Ordered Diagnosis

Davy Van Nieuwenborgh* and Dirk Vermeir**

Dept. of Computer Science
Vrije Universiteit Brussel, VUB
{dvnieuwe,dvermeir}@vub.ac.be

Abstract. We propose to regard a diagnostic system as an ordered logic theory, i.e. a partially ordered set of clauses where smaller rules carry more preference. This view leads to a hierarchy of the form *observations* $<$ *system description* $<$ *fault model*, between the various knowledge sources. It turns out that the semantics for ordered logic programming nicely fits this intuition: if the observations contradict the normal system behavior, then the semantics will provide an explanation from the fault rules. The above model can be refined, without adding additional machinery, to support e.g. problems where there is a clear preference among possible explanations or where the system model itself has a complex structure. Interestingly, these extensions do not increase the complexity of the relevance or necessity decision problems. Finally, the mapping to ordered logic programs also provides a convenient implementation vehicle.

1 Introduction

Diagnostic reasoning involves finding explanations, i.e. sets of causes, that explain certain observations. The topic has received a great deal of attention over the years, with important applications, e.g. in medicine [29].

There are two main approaches to the theory of model-based diagnosis[1].

In *consistency-based* diagnosis [25], one uses a model of the normal structure and behavior of the system under consideration. This model is typically formulated in terms of components and their relationship where, roughly, a proposition p asserts that a component p is functioning correctly. Rules such as $e \leftarrow p, q$ can be used to assert that the effect e will occur if both p and q are in working order. An observation that does not conform to the normal predicted behavior then leads to an inconsistency. E.g. the observation $\neg e$ is inconsistent with the assumption that p and q are both true (functioning correctly). Finding an explanation involves removing this inconsistency by withdrawing some assumptions on the correct functioning of some components. In the above example, both $\{\neg p\}$ ("p is faulty") and both $\{\neg q\}$ ("q is faulty") are acceptable explanations

* Supported by the FWO

** This work was partially funded by the Information Society Technologies programme of the European Commission, Future and Emerging Technologies under the IST-2001-37004 WASP project

[1] Other approaches that have been proposed in the literature include e.g. the set-covering theory of diagnosis [24] or hypothetico-deductive diagnosis [20].

of the observation $\neg e$, as is $\{\neg p, \neg q\}$. The approach has been applied in several areas such as fault finding in electronic circuits [5].

If an explicit *fault model* of the system is available, *abductive* reasoning can be used to perform diagnosis [21, 3]. Rules in the fault model specify cause-effect relationships that govern abnormal behavior. E.g. a rule such as $fever \leftarrow flu$ asserts that the fever might be caused by flu. An explanation then consists of a set of causes that, when combined with the fault model, are sufficient to entail the observations.

In this paper, we formalize both of the above approaches in terms of ordered logic programs [33], i.e. partially ordered sets of rules, by providing a translation that, given a diagnostic problem D, constructs an ordered program $L(D)$ such that the explanations for D correspond exactly to the preferred answer sets of $L(D)$. Roughly, this is achieved by structuring $L(D)$ as shown below where rules in lower levels are smaller (more

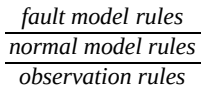

Fig. 1.

preferred) than rules in higher levels.

Intuitively, when faced with inconsistencies between the rules making up an ordered program, the semantics of [33] extends the usual answer set semantics of [19] by producing preferred answer sets that satisfy as many rules as possible, giving priority to the satisfaction of more preferred rules, possibly at the expense of defeating less preferred ones.

In $L(D)$, an observation o is typically represented by a most preferred rule of the form $o \leftarrow \mathit{not}(o)$, which, although it cannot be used to justify o, can only be satisfied by answer sets containing o. The overall effect is then that a preferred answer set will explain all observations, and satisfy as much as possible of the normal behavior rules, with minimal recourse to the fault model, in order to motivate the observations. An explanation can then be retrieved by selecting the "cause" literals from the answer set.

Not all explanations are equally convincing. E.g. when troubleshooting a circuit, an explanation $\{\neg c_1, \neg c_2\}$, asserting that both of the components c_1 and c_2 are broken, would be unlikely if the more parsimonious $\{\neg c_1\}$ were also an explanation. Thus it has been argued [25, 21, 8, 7] that minimal explanations are to be preferred, where one usually considers the subset ordering, or, possibly, the cardinality ordering, among alternative explanations.

We extend the above approach by allowing an arbitrary partial order structure on the set of causes that may occur in explanations. This partial order may reflect e.g. the likelihood that a cause actually occurs, the expense associated with verifying/removing the cause or any other preference criterion. One then prefers explanations that are minimal w.r.t. a partial order induced by the one on causes. The approach depicted in Figure 1

can be adapted such that $L(D)$ reflects the order on the causes and, moreover, the preferred answer sets of $L(D)$ correspond with preferred explanations.

We also consider diagnostic problems where the system model itself has a preference structure, which introduces another natural preference relation on explanations. E.g. laws are often ordered w.r.t. legal precedence. An abductive problem that uses such a system model would prefer explanations that are based on laws with higher precedence. We show that such problems can also be mapped to ordered programs where the preferred answer sets correspond to preferred explanations.

The remainder of this paper is organized as follows: Section 2 presents a brief overview of consistency-based and abductive diagnosis. In Section 3, these approaches are generalized to allow for a partial order relation on the possible causes, which induces a preference relation on the explanations. We present an algorithm to transform such an ordered diagnostic system into an ordered extended logic program. Diagnostic problems with ordered system descriptions are considered in Section 4. Some complexity results are presented in Section 5. Section 6 discusses the relationships with other approaches. Conclusions and directions for further research are stated in Section 7. All proofs can be found in the appendix.

2 Classical Diagnosis

We use simple logic programs, i.e. programs containing only classical negation, to describe the system behavior of a diagnostic problem.

We use the following basic definitions and notation. A *literal* is an *atom* a or a negated atom $\neg a$. For a set of literals X we use $\neg X$ to denote $\{\neg p \mid p \in X\}$, where $\neg(\neg a) \equiv a$. Also, X^+ denotes the positive part of X, i.e. $X^+ = \{a \in X \mid a \text{ is an atom}\}$. The *Herbrand base* of X, denoted $\mathcal{B}_X$, contains all atoms appearing in X, i.e. $\mathcal{B}_X = (X \cup \neg X)^+$. A set I of literals is *consistent* if $I \cap \neg I = \emptyset$. Furthermore, I is *total* w.r.t. a set of atoms J iff $J \subseteq I \cup \neg I$.

A *rule* is of the form $a \leftarrow \beta$, where $\{a\} \cup \beta$ is a finite set of literals. A countable set of rules P is called a *simple program*. The *Herbrand base* $\mathcal{B}_P$ of P contains all atoms appearing in P. For a simple program P we use $P^\star$ to denote the unique minimal[30] model of the positive logic program consisting of the rules in P, where negative literals $\neg a$ are considered as fresh atoms. This operator is monotonic, i.e. if $R \subseteq Q$ then $R^\star \subseteq Q^\star$. Then, P is *consistent* iff $P^\star$ is consistent. A consistent set of literals $S \subseteq \mathcal{B}_P \cup \neg\mathcal{B}_P$ is an *answer set* of P iff $P^\star = S$.

First recall the general framework of consistency-based diagnoses [25].

Definition 1. *A **C-diagnostic system** is a triple $S = (T, C, O)$, where T is a simple program describing the normal system behavior, C is the set of possible **causes**, i.e. negated atoms, each representing a malfunctioning component, and O is a set of **observation** literals.*

*A **hypothesis** for S is a set of causes. A hypothesis $H \subseteq C$ is a **C-explanation** or **C-diagnosis** for S iff $T \cup \{x \leftarrow \mid x \in H \cup \neg(C \setminus H) \cup O\}$ is consistent.*

Thus rules in T are of the form $a \leftarrow c_1, c_2$ with c_1 and c_2 corresponding to components that are in working order. Causes such as $\neg c_1$ express that c_1 is faulty. For an explana-

tion H, components not in H are assumed to be working correctly, as witnessed by the $\neg(C \setminus H)$ construction in the definition.

Example 1. Consider the C-diagnostic system $S = (T, C, O)$, with $T = \{light \leftarrow power, bulb\}$, $C = \{\neg power, \neg bulb\}$ and $O = \{\neg light\}$. The system model asserts that if power is available and the bulb is OK, then light should be observed. The observation $\neg light$ has three explanations: $H_1 = \{\neg power\}$, $H_2 = \{\neg bulb\}$ and $H_3 = \{\neg power, \neg bulb\}$.

The abductive framework captures diagnostic reasoning using a system model that describes abnormal system behavior [4, 22, 21, 3, 17, 23, 9]. We use the definition from [28], which is based on the belief set semantics from [13].

Definition 2. *An **A-diagnostic system** is a triple $S = (T, C, O)$, where T is a simple program describing the abnormal behavior of the system, C is a set of literals representing the possible causes in the system, and O is a set of literals containing the observations.*

*Any subset $H \subseteq C$ is called a **hypothesis** for S. A hypothesis H is an **A-explanation** or **A-diagnosis** for S iff there is answer set Q for $T \cup \{x \leftarrow\mid x \in H\}$ such that $O \subseteq Q$ and $H = Q \cap C$.*

Thus, while an explanation in a C-diagnostic system prevents the derivation of $\neg o$ for some observable o, in an abductive system, the explanation is used to actually derive o.

Example 2. Suppose the screen of your computer is working unreliably, i.e. $O = \{unreliable\_screen\}$, which may be caused by any of the causes in $C = \{broken\_lcd, broken\_adapter, cable\_problem, broken\_cooler\}$. The fault model T describes how these causes can affect the system.

$$\begin{aligned} heating &\leftarrow broken\_cooler \\ no\_screen &\leftarrow broken\_lcd \\ no\_screen &\leftarrow cable\_problem \\ no\_screen &\leftarrow broken\_adapter \\ unreliable\_adapter &\leftarrow heating \\ unreliable\_screen &\leftarrow cable\_problem \\ unreliable\_screen &\leftarrow broken\_adapter \\ unreliable\_screen &\leftarrow unreliable\_adapter \end{aligned}$$

E.g., a broken cooler causes a heating problem that makes the graphics adapter behave unreliably which, in turn, affects the screen performance. Alternatively, a cable problem or a broken LCD may also cause screen problems.

Some of the A-explanations for $unreliable\_screen$ are $H_1 = \{broken\_cooler\}$, $H_2 = \{cable\_problem\}$, $H_3 = \{broken\_adapter\}$, $H_4 = \{cable\_problem, broken\_cooler\}$ and $H_5 = \{broken\_cooler, broken\_lcd\}$.

Note that H_4 in Example 2 is likely to be redundant since $H_4 = H_2 \cup H_1$, i.e. the observations can be explained using a subset of the causes in H_4.

In general, we assume that sets of causes may carry a preference order. Preferred explanations then correspond to explanations that are minimal with respect to this order.

Definition 3. *Let $S = (T, C, O)$ be a diagnostic system, with $\leq$ a partial order relation on 2^C. An explanation of S is called $\leq$-**preferred** iff it is minimal w.r.t. $\leq$.*

Often, $\leq$ is taken as the subset order although cardinality order is also used (e.g. if all components in a circuit are equally likely to fail, cardinality-preferred explanations are more likely). In Example 1, both H_1 and H_2 are $\subseteq$-preferred explanations, which are also cardinality preferred. In Example 2, H_1, H_2 and H_3 are $\subseteq$-preferred explanations.

Definition 2, which follows [15, 16], differs from the definition in [9] which does not require the $H = Q \cap C$ condition. This influences the set of explanations, as illustrated in the following example.

Example 3. Consider the A-diagnostic system $S = (T, C, O)$ with T containing $air\_in\_fuel\_pump \leftarrow out\_of\_diesel$ and $car\_does\_not\_start \leftarrow out\_of\_diesel$, $C = \{out\_of\_diesel, air\_in\_fuel\_pump\}$ and $O = \{car\_does\_not\_start\}$.

The semantics of [9] yields $\{out\_of\_diesel\}$ as a $\subseteq$-preferred explanation while $\{out\_of\_diesel, air\_in\_fuel\_pump\}$ is $\subseteq$-preferred according to Definition 2. Thus, [9] returns the "root cause" of the problem, leaving out side effects. On the other hand, Definition 2's solution includes the side effects, which is useful in this example, as just refueling diesel will not completely fix the problem: one also must ventilate the fuel pump.

One may wonder whether a C-diagnostic system could be easily converted to an equivalent abductive one by simply replacing "normal behavior" rules $a \leftarrow \alpha$ by "fault model" rules $\neg a \leftarrow \neg\beta$ where β is a minimal set of literals such that $\beta \cap \alpha \neq \emptyset$ for each a-rule $a \leftarrow \alpha$. This may, however, not produce all explanations warranted by the consistency-based system. E.g. if the C-system contains just $a \leftarrow b, c$, $c \leftarrow d$ and $e \leftarrow d$, the above construction would yield the "abductive rules" $\neg a \leftarrow \neg b$, $\neg a \leftarrow \neg c$, $\neg c \leftarrow \neg d$ and $\neg e \leftarrow \neg d$. For the observations $\{\neg a, \neg e\}$ and set of causes $\{\neg b, \neg c, \neg d\}$, the original consistency based system yields both $\{\neg b, \neg d\}$ and $\{\neg c, \neg d\}$ as $\subseteq$-preferred explanations, while the abductive variant only supports $\{\neg c, \neg d\}$.

3 Ordered Diagnosis

Often, causes are themselves partially ordered according to some preference. E.g. in Example 1 it may be much more likely that the bulb is broken than that the power is off, or the reverse (depending on where one is located).

In an ordered diagnostic system, explanations are ordered according to the partial order induced by the order on causes.

Definition 4. *An **ordered diagnostic system** is a tuple $D = (S, <)$, where S is either a C- or A-diagnostic system $S = (T, C, O)$ and $<$ is a strict[2] partial order relation on the elements in C. When S is a C-diagnostic system (A-diagnostic system), we call D a C-ordered diagnostic system (A-ordered diagnostic system respectively). The explanations of D are the explanations of S.*

For explanations H_1 and H_2, $H_1 \sqsubseteq H_2$ iff $\forall c \in H_1 \setminus H_2 \cdot \exists c' \in H_2 \setminus H_1 \cdot c < c'$.

[2] A strict partial order $<$ on a set X is a binary relation on X that is antisymmetric, anti-reflexive and transitive.

Intuitively, H_1 is preferred over H_2 if any cause c_1 from H_1 but not in H_2 is "covered" by a "less preferred" cause $c_2 > c_1$ in H_2 but not in H_1. It can be shown that $\sqsubseteq$ is a partial order, provided that the inverse of $<$ is well-founded, see Lemma 1 in [32].

Example 4. If $\neg bulb < \neg power.$ in Example 1, then $H_2 \sqsubseteq H_1$ because $H_2 \setminus H_1 = \{\neg bulb\}$, $H_1 \setminus H_2 = \{\neg power\}$ and $\neg bulb < \neg power$. Consequently, H_2 is $\sqsubseteq$-preferred.

Example 5. Extend Example 2 to an ordered diagnostic system $D = (S, <)$ with $cable\_problem < broken\_adapter$, $cable\_problem < broken\_lcd$, $broken\_cooler < broken\_adapter$, $broken\_cooler < broken\_lcd$. It follows that the explanations $H_1 = \{broken\_cooler\}$ and $H_2 = \{cable\_problem\}$ are both $\sqsubseteq$-preferred.

Each $\sqsubseteq$-preferred explanation is also $\subseteq$-preferred.

Theorem 1. *Let $D = (S, <)$ be an ordered diagnostic system. Every $\sqsubseteq$-preferred explanation H of D is a $\subseteq$-preferred explanation of S.*

If the order on the causes is empty, $\sqsubseteq$-preference reduces to $\subseteq$-preference.

Theorem 2. *Let S be a diagnostic system, either C or A. Then, all $\subseteq$-preferred explanations of S coincide with the preferred explanations of $D = (S, \emptyset)$.*

We will show that the $\sqsubseteq$-preferred explanations (and, by Theorem 2, also the $\subseteq$-preferred explanations) of an ordered diagnostic system D can be retrieved from the preferred answer sets of an extended ordered logic program (EOLP) $L(D)$ that can be constructed from D.

First, we review the definition and semantics of EOLPs [31].

An *extended literal* is a literal or a *naf-literal* of the form *not*(l) where l is a literal. The latter form denotes negation as failure. We use l^- to denote the literal underlying the extended literal l. An extended literal l is true w.r.t. an interpretation I, denoted $I \models l$ if $l \in I$ in case l is ordinary, or $I \not\models a$ if $l =$ *not*(a) for some ordinary literal a. As usual, $I \models X$ for some set of (extended) literals l iff $\forall l \in X \cdot I \models l$.

An *extended rule* is a rule of the form $\alpha \leftarrow \beta$ where $\alpha \cup \beta$ is a finite set of extended literals and $|\alpha| \leq 1$. An extended rule $r = \alpha \leftarrow \beta$ is *satisfied* by I, denoted $I \models r$, if $I \models \alpha$, $\alpha \neq \emptyset$, whenever $I \models \beta$, i.e. if r is *applicable* ($I \models \beta$), then it must be *applied* ($I \models \alpha \cup \beta$).

A countable set of extended rules is called an *extended logic program* (ELP). For an ELP P and an interpretation I we use $P_I \subseteq P$ to denote the *reduct* of P w.r.t. I, i.e. $P_I = \{r \in P \mid I \models r\}$. We also define the *GL-reduct* for P w.r.t. I, denoted P^I, as the program consisting of those rules $\alpha \setminus$ *not*$(\alpha^-) \leftarrow (\beta \setminus$ *not*$(\beta^-))$ where $\alpha \leftarrow \beta$ is in P, $I \models$ *not*(β^-) and $I \models \alpha^-$. Note that all rules in P^I are free from negation as failure, i.e. P^I is a simple program. An interpretation I is then an *answer set* of P iff I is an answer set of the reduct P^I. An extended rule $r = \alpha \leftarrow \beta$ is *defeated* w.r.t. P and I iff there exists an applied *competing rule* $r' = \alpha' \leftarrow \beta'$ such that $\{\alpha, \alpha'\}$ is inconsistent. An *extended answer set* for P is any interpretation I such that I is an answer set of P_I and each unsatisfied rule in $P \setminus P_I$ is defeated.

An *extended ordered logic program* (EOLP) is a pair $(R, <)$ where R is an ELP and $<$ is a well-founded strict partial order on the rules in R. Intuitively, $r_1 < r_2$ indicates

that r_1 is more preferred than r_2. In the examples we will often represent the order implicitly using the format

$$\frac{\cdots}{\frac{R_2}{\frac{R_1}{R_0}}}$$

where each R_i, $i \geq 0$, represents a set of rules, indicating that all rules below a line are more preferred than any of the rules above the line, i.e. $\forall i \geq 0 \cdot \forall r_i \in R_i, r_{i+1} \in R_{i+1} \cdot r_i < r_{i+1}$ or $\forall i \geq 0 \cdot R_i < R_{i+1}$ for short.

Let $P = \langle R, < \rangle$ be an EOLP. For subsets R_1 and R_2 of R we define $R_1 \preceq R_2$ iff $\forall r_2 \in R_2 \setminus R_1 \cdot \exists r_1 \in R_1 \setminus R_2 \cdot r_1 < r_2$. We write $R_1 \prec R_2$ just when $R_1 \preceq R_2$ and $R_1 \neq R_2$. For M_1, M_2 extended answer sets of R, we define $M_1 \preceq M_2$ iff $R_{M_1} \preceq R_{M_2}$. As usual, $M_1 \prec M_2$ iff $M_1 \preceq M_2$ and $M_1 \neq M_2$. An *answer set* for an EOLP P is any extended answer set of R. An answer set for P is called *preferred* if it is minimal w.r.t. $\preceq$. An answer set is called *proper* if it satisfies all minimal (according to $<$) rules in R.

Let $D = (S, <)$ with $S = (T, C, O)$ be an ordered diagnostic system. We construct an EOLP $L(D)$ which is such that the proper preferred answer sets of $L(D)$ represent the $\sqsubseteq$-preferred explanations of D, i.e. for any proper preferred answer set M of $L(D)$, $M \cap C$ is a preferred explanation and the other way around.

The construction of $L(D)$ follows the intuition sketched in Section 1.

- The bottom component R_b of $L(D)$, whose rules will always be satisfied, consists of the system description T and a set of "constraint" rules R_o that enforce the observations, without providing a justification for them. If D is an A-system, each observation o should be derived from an explanation while for a C-system, it suffices to be consistent with O, i.e. to prevent the derivation of $\neg o$. Therefore, constraint rules for A-systems will have the form $o \leftarrow \mathit{not}(o)$, $o \in O$, while for C-systems, rules of the form $o \leftarrow \neg o$ will be used.
- On top of the bottom component, we put a component R_n with rules that simulate the normal behavior of the system. To this end, R_n contains, for each cause $c \in C$, a rule r_c asserting that this cause is not valid. For an A-system, this can be achieved by defining r_c as $\mathit{not}(c) \leftarrow$ thus ensuring that the semantics will prefer answer sets that maximize false causes. For a C-system, Definition 1 demands that the negation of any cause not in the explanation holds, hence r_c will be of the form $\neg c \leftarrow$. To take into account the preference relation between causes, we order the rules in R_n such that the EOLP semantics, when confronted with the necessity to defeat either r_c or $r_{c'}$, it will defeat r_c if $c < c'$. Thus, it suffices to have $r_{c'} < r_c$, i.e. the order on R_n is the reverse of the order on C.
- The topmost component $R_a > R_n$ introduces the possibility of abnormal behavior. For each $c \in C$, R_a contains a rule $c \leftarrow$ that provides a justification, if necessary, for c. Note that all rule in R_a have a stronger competitor in R_n.

Intuitively, if no causes are necessary to explain the observations, any proper preferred answer set will satisfy all rules in $R_b \cup R_n$, defeating all rules in R_a. If, however, the

observations cannot be explained without assuming some causes, the semantics will, in order to satisfy the rule in R_o, call upon rules in R_a to introduce them.

The following definition formalizes the above construction.

Definition 5. *Let $D = (S, <)$ be an ordered diagnostic system, with $S = (T, C, O)$ either a C- or A-diagnostic system. The EOLP version of D, denoted $L(D)$, is defined by $L(D) = \langle R_a \cup R_n \cup T \cup R_o, (R_o \cup T) < R_n^< < R_a \rangle$, where $R_a = \{c \leftarrow | \ c \in C\}$, $R_n = \{\phi(c) \leftarrow | \ c \in C\}$ and $R_o = \{o \leftarrow \phi(o) \mid o \in O\}$, with $\phi(l) = \neg l$ if S is a C-diagnostic system and $\phi(l) = \mathit{not}(l)$ if S is an A-diagnostic system. Furthermore, $R_n^<$ stands for $\phi(c_1) \leftarrow < \phi(c_2) \leftarrow$ with $c_1, c_2 \in C$ iff $c_2 < c_1$.*

Example 6. The program corresponding to the abductive ordered diagnostic system from Example 5 is shown below.

$$
\begin{array}{rl}
broken\_adapter & \leftarrow \\
broken\_lcd & \leftarrow \\
cable\_problem & \leftarrow \\
broken\_cooler & \leftarrow \\
\hline
\mathit{not}(cable\_problem) & \leftarrow \\
\mathit{not}(broken\_cooler) & \leftarrow \\
\hline
\mathit{not}(broken\_adapter) & \leftarrow \\
\mathit{not}(broken\_lcd) & \leftarrow \\
\hline
heating & \leftarrow broken\_cooler \\
no\_screen & \leftarrow broken\_adapter \\
no\_screen & \leftarrow broken\_lcd \\
no\_screen & \leftarrow cable\_problem \\
unreliable\_screen & \leftarrow broken\_adapter \\
unreliable\_screen & \leftarrow unreliable\_adapter \\
unreliable\_adapter & \leftarrow cable\_problem \\
unreliable\_adapter & \leftarrow heating \\
unreliable\_screen & \leftarrow \mathit{not}(unreliable\_screen)
\end{array}
$$

This program has two proper preferred answer sets: $N_1 = \{broken\_cooler, heating, unreliable\_adapter, unreliable\_screen\}$, corresponding with the preferred explanation $H_3 = N_1 \cap C$, and $N_2 = \{cable\_problem, unreliable\_adapter\}$, corresponding with the preferred explanation $H_2 = N_2 \cap C$.

Example 7. The program corresponding with the ordered C-diagnostic system of Example 4 is shown below.

$$
\begin{array}{c}
\neg power \leftarrow \quad \neg bulb \leftarrow \\
\hline
bulb \leftarrow \\
\hline
power \leftarrow \\
\hline
light \leftarrow power, bulb \\
\neg light \leftarrow light
\end{array}
$$

This program has only one proper preferred answer set $N = \{power, \neg bulb\}$, corresponding to the single preferred explanation $H_2 = N \cap C$.

In general, we have the following correspondence.

Theorem 3. *Let $D = (S, <)$ be an ordered diagnostic system with $S = (T, C, O)$ either a C- or A-diagnostic system. Then, H is a preferred explanation for D iff there is a proper preferred answer set N of $L(D)$ such that $H = N \cap C$.*

We illustrate the usefulness of the approach with a (simplified) example from software configuration management.

The goal of the installation (of a Linux system) is the availability of a set of packages. These will be considered as observables in an abductive diagnostic system where the system model contains rules and constraints representing inter-package dependencies and incompatibilities. Causes correspond to installation instructions for particular (versions of) packages.

Consequently, a preferred explanation will provide an "optimal" list of detailed install instructions that are necessary to achieve the objective.

Example 8. In the example, the goal is to have KDE installed as well as a music program called `bpmdj`. The owner being a version freak, the most recent version of a package is to be preferred but installing two versions of the same package should be avoided if possible.

Below we show the EOLP representation of the corresponding A-diagnostic program where a system model rule such as $bpmdj(1) \leftarrow install\_bpmdj(1), qt(2)$ asserts that `bpmdj` depends on `qt(2)`.

$$
\begin{array}{rl}
install\_qt(2) \leftarrow & \\
install\_qt(3) \leftarrow & \\
install\_kde(3) \leftarrow & \\
install\_bpmdj(1) \leftarrow & \\
\hline
\mathit{not}(install\_qt(3)) \leftarrow & \\
\hline
\mathit{not}(install\_qt(2)) \leftarrow & \\
\mathit{not}(install\_kde(3)) \leftarrow & \\
\mathit{not}(install\_bpmdj(1)) \leftarrow & \\
\hline
qt(2) \leftarrow & install\_qt(2) \\
qt(3) \leftarrow & install\_qt(3) \\
kde(3) \leftarrow & install\_kde(3), qt(X) \\
bpmdj(1) \leftarrow & install\_bpmdj(1), qt(2) \\
kde(3) \leftarrow & \mathit{not}(kde(3)) \\
bpmdj(1) \leftarrow & \mathit{not}(bpmdj(1))
\end{array}
$$

There is a single preferred explanation $E = \{install\_kde(3), install\_qt(2), install\_bpmdj(1)\}$, corresponding with the only proper preferred answer set $E \cup \{kde(3), qt(2), bpmdj(1)\}$. From the solution, it appears that a less recent version of `qt` is preferred because `bpmdj` compiles only with version 2 (not with 3) and `kde` can work with both version 2 and 3, making it unnecessary to install both `qt(2)` and `qt(3)`.

4 Diagnosing Ordered Systems

In this section, we consider problems where the system model is itself an ordered program. Naturally, in such a case, one would prefer explanations that maximally satisfy the system model, in particular more preferred rules should only be defeated as a last resort.

Definition 6. *A **diagnostic ordered system** is a triple $D = (P, C, O)$, where $P = \langle R, <_R \rangle$ is an OLP[3] describing either the normal behavior (consistency-based) or the abnormal behavior (abductive) of the system. Further, C is either a set of negated atoms in the case of consistency based diagnosis or a set of literals in the case of abductive diagnosis, representing the possible causes in the system; and O is a set of literals containing the observations.*

*A **hypothesis** is any subset $H \subseteq C$.*

- *If D is consistency-based, a hypothesis H is an **explanation** iff there exists an extended answer set Q of $R \cup \{h \leftarrow \mid h \in H \cup \neg(C \setminus H)\}$, such that $Q \cup O$ is consistent and $H = Q \cap C$.*
- *If D is abductive, a hypothesis H is an **explanation** iff there exists an extended answer set Q for $R \cup \{h \leftarrow \mid h \in H\} \cup \{not(h) \leftarrow \mid h \in C \setminus H\}$, such that $O \subseteq Q$ and $H = Q \cap C$.*

For an explanation H, we use $R_{H,Q}$ to denote the set $R_Q \subseteq R$, i.e. the reduct of R w.r.t. the extended answer set Q.

Note that there may be several extended answer sets Q, and associated reducts R_Q, to justify an explanation H.

Example 9. Consider the abductive diagnostic ordered system $D = (P, C, O)$ representing the trial of shooting incidents, where P is depicted below and $C = \{shoot, dead, unarmed, threatened\}$.

$$
\begin{array}{rl}
r_1: & guilty \leftarrow shoot, dead \\
r_2: & self\_defense \leftarrow threatened \\
\hline
r_3: & \neg guilty \leftarrow shoot, dead, self\_defense \\
r_4: & \neg self\_defense \leftarrow shoot, unarmed
\end{array}
$$

The preferred rules r_3 and r_4 state that one cannot be found guilty if one acted out of self defense and that self defense cannot be invoked if one shot an unarmed person. The more general rules r_1 and r_2 present the default treatment for a fatal shooting and a possible cause (having been threatened by the victim) for self defense.

Assuming that the facts of the case (i.e. the observations) are $F = \{shoot, dead, threatened\}$, the latter claimed by the defendant, a lawyer eager to obtain a conviction will search for an optimal explanation of $O = F \cup \{guilty\}$.

D has two explanations for O, namely $H_1 = \{shoot, dead, threatened\}$, corresponding to the answer set $Q_1 = O \cup \{self\_defense\}$ and $H_2 = H_1 \cup \{unarmed\}$ corresponding to both $Q_2 = O \cup \{unarmed, \neg self\_defense\}$ and $Q_2' = O \cup \{unarmed,$

[3] An OLP is an EOLP without negation as failure in the rules, i.e. R is a simple logic program, see [33].

$self\_defense\}$. The corresponding sets of satisfied rules w.r.t. these explanations are $R_{H_1,Q_1} = P \setminus \{r_3\}$, $R_{H_2,Q_2} = P \setminus \{r_2\}$ and $R_{H_2,Q_2'} = P \setminus \{r_3, r_4\}$.

The preference order among explanations is based on the $\preceq$ order among the corresponding sets of satisfied rules.

Definition 7. *Let D be a diagnostic ordered system with R_{H_1,Q_1} and R_{H_2,Q_2}[4] sets of rules corresponding with the explanations H_1 and H_2. Then, R_{H_1,Q_1} is preferred upon R_{H_2,Q_2}, denoted $R_{H_1,Q_1} \sqsubset R_{H_2,Q_2}$ iff* $\begin{cases} H_1 \subset H_2 & \text{if } R_{H_1,Q_1} = R_{H_2,Q_2} \\ R_{H_1,Q_1} \prec R_{H_2,Q_2} & \text{otherwise} \end{cases}$.
*An explanation H is **preferred** iff it corresponds to a minimal (w.r.t. $\sqsubset$) $R_{H,Q}$.*

Note that the special clause for $R_{H_1,Q_1} = R_{H_2,Q_2}$ is necessary, e.g. when both Q_1 and Q_2 satisfy all rules in R. In such a case, the smaller (w.r.t. $\subseteq$) explanation is preferred.

Example 10. In Example 9, $R_{H_2,Q_2} = P \setminus \{r_2\}$ is the unique minimal (w.r.t. $\sqsubset$). Therefore, the lawyer should attempt to establish $unarmed$ in order to obtain a conviction.

Using a similar intuition as in Definition 5, we can construct an EOLP program $L(D)$ that has exactly the $\sqsubset$-preferred explanations of a diagnostic ordered system as proper preferred answer sets.

Definition 8. *Let $D = (P = (R, <_R), C, O)$ be a diagnostic ordered system. The EOLP version $L(D)$ of D is defined by $L(D) = \langle R_a \cup R_n \cup R \cup R_o, R_o <<_r< R_n < R_a \rangle$, where $R_a = \{c \leftarrow | \ c \in C\}$, $R_n = \{\phi(c) \leftarrow | \ c \in C\}$ and $R_o = \{o \leftarrow \phi(o) \mid o \in O\}$, with $\phi(l) = \neg l$ if P is consistency based and $\phi(l) = \mathit{not}(l)$ if P is abductive.*

Example 11. The EOLP corresponding with the system from Example 9 is shown below.

$$
\begin{array}{rl}
dead \leftarrow & \\
shoot \leftarrow & \\
unarmed \leftarrow & \\
threatened \leftarrow & \\
\hline
\mathit{not}(dead) \leftarrow & \\
\mathit{not}(shoot) \leftarrow & \\
\mathit{not}(unarmed) \leftarrow & \\
\mathit{not}(threatened) \leftarrow & \\
\hline
guilty \leftarrow & shoot, dead \\
self\_defense \leftarrow & threatened \\
\hline
\neg guilty \leftarrow & shoot, dead, self\_defense \\
\neg self\_defense \leftarrow & shoot, unarmed \\
\hline
shoot \leftarrow & \mathit{not}(shoot) \\
dead \leftarrow & \mathit{not}(dead) \\
guilty \leftarrow & \mathit{not}(guilty) \\
threatened \leftarrow & \mathit{not}(threatened)
\end{array}
$$

The only preferred answer set is $Q = \{guilty, shoot, dead, unarmed, \neg self\_defense, threatened\}$, corresponding with the unique preferred explanation H_2.

[4] We abuse notation by considering $R_{H,Q}$ as a tagged set, such that $R_{H',Q'}$ may not be the same as $R_{H,Q}$ although, as sets of rules, $R_{H,Q} = R_{H',Q'}$.

In general we have the following correspondence.

Theorem 4. *Let $D = (P, C, O)$ be a diagnostic ordered system. Then, H is a preferred explanation for D iff $H = M \cap C$. for some proper preferred answer set M of $L(D)$.*

5 Complexity of Ordered Diagnosis

In the context of complexity for diagnostic systems [9, 28], the properties consistency, relevance and necessity are of natural interest, where consistency means deciding if there exists a preferred explanation, relevance and necessity refer to checking whether a given cause c is contained in some, resp. all, preferred explanation(s).

The availability of a transformation to EOLP's, suggests that ordered diagnostic reasoning resides in the same level of complexity[5].

Obviously, checking whether a hypothesis H is an explanation for an ordered diagnostic system D can be done in polynomial time. Thus, checking whether H is *not* a preferred explanation is in NP, i.e. guess a hypothesis H' such that $H' \sqsubseteq H$, which can be done in polynomial time, and verify if it is an explanation. Now, finding a preferred explanation H can be done by an NP algorithm that guesses H and uses an NP oracle to verify that it is not the case that H is not a preferred explanation. Hence, the following theorem.

Theorem 5. *Let $D = (S, <)$ be an ordered diagnostic system, with $S = (T, C, O)$ either a C- or A-diagnostic system. Deciding the problem of consistency for D is in NP. Deciding the problem of relevance, for a given cause $c \in C$, for D is in Σ_2^P, while deciding necessity for c is in Π_2^P.*

Showing that relevance (necessity) is also Σ_2^P-hard (Π_2^P-hard), can be done by a reduction to the known Σ_2^P problem of deciding whether a quantified boolean formula $\phi = \exists x_1, \ldots, x_n \cdot \forall y_1, \ldots, y_m \cdot F$ is valid, where we may assume that $F = \vee_{c \in C} c$ with each c a conjunction of literals over $X \cup Y$ with $X = \{x_1, \ldots, x_n\}$ and $Y = \{y_1, \ldots, y_m\}$ ($n, m > 0$). The construction is inspired by a similar result for abductive diagnosis under $\subseteq$-preferredness in [9], illustrating that the preferred explanation semantics does not involve any computational overhead w.r.t. classical diagnostic frameworks. Together with Theorem 5, this yields.

Theorem 6. *Let $D = (S, <)$ be an ordered diagnostic system, with $S = (T, C, O)$ either a C- or A-diagnostic system and let $c \in C$ be a cause. Deciding the problem of relevance for D and c is Σ_2^P-complete and deciding the problem of necessity for D and c is Π_2^P-complete.*

Similar results can be obtained for diagnostic ordered systems (Section 4).

Theorem 7. *Let $D = (P, C, O)$ with $P = (R, <_R)$ be a diagnostic ordered system. Deciding the problem of consistency for D is in NP. Deciding the problem of relevance, for a given cause $c \in C$, for D is Σ_2^P-complete, while deciding necessity for c is Π_2^P-complete.*

[5] The results in this section hold for both C- and A-ordered diagnostic reasoning.

6 Relationships To Other Approaches

Consistency-based diagnosis was proposed by [25], and extended in [6]. Definition 1 of C-diagnostic reasoning closely mirrors the original definition from [25], except that the notion of hypothesis in this paper contains only "malfunctioning" components, while the original definition considers total subsets of $C \cup \neg C$, i.e. also the correctly working components are mentioned.

A number of different characterizations of abductive diagnosis exist, both in the context of logic and logic programming, e.g. [4, 22, 21, 3, 17, 10, 9]. Earlier formalizations of abductive diagnosis used first order logic, while [17] introduced an abductive framework in the context of logic programming. Later, generalized stable models [11] were introduced as an extension of the stable model semantics [14] to handle abductive reasoning. Independently, [13] formalized a similar idea, called the belief set semantics, providing an abductive reasoning formalism for systems containing disjunction, negation as failure and classical negation. In [15, 16] this semantics was used to formalize abductive extended disjunctive programs. Another formalization of abduction for logic programming was given in [9], using definitions closer to ones used in first order logic approaches. Example 3 illustrates the difference between [9] and [15, 16].

Subset minimality has been recognized from the start [25] as a desirable property for explanations, along with other preference criteria such as single-error diagnosis, accepting only explanations containing a single cause, and minimality w.r.t. cardinality.

[8] mentions a possible formalization of preferred explanations for a linearly prioritized set of causes in the context of abduction for classical logic. The more general preference relation on explanations of Definition 4 reduces to the one used in [8], for the case where the underlying partial order on causes is linear.

Although a variety of proposals exist for extending logic programs with some kind of preference relation [18, 27, 1, 2, 34, 33], we are not aware of any prior work on abduction for such ordered programs (Section 4). In fact, many of these systems do not lend themselves to an approach along the lines of this paper. E.g. proposals such as [1], that select preferred answer sets from the collection of answer sets of the unordered version of the program, cannot deal with contradictions as appear e.g. in Example 9. On the other hand, several formalisms, e.g. [18], while similar to OLP, allow a rule to be defeated only by a better rule with opposite head. This prevents rules modeling normal behavior from being defeated by less-preferred opposite rules that may be needed to explain an abnormal observation, a feature of OLP that supports a natural and intuitive representation for abductive problems, see e.g. Example 7.

Reducing abduction to model computation has been done before, e.g. [26, 28] provide a different method for transforming abductive logic programs into disjunctive logic programs, using the possible model semantics, but only for the subset-preferred case. Section 3 can be regarded as an extension to more general preference relations. Moreover, our approach does not need disjunction to obtain the simulation as it relies on a single mechanism (order) to simulate both abduction and minimality.

7 Conclusions and Direction for Further Research

We have extended diagnostic reasoning to systems involving preference, in either the description or the set of causes. Since such reasoning can be simulated using EOLP, which is equivalent to OLP (i.e. programs without negation as failure) [31], an implementation of OLP[6], e.g. using the algorithms described in [33], can be envisaged to perform diagnostic reasoning.

The approach to preference in diagnostic reasoning can also be extended, e.g. by combining both preference on causes and in the system model.

The proposed diagnostic framework could also be useful to refine the concept of therapy from [12], where one tries to suppress some of the undesired observations by repairing (a subset of) the possible causes. This results in an iterative process of diagnosing the system, repairing some of the causes, called a treatment, and checking if the undesired observations have disappeared, in which case the therapy is finished. In an ordered diagnostic system (see Definition 4), it seems reasonable to let the choice of causes to repair depend on the preference order. E.g., if the order represents (repair) cost, only minimal elements of the explanation would be selected.

References

1. Gerhard Brewka and Thomas Eiter. Preferred answer sets for extended logic programs. *Artificial Intelligence*, 109(1-2):297–356, April 1999.
2. Francesco Buccafurri, Wolfgang Faber, and Nicola Leone. Disjunctive logic programs with inheritance. In Danny De Schreye, editor, *Logic Programming: The 1999 International Conference*, pages 79–93, Las Cruces, New Mexico, December 1999. MIT Press.
3. L. Console and P. Torasso. A spectrum of logical definitions of model-based diagnosis. *Computational Intelligence*, 7(3):133–141, 1991.
4. P.T. Cox and T. Pietrzykowski. General diagnosis by abductive inference. In *Proceedings of the IEEE Symposium on Logic Programming*, pages 183–189, 1987.
5. J. De Kleer. Local methods for localizing faults in electronic circuits. *MIT AI Memo*, (394).
6. J. De Kleer, A. K. Mackworth, and R. Reiter. Characterizing diagnoses and systems. *Artificial Intelligence*, 52:197–222, 1992.
7. Thomas Eiter, Wolfgang Faber, Nicola Leone, and Gerald Pfeifer. The diagnosis frontend of the dlv system. *AI Communications*, 12(1-2):99–111, 1999.
8. Thomas Eiter and Georg Gottlob. The complexity of logic-based abduction. *Journal of the Association for Computing Machinery*, 42(1):3–42, 1995.
9. Thomas Eiter, Georg Gottlob, and Nicola Leone. Abduction from logic programs: Semantics and complexity. *Theoretical Computer Science*, 189(1-2):129–177, 1997.
10. Thomas Eiter, Georg Gottlob, and Nicola Leone. Semantics and complexity for abduction from default logic. *Artificial Intelligence*, 90(1-2):177–222, 1997.
11. K. Eshghi and R.A. Kowalski. Abduction compared with negation by failure. In *Proceedings of the 6th International Conference on Logic Programming*, pages 234–254. MIT Press, 1989.
12. Gerhard Friedrich, Georg Gottlob, and Wolfgang Nejdl. Hypothesis classification, abductive diagnosis and therapy. In *Expert Systems in Engineering*, volume 462 of *Lecture Notes in Computer Science*, pages 69–78. Springer, 1990.

[6] A prototype system exists, computing preferred answer sets for EOLPs.

13. Michael Gelfond. Epistemic approach to formalization of commonsense reasoning. Technical report, University of Texas at El Paso, 1991. Technical Report TR-91-2.
14. Michael Gelfond and Vladimir Lifschitz. The stable model semantics for logic programming. In *Logic Programming, Proceedings of the Fifth International Conference and Symposium*, pages 1070–1080, Seattle, Washington, August 1988. The MIT Press.
15. Katsumi Inoue and Chiaki Sakama. Transforming abductive logic programs to disjunctive programs. In *Proceedings of the 10th International Conference on Logic Programming*, pages 335–353. MIT Press, 1993.
16. Katsumi Inoue and Chiaki Sakama. A fixpoint characterization of abductive logic programs. *Journal of Logic Programming*, 27(2):107.136, May 1996.
17. A. C. Kakas, R. A. Kowalski, and F. Toni. Abductive logic programming. *Journal of Logic and Computation*, 2(6):719–770, 1992.
18. Els Laenens and Dirk Vermeir. Assumption-free semantics for ordered logic programs: On the relationship between well-founded and stable partial models. *Journal of Logic and Computation*, 2(2):133–172, 1992.
19. Vladimir Lifschitz. Answer set programming and plan generation. *Journal of Artificial Intelligence*, 138(1-2):39–54, 2002.
20. F.J. Macartney. Diagnostic logic. *Logic in Medicine*, 1988.
21. Y. Peng and J. Reggia. Abductive inference models for diagnostic problem solving. *Symbolic Computation - Artificial Intelligence*, 1990.
22. D. Poole. Explanation and prediction: An architecture for default and abductive reasoning. *Computational Intelligence*, 5(1):97–110, 1989.
23. C. Preist, K. Eshghi, and B. Bertolino. Consistency-based and abductive diagnosis as generalized stable models. *Annals of Mathematics and Artificial Intelligence*, 11:51–74, 1994.
24. J.A. Reggia, D.S. Nau, and Y. Wang. Diagnostic expert systems based on a set covering model. *International Journal of Man Machine Studies*, (19):437–460, 1983.
25. Raymond Reiter. A theory of diagnosis from first principles. *Artificial Intelligence*, 32(1):57–95, 1987.
26. Chiaki Sakama and Katsumi Inoue. On the equivalence between disjunctive and abductive logic programs. In Pascal Van Hentenryck, editor, *Logic Programming, Proceedings of the Eleventh International Conference on Logic Programming*, pages 489–503, Santa Margherita Ligure, Italy, June 1994. MIT Press.
27. Chiaki Sakama and Katsumi Inoue. Representing priorities in logic programs. In Michael J. Maher, editor, *Proceedings of the 1996 Joint International Conference and Syposium on Logic Programming*, pages 82–96, Bonn, September 1996. MIT Press.
28. Chiaki Sakama and Katsumi Inoue. Abductive logic programming and disjunctive logic programming: their relationship and transferability. *The Journal of Logic Programming*, 44(1-3):71–96, 2000.
29. E.H. Shortliffe. Computer-based medical consultations: Mycin. 1976.
30. M. H. van Emden and R. A. Kowalski. The semantics of predicate logic as a programming language. *Journal of the Association for Computing Machinery*, 23(4):733–742, 1976.
31. Davy Van Nieuwenborgh and Dirk Vermeir. Order and negation as failure. Accepted.
32. Davy Van Nieuwenborgh and Dirk Vermeir. Ordered diagnosis. Technical report, Vrije Universiteit Brussel, Dept. of Computer Science, 2003.
33. Davy Van Nieuwenborgh and Dirk Vermeir. Preferred answer sets for ordered logic programs. In *European Workshop, JELIA 2002*, volume 2424 of *Lecture Notes in Artificial Intelligence*, pages 432–443, Cosenza, Italy, September 2002. Springer Verlag.
34. Kewen Wang, Lizhu Zhou, and Fangzhen Lin. Alternating fixpoint theory for logic programs with priority. In *CL*, volume 1861 of *Lecture Notes in Computer Science*, pages 164–178, London, UK, July 2000. Springer.