Reasoning about Simultaneous Change in Trust and Belief

Aaron Hunter[1,*,†]

[1]British Columbia Institute of Technology, Burnaby, BC, Canada

Abstract

We consider domains where agents can receive information from observations, and also from reports. When an agent receives a sequence of observations and reports, this can trigger two changes. First, the agent's beliefs should change to incorporate new information received from trusted agents and from observations. Second, the agent's trust in other agents should change, depending on the accuracy of their reports. In this paper, we address this problem incrementally by first considering trust change. We introduce trust states, and we demonstrate how trust states can be explicitly updated by a new class of trust change operators. Trust change postulates are introduced, and a representation result is presented for these operators. We then demonstrate how we can use trust change operators to make implicit updates to trust in other agents, based on the accuracy of the reports provided by other agents. Broadly, agents are more strongly trusted when they provide reports that agree with observation and they are less strongly trusted when they provide reports that conflict with observation. We define combined trust-belief change operators that allow an agent to simultaneously update their trust in other agents while also revising their beliefs. We then introduce a software tool that automates this entire process. In other words, the software allows us to enter a sequence of reports and observations, and it returns both a new belief state and a new trust state. Applications and directions for future work are considered.

Keywords

Belief Revision, Trust, Knowledge Representation

1. Introduction

Belief revision occurs when an agent receives new information that must be incorporated with some previous beliefs. The most influential approaches to belief revision make the assumption that new information is "better" than the original beliefs. Hence, an agent should believe the new information is true while keeping as much of the initial belief state as consistently possible. However, when the new information comes from an agent that may not be honest, then this is no longer sensible. In these cases, there are actually two different concerns related to trust. First, trust impacts the likelihood that we will believe reports from other agents. This problem has been addressed to some extent in the literature. The second concern is related to *trust change*. Our level of trust in other agents should change, depending on how often their reports agree with our observations. This problem has not been addressed extensively in the context of belief revision operators. In this paper, we introduce a formal approach to modeling the simultaneous dynamics of trust and belief along with a prototype system that calculates the result of belief revision when new information includes observations and reports from other agents.

Our approach is the following. We first introduce *trust states*, along with a set of postulates for trust change; these postulates specify basic conditions that we expect to hold when we determine that some agent should be more (or less) trusted. We then prove a representation result for operators satisfying these postulates. Next, we introduce new *combined* change operators, which specify both how beliefs and trust should change when an agent receives a sequence of reports followed by an observation. Finally, we introduce our implemented system for solving problems involving the joint revision of trust and belief.

This paper makes several contributions to the literature on belief change. First, *trust change operators* have not been explored in detail in the theory of belief change. As such, both the set of trust-change postulates and the representation result are new contributions in the area. Another contribution is the introduction of combined change operators for trust and belief. Our approach makes the relationship between belief change and trust change explicit, framed in the context of a classical model of belief revision. Finally, the work here introduces a practical software tool that can opens up the opportunity to reason about practical applications related to reputation systems.[1]

2. Preliminaries

2.1. Belief Revision

The theory of belief revision is framed in the context of propositional logic. We assume an underlying propositional signature F, and we define propositional formulas in the usual manner using connectives $\wedge, \vee, \rightarrow$, and $\neg$. A *belief state* K is a logically closed set of formulas. The most influential approach to belief revision is the AGM approach, where a belief state K and a formula ϕ are mapped to a new belief state $K * \phi$ [3]. AGM revision is defined with respect to a set of postulates, and the revision is calculated semantically by finding the most plausible states that are consistent with the new formula [4].

AGM revision can only be used for single-shot belief revision. If we want to perform iterated belief revision, then the dominant approach is the DP approach [5]. In DP revision, the initial beliefs are represented by an *epistemic state*. Although the exact composition of an epistemic state is flexible [6], one important feature is that it includes a total pre-order over states. Our formal approach is defined for DP revision, but the implemented tool is based on variations of the Dalal revision operator [7]. This is an operator where plausibility is defined in terms of the Hamming distance between states; it is the rare example of an AGM revision operator that can be iterated. When we discuss our theoretical framework, we give all definitions with respect to epistemic states in the DP sense. However, in our practical tool, epistemic states are specified by a set of formulas along with a distance-based operator that defines the total pre-order.

22nd International Workshop on Nonmonotonic Reasoning, November 2-4, 2024, Hanoi, Vietnam

aaron_hunter@bcit.ca (A. Hunter)

[1] This paper is a combination of two existing papers. The content of sections 1-4, with some small modifications, iappears in [1]. The system description in section 5 was previously published in [2].

Most approaches to belief revision require that the new information *must* be believed; this is formalized by the so-called *success postulate* of AGM revision. As noted, this is not reasonable if information is reported by other agents. Two different approaches to this problem have been explored in the literature. There has been work on knowledge-based trust, where we only consider the 'part' of the information where the reporting agent has expertise[8, 9]. There has also been work on trust in terms of the reliability of information provided by different agents [10, 11, 12]. Throughout this paper, we assume an underlying revision operator which is then modified to capture some form of trust.

2.2. Trust

Trust has been studied extensively in distributed systems and network communication [13, 14]. However, it is not considered in many formal models of belief revision, where new information must be believed following revision. This is, of course, not reasonable if the new information is a report from another agent.

We distinguish between knowledge-based trust and reliability-based trust. Knowledge-based trust is concerned with the domain expertise of a reporting agent. For example, a doctor is trusted on medicine; they may not be trusted on other topics. Knowledge-based trust has been used as a means for ranking search results on the Internet [15]. There has also been work on knowledge-based trust in formal models of belief change[8, 16]. However, knowledge-based trust is not the focus of this paper.

We are concerned with reliability-based trust. An agent is reliable if their reports agree with known facts or direct observation. If an agent provides inaccurate reports, they will not be trusted. This has been addressed in [11], where a notion of *conflict* is introduced to determine which reports should be ignored. On the other hand, an agent that is not initially trusted may earn trust by continually providing accurate reports. This problem has not been directly addressed in connection with belief revision. We remark that honesty is one factor related to reliabiliy, but we do not assume agents are lying when a report is incorrect.

3. Trust Change

3.1. Motivating Example

Suppose we are investigating a crime scene and we can receive reports from two agents: Juan(J) and Alma(A). Juan is considered to be trustworthy, whereas Alma is not.

We are initially unsure if the door was forced open (F), and we believe that there is no crowbar in the house($\neg C$). So if our initial epistemic state is $\mathbb{E}$, then $B(\mathbb{E})$ should be the set of models of $(F \wedge \neg C) \vee (\neg F \wedge \neg C)$. Now suppose that we receive a report from Alma that the door was forced open and there is a crowbar in the house. Since Alma is not trusted, this report does not initially trigger a belief change. Juan then reports that the door was not forced open and that there is no crowbar in the house. The most plausible states in our new epistemic state $\mathbb{E}'$ are the models of $\neg F \wedge \neg C$. Suppose that we now observe a crowbar is in the living room. What should be believe now, and who should we trust?

It seems like we should decrease our trust Juan, since he has provided incorrect information. We also need to revisit our trust in Alma. She has provided a report that is consistent with our observation, so our trust in her should increase. We might even want to retroactively believe her initial report.

Reasoning about this kind of problem requires a model that keeps track of beliefs as well as trust in reporting agents. As information is acquired, we not only need to revise our beliefs - we also need to increase (resp. decrease) trust in agents that have provided accurate (resp. inaccurate) reports. In this paper, we introduce a family of combined trust-belief change operators for this purpose. We remark that this kind of reasoning does not only occur in commonsense problems; it is also the basis for trust in reputation systems [17, 18].

3.2. Graded Trust Change Operators

We introduce a model of trust change that is defined with respect to a set of agents $\mathbf{A}$. Belief change will be added in section 4. We first define *trust states*, which are ranking functions that capture the trust held in other agents.

Definition 1. *A trust state T is a function $T : \mathbf{A} \rightarrow \mathbb{Z}$. We write $\alpha(T) = \{A \mid T(A) \leq 0\}$, and we refer to this as the set of* trusted agents.

Informally, if $T(A) < T(B)$, then A is trusted more than B. The set $\alpha(T)$ is similar to the set of "believed" states in an epistemic state, but there is an important difference. Although the agents in $\alpha(T)$ are all trusted, we still can rank them and to determine which agents are most *strongly* trusted.

We now introduce a simple class of *trust change operators*. In the following definition, an *agent literal* is either A or -A, where $A \in \mathbf{A}$.

Definition 2. *A basic trust change operator is a function $\star$ that maps a trust state T and an agent literal L to a new trust state $T \star L$.*

Intuitively, $T \star A$ is the operation that occurs when A has done something that causes them to be more trusted. For example, if an agent provides a report that is consistent with direct observation, then we will increase trust in that agent. On the other hand, $T \star$ -A captures the situation where an agent becomes less trusted. This would occur, for example, when the agent has provided a report that is inconsistent with direct observation.

We give some desirable properties for basic trust change operators. The following postulates are all implicitly universally quantified over trust states T, T' and agents A, B. For clarity, we use square brackets to write $[T \star A](A)$, which is the value assigned to A by the trust state $T \star A$.

R1. $[T \star A](A) < T(A)$.
R2. $[T \star A](\text{-}A) > T(A)$.
R3. If $B \neq A$, then $T(B) = [T \star A](B)$ and $T(B) = [T \star \text{-}A](B)$.

Postulate $R1$ says that, when an agent A becomes more trusted, the T-ranking for A decreases. Postulate $R2$ makes the dual statement for agents that become less trusted. Postulate $R3$ states that changing the trust level associated with an agent A does not affect the trust level of any other agent.

We also introduce two postulates to ensure that $\star$ treats all agents equally. In other words, the magnitude of the trust change is equal for all agents:

R4. $T(A) - [T \star A(A)] = T(B) - [T \star B(B)]$.

R5. $[T \star \text{-}A](A) - T(A) = [T \star \text{-}B](B) - T(B)$.

Finally, the change in trust induced by $\star$ is the same for all trust states; the magnitute of trust change is determined by $\star$ and not by the initial trust state:

R6. $T(A) - [T \star A(A)] = T'(A) - [T' \star A(A)]$.
R7. $[T \star \text{-}A(A)] - T(A) = [T' \star \text{-}A(A)] - T'(A)$.

Taken together, these postulates define a class of basic trust change operators.

Definition 3. *A basic trust change operator T that satisfies postulates $R1 - R7$ is called a graded trust change operator.*

Some basic properties follow immediately.

Proposition 1. *Let T be a graded trust change operator. Then the following conditions hold: (1) If $A \in \alpha(T)$, then $A \in \alpha(T \star A)$ and (2) If $A \notin \alpha(T)$, then $A \notin \alpha(T \star \text{-}A)$.*

Hence, trusted agents remain trusted when we use $\star$ to increase trust. The reverse holds for untrusted agents that lose trust. These properties are immedidate consequences of $R1$ and $R2$, respectively.

The following proposition states that an agent can always become trusted after a finite number of trust strengthenings, and they can always become untrusted after a finite number finite number of weakenings.

Proposition 2. *If T is a graded trust change operator and $A \in \mathbf{A}$, then there is some n such that $A \in \alpha(T \star^n A)$. Similarly, there is some m such that $A \notin \alpha(T \star^m \text{-}A)$.*

This property is reminiscent of the key postulate for belief improvement operators [19]. This is not surprising, as graded trust change operators are also defined to induce incremental change.

3.3. Representation Result

We introduce a class of transformations on ranking functions over agents.

Definition 4. *Let $r : \mathbf{A} \to \mathbf{Z}$. If $n \in \mathbf{N}$, then define the ranking functions $r + (A, n)$ and $r - (A, n)$ as follows:*

$$[r + (A, n)](B) = \begin{cases} r(A) + n, & \text{if } A = B \\ r(A), & \text{otherwise} \end{cases}$$

$$[r - (A, n)](B) = \begin{cases} r(A) - n, & \text{if } A = B \\ r(A), & \text{otherwise} \end{cases}$$

Hence $r + (A, n)$ increases the ranking of A and $r - (A, n)$ decreases the ranking.

We can now give a representation result for graded trust change operators.

Proposition 3. *The function $\star$ is a graded trust change operator if and only if there exist positive integers s, w such that*

$$T \star L = \begin{cases} T - (A, s), & \text{if } L = A \text{ for some } A \in \mathbf{A} \\ T + (A, w), & \text{if } L = \text{-}A \text{ for some } A \in \mathbf{A} \end{cases}$$

Proof Suppose $\star$ is a graded trust operator. Let T_0 be a trust state, and let A_0 be a particular agent. By $R1$, there is some s such that

$$[T_0 \star A_0](A_0) + s = T_0(A_0)$$

. By $R4$, it follows that $[T_0 \star A](A) + s = T_0(A)$ for all agents A. By $R3$, we also know that $[T_0 \star B] = T_0(B)$ for all $B \neq A$. Moreover, by $R6$, we know that these equalities are actually true for all trust states T. Therefore $T \star A = T - (A, s)$; so the result holds for positive literals. By parallel reasoning, we can use propositions $R2$, $R3$, $R5$ and $R7$ to show that there is some w that validates the result for negative literals as well.

To prove the converse, suppose that we have two positive integers s, w that define $\star$ as in the definition. Then $R1$ holds because $[T \star A](A) + s = T(A)$ and $s > 0$. Similarly $R2$ holds for w. Postulate $R3$ holds from the definition of the $+$ and $-$ operators, which only increment the ranking for one agent at a time.

For any A, B and any T, T', we have the following equalities:

$$T(A) - [T \star A](A) = \quad s \quad = T(B) - [T \star B](B)$$
$$T(A) - [T \star A](A) = \quad s \quad = T'(A) - [T' \star A](A)$$

The first equality shows that $R4$ holds, and the second equality shows that $R6$ holds. We can prove $R5$ and $R7$ holds through similar equalities, using w as the middle value. Hence, $\star$ is a graded trust change operator. So graded trust change operators can be fully characterized by two positive integers: the strengthening constant s and the weakening constant w.

There are some interesting variations that we can give to characterize a larger set of basic trust change operators. The following is one such instance.

Proposition 4. *A basic trust change operator $\star$ satisfies postulates $R1 - R3$ and $R6 - R7$ if and only if, for each agent A there is a pair of positive integers s_A, w_A such that:*

$$T \star L = \begin{cases} T - (A, s_A), & \text{if } L = A \text{ for some } A \in \mathbf{A} \\ T + (A, w_A), & \text{if } L = \text{-}A \text{ for some } A \in \mathbf{A} \end{cases} \quad (1)$$

We call such an operator a non-uniform *graded trust operator.*

This proposition states that, if we omit postulates $R4$ and $R5$, then we have a class of operators that is characterized by strengthening and weakening constants that could be distinct for each agent. The proof is similar to Proposition 3.

Additional operators can be defined by modifying postulates $R6$ and $R7$. For example, we could model situations where trust is resilient by making trust decreases very small for strongly trusted agents. We leave a full exploration of such variations for future work.

3.4. Iterated Trust Change

We have defined graded trust change operators for a single agent literal L. However, we will generally be interested in sequences of literals $\overline{L} = L_1, \ldots, L_n$. We will write $T \star \overline{L}$ as a shorthand for $T \star L_1 \star \cdots \star L_n$. Each literal L_i represents a single data point, indicating evidence that a particular agent should be more (or less) trusted. We adopt the following notation:

$$\overline{L}_a = |\{A \mid A \text{ in } \overline{L}\}|$$
$$\overline{L}_c = |\{A \mid \text{-}A \text{ in } \overline{L}\}|$$

Hence $\overline{L}_a$ is the number of postive literals in $\overline{L}$ and $\overline{L}_c$ is the number of negative literals in $\overline{L}$. The a stands for *agreement* while the c stands for *conflict.*

Proposition 5. *Let $\star$ be a graded trust operator, defined by s and w. Then*

$$[T \star \overline{L}](A) = T(A) - \overline{L}_a s + \overline{L}_c w.$$

This result follows directly from Proposition 3, since each increase or decrease is handled independently. So the iterated trust over a sequence of changes is just the aggregate of individual trust change operations. As a result, for any operator $\star$, any sequence $\overline{L}$, and any agent A we can define the following value:

$$\Delta(\star, \overline{L}, A) = [T \star \overline{L}](A) - T(A).$$

Hence, Δ represents the *change* in trust for agent A given the operator $\star$ and the sequence $\overline{L}$. The properties of this change value are given below.

Proposition 6. *Let $\star$ be a graded trust change operator. Then:*

1. *If $\overline{L}_a = \overline{L}_c = 0$, then $\Delta(\star, \overline{L}, A) = 0$.*
2. *If $\overline{L}_a = 0$ and $\overline{L}_c > 0$, then $\Delta(\star, \overline{L}, A) > 0$.*
3. *If $\overline{L}_c = 0$ and $\overline{L}_a > 0$, then $\Delta(\star, \overline{L}, A) < 0$.*
4. *If $\overline{L}_c = \overline{M}_c$ and $\overline{L}_a > \overline{M}_a$ then $\Delta(\star, \overline{L}, A) < \Delta(\star, \overline{M}, A)$.*
5. *If $\overline{L}_a = \overline{M}_a$ and $\overline{L}_c < \overline{M}_c$ then $\Delta(\star, \overline{L}, A) > \Delta(\star, \overline{M}, A)$.*

Item (1) asserts that trust in A does not change if A does not occur in $\overline{L}$. Item (2) says that A becomes less trusted if they only occur in *conflict* literals, while item (3) says the reverse for agents that occur only in *agreement* literals. Item (4) compares different sequences. It says that, if two sequences include the same number of conflict literals, then the one with more agreement literals will have a more positive impact on trust for A. Item (5) makes a similar statement for the case where the number of agreement literals is the same.

Proposition 6 summarizes the properties of aggregate trust change. However, since s and w are not constrained, we can not say anything specific about the aggregate change due to a sequence that includes both confict and agreement.

Proposition 7. *Let T be a trust state, let $A \in \mathbf{A}$, and let $\overline{L}$ be any sequence of agent literals that contains at least one instance of A and at least one instance of -A. Then there are graded trust change operators $\star_1$, $\star_2$ and $\star_3$ such that*

$$\Delta(\star_1, \overline{L}, A) < 0 = \Delta(\star_2, \overline{L}, A) < \Delta(\star_3, \overline{L}, A).$$

Hence, in the general case, there is no way to determine if $\Delta(\star, \overline{L}, A)$ is positive or negative. This flexibility allows us to define graded trust change operators that handle agreement and conflict very differently. For example, a single conflict might increase the trust ranking as much as a million agreements. So if $\overline{L}$ contains both a strengthening and a weakening for A, then we can not say anything about whether or not A will be trusted unless we know the specific change operator.

4. Interacting Trust and Belief

4.1. Reports and Histories

We now move to the case involving both trust and belief. So we need a signature that includes both agents and properties of the world.

Definition 5. *A multi-agent signature is a pair $\langle \mathbf{A}, \mathbf{V} \rangle$ where $\mathbf{A}$ is a set of agents, $\mathbf{V}$ is a propositional signature.*

The important connection between agents and formulas is that agents provide *reports*, and the content of a report is a propositional formula.

Definition 6. *A* report *is a pair (A, ϕ) where $A \in \mathbf{A}$ and ϕ is a formula over $\mathbf{V}$. We write $\overline{R} = (A_1, \phi_1), \dots, (A_n, \phi_n)$ for a finite sequence of reports.*

The problems that we address involve both reports and *observations*. We will normally be concerned with sequences of reports followed by an observation. This concept is formalized below.

Definition 7. *A history-sensitive observation (hs-observation) is a pair $\langle \overline{R}, \phi \rangle$ where $\overline{R}$ is a report history and ϕ is a formula.*

Defining belief change with respect to hs-observations allows us to consider how the observation ϕ informs the extent to which the reports in $\overline{R}$ should be incorporated. In order to represent an agent's beliefs along with their trust in other agents, we define the following notion of an *epistemic trust state*.

Definition 8. *An* epistemic trust state *is a pair $\langle \mathbb{E}, T \rangle$ where $\mathbb{E}$ is an epistemic state over $\mathbf{V}$ and T is a trust state over $\mathbf{A}$.*

Note that $\mathbb{E}$ and T are independent, but we will define change operators that impact them both at the same time.

4.2. A Family of Combined Change Operators

We now define combined change operators for trust and belief. The first step is to show how an hs-observation defines a sequence of trust change operations.

Definition 9. *Let $\langle \overline{R}, \phi \rangle$ be an hs-observation where $\overline{R} = (A_1, \phi_1), \dots, (A_n, \phi_n)$. For each $i \leq n$, let:*

$$L_i = \begin{cases} A_i, & \text{if } \phi_i \not\models \phi \\ \text{-}A_i, & \text{if } \phi \models \phi. \end{cases}$$

Let $\tau(\langle \overline{R}, \phi \rangle) = L_1, \dots, L_n$.

Hence, $\tau(\langle \overline{R}, \phi \rangle)$ is a sequence of literals. The literal in position i is A_i if the formula reported by A_i in position i is consistent with ϕ. The literal in position i is -A_i if the formula reported by A_i in position i is inconsistent with ϕ. Intuitively, this sequence encodes how our trust in each agent should change given the hs-observation $\langle \overline{R}, \phi \rangle$; the agent should be more trusted if they have provided reports consistent with ϕ and they should be less trusted if they have provided reports inconsistent with ϕ.

We use Definition 9 to overload the $\star$ operator, by allowing it to take an hs-observation as input. Specifically, we adopt the following notation::

$$T \star \langle \overline{R}, \phi \rangle = T \star \tau(\langle \overline{R}, \phi \rangle).$$

Hence, when we given an hs-observation as an input to $\star$, we simply pass to the sequence of agent literals $\tau(\langle \overline{R}, \phi \rangle)$. This sequence of literals captures the number of conflict and agreement reports that each agent has provided.

We need one more piece of notation. Given a report history $\overline{R}$ and a set of agents β, we write $\overline{R} \restriction_\beta$ as a short hand for the sequence of formulas ϕ_i where $A_i \in \beta$. So if β represents the set of trusted agents, then $\overline{R} \restriction_\beta$ represents the sequence of formulas reported by trusted agents.

We can now define an approach to combined change for trust and belief.

Definition 10. *Let $\langle \mathbb{E}, T \rangle$ be an epistemic trust state, let $*$ be a DP operator, and let $\star$ be a graded trust change operator. Then $\circ$ is defined as follows:*[2]

$$\langle \mathbb{E}, T \rangle \circ \langle \overline{R}, \phi \rangle = \langle E * \overline{R} \restriction_\beta *\phi, T \star \langle \overline{R}, \phi \rangle \rangle$$

where $\beta = \alpha(T \star \langle \overline{R}, \phi \rangle)$.

Hence, the new trust state is obtained by strengthening and weakening trust in agents, based on whether they have provided reports that are consistent with the observation ϕ. The new epistemic state is obtained by iteratively revising by all reports from trusted agents, and then revising by the observation ϕ. Note that the set of trusted agents used for this revision is determined *after* any trust changes resulting from the given sequence of reports. We illustrate by returning to our motivating example.

Example We can further formalize our motivating example involving Juan(J) and Alma (A) at the crime scene. Let T be the trust state where $T(J) = -1$ and $T(A) = 1$, which reflects our assumption that Juan is initially trusted, while Alma is not. Suppose that $\star$ is the operator defined by the constant 2 for both strengthening and weakening. Recall that Alma reports $F \wedge C$, then Juan reports $\neg F \wedge \neg C$, then we observe C. So we need to calculate the following:

$$\langle \mathbb{E}, T \rangle \circ \langle (A, F \wedge C), (J, \neg F \wedge \neg C), C \rangle.$$

From Definition 10, our new trust state T' assigns $T'(J) = 1$ and $T'(A) = -1$. So after all events, only Alma will both be trusted. This also means that the final epistemic state will be $\mathbb{E} * (F \wedge C) * C$. Hence, we will not only believe the crowbar is in the house, but we will also believe the door was forced open. This is because Alma's report has been incorporated, since she is now trusted.

Note that the we can get a different result, if we return to the example with one small tweak.

Example Consider the same example, except that $T(J) = -3$ while $T(A) = 1$. In this case, the new trust state T' assigns $T'(J) = -1$ and $T'(A) = -1$. So, despite the fact that Juan has provided an erroneous report, he is still trusted. The intuition here is that Juan has built such a strong reputation that he will still be trusted after a single mistake. In this case, the final epistemic state will be $\mathbb{E} * (F \wedge C) * (\neg F \wedge \neg C) * C$. Following this sequence of revisions, we will believe the crowbar is in the house but we will not believe the door was forced open. This holds despite the fact that Alma has reported otherwise, because Juan is still a trusted source.

Many other small tweaks that could be made to get different results. For example, if the $\star$ operator only strengthens trust with a constant $s = 1$, then Alma will not be trusted despite the accuracy of her report. In all of these cases, the basic framework handles the subtle distinctions without any difficulty.

4.3. An Observation-Consistent Variation

A *report filter* is any function that maps an hs-observation $\langle \overline{R}, \phi \rangle$ to a new hs-observation including a subsequence of the reports from the original. A report filter is *observation-consistent* if every report in the output subsequence must be consistent with ϕ.

Proposition 8. *Let T be a trust state. The report filter $\restriction_{\alpha(T)}$ is not guaranteed to be observation consistent.*

This result is important, because it means the $\circ$ operator is based on a filter that can include reports that are inconsistent with the observation.

We have seen this in Example 2, where Juan's report influences the final epistemic state despite the fact that it is inconsistent with the observation. This seems reasonable in this case, because the report is a conjunction and we end up keeping only the "consistent part." However, in some applications, it would be preferable to discard all inconsistent reports regardless of how much the sender is trusted. We can enforce this condition by providing a modified definition of $\circ$.

Let $\langle \overline{R}, \phi \rangle$ be an hs-observation and let γ be the set of agents that have provided a report in $\overline{R}$ that is inconsistent with ϕ. The following is immediate.

Proposition 9. *Let T be a trust state. For every hs-obsrvation, the report filter $\restriction_{\alpha(T) \cup \gamma}$ is observation consistent.*

This simple change gives us a variation of $\circ$ that is based on an observation-consistent filter. Specifically, we can modify the definition of $\circ$ to define $\circ_{OC}$ as follows:

$$\langle \mathbb{E}, T \rangle \circ_{OC} \langle \overline{R}, \phi \rangle = \langle E * \overline{R} \restriction_{\beta \cup \gamma} *\phi, T \star \langle \overline{R}, \phi \rangle \rangle$$

where β is defined as it was previously. Hence $\circ_{OC}$ is just like $\circ$, except that it filters out all reports inconsistent with ϕ before performing belief revision.

5. System Description

5.1. Overview

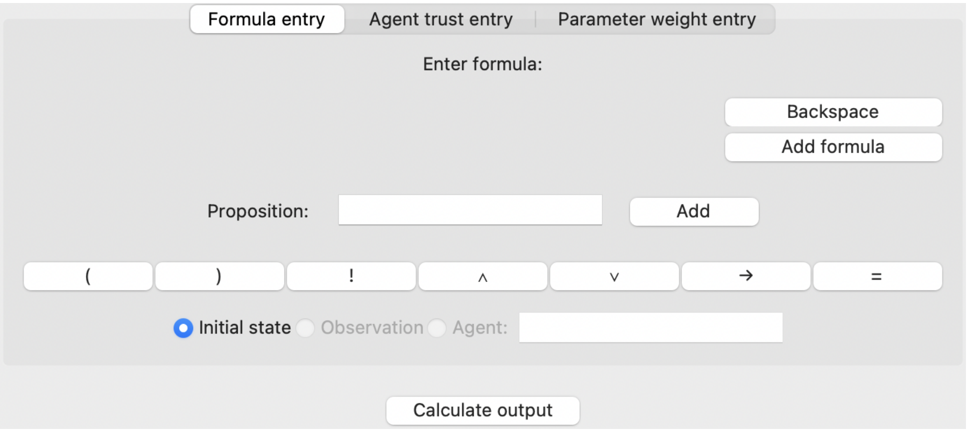
The Honesty-based Belief revision System (HBS) is a tool written in Python to automatically solve belief revision problems[3]. The system allows the user to specify an initial belief state, along with a sequence of reports and observations. The system has a graphical user interface for interactive model, where the user enters the reports and observations directly; it also allows users to enter the required information through external files. In this section, we describe the main funcationality of the software.

5.2. Interface

The interface for HBS is shown in Figure 1. When HBS is launched, it will initially be set to the *Formula Entry* tab. While the *Initial state* radio button is highlighted, this allows the user to enter a set of formulas that define the initial belief state. In interactive mode, formulas are entered with a simple graphic interface that prevents syntax errors. The formula being defined is displayed above the entry box, and it is entered as part of the initial belief state when the user clicks on *Add Formula*.

[2] Note that $\circ$ actually depends on $*$ and $\star$, so it would be more appropriate to write $\circ_{*,\star}$. But this notation is cumbersome, so we omit the subscripts unless they are required to reduce ambiguity.

[3] Software available at https://github.com/amhunter/HBS.



● Initial state ○ Observation ○ Agent:

Figure 1: HBS Interface

The initial belief state can be modified iteratively by adding more formulas, and a panel on the right will display the set of states believed possible. The radio button at the bottom can be toggled to add observations or reports. For observations, the formula is added to the right panel and labelled as an observation. For reports, an agent name must also be provided. All of the items listed in the right panel can be deleted at any time by clicking the X in the corner. As such, what the interface allows the user to do is to specify an expression of the form:

$$K * (A_1, \phi_1) * \psi_1 * \cdots * (A_n, \phi_n) * \psi_m.$$

The user can click *Calculate Output* to determine the new belief state after the given sequence of operations. Figure 2 shows the contents of the right panel after entering the following:

$$Cn[\neg(A \land \neg B)] * (\text{alice}, A \lor B) * (\text{bob}, A \land B) * (A \to \neg B)$$

The output at the bottom indicates the new belief state; we explain below how this is determined. We remark that the display can be modified to a simplified form; this will hide the lists of truth values for variables. This can be helpful for large examples with many variables.

Note that the main interface also includes the Agent Trust Entry tab. In this tab, the user can enter the initial trust ranking for all agents.

5.3. Trust

Trust change in HBS is based on a set of parameters. There are six different parameters available, listed in the following table.

Obs. Decrease (to^-)	Rep. Decrease (tr^-)
Obs. Increase (to^+)	Rep. Increase (tr^-)
No Trust Threshold (min)	Difference Threshold (L)

Informally, to^- and to^+ are the amounts to decrease (resp. increase) the trust in a reporting agent when they have provided a report that conflicts with an observation. These paramaters represent the trust strengthening and trust weakening parameters from the previous section. Similarly, min is a flexible parameter that allows us to specify set of trusted agents; in our formal theory this was fixed at 0.

The remaining parameters are new, and they are concerned with conflicting reports. The paramater tr^- is a new change constant, which is the amount that $T(A)$ should decrease if A provides a report that conflicts with a more trusted agent. Similarly, tr^+ is the amount that $T(A)$ should increase if A provides a report that agrees with a more trusted agent. However, we do not necessarily make these changes in all cases. We only make these changes if the trust differential between A and the other agent is greater than the difference threshold L.

Essentially, these parameters allow the implementation to deal with a slightly more complex notion of trust change. Changes in trust no longer depend on conflict with observation, agents can now become more (or less) trusted based on how much they agree with other reporting agents. However, we remark that we essentially have graded trust change operators if we restrict tr^- and tr^+ to be 0.

The change in trust that occurs with this full set of parameters is given in the following definition.

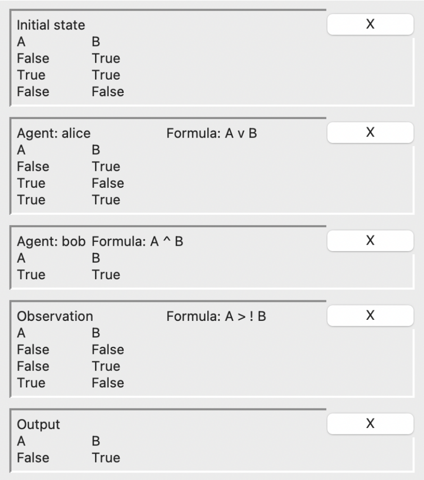
Definition 11. *Let $T_{\min,t}$ be a trust state and let $P = \langle to^+, to^-, tr^+, tr^-, L\rangle$ be a tuple of natural numbers (called the trust parameter). Define $T \cdot (\langle \overline{R}, \phi\rangle, P) = T'$ where $T'(A)$ is specified by applying the following procedure:*

1. *Initially, set $T'(A) = T(A)$. Update as follows by comparison with ϕ:*
   - *If there is a report (A, ψ) in $\overline{R}$ such that $\phi \models \neg\psi$, then $T'(A) = T(A) - to^-$.*
   - *If there is a report (A, ψ) in $\overline{R}$ such that $\phi \models \psi$, then $T'(A) = T(A) + to^+$.*
2. *Next, compare with reports. For all agents B with $T(A) < T(B)$, if $T(B) - T(A) > L$, then:*
   - *If there are reports (A, ψ) and (B, τ) in $\overline{R}$ with $\tau \models \neg\psi$, then $T'(A) = T(A) - tr^-$.*
   - *If there are reports (A, ψ) and (B, τ) in $\overline{R}$ with $\tau \models \psi$, then $T'(A) = T(A) + tr^+$.*

In HBS, the default values for all parameters is 1, but these values can be modified either through the interface or by editing the values in the `revision.py` file.

5.4. Belief Revision Operators

We specify a default "idealized" revision operator. The idealized operator is the revision operator that would be used if we only had to incorporate observations. In HBS, the default revision operator is the Dalal operator based on the



```
Initial state
A           B
False       True
True        True
False       False

Agent: alice            Formula: A v B
A           B
False       True
True        False
True        True

Agent: bob  Formula: A ^ B
A           B
True        True

Observation             Formula: A > ! B
A           B
False       False
False       True
True        False

Output
A           B
False       True
```

Figure 2: HBS Output

Hamming distance between states. However, this is not the only revision operator that HBS can capture.

Under the edit menu, the user can change to a weighted Hamming distance operator. In this case, a weight needs to be associated with each variable and these weights are used in the distance calculation. It is easy to see that this approach to revision can capture many operators beyond the standard Dalal operator. For example, if we use powers of two for the weights, we can essentially specify a priority ordering over paramaters. Hence, the weighted Hamming distance can be used to capture any parametrized difference operator [20]. This is a natural class of revision operators suitable for iteration, with nice computational properties.

5.5. Calculation

Suppose that a series of reports followed by a single observation has been entered, and the user presses 'Calculate output.' Then the following calculations are performed:

1. First, the trust values are updated in accordance with Definition 11.
2. The new trust values are used to determine the subsequence of formulas for revision.
3. The revision is performed based on the selected revision operator.
4. The new belief state is displayed; it will be used for future revisions.

If the sequence of reports and observations involves several observations (rather than a single terminal observation), then step (1) and (2) involve multiple sweeps through the reports to remove those that are inconsistent with *any* observation.

Hence, HBS takes a sequence of reports and a single observation, and it returns a new epistemic trust state. It includes the calculation of graded trust change operators as a special case, but it also allows a wider range of operators to be specified. Of course, when we use non-zero values for the new parameters, we no longer have a clear synactic definition for the operators.

We remark also that HBS can actually be used to solve iterated change problems with multiple observations. Since all of the revision operators included support iteration, we simply need to update the trust levels for each agent after every observation before continuing with more reports and observations.

5.6. Creating Test Cases

Although the interface allows the user to enter a long sequence of formulas, it can be cumbersome to do so. In order to make the software easier to use, there is also a mechanism to load test cases from a text file in the following format:

```
(Av!A)^B/!A^!B
1,2
alice:AvB
bob:A^B
:A>!B
```

The first line specifies a set of formulas, separated by slashes. The second line gives weights to all variables, in the order that they appear in the input. These values are for the weighted Hamming distance; they should all be set to 1 if Dalal revision is preferred. The remaining lines specify reports in the form "agent:formula." If the agent part is left blank, then the line represents an observation. When a test case is loaded from a file in this format, then it automatically populates the right panel with all of the information. This is a much easier way to enter examples involving a long sequence of reports.

6. Discussion

6.1. Related Work

Trust has been explored in a variety of formal settings involving agents with limited beliefs exchanging information [10, 8, 12, 16, 9]. The work in this paper differs in that we focus explicitly on the interaction of a belief revision operator with a dynamic notion of trust that changes as reports are received. The work in this paper is also distinguished by the fact that we provide an implemented system for experimentation with trust and belief change.

There has been previous work on the interaction between observations and reports in [11], where a notion of *conflict* is used to determine which reports should be ignored. However, the notion of conflict introduced is restricted, as it is based solely on counting inconsistent reports. More importantly, the framework does not include trust rankings or any model of trust dynamics. There has also been previous work on trust revision, in the tradition started with [21]; however, this work does not consider any direct connection with belief change operators.

Perhaps the closest work in the literature to our approach is in [22], where the authors argue that trust change and belief change can not be separated. They introduce a new class of *information revision* operators that operate on a hybrid state which includes both beliefs and trust. While the motivation of this work is similar to ours, the framework is quite different. Whereas information revision operators are built from scratch to model a single change operation, we build our approach from independent change operators for beliefs and trust. Hence, we maintain that belief and trust change are distinct operations; however, they need to be combined to effectively incorporate reports and observations. In future work, we intend to explore the formal relationship between the two approaches, and the extent to which information revision can be embedded in our work.

6.2. Speculative Application

We briefly introduce a potential application for our framework. We propose that graded trust change operators and HBS are well suited for for reasoning about *reputation systems*, where we have a seller that has been rated based on a series of transactions. The information provided by ratings need not simply be judgments about the "goodness" of the seller; they might include information about the product, the promptness of delivery, or anything else about the transaction. All of this can be encoded in a suitable logical theory. When we read a series of reviews and then make a purchase, we are able to simultaneously update our beliefs and our trust in the ratings through the framework introduced in this paper. Automating this process, we can implement a simple reputation system that can maintain a sound trust ranking over all agents providing reviews.

The automation step here is actually not difficult. Using HBS, we can solve report revision problems by encoding them in the format specified.

```
(Av!B)^(B^!C)
Jordan:AvB, Alma:A^B, obs:AV!B
```

There is a problem here in that real reputation systems often include thousands of reviews; the current iteration of HBS does not use a fast revision solver, so it runs slowly on problems involving many formulas. However, it is possible to signicantly improve the running time for revision solvers by using a competition-level ALLSAT solver [23]. In the next iteration of the software, we will use this approach to produce a tool that is useful for reasoning about much larger problems. We leave the application to reputation systems for future work.

6.3. Conclusions

We introduced graded trust change operators, which let us incrementally change how much we trust information-providing agents. We then introduced a set of trust-change postulates, and proved that every operator satisfying the postulates is defined by two values: a strengthening constant and a weakening constant. Trust change operators can be combined with DP belief revision operators to define a new class of combined report-revision operators. These operators take a sequence of reports along with an observation as input, and they simultaneously revise the agent's beliefs and modify their trust in reporting agents. The result is a single operator that combines two rational change functions to ensure both beliefs and trust are changed appropriately.

There are many directions for future work. As noted, the current implementation could be improved in terms of efficiency. The current version of HBS is a proof of concept that is only suitable for small toy problems, due to the computational complexity of belief revision.

At a theoretical level, we remark that basic trust change operators are quite restrictive in that they can only take a literal as input. In future work, we would like to extend the vocabulary of "trust formulas" to permit updates by more complex trust statements. Another direction for future research is to axiomatize the interaction properties of report revision operators. Right now, we know that the revision part satisfies the DP postulates and the trust part satisfies our new trust change postulates. But it would be useful to provide a further set of interaction postulates to describe the properties that must hold when the operators are combined. We are also interested in extending the current framework, so that it can model the interaction between knowledge-based trust and honesty-based trust.

References

[1] A. Hunter, Combined change operators for trust and belief, in: Proceedings of the 37th Australasian Joint Conference on Artificial Intelligence, 2024.

[2] A. Hunter, A. Iglesias, A tool for reasoning about trust and belief, in: Proceedings of the International Conference on Logic for Programming, Artificial Intelligence and Reasoning (LPAR), 2024, pp. 127–135.

[3] C. E. Alchourrón, P. Gärdenfors, D. Makinson, On the logic of theory change: Partial meet functions for contraction and revision, Journal of Symbolic Logic 50 (1985) 510–530.

[4] H. Katsuno, A. Mendelzon, Propositional knowledge base revision and minimal change, Artificial Intelligence 52 (1992) 263–294.

[5] A. Darwiche, J. Pearl, On the logic of iterated belief revision, Artificial Intelligence 89 (1997) 1–29.

[6] N. Schwind, S. Konieczny, R. P. Perez, Darwiche and Pearl's epistemic states are not total preorders, in: Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning (KR 2022), 2022.

[7] M. Dalal, Investigations into a theory of knowledge base revision, in: Proceedings of the National Conference on Artificial Intelligence (AAAI), 1988, pp. 475–479.

[8] R. Booth, A. Hunter, Trust as a precursor to belief revision, Journal of Artificial Intelligence Research 61 (2018) 699–722.

[9] J. Singleton, R. Booth, Who's the expert? on multi-source belief change, in: Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning (KR 2022), 2022.

[10] Y. Ammar, H. O. Ismail, Trust is all you need: From belief revision to information revision, in: Proceedings of the 17th European Conference on Logics in Artificial Intelligence (JELIA), 2021, pp. 50–65.

[11] A. Hunter, Reports, observations, and belief change, in: Proceedings of the 36th Australasian Joint Conference on Artificial Intelligence, 2023, pp. 54–65.

[12] D. Jelenc, L. H. Tamargo, S. Gottifredi, A. J. García, Credibility dynamics: A belief-revision-based trust model with pairwise comparisons, Artificial Intelligence 293 (2021) 103450.

[13] A. Salehi-Abari, T. White, Towards con-resistant trust models for distributed agent systems, in: Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI), 2009, pp. 272–277.

[14] J. Wang, Z. Yan, H. Wang, T. Li, W. Pedrycz, A survey on trust models in heterogeneous networks, IEEE Communications Surveys and Tutorials 24 (2022) 2127–2162.

[15] X. Dong, E. Gabrilovich, K. Murphy, V. Dang, W. Horn, C. Lugaresi, S. Sun, W. Zhang, Knowledge-based trust: Estimating the trustworthiness of web sources, Proceedings of the VLDB Endowment 8 (2015).

[16] F. Liu, E. Lorini, Reasoning about belief, evidence and trust in a multi-agent setting, in: PRIMA 2017: Principles and Practice of Multi-Agent Systems - 20th International Conference, volume 10621, 2017, pp. 71–89.

[17] T. D. Huynh, N. R. Jennings, N. R. Shadbolt, An integrated trust and reputation model for open multi-agent systems, Autonomous Agents and Multi-Agent Systems 13 (2006) 119–154.

[18] R. Govindaraj, P. Govindaraj, S. Chowdhury, D. Kim, D.-T. Tran, A. N. Le, A review on various applications of reputation based trust management., International Journal of Interactive Mobile Technologies 15 (2021).

[19] S. Konieczny, R. P. Péréz, Improvement operators, in: Eleventh International Conference on Principles of Knowledge Representation and Reasoning (KR'08), 2008, pp. 177–186.

[20] P. Peppas, M.-A. Williams, Parametrised difference revision, in: Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning (KR), 2018, pp. 277–286.

[21] J. Ma, M. A. Orgun, Trust management and trust theory revision, IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans 36 (2006) 451–460.

[22] Y. Ammar, H. O. Ismail, Trust is all you need: From belief revision to information revision, in: Proceedings of the 17th European Conference on Logics in Artificial Intelligence (JELIA), 2021, pp. 50–65.

[23] A. Hunter, J. Agapeyev, An efficient solver for parametrized difference revision, in: Proceedings of the Australasian Conference on Artificial Intelligence, 2019, pp. 143–152.