Proceedings of the Twenty-Ninth International
Florida Artificial Intelligence Research Society Conference

Iterated Abduction

Joshua Eckroth
Math/CS Department
Stetson University
jeckroth@stetson.edu

Abstract

Abduction is a pattern of inference in which an agent seeks an explanation for an observation or report. Iterated abduction is a variety of abduction in which evidence is acquired and explained over time. The long-term goal is to maintain highly plausible consistent explanations for as much of the evidence as possible. Some reports, at the time of acquisition, may be inconsistent with the agent's present beliefs, so some beliefs must be contracted in order to find an explanation of the new reports. Existing work in iterated belief revision only addresses how to maintain consistent beliefs in light of inconsistent observations; whether or not existing beliefs serve as explanations is not considered. What is needed to meet the goal of iterated abduction is a means of seeking new explanations for old evidence when previously-accepted explanations are contracted. We develop a logical formalism for this process as well as a computational implementation.

Abduction is a widely useful pattern of inference that reasons from evidence to explanations. Common examples of abduction include medical diagnosis, story understanding, and plan recognition. Given an observation or report q, abduction is the process of finding which p, among a set of alternatives consistent with existing beliefs, would best explain the evidence q.

The present work addresses the problem of *iterated abduction*, i.e., abducing explanations for a stream of evidence over time. We take evidence, explanations, and beliefs to be propositional statements. An explanation p can explain evidence q if p and q have a specific formal relationship, defined below, and p is consistent with prior beliefs. We assume that a single agent is performing abduction. An abductive operation at time t would, if successful, produce an explanation p for some received evidence q; this explanation p would be added to the agent's beliefs for future reasoning at times $t' > t$. Iterated abduction characterizes the way the agent both keeps track of a changing world and integrates new evidence about the current world. But new evidence may cast doubt on previously abduced explanations or outright contradict them. The agent must decide whether or not to contract beliefs, and which ones, in order to make sense of new evidence, while continuing to explain as much of the accumulated evidence as possible.

Iterated abduction is a combination of abduction and belief revision. Given evidence q, abduction produces an explanation p that is taken on as a belief. Upon learning new evidence q', an explanation is sought that is consistent with q, p, and the agent's other beliefs. Should no such explanation be found because, for example, every possible explanation p' for q' contradicts p, the agent must perform some kind of belief revision: either contract q' (disbelieve the recent evidence, as it is inconsistent with previously held beliefs), find an alternative explanation for q that does not contradict any p', or contract belief in prior evidence q as well as its explanation p.

As will be shown, related work on abduction, belief revision, and iterated belief revision miss a key feature of iterated abduction. Whenever the agent is forced to give up an explanation p for evidence q, the agent should seek an alternative explanation for q, not simply leave it unexplained upon contracting p. It is not always the case that such alternative explanations are available, in which case we require that the belief q must also be contracted for having no explanation.

In this report, we first describe the formal properties of iterated abduction, as well as a desideratum for belief revisions that occur during iterated abduction. We next describe a computational implementation that satisfies the desideratum, followed by a discussion. Next we review prior work and finish with concluding remarks.

Formal Properties

Our treatment of iterated abduction avoids restricting the language of beliefs. Let $\mathcal{L}$ be a propositional language and $p \in \mathcal{L}$ identify a statement in that language. Let $\rightsquigarrow \subset \mathcal{P}(\mathcal{L}) \times (\mathcal{L} \cup \{\bot\})$ be a relation between statements such that $P \rightsquigarrow q$, where $P = \{p_1, \ldots, p_n\} \subset \mathcal{L}$, means "the conjunction $\cap P$ can, if true, explain q." We write $p \rightsquigarrow q$ as a notational shorthand for $\{p\} \rightsquigarrow q$. A statement q may be *assumable*, i.e., requiring no explanation, which we denote as $\mapsto q$ as a shorthand for $\emptyset \rightsquigarrow q$. A set P of inconsistent statements may be denoted $P \rightsquigarrow \bot$; in this case, not all statements in P may be consistently simultaneously believed by the agent.

We assume the set of explanatory relations believed by

Copyright © 2016, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

the agent is exhaustive. This is a closed-world assumption in terms of the agent's beliefs about explanations and evidence. This closed-world assumption allows us to say that an agent must have an explanation for every belief q that requires explanation, otherwise it believes $\neg q$. Let $\mathbb{B}$ be the set of explanatory relations (including assumables) and other beliefs that the agent holds at any time. We restrict $\mathbb{B}$ such that for every atomic statement $q \in \mathcal{L}$ that cannot be assumed ($\mapsto q \notin \mathbb{B}$) but is nevertheless believed ($q \in \mathbb{B}$), then q has a believed explanation $P \subset \mathbb{B}$ where $P \rightsquigarrow q \in \mathbb{B}$. For every statement $q \in \mathcal{L}$ that participates in an explanatory relation (either $\mapsto q \in \mathbb{B}$ or $\exists P \subset \mathcal{L} : P \rightsquigarrow q \in \mathbb{B}$), we say q is *atomic*, identified by the predicate $A(q)$, and that the agent either believes q (i.e., $q \in \mathbb{B}$) or disbelieves q (i.e., $\neg q \in \mathbb{B}$).

We say $\mathbb{B}$ is a *belief system* if it meets the following criteria:

1. $\forall p :\mapsto p \in \mathbb{B} \rightarrow A(p)$.
2. $\forall P, q : P \rightsquigarrow q \in \mathbb{B} \rightarrow (\forall p \in P : A(p)) \wedge A(q)$.
3. $\forall p : A(p) \rightarrow (p \in \mathbb{B} \vee \neg p \in \mathbb{B})$.
4. $\forall P, q : P \rightsquigarrow q \in \mathbb{B} \rightarrow q \notin P$, i.e., atoms cannot directly participate in explaining themselves.
5. $\forall P, q : P \rightsquigarrow q \in \mathbb{B} \wedge P \subset \mathbb{B} \rightarrow q \in \mathbb{B}$, i.e., explanations imply what they explain.
6. $\neg\bot \in \mathbb{B}$, thus ensuring $\mathbb{B}$ is internally consistent. It is assumed, but not written for brevity's sake, that $\{p, \neg p\} \rightsquigarrow \bot$ for all atomic statements p.
7. $\forall q : (A(q) \wedge (q \in \mathbb{B})) \rightarrow (\mapsto q \in \mathbb{B} \vee (\exists P : P \rightsquigarrow q \in \mathbb{B} \wedge (P \subset \mathbb{B})))$, i.e., every belief that requires explanation is explained by some other belief(s).

The definition of a belief system forms the basis for iterated abduction. The crucial last property (7), that every belief requiring explanation has a believed explanation, constrains what counts as a valid abduction and ensures that belief revisions always result in an explanation for all beliefs that require an explanation.

Contraction and Abduction

During the course of reasoning about a stream of evidence, an agent may engage in two distinct kinds of reasoning activity:

- *Contraction*, or incorporating $\neg q$ into $\mathbb{B}$ for some $q \in \mathcal{L}$, denoted $\mathbb{B} - q$;
- *Abduction*, or incorporating q into $\mathbb{B}$ for some $q \in \mathcal{L}$, denoted $\mathbb{B} + q$.

In both cases, in order to ensure that the result is a belief system, explanations for prior beliefs may need to be contracted, and new explanations for prior and new beliefs may need to be established. These operations differ from contraction and expansion found in prior work in belief revision. Our contraction and abduction operations seek explanations for all believed evidence. Beliefs that require explanation (are not assumable) but cannot be explained must be contracted.

Desideratum for Iterated Abduction

Suppose that upon learning evidence q, p is picked out as the best explanation among a set of alternatives include the next-best alternative p'. Then, in order to consistently explain new reports, suppose p must be contracted. If q is thereafter left unexplained, then the long-term goal of iterated abduction may not be met. Rather, p' should be examined again and possibly accepted as the new best-explanation of earlier evidence q. Thus, the desired feature of any inference system that engages in iterated abduction is that the system seeks alternative explanations for prior evidence whenever prior explanations are contracted.

Most work on belief revision takes the reasonable stance that the optimal revision to an agent's beliefs is the revision that results in the fewest changes (acquisition of new beliefs, loss of existing beliefs). Finding minimal abductions is NP-complete (Bylander et al. 1991), as is minimal contraction (Tennant 2012). Thus, any practical iterated abduction system must employ heuristics for finding best explanations and contractions. The implementation we describe below supports such heuristics but we do not investigate specific heuristics in the present work.

Example

Consider the common example of wet grass w that can be explained by either rain r or sprinkler s. An agent has the following belief system: $\mathbb{B} = E \cup \{\neg r, \neg s, \neg w\}$, where $E = \{\mapsto r, \mapsto s, r \rightsquigarrow w, s \rightsquigarrow w\}$. Upon learning that the grass is wet, suppose the agent abduces rain: $\mathbb{B}{+}w = E \cup \{r, w, \neg s\}$. Then, upon learning that there was no rain, the agent alternatively abduces sprinkler: $(\mathbb{B}{+}w){-}r = E \cup \{s, w, \neg r\}$. Upon learning further that sprinkler is impossible, the agent has no remaining possible explanation for the grass being wet, so it contracts that belief as well: $((\mathbb{B} + w) - r) - s = \mathbb{B}$.

Implementation

We now describe a computational implementation of contraction and abduction operations that satisfy the postulates. Our implementation is based off Tennant's *finite dependency networks* (FDNs), which are capable of representing dependencies or *justifications* among statements (Tennant 2012). We reify *justification* as *explanation*. Tennant defined contraction and expansion operations for beliefs represented in FDNs. Tennant's expansion operation is only capable of determining consequences of beliefs and does not perform abduction. The present work adds the ability to perform abduction on FDNs and generalizes the contraction and abduction operations to a single priority-based consistency-restoration operation, described in detail later.

An FDN consists of nodes, strokes, and directed arrows connecting nodes and strokes. Every node represents a unique atomic statement. Strokes represent conjunctions of atomic statements. Notationally, Nx means x is a node, Sx means x is a stroke, and Axy means there is an arrow from x to y. Note that strokes may only point to nodes, and nodes may only point to strokes. In other words, for each arrow Axy, either $Sx \wedge Ny$ or $Nx \wedge Sy$. Furthermore, a stroke must point to exactly one node. We borrow Tennant's

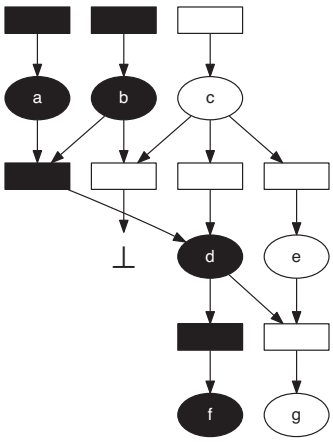


Figure 1: An example finite dependency network (FDN). Nodes a, b, d, f are believed while $c, e, g, \bot$ are disbelieved. Because b and c share a stroke that points to $\bot$, it is inconsistent for both b and c to be believed. The equivalent belief system is $\{\{a, b\} \rightsquigarrow d, c \rightsquigarrow d, c \rightsquigarrow e, d \rightsquigarrow f, \{d, e\} \rightsquigarrow g, \{b, c\} \rightsquigarrow \bot, a, b, \neg c, d, \neg e, f, \neg g, \neg\bot\}$.

convention and color every node and stroke either black (believed) or white (disbelieved). Notationally, Wx means x is white and Bx means x is black.

A belief system is isomorphic to an FDN in the following way.

- Atomic statements are nodes in the FDN: $(A(q) \wedge (\mapsto q \in \mathbb{B} \vee (\exists P : P \rightsquigarrow q \in \mathbb{B})) \leftrightarrow Nq$.
- Believed atomic statements are black nodes in the FDN: $(A(q) \wedge q \in \mathbb{B}) \leftrightarrow Bq$.
- The $\rightsquigarrow$ relation is represented by a stroke in the FDN: $P \rightsquigarrow q \in \mathbb{B} \leftrightarrow (\exists s : Ss \wedge (\forall p \in P : Aps) \wedge Asq)$.
- An assumable atomic statement p is represented by a stroke with no incoming arrows: $\mapsto p \in \mathbb{B} \leftrightarrow (\exists s : Ss \wedge Asp \wedge \neg(\exists p' : Ap's))$.

Figure 1 shows an example FDN and the equivalent belief system.

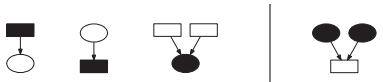
Axioms of Coloration

Tennant defined axioms of coloration that must be satisfied in order for the FDN to be consistent. Equivalently, these axioms codify the requirements for a consistent belief system.

(C1) Every black node receives an arrow from some black inference stroke, i.e., $\forall x((Bx \wedge Nx) \rightarrow \exists y(By \wedge Sy \wedge Ayx))$.

(C2) Every white node receives arrows (if any) only from white inference strokes, i.e., $\forall x((Wx \wedge Nx) \rightarrow \forall y(Ayx \rightarrow (Wy \wedge Sy)))$.

(C3) Every black inference stroke receives arrows (if any) only from black nodes, i.e., $\forall x((Bx \wedge Sx) \rightarrow \forall y(Ayx \rightarrow (By \wedge Ny)))$.



Deterministic bad strokes and nodes in contraction. In each case, the black node or stroke should be white.

Nondeterministic bad nodes in contraction. At least one black node should be white.

Figure 2: Patterns of bad strokes and nodes in a contraction operation. Note that a pair of nodes or strokes pointing to the same object represents any number of nodes or strokes, all sharing the same color and bearing similar arrows.

(C4) Every white inference stroke that receives an arrow receives an arrow from some white node, i.e., $\forall x((Wx \wedge Sx \wedge \exists z Azx) \rightarrow \exists y(Wy \wedge Ny \wedge Ayx))$.

(C5) The node $\bot$ is white, i.e., $W\bot$.

Contraction by p in an FDN begins by turning the node p to white (if the node is already white and the axioms of coloration are met, there is nothing to do). Once p has changed from black to white, the axioms of coloration may be broken, and consistency must be restored by appropriate color changes. Likewise, abduction by p begins by turning the node p to black (if the node is already black and the axioms of coloration are met, there is nothing to do), and then finding which other nodes and strokes must be black. The axioms of coloration identify which nodes and strokes have invalid local patterns of coloration. We address these patterns below.

Patterns of Invalid Coloration

After a contraction, nodes and strokes may be in any of four patterns of invalid coloration. These are shown in Figure 2. We call black nodes and strokes that should be white in order to meet the axioms of coloration by the names "bad nodes" and "bad strokes." As shown in the figure, three cases are deterministic. In those cases, there is no ambiguity about which nodes and strokes must be turned white in order to satisfy the axioms of coloration. However, in one pattern (right side of the figure), the choice is nondeterministic. At least one of the black incoming nodes must be turned white, but it is not certain which one (or more) should be white.

After an abduction (turning one or more nodes black), nodes and strokes may exhibit those patterns shown in Figure 3. These closely but not identically match the patterns related to contraction. Again, there is a single nondeterministic pattern in which a choice must be made which incoming white stroke must turn black in order to satisfy the axioms of coloration.

The patterns of invalid coloration for contraction are sufficient to define a contraction algorithm that restores consistency, according to the axioms of coloration, after contracting (turning white) a particular node. The procedure could simply identify all bad nodes and strokes in terms of contraction (Figure 2), color them white, and then repeat until the axioms of coloration are met. However, such a contraction algorithm behaves like those in prior work on belief revision: it is not capable of finding alternative explanations

Deterministic bad strokes and nodes in abduction. In each case, the white node or stroke should be black.

Nondeterministic bad strokes in abduction. At least one white stroke should be black.

Figure 3: Patterns of bad strokes and nodes in an abduction operation. The figure should be interpreted in the same manner as Figure 2.

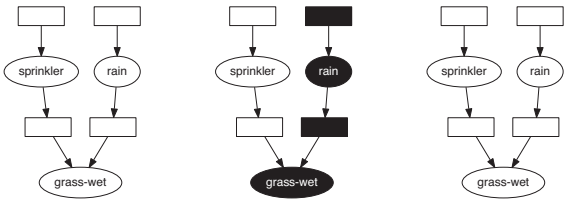


Figure 4: Upon learning *grass-wet*, the agent abduces *rain* (suppose) and moves from the left-most FDN to the middle FDN. But upon contracting *rain*, the axioms of coloration require that *grass-wet* be contracted as well. This does not meet our desideratum of iterated abduction that states *sprinkler* should be abduced to explain *grass-wet* when *rain* is contracted.

for previously explained evidence when it contracts those explanations.

For example, consider the wet grass scenario again. Upon learning that the grass is wet, the agent comes to believe (by abduction) that rain explains the wet grass. This is shown in the middle FDN of Figure 4. Then, rain is contracted, leaving the wet grass with no incoming black strokes. According to axiom (C1), the node representing the wet grass must turn white, as shown in the right-most FDN of the figure; in other words, belief in the wet grass must be contracted. However, according to our desideratum for iterated abduction, sprinkler should be automatically abduced instead as the alternative explanation for the wet grass. In other words, it should be the case that the node *grass-wet* is not a "bad node" and thus is able to remain black.

Priorities

Iterated abduction requires that at the agent know the order of abductions and contractions, specifically that *grass-wet* was abduced before *rain* was contracted, and that *sprinkler* was not (recently) abduced or contracted. This order of node and stroke color changes means that *sprinkler* remains a possible explanation for *grass-wet* but *rain* does not (after *rain* is contracted).

We can record the *priority* (or *timing*) of color changes by defining a function $T(\cdot)$ that maps nodes and strokes to the set of natural numbers. $T(x) = t$ means that node or stroke x acquired its current color (for the FDN in question) at time t. $T(x) = 0$ initially for all nodes and strokes. Whenever the agent acquires a new observation, the time counter increments. Suppose an observation induces an abduction or contraction at time t. Then every node and stroke x that changes color as a result of that abduction or contraction will be recorded as $T(x) = t$. Nodes and strokes retain their priority value until they change color again.

Consistency-Restoration

After observing p or $\neg p$ and coloring p black or white (respectively), consistency of the FDN may need to be restored according to the axioms of coloration. Previous work (Tennant 2012) distinguished between contraction and expansion operations. Contraction involves changing black nodes and strokes to white, and never changing a white node or stroke to black. Expansion (or abduction) only changes colors from white to black. Iterated abduction complicates this story and requires that nodes and strokes sometimes change white to black and back again as beliefs are contracted and alternative explanations are sought.

Contraction and abduction coloring behavior can be generalized to a consistency-restoration process that makes use of node and stroke priorities. The process is iterative. At each step, one or more nodes and/or strokes change color until the axioms of coloration are met. Which nodes and strokes may require a change of color are identified by the "bad" node and stroke criteria. The patterns of bad nodes and strokes from Figures 2 and 3 may be refined to take account of priority. In the formulas below, sets labeled $\mathcal{B}$ indicate bad black nodes and strokes (which should turn white to restore consistency), and sets labeled $\mathcal{W}$ indicate bad white nodes and strokes. Subscript $\mathcal{SD}$ indicates deterministic bad strokes, $\mathcal{SN}$ indicates nondeterministic bad strokes, and similarly for $\mathcal{ND}$ and $\mathcal{NN}$. Later, we also use the notation $\mathcal{B}_*$ and $\mathcal{W}_*$ to refer to any of $\mathcal{B}_{\mathcal{SD}}, \mathcal{B}_{\mathcal{ND}}, \mathcal{B}_{\mathcal{NN}}$ and $\mathcal{W}_{\mathcal{SD}}, \mathcal{W}_{\mathcal{SN}}, \mathcal{W}_{\mathcal{ND}}$, respectively.

$$
\begin{aligned}
\mathcal{B}_{\mathcal{SD}} &= \{s | Ss \wedge Bs \wedge \\
&\quad ([\exists n : Asn \wedge Wn \wedge T(s) \leq T(n)] \vee \\
&\quad [\exists n' : An's \wedge Wn' \wedge T(s) \leq T(n')])\} \\
\mathcal{B}_{\mathcal{ND}} &= \{n | Nn \wedge Bn \wedge (\forall s : Asn \rightarrow Ws) \wedge \\
&\quad (\forall s : Asn \rightarrow T(n) \leq T(s))\} \\
\mathcal{B}_{\mathcal{NN}} &= \{n | Nn \wedge Bn \wedge \\
&\quad (\exists s : Ans \wedge Ws \wedge \\
&\quad (\forall n' : An's \rightarrow Bn') \wedge T(n) \leq T(s))\} \\
\mathcal{W}_{\mathcal{SD}} &= \{s | Ss \wedge Ws \wedge (\forall n : Ans \rightarrow Bn) \wedge \\
&\quad (\exists n : Ans \wedge T(s) < T(n))\} \\
\mathcal{W}_{\mathcal{SN}} &= \{s | Ss \wedge Ws \wedge \\
&\quad (\exists n : Asn \wedge Bn \wedge \\
&\quad (\forall s' : As'n \rightarrow Ws') \wedge T(s) < T(n))\} \\
\mathcal{W}_{\mathcal{ND}} &= \{n | Nn \wedge Wn \wedge \\
&\quad ([\exists s : Asn \wedge Bs \wedge T(n) < T(s)] \vee \\
&\quad [\exists s : Ans \wedge Bs \wedge T(n) < T(s)])\}
\end{aligned}
$$

Notice that in order for a white node or stroke to be considered bad (a candidate for turning black), its priority must

be strictly less than its neighbors' priorities. Black nodes and strokes, on the other hand, are only required to be less than or equal in priority to their neighbors. This means that our system is more cautious about abducing beliefs (white to black) than it is about contracting beliefs (black to white).

Consistency-Restoration Algorithm

Restoring consistency after changing the color of a node is a straightforward iterative process.

1. If the FDN satisfies the axioms of coloration, we are done.
2. Otherwise, let $\mathcal{D} = \mathcal{B}_{\mathcal{SD}} \cup \mathcal{B}_{\mathcal{ND}} \cup \mathcal{W}_{\mathcal{SD}} \cup \mathcal{W}_{\mathcal{ND}}$, i.e., all deterministic bad nodes and strokes. Change the color of all of these (black to white or white to black), and repeat at step (1).
3. If no deterministic bad nodes or strokes exist, then let $\mathcal{N} = \mathcal{B}_{\mathcal{NN}} \cup \mathcal{W}_{\mathcal{SN}}$, i.e., all nondeterministic bad nodes and strokes. Select one of these according to a heuristic and change its color. Repeat at step (1).

Because the algorithm is iterative, it is necessary to show that it always terminates. The following lemma assists in proving termination.

Lemma 1. *The consistency-restoration algorithm will never color a node or stroke from black to white and then back to black.*

Proof. Notice that once node or stroke x is changed from black to white, $\forall y : (Ny \vee Sy) \rightarrow T(x) \geq T(y)$. Thus, x cannot be a member of any $\mathcal{W}_*$ since each such set requires that $T(x) < T(y)$ for certain strokes or nodes y. Hence, x will not turn black again after turning white while the algorithm is looping. □

Theorem 1. *The consistency-restoration algorithm always terminates.*

Proof. Notice that the algorithm terminates only when $\mathcal{X} = \mathcal{B}_{\mathcal{SD}} \cup \mathcal{B}_{\mathcal{ND}} \cup \mathcal{B}_{\mathcal{NN}} \cup \mathcal{W}_{\mathcal{SD}} \cup \mathcal{W}_{\mathcal{SN}} \cup \mathcal{W}_{\mathcal{ND}} = \emptyset$. At every step, at least one node or stroke in $\mathcal{X}$ changes color. Nodes and strokes in $\mathcal{W}_*$ (may) transition to $\mathcal{B}_*$ and vice versa. Since FDNs are finite, in order to get infinite looping, at least one node or stroke would have to cycle between $\mathcal{W}_*$ and $\mathcal{B}_*$ infinitely (not necessarily at every step in the algorithm). But that is not possible, since any node or stroke in $\mathcal{W}_*$ that changes color (to black) will never again appear in $\mathcal{W}_*$ due to the lemma. Thus, no node or stroke can infinitely cycle among $\mathcal{W}_*$ and $\mathcal{B}_*$. Therefore, eventually $X = \emptyset$ and the algorithm terminates. □

Example

Our wet grass example in Figure 4 failed to exhibit the right properties for iterated abduction. Now utilizing priorities and the consistency-restoration algorithm, we have the behavior shown in Figure 5. The agent starts with the following belief system: $\mathbb{B} = \{\mapsto r, \mapsto s, r \rightsquigarrow w, s \rightsquigarrow w, \neg r, \neg s, \neg w\}$, and priorities $T(\neg r) = T(\neg s) = T(\neg w) = 0$. Upon learning that the grass is wet, suppose the agent abduces rain: $\mathbb{B} + w = \mathbb{B} \cup \{r, w, \neg s\}$ and $T(w) = T(r) = 1, T(\neg s) = 0$. Then, upon learning that there was no rain, consistency restoration automatically abduces sprinkler: $(\mathbb{B} + w) - r = \mathbb{B} \cup \{s, w, \neg r\}, T(\neg r) = 2, T(s) = 2, T(w) = 1$. Upon learning further that sprinkler is impossible (not shown in the figure), there is no remaining possible explanation for the grass being wet, so consistency restoration results in the contraction of w as well: $((\mathbb{B} + w) - r) - s = \mathbb{B}, T(\neg r) = 2, T(\neg s) = T(\neg w) = 3$.

Discussion

The consistency-restoration algorithm makes use of priorities in order to determine which nodes and strokes are eligible for retraction. Whenever an explanation of a believed statement (black node) is contracted, an alternative explanation is abduced (colored black) if the statement was acquired more recently than at least one of its possible explanations. In this way, newer evidence takes priority but prior evidence remains explained, perhaps with alternative explanations, as long as possible. Note that we do not guarantee that all evidence remains explained no matter what kind of evidence is acquired. This, in general, is not possible to guarantee as the explanations for new evidence may be inconsistent with explanations for old evidence, and new evidence takes priority. Likewise, we cannot reasonably ensure that the fewest nodes are abduced or contracted, as doing so is computationally intractable.

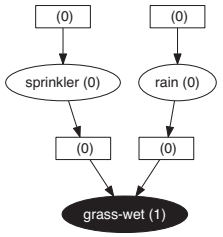
Our procedures rely on the closed-world assumption: once all possible explanations for a belief are contracted, the belief itself must be contracted (if it requires explanation, i.e., it is not assumable). However, this closed-world assumption may be useful in practice even though on the face of it the assumption is constrictive. Whenever a belief loses all of its explanations, an alarm can be presented to a knowledge engineer, who can then decide how to improve the knowledge base by providing more alternative explanations.

We suspect that priorities need not be discrete as we have done in this work, nor monotonically increasing as evidence is acquired. It may be possible to meet the desideratum of iterated abduction while allowing priorities to be continuous values and/or otherwise represent confidence in the evidence rather than a marker of time.

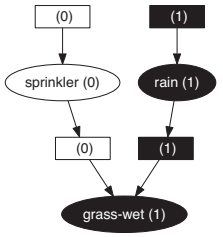
Related Work

Much of the work in belief revision is grounded in the AGM model, which defines a set of postulates for contraction and revision operations. Iterated abduction breaks the contraction postulate known as "inclusion," among others. The inclusion postulate states that the resulting beliefs after contraction should be a subset of the starting beliefs. Suppose p is abduced by our system to explain q. Then, upon contracting p, suppose p' is abduced instead to explain q. The resulting beliefs are not a subset of the beliefs prior to contraction.

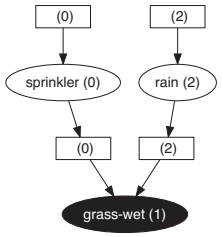
Abduction has been cast as belief revision (Aliseda 2006). In these approaches, a set of beliefs undergoes expansion by q when an explanation p may be found that implies q. Only p is explicitly added to the beliefs but q is a member of the logical closure of the set of beliefs. Revision involves



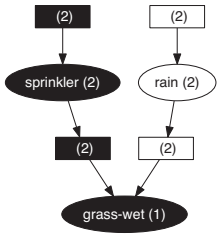
Time 1, observed grass-wet. Both strokes pointing to grass-wet are bad strokes (in $\mathcal{W}_{\mathcal{SN}}$), so a choice must be made.



Suppose the stroke from rain is chosen to turn black. The rain node and its incoming stroke are likewise turned black. The axioms of coloration are now satisfied.



Time 2, contracted rain; whitening spreads to nearby strokes. The grass-wet node does not meet the criteria for a bad node, since an incoming stroke has lower priority than grass-wet.



The only bad stroke or node is the stroke coming from sprinkler (the single member of $\mathcal{W}_{\mathcal{SN}}$), so it is turned black. Ultimately, consistency is restored again.

Figure 5: The wet grass example from Figure 4 but enhanced with priorities (shown in parentheses on the nodes and strokes). Only times 1–2 are shown. The resulting belief changes over time match our desideratum for iterated abduction.

contracting first and then expanding, in the usual way described by Levi's identity. In these approaches, however, future abductions are entirely novel problems and there is no explicit support for finding alternative explanations for prior evidence whose explanations are contracted.

Nayak and Foo developed a method of iterated abduction that left open alternative explanations as long as possible (Nayak and Foo 1999). They did so by only eliminating the worst explanations at each step in the stream of evidence rather than selecting the best explanation up to that point. Eventually, the best explanation that survived all the eliminations would be found. Thus, they reject that abductions should be minimal. However, their technique does not handle inconsistent evidence, as the gradual narrowing of possible explanations cannot be reversed.

Eckroth and Josephson developed an abductive reasoning system capable of reconsidering and revising prior abductions (Eckroth and Josephson 2014). Whenever evidence could not be plausibly explained, a metareasoning component attempted to identify which prior explanations should be retracted in order to allow an explanation for the current evidence. The metareasoning component's operational characteristics were not formally detailed, thus limiting a careful analysis of its behavior.

Finally, Beirlaen and Aliseda recently described a conditional logic for abduction that supports defeasible explanations that need not entail what they explain (Beirlaen and Aliseda 2014). For our purposes, the defeasible aspect is interesting because an explanation is therefore able to encode the conditions in which the explanation will no longer hold. Their approach properly identifies the explanations that are still viable (still consistent) after other alternative explanations have been defeated. However, they do not address the possibility of contradictory evidence or other belief dynamics beyond defeaters.

Conclusion

This work defined iterated abduction and presented a formal model and computational implementation of a system that meets the desideratum of iterated abduction. Upon acquiring new evidence or contracting existing beliefs, the system restores consistency while seeking alternative explanations for any prior evidence that has lost its support in the process. This is the first system that explicitly handles maintenance and recovery of prior abductions across iterated inferences.

Acknowledgments

This project was supported by the Brown Center for Faculty Innovation & Excellence at Stetson University. We also wish to thank John R. Josephson and the anonymous reviewers for their insightful comments.

References

Aliseda, A. 2006. *Abductive reasoning: Logical investigations into discovery and explanation*. Norwell, MA: Kluwer Academic Publishers.

Beirlaen, M., and Aliseda, A. 2014. A conditional logic for abduction. *Synthese* 191(15):3733–3758.

Bylander, T.; Allemang, D.; Tanner, M.; and Josephson, J. 1991. The computational complexity of abduction. *Artificial Intelligence* 49(1-3):25–60.

Eckroth, J., and Josephson, J. R. 2014. Anomaly-driven belief revision and noise detection by abductive metareasoning. *Advances in Cognitive Systems* 3:123–143.

Nayak, A. C., and Foo, N. Y. 1999. Abduction without minimality. In *Advanced Topics in Artificial Intelligence*. Springer. 365–377.

Tennant, N. 2012. *Changes of Mind: An Essay on Rational Belief Revision*. Oxford University Press.