Le problème du décor revisité : un modèle logique pour le diagnostic d'erreurs humaines

Valentin Fouillard, Nicolas Sabouret, Safouan Taha, Frédéric Boulanger

Le problème du décor revisité : un modèle logique pour le diagnostic d'erreurs humaines

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

[1] Université Paris-Saclay, CNRS, LISN, 91405, Orsay, France
[2] Université Paris-Saclay, CNRS, ENS Paris-Saclay, CentraleSupélec, LMF, 91190, Gif-sur-Yvette, France

prenom.nom@universite-paris-saclay.fr

Résumé

Cet article présente un modèle logique pour étudier et diagnostiquer les erreurs de décision commises par des humains dans des systèmes critiques comme les avions. Notre approche s'appuie sur la révision de croyance pour calculer les états mentaux possibles de l'opérateur à partir des observations et actions enregistrées. Nous montrons que notre modèle capture différentes formes d'incohérences qui correspondent à des erreurs de raisonnement possibles de l'opérateur. Certaines de ces erreurs sont liées à un problème de distorsion du décor*, c'est-à-dire à des déviations du comportement attendu selon le principe d'inertie connu sous le nom de « problème du décor ».*

Mots-clés

Révision de croyance, diagnostic, erreur humaine

Abstract

This paper presents a logic-based framework to study and put forward explanations for erroneous decision-making by human operators. Our approach is based on belief revision to compute the operator's possible mental states from recorded actions and observations. We demonstrate how our model captures different forms of inconsistencies in the logic model, which correspond to possible reasoning errors by the operator. Some of these errors can be related to a Frame Distortion Problem*, namely a deviation from the expected behaviour under the inertia principle known as the Frame Problem.*

Keywords

Belief revision, diagnosis, human error

1 Introduction

L'erreur est humaine, c'est pourquoi nous la retrouvons dans de nombreuses situations : erreur d'interprétation, distraction, violation des règles, *etc* [11]. Ces erreurs peuvent conduire à des accidents graves dans des domaines critiques comme la sûreté nucléaire, le transport aérien, la médecine ou l'ingénierie spatiale. Comprendre ce qui s'est passé dans le cas d'un incident ou d'un accident est donc essentiel pour éviter qu'une telle situation se répète et pour concevoir de nouveaux systèmes qui prennent en compte ces facteurs humains [17]. La littérature du domaine parle d'« analyse de l'erreur humaine » (*human error analysis* en anglais).

Pour mener à bien cette analyse de l'erreur humaine, les experts doivent comprendre précisément la situation et poser des hypothèses sur les états mentaux des opérateurs humains pour expliquer leurs erreurs. Prenons l'exemple du transport aérien. Dans ce domaine, les enregistreurs de vol (les « boites noires ») permettent aux experts d'avoir accès à l'ensemble des informations des instruments de vol, à l'état de l'appareil et aux conversations dans le cockpit. Ils utilisent ces informations pour comprendre ce que les pilotes ont compris de la situation et pour expliquer pourquoi ils ont effectué des actions erronées.

La thèse que nous défendons dans cet article est que la modélisation formelle en logique constitue un fondement solide pour développer des outils semi-automatiques d'analyse de l'erreur humaine. En effet, le problème de l'identification des erreurs de décision dans une séquence d'actions est similaire à un problème de diagnostic, un domaine de l'IA très actif depuis les années 80 et pour lequel de nombreux modèles logiques ont été proposés [19, 16, 18]. Cependant l'utilisation de ces modèles logiques pour l'analyse de l'erreur humaine est difficile car ces modèles implémentent des raisonnements complètement rationnels, comme le principe d'inertie des croyances lié au problème du décor de McCarthy et Hayes [15]. Au contraire, les facteurs humains qui rentrent en jeu dans les erreurs de prise de décision ont tendance à casser ces principes. Ainsi le diagnostic logique des erreurs humaines nécessite de revoir ces principes rationnels pour permettre des déviations des comportmeents attendues. C'est ce que nous nommons le *problème de la distorsion du décor*.

Cet article propose un modèle logique qui supporte la distorsion du décor pour construire une série d'état mentaux menant à un raisonnement erroné. Il est organisé de la manière suivante. La section 2 introduit la problématique de recherche. La section 3 décrit notre modèle qui permet de représenter la situation d'accident et de calculer les états mentaux successifs en prenant en compte l'inertie des croyances. La section 4 montre, à travers différentes incohérences, dont la « distorsion du décor » que nous avons men-

tionnée ci-dessus, comment nous pouvons diagnostiquer les erreurs humaines dans notre modèle. Enfin, les deux dernières sections abordent les travaux connexes (section 5) et des perspectives de notre modèle (section 6)

2 Définition du problème

Notre objectif est d'utiliser le diagnostic pour retrouver les séquences d'états mentaux d'un opérateur, représenté formellement comme un agent, qui permettent d'expliquer un accident. Pour cela, nous utilisons le *consistency based diagnosis* présenté dans la section suivante.

2.1 Consistency-based diagnosis

Le diagnostic consiste à déterminer quels composants d'un système ont causé un comportement anormal sachant un ensemble d'observations. Dans ce contexte, le *consistency based diagnosis* proposé par Reiter [19] est un modèle logique pour le diagnostic automatique qui consiste à retrouver la cohérence entre la description du comportement du système et les observations. Ce modèle comporte trois ensembles logiques :

- SD un ensemble de formules logiques qui décrit le système,
- ASS un ensemble de prédicats qui décrit les *hypothèses* de la forme $\neg ab(c)$, *i.e* le composant c est supposé se comporter normalement,
- OBS une conjonction de prédicats qui décrit une observation du système.

Quand $SD \cup ASS \cup OBS$ est incohérent, un diagnostic Δ est un ensemble minimal d'*hypothèses* tel que $SD \cup (ASS \setminus \Delta) \cup OBS$ est cohérent. En d'autres termes, un diagnostic est un ensemble minimal d'éléments dont on doit supposer qu'ils ont un comportement anormal pour retrouver la cohérence avec les observations.

Notre modèle d'analyse d'erreur humaine est basé sur cette approche de diagnostic par recherche de la cohérence (*consistency-based diagnosis*) avec quelques adaptations que nous présentons dans les sections suivantes. La principale modification provient de ce que nous voulons étudier les états mentaux, qui peuvent différer de la vérité du monde. C'est pourquoi nous considérons qu'un état mental est composés de différents éléments (croyances, règles de raisonnement, observations...) qui sont par défaut ceux qui correspondent à la réalité du monde physique. Toutefois, l'état mental d'un agent humain peut être altéré par différents facteurs (erreurs de raisonnement, occultation d'informations), et nous considérons qu'un diagnostic d'erreur humaine est un ensemble minimal d'éléments qu'il faut retirer de l'état mental de l'agent pour retrouver la cohérence avec les actions qu'il a effectuées. Nous présentons dans la section 3 ce modèle et l'algorithme de diagnostic qui va avec.

2.2 Le problème du décor

Le problème du décor (*frame problem* en anglais) a été introduit par John McCarthy et Patrick Hayes [15] en 1969. C'est un problème très connu en modélisation logique qui peut être décrit comme la difficulté à représenter les effets d'une action en logique sans avoir à représenter explicitement tous les non-effets évidents [22]. Pour illustrer ce propos, considérons une formalisation simple, en logique du premier ordre, du *Yale Shooting Problem* (YSP) proposé par Hanks et McDermott [10] :

$$\begin{array}{lcl} \textit{init} & = & \mathrm{alive}(0), \neg\,\mathrm{loaded}(0) \\ \textit{actions} & = & \mathrm{Load}(0), \mathrm{Wait}(1), \mathrm{Shoot}(2) \\ \textit{rules} & = & \mathrm{Load}(0) \rightarrow \mathrm{loaded}(1), \\ & & (\mathrm{Shoot}(2) \wedge \mathrm{loaded}(2)) \rightarrow \neg\,\mathrm{alive}(3) \end{array}$$

0,1,2,3 correspondent à des instants successifs. Les prédicats dans *init* décrivent la situation initiale : la dinde est vivante et le pistolet est chargé. Les prédicats dans *actions* modélisent le déroulement des actions (l'agent charge le pistolet, attend et tire) et les prédicats dans *rules* modélisent la physique du monde : tirer sur la dinde avec un pistolet chargé la tue.

Avec cette modélisation, le bon sens nous dit que $\neg\,\mathrm{alive}(3)$ est vrai. Toutefois, les effets de l'action *Wait* ne sont pas définis dans le modèle et ne peuvent être déduit sans des règles logiques qui indiquent que $\mathrm{loaded}(t+1) = \mathrm{loaded}(t)$ quand rien d'autre ne dit le contraire. Écrire ces règles manuellement pour chaque fait initial et toutes les actions possibles est une charge trop lourde dans le cas général. C'est ce qu'on appelle le problème du décor.

Une première solution à ce problème a été proposée par McCarthy [14] en utilisant la *circumscription* et en sélectionnant les modèles (c'est-à-dire les ensembles de propositions) qui minimisent le nombre de changements dans le monde. Toutefois, cela ne marche pas sur le YSP où deux solutions minimales peuvent être trouvées : la solution conforme à notre intuition, dans laquelle $\mathrm{loaded}(2) \wedge \neg\,\mathrm{alive}(3)$ est vrai, mais aussi une solution contre-intuitive $\neg\,\mathrm{loaded}(2) \wedge \mathrm{alive}(3)$ dans laquelle le pistolet se décharge magiquement pendant l'action *Wait*.

D'autres solutions on été proposées au fil du temps pour résoudre ce problème, comme les axiomes de Reiter sur les états successeurs [20] ou l'occlusion de Sandewall [21]. Toutes les solutions consistent à ajouter des éléments pour distinguer les prédicats ou propositions qui doivent sortir de l'inertie des clauses du décor. En d'autres termes, ils permettent une description de comment le monde change ou ne change pas sachant les actions occurrentes.

Comme tout autre modèle logique d'actions et de changements, notre modèle d'analyse d'erreur humaine fait face au problème du décor. Mais en plus de cela, nous devons faire face à un autre problème : celui de la *distorstion du décor* que nous illustrons dans la prochaine section en utilisant une version remaniée du YSP.

2.3 Le problème de la distorsion du décor

Supposons, dans le YSP, que l'agent ne voulait pas tuer la dinde. Nous sommes face à une situation d'erreur de prise de décision et nous voulons comprendre pourquoi l'agent a finalement appuyé sur la détente. Pour modéliser cette situation, nous ajoutons un désir dans la liste des prédicats :

$$\textit{desires} \quad = \quad \mathit{desire}(\mathrm{alive}(3))$$

Cela indique que l'agent veut que $\mathrm{alive}(3)$ soit vrai. Du point de vue d'un observateur extérieur, en faisant l'hypothèse que le problème du décor est résolu dans le modèle (i.e que l'agent devrait croire à $t = 2$, que le pistolet reste chargé), il n'y a aucune raison que l'agent choisisse de tirer ($\mathrm{Shoot}(2)$). Mais il l'a fait. La question est alors : qu'est ce qui peut expliquer ce comportement apparemment irrationnel ?

Une explication possible serait que l'agent a oublié que le pistolet était chargé. Cela implique que l'agent croyait que le pistolet était chargé au pas de temps 1, mais plus au pas de temps 2 : $\mathrm{loaded}(1) \wedge \neg\,\mathrm{loaded}(2)$. Pourtant cela n'est pas attendu avec l'inertie des croyances de l'agent, bien que ce soit une explication plausible du comportement irrationnel de l'agent.

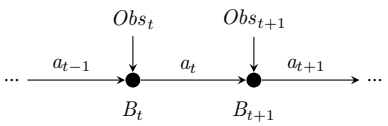
Cet exemple montre que le diagnostic des erreurs humaines selon un modèle logique nécessite de capturer des déviations à tous les niveaux, y compris dans le principe d'inertie des croyances.

Cette situation n'est pas une simple construction de l'esprit. Considérons par exemple le crash du Mont-Saint-Odile [3]. Les pilotes ont effectué une erreur de programmation du pilote automatique avec une vitesse verticale trop grande parce qu'ils avaient oublié qu'ils avaient précédemment configuré le système en mode Vertical Speed (vitesse de descente) au lieu de Flight Path Angle (angle de descente). Les humains peuvent réellement oublier ce qu'ils avaient fait précédemment et ainsi aller à l'encontre du principe d'inertie des croyances. Concrètement, la base de croyance de l'agent a été modifiée entre les deux pas de temps, sans aucune raison !

Ce que nous appelons le *problème de la distorsion du décor* consiste à trouver un diagnostic par recherche de la cohérence (*consistency based diagnosis*) à une situation qui considère des déviations possibles de l'inertie du décor. Dans la prochaine section, nous présentons notre modèle logique qui permet de réaliser ce type de diagnostic.

3 Un modèle logique pour l'analyse des erreurs humaines

Notre approche repose sur la modélisation des actions comme des transitions entre les états mentaux de l'agent, de manière similaire au *calcul des situations* [20], à ceci près que nous modélisons les changements de l'état mental et non du monde lui-même. Nous ajoutons également les observations à chaque pas de temps, qui correspondent aux nouvelles informations reçues par l'agent depuis le monde.



3.1 Modélisation de situation

Nous décrivons les croyances et actions de l'agent par un ensemble de propositions $p_t \in \mathcal{P}$. Chaque proposition p est indexée par un pas de temps t. Par exemple :

$\neg\,\mathrm{alive}_3$	$\rightarrow$	L'agent croit que la dinde est morte à $t = 3$
$\mathrm{Load}_2 \in \mathcal{A}ct$	$\rightarrow$	l'agent charge le pistolet à $t = 2$

L'utilisation d'une valeur d'indice temporel t' égale, plus grande ou plus petite que le pas de temps courant t nous permet ainsi de représenter des croyances à propos du présent, du passé ou du futur.

Nous définissons l'ensemble des croyances initiales $\mathcal{I}nit$ qui représente les croyances des agents au pas de temps $t = 0$. Par exemple $\mathrm{alive}_0 \in \mathcal{I}nit$ indique que l'agent croit initialement que la dinde est vivante. Nous considérons aussi les ensembles de prédicats $Obs_0, \ldots, Obs_n$, chaque Obs_t correspond à des observations possibles au pas de temps t. Par exemple, si $\mathrm{loaded}_1 \in Obs_1$, alors il est possible que l'agent observe que le pistolet est chargé au pas de temps 1. Enfin, nous considérons la liste $\mathcal{T} = [a_0, \ldots, a_n]$ avec $a_i \in \mathcal{A}ct$, qui représente la liste des actions effectuées par l'agent (une action par pas de temps). Par exemple, $\mathrm{Load}_0 \in \mathcal{T}$ indique le fait que l'agent a chargé le pistolet au pas de temps 0.

Nous utilisons des formules logiques pour modéliser le raisonnement et le comportement de l'agent. Nous notons $\mathcal{R}$ l'ensemble de toutes les règles que l'agent peut utiliser. Chaque règle $R_i \in \mathcal{R}$ obéit à la grammaire suivante :

$$R_i ::= \varphi \mid [\varphi]\, a \mid a :: \varphi \qquad \varphi \in \mathfrak{F}(\mathcal{P}, \wedge, \neg),\ \ a \in \mathcal{A}ct$$

où φ est une formule logique propositionnelle utilisant des propositions atomiques de $\mathcal{P}$ ainsi que tous les connecteurs logiques classiques ($\wedge, \vee, \rightarrow$, etc). $[\varphi]a$ indique que la pré-condition φ doit être vraie pour effectuer l'action a. Enfin, $a :: \varphi$ indique que la post-condition φ est vraie après que a a été effectué. L'indice temporel t dans une règle est toujours une variable libre. Par exemple :

$$R_1 \equiv \mathrm{Load}_t :: \mathrm{loaded}_{t+1}$$

modélise l'effet de l'action « charger le pistolet » à n'importe quel pas de temps t.

Pour notre algorithme d'analyse de l'erreur humaine qui sera présenté dans la prochaine section, nous définissons deux fonctions :

- $prec : \mathcal{A}ct \longrightarrow \mathcal{R}$ qui retourne l'ensemble des règles R_i qui définissent les pré-conditions d'une action donnée.
- $effect : \mathcal{A}ct \longrightarrow 2^{\mathcal{P}}$ qui retourne l'ensemble des propositions qui apparaissent dans les post-conditions (i.e après le séparateur :: dans une règle) d'une action donnée.

Nous modélisons les buts de l'opérateur (les désirs qui doivent être satisfaits à chaque pas de temps) par un ensemble $\mathcal{D}$ de littéraux négatifs ou positifs avec un indice temporel libre. Par exemple, $\mathrm{alive}_t \in \mathcal{D}$ indique que l'agent veut que la dinde soit en vie à chaque pas de temps. Ainsi, chaque situation dans laquelle $\neg\,\mathrm{alive}_t$ peut être inféré sera considérée comme incohérente du point de vue de l'agent.

3.2 États mentaux

À partir de la modélisation des situations présentée ci-dessus, notre objectif est de construire un diagnostic sous la forme d'une suite d'états mentaux $B_0 \dots B_n$ dans laquelle chaque état mental B_t est un ensemble de propositions dans $\mathcal{P}$ et de règles dans $\mathcal{R}$ qui décrivent les croyances, observations, désirs et règles de raisonnement de l'agent pour le pas de temps t.
L'état mental initial est :

$$B_0 = \mathcal{I}nit \cup \mathcal{R} \cup \mathcal{D}$$

Chaque état successif B_t est construit à partir de l'état précédent B_{t-1} en ajoutant les observations Obs_t et l'action effectuée a_t :

$$B_t = B_{t-1} \cup Obs_t \cup \{a_t\}$$

Nous notons $B_t \vdash \varphi$ pour indiquer que φ est une conséquence de B_t dans la logique propositionnelle classique. Par exemple :

$$\frac{B_t \vdash \psi \to \varphi, \quad B_t \vdash \psi}{B_t \vdash \varphi} \qquad \frac{a_t \in B_t, \quad [\varphi]\ a_t \in B_t}{B_t \vdash \varphi}$$

Comme nous nous reposons sur la logique propositionnelle, nous utilisons un solveur SAT pour calculer la valeur de vérité des propositions. Concrètement, $B_t \vdash \varphi$ si et seulement si $B_t \wedge \neg\varphi$ est insatisfaisable.

Nous disons que B_t est incohérent quand celui-ci est insatisfaisable, ce qui est équivalent à $B_t \vdash \bot$. L'objectif de notre modèle de diagnostic est de « réparer » ces états de croyances incohérents.

3.2.1 Les prédicats *known*

Une première difficulté que nous devons résoudre est la gestion des propositions qui ne sont pas connues par l'agent. En effet, puisque que nous gérons des croyances et non des faits, une proposition φ est *connue* par un agent au pas de temps t si et seulement si $B_t \vdash \varphi$ ou $B_t \vdash \neg\varphi$. Dans le cas contraire (φ n'apparaît pas dans l'état mental), nous disons qu'elle est inconnue. Le raisonnement de l'agent peut ne pas être le même selon qu'une proposition est connue ou non.
Considérons par exemple l'état mental B_t comprenant les règles et observations suivantes :

$$\mathcal{R} \equiv \left\{ \begin{array}{lcl} R_1 & \equiv & \varphi \to \neg\psi \\ R_2 & \equiv & \neg\varphi \to \neg\gamma \end{array} \right\} \qquad Obs_t \equiv \{\gamma, \psi\}$$

Le comportement que nous souhaitons dans notre modèle est le suivant :

— Si φ est connu dans B_t, φ doit être considéré comme vrai ou faux par l'agent et, par conséquent, $\neg\psi$ ou $\neg\gamma$ doit être inféré.
Cela mène à une incohérence avec Obs_t (et donc à la nécessité d'une correction, qui est l'objet de notre travail de diagnostic).

— À l'inverse, si φ est inconnu, le comportement attendu est qu'il n'y ait pas d'incohérence : l'agent ne sait rien sur φ, donc il ne peut rien en déduire et donc son état mental n'est pas contradictoire.
Malheureusement, le solveur SAT peut inférer φ et $\neg\varphi$ par les formules ci-dessus (il cherche en effet les valeurs possible pour nier la formule passée en paramètre) et il renvoie donc que B_t est incohérent.

Pour implémenter le comportement souhaité (pas d'incohérence trouvée quand φ est inconnu), nous introduisons les prédicats $known$ dans notre modèle. Notre objectif est de forcer le modèle à appliquer les règles de raisonnement uniquement sur les propositions connues. Concrètement, pour chaque proposition φ, $known_\varphi$ indique que φ est connu par l'agent. Chaque proposition φ ou $\neg\varphi$ qui apparaît dans une règle $R_k \in \mathcal{R}$ est transformée respectivement en $\varphi \wedge known_\varphi$ ou $\neg\varphi \wedge known_\varphi$. Ainsi l'exemple précédent devient :

$$\begin{array}{lcl} R_1 & \equiv & (\varphi \wedge known_\varphi) \to (\neg\psi \wedge known_\psi) \\ R_2 & \equiv & (\neg\varphi \wedge known_\varphi) \to (\neg\gamma \wedge known_\gamma) \end{array}$$

En conséquence, une proposition est forcée à connu ($known$) seulement si c'est une conséquence d'une règle avec une prémisse qui est $known$ et vraie.
Les croyances initiales dans $\mathcal{I}nit$, les observations et les actions à chaque pas de temps t sont forcées à être $known$ dans B_t afin de déclencher les règles de raisonnement. Par inférence, toutes les croyances dérivées par de telles règles de raisonnement sont mises à $known$ aussi. Les autres prédicats restent inconnus de l'agent tout au long du processus, sauf s'ils peuvent être déduits.

3.3 Inertie du modèle : le problème du décor

Le Problème du Décor nous dit que tout ce que l'agent connaît (ce qui est défini par les prédicats $known$ à $t-1$) doit être aussi connu à t, sauf si un changement est induit par les observations ou les effets d'une action. Par exemple, si $\text{alive}_0 \in B_0$, nous attendons que $\text{alive}_1 \in B_1$ sauf indication contraire.
Pour résoudre ce problème, nous introduisons deux prédicats qui forcent une proposition à garder la même valeur et le même état de connaissance :

— $keep^{(v)}$ pour maintenir la valeur de vérité d'une proposition φ :

$$keep^{(v)}_{\varphi_t} \equiv (\varphi_t \longleftrightarrow \varphi_{t-1})$$

— $keep^{(k)}$ pour maintenir l'état de connaissance d'une proposition φ :

$$keep^{(k)}_{\varphi_t} \equiv (known_{\varphi_t} \longleftrightarrow known_{\varphi_{t-1}})$$

Nous définissons les propositions $keep^{(k)}_{\varphi_t}$ et $keep^{(v)}_{\varphi_t}$ pour chaque prédicat qui apparaît dans les croyances initiales, les observations et les effets des actions. Nous les appelons *prédicats primitifs* et notons $\mathfrak{P}_t$ l'ensemble des nouvelles propositions primitives qui doivent être maintenues pour le

pas de temps t :

$$\begin{aligned}\mathfrak{P}_1 &= \mathcal{I}nit \\ \mathfrak{P}_{t>1} &= Obs_{t-1} \cup \mathit{effect}(a_{t-1})\end{aligned}$$

Les propositions *keep* qui doivent être maintenues au pas de temps t accumulent les propositions *keep* précédentes du pas de temps $t-1$ ainsi que les nouvelles. Nous définissons les ensembles $\mathfrak{K}_t^{(v)}$ et $\mathfrak{K}_t^{(k)}$ qui contiennent respectivement toutes les propositions $keep^{(v)}$ et $keep^{(k)}$ qui doivent être dans la base de croyance pour le pas de temps t :

$$\begin{aligned}\mathfrak{K}_t^{(v)} &= \mathfrak{K}_{t-1}^{(v)} \cup \bigcup_{\varphi \in \mathfrak{P}_t} \{keep_\varphi^{(v)}\} \\ \mathfrak{K}_t^{(k)} &= \mathfrak{K}_{t-1}^{(k)} \cup \bigcup_{\varphi \in \mathfrak{P}_t} \{keep_\varphi^{(k)}\}\end{aligned}$$

Ajouter ces propositions *keep* dans l'état mental nous permet de traiter le problème du décor. À partir des prédicats primitifs et des règles de l'agent, nous pouvons retrouver toutes les autres propositions par inférence. La prochaine section montre comment les actions et changements peuvent modifier ces règles *keep*.

Pour faciliter la lecture, les prédicats *known* et les propositions *keep* sont omises dans les prochains exemples.

3.4 La révision de croyance dans un agent rationnel

Notre premier objectif est de modéliser un agent rationnel qui reçoit des observations et effectue des actions. En d'autres termes, chaque B_t successif doit être cohérent afin de représenter le fait que l'agent *avait une bonne raison d'effectuer l'action* sachant ce qu'il pouvait observer et croire. Toutefois, deux cas d'incohérence peuvent apparaître dans le modèle :

(1) Une observation dans Obs_t n'est pas cohérente avec l'état mental précédent B_{t-1}

(2) Une action a_t n'est pas cohérente avec les observations, désirs et l'état mental précédent

Ce sont deux problèmes distincts. Résoudre (1) revient à faire une *révision de croyance* [8]. Le but est de déterminer quelle croyance ignorer (celle à retirer du système) face à des informations contradictoires. Ce problème a été étudié par [1] et a résulté en la définition d'AGM, un ensemble d'axiomes qui caractérise un opérateur de révision de croyance *minimal*. L'idée derrière cette minimalité est que les agents rationnels ont tendance à garder leurs anciennes croyances.

Résoudre (2) correspond à trouver un *consistency based-diagnosis* que nous avons présenté dans la sous-section 2.1 (*i.e* le comportement de l'agent n'est pas celui attendu). Toutefois, Wassermann montre qu'un opérateur de révision de croyance peut être utilisé pour effectuer une *consistency based-diagnosis* et vice versa [27] car ces deux méthodes recherchent une solution minimale.

Pour cette raison, nous proposons d'utiliser le *Minimal Correction Set* (MCS) pour implémenter l'opérateur de révision de croyance qui permet de résoudre les deux problèmes.

Pour un système $\Phi = \{\phi_1, \phi_2 \dots \phi_n\}$ donné, $M \subseteq \Phi$ est un MCS de Φ ssi :

— $\Phi \setminus M$ est cohérent

— $\forall \phi_i \in M, (\Phi \setminus M) \cup \{\phi_i\}$ est incohérent

Pour calculer le MCS nous utilisons l'algorithme de [13] que nous avons implémenté avec le SMT-solver z3 [5]. Nous notons $\mathfrak{M}\big(\Phi, \mathit{shielded}\big)$ la sortie de cet algorithme. Φ est l'ensemble qui doit être corrigé, et $\mathit{shielded} \subset \Phi$ est un ensemble de propositions et règles qui ne peuvent être retirées du système par l'algorithme de MCS (c.-à-d. $\mathfrak{M}\big(\Phi, \mathit{shielded}\big) \cap \mathit{shielded} = \emptyset$). La prochaine section montre comment nous utilisons $\mathfrak{M}$ pour l'analyse des erreurs humaines dans notre algorithme.

Il faut souligner que nous n'obtenons pas nécessairement un MCS unique pour un Φ donné. En d'autres termes, il existe de multiples solutions pour retrouver la cohérence dans l'état mental B_t :

$$B_t = (B_{t-1} \cup Obs_t \cup \{a_t\}) \setminus M$$

où $M \in \mathfrak{M}\big(\Phi, \mathit{shielded}\big)$. Par conséquent, à partir d'un état mental B_{t-1}, nous obtenons un arbre d'états où chaque branche correspond à un « choix de révision possible » et donc un état mental possible. Chaque chemin dans l'arbre est alors un *scénario*, c'est à dire, une suite d'état mentaux qui est cohérent avec les actions constatées de l'agent.

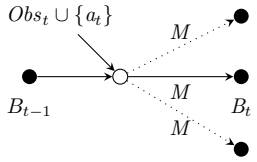


FIGURE 1 – Calcul de B_t (un cercle noir représente un état cohérent, un blanc un état potentiellement incohérent)

Dans la prochaine section, nous nous appuyons sur ces scénarios et les MCS successifs pour diagnostiquer les erreurs humaines dans la situation d'accident que nous modélisons.

4 Analyse de l'erreur humaine

La difficulté de ce travail d'analyse de l'erreur humaine est de distinguer ce qui correspond à de la révision de croyance (cas 1 dans la section précédente) de ce qui relève d'une prise de décision erronée (cas 2) dans les séries des MCS d'un scénario.

4.1 Révision de croyance *vs* erreur humaine

Certains choix de révision dans un scénario correspondent en effet à une révision de croyance rationnelle dans le contexte de l'inertie du décor. Cela se produit dans deux cas :

(1) La base de croyance doit être révisée à cause des effets attendus d'une action (p. ex. si l'agent charge le pistolet, la valeur de loaded_t doit changer, ce qui est en contradiction avec $keep^{(v)}_{\mathrm{loaded}_t}$).

(2) La base de croyance doit être révisée à cause d'une nouvelle information (p. ex. si l'agent croit initia-

lement que le pistolet est chargé et observe ensuite qu'il ne l'est plus).

Toutes les autres incohérences qui apparaissent dans les états mentaux correspondent à des erreurs dans le raisonnement de l'agent. Toutefois, nous pouvons à nouveau distinguer deux types d'incohérences. Le premier type vient directement du scénario (p. ex. les actions qui n'auraient pas dû être effectuées connaissant les observations). Nous parlons alors d'*incohérences locales*. Le deuxième type correspond au problème de la distorsion du décor. Dans ce qui suit, nous présentons notre algorithme pour séparer les trois types d'incohérences : les incohérences locales, les distorsions du décor, et les révisions de croyance « normales » qu'il faut accepter comme ne relevant pas de l'erreur humaine.

4.2 Un algorithme pour l'analyse de l'erreur humaine

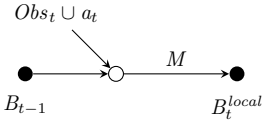
Notre algorithme fonctionne en trois phases. Nous commençons par détecter les MCS qui correspondent à des incohérences locales. Nous détectons ensuite les changements dans la base de croyances liés à l'inertie du décor (qui peuvent créer des incohérences et des MCS, mais qui ne sont pas des erreurs). Enfin nous détectons les MCS qui correspondent à des distorsions du décor.

4.2.1 Incohérence locale

En commençant par B_{t-1}, avant d'ajouter $\mathfrak{K}_t^{(v)}$ et $\mathfrak{K}_t^{(k)}$ dans la base de croyance, nous vérifions les incohérences locales en calculant :

$$\begin{aligned} B_t^{local} &= (B_{t-1} \cup Obs_t \cup \{a_t\}) \setminus M \\ &\text{avec } M \in \mathfrak{M}\big(B_{t-1} \cup Obs_t, \{a_t\}\big) \end{aligned} \tag{1}$$

Ceci nous permet de capturer les situations où un agent effectue des actions qui ne sont pas cohérentes avec l'état courant des observations et des croyances. L'état B_t^{local} qui en résulte est cohérent avec l'action mais peut contenir des corrections de la base de croyance qui correspondent à une prise de décision erronée.



Un exemple classique d'une telle situation est le crash du vol Rio-Paris en 2009 [4]. Les pilotes recevaient des informations contradictoires et prirent la mauvaise décision (l'avion était en décrochage et ils ont tiré le manche, ce qui a maintenu l'appareil en décrochage). Voici une représentation simplifiée de cette situation dans notre modèle :

$$\begin{array}{lcl} \mathcal{I}nit & = & \emptyset \\ Obs_1 & = & \{\mathrm{alarm}_1, \mathrm{acceleration}_1\} \\ \mathcal{R} & = & \left\{ \begin{array}{lcl} R_1 & \equiv & \mathrm{alarm}_t \rightarrow \mathrm{stall}_t \\ R_2 & \equiv & \mathrm{acceleration}_t \rightarrow \mathrm{overspeed}_t \\ R_3 & \equiv & \mathrm{overspeed}_t \rightarrow \mathrm{Pull}_t \\ R_4 & \equiv & \mathrm{stall}_t \rightarrow \mathrm{Push}_t \\ R_5 & \equiv & \mathrm{Pull}_t \wedge \mathrm{Push}_t \rightarrow \bot \end{array} \right\} \\ \mathcal{T} & = & \{\mathrm{Pull}_1\} \end{array}$$

Dans cet exemple, les observations de l'alarme et de l'accélération sont incohérentes entre elles (elles mènent à violer R_5 si on suit $R_{1\ldots4}$). Le MCS qui est cohérent avec l'action Pull_1 consiste à garder les observations et règles liées à la survitesse au lieu du décrochage. Cela correspond à ce qui s'est vraiment passé : les enregistreurs du cockpit montrent que les pilotes croyaient qu'ils étaient en survitesse et essayaient d'en sortir en tirant le manche.

4.2.2 Incohérence du décor

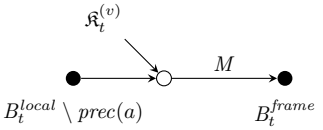
Pour chaque B_t^{local} possible, nous introduisons les propositions $keep^{(v)}$ qui sont responsables de l'inertie des croyances. Nous calculons ensuite :

$$\begin{aligned} B_t^{frame} &= ((B_t^{local} \setminus prec(a)) \cup \mathfrak{K}_t^{(v)}) \setminus M \\ &\text{avec } M \in \mathfrak{M}\big((B_t^{local} \setminus prec(a)) \cup \mathfrak{K}_t^{(v)}, \\ &\qquad B_t^{local} \setminus prec(a)\big) \end{aligned} \tag{2}$$

Cela nous permet de capturer les corrections dans la base de croyance qui sont liées au problème du décor. En effet, en calculant les MCS seulement sur les propositions *keep* en protégeant tout ce qui est dans B_t^{local}, nous détectons les incohérences entre les *keep* et les croyances dérivées des observations et des effets d'action. Plus précisément, l'ajout de $\mathfrak{K}_t^{(v)}$ maintient la valeur de vérité des croyances précédentes, mais laisse leur état de connaissance (les prédicats *known*) libre. Si une incohérence est détectée, nous savons qu'elle vient de ces croyances dérivées.

Toutefois, comme nous considérons que l'action a est faite à l'instant t, ses pré-conditions et les *known* associés doivent être vrais, ce qui au final force à vrai toutes les propositions qui peuvent être inférées par ces pré-conditions, ce que nous ne voulons pas. En effet, de telles propositions peuvent être incohérentes avec les déductions des règles *keep* que nous voulons appliquer à ce stade pour traiter seulement le problème du décor. C'est pourquoi nous retirons temporairement $prec(a)$, l'ensemble des règles de précondition de l'action a, du calcul de l'état mental (il sera réintroduit dans la phase suivante).

Toutes les incohérences capturées à ce stade correspondent à des révisions de croyances rationnelles dans le contexte de l'inertie du décor. Ce ne sont pas des erreurs humaines. Les états B_t^{frame} qui en résultent sont tous des états mentaux cohérents où les incohérences locales et les révisions de croyances « normales » ont été résolues.



La situation suivante illustre cette phase :

$$\begin{array}{lcl} \mathcal{I}nit & = & \{\mathrm{cloud}_0, \mathrm{sun}_0\} \\ Obs_1 & = & \{\neg\, \mathrm{cloud}_1\} \\ \mathcal{R} & = & \{\ R_1 \equiv [\neg\, \mathrm{cloud}_t \wedge \mathrm{sun}_t]\, \mathrm{GoOut}_t\ \} \\ \mathcal{T} & = & \{\mathrm{GoOut}_1\} \end{array}$$

Sortir (action GoOut_1) est un comportement rationnel dans cette situation, bien que cela crée une incohérence avec les

prédicats *known* et les propositions *keep* que nous avons introduites pour le Problème du Décor.
En effet, les pré-conditions de GoOut sont $\neg\,\mathrm{cloud}_1$ et sun_1 qui ne sont pas en contradiction avec les croyances et les observations, le MCS est donc vide pour l'incohérence locale et nous avons :

$$B_1^{local} = \{\mathrm{cloud}_0, \mathrm{sun}_0, R_1, \neg\,\mathrm{cloud}_1, \mathrm{GoOut}_1\}$$

Pour calculer B_1^{frame}, nous retirons la règle de précondition R_1 et ajoutons les *keep*$^{(v)}$:

$$\{\mathrm{cloud}_0, \mathrm{sun}_0, \neg\,\mathrm{cloud}_1, \mathrm{GoOut}_1, keep^{(v)}_{sun}, keep^{(v)}_{cloud}\}$$

Le MCS sur les *keep* contient $keep^{(v)}_{cloud}$ afin de rendre $\neg\,\mathrm{cloud}_1$ possible, et nous avons finalement :

$$B_1^{local} = \{\mathrm{cloud}_0, \mathrm{sun}_0, \neg\,\mathrm{cloud}_1, \mathrm{GoOut}_1, keep^{(v)}_{sun}\}$$

On a bien traité ici uniquement le problème du décor, en supprimant la règle d'inertie de la proposition cloud afin de lui permettre de changer conformément aux observations.

4.2.3 Incohérence de la distorsion du décor

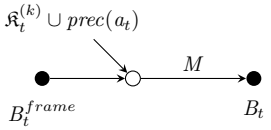
Pour chacun des états mentaux B_t^{frame} possibles, nous introduisons maintenant les $keep^{(k)}$, qui maintiennent l'état de connaissance des croyances (cela ne s'applique qu'aux croyances, pas aux propositions que l'on peut en dériver). L'analyse de cet ensemble de propositions nous permet de détecter les incohérences suivantes :

- — Un changement de croyance inattendu, que le problème du décor aurait dû éviter.
- — Un changement de croyance imposé par les pré-conditions de l'action effectuée ou par des désirs de l'agent.

Concrètement, nous calculons :

$$\begin{aligned} &B_t = (B_t^{frame} \cup prec(a) \cup \mathfrak{K}_t^{(k)}) \setminus M \\ &\text{avec } M \in \mathfrak{M}\big(B_t^{frame} \cup \mathfrak{K}_t^{(k)} \cup prec(a), \{a_t\}\big) \end{aligned} \tag{3}$$

Les états B_t résultants sont des états de croyances cohérents qui reflètent les éventuelles erreurs de décision capturées par les MCS.



L'exemple type est le crash du Mont-Saint-Odile présenté en Section 2.3. Les pilotes peuvent régler la vitesse verticale soit en pieds par minute (Vertical Speed) ou en degré d'angle de descente (Flight Path Angle). Quand l'indicateur est en FPA, un affichage de 33 veut dire que l'avion descend avec un angle de 3.3 degrés, ce qui correspond à environ 800 pieds par minute en termes de vitesse verticale. Quand l'indicateur est en VS, le même affichage veut dire que l'avion descend à la vitesse de 3300 pieds par minute (soit quatre fois plus vite) !

En voici une représentation simplifiée dans notre modèle :

$$\begin{aligned} \mathcal{I}nit &= \{\mathrm{onVS}_0, \neg\,\mathrm{onFPA}_0\} \\ Obs_1 &= \emptyset \\ \mathcal{R} &= \left\{ \begin{array}{ll} R_1 = & \mathrm{display}(33)_t \wedge \mathrm{onVS}_t \rightarrow \neg\,\mathrm{goodAngle}_t \\ R_2 = & \mathrm{display}(33)_t \wedge \mathrm{onFPA}_t \rightarrow \mathrm{goodAngle}_t \\ R_3 = & \mathrm{SetValue}(x)_t :: \mathrm{display}(x)_t \\ R_4 = & \mathrm{onVS}_t \leftrightarrow \neg\,\mathrm{onFPA}_t \end{array} \right\} \\ \mathcal{D} &= \{\mathrm{goodAngle}_t\} \\ \mathcal{T} &= \{\mathrm{SetValue}(33)_1\} \end{aligned}$$

L'enquête a montré que les pilotes savent initialement que l'indicateur est en VS (Vertical Speed) mais « oublient » cette information et règlent la valeur sur 33, ce qui aurait été la bonne valeur s'ils avaient été sur le mode FPA (Flight Path Angle). En appliquant notre algorithme sur ce modèle, nous avons un MCS vide pour l'incohérence locale car l'agent ne peut pas déduire $\neg\,\mathrm{goodAngle}_1$ vu qu'il n'observe pas onVS_1 au temps 1. Nous avons alors :

$$B_1^{local} = \left\{ \begin{array}{l} \mathrm{onVS}_0, \neg\,\mathrm{onFPA}_0, \mathrm{goodAngle}_0, \\ \mathrm{goodAngle}_1, \mathrm{SetValue}(33)_1, \\ R_1, R_2, R_3, R_4 \end{array} \right\}$$

Pour calculer B_1^{frame} nous ajoutons les $keep^{(v)}$, c'est-à-dire les clauses qui gardent la valeur de vérité des propositions entre deux pas de temps. Aucune incohérence n'est détectée du fait que $known_{onVS_1}$ est libre et peut être donc faux pour que le système soit toujours cohérent. Nous avons alors :

$$B_1^{frame} = \left\{ \begin{array}{l} \mathrm{onVS}_0, \neg\,\mathrm{onFPA}_0, \mathrm{goodAngle}_0, \\ \mathrm{goodAngle}_1, \mathrm{SetValue}(33)_1, \\ R_1, R_2, R_3, R_4, \\ keep^{(v)}_{\mathrm{onVS}_1}, keep^{(v)}_{\mathrm{onFPA}_1} \end{array} \right\}$$

Pour calculer B_1 en prenant en compte la distorsion du décor, nous ajoutons les $keep^{(k)}$, c'est-à-dire les clauses qui gardent la valeur de connaissance (*known*) entre deux pas de temps : $\{keep^{(k)}_{\mathrm{onVS}_1}, keep^{(k)}_{\mathrm{onFPA}_1}\}$ Nous avons alors une incohérence avec la proposition $\mathrm{goodAngle}_1$ car $\neg\,\mathrm{goodAngle}_1$ peut être maintenant déduite par R_1. Un MCS possible est alors de retirer $\{keep^{(k)}_{\mathrm{onVS}_1}, keep^{(v)}_{\mathrm{onFPA}_1}\}$ et ainsi avoir :

$$B_1 = \left\{ \begin{array}{l} \mathrm{onVS}_0, \neg\,\mathrm{onFPA}_0, \mathrm{goodAngle}_0, \\ \mathrm{goodAngle}_1, \mathrm{SetValue}(33)_1, \\ R_1, R_2, R_3, R_4, \\ keep^{(v)}_{\mathrm{onVS}_1}, keep^{(k)}_{\mathrm{onFPA}_1} \end{array} \right\}$$

Nous avons par cette dernière étape traité le problème de la distorsion du décor : l'inertie du décor n'est pas respectée par l'agent afin d'être cohérent avec son désir d'avoir un bon angle d'approche, en oubliant notamment qu'il était en mode Vertical Speed.

Dans cette section, nous avons illustré notre algorithme avec 3 exemples distincts mais sur une seule étape temporelle. Dans le cas général, les différents MCS obtenus à chacun des pas de temps créent des branches, ce qui produit pour des cas d'études réels des arbres d'états mentaux de grande taille, qui décrivent de nombreux scénarios possibles. Nous discutons de ce point dans les perspectives.

5 Travaux connexes

Le diagnostic en Intelligence Artificielle, c'est-à-dire la recherche d'explications à une situation en utilisant la modélisation logique, est étudié depuis les années 80 [19]. Toutefois à notre connaissance aucun des modèles proposés jusqu'ici n'est appliqué dans le contexte de l'analyse d'erreurs de prise de décision humaine. Bien que plusieurs travaux proposent des solutions pour diagnostiquer des systèmes dynamiques (*i.e* en prenant en compte des actions et des changements) [16, 24] tous supposent que les solutions doivent respecter l'inertie du décor. Le problème de la distorsion du décor qui apparaît lors de l'analyse des erreurs humaines n'est pas pris en compte. L'objectif de notre modèle est de dépasser cette limitation.

Pourtant, depuis quelques années, la recherche en IA s'est intéressée à la modélisation des erreurs de raisonnement humain ou, plus généralement, aux limites du raisonnement humain, à des fins de prédiction en simulation. Par exemple, [26] utilise un automate fini pour simuler la dynamique d'opinion sur la vaccination. Il permet d'expliquer une décision de non-vaccination alors que l'ensemble des informations rationnelles à la disposition de l'agent devrait l'amener à prendre la décision d'accepter la vaccination. Dans un contexte différent, [12] propose les *Synthetic Cognitive Models* pour simuler la prise de décision dans un contexte militaire en prenant en compte la rationalité limitée de l'humain. Enfin, les auteurs de [2] utilisent le paradigme BDI pour implémenter des fonctions probabilistes qui mènent à des croyances erronées dans une situation de feux de forêt. Tous ces modèles proposent une solution viable pour simuler des erreurs de prise de décision par des humains mais ils ne peuvent pas être utilisés dans un objectif de diagnostic dans un cas général.

Une autre approche pour capturer les croyances et les raisonnements erronés consiste à s'affranchir de l'hypothèse d'*omniscience logique* telle que définie dans [9], c'est-à-dire la capacité à inférer toutes les conséquences d'une croyance φ. Par exemple, dans [23], les auteurs proposent un modèle basé sur les *mondes impossibles* (*i.e* les mondes qui ne sont pas fermés sous conséquences logiques) pour simuler des erreurs de raisonnement. Pour cela ils associent à chaque règle de raisonnement une *consommation de ressources*, ce qui limite l'application de longs raisonnements. Toutefois, le calcul de tous les *mondes impossibles* pour sélectionner le plus plausible nécessite une puissance de calcul exponentielle. De plus leur modèle ne considère pas les actions et les changements.

Toutes ces approches donnent des solutions intéressantes bien que partielles à notre problème : elles ne prennent pas en compte le problème de la distorsion du décor et ne sont pas adaptées pour un objectif de diagnostic. Notre modèle reprend les idées proposées par ces différents auteurs pour modéliser des erreurs de prise de décision et calculer des diagnostics qui prennent en compte les erreurs humaines. De plus, l'utilisation d'un solveur SMT nous permet de traiter à la fois des variables continues et des variables discrètes, et d'augmenter ainsi l'expressivité du modèle, comme l'a montré [7].

6 Conclusion et perspectives

Notre modèle utilise un diagnostic basé sur la révision de croyance pour calculer les suites d'états mentaux possibles qui peuvent expliquer des erreurs de prise de décision humaine. Ces état mentaux sont cohérents avec les observations et les actions effectuées par l'agent et prennent en compte le *problème de la distorsion du décor*, à savoir le fait que les croyances humaines peuvent être modifiées sans cause externe. Alors que le problème du décor stipule que les propositions doivent être maintenues quand elles ne sont pas modifiées par une action, l'analyse des erreurs humaines doit prendre en compte des changements spontanés dans le décor tout en maintenant un comportement rationnel de la part de l'agent.

Nous avons implémenté ce modèle à l'aide d'un solveur SMT. Notre algorithme construit un arbre où chaque chemin correspond à un scénario possible pour les actions observées.

Pour le moment, notre modèle calcule l'ensemble des scénarios possibles mais n'identifie pas le plus plausible. Par exemple, dans le modèle complet du crash Rio-Paris, nous trouvons plus de 6000 scénarios, ce qui est beaucoup pour une analyse par un expert humain. Pour pallier cette limitation, nous envisageons d'étendre notre algorithme de manière à filtrer l'ensemble des scénarios et extraire les erreurs humaines « classiques », identifiées dans la littérature en sciences humaines sous le terme de « biais cognitifs » [25]. Une première proposition a été faite dans [6] pour définir des motifs logiques permettant d'identifier quelques biais cognitifs dans les accidents. Notre proposition à terme est d'inclure ces motifs dans notre modèle et de les étendre pour capturer d'autres biais afin de donner un ordre de priorité aux scénarios présentés aux experts.

À plus long terme, notre objectif est de fournir des outils pour aider les experts à comprendre et anticiper les situations d'accident. Nous pensons qu'en utilisant la modélisation logique, nous pouvons générer des scénarios non anticipés par les experts humains. Nous voulons aussi que notre modèle permette de faire la distinction entre une conception ergonomique médiocre et susceptible de conduire à des erreurs (comme l'accident du Mont-Saint-Odile) et les erreurs humaines qui peuvent être identifiées et évitées par la formation. Ces erreurs peuvent provenir de la charge cognitive et peuvent être modélisées par la rationalité limitée. Nous avons l'intention d'étendre notre cadre logique pour capturer ce phénomène cognitif majeur.

Références

[1] Carlos E Alchourrón, Peter Gärdenfors, and David Makinson. On the logic of theory change : Partial meet contraction and revision functions. *The journal of symbolic logic*, 50(2) :510–530, 1985.

[2] Maël Arnaud, Carole Adam, and Julie Dugdale. The role of cognitive biases in reactions to bushfires. In *ISCRAM*, Albi, France, May 2017.

[3] BEA. Bea f-ed920120. Technical report, Bureau d'Enquêtes et d'Analyses pour la sécurité de l'aviation civile, 1993.

[4] BEA. Bea f-cp090601. Technical report, Bureau d'Enquêtes et d'Analyses pour la sécurité de l'aviation civile, 2012.

[5] Leonardo De Moura and Nikolaj Bjørner. Z3 : An efficient smt solver. In *International conference on Tools and Algorithms for the Construction and Analysis of Systems*, pages 337–340. Springer, 2008.

[6] Valentin Fouillard, Nicolas Sabouret, Safouan Taha, and Frédéric Boulanger. Catching cognitive biases in an erroneous decision making process. *IEEE International Conference on Systems, Man and Cybernetics (SMC)*, 2021.

[7] Alban Grastien. Diagnosis of hybrid systems with SMT : opportunities and challenges. *ECAI 2014*, pages 405–410, 2014.

[8] Peter Gärdenfors and Hans Rott. *Belief Revision*. Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 1992.

[9] Joseph Y. Halpern and Riccardo Pucella. Dealing with logical omniscience : Expressiveness and pragmatics. *Artificial Intelligence*, 175(1) :220–235, 2011. John McCarthy's Legacy.

[10] Steve Hanks and Drew McDermott. Nonmonotonic logic and temporal projection. *Artificial intelligence*, 33(3) :379–412, 1987.

[11] Erik Hollnagel. *Cognitive reliability and error analysis method (CREAM)*. Elsevier, 1998.

[12] Jonathan Kulick and Paul K Davis. Modeling adversaries and related cognitive biases. *Modeling Adversaries and Related Cognitive Biases*, 2003.

[13] Mark H Liffiton and Karem A Sakallah. Algorithms for computing minimal unsatisfiable subsets of constraints. *Journal of Automated Reasoning*, 40(1) :1–33, 2008.

[14] John McCarthy. Applications of circumscription to formalizing common-sense knowledge. *Artificial intelligence*, 28(1) :89–116, 1986.

[15] John McCarthy and Patrick J Hayes. Some philosophical problems from the standpoint of artificial intelligence. *Machine Intelligence*, page 463–502, 1969.

[16] Sheila A Mcllraith. Explanatory diagnosis : Conjecturing actions to explain observations. In *Logical Foundations for Cognitive Agents*, pages 155–172. Springer, 1999.

[17] Atsuo Murata, Tomoko Nakamura, and Waldemar Karwowski. Influence of cognitive biases in distorting decision making and leading to critical unfavorable incidents. *Safety*, 1(1) :44–58, 2015.

[18] Gabriele Paul. Approaches to abductive reasoning : an overview. *Artificial intelligence review*, 7(2) :109–152, 1993.

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

[20] Raymond Reiter. 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*, pages 359–380. Citeseer, 1991.

[21] Erik Sandewall. Cognitive robotics logic and its metatheory : Features and fluents revisited. *Electron. Trans. Artif. Intell.*, 2 :307–329, 1998.

[22] Murray Shanahan. The Frame Problem. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, Spring 2016 edition, 2016.

[23] Anthia Solaki, Francesco Berto, and Sonja Smets. The logic of fast and slow thinking. *Erkenntnis*, 86(3) :733–762, 2021.

[24] Michael Thielscher. A theory of dynamic diagnosis. *Electronic Transactions on Artificial Intelligence*, 1(4) :73–104, 1997.

[25] Amos Tversky and Daniel Kahneman. Judgment under uncertainty : Heuristics and biases. *Science*, 185(4157) :1124–1131, 1974.

[26] Marina Voinson, Sylvain Billiard, and Alexandra Alvergne. Beyond rational decision-making : modelling the influence of cognitive biases on the dynamics of vaccination coverage. *PloS one*, 10(11), 2015.

[27] Renata Wassermann. An algorithm for belief revision. In *Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning*, pages 345–352, 2000.