An incremental diagnosis algorithm of human erroneous decision making

Valentin Fouillard[1,2], Nicolas Sabouret[1], Safouan Taha[2], and Frédéric Boulanger[2]

[1] Université Paris-Saclay, CNRS, Laboratoire Interdisciplinaire des Sciences du Numérique, 91405, Orsay, France.
[2] Université Paris-Saclay, CNRS, ENS Paris-Saclay, CentraleSupélec, Laboratoire Méthodes Formelles, 91190, Gif-sur-Yvette, France.

firstname.lastname@universite-paris-saclay.fr

Abstract. This paper presents an incremental consistency-based diagnosis (CBD) algorithm that studies and provides explanations for erroneous human decision-making. Our approach relies on minimal correction sets to compute belief states that are consistent with the recorded human actions and observations. We demonstrate that our incremental algorithm is correct and complete *wrt* classical CBD. Moreover, it is capable of distinguishing between different types of human errors that cannot be captured by classical CBD.

Keywords: Diagnosis · Human errors · Belief revision.

1 Introduction

Erroneous decision making can cover many situations, from information misinterpretation to distraction or even rule-breaking [10]. Such errors led to well-documented critical accidents in nuclear safety, air transport, medical care, or aerospace engineering. Understanding what happened in such accidents is essential to preventing it from happening again and designing new systems that consider human fallibility [19]. This process is called Human Error Analysis [31].

Human error analysts try to precisely understand the situation and guess the operators' belief states that can explain their errors. For example, in the context of air transports, they use flight recorders (that record information from the flight instruments and all conversations in the cockpit) to understand what the pilots thought and why they adopted an erroneous course of action.

This reconstruction of human beliefs is done manually. However, we claim that logic-based modeling could help analysts identify possible mental states that can explain the accident. Indeed, logic-based diagnosis has been a very active field of AI since the 1980s with many famous frameworks [23,17,21]. However, applying logic-based models to human error analysis is difficult because these frameworks implement purely rational reasoning. In contrast, human factors that come into play in erroneous decision-making tend to break the basic rationality principles. Logic-based diagnosis of human error thus requires taking into account possible deviations from rationality. This is the goal of our research.

This paper is organized as follows: Sections 2 and 3 present different types of human errors that we consider in this paper, and classical logic-based models that serve as a basis for our framework. Section 4 and 5 present our model and incremental diagnosis algorithm. Sections 6 and 7 demonstrate the correctness and completeness of our algorithm. Section 8 discusses the related work, and the last section presents the perspectives of our approach.

2 Different types of human errors

While human error covers a wide range of situations [10], our research focuses on four human cognitive mechanisms which we find present in our context of application (airplane accidents): selection, memory, attention and reasoning. All these mechanisms can produce incorrect beliefs and lead to critical erroneous decisions. Two accidents will serve as examples in this paper: the Air Inter Flight 148 in 1992 (Mont Saint Odile)[3] and the Air France Flight 447 in 2009 (Rio-Paris)[4]

Information selection and preference When facing contradictory information, human beings tend to prefer the one that confirms their own beliefs or preferences, possibly falling into a confirmation bias [20]. This is one possible explanation to what happened in AF447 when the pilots ignored information that was in contradiction with their initial interpretation of the situation.

Forgetting and false memories Human beings are not omniscient and can misremember some information, forget it entirely or even build false memories [11]. This happened in AI148 when the pilots forgot that they had previously configured the system in Vertical Speed mode instead of Flight Path Angle.

Attention error Human beings have limited attention capacities. Consequently, they might miss some critical information and make erroneous decisions based on incomplete information [5]. For example in AF447, even though the stall alarm rang more than 75 times, the pilots concentrated all their attention on the overspeed information.

Reasoning error The principle of bounded rationality [25] tells us that we cannot always process information with complete and perfect reasoning, which can lead us to draw incorrect conclusions from accurate information. For example in AF447, the pilots observed a vibration on the control stick and should have

[3] https://bea.aero/uploads/tx_elydbrapports/F-GGED.pdf

[4] https://bea.aero/docspa/2009/f-cp090601/pdf/f-cp090601.pdf

concluded to a stall. However, the investigators showed that the pilots probably concluded to an overspeed situation due to erroneous reasoning.

Our research consists in taking into account such human errors in logic-based diagnosis. The following section shows how the four errors presented above relate to some classical theories and models in the field of Knowledge Representation and Reasoning.

3 Logic-based modeling of human reasoning

3.1 Information selection and preferences

Computing changes in an agent's belief base when facing new information has been studied in the 1980s as the problem of the *belief revision* [1]. This problem consists in restoring the consistency of the agent's belief base when confronted with new (and conflicting) information.

Belief revision models seem like a promising direction to capture the information preference errors presented in section 2. Indeed, [7] showed empirically that the *screened revision* operator can capture the *belief bias*, namely the tendency to judge arguments based on the plausibility of the conclusion instead of how well they support the conclusion. This revision operator can take into account a set of *screened* propositions, which are immune to revision (i.e. they cannot be among the eliminated propositions). This makes it possible to consider that, from the agent's point of view, certain beliefs cannot be abandoned because of their importance.

3.2 Forgetting and false memories

The *Frame Problem* is a seminal to the field of Reasoning about Actions and Changes. Introduced by McCarthy and Hayes [16], it can be summarized as the challenge of representing the effects of an action without explicitly representing a large number of intuitively obvious non-effects. Three different aspects appear when studying the Frame Problem: *inertia, update* and *extrapolation.*

Inertia McCarthy and Hayes showed that logic modeling requires to describe explicitly the inertia of beliefs. More formally, if φ holds in state s, then φ must still hold in the state $do(a, s)$ resulting from the execution of action a, unless a explicitly modifies φ. Several solutions were proposed to address the Frame Problem [18]. The general idea is to model the inertia of beliefs using inertia clauses of the form $\varphi_{t+1} = \varphi_t$ (depending on the representation of time and states). The difficulty is now to correctly update a belief when the agent acts upon or perceives its environment.

Update Assume that action a changes the value of φ. Applying the effects of a in the presence of a general inertia clause results in an inconsistent belief base in which both φ_{t+1} and $\neg\varphi_{t+1}$ hold. The KM theory [12] describes a set of axioms that a logical operator must verify to restore consistency when performing a belief update.

Extrapolation Agents can also observe changes in the world's state caused by other agents. The integration of the new information that conflicts with the inertia clauses is known as the *extrapolation* process. [6] showed that this extrapolation process is an instance of a belief revision on the clauses of inertia of a temporally indexed logic. A belief revision operator can then be used to perform belief extrapolation.

Relation between inertia and human errors modeling We claim that inertia and its operations (*update*, *extrapolation*) can be a way of capturing forgetting and false memories. More precisely, an error in the *update* or *extrapolation* operation can result in this type of memory error. For example in AI148, the pilots didn't perform a correct *belief update* about the autopilot configuration, which led them to believe that they were still in Vertical Speed mode, through inertia. Therefore, we define both an *extrapolation* operator and an *update* operator that support such distortions (we call them "*frame distortions*"). To implement this, we propose to rely on *circumscription.*

Circumscription Proposed by [15] as a first attempt to deal with the Frame Problem, *circumscription* consists in selecting models that minimize the number of changes in the world. While this method does not correctly solve the Frame Problem because it computes some changes or non-changes in the inertia that are not expected in a rational reasoning [9], it can be useful to compute our *frame distortions* that represent human omissions or false memories.

Moreover, [13] showed that *circumscription* is equivalent to a belief revision operator. We might thus use the same revision operator to capture information preferences, memory errors, and information preference errors!

3.3 Attention and reasoning errors

Logic-based diagnosis can refer to different approaches. Deduction [4] and abduction [22] consist in computing possible explanations given some knowledge about errors and symptoms of these errors. On the contrary, consistency-based diagnosis [23] considers only knowledge about "how the system usually works" (without any information about the possible errors). The goal is to identify deviations from the system's expected behavior.

This approach (consistency-based diagnosis or CBD) is well suited for Human Error Analysis since analysts usually don't know all possible errors and symptoms that can explain an operator's erroneous decision: all they know is which decision was expected. This is why we will use CBD in our work.

The logic-based framework proposed by [23] consists in restoring consistency (hence the name) between the description of the system's expected behavior and the observations of the system's behavior. It is composed of three elements. First, a set of logic formulas SD that describe the system. Second, a set of predicates ASS that describe the *assumables* of the form $\neg ab(c)$ which represent the fact that component c is supposed to behave normally. Third, a conjunction of predicates OBS that describe an observation of the system.

When $SD \cup ASS \cup OBS$ is inconsistent, a diagnosis Δ is a minimal set of *assumables* such that $SD \cup (ASS \setminus \Delta) \cup OBS$ is consistent. In other words, a diagnosis is a minimal set of elements that must be assumed "abnormal" to be consistent with the observations.

Our proposal is to capture attention and reasoning error by applying such a CBD. Indeed, attention errors can be seen as ignoring some information and reasoning errors as inference rules that weren't applied. Let us assume that:

- OBS contains a representation of the observed action (*i.e.* the result of the human operator's erroneous decision).
- SD contains the operator's inference rules and information available to him (*i.e* observations) that allow him to make a decision.
- $ASS = SD$ which means that we assume all available information and all inference rules of the human operator to behave normally, *i.e.* they are used and don't infer anything beyond the scope of their behavior.

The diagnosis Δ will thus contain the information and rules that the human operator ignored to make his decision and perform his action.

Belief revision and consistency-based diagnosis Several research emphasize the strong connection between a belief revision operator and a CBD [30,3]. They show that a belief revision operator can be used to compute a CBD and conversely, a CBD can be used as a belief revision operator. The benefit of CBD is that there are algorithms for calculating the diagnosis, in particular the Liffiton algorithm that we will be using, which is based on MCS (see Section 4.3).

This means that we can use a CBD algorithm to capture all four kinds of human errors: not only attention errors and reasoning errors, but also information preferences (which correspond to belief revisions) and memory errors (which correspond to frame distortions, captured by circumscription, also equivalent to a belief revision operator). The following sections present our framework.

4 Diagnosis framework

Our approach relies on restoring consistency in a set of logical formulas that represent the beliefs of the agent at a given time, the information it received, the inference rules he could use and the action he selected eventually.

4.1 Logical Modeling

Our model is based on the continuity of Reiter's situation calculus [24]: we give ourselves a starting situation S_0 (which is a set of fluents) and a set of actions (which are exactly the trace of the operator's actions that led to the accident) with S_t the situation "at time t" resulting from performing action a_t in situation

S_{t-1}. We extend this model by adding information communicated to the operator (that we call *observations*). Moreover, we allow each fluent to be time-indexed to represent beliefs of the form "in situation S_t, the agent believes that at time t', fluent φ was true".

This paper focuses on the CBD algorithm for one single time step. We thus only consider one situation and one action performed by the agent. This leads us to propose the following model.

Model We consider a set of propositions $\mathcal{P}$, all indexed temporally. $\varphi_t \in \mathcal{P}$ represents the concept "φ holds at time t". Based on these propositions, we define the language $\mathcal{L}_0$ with the following grammar:

$$\alpha ::= \varphi_t \mid \bot \mid \top \mid \neg\alpha \mid \alpha \wedge \alpha \mid \alpha \vee \alpha$$

where α is a valid formula from $\mathcal{L}_0$, $\varphi_t \in \mathcal{P}$, $\bot$ is always false and $\top$ always true.

We also consider a set $\mathcal{A}ct$ of atomic propositions representing the actions. We define the language $\mathcal{L}$ as an extension of $\mathcal{L}_0$ by adding the three following operators :

$$\phi ::= \alpha_1 \rightarrow \alpha_2 \mid \{\alpha\}act \mid act \wedge \alpha :: \varphi_{t+n}$$

with $\alpha \in \mathcal{L}_0$, $\alpha_1 \in \mathcal{L}_0$, $\alpha_2 \in \mathcal{L}_0$, $act \in \mathcal{A}ct$ and $n \in \mathbb{N}^*$.

- $\alpha_1 \rightarrow \alpha_2$ means that the agent can infer α_2 from α_1.
- $\{\alpha\}act$ mean that α is the precondition of action act. In other words, α must be true for the action to be done.
- $act \wedge \alpha :: \varphi_{t+n}$ mean that φ_{t+n} is the effect of the action act when α is true. In other words, φ_{t+n} is true if act is done and α is true.

4.2 Definition of the diagnosis problem

Let us consider the following elements:

- The belief state $B_{t-1} \in 2^{\mathcal{P}}$ which correspond to the current situation.
- A set of rules $\mathcal{R} \in 2^{\mathcal{L}}$ that the agent can use. For example, $\text{alarm}_t \rightarrow \text{stall}_t$ says that if the agent believes that there is an alarm, it should believes that he is in a stall situation.
- A set of possible observations in the environment $Obs \in 2^{\mathcal{P}}$. For example, $\text{alarm}_t \in Obs$ mean that the agent could observe an alarm.
- $a \in \mathcal{A}ct$ the action selected by the agent. For example, $a = \text{Push}$ means that the agent decided, based on his beliefs, observations and rules, to push the plane's control stick.
- In addition, we introduce the set of inertia clauses:

$$\mathfrak{K} = \{\varphi_t = \varphi_{t-1}\}_{\forall \varphi \in \mathcal{P}}$$

Given these elements, our problem is to compute a new belief state B_t that is consistent and that integrates all these elements. Formally, we apply a consistency-based diagnosis, *i.e.* we compute the minimum Δ such that :

$$B_t = ((B_{t-1} \cup \mathcal{R} \cup \mathit{Obs} \cup \mathfrak{K}) \setminus \Delta) \cup \{a\} \text{ is consistent}$$

In other word, we want to compute the set Δ of previous propositions, inference rules or possible observations that should be ignored by the agent to perform the action a.

The connection with human error diagnosis is the following: experts know the possible observations (Obs) and the action performed by the operator (a), as well as the domain rules ($\mathcal{R}$), and must find out which rules, observations, previous beliefs or inertia clauses were ignored by the operator. Note that the action is not an *assumable*: it was actually done and cannot be ignored.

4.3 Diagnosis computation

To compute Δ using CBD, we use the notion of *Minimal Correction Set* (MCS). For a given system $\Phi = \{\phi_1, \phi_2 \dots \phi_n\}$, $M \subseteq \Phi$ is a MCS of Φ if and only if $\Phi \setminus M$ is consistent and $\forall \phi_i \in M, (\Phi \setminus M) \cup \{\phi_i\}$ is inconsistent.

We use the algorithm proposed by [14] which supports *screened revision*. We have implemented this algorithm with the help of the SMT-solver Z3. The implementation of our algorithm in C# is available on a git repository[5].

We note $\mathfrak{M}\big(\Phi, \mathit{screened}\big)$ the set of MCSes, with Φ the system to be corrected, and $\mathit{screened} \subset \Phi$ the set of propositions and rules that cannot be removed by the MCS algorithm (i.e. $\mathfrak{M}\big(\Phi, \mathit{screened}\big) \cap \mathit{screened} = \emptyset$). We have our diagnostic reference Algorithm 1 defined as:

$$B_t = (B_{t-1} \cup \mathcal{R} \cup \mathit{Obs} \cup \mathfrak{K} \cup \{a\}) \setminus \Delta$$

where $\Delta \in \mathfrak{M}\big(\Phi, \mathit{screened}\big)$, $\Phi = \{B_{t-1} \cup \mathcal{R} \cup \mathit{Obs} \cup \mathfrak{K} \cup \{a\}\}$ and $\mathit{screened} = \{a\}$.

4.4 Considering different types of error

As presented in Section 3, many errors can explain an erroneous decision and can be captured by a CBD operator. However it is difficult to know which element in Δ corresponds to which error (information preferences, memory, attention or reasoning). For example, if Δ contains some element from Obs, we can't know if this is due to a preference error (selection between two contradictory pieces of information) or to an attention error. Yet this a vital information for human error analysis.

To ease the understanding of the belief states and errors of the agent, we propose to use an incremental diagnosis algorithm. The idea of this algorithm is to perform a computation of the MCSes by increment, where each of the increment is focused on a specific error. Hence we can easily find the increment at the origin of a proposal in an MCS. The following section presents this algorithm.

[5] https://gitlab.dsi.universite-paris-saclay.fr/valentin.fouillard/humandiagnosis

5 Incremental diagnosis algorithm

Our algorithm works in four steps: 1) Detection of MCSes that correspond to belief revisions, 2) MCSes related to an erroneous decision, 3) MCSes corresponding to extrapolation 4) MCSes corresponding to an update or a frame distortion. To illustrate our algorithm, we shall use the following example, which is a very simplified representation of the AF447 (Rio-Paris flight) situation:

$$\begin{array}{ll}
B_{t-1} = & \{\neg\,\text{acceleration}_{t-1}, \neg\,\text{alarm}_{t-1}, \text{buffet}_{t-1}\} \\
Obs = & \{\text{acceleration}_t, \text{alarm}_t\} \\
\mathcal{R} = & \left\{\begin{array}{l}
R^1 \equiv \text{alarm}_t \rightarrow \text{stall}_t \\
R^2 \equiv \text{acceleration}_t \rightarrow \text{overspeed}_t \\
R^3 \equiv \text{buffet}_t \rightarrow \text{stall}_t \\
R^4 \equiv \{\neg\,\text{overspeed}_t\}\,\text{Push} \\
R^5 \equiv \{\neg\,\text{stall}_t\}\,\text{Pull} \\
R^6 \equiv \text{stall}_t \wedge \text{overspeed}_t \rightarrow \bot
\end{array}\right\} \\
a = & \text{Pull}
\end{array}$$

In this situation the pilot believes that there is no sign of acceleration and no stall alarm, the control stick is vibrating (*a.k.a.* buffet). This is given in B_{t-1}. The pilot can observe both an acceleration and a stall alarm (Obs) and decides to pull the stick ($a = \text{Pull}$). The set of rules $\mathcal{R}$ represents classical pilot knowledge about stall and overspeed situations. In particular, it is expected to push the stick in case of stall, and to pull it in case of overspeed, not the contrary.

5.1 Information preference

Starting from B_{t-1}, we add only the observations Obs and the rules $\mathcal{R}$ in the belief base of the agent. This ensures that the MCS captures only inconsistencies due to the observations (and possible reasoning about these observations), not the action or the inertia clauses. If we are in a situation where the agent observes two contradictory information or an information inconsistent with his beliefs, we compute a belief revision to build all possible revisions:

$$\begin{gathered}
B_t^{rev} = \Phi \setminus M_{rev} \\
\text{with } M_{rev} \in \mathfrak{M}\big(\Phi, screened\big), \Phi = \{B_{t-1} \cup \mathcal{R} \cup Obs\}, screened = \{\emptyset\}
\end{gathered} \tag{1}$$

In the example introduced in the beginning of this section, the observations of the alarm, the buffet and the acceleration are inconsistent with each other. Therefore a possible MCS M_{rev} computed at this step is $\{\text{acceleration}_t\}$: the pilot prefers to keep the stall alarm rather the acceleration information.

Note that there are several possible MCSes and thus several B_t^{rev}. For instance, the alternative correction (ignoring the alarm instead of the acceleration) is also a possible belief state. This is true for all steps of our algorithm.

5.2 Attention and reasoning error

For each possible belief base B_t^{rev}, we introduce the action in it. Since observations inconsistencies were already corrected, we ensure that the MCSes detected at this step are related to the action (*i.e.* reasoning error). We compute the belief base B_t^{diag} resulting from this erroneous decision by:

$$\begin{gathered} B_t^{diag} = \Phi \setminus M_{diag} \\ \text{with } M_{diag} \in \mathfrak{M}\big(\Phi, \mathit{screened}\big), \Phi = \{B_t^{rev} \cup \{a\}\}, \mathit{screened} = \{a\} \end{gathered} \tag{2}$$

To illustrate this step, let's consider that the previous step computes $M_{rev} = \{\text{acceleration}_t\}$. Adding the action in the system creates an inconsistency: from R^1 and R^5, we can infer that we should not perform the *Pull* action. One of the possible MCS computed at this step is $M_{diag} = \{R^1\}$: the pilot draws an incorrect conclusion about the alarm.

5.3 Extrapolation

From each possible belief base B_t^{diag}, we introduce the inertia clauses $\mathfrak{K}$. For each proposition φ_{t-1} in B_{t-1}, $\mathfrak{K}$ contains the clause $\varphi_t = \varphi_{t-1}$. We also remove the action done by the agent from the belief base. This ensures that the inconsistencies will be related to the observations and the inertia clauses, and thus to the changes in the world that the agent has to consider. In other words, we compute an extrapolation to build:

$$\begin{gathered} B_t^{ext} = \Phi \setminus M_{ext} \\ \text{with } M_{ext} \in \mathfrak{M}\big(\Phi, \mathit{screened}\big), \Phi = \{(B_t^{diag} \setminus \{a\}) \cup \mathfrak{K}\}, \mathit{screened} = \{B_t^{diag}\} \end{gathered} \tag{3}$$

To illustrate this step, let's start from the previous corrections on our example (*i.e.* ignoring acceleration_t and R^1). By adding the inertia clauses in the logic system, we create an inconsistency on the alarm belief: we can deduce $\neg\,\text{alarm}_t$ from B_{t-1} but we have alarm_t in *Obs*. One possible MCS computed at this step is $M_{ext} = \{\text{alarm}_t = \text{alarm}_{t-1}\}$: the pilot simply updates their belief base with the new information about the alarm (and this is not an error).

5.4 Update and distortion

For each possible belief base B_t^{ext}, we re-introduce the previously removed action. Since inconsistencies related to observations and the inertia were already solved, we ensure that the MCSes detected in this step correspond either to an update (the action needs to change the inertia) or a frame distortion (the action should not impact the inertia but it is inconsistent with it), which captures a possible memory error. Indeed as we performed a circumscription (equivalent to CBD, see Section 3), both are captured. We compute the final belief base of the agent:

$$\begin{gathered} B_t = \Phi \setminus M_{dist} \\ \text{with } M_{dist} \in \mathfrak{M}\big(\Phi, \mathit{screened}\big), \Phi = \{B_t^{ext} \cup \{a\}\}, \mathit{screened} = \{a\} \end{gathered} \tag{4}$$

To illustrate this step, let's consider the previous corrections. When adding the action in the system, we have an inconsistency between the buffet and the pull action, via rules R^5 and R^3. One possible MCS for this step is $M_{dist} = \{\text{buffet}_{t-1} = \text{buffet}_t\}$: the pilot believes that the truth value of the buffet has changed between the two time steps, without any reason (which might be explained by forgetting the previous information).

5.5 Resulting explanation

Each successive correction to reach B_t from B_{t-1} leads to several possible solutions. Therefore, we can say that a possible belief state B_t is computed through a sequence x of MCS choices. We note Δ^x the union of all MCSes in the sequence x: $\Delta^x = \{M^x_{rev} \cup M^x_{diag} \cup M^x_{ext} \cup M^x_{dist}\}$ where x is a sequence computed by our incremental algorithm. There are as many possible Δ^x as correction choices at each step in the algorithm.

From this algorithm, we can easily, determine which type of error corresponds to a proposition $\varphi \in \Delta^x$, by finding the corresponding subset. In the next section, we prove that our algorithm correctly captures a consistency-based diagnosis.

6 Correctness and Completeness

In this section we consider the results of our incremental algorithm (Algorithm 2) compared to the results of the reference Algorithm 1, defined Section 4.3. Our algorithm is correct if its solutions are effective corrections of Φ and it is complete if its solutions include all the solutions provided by Algorithm 1. However even if our algorithm is correct, it can compute non-minimal corrections, contrary to Algorithm 1.

6.1 Correctness

Since Algorithm 1 returns a minimal set of ignorance to make the beliefs consistent, we need to check that each Δ^x proposed by Algorithm 2 ignores at least the same thing as one of the possible Δ given by Algorithm 1 to make the beliefs consistent. It can ignore more (this would be a non-minimal solution) but not less (this would not be sound).

Let's consider that Algorithm 1 consists in correcting the set B to make it consistent. Then, Algorithm 2 corresponds to four increments for the form: (1) add some propositions from B to a subset A of B; (2) correct A. These increments are repeated until all the propositions missing from the subset A of the first increment have been added to B. Thus, if we show the soundness and the completeness over two increments, by induction, our 4-step Algorithm 2 is also complete (we are only adding propositions to a subset to arrive at B: it is a recursive algorithm).

Let consider two increments: first in computing an MCS M_1 on a set A then computing an MCS M_2 on the set $B \setminus M_1$ where $A \subseteq B$. The ignorance which results from these two time steps is then the union of the two MCses : $M_1 \cup M_2$. On the contrary, Algorithm 1 consists in computing an MCS M on the set B.

We must prove that $M \subseteq M_1 \cup M_2$, i.e., Algorithm 2 ignores at least the same propositions as Algorithm 1. More formally, we must prove theorem (T1):

$$\begin{aligned} &A \subseteq B \wedge M_1 \in \mathfrak{M}\big(A, \emptyset\big) \wedge M_2 \in \mathfrak{M}\big(B \setminus M_1, \emptyset\big) \\ &\Rightarrow \exists M \in \mathfrak{M}\big(B, \emptyset\big),\, M \subseteq M_1 \cup M_2 \end{aligned} \tag{T1}$$

Under the premises, we can say that :

(a) $B \setminus (M_1 \cup M_2) \subseteq B$ since removing sets from B can only give a subset of B.
(b) $B \setminus (M_1 \cup M_2) \nvdash \bot$ because, by construction, M_2 makes $B \setminus M_1$ consistent (it is a MCS). Let alone $B \setminus (M_1 \cup M_2)$ can only be consistent.

We know that M is minimal, or that $B \setminus M$ is a maximal subset of B which does not imply $\bot$. Moreover, we know from (a) and (b) that $B \setminus (M_1 \cup M_2)$ is a subset of B which does not imply $\bot$. Therefore, $M_1 \cup M_2$ can only be a superset of an MCS M of B. Indeed, because M is minimal and makes B consistent, as well as because $M_1 \cup M_2$ also makes B consistent, $M_1 \cup M_2$ can only be a superset of M. Otherwise, M would not be minimal, which is a contradiction.

We can thus conclude that $\exists M \in \mathfrak{M}\big(B, \emptyset\big), M \subseteq M_1 \cup M_2$, which proves theorem $(T1)$.

6.2 Completeness

Theorem $(T1)$ subsection 6.1 tells us that the solutions found by Algorithm 2 contain at least the propositions corrected by Algorithm 1. Moreover, it tells us that the solutions found by increments are not necessarily minimal, *i.e.* increments can lead us to find solutions where the agent ignores more than necessary. Determining whether Algorithm 2 is complete is then equivalent to determining whether it computes *all* minimal solutions given by Algorithm 1. More formally, theorem (T2) states that, for any minimal solution M, there exist M_1 and M_2 obtained by Algorithm 2 such that $M_1 \cup M_2 = M$:

$$\begin{aligned} &A \subseteq B \wedge M \in \mathfrak{M}\big(B, \emptyset\big) \\ &\Rightarrow \exists M_1, M_2,\, M_1 \in \mathfrak{M}\big(A, \emptyset\big) \wedge\, M_2 \in \mathfrak{M}\big(B \setminus M_1, \emptyset\big) \wedge M = M_1 \cup M_2 \end{aligned} \tag{T2}$$

To begin with, let us note that if $B \setminus M$ is consistent with $A \subseteq B$, then not only is $A \setminus M$ consistent (there are fewer propositions) but also there is a $M' \setminus M$ such that $A \setminus M'$ is consistent (some of the propositions of M are not present in A so we can remove them). Thus, we can state that:

(a) $\forall A \subseteq B, \exists M_1 \in \mathfrak{M}\big(A, \emptyset\big)$ with $M_1 \subseteq M$.
(b) $\forall M_1, \exists M_2 \in \mathfrak{M}\big(B, \emptyset\big)$ with $M_2 \subseteq M$.

We then deduce:

(c) $M_1 \cup M_2 \subseteq M$ by (a) and (b).

(d) by (T1), $\exists M' \in \mathfrak{M}(B, \emptyset)$ such that $M_1 \cup M_2 = M'$.
(e) We thus have $M' = M_1 \cup M_2$ and $M' \subseteq M$. However since both M and M' are MCSes, by definition of minimality, $M' = M$.

Algorithm 2 is thus complete in the sense that it returns, like Algorithm 1, all possible minimal solutions to make the agent's beliefs consistent ($T2$). These proofs have been verified in Isabelle/HOL and are available on a git repository[6].

However, theorems ($T1$) and ($T2$) tell us that it also computes some non-minimal solutions which cannot be computed by Algorithm 1. The next section discusses these non-minimal solutions.

7 Discussion

From a purely logic-based diagnosis point of view in logic, the CBD defines the best solutions to explain a non-expected behavior as the minimal solutions that allow to recover the consistency.

However, the solutions obtained by our incremental algorithm are far from uninteresting. To illustrate this, let us consider another simplified representation of the AF447 situation:

$$\begin{aligned} Obs &= \{\text{alarm}_t, \text{acceleration}_t\} \\ \mathcal{R} &= \left\{ \begin{aligned} &R^a = \text{alarm}_t \to \text{stall}_t \\ &R^b = \text{acceleration}_t \to \neg\,\text{stall}_t \\ &R^c = \{\neg\,\text{stall}_t\}\,\text{Pull} \end{aligned} \right\} \\ a &= \text{Pull} \end{aligned}$$

In this example, the agent decides to pull the control stick when faced with two contradictory pieces of information. Algorithm 1 will compute the possible MCSes:

$\Delta^a = \{\text{alarm}_1\}$, $\Delta^b = \{R^a_1\}$, $\Delta^c = \{R^c_1, R^b_1\}$, $\Delta^d = \{R^c_1, \text{acceleration}_1\}$

With Algorithm 2, we obtain:

$\Delta^a = \{\text{alarm}_1\}$ $\Delta^b = \{R^a_1\}$ $\Delta^c = \{R^c_1, R^b_1\}$ $\Delta^d = \{R^c_1, \text{acceleration}_1\}$
$\Delta^e = \{\text{acceleration}_1, \text{alarm}_1\}$, $\Delta^f = \{\text{acceleration}_1, R^a_1\}$, $\Delta^g = \{R^b_1, R^a_1\}$,
$\Delta^h = \{R^b_1, \text{alarm}_1\}$

Algorithm 2 returns the same solutions as Algorithm 1 but also gives non minimal solutions Δ^e to Δ^h. These solutions explore belief revisions that go against the decision made by the agent. For example, in Δ^e and Δ^f, the agent does not take into account the acceleration information, even though it is consistent with their decision to pull the control stick.

In other words, while it is not necessary to ignore the acceleration to restore consistency, since it does not ultimately contradict the agent's decision, these corrections come from the fact that the agent had to manage an inconsistency during the revision phase. Algorithm 1 cannot explore such corrections: the revision of belief chosen by the agent is always consistent with their decision. On

[6] https://gitlab.dsi.universite-paris-saclay.fr/valentin.fouillard/incrementalcompleteness

the contrary, Algorithm 2 explores more complex behaviors where for example the agent prefers one piece of information rather than another while ignoring reasoning rules allowing him to use this information correctly. For example, we can consider that Δ^f means that the agent pays attention to the alarm instead of the acceleration, but considers that the alarm is faulty and that it does not indicate a stall.

8 Related work

Finding explanations for a situation through logic modeling, namely diagnosis, has thrived since the 1980s [23]. However to our knowledge, none of the proposed models are applied in the context of erroneous human decision-making. While several works proposed solutions for diagnosing dynamic systems (*i.e.* taking actions and changes into account) [17,27], all assume that the solution must comply with the frame problem's inertia. However, unlike logic-based models, human beings sometime forget information, which conflicts with the frame inertia, namely, a *frame distortion*. One of our contributions is to take into account such distortions in the logical model when diagnosing human errors.

Research in AI has attempted to model human reasoning errors or, more generally, human reasoning limitations, for predictive purposes in simulation. For instance, [29] uses a finite state automaton to simulate opinion dynamics regarding vaccination. Their model supports the decision of non-vaccination even when the rational information should lead the agent to accept it. In a different context, [2] uses the BDI paradigm to implement probabilistic functions that lead to erroneous beliefs in reaction to bushfires. All these models propose valuable solutions to simulate human decisions, but they cannot be used for diagnosis purposes in general cases.

Another approach for capturing false beliefs and human reasoning errors is to get rid of *logical omniscience*, *i.e.* the capacity to infer all the consequences of a belief φ. For example, [26] proposes a framework based on the *impossible world* (*i.e.* worlds that are not closed under logical consequences) to simulate reasoning errors. They associate *resources consumption* to each reasoning rule, which limits the applicability of lengthy inferences. However, the computation of all *impossible* worlds to select the most plausible one requires exponential computation power. Moreover, their model does not consider actions and changes.

All these approaches give interesting, yet partial, solutions to our problem: they neither handle the frame distortion problem, nor do they work for diagnosis purposes. Our framework combines these ideas to model erroneous decision making and to compute diagnosis that take human errors into account.

9 Conclusion and perspectives

We proposed an *incremental consistency-based diagnosis* to compute belief states that could explain erroneous decision making of a human operator. This Algorithm is based on the computation of Minimal Correction Sets. The resulting belief states are consistent with the observations and actions performed by the operator and take into account four kinds of human errors: information preference, memory, attention and reasoning errors. While this paper presented the algorithm on one single time step, it has been implemented and works with several successive states, thus building a tree of all possible successive belief states of the agent.

In its current version, our model computes all possible scenarios, but it does not identify the most "plausible" ones. For example, the complete model of the Rio-Paris crash returns over 9000 scenarios, which is overwhelming for a human expert. To address this limitation, we propose filtering this set to extract classical human errors, identified in the literature as cognitive biases [28]. To this goal, [8] proposed some logic-based patterns to identify cognitive biases in accident scenarios. We propose to include such patterns in our model and extend them to capture other cognitive biases, so as to reduce the set of possible scenarios for the experts.

Abbreviation definitions

– CBD: Consistency Based diagnosis, a logical framework that restores consistency (hence the name) between the description of the system's expected behavior and the observations of the system's behavior (see Section 3.3 for a complete definition).
– MCS: A set of minimal corrections to be removed from a system to restore coherence (see Section 4.3 for a complete definition).

References

1. Alchourrón, C.E., Gärdenfors, P., Makinson, D.: On the logic of theory change: Partial meet contraction and revision functions. The journal of symbolic logic **50**(2), 510–530 (1985)
2. Arnaud, M., Adam, C., Dugdale, J.: The role of cognitive biases in reactions to bushfires. In: ISCRAM. Albi, France (May 2017)
3. Boutilier, C., Beche, V.: Abduction as belief revision. Artificial intelligence **77**(1), 43–94 (1995)
4. Buchanan, B., Shortliffe, E.: Rule-based Expert System – The MYCIN Experiments of the Stanford Heuristic Programming Project (01 1984)
5. Cisler, J.M., Koster, E.H.: Mechanisms of attentional biases towards threat in anxiety disorders: An integrative review. Clinical psychology review **30**(2), 203–216 (2010)
6. Dupin de Saint-Cyr, F., Lang, J.: Belief extrapolation (or how to reason about observations and unpredicted change). Artificial Intelligence **175**(2), 760–790 (2011)
7. Dutilh Novaes, C., Veluwenkamp, H.: Reasoning biases, non-monotonic logics and belief revision. Theoria **83** (12 2016)
8. Fouillard, V., Sabouret, N., Taha, S., Boulanger, F.: Catching cognitive biases in an erroneous decision making process. IEEE International Conference on Systems, Man and Cybernetics (SMC) (2021)
9. Hanks, S., McDermott, D.: Nonmonotonic logic and temporal projection. Artificial intelligence **33**(3), 379–412 (1987)

10. Hollnagel, E.: Cognitive reliability and error analysis method (CREAM). Elsevier (1998)
11. Kaplan, R.L., Van Damme, I., Levine, L.J., Loftus, E.F.: Emotion and false memory. Emotion Review **8**(1), 8–13 (2016)
12. Katsuno, H., Mendelzon, A.O.: On the difference between updating a knowledge base and revising it, p. 183–203. Cambridge Tracts in Theoretical Computer Science, Cambridge University Press (1992)
13. Liberatore, P., Schaerf, M.: Reducing belief revision to circumscription (and vice versa). Artificial intelligence **93**(1-2), 261–296 (1997)
14. Liffiton, M.H., Sakallah, K.A.: Algorithms for computing minimal unsatisfiable subsets of constraints. Journal of Automated Reasoning **40**(1), 1–33 (2008)
15. McCarthy, J.: Applications of circumscription to formalizing common-sense knowledge. Artificial intelligence **28**(1), 89–116 (1986)
16. McCarthy, J., Hayes, P.J.: Some philosophical problems from the standpoint of artificial intelligence. Machine Intelligence p. 463–502 (1969)
17. Mcllraith, S.A.: Explanatory diagnosis: Conjecturing actions to explain observations. In: Logical Foundations for Cognitive Agents, pp. 155–172. Springer (1999)
18. Morgenstern, L.: The problem with solutions to the frame problem. The Robot's Dilemma Revisited: The Frame Problem in Artificial Intelligence. Ablex Publishing Co., Norwood, New Jersey pp. 99–133 (1996)
19. Murata, A., Nakamura, T., Karwowski, W.: Influence of cognitive biases in distorting decision making and leading to critical unfavorable incidents. Safety **1**(1), 44–58 (2015)
20. Nickerson, R.S.: Confirmation bias: A ubiquitous phenomenon in many guises. Review of general psychology **2**(2), 175–220 (1998)
21. Paul, G.: Approaches to abductive reasoning: an overview. Artificial intelligence review **7**(2), 109–152 (1993)
22. Poole, D.: Representing diagnosis knowledge. Annals of Mathematics and Artificial Intelligence **11**(1), 33–50 (1994)
23. Reiter, R.: A theory of diagnosis from first principles. Artificial Intelligence **32**(1), 57 – 95 (1987)
24. Reiter, R.: The frame problem in the situation calculus: A simple solution (sometimes) and a completeness result for goal regression. In: Artificial and Mathematical Theory of Computation. pp. 359–380. Citeseer (1991)
25. Simon, H.A.: Bounded rationality. In: Utility and probability, pp. 15–18. Springer (1990)
26. Solaki, A., Berto, F., Smets, S.: The logic of fast and slow thinking. Erkenntnis **86**(3), 733–762 (2021)
27. Thielscher, M.: A theory of dynamic diagnosis. Electronic Transactions on Artificial Intelligence **1**(4), 73–104 (1997)
28. Tversky, A., Kahneman, D.: Judgment under uncertainty: Heuristics and biases. Science **185**(4157), 1124–1131 (1974)
29. Voinson, M., Billiard, S., Alvergne, A.: Beyond rational decision-making: modelling the influence of cognitive biases on the dynamics of vaccination coverage. PloS one **10**(11) (2015)
30. Wassermann, R.: An algorithm for belief revision. In: Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning. pp. 345–352 (2000)
31. Wiegmann, D.A., Shappell, S.A.: Human error analysis of commercial aviation accidents using the human factors analysis and classification system (hfacs). Tech. rep., United States. Office of Aviation Medicine (2001)