. of IJCAI 95*, pages 1978–1984, 95.

[MT97] N. McCain and H. Turner. Causal theories of action and change. In *Proc. of AAAI*, pages 460–465, 97.

[NS96] I. Niemela and P. Simons. Efficient implementation of the well-founded and stable model semantics. In *Proc. Joint international conference and symposium on Logic programming*, pages 289–303, 1996.

[Peaa] J. Pearl. Causality (forthcoming book). http://bayes.cs.ucla.edu/jp_home.html.

[Peab] J. Pearl. Reasoning with cause and effect. In *Proc. of IJCAI 99*, pages 1437-1449, 1999.

[Pei58] C. Peirce. *Collected papers of Charles Sanders Peirce (Volume 1-8, Edited by Hartshorn et al.).* Harvard University Press, 1931-1958.

[Pei92] C. Peirce. Reasoning and the logic of things. Harvard University Press, 1992.

[PGA87] D. Poole, R. Goebel, and R. Aleliunas. Theorist: a logical reasonig system for defaults and diagnosis. In N. Cercone and G. McCalla, editors, *The Knowledge Frontier: Essays in the Representation of Knowledge*, pages 331–352. Springer-Verlag, New York, 1987.

[PMG98] D. Poole, A. Mackworth, and R. Goebel. *Computational Intelligence.* Oxford University Press, 1998.

[Poo88a] D. Poole. A logical framework for derfault reasoning. *Artificial Intelligence*, 36(1):27–48, 1988.

[Poo88b] D. Poole. Representing knowledge for logic based diagnosis. In *Proc. of International conference on Fifth generation computer systems*, pages 1282–1290, 1988.

[Poo89] D. Poole. Explanation and prediction: an architecture for default and abductive reasoning. *Computational Intelligence*, 5(1):97–110, 1989.

[Reg83] R. Reggia. Diagnostic expert system based on a set covering model. *International journal of man machine studies*, 19(5):437–460, 1983.

[Rei80] R. Reiter. A logic for default reasoning. *Artificial Intelligence*, 13(1,2):81–132, 1980.

[Rei87] R. Reiter. A theory of diagnosis from first principles. *Artificial Intelligence*, 32(1):57–95, 1987.

[San89] E. Sandewall. Filter preferential entailment for the logic of action in almost continuous worlds. In N. S. Sridharan, editor, *Proc. of the 11th Int'l Joint Artificial Intelligence*, pages 894–899, 1989.

[San94] E. Sandewall. *Features and Fluents: The representation of knowledge about dynamical systems.* Oxford University Press, 1994.

[San98] E. Sandewall. Cognitive robotics logic and its metatheory: Features and fluents revisited. *Electronic Transactions on Artificial Intelligence*, 3(17):(http://www.ep.liu.se/ej/etai/1998/017), 1998.

[Sha89] M. Shanahan. Prediction is deduction but explanation is abduction. In *Proc. of IJCAI-89*, pages 1055–1060, 1989.

[Sha93] M. Shanahan. Explanation in the situation calculus. In *Proc. of IJCAI-93*, pages 160–165, 1993.

[Sha97] M. Shanahan. *Solving the frame problem: A mathematical investigation of the commonsense law of inertia.* MIT press, 1997.

[Tur95] H. Turner. Representing actions in default logic: A situation calculus approach. In *Proceedings of the Symposium in honor of Michael Gelfond's 50th birthday (also in Common Sense 96)*, 1995.

[Tur96] H. Turner. Splitting a default theory. In *Proceedings of AAAI*, 1996.

[Tur97] H. Turner. Representing actions in logic programs and default theories. *Journal of Logic Programming*, 31(1-3):245–298, May 1997.