University of Provence (Aix-Marseille I)
UFR de Psychologie - Sciences de l'Education
29 avenue Robert Schuman
13621 Aix en Provence - France

Thesis (not) submitted in support
of the accreditation to supervise research

(Not) Presented and (not) defended publicly by

Denis Besnard

Human cognitive performance

From troubleshooting hardware faults to work in socio-technical systems

Members of Committee

Instead of displaying the list of people who accepted to be a member,, I would like to express my gratitude to 3 reviewers of my work : Nathalie Bonnardel (Professor, University of Provence), Chris Johnson (Professor, University of Glasgow) and Elaine Seery (http://www.traduction-edition-scientifique.fr/).

Page left blank intentionally

University of Provence (Aix-Marseille I)
UFR de Psychologie - Sciences de l'Education
29 avenue Robert Schuman
13621 Aix en Provence - France

Note de synthèse (non) soumise en vue de l'obtention
de l'habilitation à Diriger les Recherches

(Non) Présentée et (non) soutenue publiquement par

Denis Besnard

La performance cognitive humaine

Du diagnostic des pannes matérielles au travail dans les systèmes socio-techniques

Membres du jury

Plutôt que de donner la liste des gens qui avaient accepté de faire partie du jury, je voudrais exprimer ma gratitude à 3 relecteurs de mon travail : Nathalie Bonnardel (Professeure, Université de Provence), Chris Johnson (Professor, University de Glasgow) et Elaine Seery (http://www.traduction-edition-scientifique.fr/).

Page left blank intentionally

Table of Contents

Page left blank intentionally

Table of Figures

Acknowledgements

Preparing a thesis for a diploma, especially one that involves a viva is not trivial. Of course, the academic work that has to be put into it consumes some time and effort. But finding people who would kindly agree to act as reviewers and also attend the viva is also challenging. I know it imposes constraints and that time is a limited resource (I will come back to that). Therefore, I am very grateful for the availability and patience a number of people gave me.

I should first thank Evelyne Marmèche, Cecilia De La Garza, Chris Johnson and Frédéric Vanderhaegen very warmly for their availability. I know they all have very busy diaries, and finding time for reading and then travelling is not easy. In this respect, I hope that this long document I have inflicted on them will provide some useful thoughts for their own work. Also, given that our scientific interests show some overlap, I hope that we will have a chance to interact in the future. However, writing this very line, I sense some irony in the situation. I express a wish that will, in part, depend on a event (the viva) over which I only have partial control. Oh well...is it not an essential feature of human cognition to make decisions under uncertainty?

I am also thankful to Nathalie Bonnardel for her supervision. She took the time to explain the mechanics of the HDR[1] to me, review the thesis at various stages and make sure I was going to produce an acceptable document (in the sense that reading it wouldn't be too high a time loss to anyone). These discussions also got us to discuss more mundane matters and know each other better. Nathalie has been working for many years in the same department where I graduated as a doctor in 1999. Nevertheless, we did not really take the time to chat. This HDR and the update meetings that took place along the way gave us a few opportunities to do so, which I enjoyed.

I also express my gratitude to Erik Hollnagel (Industrial Safety Chair at Mines-ParisTech) who commented on the manuscript and heavily influenced the way I see human performance. I have worked with Erik for about a year and a half now, and I have been seduced by how easy he makes his

1 Habilitation à Diriger les Recherches (accreditation to supervise research)

collaborators' life, and how much guidance he can provide in just a few words. I also learned that cows have a different view of time than most of us. Apparently, *they* think it is something *we* invented. I remembered this thought many times, particularly when I was working long hours over week-ends to write this thesis. Looking at my watch, thinking that I should stop and call it a day, I smiled and thought: "Well, it's only an invention, isn't it?". If you ever visit Erik in his office at Mines-ParisTech in Sophia Antipolis, ask him to show you where all this comes from.

Also, I have to mention the immense effort that Gordon Baxter (University of St Andrews) produced in adding to the literature review and checking my English. I wanted to write in this language for several reasons (I will address this point in the foreword). One of these had to do with my extraordinary overconfidence in the mastery of the tongue. And then reality caught up with me. Despite having lived in the UK for several years, I now know how far I am from writing correct English, which I measure by the vast number of corrections Gordon did. These were not about major mistakes but even I have to acknowledge that fixing them has improved the reading a great deal. So if this document complies to your idea of an acceptable English standard, you know it is not entirely down to me...

I also have a thought for colleagues (Emmanuel Garbolino, Valérie Godfrin, Aldo Napoli, Eric Rigaud) who have also embarked on the HDR, which I now treat as a slow and heavy ship that doesn't let herself steer away easily from its initial heading. I thank them for feedback, comments and support, and wish them all the best for their journey. On this front, as a former sailor, I re-discovered that a challenging decision in navigation is setting the right heading early, taking all drift factors into account, and while still leaving some room for unplanned events or conditions at sea. It certainly applies to the HDR.

Franck Guarnieri is my lab director and he was the one who announced to my colleagues and myself that we had to write an HDR thesis each. He only cascaded this decision down to us since it was imposed on him from higher up. In a way, he is the person who truly is at the origin of this document.

On a personal noowt, how could Ah possibly forget to thank me dear geordie pet Elaine for letting me wark during oll'em Sondays and evenings withoowt a woard? Ah nae that this knackered of some of wor week-ends but it nivver caused no major trouble. From this aloone, Ah meaja how import'nt it is to have a gud marra as a partna', and how mouch can be doon at woark when everythin's gannin' alreet at hyame. It had nowt to do with me, like. It's all doon to me pet.

Finally, I thank very warmly the people behind openSUSE[2] and OpenOffice[3], the operating system and office suite I used at home to write this thesis.

2 More information can be found at http://www.opensuse.org and from Will Stephenson, should you have the privilege to know him.

3 More information can be found at http://www.openoffice.org

Foreword

My choice of English as a language in this thesis deserves some explanations. My main motivation has been to produce a document that I could share with international researchers. Writing in English has allowed me to contact Chris Johnson as a committee member, a less obvious option had I written in my native tongue. Also, English has allowed me to ask English-speaking colleagues to review my work. Last, I intend to extract as much value as I can from this thesis and the reflections it contains. In this respect, English will allow me to disseminate my work to a larger number of people within my scientific community.

Conversely, English has implied an effort on the part of the French members of the committee and for all the potential French readers of my work. Despite the fact that even in France, a significant part of our scientific work is done is English, I beg my compatriots to accept my blue, white and red apologies. I hope the French summary that follows this foreword, as well as the list of translated terms at the end of the manuscript, will be of some assistance to them.

Concerning the contents of the document, you (dear reader) will notice that I use “I” as a way to refer to myself, as opposed to the traditional scientific practice of writing impersonally. I thought that this thesis, by its reflexive nature on my own activities, called for such a personal reference as a way to claim ownership.

Finally, my approach to this thesis has been to put my work back into the scientific landscape it belongs. To do so, I have used my own publications of course. But I have tried to complement them with at least some elements of a literature review. As often, this dual approach has strengths and weaknesses. The good side, I hope, is that it eases the evaluation of my contribution to my discipline. The down side is that what I call elements of a literature review certainly do not cover all the issues.

French summary

Cette section résume en français la présente note de synthèse écrite en Anglais dans le cadre de l'habilitation à diriger les recherches. Puisque c'est un résumé, le lecteur y trouvera un enchaînement de thématiques identique à celui présent dans la note elle-même. L'objectif premier est ici de permettre au lecteur non anglophone d'embrasser rapidement la structure de la note.

Introduction. La performance humaine comme sujet de recherche

Mon approche de la recherche est plutôt déterministe. J'essaie de trouver des causes à des comportements humains en partant du principe que notre compréhension imparfaite des activités humaines est la raison pour laquelle le concept de variabilité est parfois utilisé comme une boîte noire. En d'autres termes, des connaissances plus précises sur la cognition humaine nous permettraient de comprendre et prédire plus finement la performance, augmentant progressivement notre connaissance des facteurs de cette variabilité. Idéalement, les modèles de performance humaine devraient être aussi précis que les modèles météorologiques. Mais est-ce réaliste?

En effet, un certain nombre de limites existent. La première touche à la diversité, au nombre et aux diverses combinaisons possibles de ces facteurs. Une autre limitation touche aux effets de la mesure de la performance sur la performance elle-même. Enfin, la mesure de la performance dans les activités naturelles modifie la tâche. Face à ces limites, il se peut qu'un modèle des conditions générales de variation de la performance constitue la limite de ce que nous pouvons élaborer. Un tel modèle fournirait une connaissances des classes de situations dans lesquelles la performance varie. La structure de la note de synthèse s'inspire de cette idée d'un découpage grossier de classes de situations. En effet, j'aborde trois grandes catégories de situations dans lesquelles j'analyse la performance humaine:

- Le diagnostic de pannes matérielles dans les systèmes statiques;
- La conduite de systèmes dynamiques;
- Le travail dans les systèmes socio-techniques.

Pour chacune ce ces catégories, j'utilise un certain nombre d'exemples

mettant en exergue des aspects positifs et négatifs (c'est à dire désirés et non désirés) de la performance humaine. Le chapitre relatif à chacune des trois catégories s'achève par une question touchant à un facteur qui influe sur la performance. Ces trois dimensions serviront à alimenter la section finale de la note dans laquelle j'aborde un projet de carrière.

Chapitre 1. La performance cognitive dans les systèmes statiques

L'interprétation de l'information qui nous entoure est une activité universelle au travers de laquelle notre compréhension du monde se construit. Cette compréhension est fortement influencée par nos connaissances, au point que la cause (immédiate et visible) d'un événement peut être négligée à la faveur d'une explication plus en conformité avec ce que nous savons. L'interprétation est une activité importante car elle conditionne en partie les actions et leur pertinence vis-à-vis du contexte. Cependant, l'interprétation à elle seule ne conditionne pas la totalité de nos actions. En revanche, lorsqu'elle est étudiée conjointement à la recherche d'information, la planification et l'action, l'interprétation s'intègre à une activité plus large et plus complexe : le diagnostic. On aborde alors un exercice de raisonnement où la compréhension de l'état d'un système défaillant va permettre de restaurer un état de fonctionnement normal. Cependant, ce dépannage ne va pas de soi. Il existe un certain nombre de stratégies, et d'objectifs qui conditionnent la manière dont il est conduit. C'est ce que les différentes sections de ce chapitre tentent de démontrer.

Les stratégies de diagnostic décrivent essentiellement les différentes manières d'effectuer un diagnostic, indépendamment du domaine.

- Les stratégies topographiques suivent les composants physiques ou logiques du système et exploitent l'idée de propagation des symptômes, à la fois vers l'amont et l'aval de la cause de panne. Cette stratégie ne nécessite pas une connaissance avancée du système mais peut se révéler coûteuse en temps. C'est typiquement le type de stratégie que développent les opérateurs novices.
- Le raisonnement par chaînage est une stratégie qui vise à inférer plutôt que suivre topographiquement les effets et les causes d'une panne. Ici également, le chaînage peut être prospectif (en aval de la panne pour en déduire ses effets) et rétrospectif (pour en déduire ses causes).
- Les stratégies symptomatiques associent des symptômes à des causes, indépendamment de l'emplacement physique ou logique des composants. C'est une association rapide qui repose sur la connaissance préalable des fonctions de chaque composant et des sources de perturbation potentielles. Cette stratégie nécessite

généralement un haut degré d'expertise sur le système concerné.

- Le split-half est une stratégie logique qui vise à scinder le système en deux parties : une qui contient la panne et l'autre qui ne la contient pas. Cela se fait généralement en isolant un composant ou une fonction, et en observant les effets de cette action sur les symptômes.

J'ai brièvement évoqué l'expertise dans la présentation de ces stratégies de diagnostic. Cette dimension de la cognition humaine a généralement des effets positifs sur la performance compte tenu de la diversité des problèmes rencontrés au fil du temps dans le domaine considéré. Cette expérience se traduit souvent par la mise en place de modes de raisonnement dits heuristiques, basés sur des règles d'association entre des symptômes et des causes possibles. Ces raisonnements heuristiques offrent des sortes de raccourcis de raisonnement qui visent un compromis entre les ressources cognitives investies dans le raisonnement et la qualité des résultats obtenus. Puisque c'est un compromis, la perfection ne peut être atteinte et il existe des cas de figure où l'expertise n'est plus une garantie de performance. C'est le cas du diagnostic de pannes rares qui présentent des symptômes similaires à des pannes fréquentes. Dans ce cas de figure, l'expert met en place une stratégie symptomatique et apparie des symptômes connus à une cause inconnue. Le caractère exceptionnel de la panne est alors omis du fait de l'identification de symptômes familiers. Dans ce cas, le diagnostic de l'expert s'oriente presque automatiquement vers (et se fixes sur) les causes de panne les plus fréquentes, au détriment de la cause de panne réelle. C'est ce que j'ai démontré expérimentalement dans le dépannage en mécanique automobile. Des mécaniciens experts ont été confrontés à une panne rare présentant des symptômes typiques d'une panne bien connue. Les résultats montrent que ces opérateurs orientent leurs tests sur la base de probabilités empiriques : les causes les plus probables sont testées en premier. Le pouvoir d'explication de ces causes supposées est tel que les experts répètent certains de leurs tests plusieurs fois, pris dans une boucle de fixation.

Une panne de nature similaire (symptômes connus mais cause rare) a été testé en électronique de manière expérimentale. On retrouve les mêmes résultats (heuristique de fréquence, fixation), augmentés d'une particularité : les opérateurs novices testent le composant en panne plus rapidement que les experts. Cependant, le niveau de connaissance des novices ne leur permet pas d'interpréter ce test correctement et la cause de panne n'est pas découverte. Ce comportement renforce l'idée que l'identification de la cause d'un événement n'obéit pas à la seule logique du raisonnement mais également aux connaissances requises par le processus d'interprétation.

Au-delà des mes travaux expérimentaux sur le dépannage, j'ai abordé le sujet

du diagnostic dans les situations de planification et de raisonnement dans des tâches plus larges. En effet, le diagnostic contribue à la régulation de l'activité de conduite de systèmes dynamiques, à la caractérisation de l'état d'un système et à la reconnaissance des états qui dévient de la normalité. C'est ce dernier aspect qui est développé dans le chapitre suivant.

Chapitre 2. La performance cognitive dans les systèmes dynamiques critiques

Les systèmes dynamiques critiques (par exemple les systèmes de transport, et en particulier les avions) ont deux propriétés essentielles : leur évolution est très liée à la variable temps et leur défaillance peut engager de lourdes pertes matérielles ou en vies humaines. La prise de décision dans ces systèmes peut se révéler complexe, notamment au niveau de la récupération d'actions ou d'événements non désirés. En effet, dans les systèmes dynamiques, le temps qui s'écoule entre un état non désiré et sa détection (puis sa récupération) modifie l'état de ce système et peut placer ce dernier dans une situation critique. Il existe de très nombreux exemples de ce phénomène dont quelques uns sont repris dans la note de synthèse.

L'interaction avec les systèmes dynamiques critiques, du fait même de leur complexité, implique souvent une part d'automatisation, sous une forme informatique par exemple. Historiquement, cette automatisation a permis d'alléger la charge de travail des opérateurs de contrôle de ces systèmes (les pilotes, par exemple). En retour, cette réduction de charge permet aux opérateurs de développer des stratégies d'anticipation des états futurs afin de se placer "en avance" par rapport au système contrôlé. Cependant, cette même automatisation a introduit de nouveaux comportements de conduite sous-optimaux dus aux surprises causées par des états du système inattendus. C'est le cas typique du pilote d'avion qui ne parvient pas à comprendre pourquoi l'appareil ne suit pas la trajectoire prévue ou pourquoi il n'arrête pas sa montée à l'altitude de vol programmée.

Un type de phénomène relié à l'automatisation dans les systèmes dynamiques critiques, en particulier les avions, est le conflit cognitif. Il concerne les situations où un opérateur développe un modèle mental de l'évolution de l'état du système qui n'est pas compatible avec son évolution réelle. Il existe alors un décalage entre ce à quoi l'opérateur s'attend et le comportement réel du système. De ce décalage peuvent survenir des événements inattendus ou une absence d'événements attendus. Dans les deux cas, le conflit cognitif peut mener à une perte de contrôle totale et causer des pertes importantes. L'aviation est un domaine particulièrement exposé à ce genre d'événement.

Les cas de la collision avec le relief d'un Boeing 757 à Cali[4], et celui de l'atterrissage train rentré d'un DC-9 à Houston[5] viennent illustrer le propos. Ces deux accidents se sont produits sans aucune défaillance technique, ce qui est typique des conflits.

Les confusions de mode sont un autre type de comportement caractéristique des systèmes dynamiques critiques étant donné leur haut niveau d'automatisation. Plusieurs exemples existent dans l'aviation ou la navigation maritime où un opérateur a entré une donnée valide ou effectué une action licite dans un mode erroné. Ce type de confusion est typique des systèmes de pilotage à base d'ordinateurs où un même instrument comporte plusieurs modes distincts mais acceptant des données d'entrée similaires. Les données d'entrée font généralement l'objet d'une certaine attention dans leur calcul. En revanche, les différents modes sont sélectionnés par des interfaces de type rotacteur (bouton rotatif) ou bouton-poussoir avec lesquelles des ratés de l'action (au niveau moteur) ou des difficultés de perception (affichage peu visible par exemple) sont possibles. Les confusions de mode dans le pilotage de systèmes dynamiques critiques peuvent déclencher des accidents avec pertes importantes. La récupération des confusions de mode est parfois impossible dans la mesure où le système ne montre pas toujours un changement de comportement saillant par rapport aux attentes construites par le ou les opérateur(s). Les confusions de mode seront illustrées par les cas du crash d'un Airbus A300 à l'aéroport de Nagoya[6], la collision avec le relief d'un Airbus A320 au mont Sainte-Odile[7], et l'échouage du paquebot Royal Majesty sur l'île de Nantucket[8]. Ces cas mettent tous en exergue la perte de contrôle des opérateurs (parfois non détectée), et l'extrême difficulté de la récupération, lorsqu'elle a lieu.

Les systèmes dynamiques critiques sont également une source d'accidents liés à la complexité des interfaces. Dans plusieurs industries (aviation, métallurgie, nucléaire), des accidents sont survenus du fait d'une décision erronée de la part d'un ou de plusieurs opérateur(s) due à l'incompréhension d'un aspect de l'interface de contrôle. Au travers de l'accident nucléaire à la centrale de Three Mile Island[9], d'un accident sur un site métallurgique Français, et du crash d'un Boeing 737 à Kegworth[10], on verra que les interfaces contribuent à l'occurrence d'accidents. Ces interfaces doivent souvent leur difficulté d'utilisation et/ou de lecture à une conception sub-

4 Colombie
5 Etats-Unis
6 Japon
7 Près de Strasbourg
8 Massachusetts, Etats-Unis
9 Etats-Unis
10 Angleterre

optimale sur le plan ergonomique. Elles contribuent à l'apparition de conditions opérationnelles défavorables au regard de la performance.

Chapitre 3. La performance cognitive dans les systèmes socio-techniques

Les systèmes socio-techniques sont soumis à plusieurs types de contributions de la part des opérateurs de ces systèmes. Dans la mesure où le travail réel n'est pas une réplication par l'action du travail prescrit, la performance humaine est presque toujours une variation de ce que les procédures prescrivent. Cette variation peut prendre plusieurs formes et avoir des conséquences diverses selon qu'elle s'éloigne fortement ou non des pratiques et des règles, et qu'elle place un opérateur dans une situation anticipée ou, au contraire, inattendue. Dans ce chapitre, une typologie des variations est proposée, qui permet de qualifier un comportement au regard de critères tels que l'intention, la légitimité, l'anticipation et l'impact. Un message fort dans ce chapitre consiste à poser que les capacités d'anticipation d'un opérateur sont un facteur important de sécurité : pouvoir anticiper correctement les effets d'une action (légitime ou pas) est un déterminant de sécurité plus fort que la simple adhésion à une procédure.

Sur un plan plus général, la coopération homme-système comporte un volet positif et un volet négatif. Les aspects positifs se rencontrent dans plusieurs cas de figure. C'est le cas lorsque le système compense la performance humaine (par l'automatisation de certaines tâches, par exemple). C'est également le cas lorsque les humains, à l'inverse, compensent les imperfections du système (dépasser les règles du travail prescrit lorsqu'elles sont inadaptées, par exemple). Ce dernier cas de figure est mis en évidence au travers de l'exemple d'ouvriers de fonderie qui délaissent l'ordinateur pour certaines tâches de mise en forme, et le cas de l'atterrissage d'urgence d'un DC-10 à Sioux City[11]. Dans ce dernier cas, les pilotes ont improvisé une nouvelle configuration de l'équipage ainsi qu'une technique de vol, suite à la rupture de plusieurs lignes hydrauliques. Ces variations ont permis, malgré un état technique de l'avion particulièrement inadapté au vol, de sauver la moitié des passagers.

La coopération comporte également des aspects négatifs lorsque le système ne permet l'interaction qu'au travers d'une interface imparfaite, génératrice d'incompréhensions. On peut également relever les cas de figure où les opérateurs mettent en place des variations de procédure dont ils ne maîtrisent pas les conséquence ou bien les cas de rejet des règles et des consignes. Les cas industriels qui illustrent ces types d'interaction sous-

11 Iowa, Etats-Unis

optimale sont ceux du système de radiothérapie défaillant Therac-25, l'accident à l'usine de combustible nucléaire JCO à Tokaimura[12], et l'explosion à la centrale de Tchernobyl[13]. A JCO, par exemple, des opérateurs ont utilisé un conteneur inapproprié au mélange d'une solution à base d'uranium. De plus, les quantités d'uranium manipulées comportaient un risque. La réaction nucléaire qui a suivi a été fatale pour certains opérateurs et a mis en évidence une dérive progressive des pratiques vers des comportements non sûrs.

D'une manière générale, la contribution humaine à la performance des systèmes socio-techniques est dépendante du contexte. L'existence même de violations et d'adaptations négatives, par exemple, est le signe d'un arbitrage entre les conditions dans lesquelles s'effectue le travail et les pressions qui pèsent sur les opérateurs. On retrouve alors dans certain cas le dilemme typique entre incitation à la productivité et obligations de sécurité. Ces scénarios prennent souvent place dans un contexte organisationnel défavorable dans lequel les adaptations négatives sont souvent vues comme des causes d'accident plutôt que comme les révélateurs de tensions opérationnelles. Il s'ensuit que lorsqu'un événement redouté survient, la culture du blâme biaise souvent le travail d'enquête et prend le pas sur la découverte des causes profondes.

Chapitre 4. Le futur : assister la cognition humaine

Compte tenu de la forte automatisation des systèmes dynamiques critiques, et des enjeux liés à leur défaillance, une question de fond reste ouverte : quel est le système d'assistance à la prise de décision qui permet de tirer le meilleur potentiel de coopération humain-machine? La réponse que je propose à cette question s'inspire des systèmes d'assistance à la prise de décision existant dans l'aviation et des défis identifiés dans chacun des chapitres 1, 2 et 3.

Les systèmes d'assistance que je présente tentent tous d'apporter une assistance proactive aux opérations humaines. Typiquement, des options d'action ou des éléments de compréhension sont envoyées à l'opérateur en prévision d'un état futur.

Pilot's Associate (Banks & Lizza, 1991) a été conçu comme un système d'assistance à la décision pour les avions de combat. Il est composé de trois modules principaux qui produisent des inférences sur les intentions du pilote, organise la présentation d'informations dans le cockpit, et fournissent de l'aide active. Hazard Monitor (Bass *et al.*, 1997, 2004) est centré sur la détection d'états futurs et leur signalisation au pilote selon plusieurs degrés

12 Japon
13 Ukraine

d'urgence. Ces urgences correspondent potentiellement à plusieurs degrés d'autonomie de la part du système afin de déclencher la ou les actions nécessaires. CASSY (Onken, 1997) repose sur l'assistance à la prise de décision (par le pré-traitement d'informations de nature stratégique) et la supervision (de la configuration de l'appareil et de l'état en cours). CATS (Callantine, 2001, 2003) développe un ensemble d'attentes à propos des états futurs du vol, sur la base des conditions opérationnelles et des actions du pilote. En cas d'état non désiré, le système d'assistance attend avant de générer une alarme. Par exemple, CATS continue de surveiller silencieusement l'évolution des autres paramètres de la situation de vol afin d'évaluer leur potentiel de résolution de l'état non désiré. Enfin, GHOST (Dehais *et al.*, 2004) est ciblé sur l'évitement des erreurs de fixation. Il compare le plan de vol et les actions de l'équipage au paramètres de l'avion. Les actions qui visent à maintenir le plan de vol en dépit de conséquences défavorables (ex : manque possible de carburant) sont signalées et des contremesures sont proposées (ex : retour à l'aéroport de départ).

A l'exception de GHOST (Dehais *et al., op. cit.*), l'architecture générale des solutions d'assistance à la décision présentées dans ce chapitre repose sur une comparaison des actions humaines à une hiérarchie de plans et au contexte opérationnel. La combinaison de ces deux sources d'information (plans et contexte) est le point de départ du support à la décision et aux alertes que ces systèmes produisent. En particulier, une telle architecture permet de tenir compte des différentes possibilités d'effectuer une même action, et de la position de cette action dans le temps. Pour certains de ces systèmes, ce n'est qu'une fois les possibilités d'action épuisées, ou après qu'une butée temporelle a été atteinte, que des recommandations ou alertes sont générées. Bien que cette architecture ne se prête qu'imparfaitement à la prise en compte de variations ou d'improvisations de la part des opérateurs, elle permet de construire des attentes vis-à-vis des conditions opérationnelles futures. Ce sont ces attentes qui sous-tendent la nature proactive des systèmes d'assistance à la prise de décision et qui permettent de supporter la nature anticipatrice de la cognition humaine.

De par leur architecture centrée sur le contexte et des librairies de plans, ces systèmes apportent une réponse possible à la variabilité de la performance humaine, dont certains facteurs sont identifiés dans la note de synthèse, à savoir:

- la mixité des niveaux d'expertise dans une équipe d'opérateurs;
- les défaillances cognitives dans le pilotage de systèmes dynamiques critiques;
- l'adaptation des actions humaines au contexte dans les systèmes socio-

techniques.

C'est a) identifier ces trois facteurs, b) expliquer leur rôle au regard de la performance, et c) trouver des pistes possibles quant à la meilleure intégration de leurs effets dans le pilotage des systèmes, qui ont constitué les fondations de mon travail de synthèse.

Conclusion générale

Dans cette thèse, en écho aux trois facteurs de variabilité identifiés plus haut, j'ai passé en revue trois types de systèmes impliquant des opérateurs : statique, dynamique et socio-technique. Ces types de systèmes tentent de rendre compte d'une progression dans le niveau de complexité dans lequel les humains sont amenés à agir. Sur le plan de l'analyse, chacun de ces niveaux possède des caractéristiques propres qui sont incluses dans le niveau de complexité supérieur. Cependant, la complexité augmentant, la granularité de l'analyse augmente compte tenu de la plus grande largeur des questions rencontrées.

La rencontre des facteurs de variabilité et des des types de systèmes étudiés m'a amené à discuter de quelques grandes dimensions de la cognition humaine:

- *L'interprétation*, au travers des activités de diagnostic et de dépannage. Cette dimension m'a permis de mettre à jour des aspects reliés à l'expertise et aux heuristiques, toutes deux influençant l'interprétation de l'information.
- *L'anticipation*, au travers des activités de conduite des systèmes dynamiques critiques. Cette dimension a permis de rassembler et discuter d'aspects tels que les conflits cognitifs, les confusions de mode ainsi que les interprétations erronées des états du système.
- *Les variations* de procédure dans les systèmes socio-techniques. Cette dernière dimension couvre notamment des aspects tels que les violations et les *workarounds*.

Sur le plan scientifique, au-delà de mon positionnement plutôt centré sur les processus cognitifs (par définition internes à l'individu), je voudrais rappeler l'impact des facteurs de performance (externes à l'individu) sur la fiabilité humaine. De mon point de vue, ces deux approches sont très complémentaires.

Sur le plan épistémologique, j'ai listé 3 limites à l'identification des facteurs de variabilité de la performance humaine : le manque d'un modèle précis de la performance humaine, les perturbations du comportement induites par sa mesure, et la transformation de la tâche pendant son étude. Parmi ces trois limites, je retiens le manque d'un modèle précis comme le frein le plus urgent

à lever au regard de la compréhension et de la prédiction des activités humaines.

Un aspect sur lequel il convient de revenir est celui de l'erreur humaine. Cette thèse a essayé de s'en éloigner le plus possible dans la mesure où le jugement que ce concept contient n'apporte que peu d'éclairage sur les causes du comportement. C'est la raison pour laquelle des termes tels que défaillance (plutôt qu'erreur) et variation de procédure (plutôt que déviation) ont été utilisés. Du point de vue scientifique, l'erreur n'est pas extrêmement utile à modéliser puisque c'est à la fois un jugement et une conséquence. Ce sont donc les causes du comportement défaillant, c'est à dire les conditions dans lesquelles ce comportement prend place, qui portent en elles les déterminants de la performance et qu'il est productif de modéliser.

Il faut également ajouter un dernier éclairage sur les contenus de cette thèse. Bien que la cognition humaine, par exemple dans son implication dans le contrôle des systèmes, soit un des nombreux éléments actifs de la sécurité industrielle, cette thèse ne porte pas directement sur la sécurité. Mon ancrage est plutôt ergonomique : il s'intéresse tout d'abord à la performance humaine au travail. En élargissant cet argument, l'ergonomie elle-même ne s'intéresse pas directement à la sécurité ; ce serait même plutôt l'inverse. En effet, ce sont les acteurs de la sécurité industrielle qui sont aujourd'hui demandeurs des compétences des ergonomes.

Enfin, sur le plan de mon projet professionnel, j'entrevois trois pistes appartenant à trois niveaux différents : m'impliquer plus activement dans la recherche sur les assistants à la prise de décision proactifs, continuer à travailler dans le domaine de la performance humaine et enfin contribuer à la maîtrise des facteurs humains et organisationnels de la sécurité industrielle (notamment avec des partenaires industriels).

Page left blank intentionally

Introduction. Researching Human Performance

My position within our scientific world

From my very first publication (Besnard & Channouf, 1994), my view of human performance in general is a rather deterministic one: human behaviour is determined by several factors that science aims to discover. Whether the data on human behaviour comes from experimentation, a series of field observations or second-hand reports, one of my concerns is to understand as much of the context as possible. The point in operating this way is important to me because I believe human performance at work is heavily determined by the prevailing conditions (Hollnagel, 1998). This is especially true when one studies human performance in the workplace (as opposed to within the laboratory). Indeed, operators within the same factory, workshop or team will often display a common culture and practices that tend to standardise over time, thereby dampening some individual sources of variability of performance. What is then left to fluctuate the most are the conditions of this performance.

Another dimension of my scientific position has to do with the category of problems I study. Science is composed of a mixture of fundamental, timeless issues (for which a prospective approach is needed) and applied, concrete, present questions (of which industry is a rich source). For instance, human memory has been studied for many decades (see, for instance, the "magical number seven" from Miller, 1956 or the classic works from Ebbinghaus, 1885, or James, 1950). In parallel with such a long research history, some very specific questions are raised by industry. One of them is the quantification of human reliability. Because my personal preference goes for scientific (sometimes fundamental) questions studied within the field, I am interested in both types of research (fundamental and applied). What I try to produce

are responses to problems that are scientifically sound, for which I choose foundations from existing work, and which might be of some interest to industry. Ideally, I then attempt to identify the conditions in which the result of my work can be generalised, in the form of recommendations, for instance. As Bisseret (1988) pointed out, thinking this way forces one to consider not so much which problem to study or not, but how to study it, with what tool, and with what expected outcomes.

To some extent, making industry and fundamental science meet is what is attempted here. Therefore, I will address a number of cognitive phenomena, but in a rather contextualised way. Each time, I will try to use examples from industry in order to show how concepts come to life at the workplace, in operators' actions. I will also attempt to demonstrate that human performance is linked to variability, the latter being linked to the conditions in which operators do what they do. This topic itself will be discussed at length. Also, the factors underpinning this variability will constitute a significant part of my reflections.

The quest for the factors of human performance

Discovering the factors that impact on human performance is not a simple enterprise. The factors affecting human work (as well as their interaction) are numerous and diverse in nature, and lead to variability. The latter can be treated as a synonym for unpredictability of performance but it needs not be so. The failure to predict human performance precisely has more to do with the failure to identify and model factors of variability than the variability itself. In domains other than behavioural sciences, the disciplines that have managed to model variability are the ones that are now in a position to anticipate specific responses from even complex sets of inputs. In weather forecasting for example, we are now all used to relatively accurate five-day forecasts, whereas less than a quarter of a century ago, twenty-four hours was as far as weather models could see.

If, as a principle, variability should not be taken as a synonym for unpredictability, it still leaves the issue of dealing with variability unaddressed. Let me spend some time on this question. From a comparative point of view, it is amazing to see how fast the predictive power of weather forecasting models has grown, whereas psychology as a scientific discipline dates back to 1890 with the seminal works of William James (1950; reprinted). And despite this long history, researchers in psychology at large have not been able to produce a predictive model of performance whose accuracy can be measured in units smaller than orders of magnitude. This is

what I would like to call the first limitation of our prediction capacity. It can be traced back to some simple causes. For instance, there is little hope that all factors of variability that impact human performance will ever be known exhaustively. But even so, mastering the effects of the various combinations of these factors might still be a significant challenge. This complexity limitation is essentially a methodological one. Indeed, if a team of researchers carried an extended and systematic program of performance observation and analysis under a large number of conditions and for a large number of tasks, then in theory, a solid list of sources of variability could be extracted and their possible interactions quantified. Psychology would then be following the steps of weather forecasting: if the original data set is big enough, then computing power can help extract the model.

If the first limitation can be overcome, one might then face a second one: accuracy of measurements. Coming back to my example of weather forecasting, measuring the temperature of the air or atmospheric pressure creates very little disturbance to the weather itself. This allows one to measure unbiased phenomena. When it comes to measuring human behaviour *in situ* so that ecologically rich data is collected, the act of getting the data has an effect on what is being measured. For instance, studying human sleep can involve sleeping with electrodes attached to the sleeper's head, which disrupts sleep itself. Similarly, people being interviewed, might not share their opinions fully because of such factors of breach of confidence or fear of judgement. These effects can be worked around so that their amplitude is diminished. However, among all the data gathering techniques used in behavioural sciences, it might be that most of them do not make it possible to measure anything significant about human behaviour without influencing it at the same time.

If the first and second limitations could be overcome, one would then obtain a predictive model of human performance capable of predicting the level of accuracy of a number of actions under a number of conditions. However, there would still be a third limitation to address. Indeed, contrary to air temperature or atmospheric pressure, human activity often takes place within a task, for which one defines goals and manage resources. In short, humans do things to reach an objective that is subject to a number of constraints (time, cost, etc.). Measuring activity in order to extract the input data to a (hypothetical) model can therefore conflict with the task itself, by introducing new constraints that do not belong to the initial task and that the operator has to accommodate. An example of such a constraint is asking an operator to think aloud, so that the line of reasoning can be recorded on tape, for

instance. When an extra task of this type is added to the operators' initial task, they have to deal with a new, additional constraint, thereby adding to their initial, natural activity. If you think of a surgeon in a middle of an operation, describing verbally all the actions taken, the reason why they have been taken, and their anticipated effects, the task no longer is an operation. it has become a verbal description of an operation that is being performed. This disturbance can range from slowing down the execution of the natural task to preventing it from taking place.

I have now listed the following three limitations to human performance prediction:

- the lack of a precise model;
- the disturbances introduced by measurement of behaviour on the behaviour itself;
- the transformation of the task caused by the study of this task.

Owing to these limitations, it might be unlikely that cognitive sciences will soon produce a model allowing the precise prediction of the level of human performance on a given task in a given set of working conditions.

This question is, of course, open for debate since the limitations might combine in such a way that some exceptions appear, creating cases where performance data can be gathered without bias. Identifying where these exceptions lie and what they would allow in terms of performance assessment might deserve some methodological reflection. Another potential point of debate touches upon whether human performance as an object can be modelled precisely. The assumption this thesis makes, without questioning it seriously enough maybe, is that it is. The last point of debate I would like to mention is whether such a precise predictive model is needed at all. Instead, it might be the case that enough is known of the gross determinants of human behaviour for one to know the classes of situations where human performance is likely to degrade.

For the time being, whether we like it or not, classes of human performance is the only basis we have for predictions. Starting from this assumption, the work reported here will address human performance from the point of view of the *type* of situation in which it took place, thereby leaving out the difficult questions of the choice and exhaustiveness of the measurements performed to assess this performance.

Three tasks and their relation to the performance of systems

Despite classes of human performance only providing a coarse granularity for

prediction, they can still provide a useful basis to prescribe improvements for the cognitive conditions of some situations of human work. Indeed, cognitive ergonomics has not penetrated all industrial domains at equal depth, thereby sometimes leaving highly discrepant performance conditions to cohabit. Beyond this rather blunt statement lies the question of the origin of this situation. It seems to me that the driver of the integration of cognitive ergonomics into the workplace is the acknowledgement that the system's performance cannot be improved by technical artefacts alone. This is very much the case of dynamic critical systems (e.g. commercial flights in western countries), an industry that shows such a high safety record[14] that safety levels are now expected to increase mainly from better practices in terms of human factors. And because the mainstream human factors issues have already been integrated with daily practice, any improvement on the system's performance relies on fine-grained issues. Conversely, commercial flights as run in countries such as Russia or on continents such as Africa show a lower technical reliability level. These are cases where the system's poor performance cannot be dissociated from the difficult financial conditions of operation.

Dynamic, critical systems have one particularity that has not been highlighted yet: their performance relies on both cognitive ergonomics (e.g. of control panels, cockpits, etc.) and soundness of organisational practices and culture (e.g. safety culture, training, etc.)[15]. This combination seems an obvious prerequisite for optimal system performance but the question is more one of proportions than mere presence or absence. Indeed, any industry needs some equilibrium in these two elements (ergonomics and organisational soundness) since they account for the sharp end and the blunt operations. If one now treats these two ends as the boundaries of a continuum, the operation of dynamic, critical systems, would probably stand somewhere half-way. The performance level in other situations, such as workshop activities (e.g. troubleshooting technical faults), could be explained to a large extent by rather simple psychological mechanisms, with the organisational dimension being of little relevance. Conversely, some situations could be best described in terms of organisational practices and safety culture, where humans essentially play a role of adaptation and compensation to contingencies.

These three views are summarised in Figure 1.

14 The modern commercial jet fleet shows an accident rate of about 1 per million departures (Boeing, 2007).

15 To be exhaustive, it should be added that regulators also have an impact on systems' performance, through the definition of safety targets, or requirements for certification, for instance.

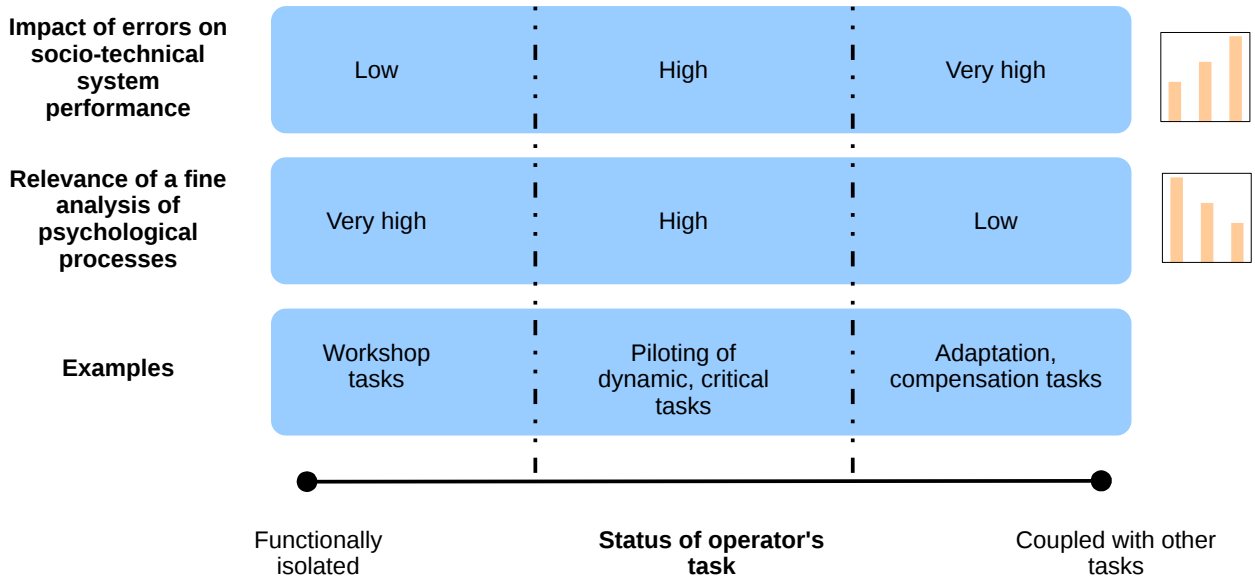


Figure 1: A continuum of tasks and their relative statuses

The figure above describes the range of situations reviewed in this thesis. This review will be done using two interwoven dimensions, the balance of which will change through the document:

- *Fine analysis of psychological processes,* the granularity of which will progressively increase to encompass more and more task-relevant systemic factors;
- *Impact of human failures on the socio-technical system performance*. The focus on system performance (and safety) will increase as the thesis progresses.

Contents and structure of the document

Concretely speaking, this thesis is an attempt to build an overarching view of cognitive ergonomics at different levels of granularity for which I will attempt to highlight my scientific contributions. With the view that cognitive ergonomics is the discipline that studies the relation between mental activities and task performance (preferably in real work settings), the objective here will be to highlight my achievements with regard to understanding human cognition. I will attempt to reach this objective by focusing on the interaction between humans and three categories of systems:

- *Static systems*. At this level, the interest is to study some cognitive mechanisms in such tasks as problem solving in general and

troubleshooting in particular. The view is essentially an individualistic one, where the focus is on cognition with little concern for the work environment. That said, troubleshooting is heavily involved in the interaction with complex, dynamic, critical systems, and this will lead me to shift one level up in the socio-technical continuum in order to consider different types of activities.

- *Dynamic, critical systems*. These systems pose a number of problems because of the number of functions they are composed of, combined with the fact that they are fast-paced and sometimes leave little room for recovery in case of degraded conditions. In interacting with such systems, ergonomics-related problems touch upon symptoms misinterpretation, cognitive conflicts, mode confusions and misleading interfaces. At this level, the focus changes because systems and operators are engaged in a dialogue, one responding to the other through a variety of conditions and interfaces. This interaction-centred vision does not take into account the contribution of humans to the performance of the wider system in which they operate, which calls for a last level of analysis.
- *Socio-technical systems*. At this level, I will show how humans contribute to the life of systems, as composed of technical and human components. These systems can have their performance enhanced or degraded depending on how humans understand their task and adapt it to the prevailing conditions. At this point, the view will become slightly more systemic than previously in the sense that the ergonomics-centred analysis of performance will have the system as a focus point.

The three levels above will each be allocated a chapter of this document and will all be looked at from a cognitive ergonomics angle. However, beyond the ergonomic analysis of each level, I will also try to pinpoint challenges in the domain of human performance. This will be done in a specific section at the end of each main chapter. In doing so, I will highlight some fundamental sci a pos g vie g iss e per I wo r rea

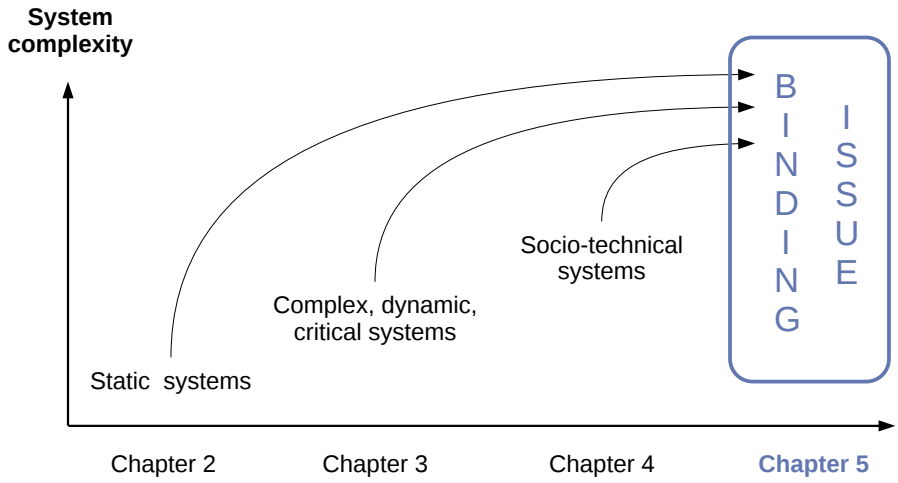


Figure 2: Structural positions of the various chapters

Chapter 1. Human Cognitive Performance In Static Systems

One activity that human cognition is supremely good at is interpreting its environment, that is ascribing meaning to it. This interpretation is a task that we all perform constantly, from having a phone conversation to finding one's way on a map, from reading a recipe to making a gesture, to pointing towards an object, etc. This search for meaning is something that cannot be easily turned off as if it were an option with or without which one decides to see the world. The relation to the world is a continuous cycle. Humans see object properties in this world, which in turn influence interpretation, which finally influences our actions (Neisser, 1976). The loop closes when actions, and the underlying intentions, guide and select what object properties one needs to see. From a learning point of view, because the acquisition of knowledge never really stops, one knows more and more about the properties of the objects in the world. Following Neisser (*op. cit.*), interpretation therefore becomes continually richer and richer. This, in turn, causes intentions to be defined better (e.g. by offering a more complete set of any object's properties to select from when triggering an action). Ultimately, this [selection-interpretation-action] endless cycle causes both the actions one plans for, and the knowledge one has, to depend and influence one another.

In this chapter, the focus will be set on the interpretation of information. I will adopt this focus for the study of a certain class of systems that I call static. With this word, it is meant that there are situations and systems whose pace of evolution is slow compared to what human cognition can process. If one takes the example of a mountain, geologists will probably say that it is a live, moving object. Now, if one thinks of a warden whose job would be to report all geological events on this mountain, that person would not have much to do over their lifetime. To summarise, the mountain is not really static, but its pace of evolution is relatively slow. If one reuses this example for cognitive

activity, one can see that there are systems that humans interact with that can be treated as slow-paced. That is, they change much more slowly than cognition can handle, or than the task requires. For instance, mechanics operators, when troubleshooting a running engine, can take virtually all the time they want to find the fault. Of course, there are faults that are more time-dependent than others (e.g. those related to overheating). However, in most cases, the system (the engine) will continue to maintain its state (running) for as long as there is petrol and air to feed it. The fact that this engine will show very limited exchanges with the surrounding world and other processes also simplifies the task of interacting with it. Therefore, let us accept that this system is relatively slow-paced, and depends on, or is influenced by, or interacts with a very limited set of external factors. This is what is called static systems in this chapter.

1.1. Knowledge and the interpretation of the environment

Looking at humans as cognitive entities, it is hard to reject the idea that we all use a combination of bottom-up and top-down reasoning processes. However, I'd like to suggest that knowledge impacts on perception itself, or to be precise, on how perception is guided through the interaction with the world. This implies that two persons with different levels of experience and knowledge would interpret the same situation differently. This might well be the case. Consider the following famous tale. Three men are walking towards each other in the savanna in a moonless night. They each walked into an object that they describe to the others. One says he is holding a large tree trunk. Another says he is holding a powerful snake. Another asserts he is holding a thin branch. Little did they know that they were each holding different limbs of an elephant. This allegory says that a) an object might only be interpreted partially, on the basis of what one can experience of it, and b) this interpretation can be distorted by experience and knowledge limitations.

Of course, giving a tale as an example could be seen as a hint that my point is somewhat abstract and without tangible relation to the so-called real life? Let me fix that. The following story is the real-life version of the above tale. One morning in October 2007, I was driving to work and I pulled over to help a lady who was trying to change a wheel on her car. I got out of mine to see how I could make myself useful. As I was jacking up the car, she explained that she probably had driven over a heap of metal shavings lying on the road, because that is what was attached to the rubber of the flat tyre. Eventually, these metal shavings must have punctured the tyre, she said. I helped her change the wheel, drove away but forgot to thank her for her wonderful story.

Indeed, she demonstrated that the way one looks at the world and try to understand events is heavily dependent on knowledge. In this particular example, I assume the lady might not have known that tyres are made of a metallic structure of interwoven threads, coated in rubber. Therefore, when these so-called metallic shavings showed, caught into the rubber of the flat tyre, they had to come from the road. What had actually happened is that the lady had been using the incriminated tyre for much too long, to the point where its metallic structure was exposed.

From this example two issues can be highlighted:

- *What is happening in the world is a matter of knowledge*. The so-called reality is something that is for the observer to build, as opposed to an objective given piece of data. Since human cognition is not a perfect reasoning machine, it follows that reality as interpreted can be distorted;
- *One might misinterpret the cause of an event,* even when presented with it. Reasoning itself should not be seen as a logical inference engine that will process any problem equally well. Just like a computer does not work without software, reasoning performance on real-world problems heavily relies on experience and knowledge. I will come back at length to this point, through numerous examples of flawed interaction with various situations.

The two preceding points, however reductionist they might be, show that human cognition is not flawless but nevertheless is the main instrument to interact with the world. What is perceived from the environment is under the constant influence of interpretation processes. These interpretative processes, in turn, impact on how the objectives of a task are defined and how decisions and plans are established (see Weick *et al.,* 2005, on the related topic of sensemaking).

1.2. Interpretation as a profession: troubleshooting

With my short example of the unfortunate driver, I only wished to demonstrate how knowledge (or lack of thereof) can influence the meaning one ascribes to the environment. That example is certainly not fully representative of more complex situations such as the ones found in highly-technical domains. However, the cognitive mechanisms by which meaning is built are similar. And building this meaning (in other words: interpreting), however insignificant it might seem, is absolutely crucial in an enormous amount of activities, from daily problems (e.g. finding why the TV remote

control no longer works) to more complicated technical situations (e.g. why an aircraft would climb when pilots try to land it). What would then be interesting to do is look beyond the interpretation process itself and see how it impacts on the chosen actions, what behaviours can emerge and how these can challenge human reliability.

To explore this idea, it would be ideal to study real cases of a complete activity that encapsulates a chain of processes such as:

- finding relevant information;
- interpreting this information;
- establishing an action plan;
- carrying out the action plan.

Such an activity exists; it is called troubleshooting (or diagnosis in the medical domain). Generally, its purpose is to find the cause of some unwanted behaviour in a system (e.g. why a car doesn't start, or why a patient is covered in red spots) and carrying out a set of correcting actions. Often, it is determining what causes the fault or illness that is the most demanding and interesting, from a cognitive point of view. Fixing the problem is generally a matter of logistics, supplies, following procedures, and implementation (such as manual skills in the case of surgery).

Because in this section a number of terms will be used, I first need to establish how I will use them. Diagnosis as a word stems from the combination of two Greek words: *dia* (by, through, separation, distinction) and *gnosi* (knowledge). The resulting meaning is one of gaining knowledge by discriminating between facts, possibilities or cases. Diagnosis refers most often to the medical domain and to a practitioner's intention to pinpoint a cause for a given set of symptoms. This is usually followed by a treatment, which is supposed to be the solution to the patient's problem. Technical domains show the same distinction between the processes of identifying the cause of a problem and fixing it: they are called fault-finding and troubleshooting, respectively.

In this thesis, a distinction will be made between the context-free, reasoning process that interprets symptoms, and the context-dependent activity of fixing problems. The term diagnosis will then refer to the mental activity that formulates candidate causes to problems, and treat troubleshooting as the wider task whose purpose is to find a physical faulty component, and carry out the required operations to locate it and fix it.

At the end of this quick preamble, I am now going to explore troubleshooting from a psychological point of view and try to analyse the process through which understanding occurs, and how it guides actions. Namely, symptom

interpretation, as well as the details of *doing* troubleshooting will be addressed. I will follow with a detailed analysis of expert troubleshooting in technical domains and conclude with the role of troubleshooting in decision making and in control and supervision tasks. With this perspective, troubleshooting will be treated as a task where a corrective action first requires a mental representation of the current state of a problem and the desired state[16].

1.3. Symptoms interpretation

In (clinical) diagnosis, the perceived symptoms are typically matched with a class of disease or a known cause (Norman *et al*., 1989). From this standpoint, diagnosis can be seen as a classification activity operating along the normal/abnormal dimension. In clinical diagnosis (Ben-Shakar *et al*. 1998) and especially medical problem solving, the study of diagnosis is the focus of a large body of research (Medin *et al*., 1982; Kuipers & Kassirer, 1984; Boshuizen *et al*. 1991; Brooks *et al*., 1991; Hassebrok & Prietula, 1992; Mumma, 1993; Arocha & Patel, 1995; Custers *et al*., 1996; Simpson & Gilhooly, 1997).

Troubleshooting, in contrast can be seen as an activity whose objective is to identify the causes of technical system states assessed as abnormal, and to understand the causes of the observed symptoms (Cellier *et* al., 1997). Troubleshooting, like most kinds of reasoning, is dependent on the operator's mental model: the diagnosis process is triggered only when a state is perceived as abnormal and a correction is required. For instance, among inexperienced computer scientists, the information that underpins the detection of a deviation is an error message from the computer (Allwood & Björhag, 1990). This is the stage where the operator needs to generate hypotheses about the observed malfunction in terms of a change in the system (Milne, 1987). The test of the hypotheses is aimed at isolating the faulty component or function, and is supported by the information the operator has extracted. This is also how Mozetic (1991) sees diagnosis, i.e. as an identification of the components or functions that, by behaving abnormally, account for a discrepancy between the actual behaviour of a system and its expected behaviour.

One could be tempted to decompose troubleshooting into a well-organised and clearly-defined chain of operations, from information gathering to resolution. This view could be adequate for describing how a machine would

16 Along with constraints and possible actions, this representation is what Newell and Simon (1972) referred to as the problem space.

perform troubleshooting. But as far as humans are concerned, they use heuristics and [SYMPTOM-CAUSE] associations in an almost automatic manner. These powerful associations rely on rules that bind together (in the operator's memory) a problem and its solution on the basis of past occurrences. These frequency-based strategy is at the core of human activity, from problem solving to process control. It is only when the case at hand defeats these strategies that operators resort to what Rasmussen, (1986) termed a knowledge-driven strategy.

In addition, operators do not need to understand what is not relevant for them to reach their objectives (Rasmussen & Jensen, 1974; Amalberti, 1996). In troubleshooting, some functional knowledge of the system is often required, but a detailed knowledge of how each function is physically fulfilled is often not required (Schraagen & Schaafstal, 1996). I could take my own example as I write these very lines. I do not need to understand the electronics of my computer or any programming language to use my word processor. From this point of view, the position of Samurçay and Hoc (1996), for whom one of the dimensions of expertise is being able to handle several levels of abstraction, is somewhat remote from a practical view of human reasoning. Indeed, practically speaking, human cognition is best described in terms of sequences of simple, local, and rapid decisions (Rasmussen & Jensen, 1974).

Isolating the relevant characteristics of a faulty system does not merely rely on a passive perceptual process. Indeed, information gathering is guided by the operator's knowledge and objectives. In the case of troubleshooting, the perceptual processes are guided by some pre-existing knowledge about the normal and abnormal system states. Allwood and Björhag (1991) have demonstrated that the performance of novice computer scientists at bug-finding can be enhanced not by teaching them more about about computing science but by just training them how to understand information flows in programs, interpret error messages, and more generally determine what information is relevant for the case at hand. This demonstrates that deriving a symptomatic value from collected information implies an active search and interpretation. It is then easy to see that the identification of symptoms among a set of functional characteristics is already, in itself, a sign of the existence of knowledge of one's working environment (Ohlsson, 1996). This view is similar to Klein's (1997) model of recognition-primed decision making whereby people with some experience of a domain can sometimes base their decisions on a set of situational cues, with potentially very little analytic processes. A somewhat similar process can be found in reasoning by analogy:

the properties of a new problem are used to retrieve and adapt a solution stored in memory from previously solved problems (Bonnardel *et al.*, 2003).

The identification of symptoms is central to diagnostic reasoning. It is the first indication of an abnormal behaviour and constitutes the first stage of the troubleshooting process (Duncan, 1985). The clarity of symptoms eases reasoning. In industrial domains, it is often obvious that there is a problem. Indeed, the process produces an output whose measurable properties or defects are indicators of the status of the process itself. In medical domains however, the identification of symptoms can be more difficult due to their potential vagueness (Schaafstal, 1993). If operators have incomplete knowledge about the relevance of a particular set of symptoms, or if sorting them relies on erroneous assumptions, then filtering the data will be based on irrelevant criteria, or will be biased by some intrinsic characteristic of a symptom (e.g. saliency, familiarity, etc.). In the end, this process of symptom selection determines the outcome of diagnosis to a large extent. Indeed, if operators make the hypothesis that one source of information is more useful than any other, their attention will be focused on this set of symptoms even if it is not the most relevant (Spérandio, 1987), and diagnosis will operate on that basis.

Using a more formal wording, the elementary task in diagnosis is to allocate entities (symptoms) into some appropriate class (Caverni, 1991). Classification has been acknowledged as a central feature to many cognitive activities (Bruner *et al.*, 1967). In diagnostic reasoning, it is done on the basis of an object's properties (the symptoms) that are compared to the prototype of a category of fault, or treated as an input to a decision rule such as [IF-THEN] (Smith & Sloman, 1994). The mere application of these canonical rules to the case of medical diagnosis can reveal itself to be flawed if the practitioner is not able to correctly assess the various features specified by the the rules (Brooks, Norman & Allen, 1991). One is then likely to perform a diagnosis where symptoms are made to point to a particular disease instead of a class of disease. Novice practitioners might be more prone to this kind of behaviour (Arocha & Patel, 1995)[17].

1.4. Troubleshooting strategies

This chapter is about troubleshooting. Finding the cause of some malfunctioning can be done in many different ways, depending on who does

17 Beyond the problem of the selection of a rule on the basis of problem or situation features, the potential brittleness of rules in exceptional cases is a topic we will return to further in this document.

it, his or her level of experience, the configuration of the system, the resources available, and so on. It would therefore difficult not to address the issue of strategies. As shown in the next sections, some are typically logical and rely on the structure of the system at hand. Conversely, others are based on empirical probabilities of failure of given components. The choice of one or the other is often a matter of objective: operators mentally represent the system differently depending on what their task is and what kind of action strategy it implies. For instance, in the case of the piloting of a nuclear reactor, Joblet (1997) isolates 9 different piloting strategies, each corresponding to a particular representation of the systems. For instance, event-based strategies (which I call symptomatic) relate to a representation of the system in terms of causal functions whereas a circuit-based approach relate to a more topographical representation. The relation between the type of mental representation of the system and the strategy also applies to troubleshooting. However, these various types of strategies are often combined within the task (for instance, the representation can be topographic at one level and symptomatic at another).

1.4.1. Topographic reasoning

According to Patrick (1993) and Rasmussen (1986, 1991), topographic reasoning is based on a structural view of the system, represented as a set of nodes linked together by physical artefacts such as pipes, wires, etc. This vision tells the operator how faults can propagate through the system according to a map of the location of physical elements. For instance, when one needs to find out why a car’s wipers no longer work, he or she can start with testing the wipers’ engine and trace back all the way, step by step, to the control knob by the steering wheel. Novice operators will almost exclusively deploy this type of strategy since their limited experience does not allow them to resort to a frequency-driven strategy. Thus, novice troubleshooting is characterised by a chain of small reasoning steps supported by a structural representation of the system.

Topographic reasoning is not system-dependent and does not assume any knowledge of fault frequencies on the operator's part. This makes it extremely versatile and easy to deploy as long as a structural map of components is available. This strategy however is not economic in terms of resources (especially time): frequencies of fault are not used or known and cannot guide the reasoning towards a particular area of the system. It follows that topographic reasoning is made of a large number of steps (Schaafstal,

1993; Fath *et al.*, 1990), an approach that can be faced with combinatorial explosion problems if the system receives inputs from other sub-systems (which may be faulty themselves).

1.4.2. Symptomatic reasoning

This is a probabilistic-like strategy where the operator looks for the fault that is empirically known to be most likely to trigger the symptoms at hand (Patrick, 1993; Rasmussen, 1986, 1991). With this strategy, troubleshooting is no longer guided by the structure of the system but by the meaning (in terms of potential fault) that the operator can assign to the symptoms. Experts tend to deploy this type of strategy first since it allows them to very quickly narrow down the set of potential faulty candidates with a very limited cognitive effort. Let me reuse the example of the faulty wipers above to make my point. If an expert troubleshoots the problem, they will first try to define the type of wipers' engine and model they are dealing with. They do so to recall a set of typical faults associated with this particular type and hence save time. When observing expert operators, one will notice that they discard testing parts that seem obvious to the layman. Instead, they start their test plan with a component sometimes located in an unexpected location, and whose role in the fault is not immediately understandable. In the case of the faulty wipers, it would not be uncommon to see an expert technician bend over into the engine compartment, reaching for a very small hidden connector which happens to be prone to corrosion. Operating in this way is guided by empirical knowledge of fault frequencies, the latter being built over repeated exposure to similar types of faults.

Symptomatic reasoning, since it is extremely system-dependent, is limited by the number of cases where it can be deployed. However, when available, it is among the most powerful strategies. It is also a very economic mode of reasoning since only a small number of tests or operations is needed before a candidate fault is located. On the downside, this is not a versatile strategy. Operators need to have extensive experience on the system they are dealing with in order to build a workable set of [SYMPTOMS-CAUSE] pairs. Ultimately, symptomatic strategy is a pattern-matching strategy where the observed symptoms are compared to potential causes in order to find a match (Sanderson, 1990).

1.4.3. The split-half test

I noted earlier that the topographic reasoning could lead to combinatorial explosion. Although it is a distinct possibility, there is a solution that is still

based on logic, and that enhances troubleshooting performance: split-half. This test aims at splitting the system's components into two sets: one that contains the fault and another that does not. For instance, if one wants to know why a CRT[18] computer screen does not display anything, one can replace the screen with another one. If the new screen works, then the faulty component is the original screen. On the other hand, if the new screen still does not display anything, the cause has to be searched for somewhere else (e.g. graphics card) rather than at the screen level. To understand the interest of a split-half test, consider the following. If one were to deploy a plain topographic strategy, then the test plan for a CRT display fault should begin with the CRT tube itself, then trace back to high tension modules of the monitor, the power transformation unit, etc. Instead, a split-half test can potentially rule out the monitor itself in just one operation.

This strategy has been preferred by authorities on troubleshooting research. For them (Goldbeck *et al.*, 1957; Dale, 1957; Rouse, 1978), this strategy allows one to isolate the faulty component with the smallest number of steps. The philosophy supporting this approach to troubleshooting is one where reasoning essentially relies on deductive methods, seen then as the best way to get the work done (Dale, 1957). An example of a classical experimental network used to study troubleshooting is presented in Figure 3 below. This type of network has been used in laboratory research to study how troubleshooters choose their test points among components on the basis of their ouput signal (in the figure below, 0 and 1 mean absence or presence of a normal output signal, respectively).

18 Cathode Ray Tube

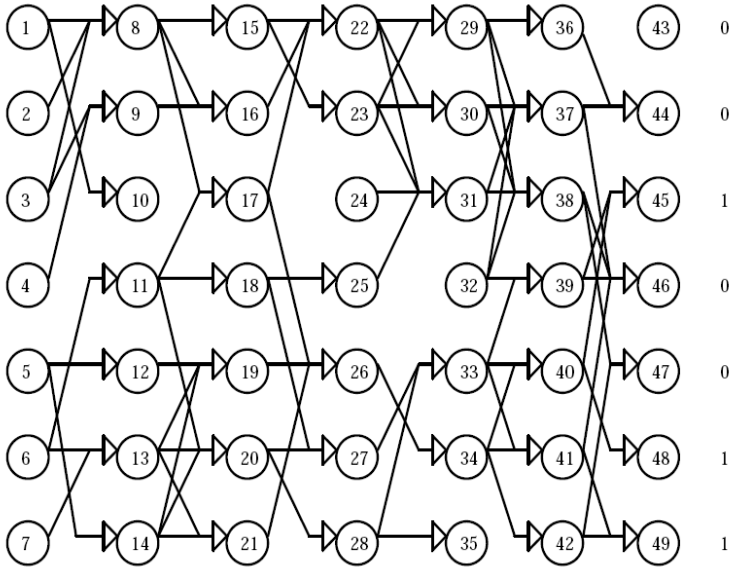


Figure 3: The network from Rouse (adapted from Rouse, 1978)

The logics-based view of reasoning described here is akin to that of Newell, Shaw and Simon's (1959) means-end analysis where problem solving could be modelled (and implemented) as the definition of a solution, and a regressive analysis of the means required to reach it. This logical view is recognised today as one that has allowed to progress on the modelling of human reasoning. However, it does not really account for the numerous frequency-based, symptomatic reasoning behaviours displayed by people at work. As far as the split-half is concerned, Konradt (1995) noted in a study on a computer-controlled manufacturing process, that expert operators only use this test in about 1% of cases. Instead, the strategy that is used most often is one that searches for information on a similar fault that has already occurred in the past, or one that uses documentation, or memory.

1.4.4. Forward, backward, and opportunistic reasoning

Forward reasoning starts a series of inferences from an initial faulty state. The aim of this reasoning is not to find a cause but to identify its consequences. Formally speaking, operators explore an event tree. This type of reasoning is well suited to anticipation whereby the forecasting of the consequences of a fault is performed step by step until the operator finds a component whose behaviour is likely to be disrupted by the fault. Backward reasoning aims at finding a faulty component. It also starts at the same initial faulty state but climbs backwards through a fault tree. A series of topographic inferences then attempts to identify the cause of the faulty

behaviour.

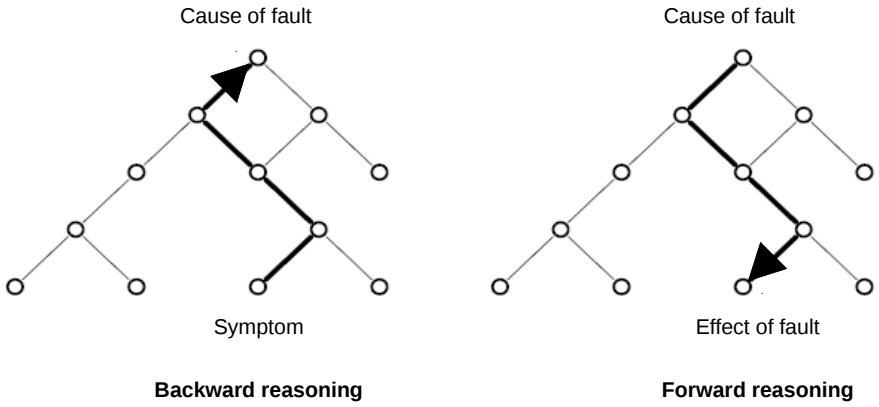


Figure 4: A graphical representation of backward and forward reasoning

An application of a combination of backward and forward reasoning can be found in industrial safety where these two strategies are used to trace the origin of a technical fault and its potential consequences. The backward phase of the reasoning process traces back to the potential sources of a technical fault; this is then represented as a fault tree. Conversely, the forward phase of the reasoning forecasts the effects of the fault on the rest of the system; this is represented as an event tree. The combination of a fault tree and an event tree, with the technical fault (undesirable event) in between, is called a bow tie model (Figure 5)[19] and should be read from left to right. From a cognitive point of view, it is a concrete and graphical example of the directionality of reasoning in fault finding and forecasting.

19 Despite the propagation arrows all pointing towards the same direction, the reasoning that underpins is bidirectional (backward to the causes of the fault or event, and forward towards its consequences).

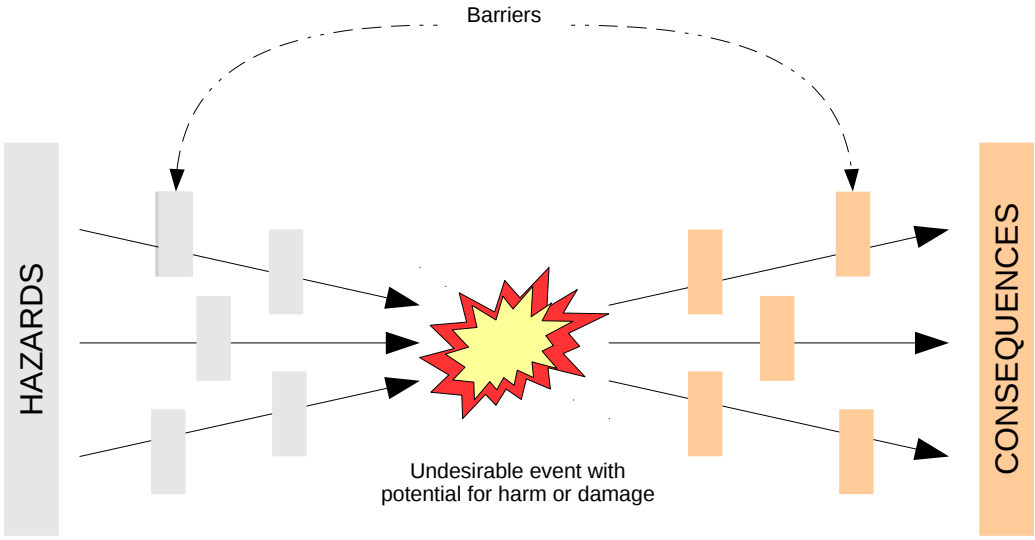


Figure 5: A bow-tie model

Humans also combine these two strategies into what is called opportunistic reasoning, where reasoning steps are implemented according to a pre-defined strategy (Hayes-Roth & Hayes-Roth, 1979; Bisseret *et al.*, 1988) but adjusted by feed-back. It is the evolution of the representation maintained by the operator, fed by intermediary results, that steers the search for the cause of the fault. In other words, operators must maintain a long-term objective (finding a fault) along with short-term goals (finding intermediary results). The split-half test strategy fits this description. An operator attempting to eliminate a set of causes from the test plan will potentially perform repetitive split-halves. One then has to a) remember the effects of these tests and b) use them to incrementally adjust the mental representation of the fault.

Most of the types of reasoning described in this section are typically topographic, and are supported by logical paths, cables, pipes or flows within the system, which make them versatile but (as stated above) subject to complexity limitations. So far, not much was said on the effects of level of expertise on the performance at troubleshooting. It is an important aspect of the topic since work as performed in the workshop generally relies on the performance of expert operators, that is those who have acquired so much experience that their reasoning (at fault-finding for instance) itself has been reshaped. To some extent, experts' reasoning is an oxymoron since most of the time, they do not not really reason. Instead, they remember. As briefly

mentioned in this section, experts recall past occurrences of a similar pattern of symptoms and the associated most likely cause. In decision making, this pattern-based strategy can be found among chess players who typically store and retrieve thousands typical game configurations and and corresponding moves (Gobet & Simon, 1996a, 1996b).

1.5. Heuristic reasoning in troubleshooting in mechanics

Expert operators deploy strategies that do not rely on reasoning as a series of inferences but on imperfect decision rules that tie together a problem and a solution, or a situation and a decision. These rules are not aimed at providing a perfect answer. Instead, they seek some economy of resources, at the cost of imprecision. There are several such heuristics that we all use on a daily basis (Tversky & Kahneman, 1974) but when it comes to problem-solving in situations such as troubleshooting, two of these heuristics account for a significant portion of expert operators' activity: frequency gambling and similarity matching (Reason, 1990). I will get into more detail as to what these heuristics entail in terms of fault-finding in technical systems but for now, let me just state that heuristics are reasoning rules that do not seek exhaustive or perfect answers. They are only ones that are good enough in terms of results, and not too difficult to obtain in terms of resources (e.g. effort, time). This behaviour has been termed *satisficing* by Simon (1957). In other words, experts act on the basis of an intuitive balance between cognitive load and probability of error. Even in risk-critical systems for instance, expert decisions reflect the existence of an operational trade-off where a residual risk is accepted if a given rule provides an acceptable solution in the most common configurations of problems (Amalberti, 1996).

1.5.1. Expertise and reasoning performance

Experience is built on top of many instances of a limited number of problems that are typical of the job. For instance, a medical practitioner will be faced with the same diseases again and again through his or her career. A mechanic will see the same faults over and over, sometimes for as long as 40 years. Over time, mostly variants of the same problems will occur for which solutions will be progressively built and stored in memory. Then, when a problem occurs, it is likely that a variant of it will have been encountered in the past, for which a solution is ready for adaptation and deployment. Only when a completely unknown case occurs will experts have to resort to first principles to solve it.

Technically, a given solution is recovered and deployed each time a problem

displays a set of features that fulfils the triggering conditions for that solution. For instance, for car mechanics, there are only a limited number of causes for a diesel engine not to start. Most likely, one or two plugs will not heat up[20], preventing the fuel mixture from igniting in the cylinder. The troubleshooting process will then start with this potential cause since it has the highest frequency of occurrence. Only when all the plugs have been proven to work will the operator switch to the next most frequent potential fault available in their memory, and so on. It is a cognitive resources saving strategy that underpins this behaviour. Expert operators recall and adapt solutions ordered by frequency or similarity instead of logically analysing problems and inferring causes.

As already said, one of the core features of expertise is an optimal performance at a minimal mental cost. From this point of view, heuristics are not meant to provide the right answer to any problem but instead, under some acceptable uncertainty, to be efficient in routine situations. In troubleshooting for instance, the more frequent a given [SYMPTOM *x* - CAUSE *y*] association, the more likely it is that this association will prevail on the next occurrence of symptom *x*. It is to investigate the strength of such an association, and measure how heuristic reasoning impacted on decision-making that I conducted an experiment on troubleshooting in car mechanics.

1.5.2. An experimental investigation

In order to obtain a tangible and precise assessment of the role of heuristic troubleshooting in real-life activities, a workshop experiment was designed that required mechanical operators to troubleshoot a fully-working engine (Besnard & Cacitti, 2001). The engine was a four-stroke, petrol type, mounted on a chassis which was itself bolted onto a trailer (see Figure 6).

20 On diesel engines, *glow* plugs (as opposed to *ignition* plugs) preheat the combustion chambers in order to enable the first explosion that will start the engine.

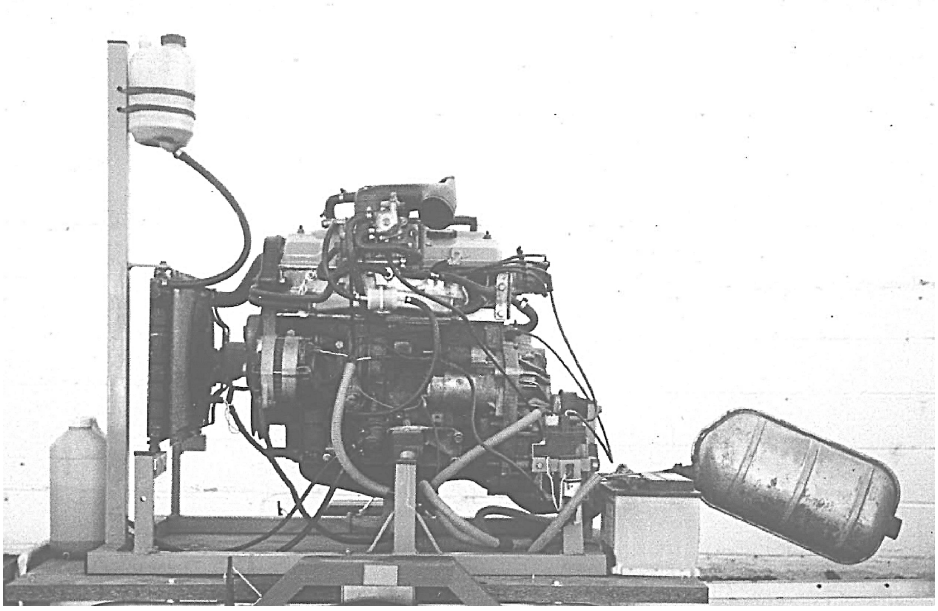


Figure 6: The engine used as the experimental device

With a petrol, carburettor-type engine (which is the kind of engine used for the study), the following applies to all the cylinders, although at different times. First, the piston descends in the cylinder and sucks in a mixture of air and petrol via the intake manifold[21]. Then the intake valve closes and the piston rises to compress the mixture (second stroke). At the top of the piston motion, the plug sparks and causes the mixture to explode. This explosion strongly pushes the piston down: this is the motor stroke (third stroke). Then the exhaust valve opens and the piston rises again to push the burned gases out (fourth stroke). The exhaust valve then closes, the inlet valve opens and the cycle starts again.

In the experiment, the fault was caused by an aluminium plate obstructing the intake tube of cylinder #4. The plate caused the following immediate symptom: the engine vibrated heavily due to the explosions being unbalanced across the four cylinders. This symptom can have several natural causes: a hole in a piston, a leaking valve, or a leaking piston ring, the most frequent of all being in fact a faulty electric component causing the spark plug not to spark.

Since expert mechanics empirically know the relative frequencies of causes of fault, the hypothesis was that they would deploy a heuristic, frequency-driven strategy and test the electrical causes first. Novices were expected deploy a more logic-driven or topographic-driven strategy. Namely, experts were

21 A moulded metallic piece that brings the four intake tubes together.

expected to detect a missing explosion in the engine cycles, work towards locating the faulty cylinder at earlier stages of the test plan, and prioritise electrical causes higher than novices would. With this set of predictions, the operations to observe and measure were attempts to locate the faulty cylinder (#4) and subsequent operations carried out on it.

The significant results (see Figure 7) show that when compared to novices, experts carry out no operation at all before attempting to locate the faulty cylinder (this is what pulling a plug cable allows), pull the plug cable #4 more often, and altogether perform more operations on cylinder #4 than novices.

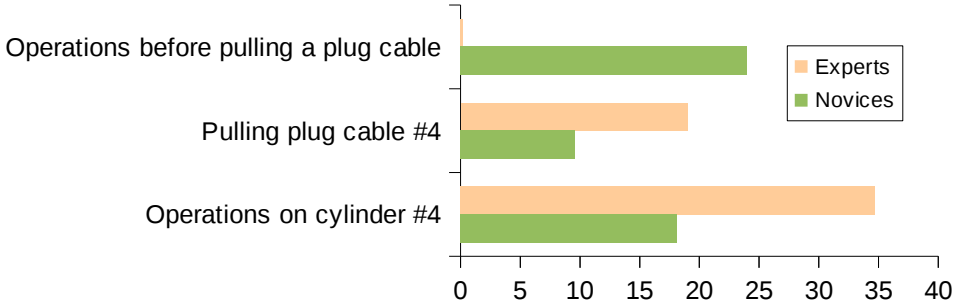


Figure 7: Main results from the experiment

When pulling plug cables, operators try to locate the faulty cylinder by listening to the speed of the engine. If it does not drop when pulling a cable, then the corresponding cylinder is faulty. Then, operators must explain the symptoms and discover why the cylinder does not explode. The electrical causes can be withdrawn by making the plug #4 spark on the engine block, outside the combustion chamber. The operators can then deduce that the whole electrical circuit is working properly. Indeed, since a plug's cable is at the very end of the electric circuit, a spark means that every upstream electric component works. Only when electrical causes are eliminated (the cylinder #4 does not explode despite the spark plug working) do experts start considering causes that are more costly to test (mechanical in this instance, that require the engine to be stripped down).

Because the split-half test of pulling a plug cable is so powerful in eliminating an entire set of causes, experts used this very early in the test plan. Also, the split-half test of trying the spark plug on the engine block (which is common practice) allows one to test an entire chain of electrical components, thereby orienting the diagnostic process towards mechanical faults. This last behaviour is typical of a frequency-driven strategy whereby electrical causes

explain the vibration symptoms most of the time.

1.5.3. Conclusion on the experiment

The results of the experiment show that experts rely very strongly on heuristics (at least the frequency heuristic) when troubleshooting. They behave as if faults were sorted in their mind by order of likelihood. The sequence of tests seems to follow this hypothetical classification. One aspect that this experiment did not test was the role of cost in choosing to run a particular test. Indeed, even when a cause of fault is extremely rare, expert mechanics sometimes decide to test it, provided it can be done quickly and without effort. This is something that I have witnessed myself but never measured under controlled conditions.

Beyond the results described above, it was interesting to notice that very few mechanics (even among experts) found the metallic plate in the the intake tube. In fact, once frequency-driven strategies had exhausted their set of candidate faults, the troubleshooting performance dropped dramatically. In this respect, it was informative to notice that experts, after testing the most likely (electrical) causes, started to think aloud and evoke causes on the basis of a forward chain-reasoning, thereby clearly resorting to an inferential, topographic reasoning. By pushing reasoning a little bit further, one could even suspect the existence of conditions where experts' reasoning performance could drop to that of novice operators. The explanation of such a phenomenon might be the extremely strong reliance on heuristic troubleshooting strategies, overemphasising the lessons of experience to the point of biasing test plans. In other words, in expert operators' mind, symptoms automatically activate a frequency-distributed set of faults. Therefore, when well-known symptoms are detected but actually originate from a rare cause, frequency-based troubleshooting rules can be erroneously applied, thereby triggering standard test plans that can fail to locate the fault. This is precisely the hypothesis that was investigated in the field of electronics (Besnard & Bastien-Toniazzo, 1999; Besnard, 2000) and that will be reported in the next section.

1.6. Failures from troubleshooting experts in electronics

A dimension that I have not addressed yet is troubleshooting errors, especially by expert operators. Because they usually perform well in their job (they are fast and precise), it is on these operators that the efficiency of workshop operations relies to a great extent. What was not said yet is that experts are not immune to failures. Indeed, they have their own failure modes

which sometimes can make their performance drop below that of novice operators. This is what will be described in this section, in the field of electronics.

1.6.1. Expertise and its shortcomings

As an operator gains experience, a given decision rule becomes more and more specialised, to the point where it will become activated in a smaller and smaller set of cases. It will finally be activated only in the situations where it is the right thing to do (Ohlsson, 1996). The result is that operators link together a given behaviour of the system and some failed components via a heuristic rule (Pazzani, 1987). Expert fault-finding then becomes the application of the rule that best explains the symptoms at hand, or that is most often activated in the current configuration of symptoms (Nooteboom & Leemeijer, 1993). Rules are implemented sequentially from the least to the most probable (Bereiter & Miller, 1989). But humans are fallible statisticians (Patrick, 1993) and for that reason, decision on the basis of the frequency of the symptoms may generate irrelevant actions.

One might have thought that expertise would be a guarantee of high performance provided that the problem at hand is contained within the operator's mental repertoire. However, things are not that simple. Indeed, the numerous problems solved by experts progressively make reasoning brittle: the [SYMPTOM-CAUSE] matching tends to trigger almost autonomously as soon as some familiar pattern of symptoms is detected. This can affect performance when a suspected fault erroneously activates a solution plan that is not adequate. This happens when the symptoms of the fault have been incompletely captured but still find a match in the operator's repertoire of causes. In such a case, the troubleshooting process is oriented towards a candidate for the fault that is irrelevant.

Following Reason (1990), I maintain that failures from expert operators originate in the way the symptoms present in the environment are processed. As already said, these symptoms can be extracted in a flawed manner (for instance only the most salient have been noticed) and can therefore trigger an irrelevant action plan. Now, a beautiful aspect of novices' reasoning is that they do not hold any sort of knowledge about the relative frequency of faults, and therefore do not operate troubleshooting on the basis of [SYMPTOM-CAUSE] pairs. With such a difference in reasoning, it might be interesting to see if conditions exist where expertise can impair performance. Could novices then deploy logical, inferential reasoning strategies but still outperform experts? This is the question that was investigated in some of my publications

(Besnard & Bastien-Toniazzo, 1999; Besnard, 2000), and that will be revisited here.

1.6.2. An experimental demonstration

To answer the above question, an experiment was conducted (Besnard & Bastien-Toniazzo, 1999; Besnard, 2000) that required electronics operators to troubleshoot a faulty audio amplifier. This bespoke, fully functional audio amplifier was assembled on test boards and was the experimental device for this study. A professional electronics technician designed and assembled the amplifier. The picture in Figure 8 presents the physical appearance of the device.

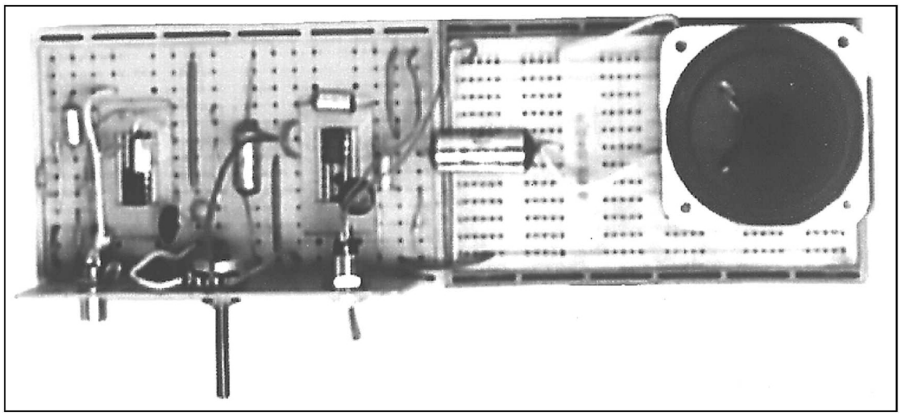


Figure 8: The experimental electronic device

The diagram in Figure 9 represents the location of the various components.

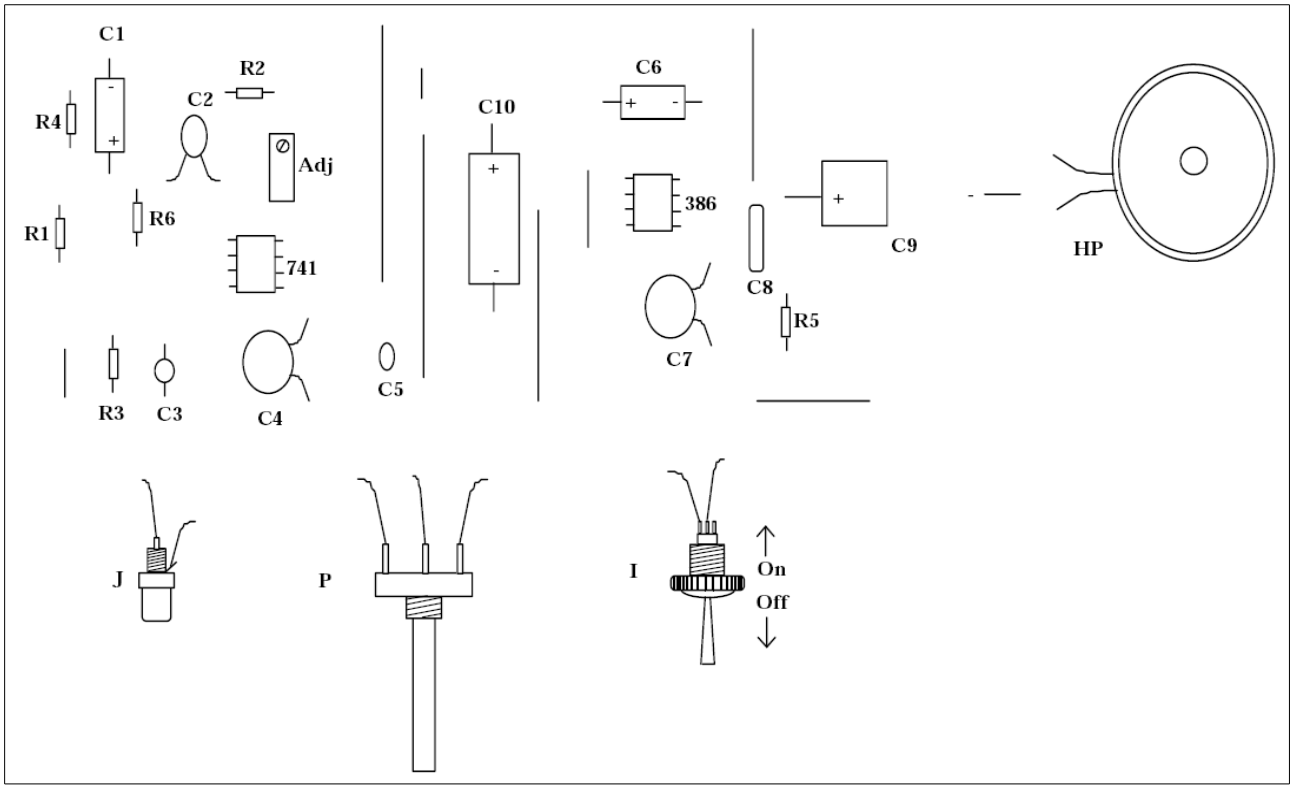


Figure 9: Layout diagram of the experimental electronic device

The functioning of this amplifier was very basic. It took an input audio signal, transformed it in two stages into a more powerful signal and finally sent it to a speaker. The amplification stages one and two were performed by integrated circuits[22] 741 and 386 respectively. In order to study troubleshooting, a fault was implemented into the circuit. This fault was located in the condenser C7 and consisted in short-circuiting it with a very thin wire soldered on its under side. As a result, the signal coming out of 386 (the second-stage integrated circuit) was sent directly to the mass track and no signal whatsoever reached the speaker.

The assumptions lying behind the choice of this particular component as the fault deserve some explanation. Condensers are extremely reliable components. Their mean time between failure is significantly greater than that of integrated circuits[23]. I also demonstrated in previous sections that experts use symptomatic troubleshooting strategies based on relative fault

22 Also known as microchips

23 Here is a side story that helps to illustrate the robustness of ceramic condensers, like C7. I was with the engineer who assembled the experimental device when he tried to destroy the condenser by running the mains' current through it. This happened in his electronics lab, in the Department of Psychology at the University of Provence. The condenser, although designed for low voltages, did not die. However, an entire half-floor of the building was without electricity.

frequencies. Therefore, one can expect that the condenser C7 will not be taken as a likely cause of fault by experts, and that the components that are more likely to fail (integrated circuits) will be tested first. Conversely, novice operators implement a more topographic strategy than experts. Therefore, one can expect that the condenser C7 will be treated as an equally probable cause of fault as integrated circuits, and that C7 might appear sooner in the novices' test plan.

The study was carried out on 10 expert and 10 novice electronics operators recruited from repair workshops of the French Air Force and in technical colleges respectively. They were all constrained to use a fixed, standard set of tools and worked on a properly equipped work bench. The experiments were videotaped and analysed according to the following criteria: time, the component being tested, the position of this test in the test plan, and the total number of tests. Although essentially quantitative, these variables were deemed appropriate to highlight the effects of expertise on troubleshooting strategies.

The results in Figure 10 show a series of significant differences between experts and novices. For the vast majority of them, one can see that:

- experts take more time than novices to reach the fault;
- experts test integrated circuits more often than novices;
- experts perform less tests than novices before testing integrated circuits (the most likely cause of fault);
- experts perform *more* tests than novices before the first test of C7;
- experts perform very few operations after having tested C7 as opposed to novices.

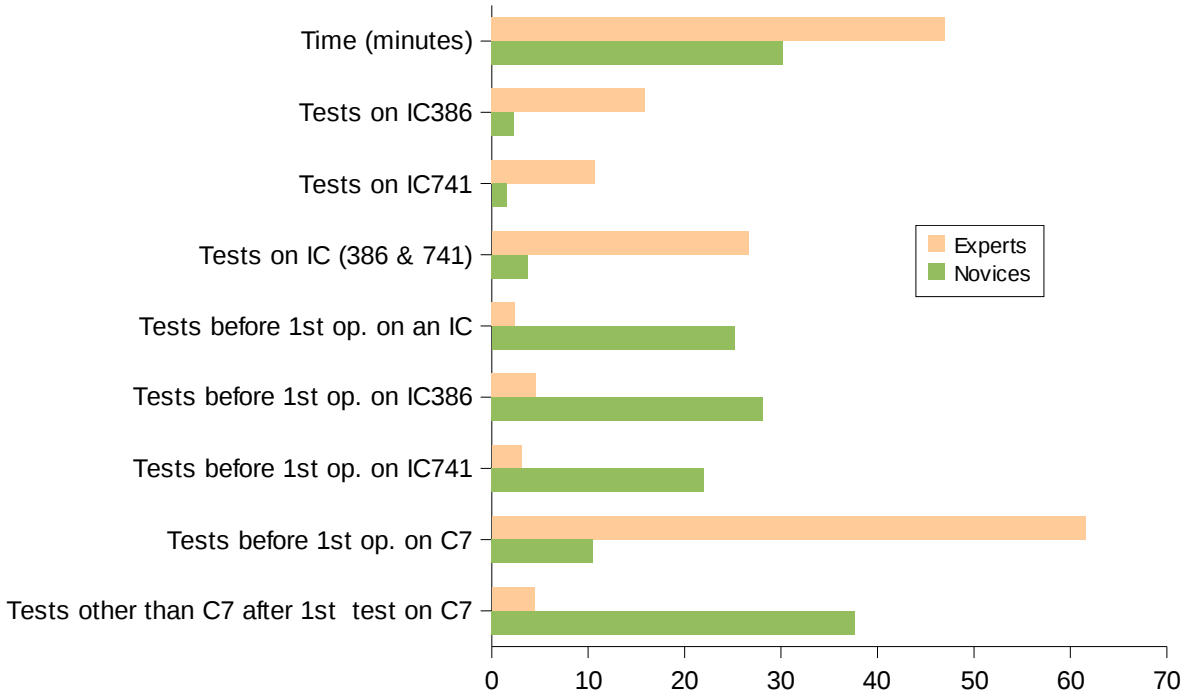


Figure 10: Main results (mean values) from the experiment

Altogether, these results converge towards a performance pattern where experts deploy a frequency-driven strategy that is outperformed by the topographic (if not opportunistic) novices' strategy. Of course, experts have based their strategy on repeated occurrences of the same faults and their associated causes. It follows that they exhibit better performance levels than novices in the vast majority of cases. However, the point here was to test whether a fault with an atypical cause could impair experts' performance. For more details on this experimentation and interpretation of results, the reader can refer to the corresponding publications (Besnard & Bastien-Toniazzo, 1999; Besnard, 2000).

1.6.3. Conclusion on the experiment

The results contributed to the idea that expert knowledge tends to become more and more brittle over time, thereby depriving experts from deploying opportunistic strategies in cases where a fault calls for a symptomatic strategy. This brittleness goes against performance in atypical cases where heuristics orient the troubleshooting process towards a pre-defined set of causes.

Following this experiment, I defined expert failure as a case where an experienced operator fails to deliver a standard performance in conditions that might not affect the performance of an inexperienced operator. The

reason has to do with the cognitive processes at play during the building of expertise itself, and the strategies that are deployed when solving a problem. Experts build their knowledge on the basis of repeated interactions with variants of the same type of problems. Over time, this experience is translated into rules that associate a set of characteristics of a problem (or symptoms) to a corresponding solution. With enough practice virtually all the possible functional links between causes and solutions have been explored, which provides an expert operator with an extremely efficient repertoire of ready-made solutions.

Expertise implies that inferential strategies are no longer used to solve problems. Instead, a [SYMPTOM-CAUSE] matching mechanism takes place that substitutes reasoning with a solution recall. The downside of this associative problem solving strategy is that it provides a high level of performance (problem solving, troubleshooting, decision making, and so on) if and only if the situation or problem at hand has been correctly identified and characterised. And this is precisely where expertise can potentially break down. Indeed, a rare situation that is mistaken for a familiar one will trigger an inappropriate solution. In technical domains, this makes operators activate an erroneous set of checking points and actions. Unfortunately, because experts rely so much on this associative strategy, they tend to go in circles, resuming the whole problem solving exercise at the same phase: extraction of symptoms. And because in static systems, symptoms do not change much from one trial to another, the same *extraction-recall-application* cycle can repeat itself several times before operators decide they are facing an unusual case. Beyond troubleshooting, the evaluation of the case at hand as an unusual one is a process that is captured by Klein's (1997) recognition-primed decision model. Indeed, decision makers can exhibit over reliance on simple, direct match decision strategies.

Novice operators, because they rely on an inferential strategy to solve problems, do not exhibit the same behaviour as experts. Their performance is jeopardized in complex unknown cases, where the number of combining hypotheses to test increases beyond what working memory can handle. However, this weakness is their strength. Indeed, contrary to experts, a rare problem that looks like a familiar one is unlikely to happen, given that experience is limited. Instead, such a case will be treated as most other cases: an unknown one. And this is where the pendulum of performance can swing the other way: novices can outperform experts. In this respect, expertise should not be seen as a guarantee of performance. Indeed, as Hollnagel (1998) put it, the conditions in which the work is done are the main

determinant of human performance. These conditions obviously include physical parameters such as light, fatigue, training etc. However, the cognitive characteristics of the task also influence the output of the reasoning process. As a result, system performance can be seen as function of the collaboration between people and technology, in a given context. This is a point that will be addressed again in this document when addressing human performance in the control and supervision of dynamic, critical systems (in Chapter 2).

1.7. Diagnosis for decision making

So far I have considered diagnosis as an aspect of troubleshooting in the sense that fixing a problem was the main objective of the operator. This does not need to be so. Diagnosis is also about characterising and categorising a situation and supporting the choice of the appropriate set of actions.

1.7.1. Diagnosis: what for and how?

According to Svenson (1990), there are two classes of decision making connected to diagnosis:

- *Fixing*. This class is about a situation where an operator must choose between several alternative solutions. In this case, troubleshooting applies to settings where action is necessary. Typically, the state of the system (e.g. an engine; a patient) has to be altered and the operator must choose some appropriate action (e.g. repair; prescribe).
- *Tuning*. The second class refers to situations where the state of the system is acceptable as is but an alternative is deemed preferable. Troubleshooting can then involved in tuning when some future system change is anticipated and triggers an anticipatory adjustment.

From another point of view, diagnosis and decision making can determine each other (Hoc & Amalberti, 1994):

- The operator can prioritise operational (mental) representations that allow them to access possible action. This is typical of operating of high-tempo systems;
- Operators can also have their actions determined by the need to obtain a change in the system state.

In decision making, especially in supervisory monitoring tasks, the need to change the state of the system is constant. It is even the main activity, and diagnosis is needed since it allows one to characterise the current conditions. Classically, the rest of the decision making process is a matter of evaluating

and choosing what precise action is needed, and executing it. These sequential components of decision making have been identified as rather basic by Hollnagel (1998) but other models exist. Indeed, Rasmussen (1986) in his step-ladder model (Figure 11), without rejecting the basic decision making steps, also conceives a heuristic mode represented as decision shortcuts. Under this mode, the characterisation of a situation (the output of the diagnosis) is matched with corresponding actions by the application of an [IF-THEN] rule. Decision making is thus seen as a process that attempts to find a match between some features and a class of situations. This is included in Klein's (1997) decision making model where at the simplest level, decision making consists of finding a match between the situation at hand and a set of typical criteria for this case (expectancies, relevant cues, plausible goals, and typical actions). This search for a match is also what Kaempf *et al.* (1996) noticed when studying decision making among officers in charge of an anti air defence system aboard surface ships.

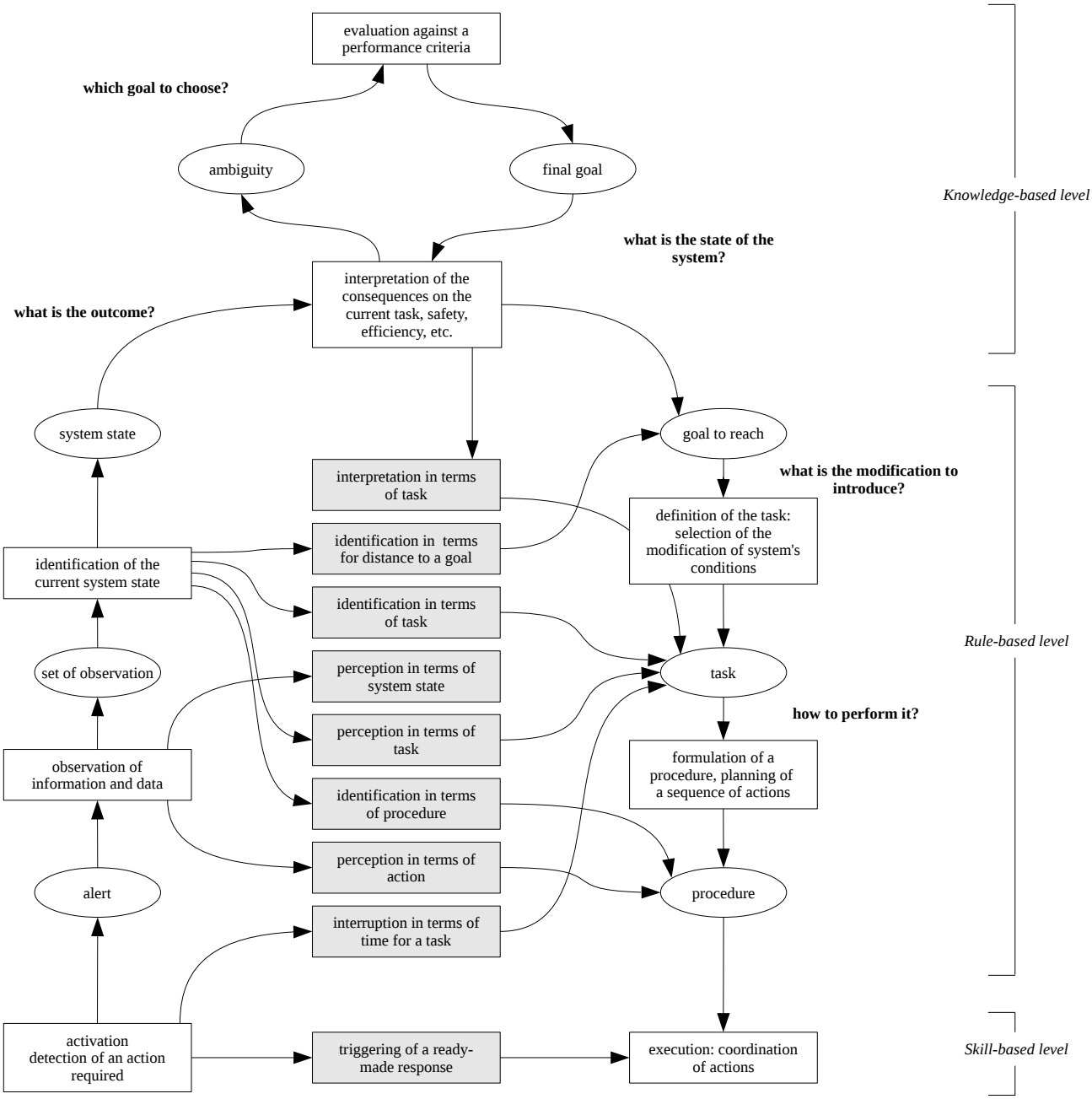


Figure 11: Rasmussen's (1986) step ladder model

1.7.2. Diagnosis and planning

Diagnosis is also linked to action selection. In this case, diagnosis is not a mere categorisation of situational data but a goal-driven search for information where the data extracted from the situation at hand is used to trigger actions. This is done by decomposing overall goals into sub-goals. This decomposition is underpinned by planning, which an essential part of the

activity since most of the time, the completion of a task is the concatenation of sub-tasks. For instance, preparing a cup of tea (final objective) will imply the need to boil water, have a mug ready, pick a tea bag, and so on. Only when all these sub-tasks have been completed can the main objective be achieved.

On the topic of planning, Richard (1990) distinguishes between:

- *Downwards planning*. This mode defines sub-goals from a general one, deals with interactions between and ordering of sub-goals. This implies an existing plan and assumes a significant amount of resources available to dedicate to the control of actions. Downwards planning also means that the quality of the plan will be determined by the extent to which the target activity is understood.
- *Upwards planning*. This consists in implementing a plan at the same time as it is being built. The feedback provided by the control of action is the main mechanism by which this type of planning is performed. This type of planning can be a case of adapting a known plan as an action is being carried out or abstracting a plan from a set of procedures.

Also, planning is involved in different levels of abstraction in decision-making:

- *Strategic*, where the general approach to a problem is established, the main objectives to be reached are identified, the gross time frame is set, etc.;
- *Tactical*, where plans are discussed and implemented into actions.

Planning requires a different type of knowledge than controlling the activity. Indeed, planning requires identifying the objective of the action and the operating modes that will need to be deployed, under which constraints, etc. This assumes that knowledge is conceptualised, i.e. skills have been developed that can be made explicit (verbalised for instance) in a variety of problems (Richard, 1990). This is consistent with Schraagen and Schaafstal (1996) who address planning from a strategic angle: knowing what to decide, what to do, and when. Namely, in a high-tempo, dynamic situation, the actions to be carried out must be chosen according to their adequacy with respect to the current and anticipated states of the situation. For instance, aircraft pilots program portions of their flight before they actually have to fly them. This allows pilots to stay in control of a situation that could otherwise evolve faster than they could handle[24]. This way of controlling systems is consistent with how Hoc (1996) sees the human cognition, in that what best

24 Pilots talk about staying "ahead of the plane".

characterises human operators is their tendency to anticipate (as opposed to react to) events.

Long-term planning in dynamic systems does not rely on cues that are present in the current situation but on a probability of occurrence of data in the future. In other words, the current state allows one to infer potential future states, provided that the current system state has been correctly determined or diagnosed, and that state forecasting relies on the right data. To sum up, in dynamic situations, the adjustment of the plan is pro-active and precedes the system state for which it prescribes actions. This is the case in such situations as controlling a high-furnace (Hoc, 1989; 1991) or piloting a fighter aircraft (Amalberti, 1991b; 1992).

1.8. Diagnosis in supervisory monitoring tasks

Troubleshooting consists of finding the cause of some abnormal behaviour whereas process control seeks to maintain the process within normal boundaries. Despite this difference, diagnosis (the interpretation phase of troubleshooting) is present within process control as a mean to check the system outputs and comparing them against a set of criteria or expectancies. Actions are then chosen on the basis of the result of this diagnosis (Cuny, 1979).

In supervisory monitoring tasks, the system must be controlled and potential malfunctions have to be detected. This is what Cellier *et al.* (1997) call monitoring. The performance of the operators in this task depends on the knowledge they have about normal and abnormal system states, the evolution of those states, and on their capacity to focus on potential sources of malfunction. Searches for explanations can be launched based on particular behaviours that the mental representation of the system cannot explain, while the system is being in operation (Amalberti *et al.*, 1995).

For Boreham and Patrick (1996), identifying deviations from the system's equilibrium is part of the whole supervisory process. Control and diagnosis are tightly linked in natural situations. Typically, the result of a control action can be used as the starting point for a diagnosis sequence. For instance, if an operator conducting a series of checks on a production line notices that a conveyor belt squeaks, then a search for potential causes can be initiated. Then, once the diagnosis has established the cause of the fault and it has been fixed, monitoring acts as a confirmation that things are back to normal. There is of course an automated version of control whereby the state of the system is controlled by sensors, the values from which directly affect the

actions of the controller (Lee, 1994). In this case, the activity of control has a diagnostic value and is then used in the detection of the causes of losses in the system's equilibrium. This is an issue that is beyond the scope of this thesis and that will not be developed further.

On a technical level, some key features of modern supervisory monitoring tasks are:

- *Intensive use of IT technology* to control virtually all systems of supervision and interface;
- *The responsibilities of operators have changed* since their role has progressively shifted from a direct, manual control to one of a supervisory nature (people have moved from doing to thinking);
- *A higher degree of qualification as well as abstract knowledge are required* in order to cope with the unfamiliar or unknown (Cacciabue, 1991).

These features have redefined the role of operators. In a typical supervisory monitoring task, a group of human operators controls a system with direct input and feedback. Today, IT-based control systems have introduced dramatic changes. Indeed, not only did operators lose the direct control they used to have of the system (such as flying a classical aircraft with cables and pulleys to control surfaces), but feedback itself is sometimes sent through the IT system (e.g. on-board computers can shake the control column of modern aircraft in order to simulate a stall) (Sheridan, 1992).

But all is not as clear-cut as it seems. Modern IT-based supervisory monitoring systems often contribute to safety, and have also allowed more flexible tasks to be defined, where operators have more freedom to choose their own strategy (Cacciabue & Kjaer-Hansen, 1993). These important issues will be addressed in the next chapter.

1.9. Contribution to the field and future challenges

So far, I have addressed the general topic of troubleshooting. I have attempted to show how interpreting symptoms plays an important part in solving problems. I have also addressed a number of troubleshooting strategies and seen how they can fail when they are triggered by misinterpreted symptoms. Last, I have addressed the issue of troubleshooting in decision making, and supervisory monitoring tasks. Through these topics, the questions of cognition in static systems and the role played by symptoms interpretation were addressed. As stated at the beginning of this chapter, this is not a top-down mechanism but is very much influenced by knowledge and

experience. These two factors, could be taken as guarantees of a high level of performance. Indeed, one might be tempted to believe that the more one knows, or the more experience one has, the better the performance. While this might be true for some routine situations, it is definitely not so for exceptional problems.

These various topics have been extracted from my research and I would like to come back to some of my publications in order to highlight my contribution to the field of troubleshooting. First, the review of the strategies and failures in troubleshooting that were presented here was done during my PhD (Besnard, 1999). This review was based on both a large number and a wide array of bibliographical sources. I still consider it today as a valuable source of material for the study of troubleshooting for the simple (and maybe naive) reason that I do not remember having seen such a review before, nor since then. The experimental demonstration of expert failure in troubleshooting in electronics was published in two articles (Besnard & Bastien-Toniazzo, 1999; Besnard, 2000). In these two publications, the fallibility of expert reasoning was highlighted, an aspect of human cognition that has not often been studied quantitatively. The reason might have to do with the difficulty of selecting tasks and offering devices that are realistic enough so that the researched phenomenon (expert failure in this instance) can be exhibited by the subjects and measured by the experimenter. The same probably applies to the experiment conducted on expert mechanics operators (Besnard & Cacitti, 2001). Indeed, experimenting with a mobile working engine required a little bit of planning, as well as the same scientific challenge of calibrating the task at the right level and for the right subjects. The quantitative approach to measuring natural expert behaviour (as in Besnard & Bastien-Toniazzo, 1999; Besnard, 2000; Besnard & Cacitti, 2001), does not have many adepts in the scientific world. Certainly, the efforts that were invested in gathering and interpreting the results exposed in this chapter were (to the best of my knowledge) unprecedented in the field of ergonomics of troubleshooting.

At the end of this chapter, I would now like to feed in my own reflection on this thesis and put my ergonomist's hat back on: How can the workplace in general be supportive of both expert and novices reasoning modes? Experts match problems with stored solutions that they adapt, while novices tend to deploy an inference-based strategy. If, for instance, one was to look for a way to enhance performance of both novice and experts in the control of some industrial process, how could one engineer a system that allows each level of expertise to act to the maximum of their capacity? I will attempt to address these issues by proposing and analysing potential solutions for real-life

systems. For now, the next step is to have a look at another important area where cognition and performance are tightly related to each other: human-machine interaction in dynamic, critical systems.

Chapter 2. Human Cognitive Performance In Critical, Dynamic Systems

So far in this thesis, I have discussed cognition and human performance in static systems, with little influence coming from time. However, it would be a reduction to not acknowledge the role of cognition in the management of dynamic systems. In such systems, contrary to what was discussed in the previous chapter, the pace of evolution of system states is a fundamental aspect of the complexity in the control and supervision activities (Decortis, 1993), which in turn impacts heavily on performance. Dynamic systems, on top of often being fast-paced, depend on, or are influenced by, or interact with a potentially high number of elements of their surrounding world. With these systems, problems are largely due to operators' workload due to complex and fast-changing, continuous processes. When dynamic systems are also safety-critical (e.g. aircraft, power plants, etc.), the potential lack of time for recovery can sometimes cause catastrophic consequences as a result of human failures. However, human limitations are only one side of the coin. Indeed, there are also sources of performance variation that are due to technology and that trigger such phenomena as cognitive conflicts, mode confusion, misleading interfaces, etc.

These are amongst the topics I will address in such diverse contexts as aviation, maritime navigation, nuclear energy production and steel production. These domains share such features as dynamics, complexity, and some level of automation. They are also the ones I have contributed to and as such, will give me an opportunity to highlight a research heading of mine. It should also be pointed out that dynamic systems need not be critical. Although the reflections presented in this chapter are probably relevant for both non critical and critical dynamic systems, my association of the two words in this chapter heading is due to the focus of my own research activities and because of my interest in systems that share these two

properties.

2.1. Dynamic systems

Dynamic systems depend heavily on time, and this property impacts on how incidents are handled. Typically, past problems must be fixed by future decisions (Pascoe & Pidgeon, 1995). Contrary to static systems, there is no such concept of reversing a system state by cancelling an action. For instance, if I enter a wrong number in my pocket calculator, I can cancel it and enter another one without affecting the outcome of the task. The result of the operation will be unaffected. In dynamic systems however, time imposes that events and their related actions flow, and that the sequence in which things happen is of importance. For instance, if an aircraft pilot introduces an erroneous descent rate into the flight management system[25], then correcting this mistake will imply entering another value. In the meantime, the aircraft will have changed position. This might seem to be a minute detail but the amount of time between the initial mistake and its correction will determine by how much the aircraft position will have changed. This precise issue will be addressed in section 2.3.2.

2.1.1. Properties

On the basis of their potential evolution over time, one can isolate two main categories of control situations:

- *Static systems*. The functioning of these systems is directly related to the actions of the operator, with little importance given to when actions are made. The only way to trigger a behaviour from such a system is to issue a command to it. For example, a pocket calculator will wait for an input indefinitely. As long as the input is not made, the state of the system will remain the same as long as functional resources are available (batteries, in this instance). This category of systems is the one that the previous chapter focused on.
- *Dynamic systems*. Here, the system comprises a process that has some degree of autonomy, is dependent on time, and whose state is only partially linked to the actions of the operator. Such systems have the property of having phases where the state can evolve autonomously without input from the operator. It is the case of a nuclear reactor (Joblet, 1997), a high-furnace (see Hoc, 1991), aircraft piloting, fire

25 The flight management system is an on-board computer-based system that calculates and controls navigation parameters such as (among others) capturing beacons, maintaining trajectory, etc.

fighting (see Brehmer & Svenmark, 1995), etc.

Of course, as is so often the case when one tries to make categories out of reality, a continuous phenomenon then finds itself broken down into a series of discrete steps. A more precise way of describing dynamics in systems would then be to think of a continuum where the importance of time varies. Another dimension of interest is the delay of feedback. Indeed, for any dynamic system, this delay determines how much time there is between any two actions, opportunities for recovery or adjustment, workload, etc., all being dimensions that are directly linked to performance. Performance also depends on the extent to which the dynamic task is composed of relatively independent sub-tasks and systems. For instance, piloting a long-haul flight can be seen as a series of successive phases whose control loops are of different granularity and length, and involve different types of systems.

This multiplicity is captured in Figure 12. Whereas flying an aircraft might be an activity of type 3, it is composed of sub-tasks and sub-systems that can be of lower complexity and that show different properties in terms of their response to interruptions. The systems that pose the greatest problems to human performance are the highly dynamic ones, i.e. those where the control loop is short. That said, there are many such systems with a lot of different properties, which are worth investigating in order to understand their cognitive implications.

Type 1: The general activity is controlled by a single control loop that is interruptible at will

Type 2: The general activity is controlled by a control loop that can be interrupted at particular time points, within which a sub-task takes place that can be interrupted at will

Type 3: The general activity is controlled by a control loop that cannot be interrupted, within which several sub-tasks take place that respond differently to interruptions

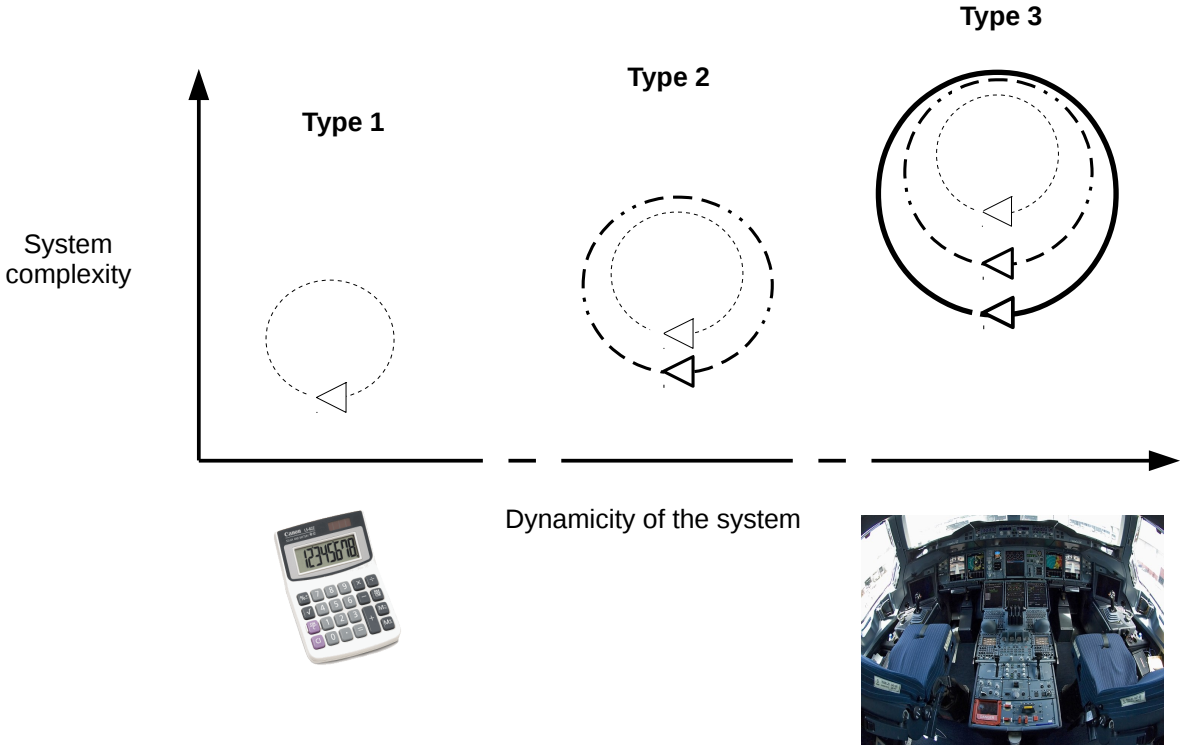


Figure 12: The superimposition of control loops in dynamic systems

Dynamic systems or situations imply interdependent decisions (any decision will impact on what the operator will have to do next), an environment that changes over time and decisions that change the state of potential problems (Brehmer, 1987; 1996; Gibson *et al.*, 1997). Decision making in such systems is dependent upon time cycles (the system is split into cycles of variable length), the number of variables under control and their interactions (Bainbridge, 1993), and the availability of feed-back.

In dynamic systems, operators must decide what to do and how, but also when they have to carry out a particular action, or delay action until some data is available (Kersholt, 1995). In other words, the selection of the correct action for the current situation is a central feature of decision making in dynamic systems (Rasmussen, 1993). As one might expect, this choice is heavily dependent on the mental representation of the system state that

operators maintain, and on their capacity to correctly anticipate future events. In air traffic control for instance, Boudes and Cellier (1998) propose the concept of *horizon of anticipation*[26]. This term refers to the set of elements taken into account by an operator to anticipate the evolution of a process within a given time frame. In the case of air traffic controllers for instance, this field can include work colleagues. Indeed, as active elements of the system, it is important to be able to foresee the actions that other operators are likely to carry. This issue touches on the importance of long-term collaboration in critical systems control: the more people work with each other, the more they are able to know in advance what their colleagues will do under given work conditions. This supports what Endsley (1996) called situation awareness, an anticipation and pro-active adjustment to future events, an essential feature of the piloting of automated, dynamic systems.

2.1.2. Decision making in dynamic systems

In dynamic situations, the basic decision-making processes and limitations are the same as those applying to static situations. However, the context changes and impacts on what constraints apply to decision-making. Indeed, the occurrence of an incident, or just maintaining the system within a safe operating envelope, forces one to make fast decisions with little room for recovery. For Decortis (1993), this constrains operators to make decisions on similarity or frequency cues. The main concern in doing so is stabilising the system state. Typically, experts involved in an emergency situation (aboard oil platforms, for instance) base their decision making strategies on the identification of situational cues. For these operators, the most important aspect of decision is to cope with the most urgent actions instead of seeking for a perfect response (Flin *et al.*, 1996). Indeed, in situations where lives are at stake and the pace of the evolution of the situation cannot be coped with, operators must react with very short delays. Resource management then becomes the task of dealing with internal (the cognitive capacity of the decision maker) and external (human and material sources available) resources. In real-life situations, coordination and communication play a key role, in order to optimise decisions, transmit them, and finally obtain feedback (Dowell, 1995; Flin *et al.*, 1996).

As already said, time is an important dimension in dynamic decision making. Different operators, depending on their task, operate at different time scales (Brehmer & Svenmark, 1995), which define the pace of the actions carried

26 The original term is *champ d'anticipation*.

out. This idea ties back to what has been said before in this chapter about phases within activities that spread themselves over long periods of time (such as long-haul flights) and enclose sub-activities. For Brehmer and Svenmark (*op. cit.*) who studied fire fighting, there is a local fire time scale, and another, larger one that accounts for the general fire crisis over its entire geographical area and time span. In crises such as fire fighting, the operator in charge of coordination must maintain a global representation of the crisis (the "big picture") when dealing with local fires. This case of multi-granularity planning is also found in process control, for instance when a local malfunction is being dealt with. Solving local contingencies with no regard to the general process is not the most efficient strategy.

2.2. Human interaction with critical, automated systems

So far in this chapter, some inputs from the scientific literature on the topic of dynamic systems and decision-making have been laid out. I am now going to look at how these translate into real-world situations, namely in the transportation domain. This positioning will set a strong focus on automation in critical systems. And because "sharp end" operators sometimes fail when interacting with imperfect interfaces[27], human-machine interaction (HMI) with automated systems can result in accidents. Therefore, the issue of human failure when interacting with dynamic, critical, automated systems will be addressed.

2.2.1. Dynamics, criticality and automation

For accidents in transportation, there are three system properties that can be regarded as important:

- *Dynamics,* as explained earlier, refers in part to the time properties of a system and its relation to the surrounding world. For instance, if a driver going across Salt Lake (Utah) jumps out of his or her car with the cruise control engaged, the vehicle will continue to progress for some time until it runs out of fuel, fails, or meets an obstacle, whichever comes first. In the meantime, the engine will run, the wheels will turn and the radio will play, just as normal.
- *Criticality* refers to the adverse consequences and potential losses (human, technical, financial, …) of a failure. For instance, if my keyboard fails now, as I write this very line, I can fairly easily replace it and resume work with no consequences on what I have written so far.

27 Of course, imperfect interfaces are themselves the consequences of errors made at earlier stages of the system lifecycle, e.g. requirements analysis.

But one could imagine instead that I am a space engineer and my keyboard is connected to a computer that controls a space drone that is about to land somewhere at the other end of the galaxy. In this case, the failure of the keyboard might cause heavy losses to the whole mission.

- *Automation* refers to the design of a system that makes it able to control tasks or devices in accordance with the input from pre-programmed instructions. Examples of such systems are the autopilots found aboard modern ships and aircraft. They can follow a given heading and can compensate for deviations from the intended route. These compensations can happen without human input since modern autopilots can receive the position of the ship via a GPS signal, compute the difference between the actual position and the intended one, and calculate a new heading if need be.

2.2.2. Staying ahead of the automation

Automation has been the response of the aviation industry to the increasing complexity of modern aircraft (which happen to be dynamic and critical systems). Indeed, we (as customers) want to fly faster and safer aircraft in more and more cluttered skies. Allocating such missions to humans alone creates insurmountable constraints (in terms of e.g. workload, fatigue, etc.). As a result, automation started to appear in aircraft cockpits in the late 1980s in the form of computer-driven aircraft. Nowadays, flying a modern aircraft (i.e. less than 20 years old) almost always implies interacting with a computer at some stage of the flight. Computers are pervasive in today's aircraft cockpits. They manage a large number of functions, ranging from fuel consumption, collision avoidance, capturing navigation beacons[28], etc.. To a very large extent, these automated functions show a very high degree of intrinsic reliability, that is their actual behaviour is virtually always what the designer programmed. But this certainly does not mean interacting with these systems is easy. Indeed, automation shows a number of properties, some of them impacting negatively on human performance:

- *a re-allocation of tasks* (operators' mission becomes one of controlling and supervising);
- *a large degree of autonomy* in function handling (e.g. the navigation task, in the case of aircraft and ships);
- *a tight integration* of, and complexity in the interactions between, the automated functions;

28 A navigation beacon is a radio marker on land that aircraft systems are programmed to locate, and use as a way point.

- *some potential for opacity* of the behaviour of the automated system.

In such conditions, it is vitally important that operators stay "ahead of the the automation". This means that operators know what the system is doing, why and when, so that they can anticipate. The expression captures the essence of what it is to control and supervise dynamic, automated systems: one needs to understand what is going to happen over a variety of time granularities (e.g. from some seconds to several minutes). The reason is simple: some phases of the activity (e.g. landing an aircraft) generate far too many events for the (limited) capacity of human cognition. It follows that when the anticipatory control is lost, operators find themselves in a high-paced flow of information within which finding what is significant for the current task becomes impossible. This is why a significant part of the time of operators in control of critical, dynamic systems is spent on preparing such aids as flight plans and possible scenarios (Amalberti, 1992).

2.2.3. Automated systems and automation surprises

Traditionally pilots were taught to aviate, navigate and communicate. The advent of the glass cockpit[29] has changed the pilot's role, in such a way that they are now taught to aviate, navigate, communicate and manage systems (as reminded by Besnard & Baxter, 2006). As the number of automated functions increases in the cockpit, more and more of the pilot’s time and effort is spent managing these individual systems. The situation is likely to get worse as more automation is introduced into the cockpit, unless some new way is found to reduce the cognitive resources required to interact with and manage the automation. One philosophy that can diminish the occurrence of automation surprises[30] is designing the system in such a way that its operation is transparent to the operators. In other words, if operators can easily understand the design principles underlying the system's behaviour, then they can then predict more accurately the future states of that system. This predictability is one of the core features of the reliability of HMI, especially in emergency situations (FAA Human Factors Team, 1996).

Systems designers assume some minimum skills of the operators as a prerequisite. However, there is also a tendency for designers to assume that the operators fully understand the functioning principles of flight deck systems. This sometimes causes systems to exhibit behaviours that operators cannot always explain, even in the absence of any obvious failure on their

29 The glass cockpit refers to the 1980s' aircraft design philosophy that introduced the cathode ray tube technology (CRT monitor-based instruments) in the cockpit.
30 A definition will be given in the next section

part. For instance, on the Bluecoat Forum[31], a pilot reported an unexpected mode reversion. The aircraft was given clearance for an altitude change from 20,000 to 18,000 ft (flight level 200 to 180). However, shortly after the crew selected the Vertical Speed (V/S) descent mode and a rate of 1000 feet per minute, the aircraft automatically switched twice to Level Change (LEV CHG) mode without any direct intervention from the crew:

> "*We were in level flight at FL200, 280kts indicated, with modes MCP SPD/ALT HLD/HDG SEL. We then received clearance to FL180, so I dialled 18000 into the MCP, and wound the V/S wheel to select 1000fpm descent. After a moment or two, the aircraft went into LVL CHG. I reselected V/S by means of the button on the MCP, and again selected 1000fpm down. Again, after a moment, the aircraft reverted to LVL CHG. After these two events, the aircraft behaved "normally" for the rest of the day. The engineers carried out a BITE check of the MCP after flight, and found no faults.*"

Here, the aircraft autonomously (and unexpectedly) changed mode against the crew's actions and did not provide explicit feedback on the conditions that supported this change. This incident indicates how even experienced pilots can encounter difficulties in interpreting behaviours generated by complex, dynamic automated systems. The triggered actions cannot always be forecast or explained by the operators. As flagged by Johnson and Holloway (2007) this is partly because the complex combination of conditions underlying these (normal or fault-masking) behaviours is managed by the automation and, for the most part, hidden from the operators.

As repeatedly emphasised here, anticipation is vital for the control of dynamic, critical systems. The entire piloting activity relies on it. However, there are cases where anticipation can be lost, due for instance to HMI-related mishaps. It is important to understand the mechanisms that can lead to loss of anticipation since in dynamic critical systems, it often corresponds to loss of control. Given that the consequences of the latter usually involve large material losses and fatalities, I am going to review and analyse three HMI flaws: cognitive conflicts, mode confusion, and failures due to ill-defined interfaces. In doing so, I will attempt to highlight the tight relationship that exists between technical artefacts, their behaviour and cognitive

31 The Bluecoat Forum is an international e-mailing list on the subject of automated flight systems and the integration of all avionics equipment in the modern cockpit. Visit http://www.bluecoat.org.

mechanisms. Several industrial cases will be used to make this demonstration. Figure 13 depicts the sequence of the three topics that will be reviewed along with associated industrial cases.

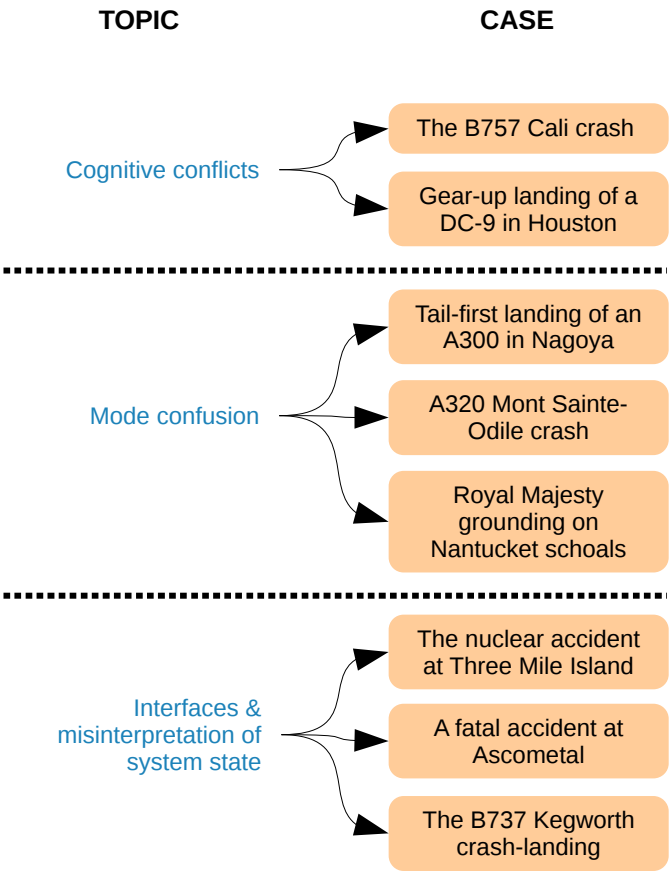


Figure 13: Sequence of topics and associated industrial cases

2.3. Cognitive conflicts

2.3.1. Definition

A cognitive conflict results from an incompatibility between an operator's mental model and the process under control. The conflict often materialises as a surprise or misunderstanding on the part of the operator[32]. Automation surprises (Billings, 1997) for instance, are cognitive conflicts that arise when

32 Formally speaking, the conflict is not in the head of the operator but only exists as a concept in the head of whoever attempts to formalise it. The operator only experiences a misunderstanding.

the automation (e.g. the autopilot) behaves unexpectedly. Conflicts have also been characterised by Dehais *et al*. (2003) in terms of the impossibility of a number of cooperating agents to reach a goal, for reasons including lack of resources or knowledge, contradictory objectives, or lack of agreement.

For the scope of this thesis, the categorisation of cognitive conflicts can follow two dimensions (as proposed by Besnard & Baxter, 2005; Besnard & Baxter, 2006; Baxter, Besnard & Riley, 2007; as seen in Figure 14):

- *Nature*. An unexpected event occurred or an unexpected non-event occurred (i.e. nothing happened when the operators were expecting something to happen);
- *Status*. The conflict is detected (the operator is alerted by a system state) or hidden (the operator is not aware of the system state).

A point of importance in this view is that it states explicitly that conflicts can remain hidden to the operator, thereby opening the possibility of an unnoticed loss of control.

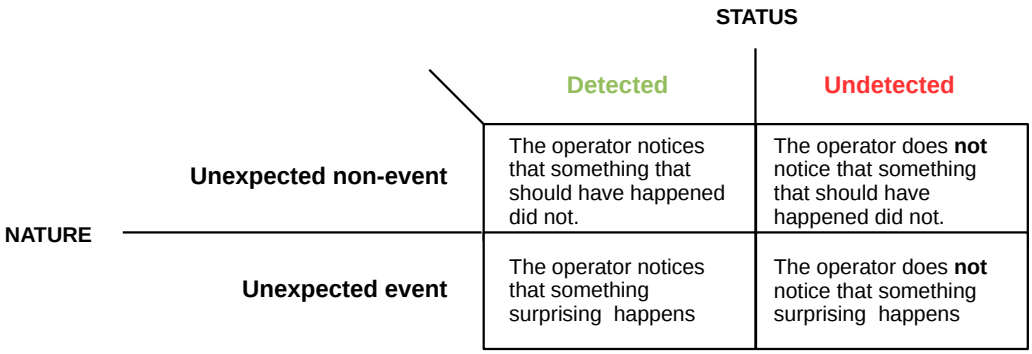


Figure 14: Nature and status of cognitive conflicts

Cognitive conflicts are a generic mechanism that is potentially involved in any control and supervision activity. They reveal an incompatibility between the mental representation of the system that the operator maintains and the actual system state. The occurrence of cognitive conflicts can be facilitated by over-computerised environments if the automation's decision rules opaquely trigger unexpected or misunderstood system behaviours. Of course, computer-based critical systems do not necessarily trigger failures but given the increased complexity of the situations that the software controls, they increase the likelihood of cognitive conflicts.

The following two cases of the Cali crash and the DC-9 gear-up landing will give examples of conflicts and will also show that this phenomenon is not specific to computerised environments.

2.3.2. Cognitive conflicts in unexpected events: the B757 Cali crash

In December 1995, a Boeing 757 flying at night from Miami (Florida) crashed into a 12,000ft mountain near Cali, Colombia, killing nearly all of the 163 people on board (Aeronautica Civil of the Republic of Colombia, 1996). This controlled flight into terrain (CFIT[33]) accident was attributed to the crew losing position awareness after they had decided to reprogram the FMC[34] to implement a switch to the direct approach to Cali that had been suggested by air traffic control (ATC).

The crew was performing a southbound approach, preparing to fly south-east of the airport and then turn back for a northbound landing (as described in Besnard & Baxter, 2006). Because wind conditions were calm and the aircraft was flying from the north, ATC suggested that the aircraft could instead land directly on the southbound runway (see Figure 15). The approach for this landing starts 63 nautical miles[35] from Cali at a beacon called TULUA, followed by another beacon called ROZO (subsequently re-named PALMA). Because the crew knew they had missed TULUA when the direct approach was suggested, they attempted to proceed directly to ROZO. They therefore reprogrammed the FMC and intended to enter ROZO as the next waypoint to capture the extended runway centre line. However, when the crew entered the first two letters of the beacon name (“RO”) in the FMC, ROMEO was the first available beacon in the list, which the crew accepted. Unfortunately, ROMEO is located 132 miles east-north-east of Cali. It took the crew over a minute to notice that the aircraft was veering off on an unexpected heading. Turning back to ROZO put the aircraft on a fatal course, and it crashed into a mountain near Buga, 10 miles east of the track it was supposed to be following on its descent into Cali.

33 The FAA (2003) gives the following definition: "CFIT occurs when an airworthy aircraft is flown, under the control of a qualified pilot, into terrain (water or obstacles) with inadequate awareness on the part of the pilot of the impending collision."

34 Flight Management Computer, a vital automated flight control system in modern aircraft.

35 One nautical mile = 1852 metres

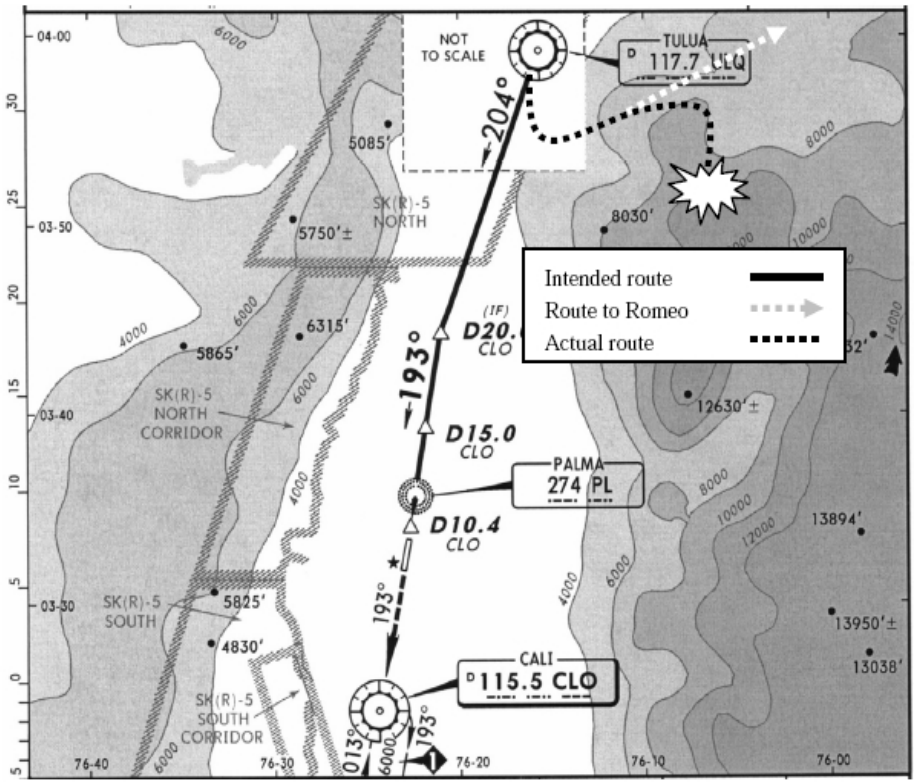


Figure 15: Partial, amended chart of the approach to runway 19 (southbound) at Cali. © Reproduced with permission of Jeppesen Sanderson, Inc.

The subsequent accident inquiry commission noted several failures in the crew's performance, most notably:

- in the acceptance of ATC guidance without having the required charts to hand;
- in continuing the initial descent while flying a different flight plan;
- in persisting in proceeding with the (new) southbound approach despite evidence of lack of time.

After erroneously entering the coordinates of the ROMEO beacon into the FMC, there was a delay before the crew noticed the aircraft's unexpected behaviour. This created the need to re-evaluate and correct the aircraft position and trajectory. The time it took the crew to perform these actions, combined with the erroneous following of the initial descent plan, put the aircraft on a collision course with a mountain.

This case highlights the criticality of delays between an action and the detection of its inappropriate outcome. The crew were in a very difficult

situation in that they were trying to reach a beacon without knowing (as recorded on the cockpit voice recorder) what their precise position was. The Cali accident was exceptional in that beacon names are supposed to be unique in the first two characters for any particular airspace. However, an aspect of the selection mistake is related to the frequency gambling heuristic (Reason, 1990). People operating in situations perceived as familiar tend to select actions based on previous successes in similar contexts. Because the workload was extremely high when the flight path was being reprogrammed, and because of the exceptional problem with the beacons database, the crew did not immediately detect their mistake. The confusion between the ROMEO and ROZO beacons postponed the detection of the cognitive conflict, thereby delaying recovery and worsening the consequences. This simple analysis illustrates that the longer the delay between an action and (the detection of) its outcome, the more difficult it is to recover if that action is subsequently judged as being erroneous.

2.3.3. Cognitive conflicts in expected non-events: Gear-up landing of a DC-9 in Houston

In February 1996, a McDonnell Douglas DC-9 landed with the landing gear up at Houston (Texas) airport (NTSB, 1997b). The timeline of events immediately prior to the accident was as follows.

- 15 minutes before landing, the first officer (pilot flying) asked for the in-range check-list. The captain forgot the hydraulics item and this omission was not detected by the crew. As a result, the pumps that drive the extension of slats[36], flaps[37] and landing gear remained idle.

- 3 and a half minutes before landing, the approach check-list was completed and the aircraft cleared for landing.

- 1 and a half minute before landing, as the aircraft was being configured for landing and the airport was in sight, the crew noticed that the flaps had not extended. Because of this configuration, the aircraft had to maintain an excessively high speed.

- 45 seconds before landing, as the copilot asked for more flaps, the landing gear alarm sounded because the undercarriage was still up.

- 30 seconds before landing, the captain rejected the option of a Go-around[38] since he knew that the aircraft had 3500 metres of runway to decelerate and was confident that the landing gear was down.

36 A flap extending from the leading edge of the wing.
37 A control surface extending from the trailing edge of the wing.
38 A flight mode that automates an abort of landing and seeks to gain altitude in order to start a new approach.

- 20 seconds before touch-down, the Ground Proximity Warning System (GPWS) generated three "*Whoop whoop pull up*" audible alarm messages because the landing gear was still up. Also, at this point, the crew had not run through (all of) the items of the landing check-list.

- At 09:01, the aircraft landed on its belly at the speed of 200 knots. Twelve passengers were injured and the aircraft was written off.

Before the landing gear can be deployed, the aircraft's hydraulics system needs to be pressurised. Because this item had been omitted when performing the in-range check-list, the hydraulics pumps (see Figure 16) had remained in a low pressure configuration, thereby preventing the landing gear and flaps from being deployed.

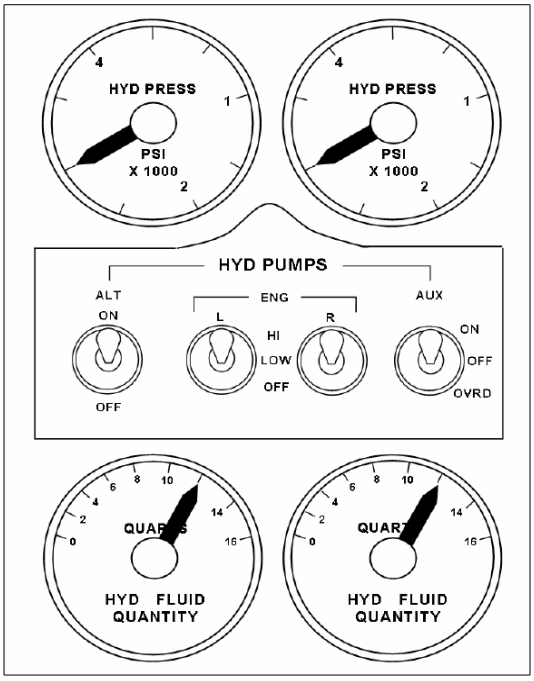


Figure 16: Simplified representation of the DC-9 hydraulic switch panel (with pumps in high pressure position). Adapted from NTSB accident report AAR-97/01

The inquiry commission noticed several deficiencies in the crew's

performance, most notably:

- in failing to configure the hydraulics system;
- in failing to determine why the flaps did not deploy;
- in failing to perform the landing check-list and confirm the landing gear configuration;
- in failing to perform the required Go-around.

The crew faced several cognitive conflicts (Besnard & Baxter, 2006). Here, the focus is set on two of them. The first conflict was caused by an unexpected non-event that was not detected. Namely, the crew thought the landing gear was down, although it had not deployed as expected. This was acknowledged by the crew when the landing gear horn sounded: 55 seconds before landing, the Cockpit Voice Recorder (CVR) tape showed the captain's reaction: "*Well, we know that, you want the gear*". The CVR also shows some further misunderstanding when one second later, one of the crew members announces: "*Gear down*". The second conflict was a detected unexpected non-event. Ninety seconds before landing, the crew noted that the flaps had not extended. The flaps indicator was on 0º whereas the flaps lever was on 40º. Again, the conflict lies in the discrepancy between the crew's expectation (that the flaps should be at 40º) and the system's behaviour (the flaps had not been set).

What is particularly interesting in this case is the over-reliance on weak cues in the system's behaviour: the National Transportation Safety Board report (NTSB, 1997b, p. 45) explicitly noted: "*Neither pilot was alerted to the status of the gear by the absence of the normal cues (increase in noise and lights)*". Despite this, the captain decided to land the plane anyway, thereby rejecting his earlier interpretation of the landing gear horn warnings.

2.3.4. Conclusion

Cognitive conflicts are a condition that can potentially occur in any control and supervision activity. They refer to an incompatibility between the mental representation of the system that the operator maintains and the actual system state. The occurrence of cognitive conflicts can be facilitated by over-computerised environments where opaque decision rules from the automation can trigger system behaviours that are difficult to understand. Of course, computer-based critical systems do not necessarily trigger failures but given the increased complexity of the situations the software controls, they increase the likelihood of such cognitive mishaps.

The two cases of Cali and Houston highlight possible consequences when the

system's behaviour is not fully understood by the operators. In the Cali B757 case, the high reprogramming workload delayed the detection, and subsequent recovery from, the unexpected departure from the intended track. In the Houston DC-9 case, the omission of an item from a check list caused the crew to misinterpret the aircraft's behaviour and alarms, and to crash-land it even though there was no technical failure. Typically, the detection of a conflict triggers some diagnostic activity as operators attempt to reconcile their expectations with the system's behaviour. However, the time pressure faced by crews during busy periods (in this case, the approach phase) can disrupt recovery. Moreover, fixation errors (DeKeyser & Woods, 1990), like those in the case of the DC-9, can sometimes impair situation assessment, rejection of erroneous plans and compliance with emergency procedures (e.g. executing a Go-around manoeuvre).

Finally, only two cases of conflicts (out of the four that Figure 14 includes) were presented: an unexpected event (the Cali case) and an unexpected non-event (the Houston case). Both were detected although their causes were not identified by the crews. In covering only these two cases, the objective was to give a quick description of cognitive conflicts in operating dynamic, critical systems. As one will see with the other phenomena investigated in the rest of this chapter (mode confusion and misinterpreted system state), the explanation of degraded HMI can also be looked at from other angles.

2.4. Mode confusion

In the previous section, two cases of human-system interaction failure were analysed. A contributing factor in both cases was a behaviour from the aircraft that the crew did not expect. In making this point, I discussed the cognitive consequences of the conflict from a rather technology-independent point of view (classical and glass-cockpit aircraft). In this section, a phenomenon will be discussed which is almost exclusive to information technology in control systems. However, despite its apparent specificity, mode confusion still shows a high potential for mishap.

2.4.1. Definition

Mode confusion is a type of automation surprise that concerns mode[39]-driven automated systems (e.g. modern aircraft). The confusion happens when the automated system behaves differently from expectations, due to a mode change (Rushby *et al.*, 1999). The causes for the change of mode can be due

39 A mode can be defined as a configuration of a system achieved via a series of pre-programmed instructions.

to operators selecting the wrong mode, or to the automation itself (e.g. when the system has entered a state where a mode change happens automatically). Mode confusion is a potential source of mishaps since the operator is facing a behaviour that departs from what they were anticipating. They then have to readjust their understanding of the system state in order to regain control. Mode changes can have dramatic consequences, especially when there is little time for recovery, or when the mode change itself has not been detected.

In the following sections, three accidents related to mode confusion in transportation systems will be depicted: the tail-first landing of an A300 at Nagoya, the A320 Mont Sainte-Odile crash, and the Royal Majesty grounding on Natucket schoals. When analysing these events, the focus will be set on the human interaction with the automation and on the cognitive processes involved.

2.4.2. Tail-first landing of an A300 at Nagoya

On April 26, 1994, an Airbus A300-600 left Taipei (Taiwan). Two hours and 22 minutes later, the aircraft crash-landed tail-first at Nagoya (Japan), killing 264 of the 271 people on-board (Ministry of Transport, 1996). A complex interaction between the Go-around mode, autonomous inputs from the trimmable horizontal stabilizer (see Figure 17), and thrust variations contributed to the crash.

During the final stage of the approach, 100 seconds from impact and at an altitude of 1070 ft, the Go-around mode had been erroneously engaged by the First Officer (FO) who was flying the aircraft (as analysed by Baxter, Besnard & Riley, 2007). In this mode, the aircraft applies maximum thrust and climb rate to regain altitude. Five seconds after engaging the Go-around mode, the Captain noticed the FO’s mistake and called out for the mode to be disengaged; this was not done. The Go-around mode caused the aircraft to climb above the intended glide path. In reaction to the climb, the FO tried to bring the aircraft’s nose down and decrease the thrust. The aircraft resisted the nose-down command, although it did level off temporarily at 1040 ft, and the FO managed to throttle back the engines.

At 87 s before impact, the autopilot was engaged. This caused the trimmable horizontal stabiliser to take control of the aircraft’s attitude. At 68 s from impact, the aircraft started to pitch up again; 4s later, the FO disengaged the autopilot. At 48 s from impact, the pitch was still increasing, giving the aircraft greater lift.

Figure 17: Location of the horizontal stabiliser on a A319 (source: Wikipedia)

Given the low airspeed, the Alpha-floor protection[40] triggered an automatic recovery from the near-stall conditions and the aircraft resumed climbing at 570 ft. The Captain took over the controls but could not lower the aircraft's nose to halt the climb. At 34 and 24 s from impact, the Captain verbally expressed his puzzlement and worries about the aircraft's behaviour. The increasing nose-up attitude could not be controlled and the thrust being set back and forth several times only worsened the situation. At 19 s before impact, the A300 went into a stall at a 52° nose-up attitude. The crew attempted several unsuccessful corrective actions on the ailerons[41] and rudder, but the aircraft crashed tail-first at Nagoya.

A direct indication of the misunderstanding of the cause of the aircraft's behaviour is the FO's attempts to counter the effects of being in Go-around mode (triggering maximum thrust and climb rate) rather than just disengaging it (which the Captain had suggested). This was then followed some time later by the captain's verbal expressions of puzzlement about the aircraft's behaviour. The situation was exacerbated by the flight deck technology. Indeed, under 1500 ft, the autopilot could not be disengaged by applying force to the control column because this particular aircraft had not

40 A system that prevents stalls by automatically increasing thrust when the angle of attack of the aircraft reaches a certain value.
41 A flap extending from the trailing edge, on the outermost part of the wing.

been upgraded to support this particular safety function. Had it been enabled, the autopilot would have disengaged when the crew applied pressure on the yoke and this would have brought the aircraft's nose down as desired.

This example highlights some of the problems that can arise when the automation behaves in a way that is not understood. The engaged Go-around mode might have been a mere slip in this case. However, the sheer amount of competing automation, combined with increasing time pressure, probably made it difficult for the FO to acknowledge and execute the Captains's suggestion to disengage the Go-around mode. Also, an explanation in terms of cognitive conflict could have been used here in order to account for the unexpected, detected behaviour of the aircraft.

2.4.3. The A320 Mont Sainte-Odile crash

On January 20, 1992, while approaching Strasbourg airport, an A320 crashed on the flanks of Mont Sainte-Odile, killing 87 people, in a case of controlled flight into terrain. The enquiry report (METT, 1993) determined that the crew intended to land in Strasbourg after having flown around the airport, thereby buying themselves some time to finish configuring the aircraft. During the communications with the air traffic manager, it was suggested to them that they land direct without flying around the airport. The crew accepted the new approach. Technically, this had an important implication: there was much less time before landing. A new approach had to be programmed, which in turn implied that a new descent rate would be needed by the autopilot. This rate was calculated and then had to be entered into the flight control unit (see Figure 18).

When programming a descent on an A320, there are two modes to choose from to enter the descent rate:

- *Vertical speed* (*VS)*. The value entered in this mode is converted into a sink rate expressed in hundreds of feet per minute;
- *Flight path angle (FPA)*. The value entered in this mode is converted into a sink rate expressed in degrees.

Figure 18: An A320 flight control unit (© Jerome Meriweather)

The crew had to select the descent mode and enter the descent value into the flight control unit. This is where the crew exhibited mode confusion. They entered 3.3 into the VS mode, thereby programming a descent at 3,300 feet per minute, which is an 11° sink rate instead of the intended 3.3°.

Several factors contributed to the occurrence of this accident (Besnard, 1999). Namely, the late acceptance of a new approach generated an important increase in workload. Also, due to the change of approach, the crew were focused on lateral navigation in order to capture the runway centreline. Last, the landing gear alarm, when it sounded, was not interpreted by the crew as a sign of low altitude. On this latter point, it is interesting to note that alarms do not necessarily cause one to revise their understanding of a situation. In the case of the A320 crew, with their undetected mode confusion, and under high workload, the crew were not likely to suspect that their descent was too fast. As a consequence, the landing gear alarm was ignored. Indeed, it provided information that could not be made consistent with the crew's understanding of their vertical position.

From the point of view of dynamic system control, two combined aspects seem to emerge from the case of this A320 crash:

- *Errors can be made that are not detected*. The cues available to assess the system state might be tenuous, especially just after an unwanted action has been performed, when the system state is unlikely to show any major discrepancy with expectancies from the operators;
- *Chances of recovery can be rejected unknowingly*. Even when information is given to the operators that something is going wrong (e.g. an alarm is sounding), this information may not be interpreted correctly if the revision of their mental model is too costly in terms of time or mental workload.

2.4.4. The Royal Majesty grounding on Nantucket schoals

On June 10, 1995, the Panamanian cruise ship grounded in Massachusetts after a navigation mistake resulting from a combination of technical and HMI-related faults (NTSB, 1997a). On their way from the Bahamas to Boston, the crew had to sail east of the Nantucket shoals (see Figure 19) and avoid their shallow waters. A route had been entered into the GPS-driven autopilot, a device that has the technical capacity to compensate for the drift caused by wind, currents and waves. Unfortunately, the cable feeding the GPS signal to the autopilot had been damaged during repairs. As a consequence, at the time of the accident, the autopilot was unable to adjust the position of the ship. Instead, it had defaulted to a mode called dead reckoning whereby the autopilot maintains the ship onto the same magnetic heading. It follows that the ship was unable to compensate for its westbound drift.

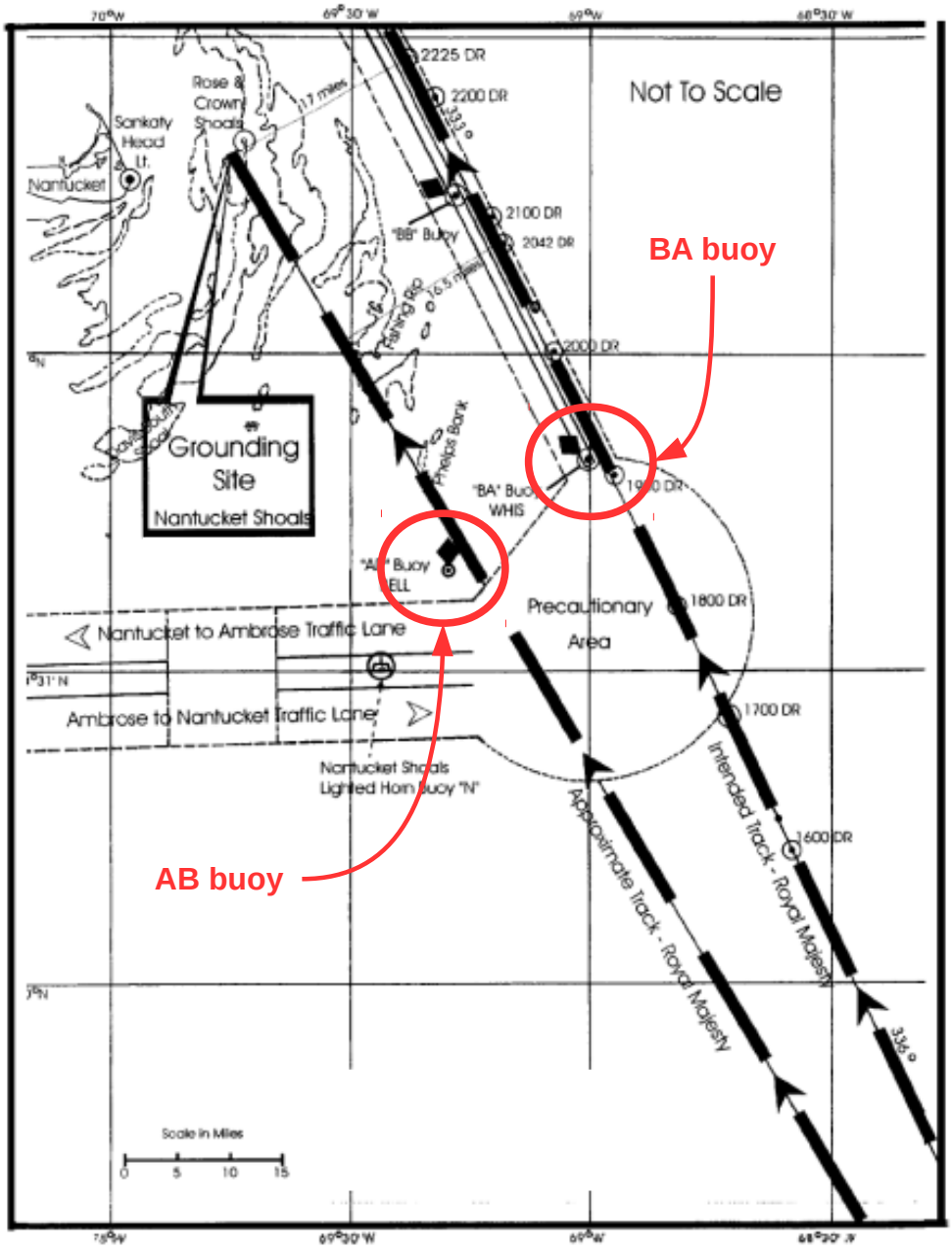


Figure 19: Planned (right) and accidental (left) trajectories of the Royal Majesty. Adapted from NTSB Marine Accident Report NTSB/MAR-97/01. *© NTSB*

The autopilot dead reckoning mode, although displayed on a panel in the instrument room, could not be detected unless one entered the instrument room which was located at the rear of the bridge. This contributed to the crew not detecting the mode change and maintaining an erroneous mental model of the navigation conditions. This erroneous mental model played a central role in the rest of the voyage. Indeed, by Nantucket shoals, the crew had to visually identify a buoy (noted BA on the map; see Figure 19) in order

to confirm the route. The crew member on watch at the time spotted a buoy at the expected time and place. He did not notice that this buoy actually was AB, located 15 miles further to the south-west. Shortly after this event, the strong confirmation of expectancies prevented the crew from correctly interpreting the white and blue reflections on the water ahead of the ship, a set of signs that are typical of waves breaking on a shore. This delayed the change of the course of the ship and grounded the Royal Majesty on a sandy bank. Fortunately, nobody aboard was injured and the double hull of the ship prevented any fuel spillage.

Once again, in this case, it is possible to see the occurrence of a phenomenon depicted in other accident cases analysed in this document: the combination of an incorrect mental model (leading to believe that the ship was still under GPS-driven autopilot) and the resulting misinterpretation of potential recovery cues (white and blue reflections on the water).

2.4.5. Conclusion

In the above examples on mode confusion, I have made the point that operators can find themselves in a situation where their actions do not produce the expected outcome due a misunderstanding of the active mode. Mode confusion has been studied for years within the HMI community. In aviation, this phenomenon is often associated with complex automation design where the latter, in a programmed response to a complex combination of flight conditions, sometimes generates an automatic change in the system configuration that operators cannot always anticipate (Sarter & Woods, 1995; Degani *et al.*, 1997; Rushby *et al.*, 1999; Crow *et al.*, 2000, Leveson *et al.*, 1997; Leveson & Palmer, 1997). The case of the Nagoya tail-first landing might be an exception in this respect. Nevertheless, the crew were confused enough to keep trying to land an aircraft that was basically attempting to increase engine thrust and gain altitude. To some extent, the Mont Sainte-Odile crash was caused by the same mode-related phenomenon, save that the crew had not detected that the inappropriate descent mode was causing too high a sink rate. Last, in the case of the Royal Majesty, the crew did not notice that the GPS signal was not feeding the autopilot and that the latter had reverted to a passive mode.

On the basis of the cases reviewed here, I would like to highlight that mode confusion:

- is linked with the interface design and the feedback given from the engaged or changed mode;
- is transversal to a number of systems (not just aviation);

- can generate catastrophic consequences when combined with time pressure;
- is not always detected, which can potentially leave operators outside of the control loop.

Generally speaking, control can be lost when the automation enters a mode that human operators did not expect or could not understand. As explained in the next section, this is not a phenomenon that is caused by mode confusion alone. Sometimes, it is how operators interpret and use the interface for their decisions that is a major contributor to loss of control. And just as with mode confusion, this can happen without any initial technical problem.

2.5. Interfaces and misinterpretation of system state

In this section, some events that relate to interface-related sub-standard performance will be described and analysed. In these events, I will show how interfaces can a) lead to failures when the data they convey can be interpreted in erroneous ways, and b) when well-known commands are applied to new functions. In order to put theory back into context, three cases will be studied that originate from the nuclear industry (the nuclear accident at Three Mile island), the steelworks industry (a fatal accident at Ascométal), and aircraft piloting (the B737 Kegworth crash-landing), respectively.

2.5.1. The nuclear accident at Three Mile island

On March 28, 1979, at 4am, a number of pumps feeding water to steam generators stopped accidentally at the Three Mile Island nuclear power plant, Pennsylvania, United States[42]. This decrease in the water flow caused pressure to rise in the pressuriser[43] (see Figure 20).

42 The information presented in this section originates from Kemeny (1981) and is reprinted here from Besnard (1999)

43 A vessel in charge of maintaining the primary water circuit under pressure.

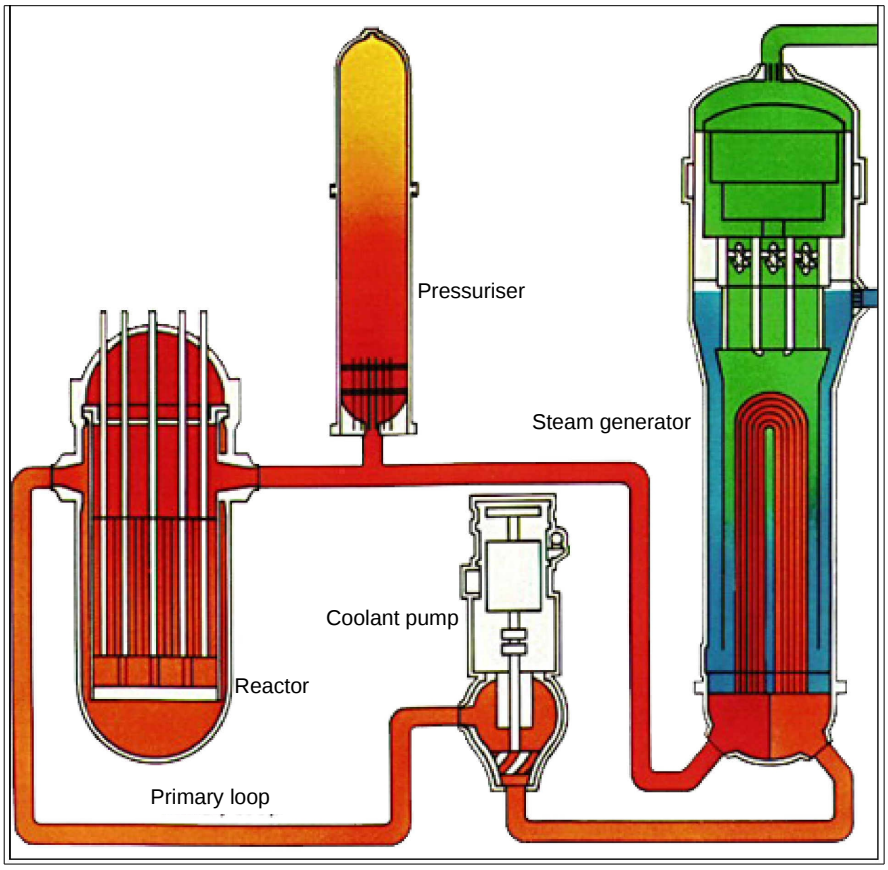


Figure 20: A simplified diagram of a nuclear reactor

In turn, this caused the PORV (Pilot Operated Relief Valve) to open in order to release some pressure. This valve stayed accidentally locked in the open position, causing an excess of cooling water to be purged out of the circuit. The resulting increase in temperature triggered a protection for the reactor core in the form of a scram. This operation consists of inserting graphite rods into the core, in order to absorb neutrons, thereby killing fission. When a reactor scrams, fission is stopped but some fission by-products continue to heat the water around the core. To cope with this residual heat, three HPI (High Pressure Injection) cooling pumps started automatically but two of the three lines were closed on the day of the accident. As a consequence, the heat around the core was not dissipated quickly enough. This caused the pressure in the pressuriser to rise even more, while the pressure in the cooling circuit dropped.

At this stage, the operators erroneously believed that the (high) pressure in the pressuriser accounted for the pressure in the rest of the cooling circuit. Therefore, they were led to believe that the cooling water circuit might be too

full, thereby explaining the excess pressure. Also, the operators knew that the reactor compartment should not be entirely filled with water. Therefore, in their mind, something had to be done with the (suspected) excess cooling water. This contributed significantly to the accident when, in order to compensate for the (actual) lack of cooling water, two emergency pumps started to inject water into the circuit. Given the already high pressure level in the pressuriser, the operators inferred that this extra flow of water might make the state of the reactor even worse. Therefore, they reduced the emergency water flow by 90%. The resulting effect was that temperature rose even more, to such an extent that steam bubbles started displacing core water, thereby contributing to the increase of pressure in the pressuriser. At this point, a vicious circle was in place and the situation was bound to get out of control. At 5am, four cooling pumps vibrated severely due to lack of water in the circuit. At 6:54, an operator attempted to restart a pump but switched it off 19 minutes later due to the vibration level. Shortly afterwards, the operators triggered a site emergency due a risk of contamination of the immediate environment.

At the time of the accident, only one third of the reactor core was covered in water. This lack of water made the zirconium plating of the fuel rods react with the steam. This reaction created a nitrogen bubble that eventually exploded, although without causing any damage. This is the consequence of a partial melting of the core following a water leak and the emergency cooling system having stopped working (Lamarsh, 1981).

This incident shows the effects that an erroneous interpretation can cause in the control of a degraded critical system state. The operators were concerned with too high a water level and had regulated it on the basis of an inadequate indicator: the pressure level in the pressuriser. This fixation on the pressuriser led the operators to maintain an erroneous mental model, thereby making them misinterpret a number of cues related to a water leak through the locked-open PORV valve:

- an abnormally high water level in the retention basin, underneath the core;
- the high temperature of the cooling water drain pipe;
- vibrations from pumps;
- abnormally high neutron rates within the core.

The dynamics of a nuclear reactor and the complexity of its operation can make undetected (and therefore not recovered) mistakes have potentially catastrophic effects. Also, a malfunction can evolve over time in a self-

maintained manner due to a flawed mental model. In the case of Three Mile Island, the water leak caused a temperature rise which in turn was reacted to by operators by draining out more water. This can be seen as a closed, self-degrading control loop. This conjecture is supported by the operators intentionally shutting down the high-pressure injection pumps in order to avoid the rise in the water level. This action deprived the core of an important safety protection but was carried out on the basis of a clear intention, whose origin lies in a flawed representation of the functioning of the reactor. It follows that each corrective action of the operators, based on indirect and invalid cues, contributed to the degradation of the situation.

Another aspect that should be mentioned here is the fact that complex, dynamic systems do not always have a fast feedback loop. This means that a system can exhibit some delay before the effects of an action show. In the case of a corrective action, operators must wait for the system to show a trend towards the expected state so that the effects of an action can be evaluated and adjusted. This makes the piloting of critical systems extremely difficult and is an added risk factor of risk degraded situations.

From a different view point, it also has to be said that some unfulfilled information duties as well as design issues with the control panels contributed to the poor performance of the operators. Indeed, in Babcock and Wilcox plants (such as Three Mile Island) nine PORV valve failures had occurred without the manufacturer informing their clients. Moreover, the PORV valve indicator display did not show the status of the valve (open or closed). Instead, it only showed that an open or close command had been sent. In such conditions, operators cannot know the position of the valve. Also, the lack of routine checks left the erroneous closure of the HPI pump lines unnoticed. Last, more than one hundred alarms were active at the time of the incident and the printouts on the system status were consequently vastly out-of-date.

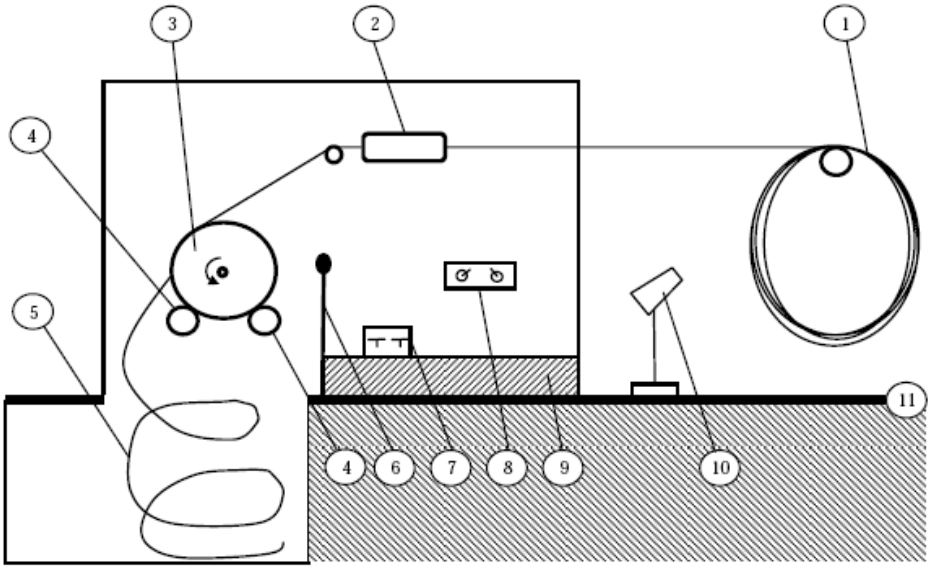
The superimposition of all these factors (the flawed mental model, and the information and design issues above), more than any one of them in particular, is what caused the accident. It is certainly interesting, from a psychological point of view, to note that the recovery from an incident is jeopardised when operators cannot know the actual state of the system (PORV valve and cooling lines closed, for instance). However, operator failures *per se*, as often in complex industrial mishaps, contributed to but did not determine the outcome of the event.

As the next example will show, interfaces and related difficulties of interpretation of the system state can happen in systems that are less

complex than a nuclear power plant. Nevertheless, they can also be at the origin of mishaps and losses. Also, the involved cognitive phenomena show some overlap, especially in the way operators can fail to mentally represent the functioning of the system and the heuristics they may use.

2.5.2. A fatal accident at Ascométal

In March 1990 at Ascométal, a French steelworks factory employing some 500 people, an accident occurred during a night (see Besnard & Cacitti, 2005). An experienced operator was working on a wire drawing machine, a device that reduces the diameter of a metal thread by a series of tractions (see Figure 21). Typically, the output thread is coiled onto a drum and kept in place by pressing wheels. Opening and closing the wheels is done by rotating a two-position button. Because of the high tension of the thread, there are times in the process where opening the pressing wheels is extremely hazardous.



Legend

1- Input thread
2- Diameter reduction tool
3- Coiling drum
4- Pressing wheel
5- Output thread coiling in the pit
6- Safety barrier
7- Coiling drum control pads
8- Pressing wheels control buttons
9- Operator's platform
10- Main control panel
11- Ground level

Figure 21: Diagram of the wire drawing machine

The operators worked on eleven wire drawing machines, ten of which were

operated in a similar way. On the machine involved in the accident, the open and closed positions of the pressing wheels button were swapped, compared to the ten other machines. This swap was well-known but was not flagged or equipped with any kind of protection. Because of the swapped commands, the operator unintentionally opened the pressing wheels at a step of the process where this action is forbidden. The operator was violently hit by the thread uncoiling from the drum, and died from his injuries.

Psychologically speaking, the accident did not occur simply because the operator took an incorrect action (see Doireau *et al*., 1995). Indeed, this would simply be a judgement made in hindsight and asserted without proper account of operating conditions. Instead, the accident occurred because of a range of interfaces being prone to confusion, in a critical environment, with no protection against unsafe acts.

When a tool changes, for instance as the result of an upgrade, skills must adapt accordingly in order to reflect the changes and maintain the accuracy of the interaction. But updating skills requires repetitive feedback from the system in a wide variety of cases so that operators can progressively reduce the discrepancies between the system's expected behaviour and the system's actual behaviour. During this sensitive period, unwanted actions in critical functions of a hazardous tool can be fatal.

To quantitatively assess the extent to which negative transfer was a potential explanation for this steelworks accident, Lucile Cacitti designed an experimental task (see screenshot in Figure 22) where subjects first had to learn how to use a keyboard-based interface (see the key-function mapping in Figure 23) and perform a number of actions (basically selecting an item on the screen and moving it from one box to the other).

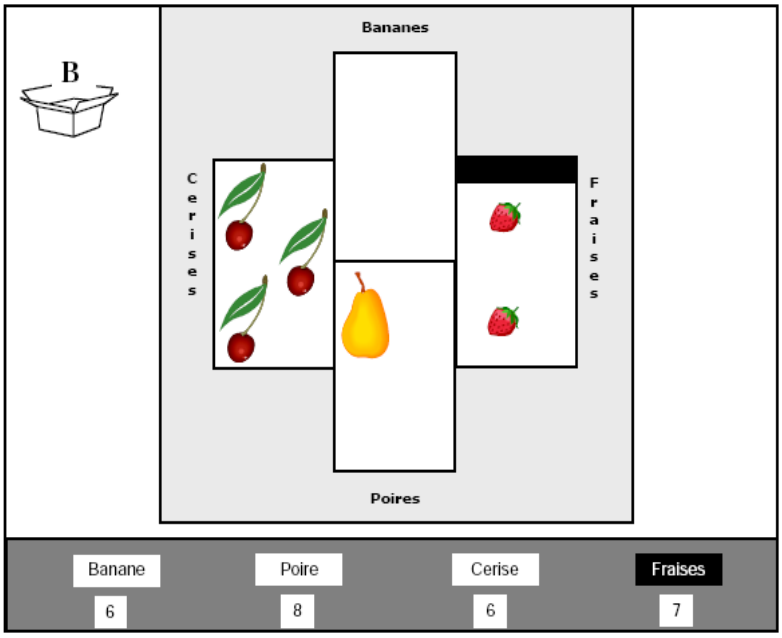


Figure 22: Screenshot of the interface in the control condition

It is only when subjects had achieved the task without a single error (at which stage they knew the keyboard interface perfectly) that the assignment of the functions to the keys was changed. With this new key-function mapping, the subjects now had to perform the same task again. The difference in performance would then be measured (via the number and type of errors performed) with the hypothesis that the number of transfer-related errors would grow with the similarity of the new interface. In other words, the easier the old and new interfaces were to confuse, the higher the failure rate should be.

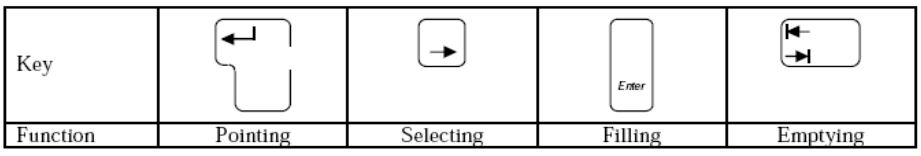


Figure 23: Key-function mapping for the control condition

The new interface was a new keyboard map with swapped keys and functions (the swapped interface) or four buttons added to the screen itself (the on-screen interface), corresponding to the same functions. The results show (see Figure 24) that the on-screen interface generates significantly lower error

rates than the swapped interface. Indeed, over the course of 3 trials with the new interface, the subjects in the on-screen condition were exhibiting an error rate almost 3 times lower than the one of the swapped condition.

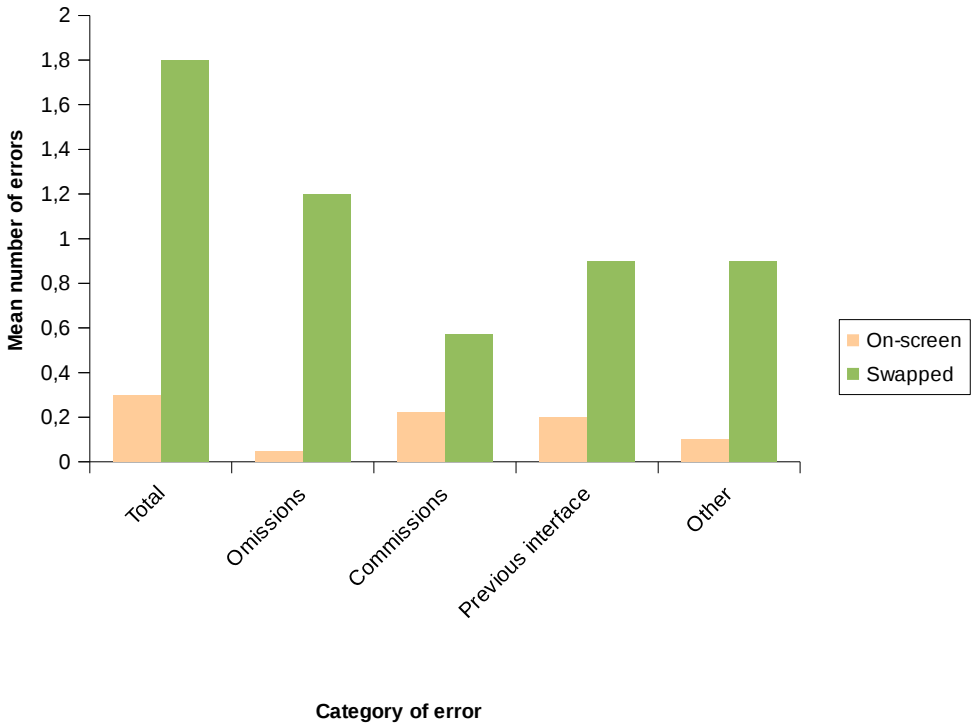


Figure 24: Graphical representation of the significant means

Reason (1990) and Decortis (1993) suggest that human reasoning globally obeys two heuristics: frequency gambling and similarity matching. The research presented here is clearly concerned with these concepts. When a novel interface shares features with a previous one, action patterns that relied on these features in the past tend to be imported and potentially unduly reused. This reuse strategy, underpinned by cognitive cost avoidance, is the cause of negative transfer.

The two devices (the computerised simulation and the wire drawing machine) seem to each provide an instance of the above mentioned similarity matching and frequency gambling heuristics, respectively. From this standpoint, the relative *similarity* between the interface in the control condition and the swapped interface condition caused a key-function mapping to persist across conditions. In the case of the actual interface of the wire drawing machine, the steelworks operator rather relied on *frequency* gambling, due to the particular machine's interface standing alone within a set of several similar

machines. Some aspects that were not investigated in this research were broader contextual elements such as performance shaping factors (as described in Miller & Swain, 1987) or common performance conditions (Hollnagel, 1998). These conditions, combined with the cognitive resources saving strategy, might have contributed to the interface discrepancy being overlooked, thereby letting the most frequent routine take over the choice of actions.

The cognitive aspects of the misinterpretation of the system state can also be highlighted in aviation. In this domain, there exists at least one case where operators were confused by the interface and then misled by the system's behaviour in response to their actions. This is what the next section aims to discuss.

2.5.3. The B737 Kegworth crash-landing

On January 8, 1989, a British Midland Airways Boeing 737-400 aircraft crashed into the embankment of the M1 motorway near Kegworth (Leicestershire, UK), resulting in the loss of 47 lives (AAIB, 1990). The crash resulted from the flight crew’s management of a mechanical incident in the left (#1) engine. A fan blade detached from the engine, resulting in vibration (severe enough to be felt by the crew) and the production of smoke and fumes that were drawn into the aircraft through the air conditioning system. The flight crew mistakenly identified the faulty engine as the right (#2) engine. The cockpit voice recorder showed that there was some hesitation in making the identification. When the captain asked which engine was faulty, the first officer replied "*It’s the le… it’s the right one*", at which point the right engine was throttled back and eventually shut down.

This action coincided with a drop in vibration and the cessation of smoke and fumes from the left (faulty) engine (as highlighted by Besnard, Greathead & Baxter, 2004). On the basis on these symptoms, the flight crew deduced that the correct decision had been taken, and sought to make an emergency landing at East Midlands airport. The left engine continued to show an abnormal level of vibration for some minutes, although this seems to have passed unnoticed by the pilots. Soon afterwards, the crew reduced power to this engine to begin descent, whereupon the vibration in the engine dropped to a point a little above normal. Approximately ten minutes later, power to the left engine was increased to maintain altitude during the final stages of descent. This resulted in greatly increased vibration, the loss of power in that engine and the generation of an associated fire warning. The crew attempted at this point to restart the right engine but this was not achieved in the time

before impact, which occurred 0.5 nautical miles from the runway.

In addition to the crew's confusion, several other factors contributed to the accident. When later interviewed, both pilots indicated that neither of them remembered seeing any indications of high vibration on the Engine Instrument System (EIS; see Figure 25). The captain stated that he rarely scanned the vibration gauges because, in his experience, he had found them to be unreliable in other aircraft. It is also worth noting that the aircraft was using a new EIS which used digital displays rather than mechanical pointers. In a survey carried out in June 1989 (summarised in the AAIB accident report), 64% of British Midland Airways pilots indicated that the new digital EIS was not effective in drawing their attention to rapid changes in engine parameters and 74% preferred the former mechanical EIS.

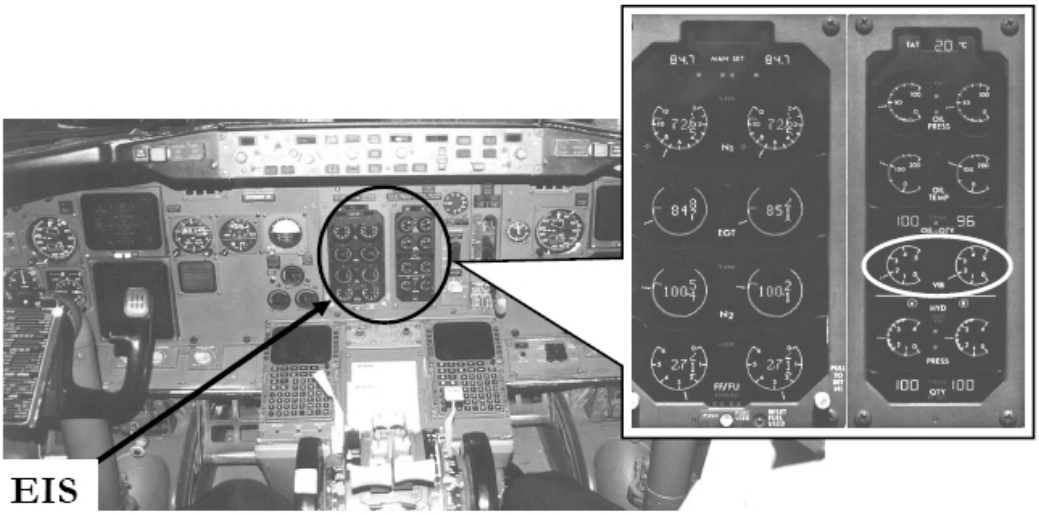


Figure 25: A Boeing 737-400 cockpit. The EIS is located in the centre (© Pedro Becken; all rights reserved). The secondary EIS is magnified on the right-hand side of the figure. The vibration indicators are circled in white (photo from Air Accident Investigation Branch accident report).

As another contributing factor, the crew workload increased out of control. Some time after the #2 (working) engine had been erroneously shut down, the captain tried without success to stay in phase with the evolution of the incident. He was heard on the cockpit voice recorder saying: "*Now what indications did we actually get (it) just rapid vibrations in the aeroplane – smoke…*". At this point, the crew were interrupted with a radio communication from air traffic control. Later, the flight service manager

entered the flight deck and reported that the passengers were very panicky. This further distracted the flight crew and the captain had to broadcast a message of reassurance to the passengers. Both the captain and first officer were also required to make further radio communications and perform other duties in preparation for the landing. All of these actions affected the degree of control of the emergency.

Although the interface-related aspects of the Kegworth accident (as analysed by Besnard, Greathead & Baxter, 2004) were published several years after the accident report, the confirmation bias triggered by an erroneous information taking had not been studied before. From this point of view, some progress was made in understanding the psychological mechanisms that were causally involved in the crash. From a more general standpoint, there was also some value in achieving this understanding for at least two reasons:

- The confirmation bias, which contributed to maintain an erroneous mental model, is believed to be independent from the domain of operation or the interface in use;
- This bias may also be independent from the operators' general level of experience. Indeed, the captain and the first officer had only 76 hours experience between them on the Boeing 737-400 series, but more than 16,000 hours flying altogether.

2.5.4. Conclusion

Operators can erroneously maintain as valid, representations that have already departed from reality. In dynamic situations, one reason is that operators try to avoid the cost of revising their mental model as long as it allows them to stay more or less in control. The vagueness of the wording is deliberate here since operators, due to limits such as bounded rationality, tend to accept imperfect solutions to problems; a phenomenon called *satisficing* by Simon (1957). Mental models are constantly compared against the feedback from the process under control. However, there exist situations where this feedback is not compatible with operators' mental model. This incompatibility, when detected, usually creates a surprise, such as partial loss of control. When this happens, some costly revision of the mental model as well as diagnostic actions are needed (Rasmussen, 1993). However, the detection of the incompatibility between one's mental model and the process at hand is not automatic. When undetected, some loss of control happens unknowingly, with the potential to degrade progressively, without intervention from operators.

Cognitive conflicts, mode confusions and misleading interfaces are factors that degrade performance in potentially any supervisory monitoring activity. They are related to an incompatibility between the mental representation of the system that the operator maintains and the actual system. The occurrence of cognitive conflicts can be facilitated by over-computerised environments. Of course, computer-based critical systems do not necessarily trigger failures. However, given the increased complexity of the situations that are managed by software, operators lose more and more control over a growing number of processes.

2.6. Contribution to the field and future challenges

In this third chapter, the emphasis was put on cognition in dynamic systems. I have analysed dynamic systems from a cognitive angle and attempted to understand why they pose problems to human reasoning. The conclusion is that these systems are sometimes so complex, designed in such opaque ways, and impose such constraints (typically: cognitive effort under time pressure) that human understanding itself can be jeopardised. In my opinion, it is this combination of factors that leads to HMI flaws such as cognitive conflicts, or mode confusion, which then impair mental models, and finally contribute to loss of control (see Figure 26).

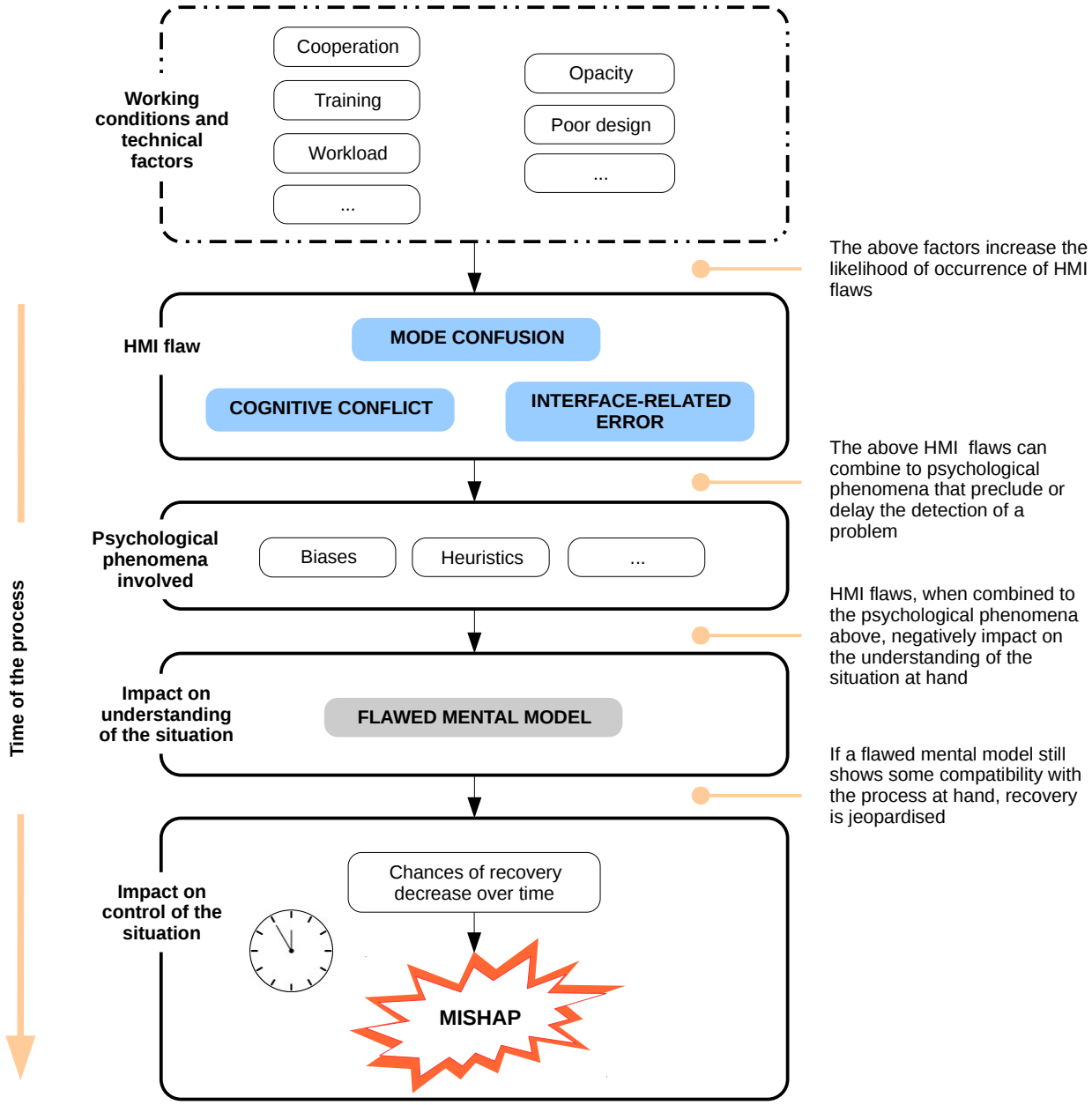


Figure 26: The contribution of flawed mental models to mishaps in dynamic systems.

The difficulties that operators sometimes experience in the understanding of the system state are factors that Perrow (1984) identified as contributors to accidents, especially in complex, tightly-coupled systems such as the Three Mile Island nuclear power plant. These systems, being characterised by such features as e.g. low tolerance to delays, little slack and sequence-dependent, can make the recovery of unwanted events extremely difficult. More

precisely, the interfaces of complex systems (and their potential opacity) can disrupt the interpretation that operators make of some patterns of symptoms, thereby delaying or precluding problem detection (Klein *et al.*, 2005). This, in turn, can leave an abnormal situations undetected, and degrade without correction from operators.

To my cognitive, HMI-centred approach, it should be added that there are so-called secondary failures (e.g. in system maintenance, hardware design and layout, software programming practices, etc.). These failures often remain largely unexplored by accident investigations (and therefore tend to remain unexplained), but nevertheless contribute to HMI-related mishaps (Johnson, 2004). It is not exactly the angle adopted in this thesis. However, it is a view that is valuable when one wants to depart from the short-sighted vision of the sharp-end operator error as the cause of mishaps, and consider the variety of nature and depth of all the contributing factors to virtually any event.

As far as its contents are concerned, this chapter has relied on a number of publications. The issue of cognitive conflicts in the domain of aviation was analysed in a number of works (Besnard & Baxter, 2006; Baxter, Besnard & Riley, 2007). Namely, two tragic events have been studied from this angle: the crash of a B757 in Cali (Columbia) in 1995, and a DC-9 wheels-up landing in Houston (Texas) in 1996. This interest in aviation also transpired in my PhD, namely in the analysis of the crash of the A320 on the Mont Sainte-Odile in 1992 (Besnard, 1999). However, other domains and events have been studied, such as the Three Mile Island nuclear accident in 1979 (Besnard, 1999). Also, following a field study in a steelworks factory, an experiment on interfaces and related misinterpretation of system state was published (Besnard & Cacitti, 2005). Last but not least, a re-analysis of the B737 crash in Kegworth, highlighting the effects of co-occurrences in high-tempo systems, was performed (Besnard, Greathead & Baxter, 2004). This latter publication was widely advertised on the British news and on the Internet since it appeared 15 years to the day after what is still today the most tragic accident in recent English aviation history.

I would now like to look into the future challenge posed by human-machine interaction in dynamic, critical systems. In my opinion, one transversal dimension emerges that impacts heavily on performance: time. In dynamic, critical systems, it combines with other parameters (such as the presence of an undetected error or lack of anticipation) and degrades the performance of the system. From this, and in light of the material discussed above, I conclude that operators of dynamic, critical systems need to anticipate the future states of the process they have under their control. This is imposed on them

by the excessive complexity of real-time control. However, to date, despite the advances of computing techniques and cognition-centred HMI design, operators still have to rely on data from instruments and past events from the process to mentally derive the future possible states of the task under control. So here is the question that I ask myself: what type of decision-making assistant system could help operators foresee upcoming events and generate some proactive advice?

One immediate objection might be that some situations show such a high tempo or such a high degree of emergency that relying on an assistant system to evaluate candidate decisions is not a viable option. Actually, this question raises once again issue of anticipation. Indeed, a large proportion of safety-critical situations in dynamic systems (aicraft for instance) are no longer caused by technical failures. Instead, many mishaps happen because an anomaly has not been detected in time (due, for instance, to automation surprise), thereby leaving the situation to degrade without correction. A typical case in aviation is CFIT[44], as in the Cali crash. Therefore, it is in detecting *early* signs of potential loss of control that operators need support. Indeed, being flagged, and reacting to, a potentially degrading situation is where the chances of recovery might be highest. Such assistance could be provided by forecasting the effects of an action on various time scales, and returning a series of candidate corrections to the operator. Given that reactive control in degraded conditions and time pressure leaves little room for recovery, assistance has to come before control starts to degrade. In this perspective, a timely and proactive assisting agent can have a potential positive impact on the performance of HMI in dynamic, critical systems. This will be addressed in detail in the final chapter of this thesis.

For the time being, I would like to keep widening the scope of my reflections. After having addressed cognitive performance in static and dynamic systems, I now need to address socio-technical systems. In doing so, I will keep the cognitive angle that I have adopted so far, but adopt a higher point of view in order to consider the larger system surrounding human activities.

44 Controlled Flight Into Terrain

Chapter 3. Human Cognitive Performance In Socio-Technical Systems

In the cognitive ergonomics' view, the reasoning modes that humans exhibit at work are based on heuristic shortcuts built on top of the experience acquired through a life-long dynamic interaction with a diverse and changing environment. Also, the principle of economy of resources operates on the basis of a trade-off between saving efforts and a perfect response to the environment. This trade-off covers a very wide continuum that allows some room for imperfections. However unsuitable to critical processes it may appear, this trade-off provides the flexibility that is required to perform actions in response to unknown problems. In the cognitive ergonomics conception, humans are no longer regarded as static components of a system. They are conceived as agents dedicating their mental resources to adapting themselves to varying environments, dealing with unknown situations and potentially contributing to system safety.

As touched upon elsewhere (Besnard, 2003), ergonomics has a much larger interest in work (i.e. people operating systems in a given environment) than just human-machine interaction. The way people process information and try to do their job has an influence on the whole system. Acknowledging this leads one to accept the viability of the concept of a socio-technical system as one composed of interacting and interdependent human, technical and organisational components. Without delving into sociology, it is fairly obvious that this denomination better accounts for how activities happen in the real world, beyond the reductive (yet immensely complex) sphere of an individual interaction with an interface.

Due to the increase of critical functions allocated to automatic agents (e.g. computers), the safety of socio-technical systems is an area where the stakes are continually being raised (Besnard & Greathead, 2003). But reducing these systems down to a set of pure technical components would discard a

very hot topic: deterministic automatic machines cohabit with humans who are in essence non-deterministic. As the actions of humans can impact very strongly on the final safety of any system where they are present, it is worth questioning the issue of the integration of humans into socio-technical systems. After Reason (1990; 1997), it is believed here that a combination of organisational and local individual factors offers an interesting analytical framework for discussing human actions. Indeed, the performance of the so-called socio-technical system can be seen as the result of a collaboration between humans, technical tools and procedures, all embedded in complex operational, political and economic interdependencies. This collaboration, partly because of the many and powerful tensions that affect the operators at the sharp-end, never achieves its objectives in a perfect manner. If one takes the example of the capsizing of the Herald of Free Enterprise (Sheen, 1987), the performance of a socio-technical system is clearly the outcome of a trade-off process that takes place between humans and their task, given a set of operational constraints. One potential outcome of such a combination is a progressive drift of practices away from the safe operational envelope, a dimensions identified by Fadier & De La Garza (2006).

In this chapter, several activities and industries will be analysed from a socio-technical perspective, such as aviation, medical systems and navigation. Each time, I will look into the activities of an individual or a small team within these systems, and try to understand how their interaction shapes the dependability[45] of the system. Sometimes, the interaction will compensate for the limitations of humans or machines. At other times, the interaction will degrade into an incident or an accident. Through this demonstration, the focus will be set on the conditions in which people were operating (be it cognitive, physical, managerial and so on) and on the impact of these conditions on the performance of the system. The point is to try to find an answer to the following question: how do socio-technical systems succeed or fail?

3.1. Prescribed work and variations

Even when work is precisely defined and supported by procedures, humans develop idiosyncratic operating modes. The objective is adapting work to personal habits or operational constraints. However, as Hoc (1996) notes,

45 Dependability is a global property of a system enabling it to do what it is supposed to do. Dependability is decomposed into a number of dimensions such as reliability, safety, availability, security, integrity and maintainability (Randell, 2000).

these adaptations are violations when they are deliberate. However, this certainly does not imply that their consequences on the system will be that of decreasing performance, far from it.

Generally speaking, automation design is confronted with the problem of exception handling, that is the processing of situations that designers could not think of. This leads to an ironic situation that Bainbridge (1987) identified: operators are trained to follow procedures and are then expected to provide intelligence in unexpected situations. Beyond this irony, exception handling is a recurrent problem in virtually any socio-technical system and so far, it is obvious that the solution does not lie in trying harder to anticipate every possible scenario. Instead, a more productive attitude is to consider that humans have adaptive and analytic capacities (that are difficult to turn into automated functions), and that can compensate for imperfect technical solutions or procedures (Dekker, 2003 ; Vanderhaegen, 2003). From this point of view, the contribution of a variation (to a procedure) to the performance of the socio-technical system is important to understand, as well how the variation redefines the task, and what shortcomings it compensates for (Poyet, 1990). This is especially important with modern systems where automation was initially believed to ease human work when it has in fact reshaped it, creating unanticipated constraints and introduced new sources of failures (Dekker & Woods, 1999).

3.1.1. The many meanings of variations

First of all, one must remember that human performance is inherently variable, which implies that it might fluctuate around any optimum one might want to think of, or from any sequence of actions the operator is supposed to follow. Of course, these fluctuations can be caused by performance conditions such as lack of training, fatigue, etc. In this case, they are an expression of the sensitivity of human performance to a number of contextual factors. For instance, operators can skip a step in a procedure when they have reached a level of fatigue that impairs their capacity of attention. On the other hand, variations can be driven by an intention. In this case, the performance conditions are not the main factor for their occurrence (although they might still contribute). Instead, operators can intentionally follow an objective that the procedure does not include or allow. For instance, a maintenance operator does not stop the machine in order to clean it and their hand gets caught by a moving part. So here are two basic cases of variations: unintentional *vs.* intentional, making intention is my first dimension for a classification of variations. From this, unintentional variations can be seen as

actions of sub-standard performance, whereas intentional variations are adaptations.

Because sub-standard performance and some of its causes have been already addressed in this document, it might be more constructive to understand adaptations, especially because some of them might be allowed by procedures, whereas others might not be covered and are therefore unsupported. To make this distinction, the dimension of legitimacy of actions needs to be introduced. Indeed, there might be cases where adaptations are both needed and supported in order to take into account variability in the system's operational conditions. For instance, a procedure might include such a recommendation as "*If gauge X displays a reading above 250 degrees, then the flow of coolant must be adapted accordingly*". In this case, the procedure leaves the precise sequence of required actions undefined and relies on the operators' knowledge and experience. Conversely, there are cases of illegitimate adaptive actions, that still have an objective, but that fall outside of procedures. These are what Reason (1990) called violations. There are several cases of such actions (workarounds, sabotage, wild experimentation, and so on) and the literature covers many descriptions of them (see later in this chapter). They are interesting to study because they pose the question of the match between procedures and people's intentions and constraints, a topic that will be covered in this section.

Another dimension must be introduced in order to fully capture the nature and consequences of variations: that of anticipation of the system's performance. Indeed, operators performing a violation can hold a mental model that shows some degree of adequacy for their actions, which in turn will determine the outcome of actions on the system. Anticipation is also present in the view of Vanderhaegen (2003) in the form of cost/benefit expectations that are built by operators, when violations are performed, for instance.

Last, it would be an oversimplification to believe that variations invariably have a negative impact on the performance of a system or on its safety level. Even gross violations sometimes are extremely beneficial to systems, including to the lives of their users. This is what could be captured by the dimension of the *impact* of actions on the system. A number of facets are at play here, the most obvious one being the extent to which operators can understand the consequences of their actions, a point that stems from what has been said above.

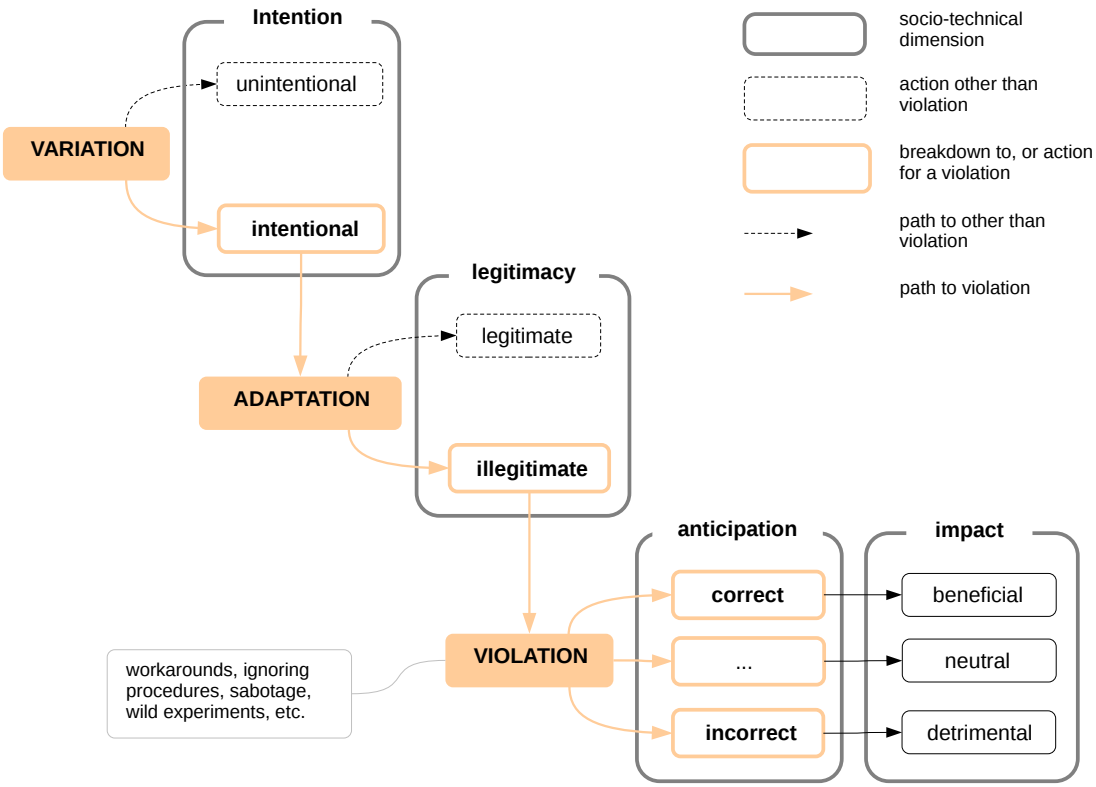


Figure 27: A classification of variations

Figure 27 presents a hindsight view of variations and possible paths. In this figure, the observer stands at the right-hand side and observes (in the workplace, for instance) the impact of the variation. This figure summarises some important dimensions to take into account when analysing variations:

- *intention*: is the variation intentional or not?
- *legitimacy:* is this variation supported (or allowed) by the system?
- *anticipation*: are the effects of this variation correctly anticipated by operators?
- *impact:* does this variation have a positive, negative or neutral impact on the system?

These are the dimensions that will be used as a reference for the discussion that follows later in this section about the human contribution to socio-technical systems. This discussion will be based on a number of real, documented cases from a variety of industrial sectors.

3.1.2. Violations mean little by themselves

Violations are usually seen as a negative contribution to system performance. Typically, instances of violations include: ignoring rules, sabotage, vandalism, etc. These examples emphasise the detrimental effects to system performance whereas the term violation by itself does not say much about the consequences on the system. In reality, the adaptive role of violations is not often highlighted. This overlooks an important facet of violations: they can contribute to system performance by compensating for the underspecification of procedures or the occurrence of an exceptional event (Reason, 2001). Therefore, one needs to understand the conditions under which violations can have a positive impact on system performance.

According to Reason (1990), violations can be seen as deliberate actions that depart from the practices that designers and regulators have defined as necessary. Violations have been mentioned or studied in a wide variety of contexts including car driving (Blockey & Hartley, 1995; Parker *et al.*, 1995; Aberg & Rimmö, 1998), aircraft piloting (Air France, 1997), large-scale accidents (Reason, 1990), computer programming (Soloway *et al.*, 1988) and bureaucratic environments (Damania, 2001). They are actions that intentionally break procedures (Reason, 1987; Parker *et al.*, 1995), usually aiming at easing the execution of a given task. They may reveal the existence of faulty organisational settings when they are the only way to get the work done (Air France, 1997). In this latter case, these violations are the result of latent organisational factors leading to the rules or procedures being broken in order to accomplish a given task. These latent factors are usually implemented by actors who are remote from the resulting risks (Reason, 1995; 1997) such as managers, company directors or even industry regulators.

As already said, violations should not be directly associated with accidents. They can have a variety of impacts on systems performance, ranging from life saving to catastrophic. However, an angle that needs to be highlighted is the effects of the combination of violations and errors, which Reason (1987) calls aberrant behaviour. After Hollnagel (1993)[46], errors are seen here as an unintentional behaviour that fails to produce an expected outcome and may lead to unwanted consequences. The harmful side of the combination of violations and errors stems from the unprotected (violated) working conditions that can potentially preclude the recovery of errors. When such conditions are in place, the normal variability of human performance cannot be compensated for, and a human failure can then degrade into major

46 See this author for a review of various classifications of errors

mishap. This is also true for system performance: the normal variability of performance can be jeopardised by such factors as sub-standard practices, flawed safety culture, etc. This is a view that James Reason (1990) promoted as the contributor to major industrial accidents. Indeed, major accidents in large-scale systems such as the Chernobyl nuclear plant (Gitus, 1988) exhibit this combination, which often finds its roots in a variety of cultural, managerial and organisational factors (Cacciabue, 2000; Reason, 1997).

3.1.3. Prescription of work: procedures

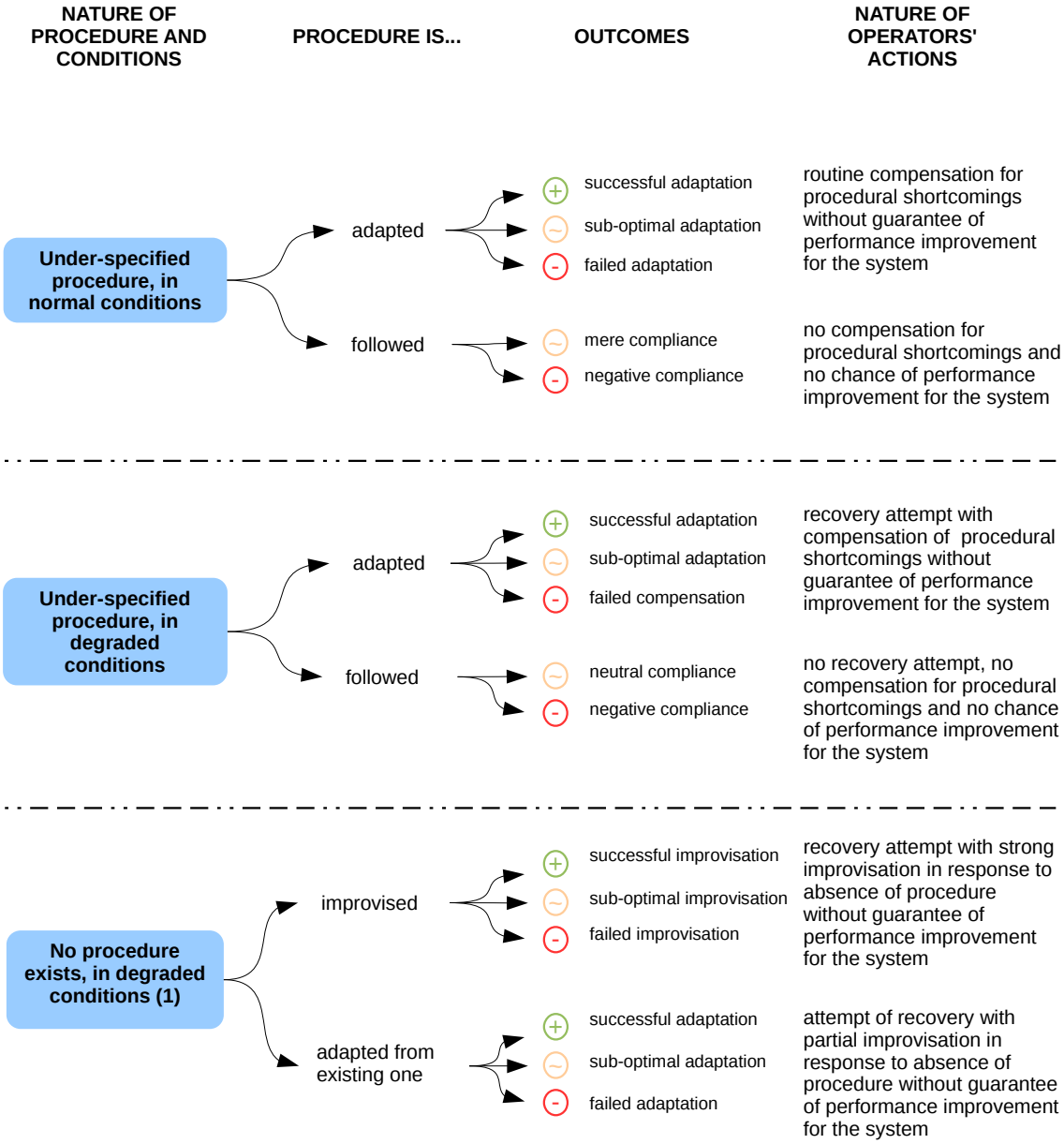
So far, the topic of variations has occupied a significant amount of space in this section, the point being to list their various types and how they relate to one another. However, because variations relate to procedures, this latter issue will now be addressed, particularly from the angle of adaptations. In a previous work (Besnard, 2006), procedures were analysed from the perspective of socio-technical systems' dependability. Within this view, three gross categories of procedures can be distinguished:

- *Procedures for normal conditions*. In this case, procedures attempt to assist in keeping the interaction with the system within a given safety envelope and also provide a predefined way of achieving an objective. Examples include routine scans within aircraft cockpits, and radio communications at predefined points of the navigation plan.
- *Procedures for degraded conditions*. The terms *degraded conditions* refer to the occurrence of an unwanted event that is covered by a procedure. Here, the procedure typically prescribes a series of actions that are meant to limit the propagation of the event and its consequences (e.g. a fire). These actions aim at restoring normal conditions or revert to a predefined safe state[47].
- *No procedure exists*. In this case, the system faces an abnormal event that is not covered by a procedure. This is where humans are most needed in order to control the system according the prevailing conditions, on the basis of their knowledge and experience. This is the case of the DC-10 crash landing in Sioux City described in section 3.2.3.

A distinction that I added to these simple categories is that of the recipient of the procedure (Besnard, *op. cit.*). If one accepts that automation (e.g. modern computer-driven flight control systems) operates on the basis of a kind of procedure (computer code), then a major difference that appears between humans and automation is that of adaptability. On this front, machines are typically blind servants that show limited capacities for capturing variations

47 This is found in aviation in the form of the Quick Reference Handbook.

in their conditions of operation. On the other hand, humans show innate adaptive capacities that are an important contribution to systems' performance. This contribution ranges from very positive to very negative in terms of impact on the system, depending on such factors as characterisation of the system state, safety culture, experience of the operators, etc. These are the dimensions that Figure 28 attempts to capture.



(1) it is assumed that *No procedure in normal conditions*, a case that does not appear here, is an exception that belongs to the top case of this classification

Figure 28: Procedures, conditions, consequences and related actions

This is a point that deserves some attention and that will be returned to later in this section. For the moment, I need to address some general issues about procedures and how humans at work relate to them. This can be summarised by the following:

- *Procedures are often underspecified*. Any complex work environment, such as process control, displays a variability that cannot be fully anticipated, neither in amplitude nor in nature. It follows that related tasks will necessarily include situations that will not be supported by a procedure, and/or which operators will not have been trained for (a situation called the *envisaged world problem*; Woods & Dekker, 2000). Also, following procedures when they exist can create a tension between performing the task as prescribed and getting the work done. This is the typical case of productivity and safety, the two often conflicting with each other.
- *Procedures are a resource for action.* According to Furuta *et al*. (2000), procedures assist work whereas procedure designers might tend to think of them as a precise, exhaustive and prescriptive guidance on how to perform a task. This latter view, with its prescriptive bias, might lead one to believe that procedures are a fair description of operators' work. Of course, this is rarely the case. A normative bias is also at play here, in that procedures are often taken as the one and only point of reference for assessing acceptability of system performance. Until the late 1980s and the seminal works of James Reason and Jens Rasmussen (for instance), this bias led industry to treat a plethora of incidents and accidents as being caused by the mystical "human error" when actually procedures were incomplete, out-of-date or unadapted to the situation at hand.
- *Procedures get adapted to changing conditions*. Humans constantly compensate for the discrepancies they perceive between the objectives of the task at hand and the work as prescribed. This capacity introduces a large amount of adaptivity into systems, even in very procedure-driven environments where contradictions in procedures exist (Vincente, 1999). This allows situations that were not anticipated by designers to be handled by the socio-technical system instead of generating a mishap. This latter case is a challenge in fully-automated systems. An infamous example is the self-destruction of the Ariane 5 launcher (Lions, 1996).

Well beyond any normative view of human performance in terms of

compliance to procedures, I will show that human-system cooperation implies tensions and trade-offs which generally reflect the collaborative aspects of socio-technical systems' operation. Departing from the rather neutral presentation of procedures that was laid out so far, the factors (and related cases) of positive and negative[48] human-system cooperation will be assessed. Namely, humans and the technical system will be presented as two categories of agents, each contributing to the achievement of an objective (e.g. some form of operation or production) by the joint, socio-technical system (Besnard & Jones, 2004; Besnard, 2003; Besnard & Baxter, 2003). Because humans and technology can compensate for each other's limitations, it is worth investigating the type of compensations that can occur, and the type of contribution this makes to the general socio-technical system performance.

3.2. Positive cooperation in socio-technical systems

In this section on positive cooperation, as well as in the one on negative cooperation, one will notice that I make a distinction between workarounds and violations. What is meant to be highlighted with this distinction is the nature of the variation with regard to procedures. In this view, I treat workarounds as an illegitimate variation :

- of limited magnitude (in nature) as compared to what procedures prescribe ;
- usually breach a limited set of procedures ;
- usually aim at easing the execution of a specific aspect of the task ;
- that is often part of the daily practice on the job ;
- that might leave operators in unprotected conditions ;
- that usually offer possibilities of adjustment from the system feedback ;
- that usually offer possibilities of error recovery.

Conversely, I treat violations as illegitimate variations :

- of rather large magnitude (in nature), although there can be a progressive drift ;
- can potentially breach a large set of procedures ;
- can redefine the execution of an entire task ;
- that can be part of the daily practice but can also be exceptional in nature (following Reason,1990) ;
- that can leave operators in grossly unprotected conditions ;

48 I acknowledge that these two terms bear a normative value. However, I wish to use them for their simplicity. Alternative, neutral terms, although less immediate in their meaning, could be wanted *vs*. unwanted.

- that might not offer possibilities of adjustment ;
- that might not offer possibilities of error recovery.

This set of properties is rather generic and draws a rather strict boundary between two categories of actions. Such a distinction can also be found in Reason (1990) under the terms routine *vs*. exceptional violations. Actually, a phenomenon such as the drift into failure calls for a continuum rather than a discrete classification. Also, I should make clear, and consistent with my view in Figure 27 (p. 85), that I treat workarounds as a type of violation. Last, the predictability of one's actions was not included in this description. All these points will be addressed in the following analysis of some cases of cooperation between humans and the system.

3.2.1. Compensation for human limitations by the system: assistance functions

Humans can be seen as cognitive agents that are limited in capacity, speed and precision, but that compensate for these limitations by saving resources and implementing heuristic decision-making. Despite these palliative strategies, failures happen due, in part, to the inherent fallible nature of humans and the wide variability of working conditions. This is where automation can assist humans and enhance performance.

A simple example of such a case can be found in computing where numerous applications (word processors, spreadsheet, image processing, etc.) allow users to undo one or several of their actions. In commercial aviation, some very elaborate pieces of technology (e.g. the Flight Management System) assist humans in such functions as keeping the aircraft on the flight plan with only a few human actions required. Other assistance systems can be found in healthcare, where a range of a patient's vital parameters can be placed under automated control.

All these technical systems provide a number of advantages over their human counterparts, namely:

- they can fulfil a task with a degree of precision that exceeds what humans can produce;
- they can compensate for human cognitive limitations (especially attention).

Of course, the purpose of this short section is to state that the technical components of socio-technical systems contribute to the system reliability by e.g. assisting humans when high precision is needed over long periods of time. But despite the positive and simplistic picture presented here, there are many classes of situations where it is humans that "save the day" by

exhibiting behaviours that deal with a problematic situation (in the case of ill-designed systems, for instance) or by inventing a response to a highly unlikely system state.

3.2.2. Compensation for system limitations by humans: positive workarounds

Besnard and Baxter (2003) take the example of a study done by Clarke *et al.* (2003) that shows how humans adapt their work to local contingencies. An ethnographic study was conducted at a steelworks factory producing steel slabs of varying sizes. The study focused on the use of a computer-driven roughing mill. Slabs are produced to a required size by rolling large metallic rolls in a series of passes. Because there are several qualities of steel and a variety of ways to reach the final slab's dimensions, operators have developed various work strategies. Some of them override the computer's control. For instance, operators sometimes shift to manual control mode for the final passes on slabs of a particular thickness. In doing so, they reduce the number of passes in order to avoid slabs taking a U-shape, an occurrence which is not always avoidable under computer mode. As quoted from Clarke *et al.* (2003): *"…because the computer, at less than 45, pisses about…does 4-5 passes… that's what's causing turn-up."*

The work reported by Clarke *et al.* (*op. cit.*) highlights how operators develop strategies that compensate for flaws in the automation. The latter may be fit-for-purpose under normal work settings but adaptations are required for any other case. In this example, the adaptations performed by the operators prevent the occurrence of undesired outcomes. If such adaptations did not take place, the production would (at best) take much longer due to the necessary corrections to malformed slabs. Another interesting point is the anticipatory skills exhibited by the roughing mill operators. They often pro-actively adjust the quality of the slabs they produce to meet the requirements of the next processing stage (finishing mill). This is a sign of expertise in piloting the system and constitutes another example of human skills compensating for the inadequacy of the automated function.

Another example of workarounds is given by Voss *et al*. (2002), based on observations at a semi-automated engine assembly factory. This factory was equipped with a material storage tower that fed the assembly line. For some reason, this tower went off-line without any error message being fed back to the control system. The operators then found themselves unable to complete the assembly of the engines, and the control operators were unaware of the situation. The assembly operators then decided to mark all the material stored in the tower as faulty so that an order for new material could be sent

to the supplier. This behaviour clearly marks the existence of a strategy that allows work to continue despite an anomaly. Marking unavailable parts as faulty is probably not recommended by the plant's procedures but it nevertheless serves the company in that production is not slowed down too dramatically.

This steelworks and engine assembly factory examples show how an acceptable level of system performance is achieved through *ad hoc* adjustments performed on an imperfect piece of technology. Humans often find a way to reach a pursued objective (e.g. production goals), even by means of illegitimate actions. That said, the extent of the illegitimacy can vary. The examples above are rather shallow in this respect. However, there are cases where genuine creativity is required, for instance in the case of mishaps that were never experienced before. This is the topic covered in the next section.

3.2.3. Compensation for system limitations by humans: positive violations

United Airlines flight 232, bound for Denver, crash-landed at Sioux City Airport, Iowa, on July 19, 1989[49]. One hundred and twelve people were killed and 184 survived. The aircraft was forced to land after a metallurgical defect in the fan disc of the tail-mounted engine (#2) caused its catastrophic disintegration. The severity of this failure was such that the engine's accessory drive system was destroyed and 70 pieces of shrapnel damaged the lines of the #1 and #3 engines (see Figure 29), resulting in a complete loss of hydraulic control (as described in Besnard & Greathead, 2003). At the time of the accident, the loss of all three, independent hydraulic systems was considered a billion to one chance.

The damage to the hydraulic lines resulted in the crew having no control over ailerons, rudder, elevators[50], flaps, slats, spoilers[51], or steering and braking of the wheels. The only control which the crew had was over the throttles of the two, wing-mounted engines.

49 Unless otherwise stated, the material in this section is from NTSB (1990) and from Captain Haynes, pilot on the United Airlines flight 232 (Haynes, 1991)

50 A flap located on the trailing edge of the horizontal stabiliser.

51 A flap extending from the upper side of a wing that acts like an air brake.

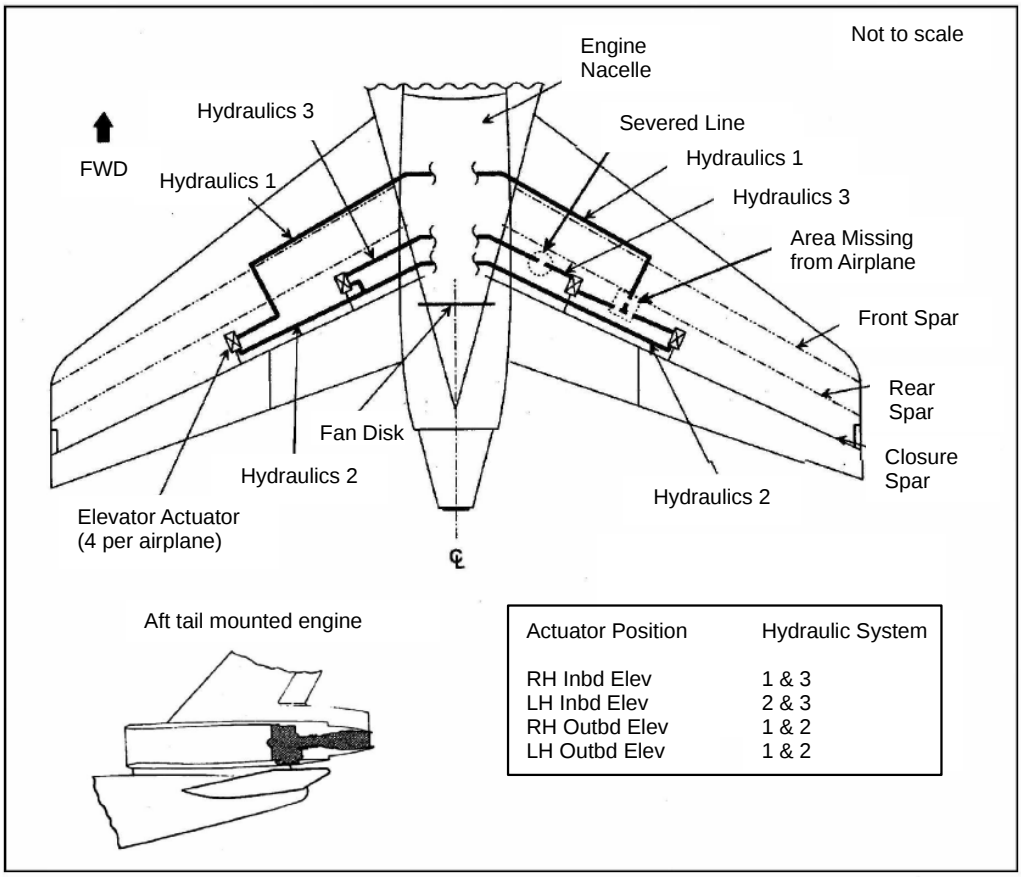


Figure 29: Location of the damage in the vicinity of the tail-mounted engine on the DC-10. Adapted from NTSB report AAR-90-06, 1990. © NTSB

By varying these throttle controls differentially, they were able, to a certain extent, to control the aircraft. However, as revealed by the radar plot diagram (see Figure 30), the control over the vertical and horizontal axes were dramatically impaired. For instance, in order to correct a bank and stop the aircraft turning onto its back, they had to cut one throttle completely and increase the other. In addition to this problem, the crew also had to react to phugoids, i.e. oscillations in the vertical flight plane whereby the aircraft repeatedly climbs and dives in association with fluctuations of speed. This was brought about when cutting the power to turn the aircraft caused the speed to drop. In turn, this caused the nose to drop and the aircraft to dive. The crew had to attempt to control this oscillation throughout the 41 minutes between the engine failure and the crash-landing. They needed to cut the throttles when the aircraft was climbing and approaching a stall (as increasing the throttles would cause the nose to rise further still). The crew also had to increase the throttles when the aircraft began to dive (to increase the speed and bring the nose up). As both the pilot and the copilot were

struggling with the control column, they could not control the throttles. It is usually possible to control all three throttles with one hand. However, as the #2 engine had been destroyed, its throttle lever was locked and the remaining two levers, on either side of the jammed lever, had to be controlled with one hand each. Fortunately, another DC10 pilot was on board as a passenger and was brought to the cockpit to assist. This second pilot could then control the throttles allowing the pilot and copilot to control the yoke and the copilot to maintain communication with the ground.

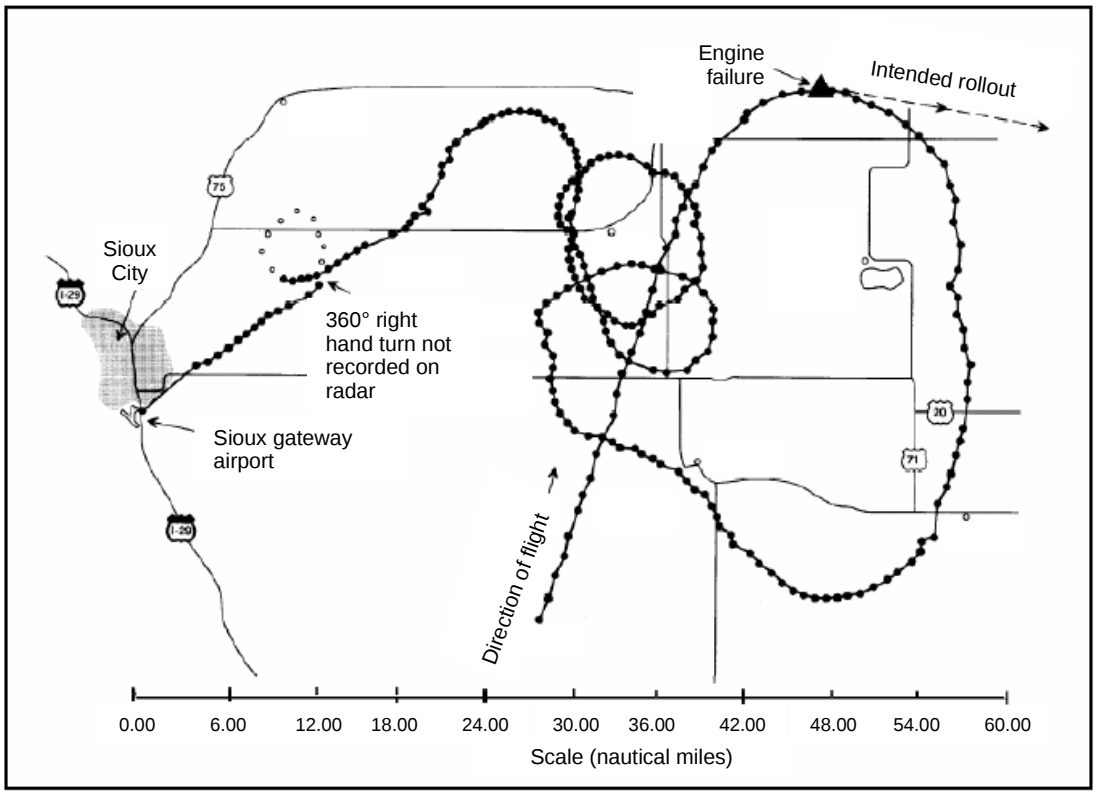


Figure 30: Radar plot diagram. Adapted from NTSB report AAR-90-06, 1990. © NTSB

This event exhibits the neutral nature of violations. These can be beneficial to system safety when they are coupled with a valid mental model. They allow operators to implement *ad hoc* control modes and to some extent, cope with unknown configurations.

Elsewhere in this document, I have written that mental models partially reflect the knowledge operators have of a system and are refined through the selection of environmental data. In this crash-landing case, the pilots used their knowledge of the aircraft's hardware to make the data displayed by the instruments converge towards a sensible representation of the situation. Revising a mental model is a crucial step in this kind of diagnosis-like activity

and it can be flawed even among expert operators. This has been experimentally demonstrated among mechanics and electronics operators (Besnard, 2000; Besnard & Cacitti, 2001) and has been the cause of other air crashes (e.g. METT, 1993). So it is fair to say that the pilots of the DC-10 achieved a very high level of performance. In comparison to the operators at the JCO nuclear fuel production plant (see section 3.3.2), the pilots developed a more anticipatory mode of control coupled with a more global and more functional view of the situation (Cellier *et al.*, 1997).

Another contributing factor in the relative success of this crash-landing probably relies on the mental model sharing that the pilots established. This component of distributed decision making (see Hollan, Hutchins & Kirsh, 2000) is a core activity in flight tasks (Doireau *et al.*, 1997). The transcripts of the dialogues inside the cockpit reveal at least two instances of such a distribution:

At 1552:34, the controller asked how steep a right turn the flight could make. The captain responded that they were trying to make a 30° bank. A cockpit crew member commented, "*I can't handle that steep of bank ...can't handle that steep of bank.*" (NTSB, 1990, p 22).

At 1559:58, the captain stated "close the throttles."

At 1600:01, the check airman stated "*nah I can't pull'em off or we'll lose it that's what's turnin' ya.*" (NTSB, *op. cit.*, p 23).

These two transcripts show that the pilots have a shared understanding of the situation. Each crew member interprets the statements of the captain with regard to the controls that one is acting on. The decisions are shared among the crew members and the mental model that is supporting the piloting activity is composed of the knowledge of several agents. Indeed, at this time, United Airlines were advocating a policy whereby flight crews were encouraged to share information and opinions and not merely obey the captain without question. Finally, contrary to the JCO operators, the pilots understood very accurately the consequences of their actions although they were under strong time pressure. In the last extract of the transcripts, 18 seconds before touchdown, the captain asked for the throttles to be closed. This is the normal practice for landing a plane and this statement was probably released as a side effect of a rule-based behaviour. Interestingly enough, the operator controlling the throttles rejected the statement, arguing that the throttles were steering the aircraft. This is an example of a safe violation supported by a valid mental model. By implementing an action

contrary to the usual procedure, one can nevertheless keep an already degraded system's state in reasonably safe boundaries.

The pilots' accurate mental model led them to define viable boundaries for possible actions and allowed them to restore some form of control on the trajectory under strong time pressure and high risks. Controlling the aircraft on the basis of such a model afforded the implementation of positive, desirable violations. These violations, although in direct contravention of procedures, were vital to the situation at hand. Also, it might be worth noting that the extremely low probability of the total loss of hydraulics[52] would probably rule out the option of creating a procedure for it. This unveils a wider issue that will not be touched upon in this thesis: that of the relation between procedures, the likelihood of the event they are supposed to cover, and how affordable the losses caused by this event might be.

3.3. Negative cooperation in socio-technical systems

Through examples of positive workarounds and violations, I have shown some conditions under which humans reacted positively to technical working configurations that were not foreseen by designers. These positive adaptations allow the socio-technical system's performance to be maintained (or have the effects of its degradation dampened) even in conditions that are not ideal. This is the positive side of human-system cooperation. However, many cases could be listed that illustrate the counterpart, i.e. situations where adverse conditions could not be compensated for. This is what this section will deal with.

More often then not, it is the technical components of the socio-technical system, or the constraints they impose on humans, that impair performance. In this view, front-line operators sometimes execute actions in degraded, unprotected conditions, thereby letting the natural variability of human performance slip into mishaps. I will review three cases of sub-standard performance of socio-technical systems caused by flawed human-machine interaction, negative workarounds and negative violations, respectively. In these cases, especially the last two, the point will be that often, rules and protections are perceived in terms of immediate costs and constraints that operators try to avoid. However, working around them (hence the term workaround used in this section) can create long-term, invisible threats which can mask degraded conditions.

52 Some sources quote the figure of a billion to one chance.

3.3.1. Degradation of system performance due to flawed HMI design

Humans are classically seen as being the last barrier before an accident. However, for them to fully play this role, they have to hold a mental representation of their task that is compatible with the situation at hand. When such a mental model is missing, operators only have a partial understanding of the world around them. An example of such a situation is prestidigitation[53]: the performer is intentionally manipulating objects at such speed and with open emphasis put on such carefully chosen gestures that the audience can only capture the salient aspects of the show. One then becomes unable to predict what the performer in going to produce in terms of effects, hence the surprise. Extended to human interaction, this example supports the idea that if some system states are not captured and understood by operators, then the capacity to correctly represent the system's behaviour is at threat. When this happens under adverse conditions (e.g. disturbances to the process) in safety-critical settings, the cooperation between humans and the system can degrade to such an extent that safety can be jeopardised.

A disastrous example is the friendly fire that happened to US army soldiers and allied forces in 2001 (Loeb, 2002). A U.S. Special Forces air controller was calculating the GPS coordinates of a target to a U.S. army aircraft for bombing. The air controller had recorded the correct value in the GPS device when the battery died. Upon replacing the battery, he sent the position the GPS unit was showing without realizing that the unit is set up to reset to its own position when the battery is replaced. The 2,000 pounds bomb landed on his position, killing three soldiers and injuring 20 others.

Another infamous example is Therac-25. This radiotherapy treatment device (see Figure 31) was involved in at least 5 deaths caused by radiation overdoses between 1985 and 1987. Accidents happened because of the conjugation of a number of causes, including programming mistakes and a flawed certification process. An exhaustive study of the Therac-25 accidents can be found in Leveson and Turner (1993) and in a subsequent updated version of the article[54].

53 Sleight of hand
54 http://sunnyday.mit.edu/papers/therac.pdf

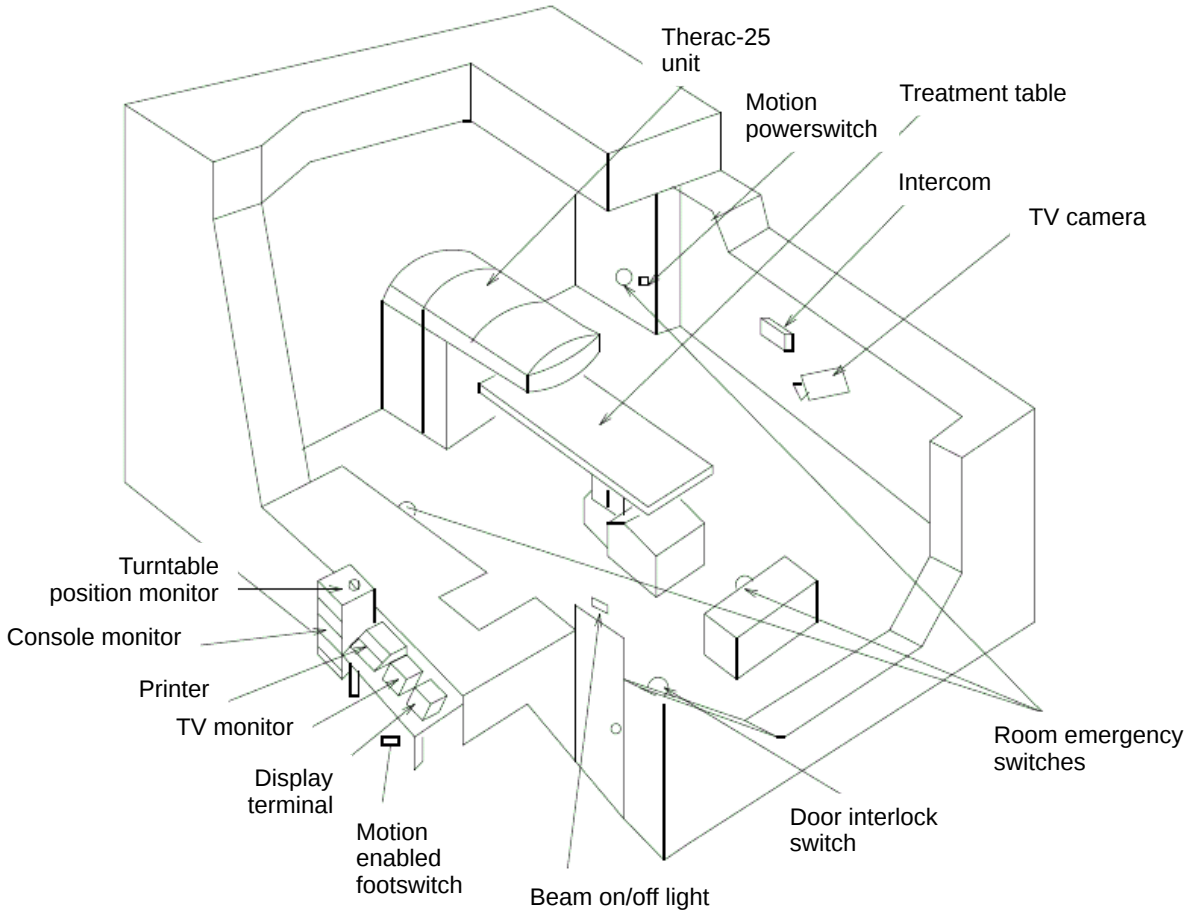


Figure 31: Graphical representation of a Therac25-equipped tumour treatment room (http://sunnyday.mit.edu/papers/therac.pdf)

In Therac-25, operators could operate the control interface faster than the system could handle, thereby triggering an error message such as MALFUNCTION 54 on the display terminal. This error message did not prescribe any type of corrective action, nor did it indicate the type of erroneous state the system had entered. The software manual did not provide any information either, apart from listing error messages. Because of the combination of frequent uninformative error messages and the possibility of miscalibrating the treatment beam due to design faults, operators could deliver lethal doses to patients. The Therac-25 accidents were caused by a complex interaction of several factors and it would be an abusive reduction to state that the HMI dimension accounts for all facets of the accidents by itself. However, the role of the uninformative error messages cannot be eliminated from the analysis, and it is this angle that was highlighted here.

In the Therac cases, medical operators were unable to compensate for the faulty design of the technical system. It is an important aspect in this case since the person at the control interface was the last barrier before an overdose was delivered. From this standpoint, the numerous design faults prevented humans from compensating for system shortcomings, thereby allowing the breakdown of the socio-technical system. This case showed how an interface-related flaw (more precisely: a sub-standard programming practice that impacted HMI) can have a detrimental effect on the performance of a system In this case, operators were interacting with an ill-designed system, through an interface that left them with no hint whatsoever as to what to do in response to the error message. Despite the extreme safety consequences of the flawed Therac 25 design, the same type of situation happens on a daily basis with more mundane systems such as office computers. Indeed, as Figure 32 shows, error messages sometimes neither provide information as to what the problem at hand is, nor what a solution to it might be.

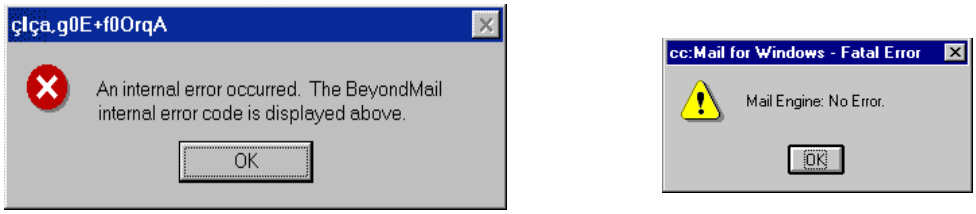


Figure 32: Examples of uninformative error messages (© Isys Information Architects Inc.)

Uninformative error messages, although still flagging that the system has entered some anomalous state, pose a crucial problem: they break the control loop by preventing operators from adjusting their actions (and understanding) to the system feedback. It is then virtually impossible to know whether the next action will be an adequate response to the current conditions or not.

3.3.2. Degradation of system performance by humans: negative workarounds

On December 30, 1999, in Tokaimura (Japan), a criticality accident occurred at the JCO[55] nuclear fuel processing plant, causing the death of two workers. The immediate cause of the accident was the pouring of approximately 15kg of uranium into a precipitation tank (see Figure 33), a procedure requiring

55 Japan Nuclear Fuel Conversion Co.

mass and volume control[56].

The workers' task was to process seven batches of uranium in order to produce a uranium solution. The tank required to process this solution is called a buffer column. Its dimensions were 17.5 cm in diameter and 2.2 m high, owing to criticality safe geometry. The inside of this tank was known to be difficult to clean. In addition, the bottom of the column was located only 10cm above the floor, causing the uranium solution to be difficult to collect. Thus, workers illegally opted for using another tank called a precipitation tank (see Figure 33). This tank was 50 cm in diameter, 70 cm in depth and located 1 metre above the floor. Moreover, it was equipped with a stir propeller making it easier to use for homogenising the uranium solution.

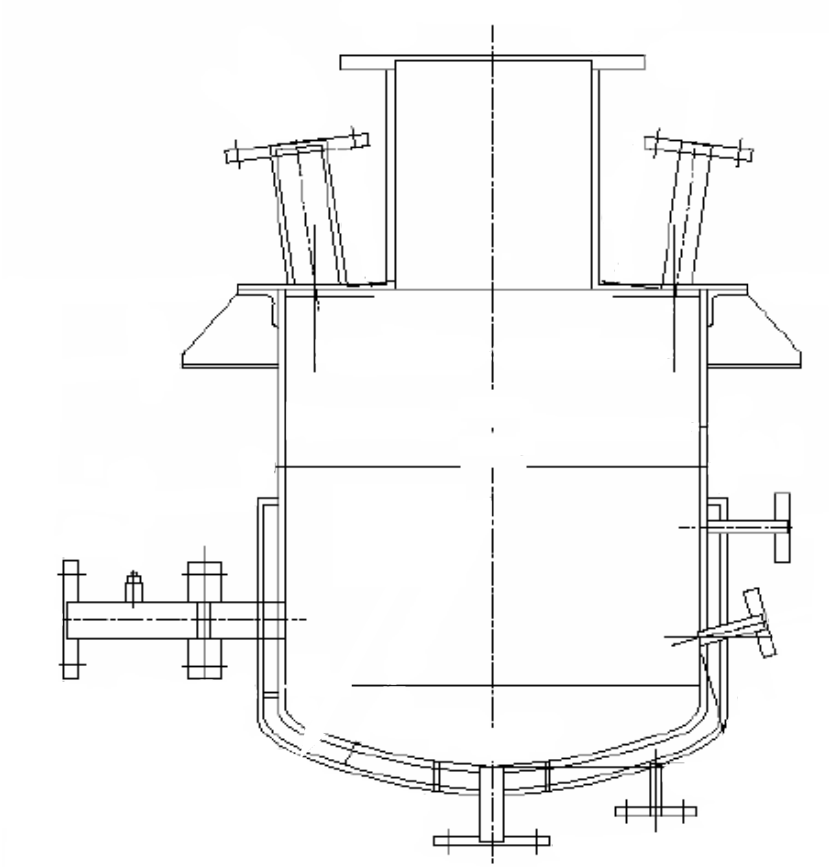


Figure 33: The precipitation tank at JCO (adapted from Furuta et al., 2000)

The workers thought it was not unsafe to pour the seven batches in the precipitation tank. However, there only is a limited amount of uranium that can be put together without initiating fission. When this critical mass is exceeded, a chain reaction occurs, generating potentially lethal radiations. This action contributed to the accident, which was rooted in a complex

56 Unless otherwise stated, the material in this section is from Furuta *et al.* (2000).

combination of deviant organisational practices. These included pressure from the managerial team to increase production without enough regard to safety implications and crew training. This policy impacted on the safety culture developed by the workers, providing them with excessive liberty, even for critical procedures. The crews' practices were embedded in a work context where routine violations were constantly approved, leading to the implementation of what Westrum (2000) calls a pathological safety culture. Ultimately, previous successful attempts at reducing the cycle time led to uncontrolled actions becoming the norm at JCO (Blackman *et al.*, 2000). These management issues are discussed extensively in Furuta *et al.* (2000).

The JCO criticality accident was caused by a management-enabled violation being coupled with the operators' erroneous processing of uranium batches above the critical mass (Besnard & Greathead, 2003). This coupling of a violation with an error has been identified by Reason (1990) as a very powerful generator of accidents. Although the causes of this accident, as they are rooted at the managerial level, call for an analysis at the system level (Bieder, 2000), Besnard and Greathead (2003) suggested a complementary individual cognitive approach highlighting the role of workarounds.

In cases of inappropriate use, precipitation tanks had already proven to be dangerous (Paxton *et al.*, 1959). In using this tank for producing so much of the uranium solution, the crews a) inaccurately assessed the situation, b) developed a flawed set of actions and c) ignored the consequences of such actions. These three components have been identified as important features in the control of dynamic systems (Sundström, 1993). In disregarding them, the crews have implemented what Hollnagel (1998) has termed *opportunistic control*. But it must also be acknowledged, after Wagenaar & Groeneweg (1987), that accidents are not necessarily caused by humans gambling and losing. They occur because people do not believe that the ongoing scenario is at all possible.

Also, one must highlight the role that conflicting objectives played in the occurrence of this accident. The management asked for an increase in production but the production unit was not designed to handle the higher level of production safely. This tension was not acknowledged, thereby leaving the operators under pressure to produce more without adequate safety conditions.

3.3.3. Degradation of general system performance: negative violations

Previously, this thesis depicted the case of of an airliner's crew who managed to crash-land their aircraft when it was bound to be totally lost after the

destruction of all hydraulic lines to control surfaces. A key element in this partial recovery was the correct understanding by the crew of the situation they were in, and the resulting successful improvisations they implemented. The other side of the coin (i.e. what happens when violations are performed with an imprecise understanding of the situation and the consequences of actions) is also very informative regarding the factors that determine degraded performance levels in socio-technical systems. This is what is attempted with the following account of the Chernobyl accident as published by the United Kingdom Atomic Energy Authority (Gitus, 1988, and analysed by Besnard, 1999).

On April 25, 1986, the operators of the Chernobyl nuclear power plant initiated a safety test plan. They wanted to know whether a generator spinning on inertia alone could supply electricity to the core water pumps while the emergency power supply units started. Until 13:05, power progressively diminished, in accordance with the test plan. At 23:10, power plummeted to 30 megawatts (MW) instead of 700MW due to the extraction of too many control rods from the core. On April 26, at 1:00am, power was stabilised around 200MW. The way this was done, owing to the particular design of the reactor, caused even more control rods to be extracted from the core in order to maintain fission at a minimum level[57]. By 01:22am, the core had become unstable. The repeated power drops had increased the water circulation within the core. This cooled an already underactive core and brought its activity down even more. At 01:24am, the required conditions for the accident were gathered: a steam explosion blew the steel core containment vessel. Three to four seconds later, a nitrogen explosion blew the roof off the concrete reactor housing. The highly radioactive core was then directly exposed to the atmosphere and burning graphite debris were ejected outside the housing.

In this accident, contributing factors were distributed among the reactor design, governmental policies, power plant management, test plan design and front-line operators. The design of the reactor was an important contributing factor to the accident since it potentially led to very unstable states. This probably contributed to the habit that operators had to manually disable safety devices. According to Gitus (*op. cit.*), the specific issue at Chernobyl was the combination of an unstable reactor with a lack of knowledge on the part of operators. This led the operators to work under very hazardous

57 The Chernobyl accidental nuclear reactor was of the RMBK type. This type of reactor is subject to power drops in at least two conditions. One is when the flow of coolant increases and slows down fission. The other is when Xenon (an unwanted by-product that kills fission) reaches a level where control rods have to be removed to maintain the power level.

conditions without even being able to acknowledge them. For Rey and Bousquet (1995), operators adopt a behaviour that is safe only with regard to how they perceive risk. This leads to the idea that there is not any direct link between risk and accident since hidden variables such as anticipation and knowledge can operate as mitigating factors. However, risk is strongly determined by the complex network of organisational factors. At Chernobyl, these factors had created latent conditions (see Reason, 1990; 1995; 1997) and operators had then taken risks in a situation where several unlikely events had combined *(USSR State committee on the utilization if atomic energy,* 1986). According to that report, operators had infringed so many rules that it is hard to think such behaviour was not within their habits. The account of the accident showed several occurrences of errors associated with violations, a combination known as aberrant behaviour which is a powerful generator of accidents (Reason, 1987). In the view defended in this thesis, a violation is a risk factor, which can be acknowledged provided one has some significant experience and knowledge about the system and the situation at hand. When this is not the case, the operator is unaware of the risk at play and harmful operating conditions leave room for failures that are no longer protected against.

3.4. Humans and the performance of socio-technical systems

What was discussed so far in this chapter is a series of socio-technical configurations where humans and technical systems interact in order to perform a task. Within this general background, I have addressed system's performance through positive and negative cooperation, respectively. The nature of this cooperation depends largely on the interaction between operators and the technical system:

- *Positive cooperation* depends on the technical system compensating for humans' limitations (e.g. through assistance function), or conversely on humans compensating for the technical system's limitations via workarounds and violations (e.g. the Sioux City emergency landing). These cases of positive cooperation, when due to human actions, rely on a mental model that shows some compatibility with the situation at hand, so that the outcome of illegitimate actions (that trigger unprotected states) can be anticipated and adapted to.
- *Negative cooperation* finds some of its causes in a technical system that prevents adaptive behaviours from humans (e.g. the Therac-25 accidents), or in cases where operators were unable to foresee the

consequences of illegitimate actions, thereby exposing themselves to harmful conditions (e.g. the JCO criticality accident).

In Figure 27 (p. 85), a distinction was made between the different types of variations and related dimensions. To recap, some of the properties of variations can be relevant to the analysis of the contribution of operators to systems' performance:

- *Intention*: The operator intends (or not) to carry out an action;
- *Legitimacy*. This action is supported (or not) by the system;
- *Anticipation*: The consequences of the actions carried out are unanticipated (or not);
- *Impact*. The impact of the action (beneficial; neutral; detrimental) on the system's performance varies depending on the above points.

From a safety point of view, my position is that legitimacy of an action is not a powerful predictor of the outcome of a variation. I hope this point was made clearly enough in this section.

In the rest of this section, I will consider actions from the violation angle, given that they show the greatest deviation from normal practice, thereby creating potential hazardous working conditions for workers. The reasons why people perform illegitimate actions will also be looked into. In doing so, the focus will be on the balance between the demands imposed by the task and the resources available to workers

This balance is the product of a trade-off whose purpose is to reconcile tensions at the workplace (e.g. unusable tank at JCO combined to the pressure to produce more). This balance sometimes takes the form of a violation when breaching procedures is the only way to get the work done. When violations are present at the workplace, they create unprotected conditions, thereby exposing workers to risk and lowering the chances of recovery in case of mistake. Sooner or later, an operator (e.g. because of a flawed mental model) fails to anticipate the consequences of their actions under these unprotected conditions. This is the general combination of factors that is laid out in Figure 34 as a summary of my understanding of the socio-technical mishaps described in this chapter.

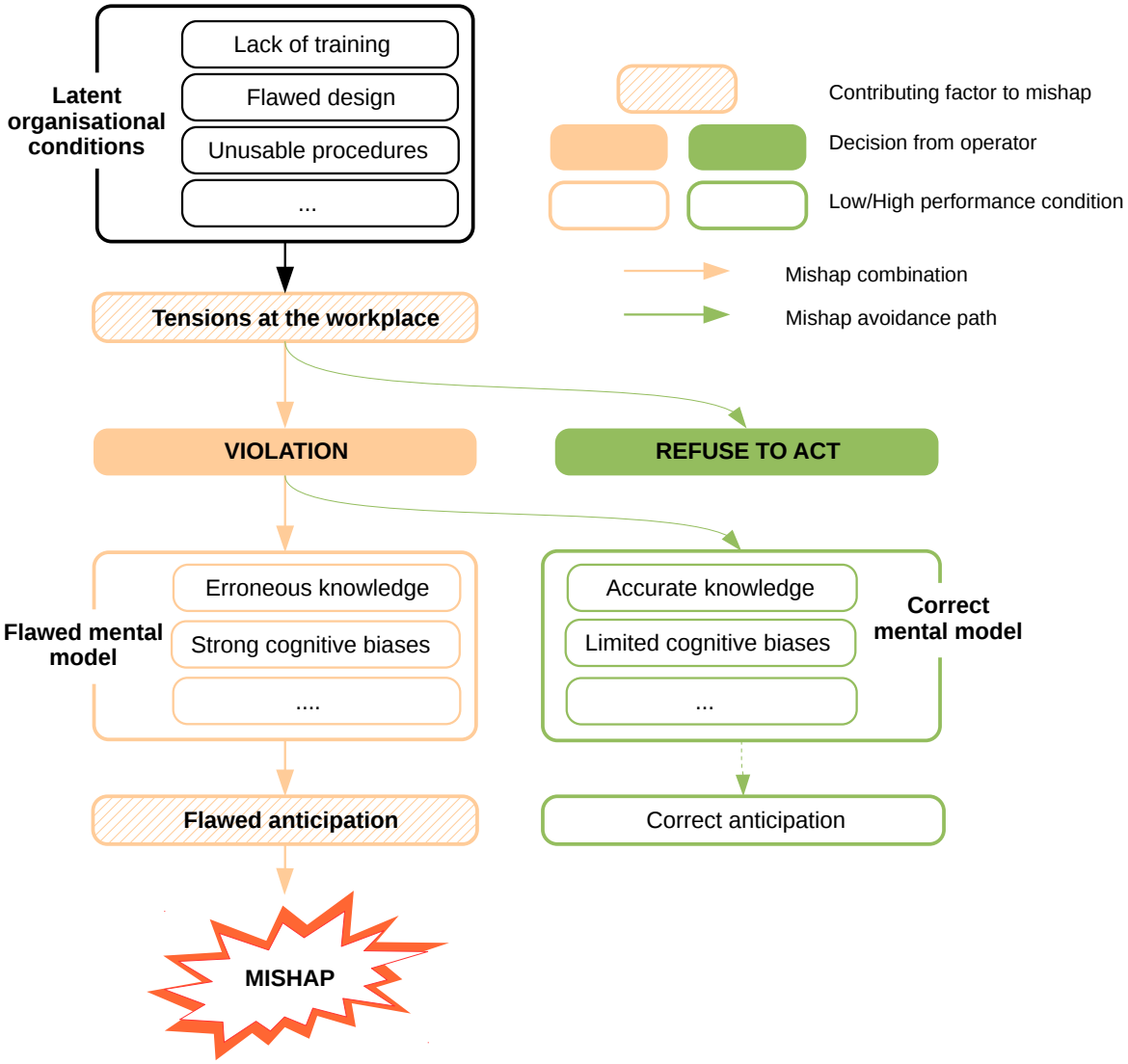


Figure 34: Some preconditions and consequences of violations

Also, along with Figure 34, a number of issues should also be highlighted:

- *A violation is a consequence before it is a cause*. It is the working conditions and the organisational settings that create the need for actions such as violations to be performed;
- *A violation is the symptom of a tension at the workplac*e. Violations are an attempt to perform a task despite non-supportive working conditions. In this view, violations answer a need and do not express a will to depart from procedures;
- *Violations alone do not create mishaps*. The latter implies the

combination of a violation and a flawed mental model. If operators hold a correct mental representation of their task, a violation is likely to have a positive effect on performance[58].

- *Violations are not domain-dependent*. The mechanisms depicted above can be found in virtually any socio-technical system. They reflect generic human adaptivity in performing one's job more than the features of a particular work domain.

3.4.1. Violations as an expression of trade-offs

After having reviewed some cases of workarounds and violating acts, it might be timely to reflect on the origin of violations and what type of situation they are a response to. First, it is an oversimplification to believe that humans at work always do as they are told. This is so for a very simple reason: it is impossible. If one takes the example of any production line under time pressure (in response to an unusual order, for instance, as was the case at JCO), one can easily foresee that managers might encourage their employees to produce as much as possible, while safety engineers will do their best to ensure that the work is carried out in acceptable safety conditions. This illustrates the tensions under which human operators typically work. In response, they seek intuitive production/safety trade-offs. However, the way in which these trade-offs are performed can lead to mishaps. This is caused by the difficulty of predicting all the consequences of one choice of action instead of another. In JCO, for instance, operators were trying to save time in order to produce more. They got blinded by the gain in using a different, larger, more usable vessel from usual (benefit seeking), and therefore rejected the procedure (cost avoidance) that prescribed the use of a smaller, less usable, safer vessel. This trade-off was carried out without a thorough assessment of the consequences of this choice.

As demonstrated by Vanderhaegen (2003) with the concepts of added modes and diverted modes[59], cost/benefit trade-offs happen everywhere constraints have to be reconciled. In production lines or in process control, they can take the form of workarounds, efficiency-thoroughness trade-offs (ETTO; Hollnagel, 2004) or boundary actions tolerated in use (BATU; Fadier *et al.*, 2003). Usually they are an attempt to reconcile contradicting tensions and deal with time pressure. They also often go unnoticed, if not implicitly

58 This statement excludes the case of sabotage which are deliberate disruptive or destructive actions.

59 These modes describe the ways operators can adjust procedures to prevailing constraints. The result is operating modes that were not anticipated by the designer (added mode), and operating modes that tweak what the designer had prescribed (diverted mode).

accepted, as long long as they do not trigger mishaps. Despite what was just said about dealing with time, and the link that was built between process control and trade-offs, the latter also belong to less dynamic (or process-related) activities. Indeed, they are a central mechanism in design activities (Bonnardel, 2004) such as product design, web design, or computer programming. It was demonstrated (in Besnard & Lawrie, 2002) that design is essentially a challenge of optimising a solution space according to a number of conflicting requirements. Reconciling constraints has a lot to do with utility functions, a concept that originates from economics. Utility functions can be be found in virtually any situation where a multi-criteria trade-off has to be made, be it buying a car, or running a chemical process. An interesting aspect of trade-offs by operators at the sharp end is that they can introduce threats in higher system functions. This is what was demonstrated in the domain of computer security (see Besnard & Arief, 2004)[60], and more precisely in the use of passwords. In this publication, we stated that when the increased intrinsic security of a computer system relies on enforcing longer passwords, users react by writing down these passwords, sometimes in the immediate vicinity of computers (e.g. under the mouse pad, on a piece of paper left nearby, on a note glued on the monitor itself, etc.). This behaviour originates in a conflict imposed by security on usability; a phenomenon identified by Sasse *et al. (2001)*. If one now analyses the above behaviour in terms of trade-offs, the cost of remembering longer passwords has reached (or gone beyond) a break point where the perceived usefulness of a higher security level does not offset the difficulty of use (see Figure 35). From the system point of view, such a violation from sharp end users introduces an interesting bias: it decreases the system's security level whereas the adopted artefact (a longer password) is intrinsically more difficult to break by intruders or illegitimate users.

60 See also Arief & Besnard (2003) and Besnard (2001) about intrusion and attack strategy issues.

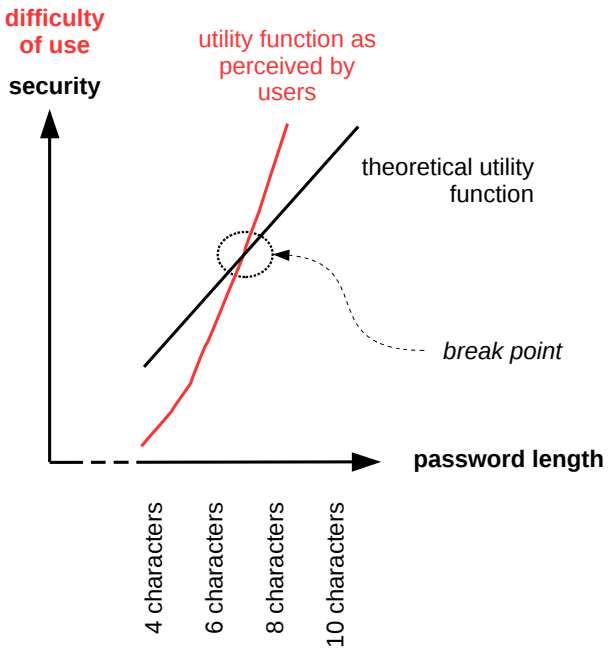


Figure 35: Graphical representation of a trade-off-induced bias in computer security

3.4.2. The role of workarounds and violations on systems' performance

There are many examples of workarounds combining with degraded modes of operation, ending up in mishaps (Johnson & Shea, 2007). Clearly, in itself, a workaround is not a guarantee of high performance. However, an important point here is acknowledging the role of the operators' mental model and the resulting understanding of the consequences of their actions. In the previous sections of this chapter, examples of positive and negative cooperation between humans and systems were described. From these examples, it is clear that violations alone explain very little of the impact of intentional variations on a system's performance. Instead, it is the degree of understanding of the consequences of one's actions that determines to the largest extent the outcome of a violation. It follows that it is only in the case of an incorrect understanding of a situation and of the consequences of actions that violations and workarounds are likely to trigger mishaps. This relation between a) one's degree of understanding of their actions and b) system performance is the point I once made (Besnard & Greathead, 2003), and that led to the distinction between negative and positive violations.

Although it is clear that not all adaptations are desirable, preventing humans from performing them is not the issue since there will always be a case that

will not be by covered by the procedures. Instead, the point is allowing workers to have enough training and understanding of the risks associated with their actions, so that the consequences of their adaptations can be anticipated (Fujita, 2000). This corresponds to Reason's (2000) view about high-reliability organisations: human compensations and adaptations to changing events and contingencies is one of the most important safeguards. From this perspective, violations can contribute to make a system safer. If operators have sufficient knowledge and available cognitive resources, they can implement an anticipatory mode of control which is a prerequisite for a safe interaction with dynamic real-time systems. In such conditions, human agents are able to conduct a safe *ad hoc* interaction in the case of e.g. emergency situations that were not expected by designers. Then, the flexibility of operators can maintain or improve safety by enlarging the span of the control they have over the system.

An aspect of things that has not been touched upon yet is the systemic dimensions of violations: they might open long-term threats that are difficult to see, and whose consequences are hard to predict. To start tackling this issue, let us remember that Reason (1990) identified two main categories of violations. Exceptional violations happen when an operator or a team is performing an action in a context identified as rather exceptional, not covered by procedures, and requiring some departure from the prescribed practice. Conversely, routine violations (called workarounds in this thesis) account for a regular practice, in the sense that an operator or a team regularly performs a task by means which differ from the procedures. In this case, the violation can be so deeply embedded into the daily practice that it is no longer acknowledged as an illegitimate act. Such elements as following the path of least effort, managerial *laissez-faire* and badly designed procedures are contributing factors.

These two types of violations have very different effects on the life of systems. Exceptional violations might be acknowledged as clear departures from procedures, in order to carry out an action on a one-off basis. Here, operators usually know that their potentially unsafe actions fall outside of procedures. Of course, these violations do not necessarily achieve the expected level of performance (see for instance the outcome of removing control rods from the reactor core at Chernobyl) but operators do pay particular attention to what they are doing. Mishaps related to exceptional violations are usually instantaneous and visible, despite showing potential for large-scale, adverse consequences. On the other hand, routine violations (workarounds) can be more difficult to detect given their progressive drift away from procedures

and their long-lasting, deep embedding into day-to-day practice. Typically, their long-term effects are not always known. In this sense, workarounds can contribute to erode the system's protections and can contribute to the emergence of latent accidental conditions (Reason, 1990). Eventually, latent conditions leave the system exposed to risk, but in a configuration that is not trivial to detect. From this point of view, mishaps related to routine violations (workarounds) belong to a different time scale than the ones related to exceptional violations, and result from virtually invisible threats introduced progressively into system barriers.

3.4.3. Violations and the blame culture

As Reason (1990; 2000) and many others have pointed out, the existence of violations is often caused by managerial flaws that propagate through the various layers of an organisation. As a consequence, a front-line operator causing an accident must not be regarded as a sole individual cognitive failure but as a wider system failure. Even if this is not the approach adopted here, it is acknowledged that operators are too often blamed for having performed actions that a flawed cultural context or a bad management policy made inevitable. The picture may be even worse. According to Van der Schaaf (2000), rules in organisations are often developed simply to protect management from legal actions. Such alarming issues have already been raised by Rame (1995) who asserts that some incidents even lead to data obfuscation when human factors are involved. The legal side of enquiries and the individual blame policy that still prevail in western European society can be questioned as well, especially when they clearly disregard non-individual factors leading to accidents (see for instance Svenson *et al.*, 1999).

An example of the above blame bias was broadcast on a French radio station in December 2007, following the crash of a Rafale air fighter. It was announced that the Army (among other parties) was going to conduct an inquiry in order to determine the various responsibilities involved in the crash. Whether or not this was the actual intention of the Army, the public announcement as such derived from the flawed philosophy according to which identifying responsibilities helps or is equal to establishing the causes of the accident.

Also, it is an intriguing fact that one seems to be more prepared to accept violations when they lead to a happy end rather than when they cause an accident. Instead, they should be seen as the two faces of the same coin. In the end, as Woods and Shattuck (2000) suggest, the design options range from centralised control inhibiting actors' adaptation to variability, to local

actors' complete autonomy disconnecting them from decisions from the hierarchy. Obviously, the final safety of a system will rely on the right balance between these two extreme points. As far as the actual design is concerned, Woods (1993, p. 23) suggests a two-fold view: *"The tool maker may exhibit intelligence in shaping the potential of the artefact relative to a field of practice. The practitioner may exhibit intelligence in tailoring his activity and the artefact to the contingencies of the field of activity given his goals"*. This highlights the dual view that one has to have about human agents in systems. Some people design tools, others use and reshape them so that they fit their intentions better. This reshaping activity by users has been identified by Wimmer *et al.* (1999) as a source of valuable data that design teams must try to capture. This matches Van der Schaaf's (1992) position (quoted by Rauterberg, 1995) where the idea is that whenever an unexpected configuration restores or enhances the performance of a system, then this positive contribution must be analysed to improve the functioning of the system.

3.5. Contribution to the field and future challenges

In this chapter, I have presented my understanding of the contribution of humans to the performance of socio-technical systems. This understanding revolves essentially around the notion of cooperation and adaptation to changing contexts. I have reviewed such concepts such as violations, cases of positive and negative cooperation between humans and the socio-technical system, and attempted to explain the nature of workarounds and violations. These issues as presented here rely on a number of pieces of work (Besnard & Baxter, 2003; Besnard, 2003; Besnard & Jones, 2004) where the issue of human adaptation to undependable systems was investigated. In these publications, it was argued that socio-technical systems often owe some of their performance level to the compensations that humans operate on otherwise unusable technical artefacts. This is an issue I investigated further (in Besnard & Greathead, 2003) by demonstrating the possibility that violations and workarounds do not imply a detrimental effect on socio-technical systems' performance. Instead, this paper demonstrated that the understanding of the consequences of one's actions is what matters. From a related, yet different angle, some colleagues and myself highlighted the importance of constraint resolution in socio-technical activities: people at work are constantly trying to reconcile tensions and this is sometimes a major contributor to mishaps (Besnard & Lawrie, 2002; Besnard & Arief, 2004). Finally, I should mention my PhD again (Besnard, 1999) and the analysis of

the Chernobyl accident I did then.

About the limitations of my work, I should first say that using the word violation, which I did repeatedly in my publications and in this very thesis, hides a judgement, an idea of wrongdoing. It is for this reason that the term unsafe act is used more and more often in the literature. This way, it becomes possible to express the idea that an action is departing from some established practice, without assigning an opinion to it. The same applies to such expressions as *positive* and *negative* when applied to human behaviour. One more reason to get away from these terms is that they tend to focus on the phenomenon as opposed to the cause. Indeed, when one tries to look at so-called violations from the point of view of a trade-off between a goal, constraints and resources, then capturing the adaptive aspect of human behaviour becomes more immediate. Finally, I should have emphasised the view of Ernst Mach (1905) more often. Indeed, following his view, only knowledge can distinguish between a correct and incorrect action. Therefore, I should have highlighted more clearly than I did the bias of assigning correctness or incorrectness to a behaviour when this judgement is made in hindsight. Sadly, this is commonplace in accident reports, whereas the reality of the sharp end operators (the ones who end up being blamed) is that they often do the best they can given the prevailing conditions (in terms of goals, constraints and resources).

I would like to begin the "challenge" part of this chapter with a little bit of reflection. Procedures prescribe a series of actions designed as an answer to already identified operational conditions. They can also evolve over time, on the basis of experience, changing conditions, user feedback, mishaps, etc. Despite my interest in accidents, I must acknowledge that many lives and systems are kept safe thanks to operators correctly applying (even underspecified) procedures[61]. However, socio-technical systems often comprise, and sometimes exhibit exceptions or unexpected emergency settings for which no procedure exists. When they occur, these exceptions impose such a narrow span of legitimate actions that departing from elementary rules is sometimes the only way to control the system. From the above, two messages can be recalled about variations in socio-technical systems:

- organisations should not expect humans to always act as prescribed and at the same time, expect contingencies to be accommodated;

61 I must thank Cliff Jones, from the University of Newcastle, for repeatedly highlighting this point to me.

- organisations should enable conditions where humans can still work safely when adapting procedures.

Procedures themselves do not dictate human behaviour (Fujita, 2000) and there are many ways in which humans can redefine their task and reconfigure their tools in unprotected modes. The motivation for doing so is based on cost (effort) avoidance and may be based on a heuristic evaluation. Indeed, if the intuitive cost/benefit (or safety/ease) trade-off analysis leads operators to foresee a more convenient way to do their job, then it is likely that a workaround will appear at the workplace. In this trade-off, factors such as safety culture and risk perception are key notions. And again, whether or not operators understand the potential effects of their actions determines, at least partially, the level of system performance.

In the next and last chapter of this document, after the series of descriptions of mishaps and successes discussed here, I will try to answer the following question : What sort of assistance can be provided to sharp-end operators so that the consequences of their adaptations (be it workarounds or violations) can be anticipated and supported?

Chapter 4. The Future. Assisting Human Cognition

What was attempted so far is a discussion of human reasoning in three different types of systems. It started with human cognition in static systems and looked at issues such as symptom interpretation and troubleshooting. I tried to demonstrate that finding meaning in a set of data depended upon a number of parameters and biases which make this activity prone to failure in certain conditions. Symptom interpretation and troubleshooting are also involved in the control of dynamic, critical systems. This was the link to the second big issue of this thesis: HMI. In analysing this, the point was to show that factors such as cognitive conflicts and mode confusion could heavily impact on the performance of the system. Then, a wider view on system performance finally led me to consider the adaptations that humans perform at work, how they can bypass rules in order to save the day or simply do their job.

Looking back, one should now see a particular landscape emerge from this thesis. Namely, if it were a hill, I would like it to be seen with three trees standing out sharply against the sky, each accounting for one particular issue:

- Human cognition in static systems is involved in troubleshooting operations and symptoms interpretation;
- Operators engage with dynamic, critical systems through human-machine interaction where interfaces play an important role;
- Workers routinely implement variations of procedures in socio-technical systems in order to achieve their objectives.

Up to this point within the thesis, several cases were presented where humans exhibited a particularly high level of performance, as well as failure. When comparing the two, it is difficult to get the balance right, because cases of failures, which potentially cause losses, are more likely to be documented and studied. The same is not true of success cases for the reverse reason: they do not cause losses and therefore do not trigger the same amount of

attention. The balance between failures and successes is important. Indeed, one of the conclusions of this thesis is that successes and failures are the two sides of the same coin, two different outputs from the same cognitive processes. It is a point that needs to be stated clearly: there is only one cognition operating in different conditions which, in turn, influence performance.

Scientifically speaking, understanding failures and successes is a first step towards understanding performance itself. But if failure can teach one the path to sub-standard performance, by highlighting the mechanisms that degrade performance, it does not say how a higher level of performance can be achieved. One approach towards this objective is to address performance conditions (Hollnagel, 1998) and control (or eliminate) the factors that degrade performance. Another approach, which is the one that will be followed here, is to look into the technical means available to improve performance by supporting cognitive activity, in particular when controlling processes under time constraints.

4.1. Where to, now?

Because it is now time to carry out some reflection on, and synthesise the material introduced in this thesis, I would like to raise the question of the commonality between the activities carried out within three types of systems analysed in this thesis. These activities seem to be connected, but is there one dimension that makes them similar? Where do their features overlap? In my humble opinion, the answer is *anticipation.* In all three types of systems, operators try to anticipate the consequences of their actions. They need (and try) to understand what is going to happen next. This is true for all of the three system types analysed here, although for different reasons. In troubleshooting static systems, operators build expectations about their actions, against which they compare the actual system response and adjust troubleshooting plans. In controlling dynamic, critical systems, anticipation serves to prepare future actions in order to cope with complexity and time pressure, as well as manage resources. Last, when implementing variations of procedures in socio-technical systems, operators need to understand the likely outcome of their actions, especially when safety is at stake.

I would like make explicit the possibly unspoken assumption that when things go wrong in critical, dynamic systems, it is usually in the relation between humans, the technical environment and the prevailing conditions. In this relation, the anticipatory aspects of decision making contribute to

performance to a large extent. On the basis of this sole criterion, it would already make sense to allocate more research and design effort to anticipation support. However, there are other dimensions to anticipation that make it all the more important, especially when human activity is concerned:

- anticipation is paramount in virtually any human action, and designing some form of support for it has a very wide spectrum of applications;
- the level of integration of anticipation into dynamic, critical activities (e.g. aircraft piloting) can be vastly improved.

These two points highlight a paradox: anticipation is a key feature in process control (Boy, 2005) but little is done to support activity in this respect (i.e. enabling better anticipation of future events). This paradox motivates my interest in *proactive* assistant systems. These systems show some potential for improving human performance in dynamic control and supervision tasks, precisely because they can provide support for an activity that human operators do instinctively: looking ahead. With such a design objective, proactive assistant systems introduce the prospect of higher system safety levels. For these reasons, a series of assistant models and systems will be presented, that touch on the very issue of anticipation. These solutions are instances of joint cognitive systems (Hollnagel & Woods, 2005) in that two information processing, mutually supportive components (one technical, one human) are made to collaborate with each other as a single unit. Classically, this is done by allocating any function to the component that shows the highest degree of performance on that function. I touched on this issue (Besnard, 2006; see Figure 36), long after Fitts (1951) introduced it under the acronym of MABA-MABA[62].

62 Men Are Better At – Machines Are Better At

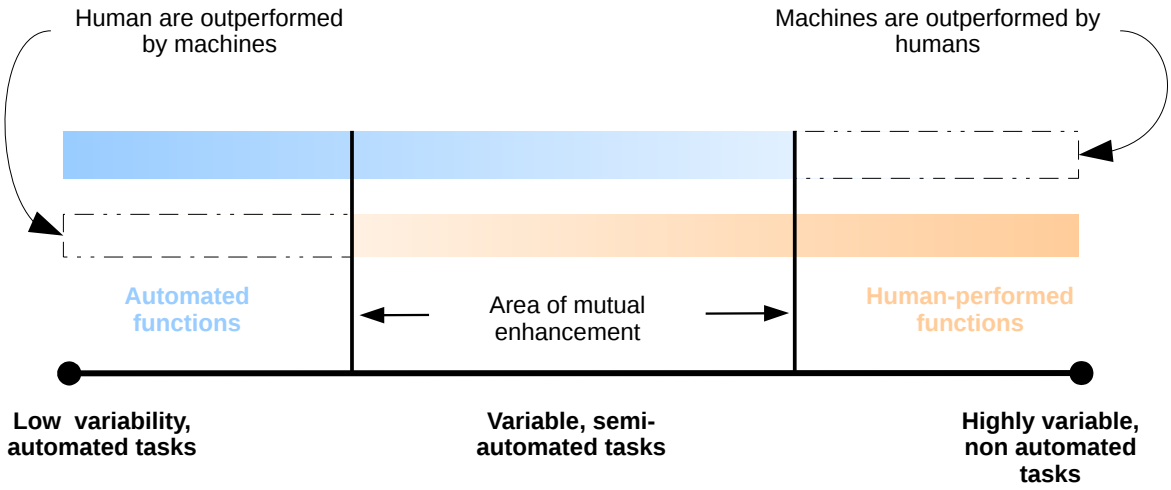


Figure 36: The overlap of human-automation cooperation (from Besnard, 2006)

The aim of what is called proactive assistants here, is not to merely allocate functions. Instead, it is to assist human decisions, primarily in the control of dynamic, critical systems. From an cognitive standpoint, the contribution of these assistants touches upon several issues identified within the three previous chapters:

- *Accounting for various levels of expertise and strategies*. The main reason for this is that assistants are expertise-independent. They do not rely on a specific level of experience but work on the basis of a comparison between some current state and a set of expectations. It follows that the knowledge held by operators, although playing its natural contribution to performance, does not determine the functioning of the decision-making assistants reviewed here.
- *Proactive support of the prediction of future states of dynamic, critical process control (e.g. aircraft piloting)*. The decision-making support assistants discussed here are essentially comparators based on artificial intelligence that try to find a match between actions from operators and a target system state. Given that this target state is generated by the assistant itself on the basis of a high-level model of the whole task (a flight plan for instance), the assistant is effectively "looking ahead" in order to make sure that timely support is given to action.
- *Anticipation of the consequences of one's actions in the case of*

variations of procedures. There are often several ways to perform a given task. This is true of altitude selection in aircraft piloting, for instance. The proactive assistants presented here do not exclude any of these possibilities. That said, variations are not supported specifically. Instead, they fall into the category above: the operator is not forced to take one particular action but assisted towards the reaching of a given system state.

These dimensions have been at the centre of a whole area of research: function allocation. Within this area, the question has been finding rules in order to (potentially dynamically) share roles within teams comprised of human and automated agents. Among the important dimensions that have been identified (see Sherry & Ritter, 2002, for a review), one touches upon automation being able to infer human and environment context and state. In this chapter, some time will be spent on this precise issue and review architectures of decision-support assistants.

I will begin with depicting the overall architecture of some of these assistants (see Figure 37) and summarising some of their main features and goals:

- they attempt to build an image of the operators' intentions from their actions;
- they monitor the system's states;
- they combine this data with a library of plans to build a set of expected future states;
- they suggest possible actions or warnings.

Proactive assistant systems integrate the global process control. They attempt to fulfil the humans' need to establish long term goals and foresee future states. As highlighted by the green dotted lines in Figure 37, assistants support anticipation by inferring possible events. Combined with the general process control and the global vision they provide to operators, assistants provide an opportunity for operators to build a representation of future states, and prepare plans for possible actions. The logical and automation aspects of assistants, as well as related considerations about artificial intelligence are not addressed in this thesis and can be found in Broderick (1997).

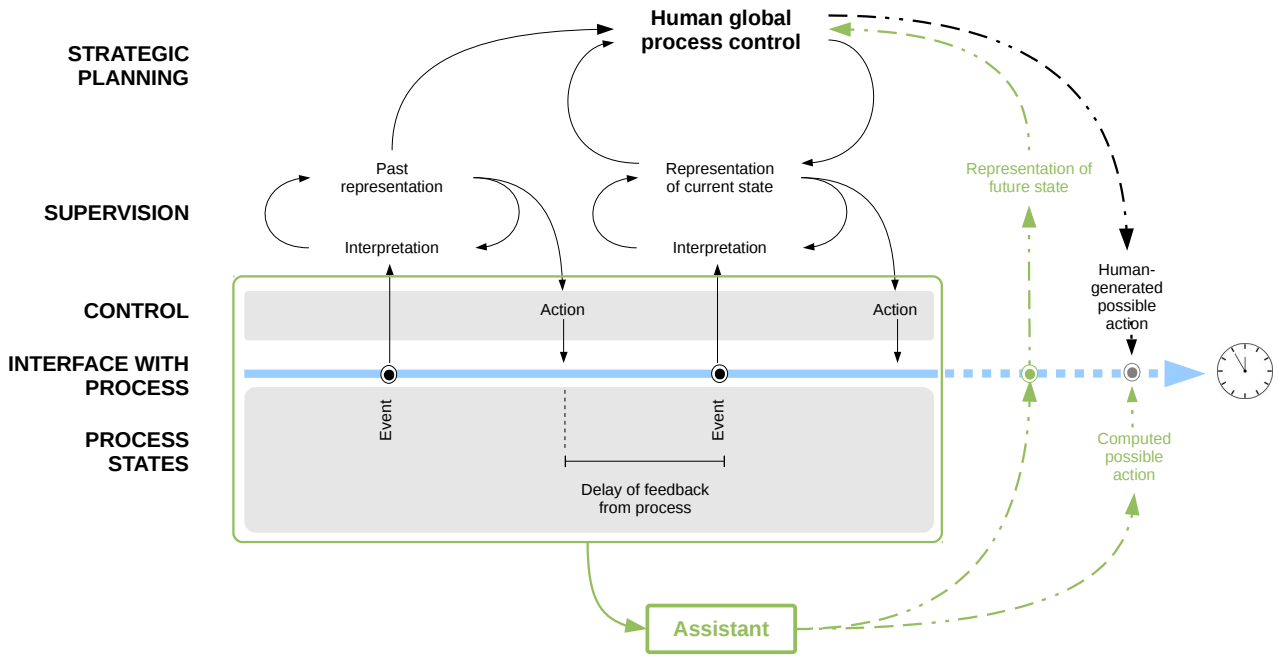


Figure 37: An architecture for proactive decision-making assistants in dynamic control and supervision tasks

4.2. Why are proactive assistants needed?

Cognitive ergonomics traditionally grounded and formulated design recommendations on the basis of such factors as information processing, workload, work conditions, interfaces, etc. This classical approach is what has underpinned socio-technical systems' performance so far. However, it no longer is sufficient given the rising complexity of the processes operators have to control, and the decreasing acceptability of human and financial losses. In a context where pure technical reliability is constantly rising[63], cognitive performance is the area where improvements might show the best prospects, and decision-making needs a new generation of assistant systems. In this respect, I believe that the level of cognitive performance in the control of industrial processes will show greatest improvements if assistant systems infer operators' intentions (by definition about the future) from their current

63 Today, some mission-critical pieces of software are designed with formal methods, a mathematics-based approach that guarantees the logical correctness of programs. Although very costly to deploy, formal methods can contribute to certify software systems at failure rates of less than 10^{-9} per demand.

actions, instead of solely reacting to operators' actions (which are at best contemporary to the observed system state).

Even in highly computerised control domains, humans still have to adapt to machines to some extent. Despite rich cognitive models and computing power being available, it seems that machines are too often seen (and designed) as the blind servants of human activities (sometimes, it is actually humans who serve machines due to e.g. poor design). Ideally, complex systems' control design should be turned by 180°. Machines should be designed so that they can dynamically adapt to human activities and enable collaboration. In dynamic systems control, this design philosophy would naturally lead to the creation of proactive assistants that would provide support to human cognition in order to stay “ahead” of the current system state. This precept was adopted by aeronautic ergonomics more than a decade ago. However, assistants still rely on reactive interfaces that a) transform the format of the data of the process into understandable information and b) flag errors on the basis on an inconsistency with the current system state. Global process vision and anticipation, so crucial to HMI reliability, still fall within the sole remit of the human operators. This state of facts has shown its dramatic limits in the cases of the Mont Sainte-Odile (see section 2.4.3), and the Cali crashes, (see section 2.3.2), where on-board systems were unable to assess the operators' flawed action plan, and assist in the recovery of the flight. The only sort of assistance that these operators ever received were alert messages at a time in the process where their occurrence was unexpected and their interpretation subject to extreme time constraints. In such conditions, recovery is jeopardised, thereby depriving humans from an important control mechanism. Also, and this is another limitation of reactive assistant systems, there is only limited time and opportunity to correct emergency situations when they are detected at a late stage.

In such a context, taking one's current actions into account for assistance is not enough. Inferring operators' intentions against the operational context and objective states is what is needed, and potentially offered by proactive assistants. This is the family of system that will be presented in the following sections and that will constitute a possible way to address the challenges identified in this thesis.

4.3. A short review of proactive assistants

4.3.1. Pilot's Associate

This assistant has been developed in the US Army (Banks & Lizza, 1991) and

was designed as a decision-support system for combat aircraft pilots. Pilot's Associate (PA) has undergone a series of versions since 1986 until the early 1990s, all built on evolutions of a number of common subsystems (see Figure 38).

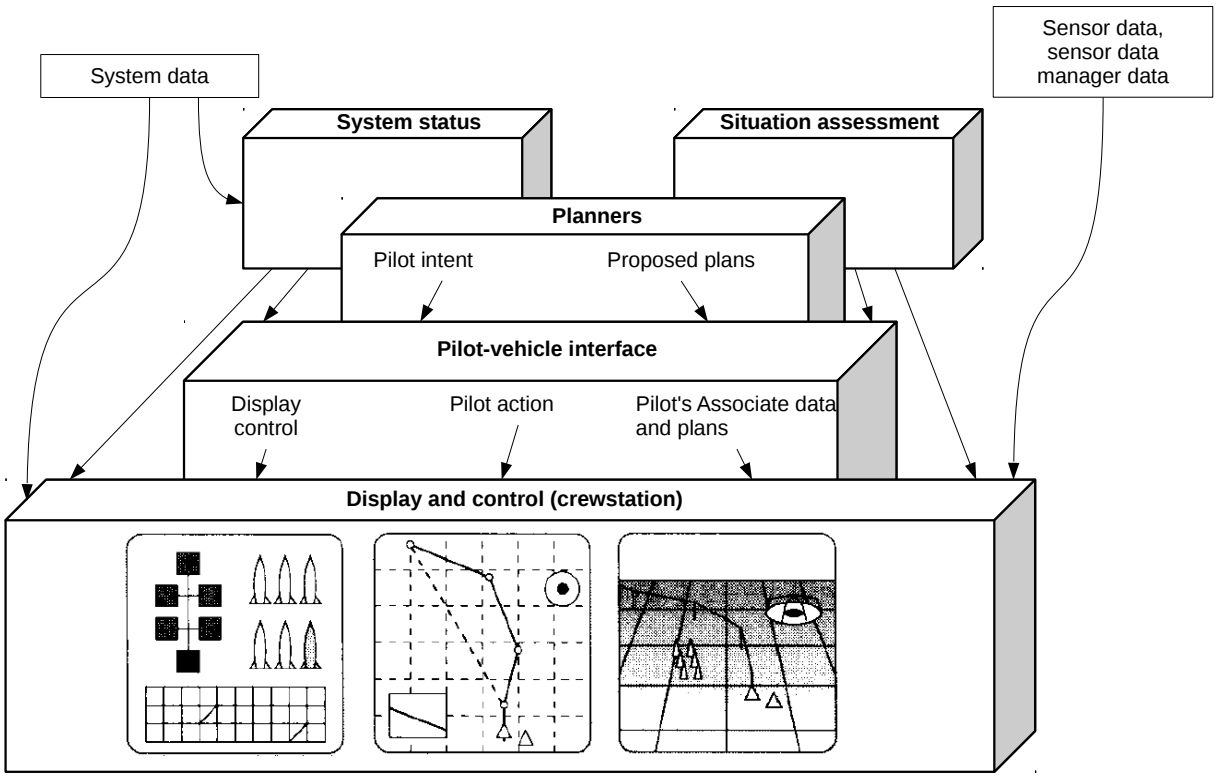


Figure 38: Dataflow in Pilot's Associate (adapted from Banks & Lizza, 1991. The repetition of "sensor data" in the rightmost upper box is reproduced from the original paper)

PA is a very complex assistant composed of several subsystems. From the standpoint of this thesis an important subsystem is the pilot-vehicle interface, which aims at reducing pilot overload through three different processes:

- intent inferencing (inferring pilot intent and communicating it to the other subsystems);
- display management (configuring displays and controls according to the pilot information requirements and communicating messages to the pilot);
- adaptive aiding (performing pre-approved tasks, detecting possible pilot failures, determining their consequences, and proposing remediation if necessary).

The Intent Inferencer (in charge of inferring pilot's intentions) takes inputs from aircraft flight systems and actions from the cockpit. Its outputs are

aircraft commands, pilot intent, and errors. In PA, errors are considered as inconsistencies that must be resolved. They are parsed through an Error Monitor, a pilot resource model and an Adaptive Aider. They are then explained and fed back as an action as input into the Intent Inferencer. A feature of the pilot-vehicle interface (implemented in the display management) is the capacity to select the form, contents and placement of information in the cockpit, on the basis of the selection of the most urgent or relevant plan to follow.

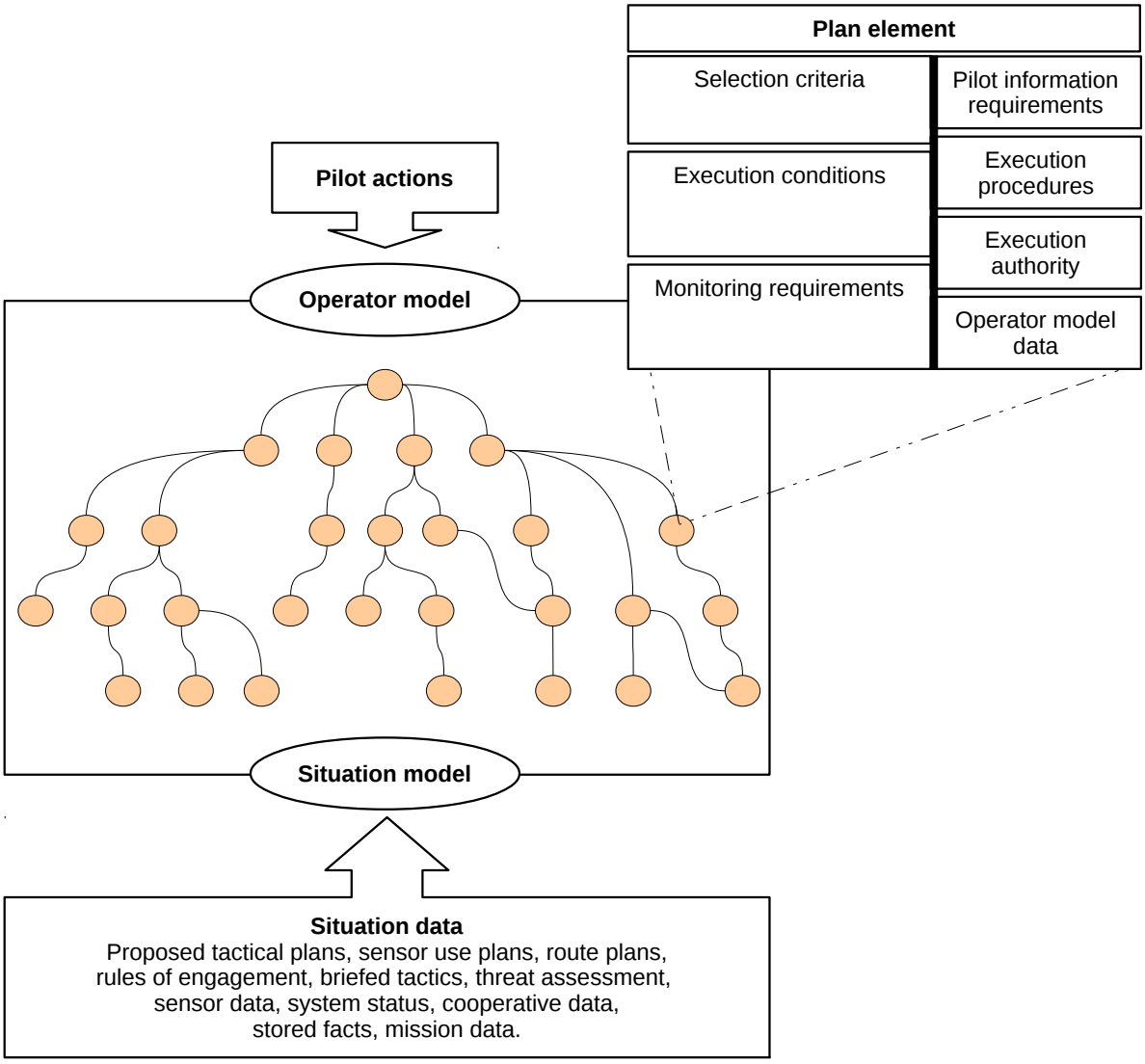


Figure 39: The plan-and-goal graph (adapted from Banks & Lizza, 1991)

From a general point of view, and in a similar way as most assistants, Pilot's Associate attempts to infer the pilot's intentions from his or her actions, combined to an operator model and a situation model. This is described in

Figure 39. The tree structure in the figure accounts for the plan that the system expects in terms of goals and sub-goals. The operator has to complete all the children goals for a parent goal to be understood as achieved. However, as in other assistants, several ways for the operator to achieve a parent goal exist and are tolerated. Also, each node is described according to the same properties (see upper rightmost table in Figure 39) for uniform interpretation by the various sub-systems.

The performance of Pilot's Associate is reported as being high, to the point where the assistant's behaviour is not noticed any more by the pilots. Banks and Lizza (*op. cit.)* report a high degree of acceptance by the pilots and the fact that PA never assumed authority nor negated the pilot's authority, a description that fits Sarter and Woods' (1997) definition of a team player.

Interpreting the pilot's intentions is a central feature in the model underpinning Pilot's Associate. This assistant is not aircraft-specific and can be applied (according to its authors) to other interactive, real-time processes such as helicopters, submarines and unmanned vehicles.

4.3.2. Hazard Monitor

Hazard Monitor (Bass *et al.*, 1997, 2004) is an assistant that provides decision-making support in the aviation domain. Figure 40 describes the overall architecture of the system.

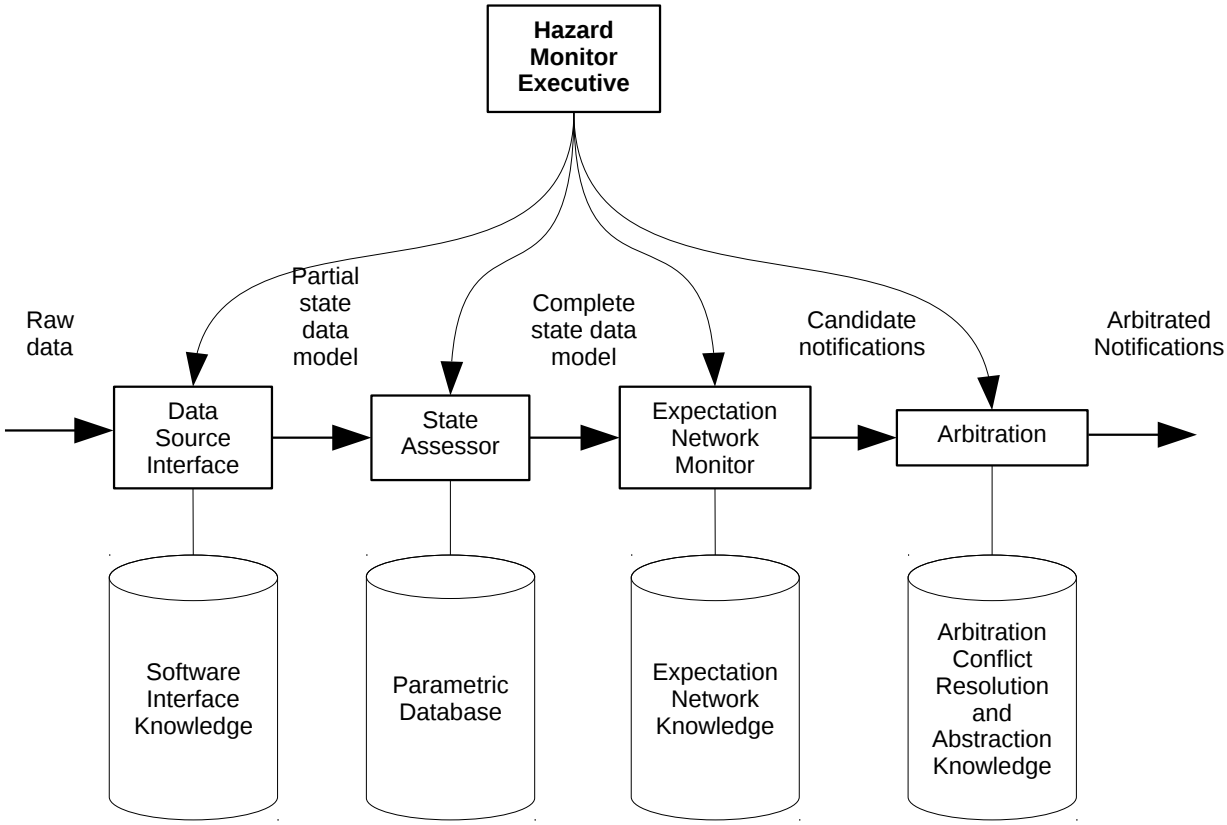


Figure 40: Hazard Monitor architecture

Hazard Monitor starts with gathering and parsing raw data in order to build a partial state model. This state is then fed into a state assessor that integrates this data into a higher level representation of the system and environmental state. On this basis, user interactions with flight systems are tracked and actions that can be treated as initiators of a typical sequence are detected. From this point on, the assistant activates a network of expected states (see Figure 41) and uses them as a normative behaviour set in order to perform plan recognition. On the basis of what the system state is assessed to be, Hazard Monitor (HM) can then trigger alerts about what the user should pay attention to. These alerts are prioritised according to four levels (information, advisory, caution and warning) related to the criticality of the situation at hand. Finally, an arbitration process filters duplicate notifications, especially when operators perform several activities at the same time.

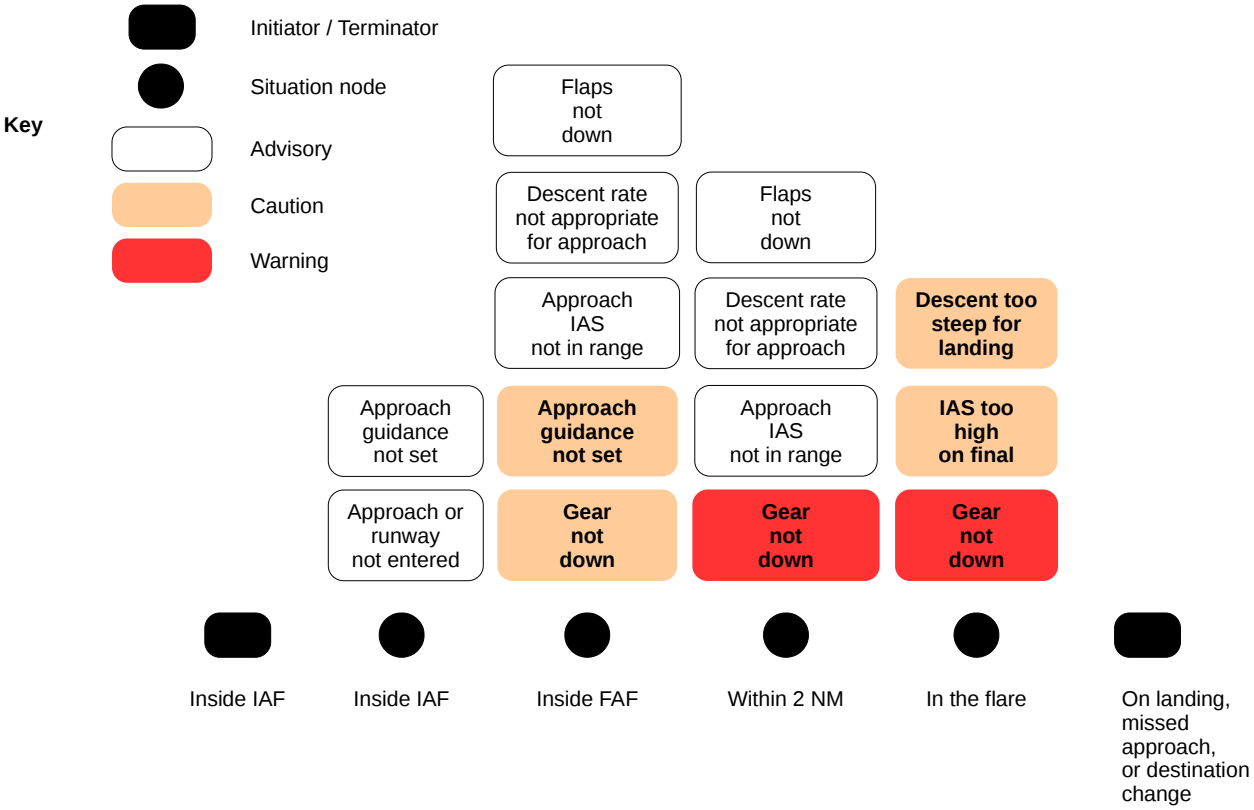


Figure 41: An example of a set of expectations in Hazard Monitor (adapted from Bass et al. 2004)

Even though HM is a sophisticated system, it still only deals with routine interaction. In other words, it is largely restricted to dealing with plans that would be used in normal circumstances. In the particular case of co-occurring events (see section 2.5.3 on the Kegworth accident, for example), or significant departures from procedures, it would not be able to offer much assistance to support decision-making or avoid potential problems. Nonetheless, HM is a concrete example of a decision support assistant that looks ahead of the current system state in order to provide operator support. Just like CATS (see section 4.3.4), although on the basis of a different philosophy, it attempts to support cognition in high-tempo systems by detecting inconsistencies between what the situation parameters are (e.g. phase of flight, descent rate, etc.) and what the configuration of the system (aircraft) is. These inconsistencies are then flagged to the operator for information or action.

4.3.3. CASSY

The cockpit assistant system (CASSY; Onken, 1997) is an assistant that focusses on providing support to a flight crew with the management of the

flight plan, related ATC instructions and possible re-routings. It relies on two basic requirements. The first is that the attention of the cockpit crew must be guided towards the most urgent task. The second requirement states that in case of excessive workload of the cockpit crew (despite requirement one), the situation has to be transformed by technical means into one that can be handled normally. Within the CASSY philosophy, the implementation counterpart of these statements implies that crew intentions have to be derived and corrected so that the situation awareness (as opposed to mere correctness of actions) is maintained at all times. CASSY identifies discrepancies as the difference between the normative model of appropriate pilot actions for the given flight plan, and the actual actions. CASSY then focuses on improving the pilot's situation awareness of flight plan executability and is responsible for autonomous activation of re-planning. Namely, if a discrepancy is detected, the crew is informed and the relevant re-planning processes are activated.

From a technical point of view, Onken (*op. cit*) highlights two features within CASSY that support decision-making: a) planning and decision-making assistance and b) monitoring.

The planning and decision-making assistance includes:

- autonomous or interactive generation and evaluation of routings or routing alternatives, and trajectory profiles for the complete flight or local portions of the flight;
- evaluation and selection of alternate airports and emergency fields;
- prediction of the remaining flight portions, when ATC[64] redirects the aircraft, or the pilot intentionally deviates from the plan.

The monitoring capabilities include:

- pilot actions with regard to nominal flight plan values, i.e. altitude, heading/track, vertical velocity, speeds and;
- configuration management, e.g. flaps, gear, spoiler and radio navigation settings;
- monitoring of violations of specific danger boundaries, including minimum safe altitudes, stall and maximum operating speeds and thrust limits.

64 Air Traffic Control

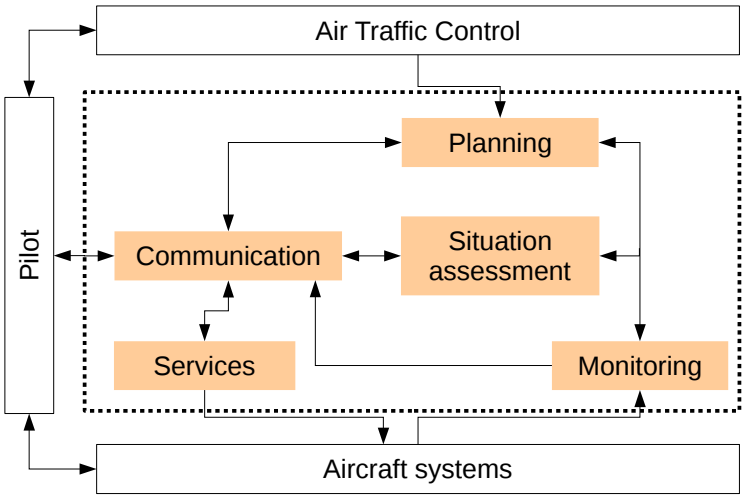


Figure 42: The main structure for CASSY (simplified from Onken, 1997)

In experimental trials run during simulated full flights (from taxi to taxi), CASSY demonstrated a high degree of acceptance from crews and a high rate of error detection and recovery. All pilot errors which occurred during the flight tests where detected. All moderate and severe errors, as well as about 70% of the light errors[65] were immediately corrected by the pilot after having received a warning or a hint from CASSY.

These results show a rather high level of collaboration between the technical system and the crew, and a high potential for real-time error detection and recovery. This latter point is of importance since dynamic systems can be high-paced and generate peaks of workload that can preclude error detection and recovery. This is especially true when a change of (flight) plan has to be made. In this respect, CASSY provides support for situations that are known to be difficult to handle by operators.

4.3.4. CATS

The Crew Activity Tracking System has been developed by Todd Callantine (2001, 2003) and aims to increase pilots' decision performance in real time by analysing a number of flight information sources. The underlying model was developed on the basis of a real data set from an experimental NASA B757. For a given flight plan, and from a corresponding library of expected actions, CATS performs intent inferencing. It activates a set of predictions about a pilot's activities against which it compares actual actions. However, there are often several ways for a pilot to perform a task (e.g. setting up an altitude

65 This terminology is the one used by Onken.

level at the master control panel). Therefore, instead of treating one of these possibilities as correct and treating the others as potential errors, CATS can delay its flagging of an error by monitoring similar or parent activities and interpreting their contribution to the ongoing task. CATS focuses on the way an action will trigger a state change, rather than the precise action that causes it.

CATS takes the prevailing flying parameters into account (e.g. autopilot status, autothrottle, aircraft configuration, etc.) and uses them as the context within which actions happen. From this context, the activity model builds predictions and interprets actions from the crew (Figure 43).

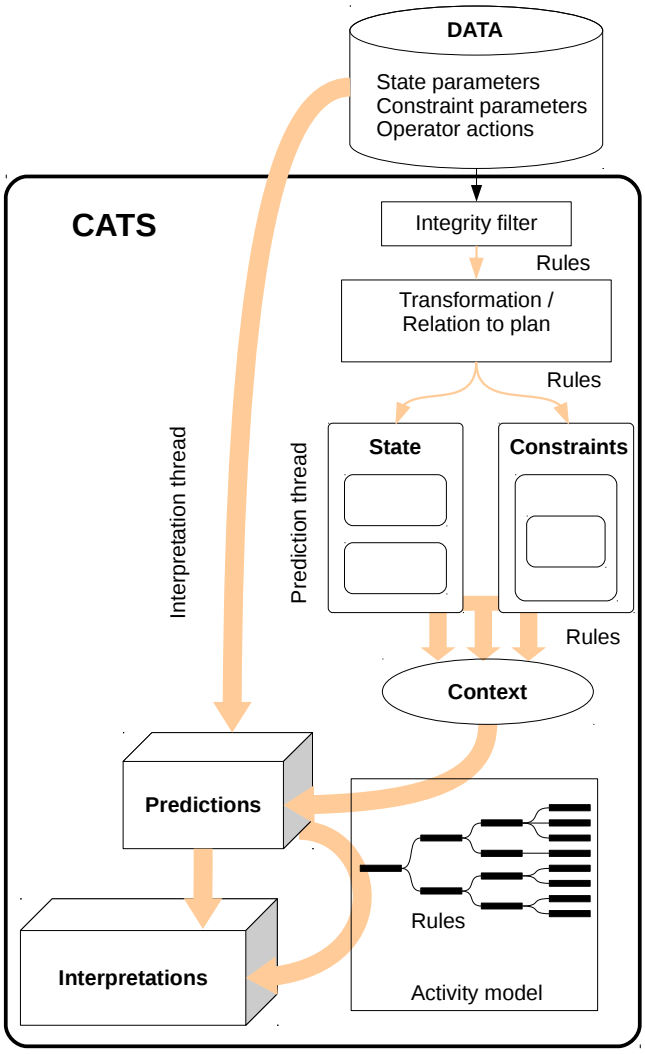


Figure 43: Information flow within CATS (adapted from from Callantine, 2003)

In one flight scenario discussed by Callantine, a crew received clearance from ATC to climb from 12,000 to 16,000 feet. The crew then selected the VNAV mode to prepare for the climb. However, the VNAV mode will not engage until a new altitude is set. Therefore, as soon as the VNAV button is pressed, CATS expects a new target altitude to be entered so that VNAV engages. The crew did not enter a value for this new altitude. From its activity model, CATS can spot failure of VNAV to engage and can flag an error (VNAV pressed but not engaged). Meanwhile, CATS still expects the crew to set a target altitude.

The Crew Activity Tracking System is an attempt to characterise a system's state with respect to its operational conditions and operator's input data. This philosophy assumes that operators' actions cannot be assessed as correct or not without assessing the system state in which they take place. This is a strong position and implies a thorough and yet flexible activity model. Indeed, the latter must be virtually exhaustive in order to capture as many possible states as possible. It must also be tolerant to variations of performance (e.g. in the sequence of actions) from operators.

4.3.5. GHOST

This system, designed by Dehais *et al.* (2003) aims at preventing the occurrence of undetected cognitive conflicts whereby operators (aircraft pilots in this instance) continue to follow a flight plan despite the presence of cues that indicate that doing so jeopardises safety. As already stated, this behaviour can have critical consequences. It is also resistant: information that should be used to reject the continuation of the flight plan is overlooked and perseveration loops constrain the pilot to following an erroneous plan.

To cope with this problem, Dehais *et al.* (*op. cit.*) tracked pilots' activity in a flight simulator in order to detect actions that would conflict with operational parameters (e.g. maintaining landing when visibility is below a certain level). The authors, via a bespoke interface integrated into the simulator, tested the effect of breaking the conflict by increasing the saliency of the pieces of information that would encourage the rejection of the flight plan. This increase in saliency is achieved by fading out an instrument display and restoring it, making it blink, or simply blanking it out. Once the saliency of a particular instrument has been increased, the pilot's attention is captured by that display. Messages are then sent inside the display area in order to highlight the current state of the flight and the need to reject the flight plan (the above cycle, from activity tracking to displaying the message in the cockpit is presented in Figure 44).

In one of their simulated scenarios, Dehais *et al. (*op. Cit.) had pilots fly to an

airport where fog made the (visual) approach imprecise, increased the time taken to find the landing point, and caused the aircraft to consume more fuel than initially planned. When GHOST was not used, pilots rejected the flight plan too late and the majority of them crashed. However, when GHOST was used, almost all the pilots noticed the alert message on the instrument panel informing them of the weather situation at their destination and flew back in time for landing at the departure airport.

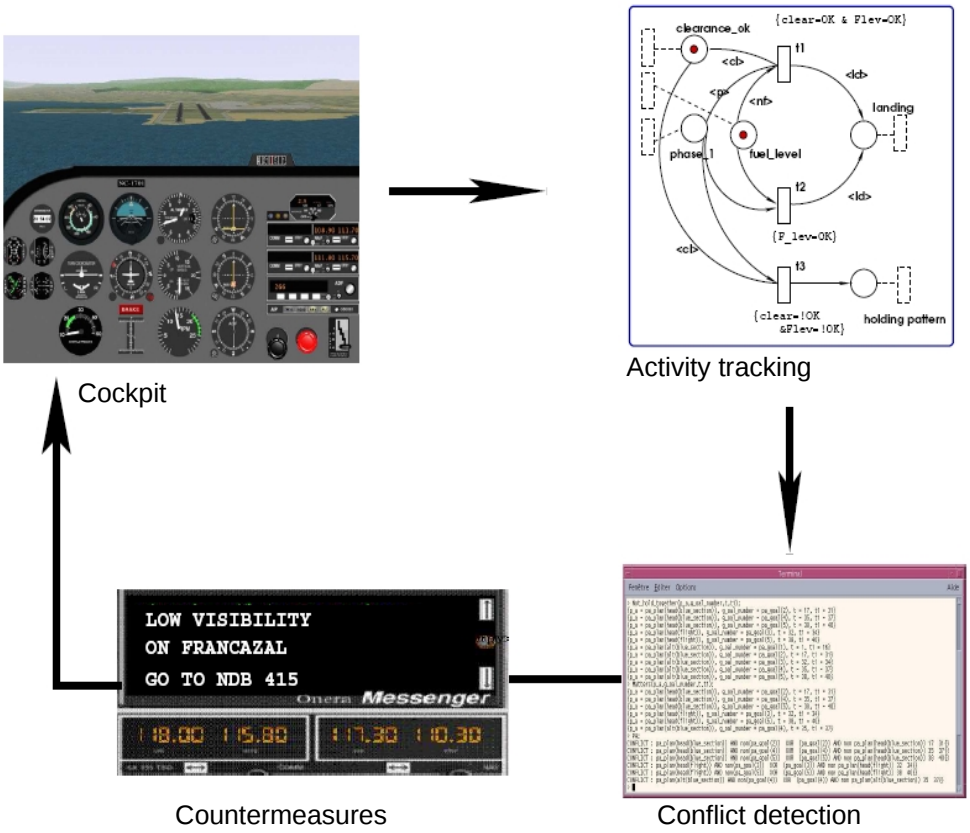


Figure 44: The GHOST countermeasures loop (adapted from Dehais et al., 2003)

GHOST is an example of integration of fine-grained knowledge about cognition into the interface of a dynamic, critical system. At the time of its publication, GHOST was a prototype that ran on, and was programmed for, a limited number of scenarios. One limitation with this assistant is that it does not address the issue of how hazards can be captured and by which mechanism they come to appear on an instrument display panel. Despite this limitation, Dehais *et al.* (*op. cit.*) showed that with reasonable technical means, assistants can demonstrate their potential contribution to human performance.

4.4. Discussion

Assistants aim at building expectations and supporting early decision-making. Virtually any operator action, assessed against its context, can change the expected future system state, and therefore trigger expectations for future needed actions. In this sense, the assistant builds a model of future system states to which operator's actions must show compatibility with, and contribute to. The projection of the potential match between operators' actions and forecast system states is an important feature since, as noted several times in this thesis, anticipation is a major dimension of human cognition. In this respect, the assistants reviewed here show some potential to contribute to human performance in the control of high-tempo systems. In this respect, some basic requirements can be listed and their rationale explained:

Requirement	Rationale
Provide information for decision-making	Because they must support decision, assistants must provide information for action as opposed to merely display more raw data.
Derive and support operator intentions	Proactive decision support has to infer what the operator will need next. This implies capturing intentions as opposed to merely observe and give feedback on contemporary data.
Gather and process data that support decisions about future states	Decision support must be targeted at future states. Allocating decision support to current states only, in dynamic systems, is a flawed philosophy.
Build sets of expected states	Such data as previous and current states, plans libraries, combined with previous and current human actions are the type of inputs needed for proactive support.
Filter the processed data according to the time band to which they apply	Decision support varies in the time horizon it is relevant for. Support should be time-specific (emergency, short-term, long-term).
Guide and support operators towards the most urgent actions	When an urgent action is needed, it should take precedence over longer-term deadlines and must not be missed. In such cases, the operator should be assisted specifically, with e.g. specific options to choose from.
Display decision-critical information in salient locations	Decisions that are mission-critical must not be omitted. Therefore, information related to such actions must be displayed in locations where missing them is unlikely, and in a format that supports action.
Account for variability in the choice and sequence of human actions	For virtually any automated system, there are several ways to carry out a procedure. Therefore, a sequence of actions that differs from a given plan should not always be treated as an error from the part of the operator.

For many years now, it has been established that automation must embed some awareness of the operators, by having some form of knowledge of human reasoning, as well as some data filtering functions (e.g. Boy, 1987; Rasmussen, 1991). This would allow machines to anticipate operator's decisions, provide more appropriate context-sensitive alarms and support for critical decisions. From the perspective of this chapter, expected benefits include the provision of some assistance in degrading situations, before matters become too critical. In this respect, the assistants discussed here demonstrate that there is a real concern from the scientific and engineering community for real-time, proactive assistance in decision-making for dynamic process control.

That said, the possibility of designing decision-support that is more computer-driven than it currently is does not necessarily imply that human operators are pushed out of the control loop. In fact, the reverse will be true as long as systems will show exceptions that designers have not captured. Indeed, it remains one of the ironies of automation that despite computers showing a continually growing capacity to control processes, the role of operators become more and more critical in order to intervene and fix problems when the automation fails (Bainbridge, 1987). It is the inherent human traits of flexibility and adaptability that supports this behaviour, and that automation designers have so much difficulty to automate.

As noted by Besnard and Greathead (2003), the goal of designing a human-machine system should be that of making the interaction between the operator and the system as smooth and efficient as the interaction between two people (Hollnagel & Woods, 1999). An essential part of human communication is that each participant should be able to continuously anticipate and modify the model held about the other. So following Amalberti (1992), machines should account for human operators' dependency on context. There may be enough knowledge in ergonomics and enough computational resources available in modern control systems to allow the implementation of screening functions dedicated to the analysis of human actions (as already suggested by Rasmussen, 1991). This kind of assistant system would dynamically and specifically support a given operators' reasoning, provide synthetic views of the system's state, anticipate which action is now required, which information will be needed next, etc. In this respect, some of the assistants presented here (e.g. GHOST) provide anticipatory protection measures for acts identified as hazardous.

This last point may be a way to foresee human-machine interaction flaws, such as mode confusion, by predicting and mitigating conflicts between the operator's actions and the system's state. Also, the main principle behind the assistants discussed above is that they can infer the operator's intentions from a combination of data from the history of the interaction, the operational state of the system, and reference plans. The assumption is that if the operator's intentions can be inferred, then context-specific monitoring and assistance can be provided by the automation. This approach is built on the assumption that in team operation, people try to understand each other and build joint expectations. This is an important feature of joint cognitive systems (Hollnagel & Woods, 2005) where the various human and technical components are considered as mutually supportive and collaborative.

Operators need more help in those situations for which they have not been

trained, than in normal settings. It implies that systems at large have to be designed in such a way that even unexpected events can be appropriately handled by support tools (through system state forecasting, for instance). Wageman (1998) argues that interfaces can typically flood operators with extra data at a time in the process (e.g. emergencies) where few resources are available. From my point of view, it is precisely because operators' intentions are not captured by automated systems that information overload occurs. This issue has been developed by Hollnagel (1987) who proposed the concept of intelligent decision support systems, and further addressed by Filgueiras (1999). However ambitious it may seem, this vision of the world is just one where classical ideas are stretched beyond current knowledge. According to Cacciabue (1991) and Hollnagel (1987), tools should fully support human decision making and improve system safety. The future reliability of the interaction between human agents and critical systems may depend on how far one succeeds in extending these principles, and how diverse the systems are in which one can turn these principles into a tangible design policy.

4.4.1. Limits

At this stage, I should make the assumptions behind my reasoning clear, especially those that bind the three main issues of this thesis (cognition in static, dynamic, and socio-technical systems) together, and link them to decision-support assistants. First, the latter can only be used when the task allows for some form of IT support. There are many work situations where this is not the case. Going back to the JCO case (see section 3.3.2) or typical workshop operations, it is clear that decision-support assistants might not be the way forward, owing to the limited use of IT in small-scale process control. Also, and this is one more limitation to my approach in this last chapter, decision-support assistants are essentially plan-based automata. A situation that requires human adaptivity to compensate for the lack of procedure (as in the Sioux city emergency landing; see section 3.2.3) is likely to be unsupported. This issue has been addressed more extensively elsewhere (Besnard & Baxter, 2006). Another limitation is related to assisting human decision-making when adapting procedures: future assistants need to be able to anticipate the consequences of an action that does not belong to a stored plan[66]. Indeed, beyond anticipating possible stored states, assisting human operators in dealing with unplanned states is a feature that is needed. This is missing from the assistants reviewed here and might imply a radical

66 A related aspect has to do with the issue of automation itself: only that which is known or foreseeable can be automated.

technological shift. If one thinks of the Kegworth crash (see section 2.5.3), it is worrying that the on-board systems never complained about the fact that the pilots were shutting down a perfectly working engine. In such conditions, human failure is difficult to avoid and recover from.

I have tried to demonstrate that research in intelligent decision-making assistant systems have the potential to enhance the reliability of human-machine systems by improving human-machine interaction in general. In essence, assistants need:

- to provide support to help operators make the right decisions at the right time;
- to act as warning systems when humans perform actions that are identified as having potentially detrimental effects.

However obvious the above might seem, it poses huge technical challenges. For instance (following Besnard, 2006) the notions of correctness and failure are difficult to capture since they are highly context-dependent. Indeed, there are situations where going against the procedure is needed. In such a case, judging correctness by its closeness to procedure is clearly counter-productive. Any assistant providing advice on such a case would have to be context-aware.

It could also be argued that agents based on inductive methods of reconstructing pilots' intentions can have flaws in the data selection process. Namely, the exhaustiveness of the data needed to characterise an operator's intention might represent a recurring challenge. Of course, these flaws may be caused by the current state of knowledge rather than by the principle of inferring mental models from interaction data. That said, it still poses a challenge for the time being.

4.4.2. Final thoughts

From a technical standpoint, an issue that opens interesting options, and that has not been touched upon yet is the configurability of assistants by operators themselves. Namely, some patterns of human-machine cooperation have been identified from the experimental study of dynamic function allocation in the context of assistant-based decision support in dynamic system control (Millot, 1990). From such studies, it appears that leaving the operators with the option of deciding which function to monitor themselves and which functions to allocate to the assistant, allows the assistant's behaviour to be tailored to the control strategy of the operator. In this respect, a broad dichotomy exists among operators with respect to the way they use function allocation: those who attempt to prioritise their workload before seeking process performance,

and those who prioritise system performance at the expense of a potentially high workload.

In highly computerised environments such as aviation, a number of on-board systems gather information in order to help the operator predict future states (e.g. ACAS or EGPWS ; see Boy *et al.,* 2007). However, pilots get plain information more often than a context-sensitive support for decision, and the information provided is not integrated to the pilots' intentions. Also, control interfaces tend to become more and more complex, and technical failures are becoming rare. I therefore see as unavoidable that future major mishaps in highly computerised domains (such as commercial aviation) will be partially caused by the unsupportive and opaque nature of interfaces. Namely, the lack of decision support assistants, combined with the growing complexity of systems might increase the likelihood of an entire category of accidents that are typically independent of technical failures: controlled flights into terrain The latter are a very hot topic in aviation and gather a lot of attention from aviation specialists, both from a managerial and cognitive point of view (FAA, 2003).

Page left blank intentionally

General Conclusion

In this thesis, I have attempted to show the thread that holds my research together and gives it coherence. What amazes me most at this stage is to see that almost everything I have done in terms of research was driven by a unique force: analysing the causes and effects of human performance. From setting up experiments with engines to looking into industrial accidents, I was pursuing the same objective. Another feature that I enjoyed was "looking back". Reading articles that I wrote almost ten years ago (for some of them) reminds me of how small a contribution a single article actually makes towards one's scientific career. Even in my case (a rather junior researcher), I see the distance that is ahead of me before I can pretend to understand the domain I'm in. But let me wrap up this thesis so that I can give the bird's eye view on this document.

Looking back...

By now, a number of factors of variability of human performance have been reviewed, in:

- static systems;
- dynamic critical systems;
- socio-technical systems.

If I had to tell the story of the preceding one hundred and something pages, a starting point could be to repeat the motto of this thesis: human performance variability is due to a number of factors that affect the system in which humans operate. The thesis started with the idea that humans give meaning to the world by constantly interpreting their environment. In this respect, troubleshooting is a particular situation. It is also one that is convenient to demonstrate that interpretation can be flawed by several factors. One of them is expertise, which can trigger irrelevant test plans, to the point where expert operators see their performance drop way below normal levels.

After a first chapter on cognition in static systems, the second chapter looked into dynamic, critical systems. It was demonstrated that many accidents could have some of their causes explained in terms of cognitive biases, combined with the dynamics and the typical complexity of these systems. Finally, human performance and its contribution to socio-technical systems were reviewed. I have attempted to depict cases of mutual compensation of performance of system components (humans and technical) and addressed the issue of variations of procedures.

Through this progression, the focus was set on one level of complexity at a time, and moved up progressively on the complexity scale. This is how I progressed from static systems, to dynamic, critical situations, and then finished with socio-technical systems. Each time, the objective was to highlight the role of some cognitive mechanisms in the performance of a task and the life of systems, allowing me to present some of my own published work in a logical manner. In doing so, three dimensions were addressed (in capital letters) that affect human performance, along with some of their cognitive aspects (in italics):

- INTERPRETATION. It is a universal activity among humans. Interpretation is about constructing meaning from the environment and presides over:
    - *Expertise*. This dimension usually correlates positively with performance. However, when a troubleshooting case is erroneously categorised as familiar, expertise can degrade performance and trigger irrelevant strategies.
    - *Heuristics* underpin many aspects of reasoning, in helping to achieve the best possible balance between cognitive cost and performance. Heuristics usually implies imperfect reasoning with an imperfect solution as a result, a situation that is seen as very common in ergonomics.

- ANTICIPATION is a key feature of dynamic process control. It aims at diminishing workload in forecasting (and planning for) future events.
    - *Cognitive conflicts*. They are the manifestation of a discrepancy between the state of a system and the mental representation of this state by the operator. Conflicts can go undetected for many hours. This can progressively degrade the interaction with the process under control and cause an unexpected mishap.
    - *Mode confusion* is a phenomenon that is usually caused by complex interfaces, where an operator interprets or handles

correct data in an incorrect (and undetected) mode. This is a very powerful generator of cognitive conflict. Indeed, operators will usually be confident in the correctness of their actions, thereby delaying recovery from a degrading situation.

- *Misinterpretation of system state* is a phenomenon where the loss of control over a process or degradation of performance is due to the misleading layout of the interface itself.

- VARIATIONS are a generic tailoring process associated with the life of socio-technical systems and help operators cope with the underspecification of procedures. These variations are an expression of the local understanding of a task, the specificity and variability of which cannot be fully captured by a procedure.
  - *Violations* are an intentional, significant, but not necessarily detrimental departure from a procedure. This behaviour can be the symptom of conflicting objectives, and operators can be left without protection against the consequences of failures.
  - *Workarounds* are an intentional but often minor departure from usual practice. They usually aim at easing the realisation of a task by slightly altering a course of action but without necessarily breaching procedures.

These dimensions can be arranged in a graphical way (see Figure 45), summarising the progression through chapters 1, 2 and 3. This figure echoes Figure 2 at the beginning of the thesis and provides the flesh on the skeleton that was presented then.

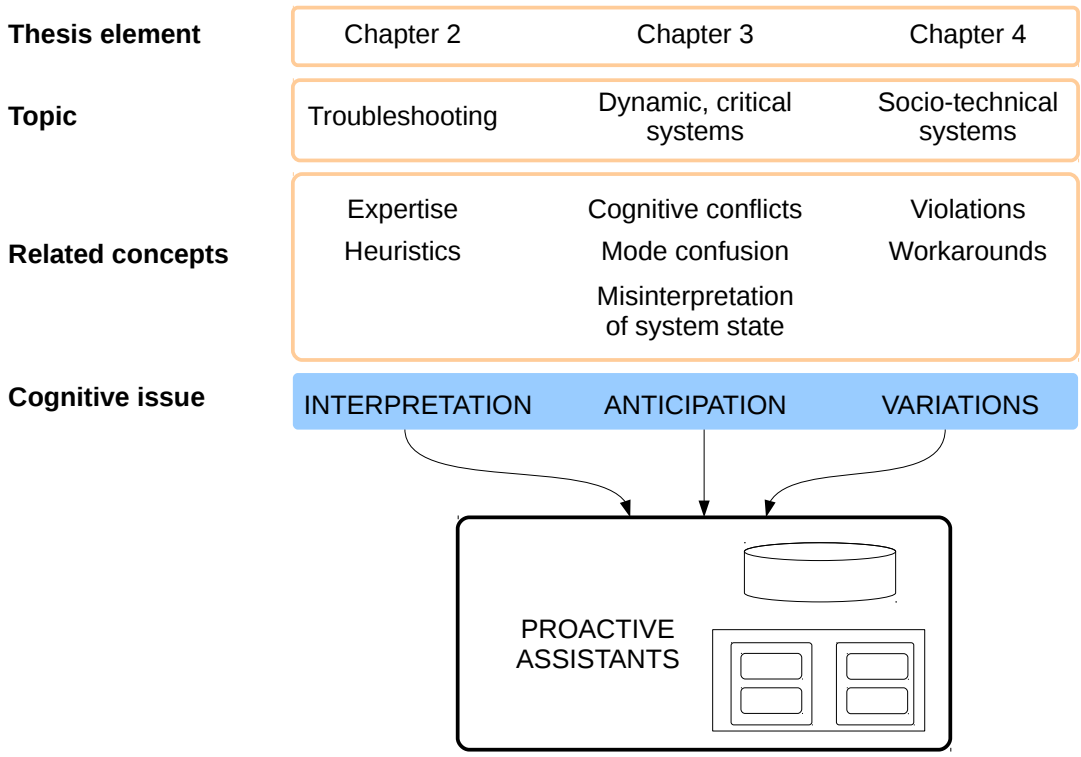


Figure 45: Graphical representation of the thesis structure

The above figure summarises the areas I have worked in, and published about. From this perspective the figure should not be taken as a representation of a strong theory. More modestly, it shows a graphical arrangement of where my work so far, helps in understanding human cognitive performance. In this respect, the concepts that are presented in this figure are essentially a map, even if I took position in favour of proactive assistants and their high potential for improving cognitive control. This closes the loop that was entered in the first chapter. There, the question of the concepts that could account for, or be related to human performance, was opened. Figure 45 summarises the answer, on the basis of my work. I hope that this answer, along with the previous chapters of reflection it relies on, can be taken as an acceptable one.

Hidden thoughts exposed

Before calling this thesis finished, I would like to make explicit some of the thoughts or assumptions that underpinned it. In a sense, it is easier to have them compiled at the end of this thesis since it saved me from raising side

issues in the middle of a chapter. Also, being of epistemological nature, these final thoughts are better suited for widening the reflection and put the technical contents exposed in this thesis into perspective.

My approach to human performance, as described in this work, has a rather strong cognitive flavour. As such, it can be perceived as contradicting today's focus on the context of human performance, as opposed to its underlying processes. Indeed, after decades of human performance analysis carried out on the basis of situation-independent human failure rates, the current approach of human reliability assessment focuses heavily on the conditions under which the work is carried out. This change of point of view was such a paradigm shift that the human reliability analysis tools following this pivot point (around the end of the 1990s) were called *second generation* methods. I'd like to think about this apparent contradiction in terms of complementarity. Indeed, my position is not one where context-independent cognitive mechanisms explain the whole of human performance by themselves. Instead, the local, individualistic factors of variability addressed in this thesis are strongly linked to the performance conditions on which second generation methods operate. For instance, consider a dimension such as HMI in Hollnagel's (1998) Common Performance Conditions. This includes the interface itself as a source of variation and takes it into account in performance prediction. This simple example shows how a local analysis of HMI-related cognitive mechanisms, which represent an entire section of my work as presented here, can truly feed second generation methods instead of being divorced from them.

At this last stage of my thesis, it might also be time to come back to some epistemological arguments laid out in the Introduction. There, I tried to comment on three hurdles to the identification of the factors of variability of human performance. These hurdles were:

- the lack of a precise model;
- the disturbances introduced by measurement of behaviour on the behaviour itself;
- the transformation of the task caused by the study of this task.

I commented on these hurdles as real barriers standing between us and a thorough scientific understanding of the factors influencing human performance. However, the first hurdle might be the most urgent to overcome. Indeed, the volume of literature that is available in cognitive

ergonomics is of such a magnitude that a meta-analysis would inevitably find a number of generic factors influencing performance. These factors would probably overlap with already existing typologies such as the Performance Shaping Factors (Swain & Guttman, 1983) or the Common Performance Conditions (Hollnagel, *op. cit.*). However, this would allow one to get closer to a predictive model of performance than we are now. As my own studies demonstrated, ergonomics allows one to study finely-grained mechanisms, thereby shedding some light on some rather local factors of human performance. What is missing is a global model that would finely predict human performance under a large number of conditions. I need to come back to my former example of weather forecasting to make my point clearly: prediction is more a matter of data and modelling than one of discipline. I therefore look forward to the day when an ergonomic model will be able to predict human performance as finely as weather scientists predict the weather.

An important topic in this thesis was human error. Erik Hollnagel was very influential in the way I treated it. In his view, and more and more so in mine, the concept of human error is fallacious since:

- it carries very little explanatory power;
- it is often taken as a cause of an unwanted event, instead of being seen as a consequence of a network of contributing factors;
- it is a judgement that introduces a normative idea of human performance;
- it leads to the absurd idea that the so-called errors are a good basis for establishing responsibilities;
- it invariably points towards sharp-end operators as the perpetrators of accidents.

As said in a previous section, following Hollnagel (1998), human sub-performance, unwanted acts, etc. are phenomena whose causes belong to the prevailing conditions. Treating an unwanted act as a cause *per se* is a gross oversimplification that leaves its causes unaddressed, thereby preventing organisational learning and weakening safety. Accident investigation sometimes falls in the trap of this oversimplification, although official bodies tend to include working conditions and organisational dimensions in their investigations. However, investigators are submitted to another bias: hindsight. This bias leads one to believe, once the causes established, that an event was easier to avoid than it was in reality. It therefore introduces the

false belief that this event will be easily avoidable in the future, potentially leading to shallow recommendations. Instead, Mach's (1905) visionary thought warns us that "Knowledge and error flow from the same mental sources, only success can tell one from the other."

I would now like to say a few words on safety, a topic that this thesis addressed but never really adopted as its centre of gravity. As scientists, our power is limited in this domain (safety), since it is a societal concern, even a choice, before it is a scientific issue. And it is an illusion to believe that safety comes first. What does come first is fulfilling the *raison d'être* of a system. For instance, a car will take you (at some level of safety) from one place to another but if you want to be absolutely safe on the road, you ought to stay away from it. From this perspective, except for safety systems themselves, safety does not really help in fulfilling the mission of a system. Instead, its nature is that of a second order requirement. It certainly is important for ethical reasons, for public acceptance of technology and services, for the long-term operation of some systems (especially for safety-critical businesses), etc. However, it remains a second order requirement. The first order requirement is getting the system to do what it is supposed to do (e.g. my car must take me to where I want; a refinery must produce refined products; a nuclear power plant must produce electricity, etc.). At the end of the day, human, material and financial losses are no more than a part of an equation where a large array of dimensions (safety, ethical issues, profitability, etc.) have to be traded-off against each other towards the objective of fulfilling the mission of the system.

In this thesis, not only did I not discuss safety explicitly, but I might have led one to believe that safety was a primary objective for ergonomics. It is not. Ergonomics does not deal with safety, it will not establish safety targets nor prescribe what an acceptable level of risk is. Instead, ergonomics can assist such people as safety analysts, commissioners, designers and regulators in taking into account some human dimensions in their work. This thesis is not a thesis on safety and was never intended to be. However, through the journey through the topic of human cognitive performance, some light was cast on some of these safety-critical human dimensions and how they can contribute to the performance of socio-technical systems. Let this be my modest contribution to safety.

Paving my way...

At the very end of this thesis, and after having summarised the broad cognitive dimensions addressed during my career, the time has come for me to look ahead and pick the stones with which I would like to pave the long stretch of scientific path that lies ahead of me. My immediate reaction is that I will continue to study human performance, but there is a little bit more to it.

At a fine grained level, I have an intellectual interest in proactive decision-making assistants. Being able to provide support to human behaviour is something that I see as absolutely mandatory in highly automated environments such as commercial aviation. My prediction in this domain is that the improvements in areas such as materials, aerodynamics, etc. have been of such a magnitude over the past decades that the future of commercial aviation accidents in the western world will be one where HMI mishaps will constitute the majority of the causes of accidents.

From a slightly broader perspective, the factors of human performance is a scientific topic where a lot of effort is needed by both science and industry. From breakthroughs in this domain, one will be better equipped for understanding the factors of past mishaps, and predict future performance. These two aspects are vitally important for the industry, for accident investigation and risk assessment, respectively. As a research associate for the Industrial Safety Chair at Mines-ParisTech, I can see both the scientific feasibility of following this path, and the interest for our industrial partners.

Finally, I see my wish to contribute to industrial safety as coherent with the above. It is an area I was contributing to rather remotely before I joined the Industrial Safety Chair. Today, with my scientific link to the industrial world being so strong and direct, I can see how much more research is needed into the human aspects of safety, how important a role ergonomists can play in this respect, and how much of an uptake one can expect from the industrial world. On this front, I see on a daily basis the growing interest of industry in the assessment of human reliability. This complementary angle to the classic assessment of the technical system reliability is becoming more and more of a requirement. This is true not only for safety as a philosophy, but also for the certification of highly critical systems. It is a good thing for the promotion of ergonomics and a wonderful opportunity to demonstrate the value of the discipline.

Page left blank intentionally

References

AAIB (Air Accidents Investigation Branch) (1990). *Report on the accident to Boeing 737-400- G-OBME near Kegworth, Leicestershire on 8 January 1989*. Retrieved on 11-07-2008 from http://www.aaib.dft.gov.uk/sites/aaib/publications/formal_reports/4_1990_g_obme.cfm

Aberg, L. & Rimmö P.-A. (1998). Dimensions of aberrant driver behaviour. *Ergonomics*, 41, 39-56.

Aeronautica Civil of the Republic of Colombia (1996). *Controlled flight into terrain American Airlines flight 965 Boeing 757-233, N651AA near Cali, Colombia, December 20, 1995 (Aircraft Accident Report)*. Retrieved on 11-07-2008 from http://sunnyday.mit.edu/accidents/calirep.html

Air France (1997). Anatomie d'un accident. F28-Dryden, Canada, Mars 1989. *Bulletin d'information sur la Sécurité des Vols*, 36, 2-7.

Allwood, C. M. & Björhag, C.-G. (1990). Novices' debugging when programming in Pascal. *International Journal of Man-Machine Studies*, 33, 707-724.

Allwood, C. M. & Björhag, C.-G. (1991). Training of Pascal novices' error handling ability. *Acta Psychologica*, 78, 137-150.

Amalberti, R. (1991). Modèles de raisonnement en ergonomie cognitive. In *Actes des Entretiens Science et Défense 91.* Paris, Dunod (pp. 317-328).

Amalberti, R. (1992). Safety and process control: An operator-centered point of view. *Reliability Engineering and System Safety*, 38, 99-108.

Amalberti, R., Bastien, C. & Richard, J. F. (1995). Les raisonnements orientés vers l'action. In *Cours de Psychologie. Processus et Applications*. Paris, Dunod (pp. 379-413).

Amalberti, R. (1996). *La conduite de systèmes à risques*. Paris, P.U.F.

Arief, B. & Besnard, D. (2003). Technical and human issues in computer-based security. *Technical report CS-TR-790*, Newcastle University.

Arocha, J. F. & Patel, V. (1995). Novice diagnostic in medicine: accounting for evidence. *The Journal of the Learning Sciences*, 4, 355-384.

Bainbridge, L. (1987). Ironies of automation. in J. Rasmussen, K. Duncan & J. Leplat (Eds). *Technology and human error*. Chichester, Wiley.

Bainbridge, L. (1993). Planning the training of a complex skill. *Le Travail Humain*, 56, 211-232.

Banks, B. B. & Lizza, C. S. (1991). Pilot's Associate. A cooperative, knowledge-based system application. *IEEE Intelligent Systems and their Applications*, 6, 18-29.

Bass, E. J., Small, R. L., & Ernst-Fortin, S. T. (1997). Knowledge requirements and architecture for an intelligent monitoring aid that facilitate incremental knowledge base development. In D. Potter, M. Matthews, & M. Ali (Eds). *Proceedings of the Tenth International Conference on Industrial & Engineering Applications of Artificial Intelligence & Expert Systems*. Amsterdam, The Netherlands, Gordon & Breach Science Publishers (pp. 63-68).

Bass, E., Ernst-Fortin, S., Small, R. L. & Hogans, James Jr (2004). Architecture and development environment of a knowledge-based monitor that facilitate incremental knowledge-based development. *IEEE Transactions on Systems, Man and Cybernetics-Part A: Systems and Humans*, 34, 441-449.

Baxter, G., Besnard, D & Riley, D. (2004). Cognitive mismatches: Will they ever be a thing of the past? Article presented at the *Flightdeck of the Future* conference, Nottingham, University of Nottingham.

Baxter, G., Besnard, D. & Riley, D. (2007). Cognitive mismatches in the cockpit. Will they ever be a thing of the past? *Applied Ergonomics*, 38, 417-423[67].

Ben-Shakhar, G., Bar-Hillel, M., Bilu, Y. & Shefler, G. (1998). Seek and ye shall find: Tests results are what you hypothesize they are. *Journal of Behavioral Decision Making*, 11, 235-249.

Bereiter, S. R. & Miller, S. M. (1989). A field-based study of troubleshooting in computer-controlled manufacturing system. *IEEE Transactions on Systems, Man and Cybernetics*, 19, 205-219.

Besnard, D. & Channouf, A. (1994). Perception infraliminaire de stimulus familiers et résolution de problèmes simples. *Anuario de Psicologia*, 62, 41-53.

Besnard, D. (1999). *Erreur humaine en diagnostic* (*Human error in troubleshooting*). Unpublished doctoral thesis. Aix en Provence, University of Provence, France.

Besnard, D. & Bastien-Toniazzo, M. (1999). Expert error in troubleshooting: an exploratory study in electronics. *International Journal of Human-Computer Studies*, 50, 391-405.

Besnard, D. (2000). Expert error. The case of troubleshooting in electronics.

67 This publication is a revised version of a communication given at the *Flighdeck of the Future conference (Baxter, Besnard & Riley, 2004).*

SafeComp 2000, Rotterdam (pp.74-85).

Besnard, D. (2001). Attacks in IT systems. A human-factors centred approach. Supplement to *2001 International Conference on Dependable Systems and Networks* (DSN-2001), Göteborg (Article B-72).

Besnard, D. & Cacitti, L. (2001). Troubleshooting in mechanics: a heuristic matching process. *Cognition, Technology and Work*, 3, 150-160.

Besnard, D. & Lawrie, A. T. (2002). Lessons from industrial design for software engineering through constraints identification, solution space optimisation and reuse. *ACM Symposium on Applied Computing.* Madrid (pp. 732-738).

Besnard, D. (2003). Building dependable systems with fallible machines. *5th CaberNet Plenary Workshop*, 5-7 November, Madeira.

Besnard, D. & Baxter, G. (2003). *Human compensations for undependable systems*. Technical report CS-TR-819, University of Newcastle.

Besnard, D. & Greathead, D. (2003). A cognitive approach to safe violations. *Cognition, Technology & Work*, 5, 272-282.

Besnard, D. & Greathead, D. & Baxter, G. (2004). When mental models go wrong. Co-occurrences in dynamic, critical systems. *International Journal of Human-Computer Studies*, 60, 117-128.

Besnard, D. & Jones, C. (2004). Designing dependable systems needs interdisciplinarity, *Safety-Critical Systems Club's Newsletter*, May issue, 6-9.

Besnard D. & Arief, B. (2004). Computer security impaired by legal users. *Computers & Security*, 23, 253-264.

Besnard, D. & Cacitti, L. (2005). Interface changes generating accidents. An empirical study of negative transfer. *International Journal of Human-Computer Studies*, 62, 105-125.

Besnard, D. & Baxter, G. (2005). Cognitive conflicts in aviation. Managing computerised critical environments. Article presented at the *DIRC Annual Conference*, 15-17 Mars, Edinburgh.

Besnard, D. & Baxter, G. (2006). Cognitive conflicts in dynamic systems. In Besnard, D., Gacek, C. & Jones, C.B. (Eds) *Structure for Dependability: Computer-Based Systems from an Interdisciplinary Perspective.* London, Springer.

Besnard, D. (2006). Procedures, programs and their impact on dependability. In Besnard, D., Gacek, C. & Jones, C.B. (Eds) *Structure for Dependability: Computer-Based Systems from an Interdisciplinary Perspective.* London, Springer.

Bieder, C. (2000). Comments on the JCO accident. *Cognition, Technology & Work*, 2, 204-205.

Bisseret, A. (1988). Modèles pour comprendre et réussir. in J.-P. Caverni (Ed).

Psychologie cognitive, modèles et méthodes. Grenoble, PUG.

Billings, C. E. (1997). *Aviation automation*. Mahwah, NJ, Lawrence Erlbaum.

Bisseret, A., Figeac-Letang, C. & Falzon, P. (1988). Modélisation de raisonnements opportunistes : L'activité des spécialistes de régulation des carrefours à feux. *Psychologie Française*, 33, 161-169.

Blackman, H. S., Gertman, D. & Hallbert, B. (2000). The need for organisational analysis. *Cognition, Technology & Work*, 2, 206-208.

Blockey, P. N. & Hartley, L. R. (1995). Aberrant driving behaviour: errors and violations. *Ergonomics*, 38, 1759-1771.

Boeing (2007). *Statistical summary of commercial jet airplanes accidents – Worldwide operations 1959-2006*. Retrieved on 08-08-2008 from http://www.boeing.com/news/techissues/pdf/statsum.pdf

Bonnardel, N., Didierjean, A. & Marmèche, E. (2003). Analogie et résolution de problèmes. In C. Tijus (Ed.) *Métaphores et Analogies*. Paris, Hermès (pp. 115-149).

Bonnardel, N. (2004). *Créativité et conception: Approches cognitives et ergonomiques.* Marseille, Solal.

Boreham, N. C. & Patrick, J. (1996). Diagnosis and decision making in work situations. An introduction. *Le Travail Humain*, 59, 1-4.

Boshuizen, H. P. A., Hobus, P. P. M., Custers, E. J. F. M. & Schmidt, H. G. (1991). Cognitive effects of practical experience. In D. A. Evans & V. L. Patel (Eds). *Advanced models of cognition for medical training and practice*. Heidelberg, Springer Verlag (pp. 337-348).

Boudes, N. & Cellier, J.-M. (1998). Etude du champ d'anticipation dans le contrôle du trafic aérien. *Le Travail Humain*, 61, 29-50.

Boy, G. (1987). Operator assistant systems. *International Journal of Man-Machine Studies*, 27, 541-554.

Boy, G. (2005). Human-Centered Automation of Transportation Systems. Paper given at the *AAET 2005 Conference.* Braunschweig, Germany.

Boy, G., Barbé, J. & Giuliano, S. (2007). Automation and assistance in aeronautics and automotive: Diversity versus homogenization? Paper given at the *AAET 2007 Conference*, Braunschweig, Germany.

Brehmer, B. (1987). Development of mental models for decision in technological systems. In J. Rasmussen, K. Duncan & J. Leplat. (Eds). *New technology and human error*. Chichester, UK, Wiley.

Brehmer, B. & Svenmark, P. (1995). Distributed decision making in dynamic environments: time scales and architectures of decision making. In J.-P. Caverni, M. Bar-Hillel, F. H. Barron & H. Jungermann (Eds). *Contributions to decision making-1*. Amsterdam, Elsevier Science.

Brehmer, B. (1996). Man as a stabilizer of systems. From static snapshots of judgement processes to dynamic decision makingg. *Thinking and*

Reasoning, 2, 225-238.

Broderick, R. (1997). *Knowledge-based aircraft automation. Managers guide on the use of artificial intelligence for aircraft automation and verification and validation approach for a neural-based flight controller*. NASA report CR-205078.

Brooks, L. R., Norman, G. R. & Allen, S. W. (1991). Role of specific similarity in a medical diagnostic task. *Journal of Experimental Psychology: General*, 120, 278-287.

Bruner, J. S., Goodnow, J. J. & Austin, G. A. (1967). *A study of thinking*. John Wiley & Sons.

Cacciabue, P. C. (1991). Cognitive ergonomics: a key issue for human-machine systems. *Le Travail Humain*, 54, 359-364.

Cacciabue, C. C. & Kjaer-Hansen, J. (1993). Cognitive modeling and human-machine interactions in dynamic environments. *Le Travail Humain*, 56, 1-26.

Cacciabue, P. C., Fujita, Y., Furuta, K. & Hollnagel, E. (2000). The rational choice of error. *Cognition, Technology & Work*, 2, 179-181.

Callantine, T. (2001). Analysis of Flight Operational Quality Assurance Data Using Model-Based Activity Tracking. *SAE Technical paper 2001-01-2640.* Warrendale, PA, SAE International.

Callantine, T. (2003). Detecting and simulating pilot errors for safety enhancement. SAE Technical Paper 2003-01-2998, Warrendale, PA, SAE International.

Caverni, J.-P. (1991). Psychological modeling of cognitive processes in knowledge assessment by experts: Some convergent issues with psychological modeling in medical reasoning. In D. A. Evans & V. L. Patel (Eds). *Advanced models of cognition for medical training and practice*. Heidelberg, Springer Verlag (pp 245-254).

Cellier, J.-M., Eyrolle, H. & Marine, C. (1997). Expertise in dynamic systems. *Ergonomics*, 40, 28-50.

Clarke, K., Hugues, J., Martin, D., Rouncefield, M., Sommerville, I., Gurr, C., Hartswood, M., Procter, R., Slack, R. & Voss, A. (2003). Dependable red hot action. In K. Kuutti, E. H. Karste, G. Fitzpatrick, P. Dourish & K. Schmidt (Eds). *Proceedings of the eighth European conference on computer-supported cooperative work.*

Crow, J., Javaux, D. & Rushby, J. (2000). Models of mechanised methods that integrate human factors into automation design. *International Conference on Human-Computer Interaction in Aeronautics: HCI-Aero 2000.* Toulouse, France.

Cuny, X. (1979). Different levels of analysing process control tasks. *Ergonomics*, 22, 415-425.

Custers, E. J. F., Boshuizen, H. P. A. & Schmidt, H. G. (1996). The influence of

medical expertise, case typicality and illness script component on case processing and disease probability estimates. *Memory and cognition*, 24, 384-399.

Dale, H. C. (1957). Fault-finding in electronic equipment. *Ergonomics*, 1, 356-385.

Damania, R. (2002). Environmental policies with corrupt bureaucrats. *Environment and Development Economics*, 7, 407-427.

Decortis, F. (1993). Operator strategies in a dynamic environment in relation to an operator model. *Ergonomics*, 36, 1291-1304.

Degani, A., Shafto, M. & Kirlik, A. (1997). On the types of modes in human-machine interactions, *Proceedings of the Ninth International Symposium on Aviation Psychology*. Columbus, OH.

Dehais, F., Tessier, C. & Chaudron, L. (2003). *GHOST: experimenting conflicts countermeasures in the pilot's activity* IJCAI 2003, Acapulco, Mexico.

DeKeyser V. & Woods D.D. (1990). Fixation errors: failures to revise situation assessment in dynamic and risky systems. In A. G. Colombo & A. Saiz de Bustamante (Eds). *Systems reliability assessment*. Dordrecht, The Netherlands, Kluwer (pp. 231-251).

Dekker, S. & Woods, D. D. (1999). Automation and its impact on human cognition. In S. Dekker & E. Hollnagel (Eds). *Coping with complexity in the cockpit*. Aldershot, Ashgate (pp. 7-27).

Dekker, S. (2003). Failure to adapt or adaptations that fail: contrasting models on procedures and safety. *Applied Ergonomics*, 34, 233-238.

Doireau, P., Wioland, L. & Amalberti, R. (1995). La détection de l'erreur par un tiers en situation de pilotage d'avions. *Service de Santé des Armées. Travaux Scientifiques*, 16, 291-292.

Doireau, P., Wioland, L. & Amalberti, R. (1997). La détection des erreurs humaines par des opérateurs extérieurs à l'action : le cas du pilotage d'avion. *Le Travail Humain*, 60, 131-153.

Dowell, J. (1995). Coordination in emergency operations and the tabletop training exercise. *Le Travail Humain*, 58, 85-102.

Duncan, K. D. (1985). Representation of fault-finding problems and development of fault-finding strategies. *Programmed Learning and Educational Technology*, 22, 125-131.

Ebbinghaus, H. (1885). *Memory: A contribution to experimental pscyhology*. New York, Columbia University.

Endsley, M. (1996). Automation and situation awareness. In R. Parasuraman & M. Mouloua (Eds). *Automation and human performance: Theory and applications*. Mahwah, NJ, Lawrence Erlbaum (pp. 163-181).

FAA (2003). General aviation controlled flight into terrain awareness. Advisory Circular 31-134. Retrieved on 08-08-2008 from

http://rgl.faa.gov/Regulatory_and_Guidance_Library/rgAdvisoryCircular.nsf/0/2f2d1a8d9fe4d96586256d04006f2065/$FILE/ac61-134.pdf

FAA Human Factors Team (1996). *The Interfaces Between Flightcrews and Modern Flight Deck Systems*. Federal Aviation Administration, Washington, DC.

Fadier, E., De La Garza, C. & Didelot, A. (2003). Safe design and human activity: construction of a theoretical framework from an analysis of a printing sector. *Safety Science*, 41, 759-789.

Fadier, E. & De La Garza, C. (2006). Safet design: Towards a new philosophy. *Safety Science*, 55-73.

Fath, J. L., Mitchell, C. M. & Govindaraj, T. (1990). An ICAI architecture for troubleshooting in complex dynamic systems. *IEEE Transactions on Systems, Man and Cybernetics*, 20, 537-558.

Fitts, P. M. (1951). *Human engineering for an effective air navigation and traffic control system*. Colombus, OH, Ohio State University Foundation Report.

Filgueras, L. V. L. (1999). Human performance reliability in the design-for-usability life cycle for safety human computer interfaces. in M. Felici, K. Kannoun & A. Pasquini (Eds). *SafeComp*'99. Heidelberg, Springer-Verlag (pp. 79-88).

Flin, R., Slaven, G. & Stewart, K. (1996). Emergency decision making in the offshore oil and gas industry. *Human Factors*, 38, 262-277.

Fujita, 2000 understanding of risk associated with one's actions Fujita, Y. (2000). Actualities need to be captured. *Cognition, Technology & Work*, 2, 212-214.

Furuta, K., Sasou, K., Kubota, R,. Ujita, H., Shuto, Y. & Yagi, E. (2000). Human factor analysis of JCO criticality accident. *Cognition, Technology & Work*, 2, 182-203.

Gibson, F. P., Fichman, M. & Plaut, D. C. (1997). Learning in dynamic decision tasks: computational model and empirical evidence. *Organizational Behavior and Human Decision Processes*, 71, 1-35.

Gitus, J. H. (1988). The *Chernobyl accident and its consequences*. London, United Kingdom Atomic Energy Authority.

Gobet, F. & Simon, H. A. (1996a). Recall of random and distorted chess positions: Implications for the theory of expertise. *Memory and Cognition*, 24, 493-503.

Gobet, F. & Simon, H. A. (1996b). Templates in chess memory: a mechanism for recalling several boards. *Cognitive Psychology*, 31, 1-40.

Goldbeck, R. A., Bernstein, B. B., Hillix, W. A. & Marx, M. H. (1957). Application of the split-half technique to problem solving tasks. *Journal of Experimental Psychology*, 53, 330-338.

Hassebrok, F. & Prietula, M. J. (1992). A protocol-based coding scheme for the analysis of medical reasoning. *International Journal of Man-Machine Studies*, 37, 613-652.

Hayes-Roth, B. & Hayes-Roth, F. (1979). A cognitive model of planning. *Cognitive Science*, 3, 275-310.

Haynes, A. (1991). *The crash of United flight 232*. Transcript of public recording. NASA Ames.

Hoc, J.-M. (1989). Strategies in controlling a continuous process with long response latencies: needs for a computer support to diagnosis. *International Journal of Man-Machine Studies*, 30, 47-67.

Hoc, J-M. (1991). Effets de l'expertise des opérateurs et de la complexité de la situation dans la conduite d'un processus continu à long délai de réponse : le haut fourneau. *Le Travail Humain*, 54, 225-249.

Hoc, J.-M. & Amalberti, R. (1994). Diagnostic et prise de décision dans les situations dynamiques. *Psychologie Française*, 39, 177-192.

Hoc, J.-M. (1996). *Supervision et contrôle de processus. La cognition en situation dynamique*. Grenoble, PUG.

Hollan, J., Hutchins, E. & Kirsh, D. (2000). Distributed cognition: toward a new foundation for human-computer interaction research. *ACM Transactions on Computer-Human Interaction*, 7, 174-196.

Hollnagel, E. (1987). Information and reasoning in intelligent decision support systems. *International Journal of Man-Machine Studies*, 27, 665-678.

Hollnagel, E. (1993). The phenotype of erroneous actions. *International Journal of Man-Machine Studies*, 39, 1-32.

Hollnagel, E. (1998). *Cognitive Reliability and Error Analysis Method*. Elsevier.

Hollnagel, E. & Woods, D. (1999). Cognitive systems engineering: New wine in new bottles. *International Journal of Human-Computer Studies*, 51, 339-356.

Hollnagel, E. (2004). *Barriers and accident prevention*. Aldershot, Ashgate.

Hollnagel, E. & Woods, D. D. (2005). *Joint Cognitive Systems: Foundations of Cognitive Systems Engineering*. CRC Press, Boca Raton, FL.

James, W. (1950). *The principles of psychology* (Reprinted from the original edition from 1890). Dover Publications Inc.

Joblet, L. (1997). *Approches de conduite du nucléaire et répartition des tâches entre opérateurs*. Unpublished internship report. Aix en Provence, University of Provence.

Johnson, C. (2004). Looking Beyond the Cockpit: Human Computer Interaction in the Causal Complex of Aviation Accidents". In A. Pritchett & A. Jackson. *HCI in Aerospace 2004*, Eurisco, Toulouse, France.

Johnson, C. & Holloway, M. (2007). The dangers of failure masking in fault-tolerant software: aspects of a recent in-flight upset event. *The 2nd*

Institution of Engineering and Technology International Conference on System Safety, London, UK.

Johnson, C. & Shea, C. (2007). A comparison of the role of degraded modes of operation in the causes of accidents in rail and air traffic management. *The 2nd Institution of Engineering and Technology International Conference on System Safety*, London, UK.

Kaempf, G. L., Klein, G., Thordsen, M. L. & Wolf, S. (1996). Decision making in complex naval command and control environments. *Human Factors*, 38, 220-231.

Kemeny, J. G. (1981). *The President's commission on the accident at Three Mile Island*. New York, Pergamon Press.

Kersholt, J. (1995). Decision making in a dynamic situation: The effects of faulse alarms and time pressure. *Journal of Behavioral Decision Making*, 8, 181-200.

Klein, G. (1997). The recognition-primed decision model: Looking back, looking Forward. In C.E. Zsambok & G. Klein (Eds). *Naturalistic decision making*. Mahwah, NJ, Lawrence Erlbaum (pp. 285-292).

Klein, G., Pliske, R. & Crandall, B. (2005). Problem detection. *Cognition, Technology & Work*, 7, 14-28.

Konradt, U. (1995). Strategies of failure diagnosis in computer-controlled manufacturing systems: empirical analysis and implications for the design of adaptive decision support systems. *International Journal of Human-Computer Studies*, 43, 503-521.

Kuipers, B. & Kassirer, J. P. (1984). Causal reasoning in medecine: analysis of a protocol. *Cognitive Science*, 363-385.

Lamarsh, J. R. (1981). Safety considerations in the design and operation of light water nuclear power plants. In T. H. Moss & D. L. Sills (Eds). *The Three Mile Island Accident: Lessons and Implications.* New York, The New York Academy of Sciences.

Lee, S. C. (1994). Sensor validation based on systematic exploration of the sensor redundancy for fault diagnosis knowledge based systems. *IEEE Transactions on Systems, Man and Cybernetics*, 24, 594-605.

Leveson, N. & Turner, C. S. (1993). An investigation of the Therac-25 accidents. *IEEE Computer*, 26, 18-41.

Leveson, N. G., Pinnel, L. D., Sandys, S. D., Koga, S. & Reese, J. D. (1997). Analysing software specifications for mode confusion potential. In C. W. Johnson (Ed). *Proceedings of a workshop on human error and system development*, Glasgow, Scotland (pp. 132-146).

Leveson, N. & Palmer, E. (1997). Designing automation to reduce operator errors. In proceedings of *IEEE Conference on Systems, Man and Cybernetics*, Orlando, FL.

Lions, J.-L. (1996). *Ariane 5 Flight 501 Failure. Report by the Inquiry Board*.

Retrieved on 08-07-2008 from http://sunnyday.mit.edu/accidents/Ariane5accidentreport.html

Loeb, V. (2002). Friendly fire traced to dead battery. Washington Post, March 24, p. A21.

Mach, E. (1905). *Knowledge and error. Sketches on the Psychology of Enquiry* (English translation: Dordrecht, Reidel, 1976).

Medin, D. L., Altom, M. W., Edelson, S. M. & Freko, D. (1982). Correlated symptoms and simulated medical classification. *Journal of Experimental Psychology: Learning, Memory and Cognition*, 8, 37-50.

Miller, G. A. (1956). The magical number seven plus or minus two: Some limits on our limits for processing information. *Psychological Review*, 63, 81-97.

Miller, D. P. and Swain, A. D. (1987). Human error and human reliability. In: G. Salvendy (Ed). Handbook of Human Factors. New York, Wiley Interscience.

Millot, P. (1990). Coopération homme-machine dans les procédés automatisés. In J. Leplat & G. de Terssac. *Les facteurs humains de la fiabilité dans les systèmes complexes*. Marseille, France, Octares.

Milne, R. (1987). Strategies for diagnosis. *IEEE Transactions on Systems, Man and Cybernetics*, 17, 333-339.

METT (Ministère de l'Equipement, des Transports et du Tourisme) (1993). *Rapport de la commission d'enquête sur l'accident survenu le 20 Janvier 1992 près du mont Sainte Odile à l'Airbus A.320 immatriculé F-GGED exploité par la compagnie Air Inter*.

Ministry of Transport. (1996). *Aircraft Accident Investigation Commission. China Airlines Airbus Industries A300B4-622R, B1816, Nagoya Airport, April 26, 1994*. (Report 96-5). Japan, Ministry of Transport.

Mozetic, I. (1991). Hierarchical model-based diagnosis. *International Journal of Man-Machine Studies*, 35, 329-362.

Mumma, G. H. (1993). Categorization and rule induction in clinical diagnosis and assessment. *The Psychology of Learning and Motivation*, 29, 283-326.

Neisser, U. (1976). *Cognition and reality*. San Francisco, Freeman & Company.

Newell, A. & Simon, H. (1972). *Human Problem Solving*. Cliffs, NJ, Prentice-Hall.

Newell, A., Shaw, J. C., Simon, H. A. (1959). Report on a general problem-solving program. *Proceedings of the International Conference on Information Processing* (pp. 256-264).

Norman, G. R., Brooks, L. R. & Allen, S. W. (1989). Recall by expert medical practioners and novices as a record of processing attention. *Journal of Experimental Psychology. Learning, Memory and Cognition*, 15, 1166-

1174.

Nooteboom, P. & Leemeijer, G. B. (1993). Focusing based on the structure of a model in model-based diagnosis. *International Journal of Man-Machine Studies*, 38, 455-474.

NTSB (1990) Aircraft accident report. United Airlines flight 232. Mc Donnell Douglas DC-10-10. Sioux Gateway airport. Sioux City, Iowa, July 19, 1989. Washington DC.

NTSB (1997a). *Grounding of the Panamanian passenger ship Royal Majesty on Rose and Crown shoal near Nantucket, Massachusetts, June 10, 1995.* Marine Accident Report NTSB/MAR-97/01. Washington, DC.

NTSB (1997b). *Wheels-up Landing, Continental Airlines Flight 1943, Douglas DC-9-32, N10556, Houston, Texas, February 19, 1996*. Air Accident Report AAR-97/01. Washington, DC.

Ohlsson, S. (1996). Learning from performance errors. *Psychological Science*, 103, 241-262.

Onken, R. (1997). The Cockpit Assistant System CASSY as an onboard player in the ATM environment. Paper presented at the *1st USA/Europe Air Traffic Management Research & Development Seminar*, Saclay, France.

Parker, D., Reason, J. T., Manstead, S. R. & Stradling, S. G. (1995). Driving errors, driving violations and accident involvement. *Ergonomics*, 38, 1036-1048.

Pascoe, E. & Pidgeon, N. (1995). Risk orientation in dynamic decision making. In J.-P. Caverni, M. Bar-Hillel, F. H. Barron & H. Jungerman (Eds). *Contributions to decision making - 1*. Amsterdam, Elsevier Science.

Patrick, J. (1993). Cognitive aspects of fault-finding training and transfer. *Le Travail Humain*, 56, 187-209.

Paxton, H. C., Baker, R. D. & Reider, W. J. (1959). Los Alamos criticality accident. *Nucleonics*, 17, 107.

Pazzani, M. J. (1987). Failure-driven learning of fault diagnosis heuristics. *IEEE Transactions on Systems, Man and Cybernetics*, 17, 380-394.

Perrow, C. (1984). *Normal accidents*. New York, Basic Books.

Poyet, C. (1990). L'homme agent de fiabilité dans les systèmes automatisés. In J. Leplat & G. de Terssac. *Les facteurs humains de la fiabilité dans les systèmes complexes*. Octares, Marseille, France.

Rame, J.-M. (1995). Rôle des industriels dans la prévention des accidents. *Pilote de ligne*, 5, 20-21.

Randel, J. M. & Pugh, H. L. (1996). Differences in expert and novice situation awareness in naturalistic decision making. *International Journal of Human-Computer Studies*, 45, 579-587.

Randell, B. (2000). Facing up to faults. *Computer Journal*, 43, 95-106.

Rasmussen, J. & Jensen, A. (1974). Mental procedures in real life tasks. A case

study in electronics trouble shooting. *Ergonomics*, 17, 293-307.

Rasmussen, J. (1986). *Information processing and human-machine interaction*. North Holland, Elsevier Science.

Rasmussen, J. (1991). Technologie de l'information et analyse de l'activité cognitive. In R. Amalberti, M. De Montmollin & J. Theureau *Modèles en analyse du travail*. Liège, Mardaga (pp. 49-73).

Rasmussen, J. (1993). Diagnostic reasoning in action. *IEEE Transactions on Systems, Man and Cybernetics*, 23, 981-992.

Rauterberg, M. (1995). About faults, errors and other dangerous things. In H. Stassen & P. Wieringa (Eds). Proceedings of the *XIV European annual conference on human decision making and manual control*, Delft University of Technology (pp. 1-7).

Reason, J. (1987) Chernobyl errors. *Bulletin of the British Psychological Society*, 40, 201-206.

Reason, J. (1990). *Human error*. Cambridge, Cambridge University Press.

Reason, J. (1995). A systems approach to organized error. *Ergonomics*, 38, 1708-1721.

Reason, J. (1997). *Managing the risks of organizational accidents*. Aldershot, Ashgate.

Reason, J. (2000). Human errors. Models and management. *British Journal of Management*, 320, 768-770.

Reason, J. (2001). Heroic compensations. *Flight Safety Australia*, January-February, 28-31.

Rey, P. & Bousquet, A. (1995). Compensation for occupational injuries and diseases : Its effects upon prevention at the workplace. *Ergonomics*, 38, 475-486.

Richard, J. F. (1990). Les activités mentales. Paris, PUF.

Rouse, W. B. (1978). Human problem solving performance in a fault diagnosis task. *IEEE Transactions on Systems, Man and Cybernetics*, 8, 258-271.

Rushby, J., Crow, J. & Palmer, E. (1999). An automated method to detect potential mode confusions. Proceedings of the *18th AIAA/IEEE Digital Avionics Systems Conference*, St Louis, MO.

Samurçay, R. & Hoc, J. M. (1996). Causal versus topographical support for diagnosis in a dynamic situation. *Le Travail Humain*, 59, 45-68.

Sanderson, P. M. (1990). Knowledge acquisition and fault diagnosis: experiments with PLAULT. *IEEE Transactions on Systems, Man and Cybernetics*, 20, 255-242.

Sarter, N. & Woods, D. D. (1995). How in the world did we ever get into that mode? Mode error and awareness in supervisory control. *Human Factors*, 37, 5-19.

Sarter, N. & Woods, D. D. (1997). Teamplay with a powerful and independent agent: A corpus of operational experiences and automation surprises on the airbus A-320. *Human Factors*, 39, 553-569.

Sasse, M. A. Brostoff, S. & Weirich, D. (2001). Transforming the “weakest link” – a human/computer interaction approach to usable and effective security. *BT Technology Journal*, 19, 3, 122-131.

Schaafstal, A. (1993). Knowledge and strategies in diagnostic skill. *Ergonomics*, 36, 1305-1316.

Schraagen, J. M. & Schaafstal, A. M. (1996). Training of systematic diagnosis: A case study in electronics troubleshooting. *Le Travail Humain*, 59, 5-21.

Shappell, S. A. & Wiegmann, D. A. (2003). A human error analysis of general aviation controlled flight into terrain accidents occurring between 1990-1998.

Sheen, J. (1987). *MV Herald of Free Enterprise. Report of Court No. 8074 Formal Investigation*. London, Department of Transport.

Sheridan, T. B. (1992). *Telerobotics, automation, and human supervisory control.* Cambridge, MA, MIT Press.

Sherry, R. R. & Ritter, F. E. (2002). *Dynamic task allocation: Issues for implementing adaptive intelligent automation*. Technical Report No ACS 2002-2, The Pensylvania State University.

Simon, H. A. (1957). *Models of man - social and rational*. New York, John Wiley and Sons

Simon, H. A. (1996). *The sciences of the artificial* (3rd ed.). Cambridge, MA, MIT Press.

Simpson, S. A. & Gilhooly, K. J. (1997). Diagnostic thinking processes: evidence from a constructive interaction study of electrocardiogram (ECG) interpretation. *Applied Cognitive Psychology*, 11, 543-554.

Smith, E. E. & Sloman, S. A. (1994). Similarity versus rule-based categorization. *Memory and Cognition*, 22, 377-386.

Soloway, E., Adelson, B. & Ehrlich, K. (1988). Knowledge and processes in the comprehension of computer programs. In M. T. H. Chi, R. Glaser & M. J. Farr *The nature of expertise*. Hillsdale, NJ, Lawrence Erlbaum.

Spérandio, J.-C. (1987). Les aspects cognitifs du travail. in C. Levi-Leboyer & J.-C. Spérandio *Traité de psychologie du travail*. Paris, PUF (pp. 646-658).

Sundström, G. A. (1993). Towards models of tasks and task complexity in supervisory control applications. *Ergonomics*, 11, 1413-1423.

Svenson, O, Lekberg, A. & Johansson, A. E. L. (1999). On perspective, expertise and differences in accident analyses: Arguments for a multidisciplinary approach. *Ergonomics*, 42, 1567-1571.

Svenson, O. (1990). Some propositions for the classification of decision

situations. In K. Borcherding, O. I. Larichev & D. M. Messik (Eds). *Contemporary issues in decision making*. North Holland, Elsevier Science.

Swain, A. D., & Guttman, H. E. (1983). *Handbook of human reliability analysis with emphasis on nuclear power plant applications*. Washington D.C., NUREG/CR-1278.

Tversky, A. & Kahneman, D. (1974). Judgements under uncertainty: Heuristics and biases. *Science*, 185, 1124-1131.

USSR State committee on the utilization of atomic energy (1986). *The accident at the Chernobyl nuclear power plant and its consequences*. Information compiled for the IAEA Experts Meeting, 25-29 August, Vienna (pp. 1-13).

Vanderhaegen, F. (2003). *Analyse et contrôle de l'erreur humaine dans les systèmes homme-machine*. London, Hermes Science Publications.

Van der Schaaf, T. (1992). Near miss reporting in the chemical process industry. Proefschrift, TU Eindhoven.

Van der Schaaf, T. (2000). Near miss reporting changes the safety culture (Report after a visit to the University of Wisconsin-Madison). *The Human Element*, 5, 1-2.

Vincente, K. (1999). *Cognitive Work Analysis : Toward Safe, Productive, and Healthy Computer-Based Work*. Mahwah, NJ, Lawrence Erlbaum.

Voss, A., Slack, R., Procter, R., Williams, R., Harstwood, M. & Rouncefield (2002). Dependability as ordinary action. In S. Anderson & M. Felici (Eds). *SafeComp 2002*, LNCS 2434, Berlin, Springer-Verlag (pp. 32-43).

Wagemann, L. (1998). Analyse des représentations initiales liées aux interactions homme-machine en situation de conduite simulée. *Le Travail Humain*, 61, 129-151.

Wagenaar, W. A. & Groeneweg, J. (1987). Accidents at sea: Multiple causes and impossible consequences. *International Journal of Man-machine Studies*, 27, 587-598.

Weick, K. e., Suthcliffe, K. M & Obstfeld, D. (2005). Organising the process of sensemaking. *Organisation Science*, 16, 409-421.

Westrum, R. (2000). Safety planning and safety culture in the JCO criticality accident: Interpretive comments. *Cognition, Technology & Work*, 2, 240-241.

Wimmer, M., Rizzo, A. & Sujan, M. (1999). A holistic design concept to improve safety related control systems. in M. Felici, K. Kannoun & A. Pasquini (Eds). *SafeComp'99*, Springer-Verlag, Heidelberg (pp. 297-309).

Woods, D. (1993). The price of flexibility. Proceedings of *Intelligent User Interfaces '93,* Orlando, FL (pp. 19-25).

Woods, D. D. & Shattuck, L. G. (2000). Distant supervision-local action given

the potential for surprise. *Cognition, Technology & Work*, 2, 242-245.

Woods, D. D. & Dekker, S. (2000). Adapting the effects of technological change: a new era of dynamics for human factors. *Theoretical Issues in Ergonomics Science*, 272-282.

French Translation Of Technical Terms

This section is not a glossary since it does not provide definitions, the latter being inserted as footnotes in the main text. The terms listed below are only translations for the French reader.

Term	Section	Translation
Split-half test	French summary	Test d'élimination par moitiés
Bow-tie model	1.4.4	Modèle papillon
Intake manifold	1.5.2	Collecteur d'admission
Condenser	1.6.2	Condensateur
Beacon	2.2.2	Balise
Slat	2.3.3	Bec (de bord d'attaque)
Flap	2.3.3	Volet (de bord de fuite)
Thrust	2.4.2	Poussée
Rudder	2.4.2	Dérive
Dead reckoning	2.4.4	Navigation à l'estime
Wire drawing machine	2.5.2	Tréfileuse
Fan blade	2.5.3	Ailette
Steel slab	3.2.2	Plaque d'acier
Roughing mill	3.2.2	Dégrossisseur