



HAL
open science

**CONTRIBUTION A LA CONCEPTION
D'INTERFACE "INTELLIGENTE" POUR LE
CONTRÔLE de procédés industriels: Application au
Module Décisionnel d'Imagerie**

Mustapha Tendjaoui

► **To cite this version:**

Mustapha Tendjaoui. CONTRIBUTION A LA CONCEPTION D'INTERFACE "INTELLIGENTE" POUR LE CONTRÔLE de procédés industriels: Application au Module Décisionnel d'Imagerie . Interface homme-machine [cs.HC]. l'Université de Valenciennes et du Hainaut-Cambrésis, 1992. Français. NNT : 1992VALE0020 . tel-01421142

HAL Id: tel-01421142

<https://hal.science/tel-01421142v1>

Submitted on 21 Dec 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THESE

présentée à l'Université de Valenciennes
et du Hainaut-Cambrésis
pour l'obtention du

DOCTORAT

Spécialité Automatique Industrielle et Humaine

Par

Mustapha TENDJAOUI

Ingénieur de l'Ecole Polytechnique d'Alger

**CONTRIBUTION A LA CONCEPTION
D'INTERFACE "INTELLIGENTE" POUR LE
CONTROLE
de procédés industriels :
Application au Module Décisionnel d'Imagerie**

Soutenue le 20 Novembre 1992 devant la commission d'examen :

MMe	:	Marie-Claude THOMAS	:	rapporteur
MM	:	Jean-Claude ANGUE		
		Guy BOY	:	rapporteur
		Christophe KOLSKI		
		Patrick MILLOT		

A ma famille,
à ma femme et...

AVANT-PROPOS

Le travail décrit dans ce mémoire a été entrepris à l'Université de Valenciennes et du Hainaut Cambrésis, au sein du Laboratoire d'Automatique Industrielle et Humaine, et particulièrement dans l'équipe "Informatique Industrielle et Communication Homme-Machine" dirigée par le Professeur Patrick MILLOT.

Je tiens à remercier vivement Madame Marie-Claude THOMAS, Directrice de recherche au Laboratoire I3S ainsi que Monsieur Guy BOY, Directeur de recherche à l'ONERA, d'être les rapporteurs de mon travail.

J'exprime ma sincère gratitude à Monsieur le Professeur Patrick MILLOT pour l'aide, les conseils et le soutien qu'il m'a dispensé depuis mon premier jour dans son équipe.

Je remercie particulièrement Christophe KOLSKI, Maître de conférences, pour son soutien amical et professionnel, pour le temps qu'il n'a cessé de consacrer à la direction de mes travaux.

Je tiens à exprimer l'honneur que me fait Monsieur Jean-Claude ANGUE, Directeur de recherche au L.A.I.H., de faire partie de cette commission.

Je remercie également toutes les personnes qui ont contribué à la réalisation de ces travaux de recherche et à la préparation de ce mémoire.

Je remercie vivement mes amis(es) et chercheurs du L.A.I.H., pour leur soutien et le climat de travail agréable et amical qu'ils ont créé.

Enfin, je remercie particulièrement ma femme et ma famille qui ont su être toujours présents au moment où j'ai eu besoin d'eux.

à mes amis

La science naît au moment où la
société décide de dominer le
monde ou de le reconstruire.

I.M. AL HUSSEINI (1905)

SOMMAIRE

SOMMAIRE	5
INTRODUCTION GENERALE	9
CHAPITRE I : SYSTEMES HOMME-MACHINE ET ASSISTANCE AUX OPERATEURS	14
INTRODUCTION.....	14
I. POSITIONNEMENT DES SYSTEMES D'ASSISTANCE DANS L'ARCHITECTURE DES SYSTEMES DE SUPERVISION DE PROCEDES INDUSTRIELS	16
II. LES BESOINS D'ASSISTANCE DES OPERATEURS DANS LES SALLES DE CONTROLE	18
III. CLASSIFICATION DES SYSTEMES D'ASSISTANCE	32
CONCLUSION.....	43
CHAPITRE II : SYSTEMES DE DIALOGUE HOMME-MACHINE EVOLUES.....	46
INTRODUCTION.....	46
I. LES INTERFACES FLEXIBLES	47
II. LES INTERFACES TOLERANTES AUX ERREURS HUMAINES....	52
III. LES INTERFACES EXPERTES	59
IV. LES INTERFACES ADAPTATIVES	63
V. L'OPERATEUR ASSISTANT.....	69
VI. L'AGENT INTELLIGENT	74
VII. LES INTERFACES ECOLOGIQUES	79
VIII. LES INTERFACES HYPERMEDIA.....	83
CONCLUSION.....	88
CHAPITRE III : UNE APPROCHE D'INTERFACE "INTELLIGENTE" : LE MODULE DECISIONNEL D'IMAGERIE (LE M.D.I.).....	93
INTRODUCTION.....	93

I. TRAVAUX ANTECEDENTS AU M.D.I.....	94
II. LE MODULE DECISIONNEL D'IMAGERIE (M.D.I.).....	102
III. LES CONCEPTS DU MODULE DECISIONNEL D'IMAGERIE ...	104
CONCLUSION.....	113
CHAPITRE IV REALISATION D'UNE PLATE-FORME EXPERIMENTALE INTEGRANT UNE INTERFACE "INTELLIGENTE" DE TYPE M.D.I.	115
INTRODUCTION.....	115
I. PRESENTATION GLOBALE DE LA PLATE-FORME EXPERIMENTALE.....	116
II. LE SYSTEME HOMME-MACHINE SIMULE	119
III. LES OUTILS D'AIDE A L'OPERATEUR	131
IV. LE MODULE DECISIONNEL D'IMAGERIE	136
CONCLUSION.....	148
CHAPITRE V VALIDATION TECHNIQUE DE LA PLATE-FORME EXPERIMENTALE ET PERSPECTIVES	149
CHAPITRE V : VALIDATION TECHNIQUE DE LA PLATE-FORME EXPERIMENTALE ET PERSPECTIVES	150
INTRODUCTION.....	150
I. PREMIERE VALIDATION TECHNIQUE DE LA PLATE-FORME EN LABORATOIRE.....	151
II. PERSPECTIVES DE RECHERCHE.....	163
CONCLUSION.....	172
CONCLUSION GENERALE	173
BIBLIOGRAPHIE.....	177
ANNEXES.....	193
ANNEXE A : EXEMPLES D'ARBRES DE DECISION GENERES PAR L'OUTIL FIRST-CLASS	194

ANNEXE B : EXEMPLES DE PLANS D' ACTIONS.....	201
ANNEXE C : EXEMPLES DE REGLES DE PRODUCTION.....	203
ANNEXE D : EXEMPLES DE COPIES D'ECRAN.....	205

INTRODUCTION GENERALE

Durant ces dernières années, l'évolution des technologies industrielles, le degré de complexité de certains matériels, le développement dans tous les domaines, ont provoqué des changements importants dans le contenu et les méthodes du travail humain. La place de l'homme dans le système, son rôle, son degré d'autonomie, les qualifications professionnelles exigées, s'en trouvent modifiés de façon majeure (SPERANDIO, 1988). Mais, loin de simplifier les tâches de l'opérateur humain, l'automatisation et l'informatisation des salles de contrôle de procédés industriels continus en ont considérablement accru la complexité. Cette complexité se traduit par la quantité de plus en plus importante d'informations que l'opérateur humain doit traiter afin de mener à bien ses tâches de contrôle et de supervision.

Ainsi, l'automatisation des systèmes de supervision de procédés industriels a laissé à l'opérateur humain une place des plus importantes, où les tâches sont devenues essentiellement des tâches mentales de résolution de problèmes. Et, si l'opérateur dispose de plus en plus d'outils d'aide destinés à l'aider dans ses tâches de conduite et de supervision, ces outils restent inefficaces dans certaines conditions puisque l'ensemble des dysfonctionnements n'est souvent pas modélisable de façon exhaustive.

Parallèlement, les procédés se développent en taille et en complexité, et les contraintes de production et de sécurité et les contraintes économiques deviennent de plus en plus sévères. Dans ces conditions, les risques d'erreurs humaines deviennent très importants et l'opérateur humain se voit alors confier le rôle de garant ultime de la sécurité du système lorsque celui-ci est déficient.

Ainsi, afin d'aider les opérateurs lors de leurs prises de décision, la tendance de ces dernières années consiste à les assister d'outils faisant généralement appel à des techniques de l'Intelligence Artificielle, tels que les systèmes experts. Ces systèmes sont capables de s'adapter à un grand nombre de situations de conduite, en faisant appel à des heuristiques modélisées et stockées dans leurs bases de connaissance. Par exemple, on retrouve de plus en plus, dans les salles de contrôle, des systèmes d'aide au diagnostic, des systèmes d'aide à la correction et à la reprise des défauts, etc. Se pose alors le problème de la communication entre l'opérateur humain et ces systèmes d'aide.

Un champ de recherche récent vise à résoudre celui-ci à l'aide d'une interface graphique assurant les interactions homme-machine, capable de **s'adapter** aux

différents traitements cognitifs mis en œuvre par l'opérateur lors de la réalisation de ses tâches de contrôle et de supervision.

Dans ce but, les recherches actuelles étudient la nature, la composition et l'implantation des interactions homme-machine à travers le développement d'interfaces dites "intelligentes".

Notre travail consiste à étudier et à proposer une approche d'interface "intelligente" appelée Module Décisionnel d'Imagerie (M.D.I.) (TABORIN et al., 1987). Le M.D.I. vise à choisir automatiquement le contenu informationnel des synoptiques de supervision en utilisant des techniques issues de l'intelligence artificielle pour répondre aux trois questions suivantes : (1) **QUOI** présenter à l'opérateur, (2) **QUAND** le présenter et (3) **COMMENT** le présenter. Dans ce but, ce mémoire se compose de cinq chapitres.

Le premier chapitre discute de l'évolution des systèmes homme-machine durant ces dernières années, et du rôle et des caractéristiques de l'opérateur humain dans les salles de contrôle de procédé industriel. Puis, il dresse un bilan des types de systèmes actuellement capables de l'assister dans ses tâches, en mettant en évidence la problématique de l'interface homme-machine.

Le second chapitre est consacré à une étude bibliographique sur les travaux de recherche théoriques et pratiques concernant les interfaces dites "évoluées" ou "intelligentes". On rencontre ainsi dans la littérature des approches d'interfaces évoluées sous la forme *d'interface flexible*, *d'interface tolérante aux erreurs humaines*, *d'interface experte*, *d'interface adaptative*, *d'opérateur assistant*, *d'agent intelligent*, *d'interface hypermédia* ou encore *d'interface écologique*.

Le troisième chapitre justifie d'abord l'orientation de nos travaux vers une interface dite "intelligente", à partir de travaux antérieurs et des principes recensés dans le chapitre II. Ce chapitre présente ensuite les concepts et les principes constituant le M.D.I., ainsi qu'une architecture fonctionnelle intégrant celui-ci.

Le quatrième chapitre est consacré à la description d'une plate-forme expérimentale destinée à évaluer les concepts proposés dans le chapitre précédent. Cette plate-forme est constituée principalement de plusieurs ensembles de modules : une première version du Module Décisionnel d'Imagerie, un outil d'assistance à l'opérateur intégrant un ensemble de fonctionnalités, un procédé simulé, et enfin des

modules spécifiques permettant la gestion des échanges d'informations entre les différents modules de la plate-forme au travers d'une zone mémoire commune.

Dans le cinquième chapitre, une première validation technique de la plate-forme est présentée. Ce chapitre décrit également un ensemble de perspectives de recherche concernant d'une part l'évaluation ergonomique du M.D.I., et d'autre part l'évolution de celui-ci.

Chapitre I

SYSTEMES HOMME-MACHINE ET ASSISTANCE AUX OPERATEURS

CHAPITRE I : SYSTEMES HOMME-MACHINE ET ASSISTANCE AUX OPERATEURS

INTRODUCTION

Les psychologues du travail, ainsi que les automaticiens humains, limitent le système homme-machine à l'ensemble intégré d'éléments en interaction destiné à exécuter en coopération une fonction prédéterminée (ROUHET, 1988). La notion de système est ainsi voisine de celle d'unité fonctionnelle (JOHANNSEN, 1982). En ce sens, le système peut être un poste de travail, un groupe de postes, un atelier, etc. Quand le système homme-machine comporte plusieurs opérateurs humains et plusieurs machines, il est qualifié de multidimensionnel (ROUHET, 1988).

En général, un système homme-machine prend donc la forme d'un ensemble de sous-systèmes et d'éléments dont le fonctionnement de chacun peut être caractérisé par un ensemble de paramètres.

Dans le système Homme-machine élémentaire proposé par LEPLAT et CUNY (1969), l'homme dialogue avec la machine en lui envoyant des commandes. La machine, sensible aux perturbations provenant de son environnement, renvoie différentes données vers l'homme et éventuellement vers d'autres systèmes en aval. Dans ces conditions, l'homme est considéré, particulièrement, comme un récepteur : il reçoit différentes informations provenant de la machine, mais reste très sensible aux autres types de source d'information, tels que le téléphone, les bruits, la voix d'autres opérateurs en amont ou en aval du système homme-machine, etc.

Au cours de ces dernières années, l'évolution technologique a vu les procédés industriels se développer en taille et en complexité. Dans ces conditions, la place de l'homme dans le système, son rôle, son degré d'autonomie, les qualifications professionnelles exigées, s'en trouvent modifiés de façon majeure (SPERANDIO, 1988). En effet, cette évolution se traduit, par exemple, par une quantité d'information de plus en plus importante que l'opérateur humain doit traiter, afin de mener à bien ses tâches de contrôle et de supervision.

Pour répondre aux nouveaux besoins des systèmes homme-machine de ce type et respecter les contraintes de sécurité, de production ou encore de sauvegarde de

l'environnement devenant de plus en plus sévères, une solution consiste à assister l'opérateur humain dans la réalisation de ses tâches. C'est ainsi que des systèmes d'assistance sont introduits progressivement dans les salles de contrôle et de supervision. Ces systèmes exploitent certaines données en provenance du procédé dans le but de générer de nouvelles informations utiles pour l'opérateur humain, tels que des diagnostics, des conseils d'action, etc. Ce dernier dialogue avec l'ensemble des composants de ce système homme-machine, grâce à une interface dont le rôle est de sélectionner et de présenter à l'homme les données utiles provenant de la machine d'une part et du (des) système(s) d'assistance d'autre part.

Nos travaux portent sur les systèmes d'assistance à l'opérateur (G. Boy a écrit un livre sur le sujet en 1988), intégrés dans les systèmes homme-machine de supervision de procédés industriels (de type unités de production continues, réseaux d'énergie, transports automatisés, etc). La première partie de ce chapitre positionne plus précisément la notion de système d'assistance dans l'architecture des systèmes de supervision. La deuxième partie met en évidence différents types d'assistance à apporter à l'opérateur humain, à travers une étude sur ses caractéristiques, son rôle et ses fonctions dans les salles de contrôle. Enfin, la troisième partie dresse une classification des principaux types de systèmes d'assistance existant.

I. POSITIONNEMENT DES SYSTEMES D'ASSISTANCE DANS L'ARCHITECTURE DES SYSTEMES DE SUPERVISION DE PROCÉDES INDUSTRIELS

Pour mieux définir la notion de système d'assistance, il est essentiel de situer ce dernier par rapport au système de supervision de procédés complexes. L'utilisation croissante des systèmes informatiques, tels que les calculateurs, les micro-calculateurs, ou les automates programmables mène actuellement à une architecture de conduite basée sur la notion de contrôle hiérarchisé, figure I.1 (SHERIDAN, 1985).

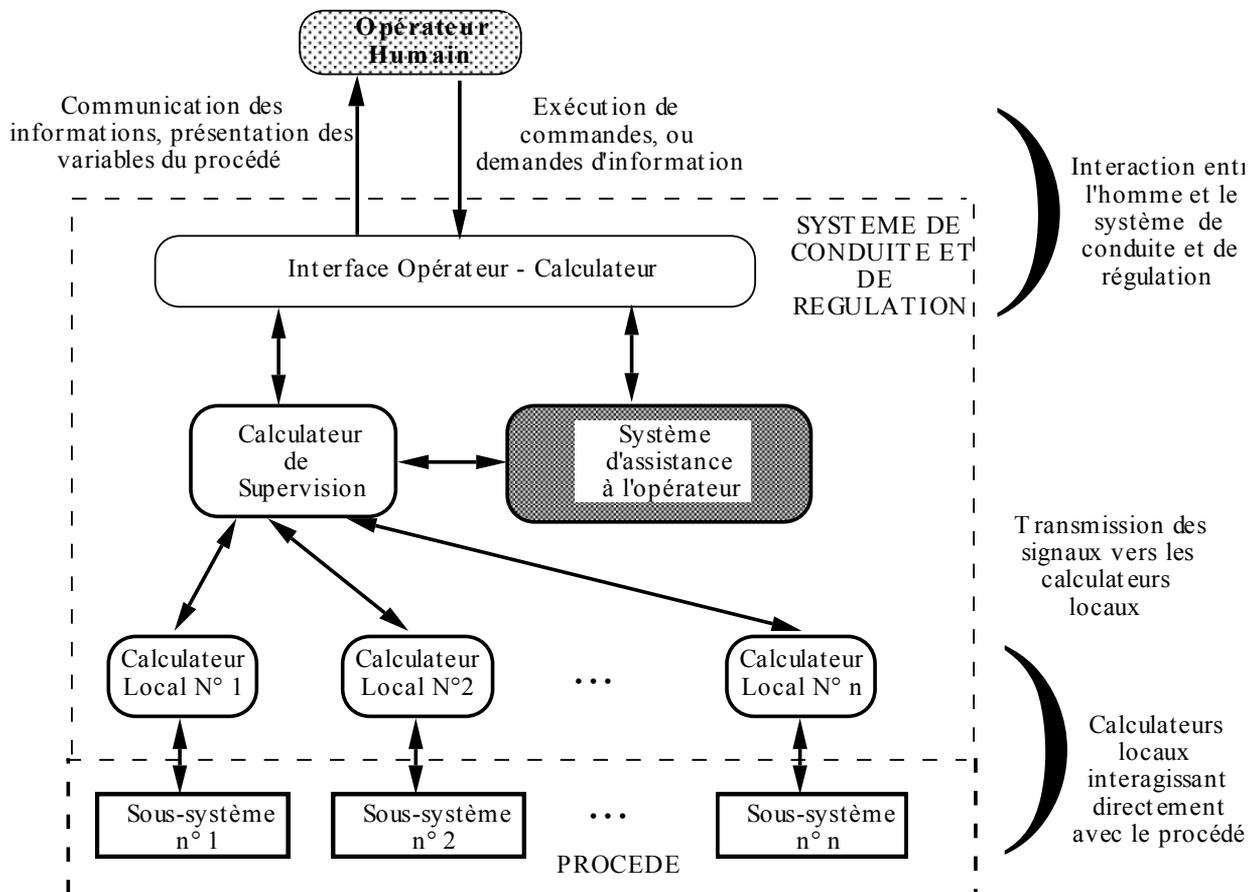


Figure I.1 : Architecture des systèmes de supervision adaptée de celle de SHERIDAN (1985)

Cette architecture, très générale, regroupe essentiellement le procédé, le système de conduite et de régulation et l'opérateur humain :

- Le procédé est composé généralement d'un ensemble de sous-systèmes. Par exemple, une centrale nucléaire est composée de plusieurs sous-systèmes tels que le circuit primaire, le circuit secondaire, le circuit d'approvisionnement en eau de refroidissement, etc.
- Le système de conduite et de régulation se compose généralement de calculateurs locaux, d'un calculateur de supervision, d'un ou plusieurs système(s) d'assistance à l'opérateur et d'une interface homme-machine :
 - *Les calculateurs locaux* sont chargés d'interagir avec chacun des sous-systèmes du procédé. Ils supportent notamment les différents algorithmes de régulation.
 - *Le calculateur de supervision* centralise un ensemble d'informations provenant des différents calculateurs locaux. Il se charge de la surveillance du fonctionnement par comparaison avec un modèle du fonctionnement du système globale. Il se charge aussi d'acheminer les différentes actions de l'opérateur vers chacun des sous-systèmes concernés, grâce aux différents calculateurs locaux, coordonnant ainsi les différents sous-systèmes en générant leurs consignes.
 - *Le(s) système(s) d'assistance à l'opérateur*, à partir de l'exploitation de différentes données centralisées au niveau du calculateur de supervision, génère(nt) de nouvelles informations. Ces informations peuvent prendre, par exemple, la forme de plans de reprise de défauts, de diagnostics, de conseils, etc.
 - *L'interface homme-machine* possède un rôle très important, car elle est chargée de sélectionner et de mettre en forme l'ensemble des informations provenant du calculateur de supervision ainsi que ceux provenant du système d'assistance.
- L'opérateur humain est le garant ultime de la sécurité du système. Son rôle ainsi que sa place dans le système de conduite seront expliqués progressivement dans chapitre.

Dans la partie suivante, nous détaillons globalement les besoins de l'opérateur humain en matière d'assistance. Ces besoins sont le résultat d'une analyse

bibliographique des caractéristiques, du rôle et de la fonction de l'opérateur humain dans les salles de contrôle de procédés industriels complexes.

II. LES BESOINS D'ASSISTANCE DES OPERATEURS DANS LES SALLES DE CONTROLE

Après avoir positionné les différents composants d'un système de supervision, il s'agit maintenant d'étudier dans un premier temps le rôle de l'opérateur humain dans les salles de contrôle de procédé industriels complexes. Puis, il importe d'analyser certaines caractéristiques de l'opérateur humain afin d'identifier les différentes aides à lui apporter.

II.1. ROLE DE L'HOMME DANS LA CONDUITE DE PROCEDES INDUSTRIELS

Il est très rare de rencontrer en 1992 des boucles de régulation en manuel, car ce que l'opérateur humain est capable de faire dans ce domaine, l'automatisme peut et doit le faire. L'opérateur devrait donc se consacrer quasi-exclusivement aux tâches intelligentes de résolution de problèmes.

A ce sujet, LEJON (1991), dans une étude sur le rôle futur de l'opérateur humain, envisage l'évolution de celui-ci selon quatre aspects : (1) assurer les régulations et les automatismes, (2) optimiser la marche du processus, (3) pallier les défauts de l'automatisme et (4) pallier les inévitables défauts du processus.

- Assurer les régulations et les automatismes. Ce rôle diminue régulièrement et doit disparaître à long terme, car, comme nous l'avons souligné précédemment ce que l'homme peut faire sous la forme de tâche "algorithmique", la machine doit le faire. Cependant, on constate que, bien que le potentiel algorithmique des systèmes numériques de contrôle commande (SNCC) soit considérable, celui-ci reste très sous-exploité dans la majorité des applications à cause de la très large utilisation des régulateurs PID (Proportionnel Intégral Dérivé). Le PID est utilisé, de nos jours, dans près de 94% des applications industrielles (LEJON, 1991)
- Optimiser la marche du processus. Ce rôle doit disparaître avec l'apparition des calculateurs, mieux adaptés aux traitements des calculs complexes

d'optimisation que le cerveau humain. De plus, les programmes d'optimisation, qu'ils soient algorithmiques ou à base de systèmes experts, peuvent désormais de plus en plus tenir compte, à la fois de l'expertise issus des opérateurs et des ingénieurs du procédé (sous la forme de règles de savoir faire typiquement humaines) et de différentes formes de modélisation des connaissances à mettre en œuvre (BENKIRANE et al., 1990).

- Pallier les défauts de l'automatisme. C'est un rôle très important, mais qui va en s'amenuisant avec l'introduction de redondances matérielles et logicielles à tous les niveaux. Ce rôle diminue donc au fur et à mesure que la fiabilité des systèmes s'améliore. Notons à ce sujet que de gros efforts sont entrepris dans ce domaine, depuis son introduction par CHAPOUILLE en 1972, faisant suite à un transfert important de technologie dans les systèmes de production, dans les années 70 (GUILLEVIC, 1990).
- Pallier les inévitables défauts du processus. C'est sans doute l'unique et le principal rôle qui restera pour l'opérateur humain en salle de contrôle. En effet, les automatismes sont incapables de résoudre ou simplement de prendre en compte l'ensemble des inévitables défauts du processus, et à ce sujet, il est évidemment impossible de rendre tout les processus redondants. Cependant, pour éviter des arrêts fréquents des installations, dont le coût est considérable, on cherche à rendre acceptables des régimes de marche dégradés, pour lesquels on préfère produire moins ou un produit de qualité moindre plutôt que d'arrêter le processus. Dans de telles situations, la prise en compte des petits défauts du processus, par l'opérateur humain, restera l'une des tâches principales de celui-ci, et cela pour quelques années encore.

Nous venons de citer les rôles de l'opérateur humain qui sont concurrencés progressivement par les technologies nouvelles telles que les SNCC. Cependant, la réalisation des objectifs de production ne peut être assurée exclusivement par des automatismes, étant donnée la limite des modèles de comportement des processus. DE TERSSAC et CHABAUD (1990) distinguent à ce niveau trois types de fonctions qui se combinent pour orienter le rôle de l'opérateur humain :

- La fonction de réalisation de tâches explicites. Le rôle de l'opérateur humain se traduit, dans ce cas, par des essais périodiques, des manœuvres exceptionnelles, l'arrêt et le démarrage des installations, etc.

- La fonction de gestion d'événements non prévus. Le rôle de l'opérateur consiste essentiellement ici à assurer la continuité de la production dans le respect des impératifs de sécurité, par le traitement de tous les événements contredisant le déroulement de la production. C'est le cas, par exemple dans les centrales d'énergie où, dans certaines conditions, l'opérateur de dispatching d'électricité doit faire face à un déséquilibre entre l'intensité et la capacité de production des centrales françaises. Ce déséquilibre peut provenir d'un accroissement de la demande en raison par exemple des conditions climatiques ou d'une baisse de production suite à la mise hors service d'une unité de production.
- La fonction de gestion des tâches implicites dans les instructions. C'est le cas où l'opérateur doit exécuter un schéma d'exécution pré-établi par d'autres personnes, les experts du processus en l'occurrence. Ce schéma représente les actions à entreprendre face à une situation donnée. Mais, la formalisation du schéma est plus ou moins complète et plus ou moins précise, d'où le caractère implicite de la fonction de l'opérateur, qui doit alors ajuster de lui-même les défauts du formalisme, tant en matière de précision qu'en matière de lacunes. La qualité de tels ajustements dépendra alors de son expérience et de ses connaissances.

Dans le souci de mieux aider l'opérateur dans les différents rôles qui lui sont attribués, des recherches visent à mieux cerner ses caractéristiques. Certaines d'entre elles sont développées dans la partie suivante.

II.2. CARACTERISTIQUES DE L'OPERATEUR HUMAIN

L'opérateur humain, élément évidemment essentiel dans un système homme-machine, est au centre de nombreuses études. Ces études sont entreprises particulièrement par des chercheurs issus des sciences cognitives. Ceux-ci tentent d'en définir les caractéristiques et il importe que les concepteurs d'outils d'assistance puissent les considérer avec attention. D'après DE KEYSER (1987), Il existe plusieurs manières de caractériser l'opérateur humain : comme un système de communication, comme un régulateur, comme un décideur statique ou encore comme un processeur d'information :

- L'homme comme un système de communication. Dans ce cas, l'homme est considéré comme un système dont le rôle est de coder et décoder l'information. Cette caractérisation de l'homme est la plus ancienne qui existe. En effet, elle est issue des travaux de SHANNON et WEAVER (1949) sur la théorie du traitement de signal. Ces auteurs notent cependant que l'homme a une capacité limitée. Cette façon de considérer l'opérateur humain trouve très vite ses limites, car elle exclut les aspects subjectifs contenus dans l'information et plus particulièrement la signification de l'information (DE KEYSER, 1987). Cette caractérisation a été exploitée lors de différents travaux visant la modélisation d'un opérateur humain, tels ceux relatifs au modèle MESSAGE (TESSIER, 1984 ; BOY, 1986) qui sera commenté dans le paragraphe II.3, ainsi que dans le chapitre suivant, paragraphe VIII.3).
- L'homme comme un régulateur. L'opérateur est considéré, dans ce cas, comme un servo-mécanisme. Il ajuste des conduites pour réduire un éventuel écart entre l'objectif assigné et l'objectif réalisé (FAVERGE, 1972). L'opérateur humain doit, par exemple, ajuster l'ouverture d'une vanne, pour combler les écarts par rapport à la consigne d'une variable du processus. Notons ici que ce type de caractérisation de l'opérateur humain n'existe presque plus car comme LEJON (1991) le suggère, cette tâche est souvent relativement facile à automatiser.
- L'homme comme décideur. Cette caractéristique sera toujours présente chez l'opérateur humain. En effet, dans des conditions particulières, l'opérateur doit identifier les différentes alternatives possibles en termes de solution à un problème. Par la suite, il doit évaluer le coût de chaque solution. Ce coût peut être évalué de façon très différente d'un opérateur à l'autre. Enfin, il doit se décider sur le choix d'une solution ou d'une action. Cette caractérisation est à la base de certains travaux de modélisation de l'opérateur humain, tels ceux de CACCIABUE et al. (1990) ou encore STASSEN (1986).
- L'homme comme processeur d'information. Une certaine analogie entre l'homme et l'ordinateur en tant qu'entités traitant de l'information est à l'origine de cette caractérisation de l'opérateur humain. Cette analogie a été notamment développée par ROUSE (1980) dans des modèles "de files d'attente" où il compare l'homme à un ordinateur traitant de l'information en temps partagé, c'est-à-dire capable d'exécuter plusieurs tâches en

échantillonnant son attention sur chacune d'elles. CARD et al. (1983) s'inspirent de cette caractérisation de l'opérateur pour définir le modèle KEYSTROKE.

Dans un contexte de contrôle d'un procédé industriel, on peut mettre en avant sans souci d'exhaustivité certaines caractéristiques de l'opérateur humain, à considérer avec attention :

- Le travail en séquence. Les opérations de lecture d'instrumentations, d'interprétation de ces lectures et d'élaboration des stratégies de reprise se font successivement et non de façon parallèle (ROUHET, 1988). Ainsi, afin d'aider l'opérateur, des systèmes de synthèse d'information peuvent être mis à sa disposition. Nous pouvons citer à ce sujet le système décrit par ROUSE et al. (1982) : celui-ci assiste un pilote effectuant une procédure d'action, en lui présentant simultanément sur un écran alphanumérique l'ensemble des informations qu'il doit considérer, facilitant ainsi ses opérations de traitement d'information.
- Le besoin constant d'information. Le cerveau humain ressent très mal l'absence d'information sur son écran de contrôle, et de ce fait, il cherche à en recueillir par tous ses capteurs disponibles (WANNER, 1987).
- La compensation de l'augmentation de difficulté d'une tâche par une augmentation de l'activité humaine. Lors de l'exécution de sa tâche, l'opérateur humain investit une fraction de sa capacité de travail pour faire face à la charge de travail induite par cette tâche. Or toute augmentation de la charge de travail se traduit par une augmentation de l'activité humaine ou par la modification des modes opératoires, ceci afin d'éviter de saturer la capacité du canal humain (MILLOT, 88).
- La possession d'une représentation mentale. L'opérateur raisonne à partir d'une représentation mentale personnelle du système qu'il supervise ; il utilise un modèle mental associant les réponses que le système fournit aux actions qu'il reçoit (voir à ce sujet OCHANINE, 1971 et WANNER, 1987). Il importera donc que l'interface homme-machine lui facilite la construction de ce modèle mental.

- L'incapacité de l'homme d'estimer les risques encourus dans une situation donnée. Dans certains cas, en essayant d'augmenter la rentabilité de son système, l'opérateur en arrive à transgresser des consignes de sécurité ou à modifier leurs règles d'emploi en réduisant leurs marges. La mise en place d'un système pouvant prédire l'évolution du processus, même si la prédiction n'est pas parfaite ou incomplète, peut aider l'opérateur dans ce type de situation (VAN DER VELDT et VAN DEN BOOMGAARD, 1986).

On tend de plus en plus à constater que "moins nombreux sont les hommes sur une installation automatisée, plus important est leur rôle et plus lourdes de conséquences sont leurs décisions et leurs actions sur l'efficacité du système" (DECOSTER, 1988). Comme on l'a souligné, les tâches de l'opérateur dans un système de plus en plus automatisé sont devenues essentiellement des tâches mentales. Celles-ci font l'objet de nombreuses études, et des chercheurs s'intéressent particulièrement aux nécessités suivantes : comprendre, surveiller, apprendre, interpréter, planifier et intervenir (HOC, 1990 ; WANNER, 1987, etc). Elles sont résumées sur la figure I.2.

- L'opérateur doit comprendre chaque information qui lui est présentée, son codage, sa signification ainsi que le contenu informationnel qu'elle apporte. Il est donc nécessaire d'adapter l'imagerie, et en particulier les synoptiques de l'interface homme-machine, afin de permettre à l'opérateur de mieux prendre conscience du processus physique et par voie de conséquence de mieux le maîtriser (LEJON, 1991). Notons que si l'opérateur *comprend* une situation donnée, il va acquérir une certaine connaissance de cette situation, ce qui lui permet *d'apprendre*.
- Il a à surveiller toutes les variations qui s'opèrent au niveau des variables caractéristiques du procédé à superviser. Afin d'aider l'opérateur dans ce type de tâche, une hiérarchisation des alarmes est par exemple nécessaire. Dans ce contexte, ANDOW (1980) décrit un système capable de présenter à l'opérateur l'ensemble des alarmes actuelles en les regroupant en plusieurs sous-ensembles disjoints, ces sous-ensembles étant classés selon le degré de gravité engendré par l'alarme. De plus, chaque sous-ensemble comprend en tête de liste l'alarme principale, puis l'ensemble des alarmes qui en sont la conséquence. Ce type d'assistance est connu sous le nom de système de filtrage d'alarme. En *surveillant* les paramètres du processus, l'opérateur doit

pouvoir facilement *interpréter* les différentes informations qui lui sont présentées, ce qui lui permettra notamment de *comprendre* la situation dans laquelle fonctionne le processus.

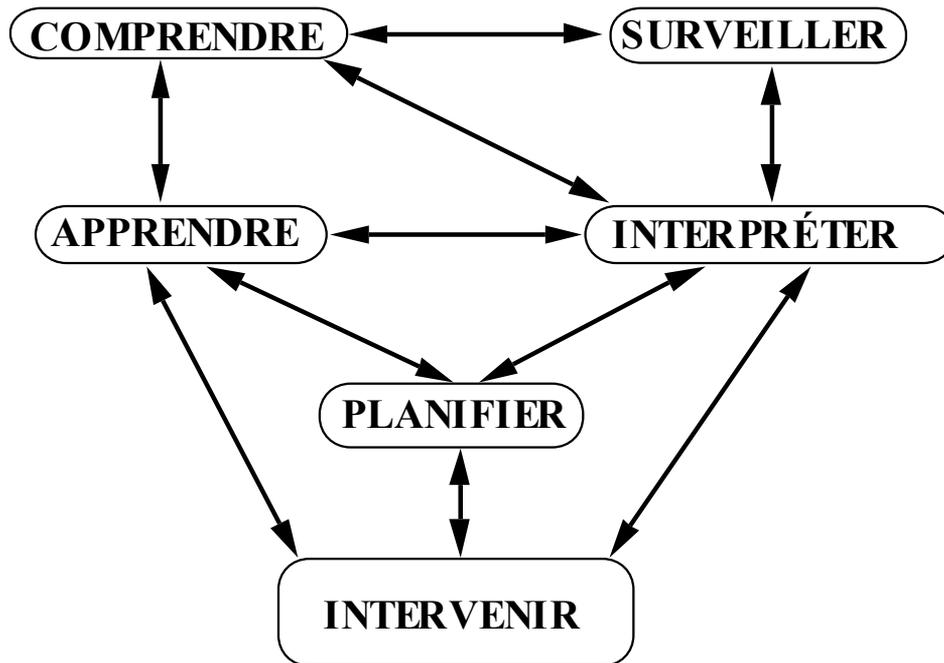


Figure I.2 : Enchaînement des tâches mentales de l'opérateur humain, inspiré des travaux de RASMUSSEN (1986)

- L'opérateur doit apprendre de nouvelles procédures de correction et de nouvelles situations de fonctionnement du procédé, afin de devenir progressivement un expert dans ce domaine. Il apprend en situation réelle ou lors de séances de formation. A ce sujet, DANIELLOU (1986) passe en revue plusieurs approches de formation, notamment :
 - La formation / Conception. Cette approche repose sur l'analyse par les opérateurs eux-mêmes de leur travail futur sur les installations. Ils peuvent ainsi élaborer des questions auxquelles les responsables techniques devront pouvoir répondre par des actions de formation.
 - La formation modulaire. Les opérateurs apprennent, de façon modulaire, les différents symptômes de bon ou de mauvais fonctionnement des différentes parties de l'installation.

- Les connaissances fondamentales. La formation se base sur la maîtrise de connaissances physiques fondamentales. Les opérateurs apprennent ainsi à maîtriser les lois régissant certains phénomènes physiques tels que les échanges de chaleur.
 - Le travail sur simulateur. C'est peut être la méthode la plus employée dans le cas où le processus est modélisable et donc simulable. C'est notamment le cas de la formation des pilotes d'avions. Cette méthode permet de simuler, sans risque réel, des situations incidentelles particulières.
 - La formation "sur le tas". L'opérateur qui débute est mis en double avec un opérateur expérimenté. C'est la méthode la plus utilisée dans le cas où le processus n'est pas modélisable, ou lorsque le travail est lié au savoir faire typiquement humain.
- Il importe que l'opérateur puisse interpréter toute alarme ou tout dépassement de seuil d'une variable afin d'identifier la situation dans laquelle fonctionne le procédé. Après avoir *interprété* une situation, l'opérateur peut apprendre celle-ci, en vue de répondre à une nouvelle situation, ou chercher à comprendre les causes de cette situation. Il peut ainsi planifier les actions compensatrices ou correctrices puis intervenir sur le processus.
- L'opérateur doit planifier des procédures de reprise suite à l'apparition d'un défaut dans le fonctionnement du procédé. Cette planification nécessite une bonne connaissance du procédé. Il existe de plus en plus des systèmes d'assistance à la planification de tâches capables de proposer à l'opérateur un diagnostic de la situation, complété par des conseils de correction sous forme de plans d'action, par exemple le système ESCORT (SACHS et al., 1986) ou le système d'assistance réalisé lors du projet ALLIANCE (CALOUD et al., 1988) qui effectuent un diagnostic de panne et proposent à l'opérateur des conseils pour corriger la situation.
- Une fois le défaut identifié et le plan d'action choisi ou établi, l'opérateur doit intervenir sur le procédé. Cette intervention doit en principe ramener le système à superviser vers une situation normale. Mais, si l'opérateur est insuffisamment expérimenté ou si le plan d'action qu'il a établi ou choisi est incomplet, la situation peut se dégrader.

On retrouve l'ensemble de ces notions dans plusieurs approches actuelles de modélisation de l'opérateur humain en salle de contrôle, faisant l'objet de la partie suivante.

II.3. APPROCHES DE MODELISATION DE L'OPERATEUR HUMAIN

Pour rendre l'assistance à l'opérateur meilleur, il est nécessaire d'aménager ergonomiquement le poste de travail de l'opérateur. Cependant, si certains aménagements peuvent être identifiés de façon empirique, ce n'est pas le cas dans des tâches complexes à forte composante mentale (MILLOT, 1988). En effet, celles-ci nécessitent un aménagement ergonomique du logiciel concernant, notamment, le mode de présentation, la sélection et la cadence des informations échangées en fonction des tâches à exécuter et du contexte opérationnel du procédé. L'ergonome doit avoir donc une connaissance ou un modèle des processus mentaux impliqués dans les différentes activités induites par la tâche. Il doit aussi savoir comment ces processus mentaux s'enchaînent les uns par rapport aux autres. De plus, il faut comprendre ce que l'opérateur est en train de faire afin de pouvoir, éventuellement, corriger certaines des erreurs humaines. Pour toutes ces raisons, des études sont activement menées. Le but de celles-ci est de définir des modèles normatifs de comportement d'un ensemble d'opérateurs, dans des contextes bien particuliers, car sans les considérations du facteur humain, l'utilisation de tout système va nécessiter un temps d'apprentissage ou d'entraînement des utilisateurs très important et peut conduire, dans certains cas limites, à une non utilisation du système qui sera alors abandonné (DE KEYSER, 1988 ; WALSH et al. 1989).

En résumé, on part d'une hypothèse de comportement de l'opérateur (une norme) et on recherche un outil permettant de représenter, voire de simuler, ce comportement (MILLOT, 1988).

Ainsi, dans un but final de conception de systèmes d'assistance adaptés aux besoins de l'opérateur, il est nécessaire d'étudier le comportement décisionnel de celui-ci et de parvenir à la définition de modèles représentatifs. Des recherches se sont donc orientées ces dernières années vers des modèles cognitifs. Sans souci d'exhaustivité, nous résumons dans ce paragraphe les principes de plusieurs approches de modélisation.

Le modèle qualitatif de résolution de problème de Rasmussen (1980), figure I.3, a constitué une ouverture très importante dans la modélisation de l'opérateur humain. Ce modèle décrit le comportement d'un opérateur impliqué dans des tâches de supervision et de résolution de problèmes. Il propose quatre étages décisionnels : la détection d'un événement anormal suite à l'apparition d'une alarme ou à la détection d'un certain seuil (A1), l'évaluation de la situation afin d'identifier l'état du procédé qu'il supervise (A2, A3 et A4), la prise de décision (A5, A6) et l'exécution des actions résultantes (A7). On remarque sur la figure l'existence de raccourcis lors de la résolution de problèmes.

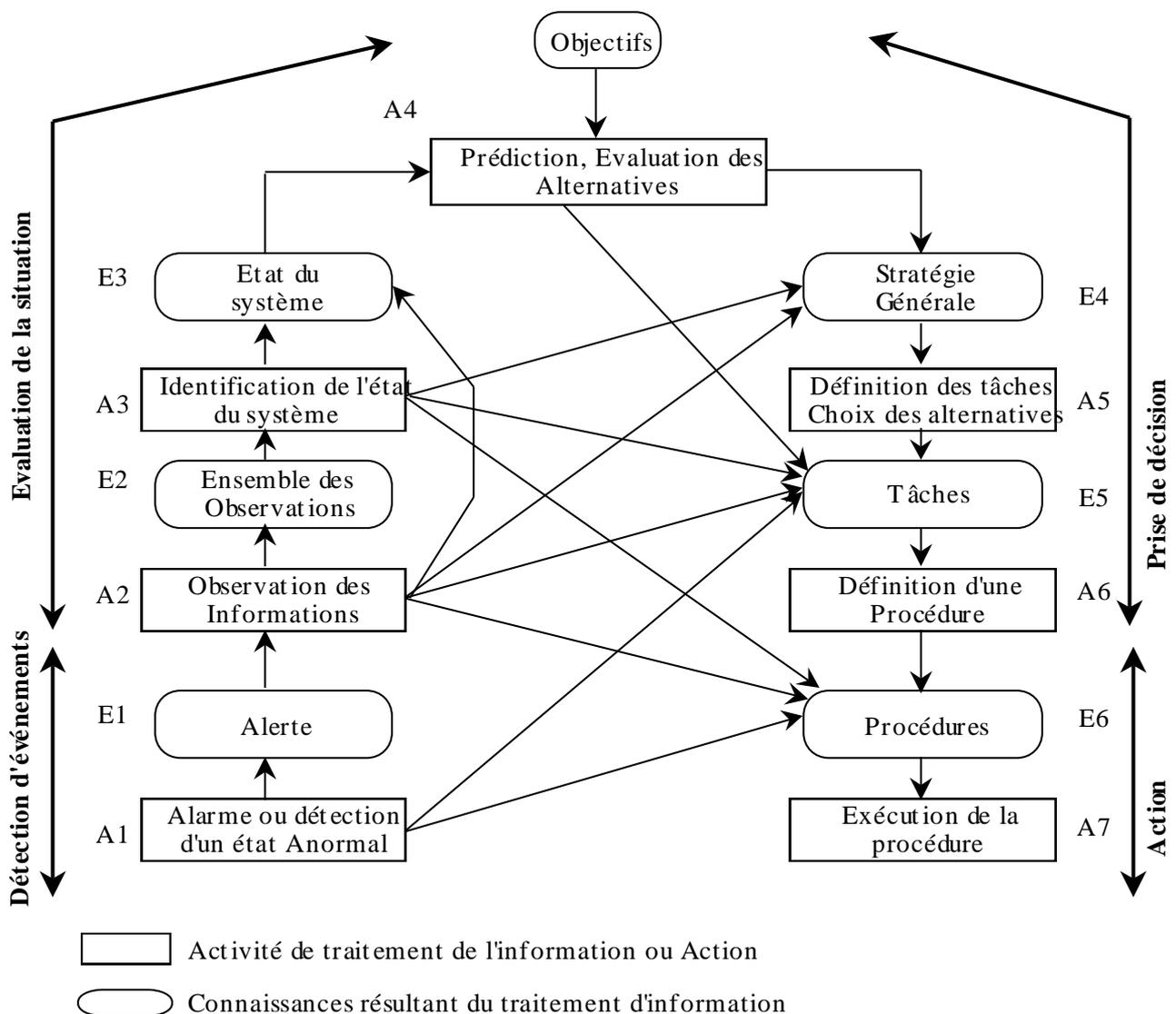


Figure I.3 : Modèle conceptuel de résolution de problème (RASMUSSEN, 1980)

Partant de ce modèle, RASMUSSEN (1983) définit pour l'opérateur humain trois types de comportements : (1) un comportement basé sur l'habileté ; (2) un comportement basé sur les règles et (3) un comportement basé sur la connaissance, figure I.4.

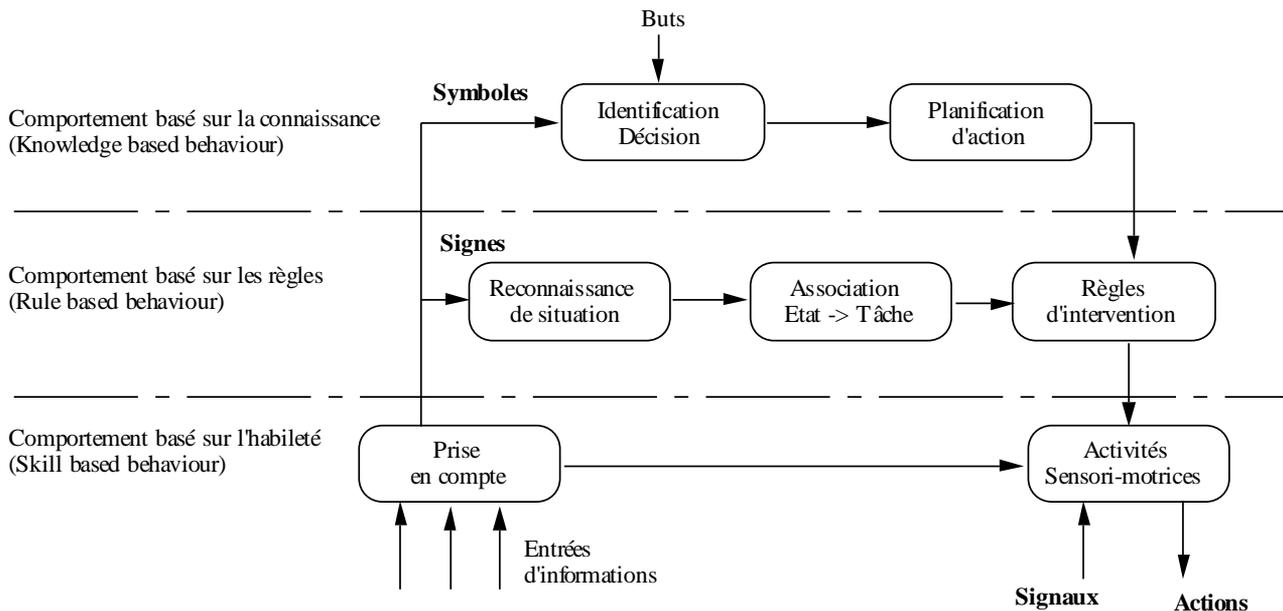


Figure I.4 : Les trois types du comportement de l'opérateur humain d'après RASMUSSEN (1983)

- Comportement basé sur l'habileté (Skill Based Behavior). L'opérateur exécute de façon quasi-réflexe des actions correctrices en réponse à des informations perçues sous forme de signaux. Dans ce cas, l'opérateur, expérimenté en général, associe directement les actions aux signaux perçus.
- Comportement basé sur les règles (Rule Based Behavior). L'opérateur, face à des situations familières, réagit par ce comportement, qui le conduit à l'application d'un certain nombre de règles pour identifier la situation courante du procédé qu'il supervise. Il adopte alors directement des actions ou des procédures de reprises pré-établies. L'opérateur dans ce cas réagit à des signes pratiques.
- Comportement basé sur la connaissance (Knowledge Based Behavior). L'opérateur adopte, en dernier recours, ce comportement lorsqu'il est face à des situations nouvelles ou imprévues. Il doit ainsi émettre des hypothèses,

les vérifier et prédire l'évolution du procédé en vue d'éventuelles actions de correction. Dans ce cas, l'opérateur réagit suite à des symboles qualitatifs caractérisant les différents éléments considérés et se référant à son propre modèle interne.

De nombreux travaux de modélisation de l'opérateur humain sont apparus depuis. Parmi ces travaux, certains ont abouti à l'élaboration d'un modèle d'opérateur, dans des conditions bien ciblées, par exemple :

- **MESSAGE** (TESSIER, 1984) modélise à la fois plusieurs systèmes impliqués dans les tâches de pilotage d'un avion ou de gestion de vol lors d'une mission simulée. Il modélise les traitements d'information de chaque sous-système (pilote, copilote, avion piloté et contrôle au sol) et les différents transferts entre ces sous-systèmes. Ce modèle visait deux objectifs majeurs : délivrer les taux d'occupation de chacun des canaux (visions centrale et périphérique, audition, parole, mains, pieds) du pilote et du copilote et fournir les indices statiques de charge induits par les afficheurs et actionneurs du poste de pilotage.
- **KARL** (KNÆUPER et ROUSE, 1985) est un système à base de connaissances qui simule, à l'aide de règles de production, le comportement humain dans les tâches de résolution de problème rencontrées dans le contrôle et la supervision d'un procédé dynamique, telles que la détection de défauts, le diagnostic et la reprise d'un défaut et le suivi d'action. L'originalité de KARL est qu'il ne vise pas à simuler les performances optimales mais à reproduire celles de l'opérateur, bonnes ou mauvaises, compte tenu des ses connaissances, de façon à mieux connaître les types d'erreurs humaines. Des comparaisons ont été faites entre les performances obtenues par quatre groupes d'opérateurs (de qualifications différentes) et KARL, dans des tâches de surveillance d'un procédé simulé. Les résultats de ces comparaisons ont montré que le modèle duplique très bien les performances globales de chacun des 4 groupes d'opérateurs mais sans appliquer tout à fait les mêmes séquences d'actions. Les auteurs expliquent en partie ce demi-échec par les fortes différences inter-individuelles constatées dans chaque groupe d'opérateurs. En conséquence, cette approche, très attractive, de modélisation du comportement humain par un système à base de connaissance se heurte au délicat problème de l'extraction des connaissances de l'opérateur à simuler,

ceci est dû à deux raisons essentielles : (1) l'opérateur ne dit pas toujours ce qu'il fait réellement, (2) il sait ce qu'il doit faire mais ne le fait pas toujours.

- **COSIMO** (CACCIABUE et al., 1990) est issu de travaux menés au centre de recherche européen de l'ISPRA. Ces travaux visent l'étude et la réalisation d'un modèle de comportement d'opérateurs de conduite de procédé automatisé. L'architecture du modèle d'opérateur humain est basé sur deux niveaux cognitifs de raisonnement et de prise de décision : (1) le niveau "HAUT" permet d'exploiter les connaissances de l'opérateur en reconnaissant, en continu, les différentes situations et en construisant des stratégies de surveillance et de contrôle, (2) le niveau "BAS" permet l'accomplissement d'une réponse pré-programmée ou d'une stratégie planifiée dans le but de satisfaire une intention clairement définie. Ce modèle permet actuellement d'étudier le comportement de l'opérateur face à des situations particulières de conduite. Cette approche de modélisation est implantée en un module cognitif basé sur des techniques résultant de l'Intelligence Artificielle Distribuée. En effet, ce module est implanté dans une architecture "Blackboard" (NII, 1986), le blackboard consistant en une zone de données communes où plusieurs agents, sources de connaissances, écrivent ou lisent dans le but de trouver une solution par coopération, à l'aide d'un système de contrôle. Ce modèle a fourni des premiers résultats encourageants concernant l'étude du comportement de l'opérateur face à des situations d'urgence. En effet, COSIMO vise à modéliser les trois types de comportement identifiés par RASMUSSEN (1983). Il se compose de 4 modules principaux assurant les fonctions suivantes :

- Le filtrage des informations issues de l'environnement externe (le procédé) simule la sélection des informations selon deux critères : la "salience physique" reflète les propriétés des informations observées, et la "salience cognitive" exprime la pertinence de l'information pour la tâche en cours.
- Le diagnostic est destiné à proposer des explications au problème rencontré et fournit des hypothèses à ce sujet. Cette activité est simulée par un système à base de connaissance selon un mécanisme dirigé par les données.

- La confirmation des hypothèses, c'est à dire leur vérification, est ensuite effectuée pour générer des symptômes fournis par le procédé. Cette activité est également simulée par un système à base de connaissance mais selon un mécanisme dirigé par les buts. A ces symptômes sont associées les actions correctrices dans la base de connaissance.
- Les actions résultantes sont enfin appliquées sur le procédé par le module d'exécution.

Dans l'état actuel, COSIMO ne permet toutefois pas de modéliser le comportement intelligent (Knowledge Based Behavior).

Notons que les techniques de l'Intelligence Artificielle Distribuée, en particulier de résolution distribuée de problème, sont de plus en plus mises à contribution pour la modélisation du comportement de l'opérateur humain. Par exemple, une autre application basée sur une architecture "Blackboard", en l'occurrence, le modèle **OFMspert** (RUBIN et al., 1988) peut être cité à cet effet. Ce modèle tend à modéliser les intentions de l'opérateur dans une architecture de type blackboard.

En conclusion, les efforts menés dans ce domaine sont considérables mais restent insuffisants. Toutefois grâce au développement de nouvelles techniques, telles que les réseaux neuronaux, d'autres portes se sont entrouvertes vers la modélisation du comportement de l'opérateur humain (BERSINI, 1989).

Dans le but de faciliter le travail de l'opérateur, il s'agit maintenant d'étudier les possibilités d'assistance informatique aux tâches humaines citées précédemment. Le paragraphe suivant classifie, à cet effet, différentes approches d'assistance.

III. CLASSIFICATION DES SYSTEMES D'ASSISTANCE

Le champ de recherche consacré aux problèmes de l'assistance effective à apporter à l'opérateur humain dans les salles de contrôle de procédés industriels est très vaste. La tendance actuelle consiste à les assister d'outils faisant généralement appel à des techniques de l'Intelligence Artificielle tels que les systèmes experts (BOY, 1988A). En effet, ces systèmes sont capables de s'adapter à un grand nombre de situations de conduite, en faisant appel à des heuristiques modélisées dans leurs bases de connaissance. De nombreux chercheurs travaillent dans le domaine de l'assistance à l'opérateur, et il n'est pas possible de décrire tous les travaux existants. Notre objectif est ici de rappeler certains principes de base relatifs à l'assistance à l'opérateur, et ceci afin de mieux situer nos travaux. Pour aller plus loin dans le domaine de l'assistance à l'opérateur, le lecteur se référera par exemple aux travaux de BOY (1988A; 1988B, 1990), SHALIN et ses collègues (1988 ; 1990), AMALBERTI et ses collègues (1991 ; 1992) ou encore ROTH, BENNETH et WOODS (1988).

Cependant, quel que soit l'outil d'assistance développé, il doit répondre à un certain nombre de critères tels que :

- Etre capable de guider l'opérateur de façon fiable vers les problèmes à résoudre, en synthétisant les informations pertinentes (TABORIN, 1989).
- Aider les opérateurs dans les situations d'urgence, en leur fournissant des informations sur l'état de fonctionnement du processus et en leur donnant des conseils d'actions.
- Favoriser la capacité de prévision des opérateurs (ROUSE, 1973), en mettant à leurs disposition des affichages prédictifs, même si la prédiction n'est pas parfaite (VAN DER VELDT et VAN DEN BOOMGAARD, 1986),
- etc.

La liste des systèmes d'assistance, qui tendent à être implantés progressivement dans les salles de contrôle de procédés industriels, s'allonge de jour en jour. Il s'agit maintenant de recenser et classer ces différents types de systèmes d'assistance. L'implantation de ces systèmes d'assistance dans les salles de contrôle doit conduire

à une réorganisation des tâches de l'opérateur et du calculateur, afin d'aboutir à une meilleure coopération entre eux, une des premières phases d'implantation de ces systèmes d'assistance étant consacrée au mode d'intégration de ces systèmes dans le système global de supervision (TABORIN, 1989).

A cet effet, FININ et KLEIN (1987) proposent une classification de différents systèmes d'assistance, tableau I.1, selon leurs possibilités d'intégration dans les systèmes de supervision.

Quatre classes de systèmes, les consultants, les servants, les agents et les contrôleurs sont successivement décrites. Puis, deux classes supplémentaires sont proposées.

Tableau I.1 : Classification des systèmes d'assistance en fonction de leur mode d'intégration, adaptée de FININ et KLEIN (1987)

Les entrées du système d'assistance proviennent Les résultats sont communiqués	de l'Opérateur Humain	du Procédé
à l'Opérateur humain	Les Consultants	Les Contrôleurs
au Procédé	Les Servants	Les Agents

III.1. LES CONSULTANTS

Les "consultants" sont des systèmes généralement non connectés au procédé, figure I.5, auxquels l'opérateur transmet toutes les données dont ils ont besoin afin qu'ils puissent lui fournir, par exemple, des conseils d'action ou un diagnostic. Dans ce cas, les techniques des systèmes experts ont montré leur efficacité. Les

consultants fonctionnent en général suivant une approche dite statique. En ce sens, le système raisonne à partir d'une description initiale du procédé et de connaissances pré-enregistrées, il pose des questions à l'opérateur lorsque le raisonnement est bloqué par un manque d'information.

Dans cette classe d'outils, nous retrouvons par exemple le système COCCINELLE (HOURIEZ et al., 1988), qui est un système expert d'aide au diagnostic des causes de défauts en coulée continue d'acier. Ce système base son raisonnement sur des symptômes introduits par l'opérateur humain. Il peut ainsi délivrer un rapport détaillé du diagnostic.

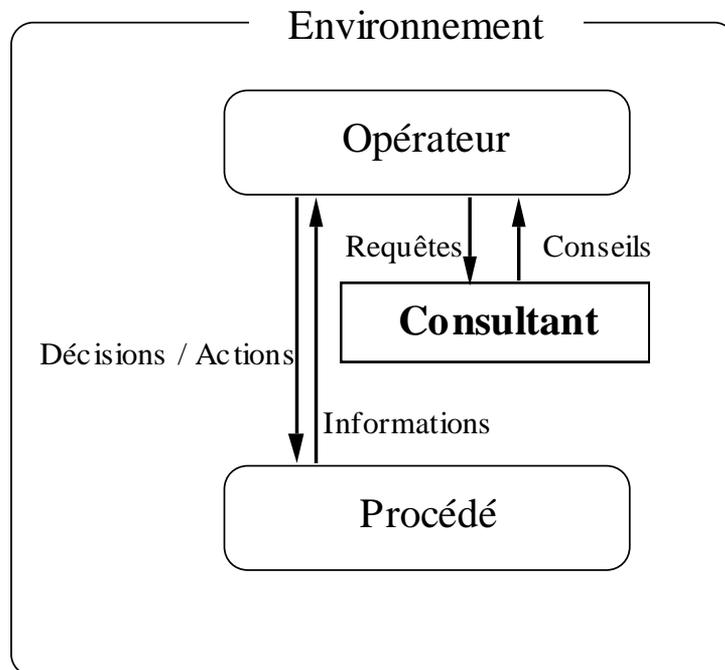


Figure I.5 : Positionnement des "consultants" dans le système homme-machine

Dans cette classe de système d'assistance, les bases de connaissance contiennent soit un modèle du procédé en question soit un modèle des traitements à effectuer sur le procédé sous la forme d'une base de données, soit sous la forme de règles d'expertise. C'est le cas notamment du système expert SE-DIAG qui regroupe ces deux classes de modèles (MARRAKCHI, 1986). Le but recherché lors de la réalisation de ce système, est la possibilité d'une adaptation, la plus aisée possible, à toutes les machines de production electro-hydrolico-mécanique. A cet effet, SE-

DIAG est doté de quatre niveaux de connaissance, les 2 premiers modélisent le procédé, les deux suivant modélisent les savoir-faire humains de diagnostic :

- Le premier niveau constitue le modèle fonctionnel de chaque composant du système. Il s'agit des modèles de référence que possède chaque dépanneur, qui lui permettent de choisir les tests les plus judicieux pour prouver la défaillance d'un composant.
- Le deuxième niveau correspond à une représentation structurelle des composants du système. Cette partie de connaissance constitue l'aspect spécifique du système relativement à un matériel particulier.
- Le troisième niveau consiste en la description d'une stratégie générale d'analyse reflétant la méthodologie d'un dépanneur. Il s'agit de décrire un mode de déplacement dans la structure de la machine pour focaliser progressivement vers l'origine d'un défaut. Ce niveau débouche sur une base de connaissances contenant des règles de production.
- Enfin, le dernier niveau concerne une description du savoir-faire de diagnostic sur la machine, c'est-à-dire la connaissance spécifique sur les dysfonctionnements et les symptômes associés. Cette description permet de focaliser plus rapidement vers l'origine du défaut.

Pour ce type de système d'assistance, l'interface homme-machine prend une part très importante, à cause notamment des aspects de saisie d'informations nécessaires au diagnostic.

III.2. LES SERVANTS

Les "servants" sont des systèmes connectés directement au procédé, figure I.6. Ils sont capables d'effectuer des commandes directes sur le procédé à partir de données provenant de l'opérateur. Dans cette classe, l'opérateur joue le rôle d'assistant du calculateur.

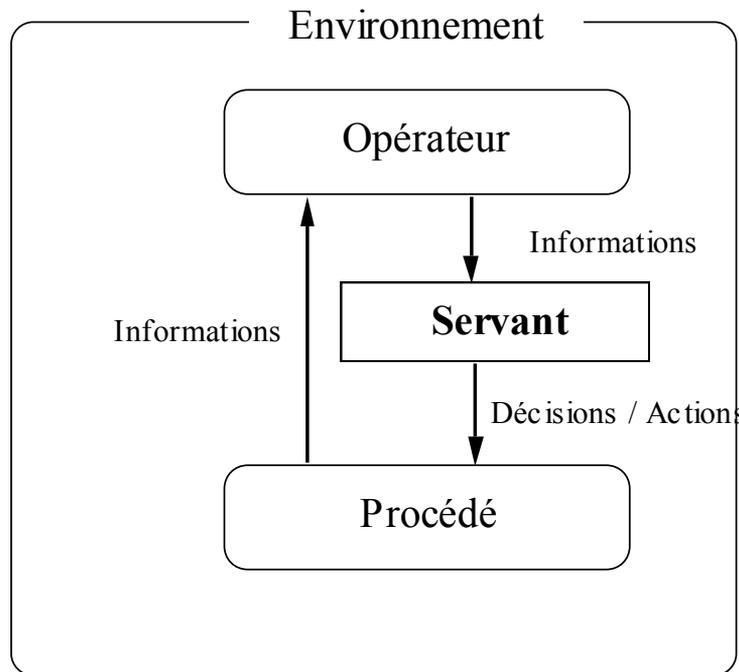


Figure I.6. Positionnement des "servants" dans le système homme-machine

Il n'existe pas encore, à notre connaissance, de système d'assistance opérationnel appartenant à cette classe. Néanmoins, nous pouvons classer dans cette catégorie, les systèmes dits "tolérants aux erreurs humaines". Ces systèmes reçoivent les commandes provenant de l'opérateur humain et traitent celles-ci afin de détecter celles dont les conséquences touchent à la sécurité ou la production du système. Dans ce cas, les commandes néfastes sont remplacées par d'autres commandes pré-établies, ou tout simplement refusées. Nous pouvons citer les travaux entrepris dans ce domaine par ROUSE et MORRIS (1985) ou MALVACHE (1990). Les systèmes "tolérants aux erreurs humaines" seront détaillés dans le chapitre suivant.

Dans les systèmes d'assistance de type "servant", l'interface homme-machine doit faciliter et optimiser les actions de l'opérateur. Mais celui-ci doit de plus lui permettre de pouvoir suivre l'évolution des variables du procédé et de comprendre les modifications éventuelles apportées à ses actions.

III.3. LES AGENTS

Les "agents" sont des systèmes "intelligents" d'auto-contrôle, connectés directement au procédé. Ils sont en principe capables de prendre des décisions et d'appliquer les actions correspondantes sur le procédé, figure I.7. La boucle de

contrôle du système est fermée en excluant l'opérateur humain, qui n'a donc qu'un rôle de superviseur à jouer. Ce dernier n'intervient donc qu'en cas de situation inhabituelle que l'agent n'est pas capable de prendre en compte.

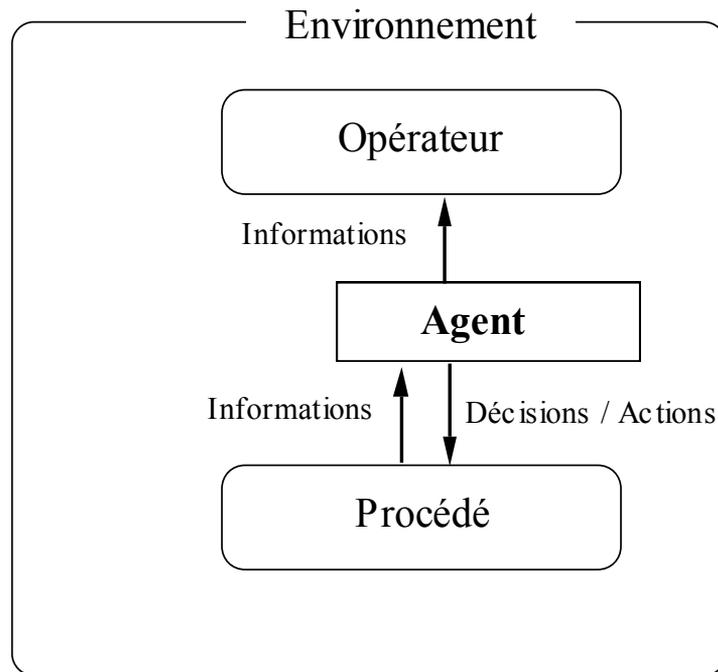


Figure I.7. : Positionnement des "agents" dans le système homme-machine.

Pour agir directement sur le procédé, un système de cette classe doit donc avoir une connaissance très complète sur celui-ci. Or, cela n'est possible que si le procédé est très simple et facilement modélisable, de sorte que le système d'assistance et de contrôle soit très fiable. C'est le cas notamment du système expert ARTIFACT (FRANCIS et LEITH, 1985). Il faut remarquer que la notion d'aide à la décision n'a plus vraiment de sens ici car pour les tâches effectuées par l'agent l'homme est pratiquement exclu de la boucle d'interaction sur le procédé. De ce fait, se substitue la notion de système d'aide à l'action qui s'inscrit dans le cadre d'une coopération "horizontale", proposée par MILLOT (1988), où l'opérateur humain et le système d'aide sont placés au même niveau décisionnel par rapport au procédé, possédant dans certaines mesures, le "même" pouvoir de contrôle sur celui-ci. Une application d'une telle coopération homme-machine est actuellement réalisée dans le domaine du contrôle aérien, dans le cadre d'une collaboration entre le L.A.I.H et le Centre d'Etude de la Navigation Aérienne. Cette application vise le partage des tâches de surveillance du trafic aérien, de détection et de résolution de conflits en vol de nuit entre le contrôleur aérien et un système expert SAINTEX (DEBERNARD et al.,

1992). Le principe est de répartir dynamiquement les tâches entre l'opérateur et SAINTEX. Ce dernier intervient essentiellement de deux façons, faisant actuellement l'objet d'expérimentations :

- de manière "explicite", quand l'opérateur estime que sa charge de travail est trop importante : il se décharge ainsi de certaines tâches en les confiant à SAINTEX ;
- de manière "implicite", quand les performances du système global se dégradent suite à une surcharge de l'opérateur : dans ce cas, un étage décisionnel supérieur décharge l'opérateur de certaines tâches en les confiant à SAINTEX.

Il importe que des considérations particulières soient données à l'élaboration de l'interface homme machine, pour cette classe de système d'assistance. En effet, l'opérateur ne s'occupant pas de certaines tâches, il doit tout de même facilement pouvoir suivre l'évolution de celles-ci. Il a besoin de comprendre aussi toutes les actions entreprises par le système d'aide.

III.4. LES CONTROLEURS ("MONITORS")

Les "contrôleurs" sont connectés directement au procédé, figure I.8, par l'intermédiaire d'un ensemble de capteurs ou de calculateurs. Ils récupèrent ainsi toutes les informations nécessaires à leur diagnostic et proposent à l'opérateur des conseils de haut niveau à partir de l'analyse des données provenant des capteurs.

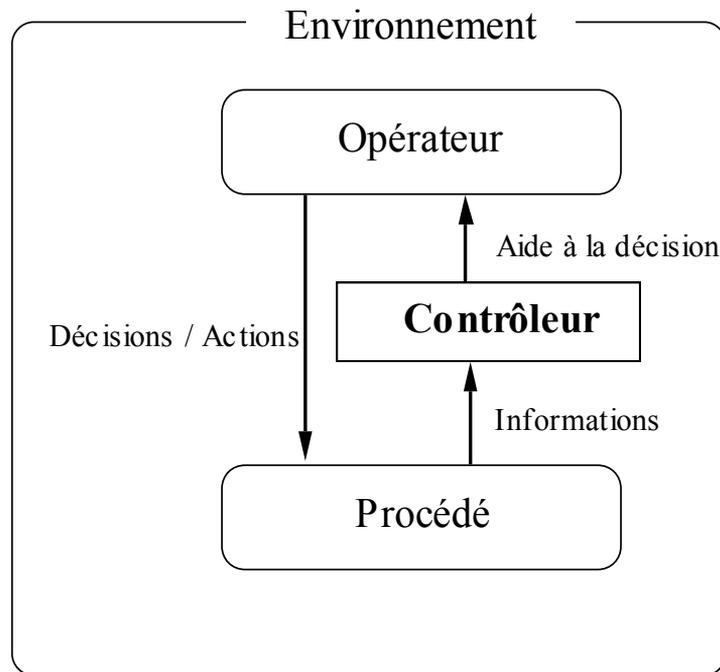


Figure I.8 : Positionnement des "contrôleurs" dans le système homme-machine

Afin de générer des conseils de haut niveau, le contrôleur doit intégrer un ou plusieurs modèle(s) du procédé pouvant prendre la forme de bases de données, de règles de production et/ou encore de réseaux de causalité. C'est le cas par exemple des systèmes suivants représentatifs de cette classe :

- Le système expert temps réel PICON (MOORE et al., 1984) est sans doute le plus connu des systèmes de cette classe. Il propose à l'opérateur, suite à une phase de diagnostic, l'action corrective à entreprendre, le temps qu'il faut pour que celle-ci soit efficace et le résultat attendu de l'action en utilisant, pour cela, deux types de connaissances : des connaissances structurelles sur le procédé et des connaissances heuristiques sur le fonctionnement de celui-ci. PICON a été enrichi pour devenir le système G2, qui est un outil de développement d'applications d'aide à la conduite de procédé en ligne (ESIA, 1992). Cet outil assure la formalisation du savoir-faire en intégrant les connaissances du procédé, d'exploitation et de maintenance. Il est organisé autour de six modules : (1) une base de données temps réel reposant sur une implantation objet, (2) un moteur d'inférence d'ordre 1, (3) un éditeur d'interface multi-fenêtres, (4) des interfaces d'acquisition de données, (5) un simulateur numérique et (6) un séquenceur temps réel de tâches.

- Le système ESCORT (SACHS et al., 1986) effectue un diagnostic de panne et propose à l'opérateur des conseils pour corriger la situation. Il est implanté sur une machine LISP et utilise des règles de production établies à partir de connaissances fonctionnelles sur les liens directs entre les alarmes et les causes possibles.
- Le système d'assistance réalisé lors du projet ALLIANCE (CALOUD et al., 1988) est capable de prédire l'état futur du procédé à partir de la situation de fonctionnement courante, et de proposer, si une mauvaise évolution a été prédite, des conseils d'actions préventives visant à contrecarrer cette évolution. De plus, dans le cas où l'évolution prédite diffère de l'évolution courante, le système est capable de diagnostiquer l'événement survenu. Dans ce cas, la connaissance est modélisée sous la forme d'un graphe de causalité. Ces principes se retrouvent dans le système PREDEX (GAMBIEZ et al., 1990 ; 1991).

III.5. AUTRES CLASSES

Nous pouvons citer encore deux autres classes de systèmes d'assistance : les "filtres décisionnels" et les "contrôleurs décisionnels".

Les "filtres décisionnels" sont semblables aux servants. Cependant les décisions sont prises par l'opérateur et sont filtrées, éliminant ainsi celles dont les conséquences peuvent être fâcheuses, figure I.9.

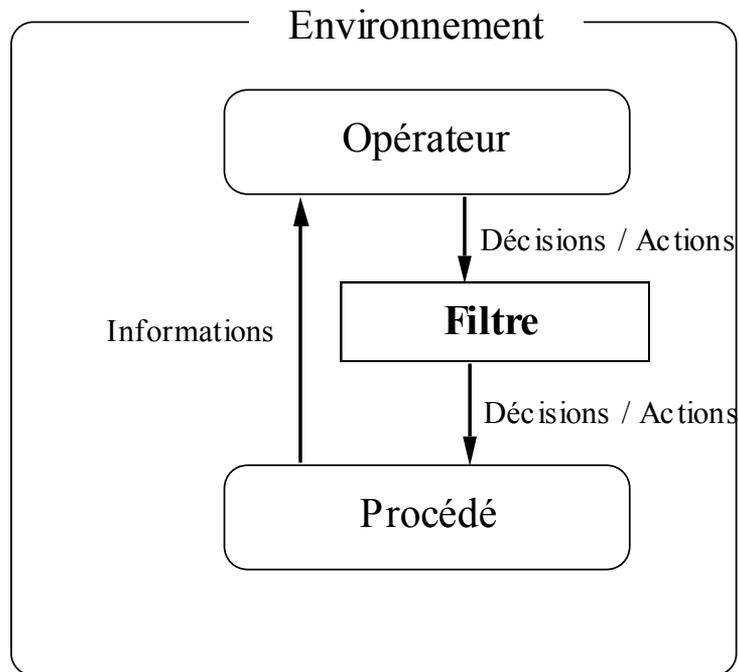


Figure I.9 : Positionnement des "filtres décisionnels" dans le système homme-machine

En combinant les systèmes de type contrôleurs et ceux de type filtres décisionnels, nous obtenons une nouvelle classe de système d'assistance. Celle-ci peut être appelée "contrôleurs décisionnels", figure I.10.

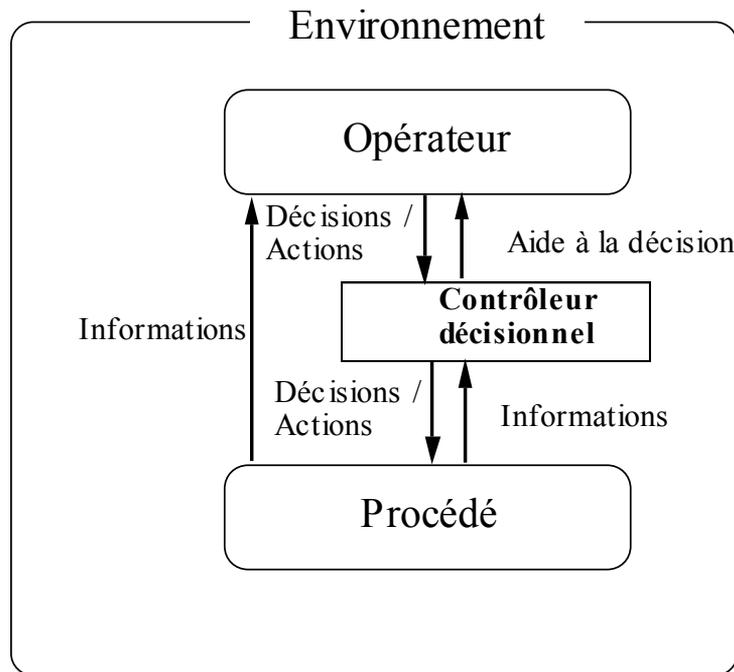


Figure I.10 : Positionnement des "contrôleurs décisionnels" dans le système homme-machine

En raison de l'importance du flux d'information provenant des "contrôleurs" et des "contrôleurs décisionnels", il est important que l'interface homme-machine soit capable de gérer ce flux, en choisissant l'information la plus utile et sa représentation la plus adaptée pour chaque situation et rendre ainsi sa compréhension et son utilisation plus faciles pour l'opérateur humain.

Nos travaux concernent plus particulièrement ces deux types de systèmes d'assistance.

CONCLUSION

Après avoir décrit le positionnement des systèmes d'assistance dans l'architecture des systèmes de contrôle et de supervision de procédé industriels, nous avons présenté dans ce chapitre l'évolution du rôle de l'opérateur humain en salle de contrôle. Ainsi, l'évolution technologique entraîne des changements dans les modes de contrôle des procédés industriels. Dans le cas de fonctionnement normal, les automatismes, de plus en plus nombreux, assurent la plus grande partie du travail de régulation grâce notamment aux PID et aux systèmes numériques de contrôle commande (SNCC). Cependant, dès qu'un défaut survient et que la situation se dégrade, l'opérateur humain devient très sollicité.

Les tâches de l'opérateur sont devenues hautement cognitives et exigent de sa part des connaissances de plus en plus importantes. Une solution visant à faciliter le travail de l'opérateur consiste à intégrer dans le système homme-machine des systèmes informatiques d'assistance à l'opérateur humain. Cette assistance peut se présenter sous plusieurs formes, et consister par exemple en une aide à la reprise de défauts, au diagnostic, ou encore à l'évaluation de la situation. Après avoir décrit un ensemble de caractéristiques humaines que le concepteur d'outil d'assistance devra considérer, il a été possible de répartir globalement les systèmes d'assistance en quatre classes : les consultants, les servants, les agents et les contrôleurs ("monitors"). Ces classes se différencient particulièrement par les modes de coopération instaurés entre l'homme et la machine, au niveau de la transmission des informations provenant du procédé et de la réalisation d'actions sur celui-ci.

Mais, un certain nombre de problèmes est lié à l'élaboration de ces systèmes d'assistance à l'opérateur. Ces problèmes concernent par exemple la modélisation du procédé, la prise en compte des tâches à réaliser par l'opérateur et surtout le mode de dialogue entre ces systèmes d'assistance et l'opérateur humain. Ces systèmes d'assistance utilisant de plus en plus des techniques issues de l'intelligence artificielle, le dialogue entre l'opérateur humain et ces systèmes est devenu un problème qu'il faut résoudre rapidement. En effet, dans ces conditions, l'opérateur humain se retrouve confronté à des heuristiques souvent inspirées des modes de traitement de l'information par lui-même (!). Mais, le rapport entre l'opérateur humain et de telles heuristiques reste mal maîtrisé actuellement et constitue un champ de recherche encore mal exploré (HOC, 1990).

Pour résoudre le problème du dialogue homme-machine, de nombreuses recherches sont entreprises. Parmi ces recherches, certaines visent l'étude, la composition et la mise en œuvre des interactions homme-machine au travers d'interfaces dites évoluées. Notre contribution s'insère dans cette nouvelle classe d'interface homme-machine. Elle vise la réalisation d'une interface "intelligente", basée sur des modèles et des connaissances sur les différents partenaires mis à contribution, en l'occurrence le procédé, le système d'assistance et l'opérateur humain.

Dans le chapitre suivant, nous parcourons différents types d'interfaces dites évoluées à travers plusieurs approches pratiques ou théoriques rencontrées dans la littérature.

Chapitre II

SYSTEMES DE DIALOGUE HOMME-MACHINE EVOLUES

CHAPITRE II : SYSTEMES DE DIALOGUE HOMME-MACHINE EVOLUES

INTRODUCTION

Comme l'a précisé le chapitre précédent, les systèmes homme-machine se sont développés ces dernières années entraînant des changements profonds au niveau des systèmes de contrôle et de supervision. Ces changements ont entraîné une réorganisation des rôles de l'homme et de la machine. Alors que les tâches de bas niveau sont laissées à l'automatisme, on a attribué à l'opérateur humain des tâches hautement cognitives. Dans ces conditions, les systèmes d'assistance mis à la disposition de l'opérateur peuvent lui offrir une aide précieuse.

Le problème du dialogue entre ces systèmes d'assistance et l'opérateur humain occupe une place très importante dans l'élaboration des systèmes homme-machine. Une solution à ce problème consiste en la mise en œuvre d'une interface "**évoluée**".

Il existe plusieurs principes d'interface capables de mieux aider les opérateurs dans leur dialogue avec les systèmes d'assistance ou le procédé. Cependant, la notion d'interface "**évoluée**" provient d'un grand nombre de propositions dans la littérature, pour caractériser les interfaces utilisant la notion "d'intelligence", ou encore les systèmes d'aide à la décision "intelligents" intégrant un système de gestion de l'interface utilisateur. L'interface évoluée se charge exclusivement du problème de dialogue entre l'opérateur humain et l'ensemble des partenaires du système homme-machine.

On rencontre dans la littérature des approches d'interfaces évoluées sous la forme *d'interface flexible*, *d'interface tolérante aux erreurs humaines*, *d'interface experte*, *d'interface adaptative*, *d'opérateur assistant*, *d'agent intelligent*, *d'interface hypermédia* ou encore *d'interface écologique*. Cette liste n'est évidemment pas exhaustive, mais se veut représentative des approches originales rencontrées. Dans ce chapitre, nous présentons successivement un ensemble d'approches en décrivant le principe de chaque type d'interface, sa structure informatique générale et un exemple représentatif d'application.

I. LES INTERFACES FLEXIBLES

I.1. PRINCIPE

Le concept d'interface flexible est apparu dans la littérature durant les années 80. Il vise à permettre l'utilisation d'un même support de dialogue pour communiquer avec plusieurs systèmes différents.

L'opérateur peut configurer l'interface et ses supports selon des critères personnels subjectifs : préférences dans les modes de présentation et/ou les styles d'interaction (WILLIGES et al., 1987). Une interface de ce type peut également être qualifiée de "configurable" ou "d'adaptable".

I.2. STRUCTURE GENERALE

Une interface flexible est composée généralement d'un ensemble de bases de données exploitées par un gestionnaire de présentation (que l'on peut retrouver également sous la dénomination "presentation manager" dans la littérature anglophone). Ce gestionnaire de présentation est piloté par l'opérateur humain selon l'application visée. Les connaissances requises pour une interface flexible sont regroupées généralement dans 3 bases de données, Figure II.1 :

- La première base contient l'ensemble des protocoles de communication avec les différentes applications constituant l'environnement de travail de l'opérateur, car généralement chaque application possède son propre système de communication. Cette base est définie a priori par le concepteur.
- La deuxième base contient les modes de présentation (couleurs, tailles de fenêtres, etc) et les différents styles d'interaction accessibles aux utilisateurs de l'interface. Cette base est définie a priori par le concepteur.
- La troisième base, spécifique à chaque utilisateur du système global, contient les préférences de chaque opérateur dans les modes de présentation et/ou dans les styles d'interaction. Ces préférences dépendent souvent de leurs propres habitudes. Cette base est modifiable par l'utilisateur.

Ainsi, en sélectionnant l'application, l'opérateur sélectionne implicitement, grâce au sélecteur d'application, le protocole de communication adéquat. Il peut alors dialoguer avec l'application à travers une interface déjà configurée mais assez flexible pour être reconfigurée de façon simple, et à tout moment par l'opérateur.

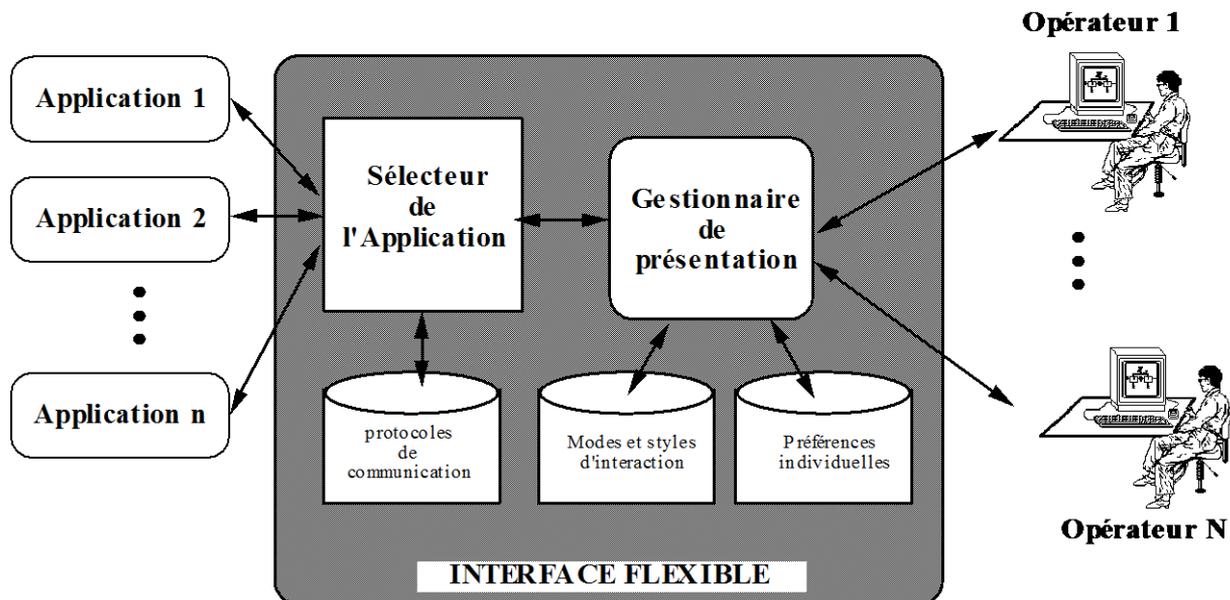


Figure II.1 : Structure d'une interface flexible

I.3. EXEMPLE D'UNE INTERFACE FLEXIBLE

Une étude de MULLER (1988) porte sur l'élaboration d'une interface flexible permettant l'utilisation de plusieurs outils d'Intelligence Artificielle (IA) et l'accès à plusieurs bases de données. Ces outils d'IA et ces bases de données sont répartis sur un réseau de trois ordinateurs non compatibles du point de vue de leur exploitation, figure II.2. Le problème consistait ici à trouver un moyen d'accéder aux bases de données et aux outils d'Intelligence Artificielle de ces ordinateurs de manière à ce que cette opération reste transparente pour l'utilisateur, car en effet, pour utiliser une ressource d'un ordinateur, l'utilisateur aurait dû connaître au moins quelques instructions du système d'exploitation de celui-ci.

La solution proposée par MULLER consiste en l'utilisation d'une interface flexible gérée par un ordinateur intermédiaire. Si l'opérateur veut accéder à l'ordinateur 1 à travers l'ordinateur intermédiaire, celui-ci se charge alors d'émuler

l'ordinateur 1, grâce aux bases de connaissance et de données qu'il possède. Il en est de même pour les deux autres ordinateurs. De ce fait, quelle que soit la destination de la requête de l'opérateur, celle-ci sera toujours exprimée de la même façon. Ainsi, la tâche de l'opérateur est simplifiée car il peut utiliser les mêmes instructions et la même interface de dialogue pour questionner n'importe quel ordinateur de ce réseau. Il peut de plus modifier certaines caractéristiques du dialogue homme-machine.

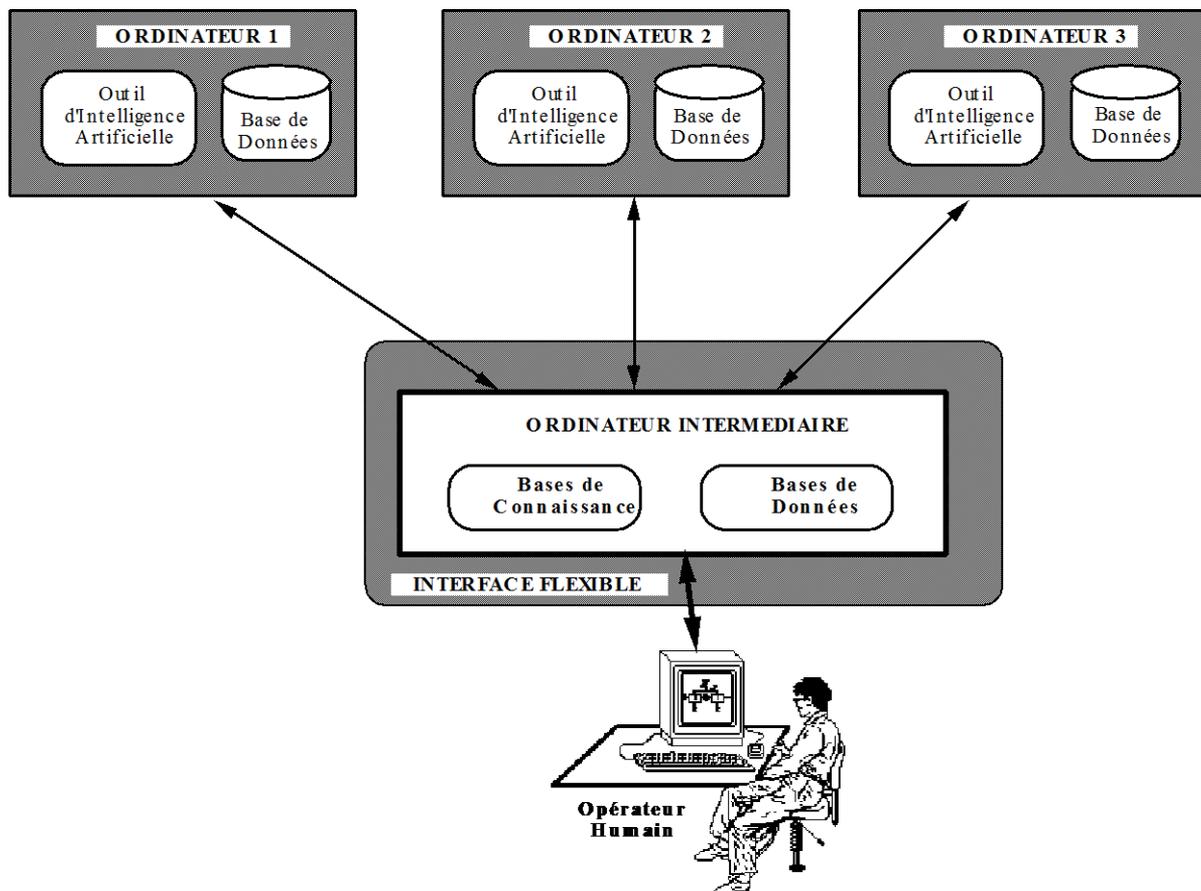


figure II.2 : Réseau de trois ordinateurs de systèmes d'exploitation différents accessibles par une interface flexible

D'après MULLER (1988), cette interface a connu beaucoup de succès chez ses utilisateurs. Validée sur les aspects de consultation de bases de données, elle a donné de bons résultats, de sorte que l'auteur envisage maintenant une extension de celle-ci vers "une interface universelle" valable pour le plus grand nombre d'ordinateurs possibles.

I.4. DISCUSSION

Ce type de système de dialogue peut être intégré dans les systèmes d'assistance de type "filtre décisionnels", figure II.3. En effet, les actions de l'opérateur sont filtrées et traduites dans le langage de l'application sélectionnée. De plus, pour ce type de système de dialogue, les informations provenant du procédé sont, de la même façon que les actions de l'opérateur, filtrées et traduites selon un style pré-défini par l'opérateur.

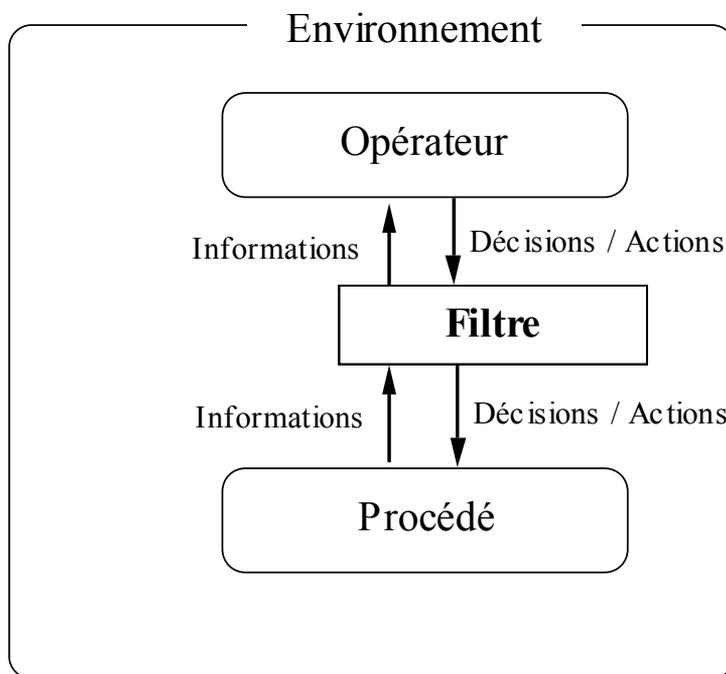


Figure II.3 : Intégration des interfaces flexibles dans des systèmes d'assistance de type "filtres décisionnels"

Le problème de la flexibilité des interfaces homme-machine se pose généralement lors de l'utilisation d'un ensemble d'applications hétérogènes.

Ce problème est par exemple très sensible chez les consultants de bases de données. En effet, chaque ordinateur possède souvent son propre protocole de dialogue. Cet état de fait augmente ainsi globalement la période d'apprentissage des utilisateurs, et par voie de conséquence le coût d'utilisation. Le principe de l'interface flexible est donc très intéressant à ce point de vue.

Cependant, pour plusieurs types d'équipements servant plusieurs systèmes d'exploitation, le coût de développement devient proportionnel au nombre d'utilisateurs, au nombre d'applications et surtout à leur hétérogénéité.

Néanmoins, ce type de système de dialogue est prometteur dans le domaine du contrôle de procédés industriels, puisqu'il vise à faciliter à l'opérateur l'utilisation des différents outils informatiques mis à sa disposition, situation que l'on retrouve très souvent dans les salles de contrôle.

II. LES INTERFACES TOLERANTES AUX ERREURS HUMAINES

II.1. PRINCIPE

La mise en œuvre d'interfaces tolérantes aux erreurs humaines se base généralement sur les résultats de travaux préalables consistant en l'observation des opérateurs dans des situations simulées. Ces observations visent à identifier leurs différentes erreurs se traduisant par des actions ne satisfaisant pas un ensemble pré-défini de critères de production et/ou de sécurité.

En situation réelle, le principe de ces interfaces consiste alors à remplacer, améliorer ou annuler les actions humaines inappropriées. Cette méthode repose sur le fait que l'opérateur humain est toujours sujet à des erreurs intempestives même s'il est assisté, lors de ses tâches, par des outils d'assistance performants (ROUSE et MORRIS, 1985 ; MALVACHE, 1990).

II.2. STRUCTURE GENERALE

La structure d'une interface tolérante aux erreurs humaine est généralement composée de trois modules principaux (figure II.4) :

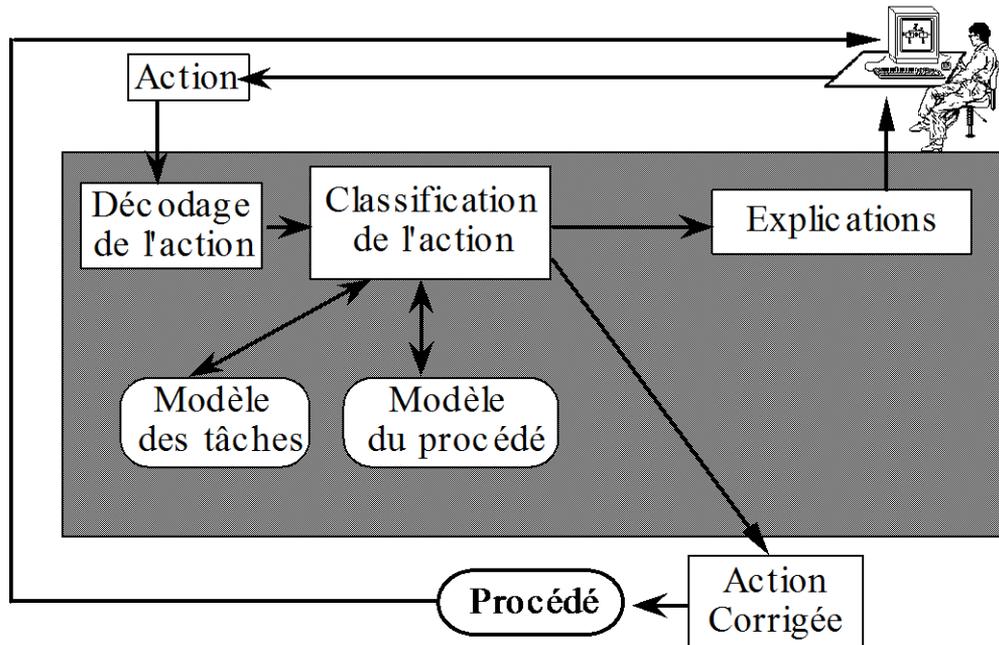


Figure II.4 : Structure générale d'une interface tolérante aux erreurs humaines

- Un premier module appelé *décodeur* traduit les intentions prenant la forme d'actions de l'opérateur en données exploitables par le système informatique.
- Un second module *classifie* d'abord l'action présumée de l'opérateur, en se basant sur un *modèle des tâches prescrites* de l'opérateur et sur un *modèle de l'application*. Puis, il la corrige, dans le cas où elle n'est pas conforme avec les consignes de production et/ou de sécurité.
- Un troisième module est chargé d'expliquer à l'opérateur les problèmes liés à l'action d'origine et les avantages apportés par l'action corrigée, vis-à-vis des consignes de départ.

II.3. EXEMPLE D'INTERFACE TOLERANTE AUX ERREURS HUMAINES

Les travaux visant la réalisation d'interfaces tolérantes aux erreurs humaines se basent généralement sur des recherches menées en fiabilité humaine parvenant à des typologies d'erreurs humaines, telles celles proposées par RASMUSSEN (82), ROUSE et ROUSE (1983), LEPLAT (1985), RASMUSSEN et VICENTE (1987), etc. Le système RESQ, décrit par HOLLNAGEL (1989), est capable de détecter en

ligne des erreurs humaines, dans un domaine bien ciblé. Il intègre des méthodes d'identification et d'évaluation de plans d'actions ainsi que d'élimination d'erreurs.

RESQ est un des composants du système développé dans le cadre du projet GRADIENT (Projet Esprit #857). Un des buts du projet est de réaliser un système d'assistance "intelligent" pour le contrôle de procédés industriels complexes de type centrale d'énergie. RESQ utilise des techniques de reconnaissance de plans afin d'assister l'opérateur de contrôle. RESQ supervise les actions de l'opérateur afin de détecter les situations où celles-ci s'avèrent incohérentes ou incorrectes vis-à-vis des buts de l'opérateur, par exemple : quand l'opérateur oublie une action commencée dans le passé, quand l'exécution de l'action est retardée, quand plusieurs actions sont exécutées en même temps, etc.

Le système RESQ est constitué de 5 modules principaux, figure II.5 :

- L'identificateur des plans détermine, en utilisant la bibliothèque de plans, dans quel plan l'action de l'opérateur se situe.
- L'évaluateur des plans utilise des règles pour classer les différents plans identifiés, selon leur probabilité d'être utilisés.
- Un Blackboard contient un ensemble de données caractérisant la meilleure estimation de ce que l'opérateur est en train de faire : le nom du plan courant, les buts de ce plan, les actions déjà exécutées de ce plan et la prochaine étape probable du plan.
- Le Gestionnaire de l'erreur calcule l'importance de l'erreur et détermine si oui ou non il faut informer l'opérateur et lui demander d'intervenir pour corriger l'erreur. Ce module est aussi capable de fournir à l'opérateur les causes de son erreur, en utilisant une base de connaissance gérant les actions de celui-ci.
- La Bibliothèque des plans contient l'ensemble des plans supposés connus par l'opérateur. Un plan est constitué par un ensemble d'actions, un but et des contraintes. Le but est atteint quand l'ensemble des actions est achevé sous les contraintes spécifiques à ce plan.

D'après HOLLNAGEL, la mise en œuvre et l'évaluation de RESQ ont donné des résultats prometteurs, ce qui ouvre des perspectives d'utilisation dans de nombreux autres domaines. Néanmoins, l'idée doit être encore affinée avant

d'envisager son transfert industriel, en particulier en profitant des avancées récentes et attendues de l'Intelligence Artificielle Distribuée (IAD) et notamment des travaux sur la génération de plans (MANDIAU et al., 1991B). De plus un tel système pourrait être complété par un mécanisme d'Apprentissage Automatique pour enrichir la bibliothèque de plans.

Le système RESQ est représentatif des approches actuelles contribuant à la tolérance d'erreurs humaine. Plusieurs autres approches existent dans la littérature. Par exemple, ASTRÖM (1991) propose des idées issues de l'intelligence artificielle, comme les systèmes à base de connaissance, les réseaux neuronaux, l'apprentissage, la simulation qualitative et les techniques du contrôle flou pour gérer intelligemment les commandes d'un procédé.

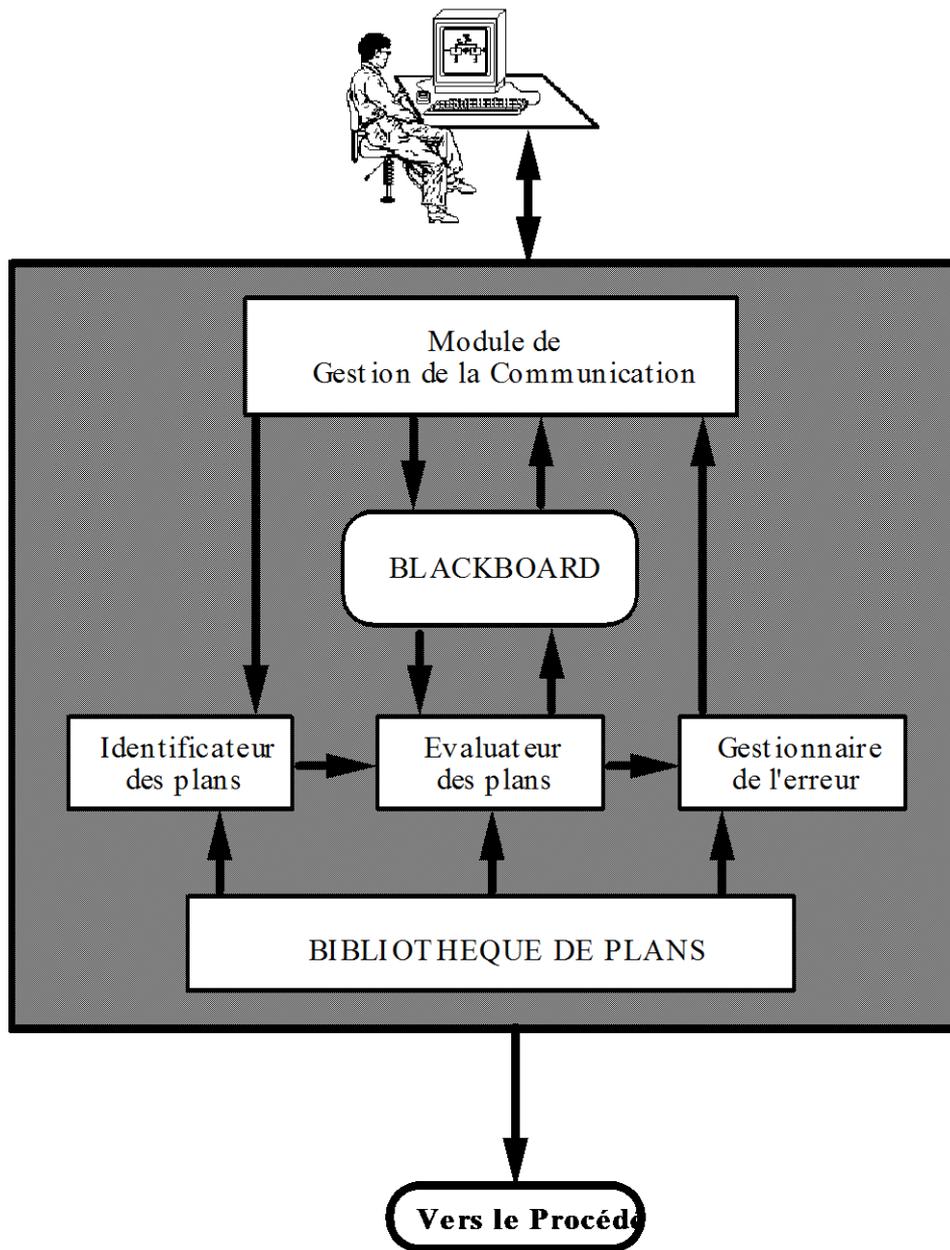


Figure II.5 : L'architecture du système RESQ (HOLLNAGEL, 1989)

II.4. DISCUSSION

Ce type de système de dialogue peut être intégré dans les systèmes d'assistance de type "filtre décisionnels", figure II.6. En effet, les actions de l'opérateur dont les conséquences peuvent être fâcheuses sont filtrées.

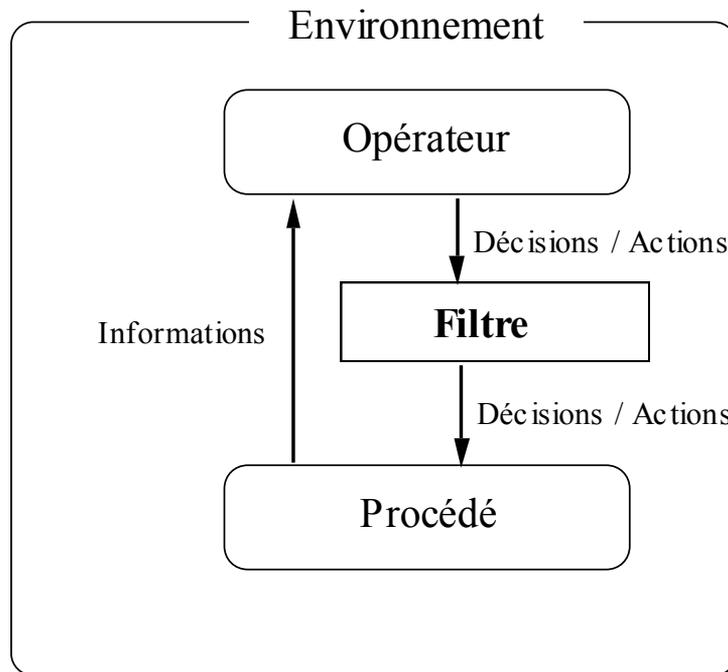


Figure II.6 : Intégration des interfaces tolérantes aux erreurs humaines dans des systèmes d'assistance de type "filtres décisionnels"

L'utilité d'un tel principe de tolérance aux erreurs humaines est indéniable. Cependant, ce principe n'est pas toujours applicable.

En effet, un tel système doit se baser soit sur un modèle de l'application (qui n'existe pas toujours et d'ailleurs quand il existe, il n'est pas toujours fiable), soit sur des procédures de correction pré-établies à l'aide d'expériences préalables. Dans ce dernier cas, la liste des actions correctrices ne peut être exhaustive, car elle est généralement établie grâce à des scénarios de pannes bien définis. Mais, dans la réalité, il arrive que de nouvelles situations de pannes surviennent. Celles-ci sont donc méconnues de l'opérateur et de l'interface tolérante aux erreurs humaines. Les risques d'erreurs deviennent alors importants, et une interface de ce type peut bien entendu s'avérer inefficace dans ces conditions.

Ce type de système de dialogue est en tout état de cause particulièrement prometteur par sa capacité à s'adapter aux commandes de l'opérateur humain. Ainsi, en plus des travaux déjà cités, ceux de nombreux autres chercheurs contribuent à la réalisation d'interfaces tolérantes aux (ou à certaines) erreurs humaines dans le domaine de l'assistance à l'opérateur. Par exemple, dans un but final d'**aide "intelligente" aux pilotes d'avion**, AMALBERTI (1986) propose une classification

des erreurs humaines en quatre types : (1) les erreurs de routine et d'inattention, (2) les erreurs guidées par un faux modèle mental, (3) les erreurs dues à un manque de connaissances, (4) les erreurs de personnalité. MASSON et DE KEYSER (1992) visent à concevoir un système "intelligent" de prévention d'erreurs de routine dans un contexte de **contrôle de procédé**. A ce sujet, de nombreux autres modèles d'erreurs humaines existent dans la littérature, tels celui proposé par NORMAN (1981), ou encore celui proposé par ROSCOE (1980) dans le domaine de l'aéronautique, ces deux modèles concernant particulièrement les erreurs d'inattention.

III. LES INTERFACES EXPERTES

III.1. PRINCIPE

Dans le concept d'interface experte ALTY et GUIDA (1985) considèrent l'interface comme un **intermédiaire** "intelligent" pouvant supporter des coopérations entre deux partenaires, l'homme et la machine, dans un contexte de résolution de problème. Une telle interface doit être capable :

- de gérer les communications homme-machine du point de vue linguistique et conceptuel, afin de rendre la compréhension de l'autre "partenaire" plus facile et plus efficace ;
- d'assister l'utilisateur pour une utilisation correcte et efficace du système, en prenant une part active dans le processus de résolution de problème, dans lequel il se trouve ;
- d'entraîner progressivement l'utilisateur dans l'utilisation du système informatique.

Une interface experte prend la forme d'un système expert intégrant dans ses bases de connaissances une expertise sur les modes possibles de coopération homme-machine.

III.2. STRUCTURE GENERALE

D'après ALTY et GUIDA, le système expert constituant cette approche d'interface "intelligente" utilise quatre types de connaissances : des connaissances sur (i) l'utilisateur de l'interface et du système, (ii) sur les buts du système à interfacier, (iii) sur le domaine du problème à résoudre et (iv) sur un module intermédiaire capable de traduire les requêtes de l'utilisateur dans le langage du système et inversement.

La structure informatique d'une interface experte comporte principalement trois modules, figure II.7 :

- Le gestionnaire de dialogue s'occupe des interactions avec l'utilisateur. Son but est de gérer le dialogue, **en langage naturel**¹, avec celui-ci.
- Le résolveur de problème est en relation directe avec l'application. Son but est de simuler différents niveaux d'expérience des utilisateurs, dans l'accès et l'utilisation de l'application.
- Le gestionnaire des modèles des utilisateurs est connecté aux deux autres modules. Son but est de gérer et mettre à jour différents modèles des utilisateurs.

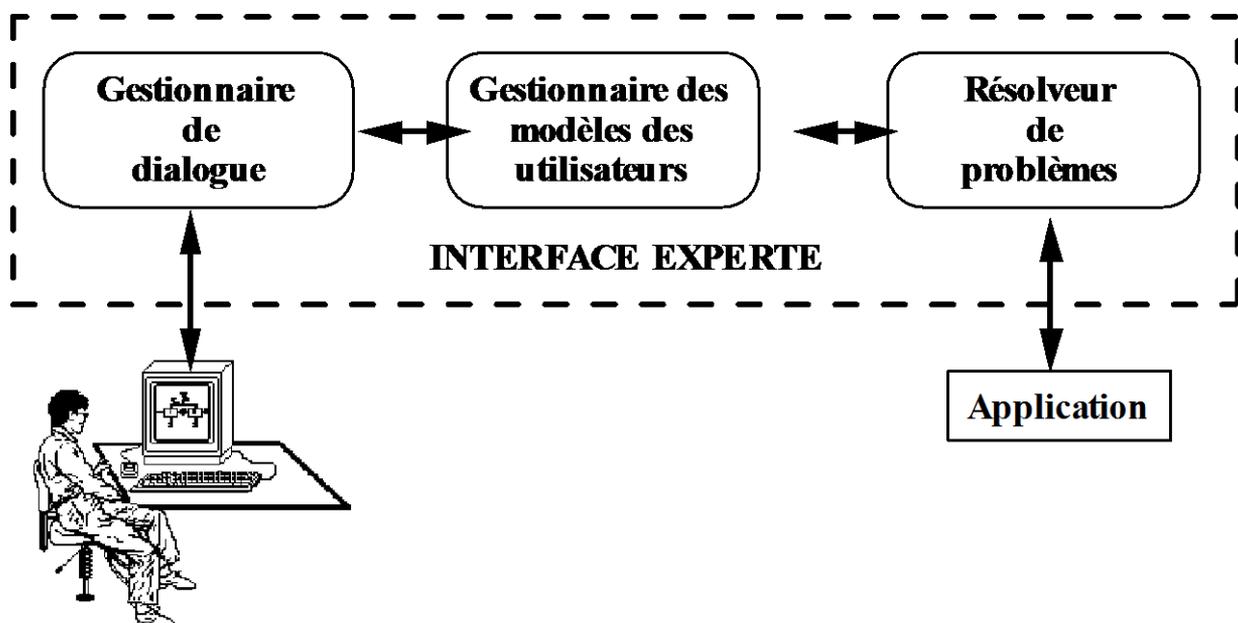


Figure II.7 : Structure générale d'une interface experte
(ALTY et GUIDA, 1985)

¹ Dans une interface en langage naturel, le dialogue homme-machine est exprimé clairement avec des mots, des phrases et des concepts familiers à l'utilisateur (MOLICH et NIELSEN, 1990). Le principe d'une interface en langage naturel est relativement intéressant dans le domaine de l'enseignement assisté par ordinateur, où il s'agit d'arriver à des systèmes informatiques capables de "remplacer" les professeurs et d'explorer les issues possibles concernant la nature et la façon dont les connaissances sont dispensées (ANDERSON et al., 1990). On y retrouve la nécessité d'une interaction homme-machine se rapprochant le plus possible de la conversation humaine et de ses règles. Une interface en langage naturel doit proposer à l'utilisateur un langage qui lui est le plus naturel possible. Ce n'est pas une mince affaire compte tenu du fait que ce qui est naturel pour un utilisateur expert ne le sera plus pour un utilisateur novice, et vice-versa, d'où la nécessité de construire des modèles d'opérateurs. Le lecteur trouvera dans l'article de SCHA (1988) la structure générale d'une interface en langage naturel.

III.3. EXEMPLE D'INTERFACE EXPERTE

Cette approche d'interface est utilisée dans un projet appelé IR-NLI (Information Retrieval - Natural Language Interface), dont le but est la réalisation et l'expérimentation d'interfaces "intelligentes", dans les systèmes de recherche en ligne de documents (GUIDA et TASSO, 1982 ; BRAJNIK et al. 1988). Dans cette application, "le gestionnaire des modèles de l'utilisateur" possède plusieurs rôles :

- la gestion des interactions avec les autres modules de l'interface ;
- la classification des répétitions, sur laquelle est basée la construction et la maintenance des modèles des utilisateurs ;
- la gestion de toutes les opérations de lecture et écriture dans des modèles d'utilisateur, au cours des activités de construction, de maintenance ou d'exploitation de ces derniers.

Le "le gestionnaire des modèles de l'utilisateur" utilise, pour ces différentes opérations :

- une base de connaissance de stéréotypes contenant des connaissances nécessaires à l'activité de modélisation des utilisateurs;
- une base de données sur les utilisateurs, contenant le modèle le plus récent de chaque utilisateur ayant accédé à l'interface ;
- une base de données de l'historique des différentes sessions d'utilisation de l'interface pour chaque utilisateur.

Les résultats de cette étude montrent que l'avantage d'une telle interface est, non seulement d'étendre l'utilisabilité d'un système complexe, mais aussi de façon plus importante d'améliorer (1) la qualité de l'interaction homme-machine, (2) les performances obtenues par la machine et (3) le degré de satisfaction de l'utilisateur.

III.4. DISCUSSION

Ce type de système de dialogue peut être intégré dans les systèmes d'assistance de type "contrôleurs décisionnels", figure II.8. En effet, grâce aux données du procédé et aux conseils de haut niveau fournis par le système d'aide, l'opérateur prend des décisions et applique des actions sur le procédé. Cependant, certaines actions de l'opérateur dont les conséquences peuvent être fâcheuses sont filtrées.

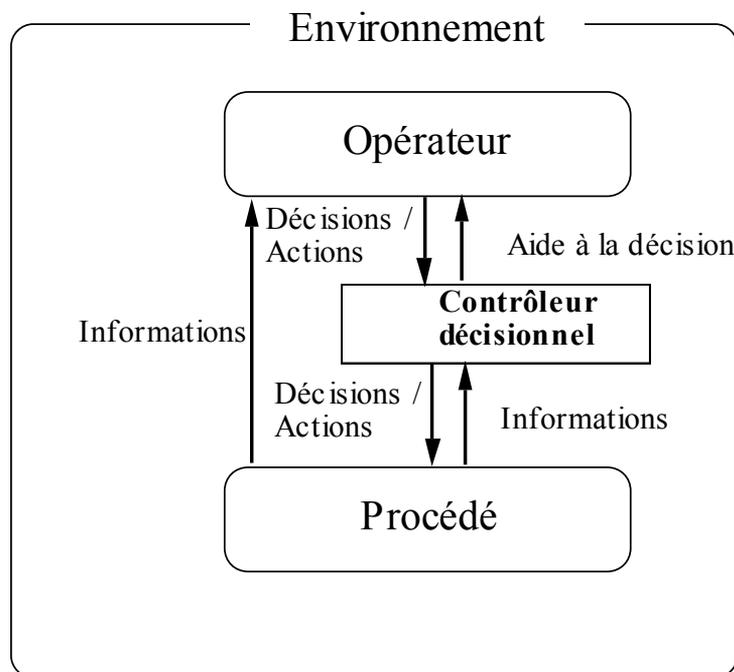


Figure II.8 : Intégration des interfaces "expertes" dans des systèmes d'assistance de type "contrôleurs décisionnels"

Cette approche d'interface, étendant la notion d'interface en langage naturel, est une généralisation des deux précédentes approches de système de dialogue. De plus, elle semble être très intéressante

Cependant, sa complexité réside surtout dans l'élaboration des différents modèles d'utilisateurs. En effet, suite à deux cas d'étude de laboratoire, les auteurs soulignent la difficulté d'arriver à un dialogue homme-machine satisfaisant quant des lacunes dans la modélisation de l'utilisateur persistent (BRAJNIK et al. 1990).

IV. LES INTERFACES ADAPTATIVES

IV.1. PRINCIPE

Le concept d'interface adaptative date de la fin des années 70. Il est en général basé sur le fait que les modes de dialogue avec l'utilisateur doivent s'adapter aux tâches que celui-ci a à accomplir. NORMAN (1979) a suggéré que ces types de systèmes intelligents soient autonomes, de sorte qu'ils soient capables de gérer leurs propres actions, leurs connaissances, leurs besoins, leurs buts ainsi que leurs intentions. Les recherches vont actuellement dans ce sens. Celles-ci visent à adapter les présentations des informations et les modes d'interaction aux utilisateurs en fonction des tâches à réaliser.

Cette adaptation prend souvent la forme d'une activité **éducatrice**, prenant soin des différences individuelles entre les utilisateurs, leurs préférences personnelles, leurs habilités et leurs problèmes. Elle vise alors à améliorer les connaissances et les performances de chaque opérateur pour des tâches pré-définies.

Une interface adaptative intègre particulièrement la notion de flexibilité en utilisant un modèle de l'opérateur qui est ajusté d'un opérateur à l'autre.

IV.2. STRUCTURE GENERALE

La structure d'une interface adaptative n'est pas unique. Cependant, les éléments de base, caractérisant l'adaptabilité de l'interface, sont généralement les mêmes. A ce sujet, de nombreuses architectures et méthodologies pour la conception d'interfaces ou d'outils adaptatifs existent dans la littérature (EDMONDS, 1981 ; ROUSE, 1988 ; NORCIO et STANLEY, 1989 ; HEFLEY, 1990).

Synthétisant plusieurs approches, HEFLEY (1990) présente une architecture théorique fonctionnelle d'un système adaptatif "intelligent", qui repose sur une modélisation de l'utilisateur. Cette architecture est orientée vers un groupement d'agents dont le but est d'assurer les interactions entre une application donnée et son utilisateur.

Cette architecture fonctionnelle, figure II.9, est composée d'une collection d'agents, qui contrôlent et servent d'intermédiaires entre l'utilisateur et l'application. Elle trace une séparation nette entre l'interface utilisateur et l'application, et intègre un modèle des tâches évolué sous forme d'un système "intelligent" d'aide à la décision (SIAD). Ce modèle des tâches est le modèle de l'application vu par l'utilisateur ; il contient une description des tâches que l'utilisateur effectue au travers de son interface. Il est composé des agents suivants :

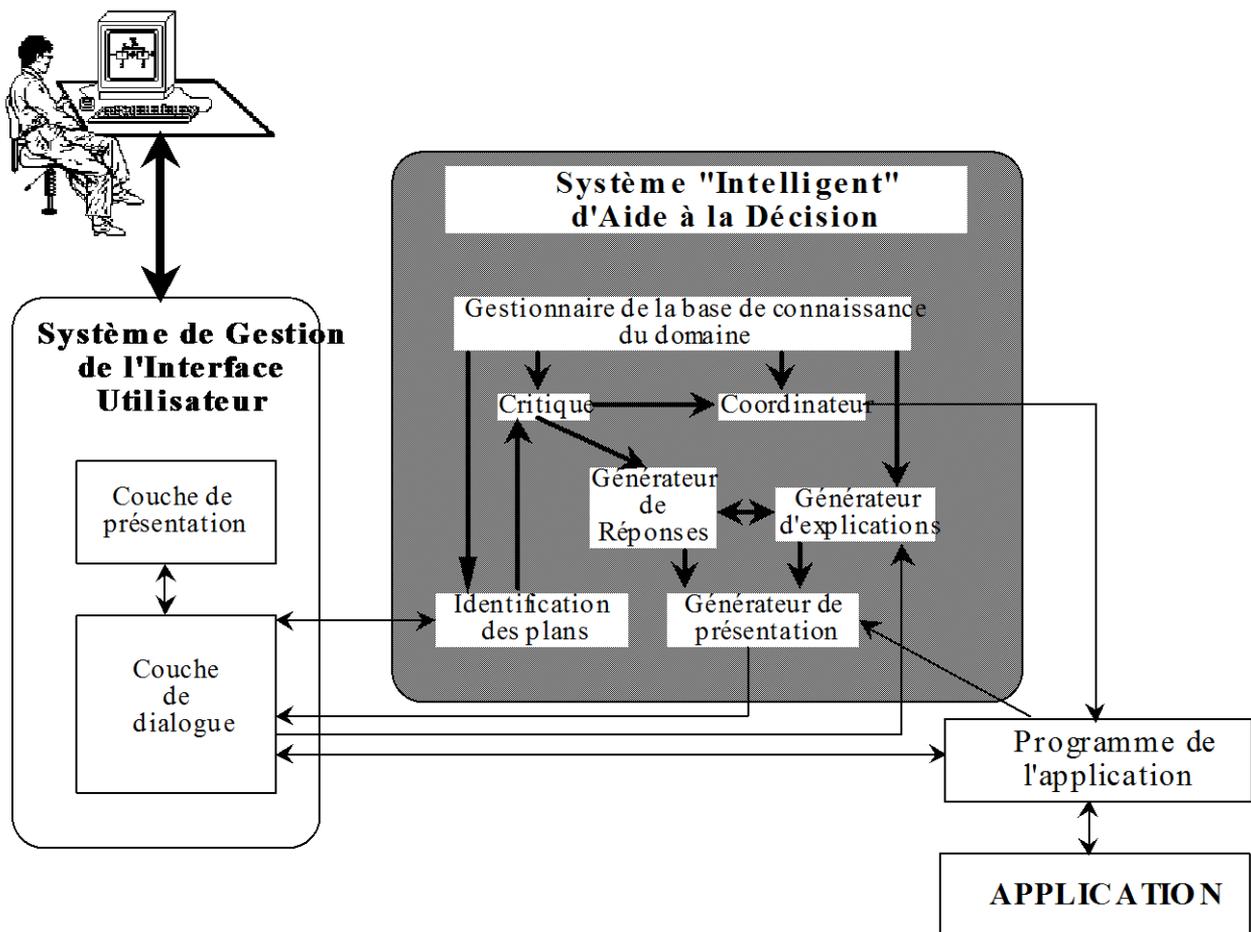


Figure II.9 : Architecture d'un système adaptatif (HEFLEY, 1990)

- *L'identificateur des plans*. En liaison avec une base de connaissance, cet agent infère des plans, représentant des actions observées de l'opérateur, afin de déterminer le but de l'action observée.
- *Le critique*. Cet agent identifie les actions des opérateurs considérées comme critiques vis-à-vis des contraintes à respecter. Ces actions critiques sont traitées par cet agent qui doit pouvoir arriver aux types de conclusions

suivantes : action non prévue, action hors ordre, paramètre non prévu, situation non prévue, etc. Notons qu'un travail important a été réalisé sur cette notion de "critique" par FISCHER et ses collègues à l'Université du Colorado (FISCHER, 1987).

- Le coordinateur des connaissances sur le domaine est capable d'allouer certaines tâches de l'opérateur au programme de l'application qui utilise des réponses planifiées. Pour cela, le coordinateur se base sur des heuristiques d'allocation de fonctions et sur les connaissances spécifiques à l'application.
- Le générateur de réponses. Cet agent se charge de répondre aux interrogations et aux questions de l'opérateur, suite à des événements inconnus. Cet agent doit permettre d'accélérer l'apprentissage de l'opérateur.
- Le générateur d'explications. Cet agent se charge de donner des explications sur les différentes conclusions du système d'aide. Ainsi, sur demande de l'opérateur, cet agent justifie les raisons qui l'ont poussé par exemple à décharger l'opérateur de telle tâche.
- Le générateur de présentation. Cet agent doit assurer des modes de présentation de l'information sous une forme compatible avec l'état courant des connaissances et des besoins de l'opérateur.

HEFLEY (1990) considère l'interface adaptative comme un système "intelligent" construit sur des facteurs et des connaissances heuristiques extraites d'un expert. Pour gérer ces connaissances, il propose l'utilisation de techniques de raisonnement classiques (telles celles disponibles dans les générateurs de systèmes experts), couplées avec des méthodes de raisonnement sur des situations non structurées (telles que la théorie des sous-ensembles flous, les techniques d'apprentissage, etc.). De plus, il souligne qu'une interface adaptative doit avoir la capacité **d'apprendre** au fur et à mesure que de nouvelles situations sont rencontrées et résolues, d'où la mise en place de techniques d'apprentissage issues de l'intelligence artificielle. Notons que cette notion d'apprentissage est visée dans les systèmes d'assistance réalisés dans le cadre du projet CID à la NASA (BOY, 1990), voir le paragraphe I.3.

IV.3. EXEMPLE D'INTERFACE ADAPTATIVE

Le système SUSI (Smart User System Interface), décrit par JERRAMS-SMITH (1989), est un essai de conception d'une interface adaptative intégrant une expertise provenant de plusieurs opérateurs experts dans l'utilisation du système d'exploitation UNIX. Le but de SUSI est d'aider un opérateur débutant lors de l'utilisation d'un système interactif utilisant UNIX.

Le système d'exploitation UNIX est une application particulièrement intéressante pour la conception d'interface adaptative. En effet, il est bien connu que les débutants éprouvent beaucoup de difficultés lors de son utilisation. De plus, un grand nombre d'utilisateurs potentiels sont disponibles pour cette étude, de même que des experts dans ce système d'exploitation.

L'un des buts du développement de SUSI est de trouver une méthode d'aide aux utilisateurs débutants d'UNIX, visant à exploiter au maximum leurs capacités. Le principe de cette aide est basé sur l'identification de certaines classes d'erreurs typiques commises par des utilisateurs débutants.

SUSI s'utilise comme une interface de dialogue entre l'utilisateur et le système UNIX qui détecte les erreurs sur les commandes des utilisateurs et leur renvoie la syntaxe exacte de ces commandes. Ainsi, l'utilisateur n'a pas à perdre de temps à chercher lui-même la bonne syntaxe lors de ses tâches d'utilisation d'un système informatique.

L'évaluation de SUSI a montré que ce système aide bien les utilisateurs débutants, car il leur permet de corriger eux mêmes leurs erreurs et d'avancer plus vite dans leur apprentissage. Cependant, SUSI reste sans effet sur les autres types d'utilisateurs, car généralement ces derniers connaissent souvent la syntaxe exacte de toutes les commandes les plus utilisées.

IV.4. DISCUSSION

Ce type de système de dialogue peut être considéré comme une généralisation de l'interface experte. Elle peut ainsi être intégrée dans les systèmes d'assistance de type "contrôleurs décisionnels", figure II.10.

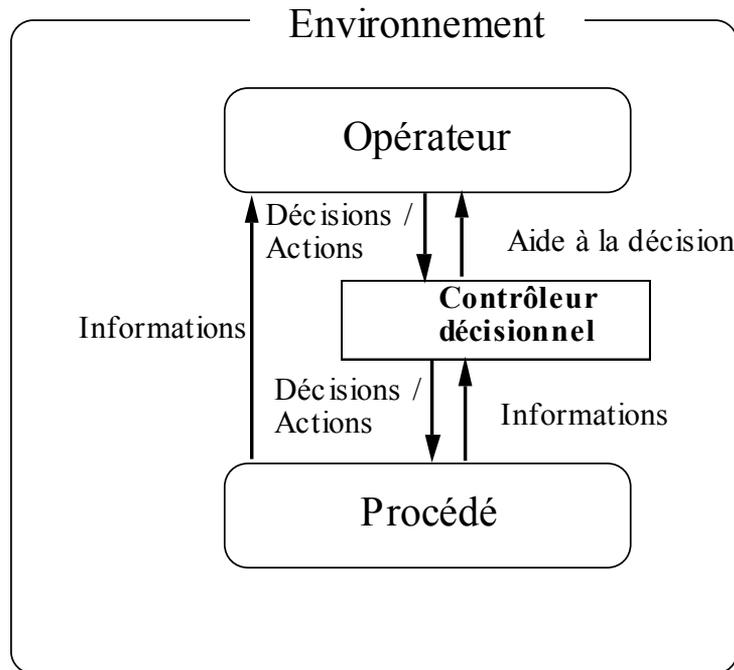


Figure II.10 : Intégration des interfaces adaptatives dans des systèmes d'assistance de type "contrôleurs décisionnels"

NORCIO et STANLEY (1989) notent qu'une interface adaptative doit être considérée comme une partie intégrante d'un système d'assistance global, de sorte que l'adaptation soit prise en compte dans le contexte de l'application. A ce titre l'interface adaptative représente une voie de généralisation de l'interface tolérante aux erreurs humaines présentée au paragraphe précédent. Ils présentent, cependant, plusieurs raisons pour lesquelles ce type d'interface n'est pas encore désirable :

- l'opérateur peut ne pas être capable de développer un **modèle cohérent** du système quand celui-ci est constamment en train de changer. Ainsi, si l'opérateur n'a pas une compréhension très claire du système qu'il contrôle ou utilise, son efficacité peut être sérieusement réduite ;

- un autre problème qui risque d'arriver est celui du sentiment de l'opérateur d'avoir **perdu le contrôle** du système;
- une dernière critique de ce type d'interface concerne le problème de la complexité et du coût de cette mise en œuvre.

VAN DER VEER (1989) souligne, quand à lui, la nécessité de considérer les différences individuelles des fonctions cognitives lors de la première installation des opérateurs débutants face à des systèmes compliqués et d'introduire, par la suite, des étapes transitoires faisant progresser l'opérateur de la phase "débutant" à la phase "expert". A ce sujet, d'après BOY (1988A), il convient de bien différencier les systèmes d'assistance pour les débutants et ceux destinés aux experts. Les comportements sont très différents. En effet, les débutants ont besoin de conseils analytiques (ils ont besoin d'apprendre), alors que pour les experts, le système d'assistance devra avoir un vocabulaire adapté et favorisant une interaction rapide.

Enfin, ce type d'approche fait l'objet de nombreuses recherches actuellement, mais reste très difficile à mettre en place surtout lorsque l'application est complexe. En effet, elle nécessite non seulement la connaissance, par les concepteurs, de certains paramètres psychologiques des utilisateurs, mais demande aussi à trouver des moyens fiables de détection des plus importants attributs des opérateurs pour lesquels le système doit s'adapter (WÆRN, 1989).

V. L'OPERATEUR ASSISTANT

V.1. PRINCIPE

Dans l'approche "d'opérateur assistant" (BOY, 1988A, 1988B ; VALOT et DEBLON, 1988), l'interface se comporte comme un assistant pour l'opérateur humain, et l'aide dans la réalisation de ses tâches. L'interface doit être capable d'initiatives personnelles, par exemple en testant des paramètres ou en prédisant des pannes déductibles, mais l'opérateur humain reste le décideur final. Le principe de l'opérateur assistant est basé sur une compréhension réciproque entre l'homme et la machine, la qualité de la communication entre deux individus dépendant de la compréhension réciproque du modèle interne de l'autre (BOY, 1988A). D'une manière générale, cet auteur définit un opérateur assistant par l'exemple suivant :

"Dans un cockpit d'avion, un copilote humain partage le travail avec le commandant de bord, mais pas la responsabilité ultime : le commandant de bord est maître à bord ! Le commandant de bord peut consulter son copilote n'importe quand pendant le vol, mais prendra les décisions ultimes. Si le commandant de bord délègue une part de ses responsabilités au copilote, alors le copilote prendra cette délégation comme une tâche à exécuter. De plus, le commandant de bord peut arrêter l'exécution d'une tâche du copilote n'importe quand, s'il le juge nécessaire. Cependant, un copilote peut avoir des initiatives personnelles, par exemple, tester des paramètres, se tenir au courant de l'évolution de la situation, prédire des pannes déductibles, etc. Un copilote peut traiter la connaissance incluse dans le manuel d'opération sur la demande du pilote. Il doit être capable d'expliquer, avec un niveau de détail approprié, les résultats de son traitement".

V.2. STRUCTURE GENERALE

La structure de l'opérateur assistant est constituée de plusieurs modules, figure II.11. Les plus importants sont :

- Le module de reconnaissance des situations. En fonction de la situation désirée, de la situation perçue et de connaissances situationnelles et

analytiques placées dans la mémoire à long terme, il sélectionne un sous-ensemble de règles à inférer.

- Le module de résolution de problème. Il utilise le sous-ensemble de règles précédent, dans le but d'assister l'opérateur, en répondant par exemple, à une requête particulière de celui-ci.

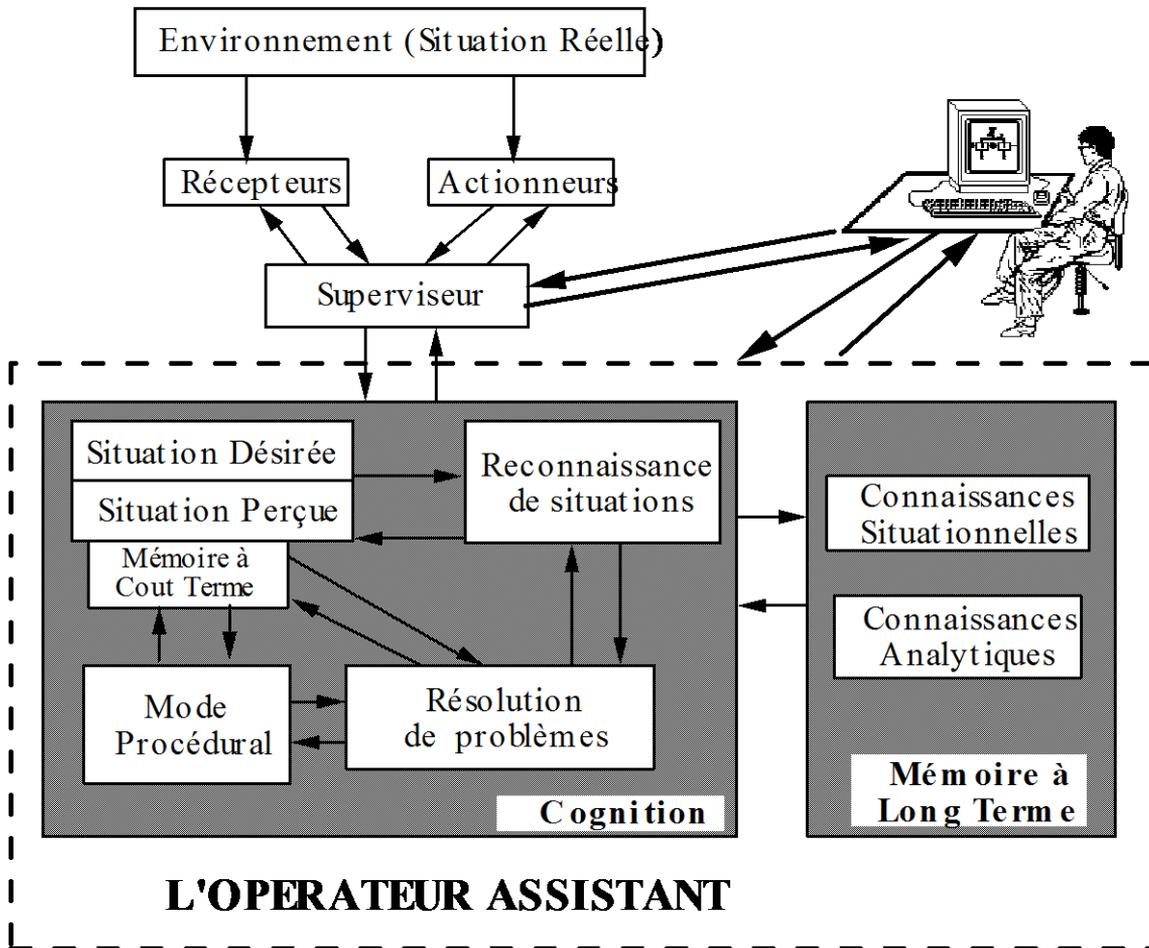


Figure II.11 : Structure du Modèle de l'opérateur (BOY, 1988B)

Grâce à la structure de ses connaissances, l'opérateur assistant peut avoir six niveaux d'autonomie :

- Le Niveau 0 : la base de connaissance n'est pas utilisée et donc l'opérateur assistant n'est pas actif.

- Le Niveau 1 : La base de connaissance est connectée au système, et l'opérateur assistant peut déduire les différentes situations possibles. Chaque situation constitue un contexte dans lequel différentes actions peuvent être effectuées.
- Le Niveau 2 : à ce niveau, l'opérateur assistant est capable de déterminer un ou plusieurs contextes susceptibles d'être sélectionnés par l'opérateur humain.
- Le Niveau 3 : à ce niveau, l'opérateur assistant choisit automatiquement un contexte correspondant à la situation en cours. Si le contexte est approuvé par l'opérateur humain, l'opérateur assistant se charge de l'exécuter et renvoie le message "exécuté" à l'opérateur humain. Dans le cas contraire, l'opérateur humain sélectionne lui-même le contexte et exécute les actions correspondantes.
- Le Niveau 4 : à ce niveau, le diagnostic analytique est complètement automatisé. L'opérateur assistant se charge d'analyser la situation, de l'identifier le contexte et de le proposer à l'opérateur humain. Durant cette phase, si les questions ne nécessitent pas une confirmation de l'opérateur humain, celui-ci ne voit évoluer aucune procédure de traitement. A la fin du traitement, les actions à entreprendre sont proposées à l'opérateur humain pour confirmation.
- Le Niveau 5 : à ce niveau, tous les traitements sont automatisés. L'opérateur assistant donne directement des conseils sur la situation en cours ainsi que les éventuels diagnostics. De plus, l'opérateur assistant doit être capable d'expliquer son raisonnement à l'opérateur humain.

V.3. EXEMPLE D'APPLICATION D'UN OPERATEUR ASSISTANT

Une étude de ce type d'interface a été réalisée sous la forme d'un modèle de l'opérateur appelé MESSAGE (BOY et TESSIER, 1985 ; BOY, 1986), lors d'un projet d'évaluation de cockpit d'avion, figure II.12. Dans ce cas, le système à contrôler est un simulateur d'un système de ravitaillement sur orbite appelé ORS (Orbital Refueling System), utilisé dans les navettes spatiales pour ravitailler les satellites.

L'opérateur assistant est implanté dans un système expert de diagnostic de pannes, appelé HORSES (Human-ORS-Expert System). Les connaissances

analytiques sont implantées sous forme de frames, de fonctions, de règles, de méta-règles et d'objets. Un objet peut être soit une variable numérique, soit une variable logique, soit une question du système vers l'opérateur ou une procédure devant être appliquée par l'opérateur ou par le système HORSES.

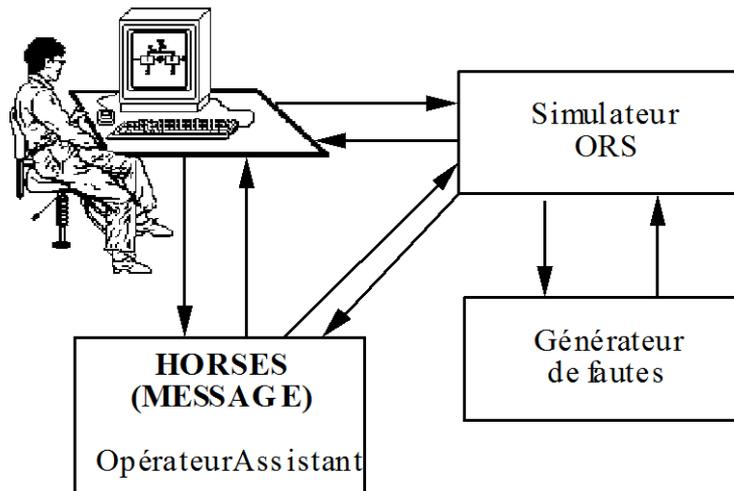


Figure II.12 : L'opérateur Assistant de Diagnostic de Pannes (BOY, 1988B)

L'étude ce type de système a permis la reproduction de la structure fine des processus mentaux de traitement d'informations, de mémorisation et de prise de décision dans des tâches hautement cognitives. Cependant, ce projet a également permis de mettre en évidence certaines difficultés et parfois même impossibilités liées à la décompilation des connaissances profondes avec les outils disponibles actuellement. Pourtant, ce type de connaissances est très important pour tester l'opérateur assistant en situation réelle, c'est-à-dire face à l'opérateur humain et au système, et ceci afin d'établir le niveau optimal d'autonomie de l'opérateur assistant (BOY, 1988B).

V.4. DISCUSSION

Ce type de système de dialogue peut être potentiellement considéré comme une généralisation de l'interface adaptative. Cependant, elle s'intègre dans les systèmes d'assistance de type "contrôleurs", car les décisions appartiennent à l'opérateur, figure II.13.

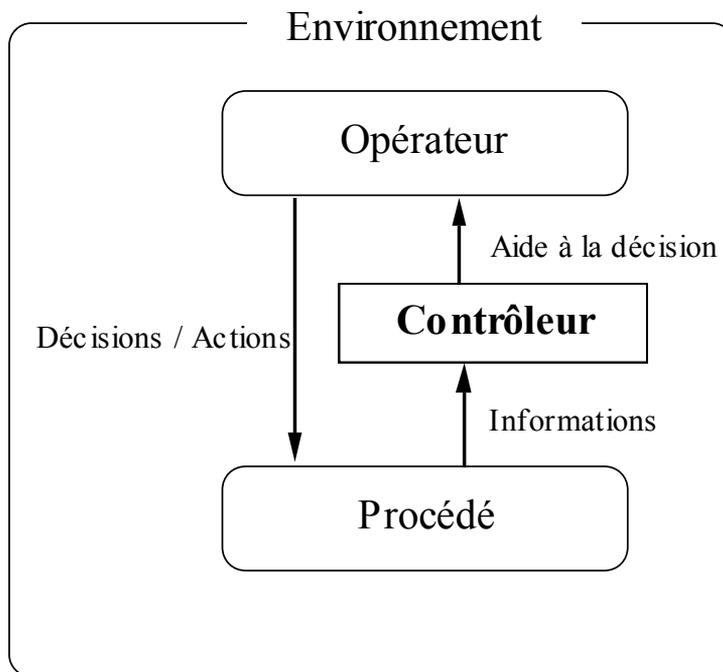


Figure II.13 : Intégration de l'opérateur assistant dans des systèmes d'assistance de type "contrôleurs"

BOY souligne que l'utilisation d'une représentation de connaissances analytiques et situationnelles permet l'interprétation et la formalisation d'observations brutes et de protocoles verbaux, permettant ainsi la mise en œuvre et la validation d'un opérateur assistant à partir de données d'expertise. Cependant, une telle type de mise en œuvre prend beaucoup de temps à cause notamment de l'absence d'outils permettant de standardiser les différentes procédures qu'appliquent les opérateurs, car elles varient d'un opérateur à l'autre.

Ce type de système de dialogue est prometteur pour le contrôle de procédés industriels, car il exige d'extraire les stratégies de contrôle et de supervision des opérateurs experts pour les mettre en œuvre dans les systèmes d'assistance.

VI. L'AGENT INTELLIGENT

VI.1. PRINCIPE

Le concept d'agent intelligent est inspiré du domaine de la résolution distribuée de problème et plus particulièrement de travaux menés en Intelligence Artificielle Distribuée, qui tentent de modéliser par un formalisme logique le comportement d'un agent rationnel, basé sur des états intentionnels, tels que les croyances, les désirs, les intentions et les engagements (CHAIB-DRAA, 1990).

Le principe d'un système de dialogue, utilisant les concepts de l'agent intelligent, repose sur la possibilité de décomposer le système homme-machine en un ensemble **d'agents Intelligents**, par exemple les sous-systèmes à superviser, les systèmes d'assistance, les systèmes de dialogue homme-machine et les opérateurs humains. Ces différents agents travaillent en parallèle et/ou coopèrent, dans le but de résoudre les problèmes de la communication homme-machine, du contrôle et de la supervision des procédés industriels complexes. Le résultat de leurs traitements est transmis à l'opérateur humain, grâce à des actes de communications sous forme de messages.

Dans ce contexte, MANDIAU et al. (1991A) considèrent trois types d'agents :

- les agents réactifs constitués par le procédé et ses environnements,
- les agents cognitifs constitués par les systèmes d'assistance,
- les agents externes constitués par le groupe d'opérateurs humains.

VI.2. STRUCTURE GENERALE

Considéré comme intégré dans un univers multi-agents (GASSER, 1991), l'agent intelligent, figure II.14, possède une structure décomposée en trois niveaux de contrôle selon MANDIAU et al. (1991A) :

- Le niveau le plus haut constitue la couche de coopération. Ce niveau correspond en fait à un module capable de raisonner sur des connaissances relatives au désir et à l'engagement d'effectuer une action, ainsi qu'aux

différents buts à atteindre. Ce module de raisonnement s'engage à réaliser le(s) but(s) sélectionné(s) soit par l'agent, soit par un autre module hiérarchiquement supérieur, en se basant sur les désirs de cet agent. Si le but est réalisable, un plan d'intention est généré, sinon le but doit être sélectionné par un autre agent. Ce plan est constitué d'intentions d'engagement et de croyances sur l'univers qui l'entoure. Ce module intègre aussi un certain nombre d'actes tels que ceux de communication ou les actes probables des autres agents.

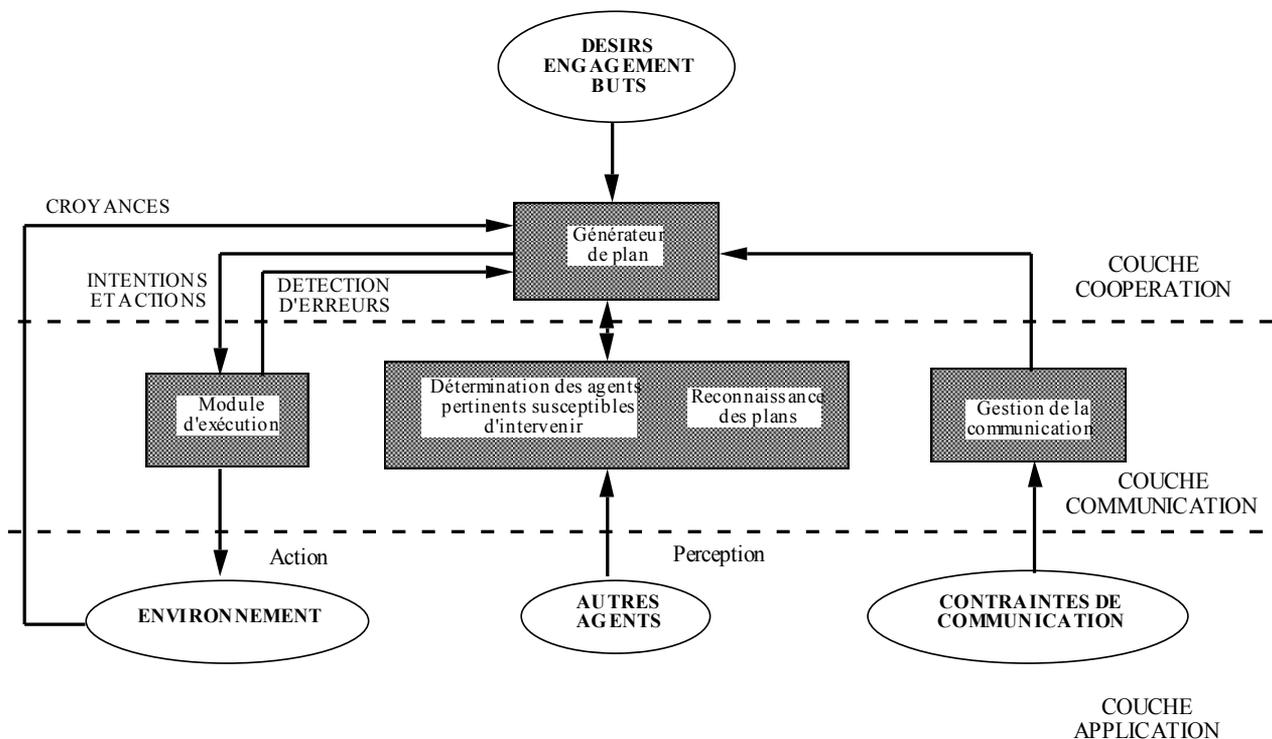


Figure II.14 : Modèle d'un agent intelligent (MANDIAU et al., 1991B)

- Le niveau médian constitue la couche de communication. Il se compose de trois processus de traitement : un module d'exécution, un module de reconnaissance des plans et un module de gestion de communication. Après l'élaboration d'un plan d'intentions, l'agent l'exécute afin de réaliser le but pour lequel il s'est engagé. Ce but est réalisé par le module d'exécution, si l'intention générée est une action. Ce but est réalisé par l'intermédiaire du module de gestion de la communication, si l'intention produite est un acte de communication vers un autre agent. Il faut noter, à ce niveau, que l'acte de communication est élaboré par le niveau le plus haut comme un plan. Par conséquent, il est considéré comme un but à réaliser. Le module central

détermine aussi les agents pertinents susceptibles d'interagir avec l'agent actuel, pour atteindre le but, afin d'éviter toute interaction inutile entre les autres agents.

- Le niveau le plus bas constitue la couche de l'application. Il est composé : (i) d'une base de données décrivant l'environnement de l'agent, (ii) d'une base de connaissance sur les autres agents constituant cet univers multi-agents et (iii) de données sur le mode et les contraintes de communication avec les différents autres agents de cet univers.

VI.3. EXEMPLE D'APPLICATION D'UN AGENT INTELLIGENT

MANDIAU et al. (1991C) décrivent un exemple de modélisation d'un agent "intelligent" dans un contexte de contrôle de procédé. Le procédé en question est un système de production d'un produit D. Ce produit est le résultat du mélange de 3 produits appelés A,B et C dans des proportions bien définies, figure II.15.

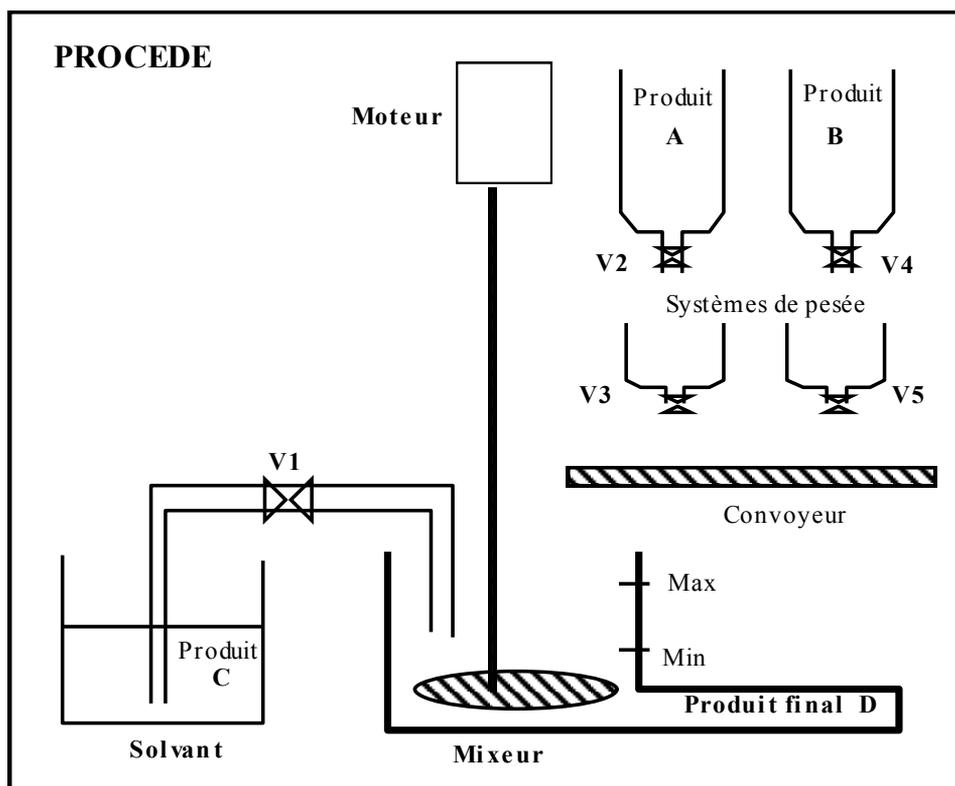


Figure II.15 : Le Mélangeur utilisé pour l'application

La décomposition de ce système homme-machine en univers multi-agents considère :

- le procédé comme un agent réactif ;
- deux systèmes d'aide comme des agents cognitifs (Assistant_1 et Assistant_2) ;
- l'opérateur humain comme un agent externe.

Le but final est de parvenir au mélange des 3 produits A, B et C pour obtenir le produit D qui sera utilisé par la suite. La mise en œuvre de ce but génère une "conscience" collective, c'est à dire un plan identique à tous les agents de cet univers. L'agent externe agit sur le premier agent cognitif en le chargeant du mélange de A et B, pendant que lui-même contrôle le niveau du solvant C. Ne pouvant se charger en même temps de la pesée et du convoyage des deux produits A et B, l'Assistant_1 demande alors à l'Assistant_2 de contrôler le convoyage des deux produits, se laissant la charge de la pesée des deux produits, figure II.16.

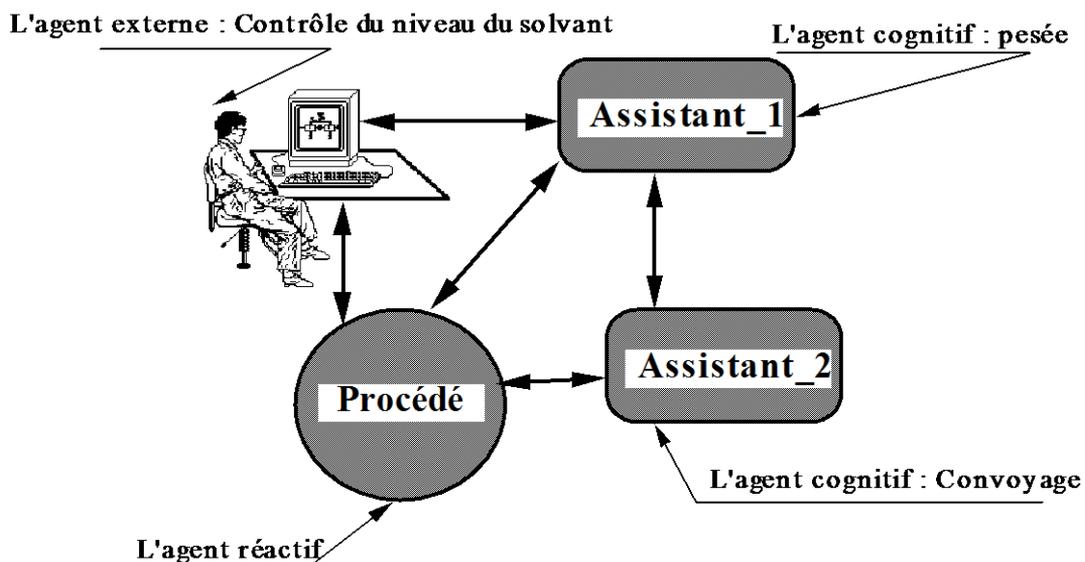


Figure II.16 : Exemple d'un univers constitué de 4 agents

Actuellement, les travaux sur l'agent "intelligent" sont en cours de mise en œuvre en LISP sur un VAX 6400.

VI.4. DISCUSSION

Cette approche d'agent "intelligent" semble prometteuse pour une modélisation d'un système homme(s)-machine(s) constitué d'agents "intelligents" interagissant entre eux, dans la mesure où les machines sont actuellement dotées de plus en plus "d'intelligence". Notons à ce sujet que cette approche multi-agents est actuellement développée dans plusieurs applications mettant en oeuvre des agents se comportant dans le contexte de tâches complexes. Par exemple, SHALIN et ses collègues (1988) développent un système visant à reconnaître les intentions d'un pilote, dans le but d'amoinrir le travail de celui-ci et d'améliorer la synergie entre homme et machine. Ce travail vise à moyen terme la reconnaissance des erreurs du pilote.

De telles approches changeront considérablement : (i) le travail de l'homme, qui aura à interagir avec un ensemble d'agents "intelligents", (ii) la conception des systèmes d'assistance qu'il s'agit de décomposer en agents autonomes, de même que (iii) les méthodes d'analyse et d'évaluation de ces systèmes.

Enfin, la complexité d'une telle approche se retrouve dans la formalisation des différentes connaissances, particulièrement quand le nombre d'agents augmente.

VII. LES INTERFACES ECOLOGIQUES

VII.1. PRINCIPE

RASMUSSEN et VICENTE (1987) ont proposé la notion d'interface écologique au travers d'une théorie visant à minimiser le potentiel de contrôle laissé à l'opérateur humain, et ceci afin de minimiser les risques d'erreurs. Il importe dans ce but de chercher à résoudre les difficultés qu'éprouve l'opérateur à comprendre le fonctionnement interne des systèmes à superviser.

Leur proposition consiste à concevoir une interface qui tire parti des ressources les plus puissantes que les hommes possèdent pour affronter des situations complexes. C'est le cas par exemple des musiciens qui, à leurs débuts, travaillent avec des notes individuelles, puis au fur et à mesure qu'il acquièrent de l'expérience, travaillent avec des séquences de notes allant des plus simples aux plus compliquées.

Parce que l'homme est plus efficace dans ses processus mentaux automatisés de bas niveau (RASMUSSEN, 1983), les auteurs déclarent qu'il est nécessaire de présenter l'information de telle sorte que l'homme ne doive que très rarement sortir de ses processus de bas niveau. A leur sens, il serait efficace de rendre l'interaction avec le système semblable à celle d'un jeu vidéo pour mettre justement à profit le type de compétence, spécifiquement humaine, que nous déployons lors de ces jeux.

VII.2. STRUCTURE GENERALE

Pour l'instant, les travaux sur les interfaces écologiques sont encore au stade théorique. Ils visent à minimiser le coût de la charge mentale induite par une tâche de contrôle, ainsi que celui de la reprise d'un défaut. La structure de cette approche repose sur les possibilités de superposer les caractéristiques du processus à contrôler, de l'interface et du modèle mental de l'opérateur dans des applications typiques de contrôle de processus (RASMUSSEN et VICENTE, 1989), et ceci dans le but de rendre visible à l'opérateur le fonctionnement du procédé qu'il contrôle. Pour cela, les auteurs se basent sur 3 critères principaux :

- (1) Synthétiser le contrôle et l'observation de surface, de sorte que les interactions soient suffisamment courtes pour être caractérisées comme des signaux.
- (2) Forcer la machine à traduire la tâche en créant une correspondance fidèle entre l'invisible (c.à.d. l'ensemble des variables d'état du procédé considéré, ainsi que le résumé de leurs propriétés), et les répliques ou signes générés par l'interface.
- (3) Afficher la structure relationnelle entre tous les composants du procédé, afin d'extérioriser le modèle mental qui va supporter les traitements basés sur la connaissance.

VII.3. EXEMPLE D'INTERFACE ECOLOGIQUE

Des premiers essais d'assistance au lecteur dans une bibliothèque publique ont été réalisés : chaque lecteur peut consulter un système informatique dans le hall de la bibliothèque.

Pour illustrer le principe de rendre visible à l'utilisateur de ce système ce qui est invisible, il est possible de prendre l'exemple de la recherche d'une livre particulier en utilisant le système d'assistance (PEJTERSEN et RASMUSSEN, à paraître). Dans ce cas, la structure invariante du domaine de travail est représentée sous forme d'une métaphore, familière à l'utilisateur, qui exploite les ressources de l'utilisateur et ses capacités d'identifier le lieu dans lequel il se trouve. L'interface doit présenter à l'utilisateur des écrans de dialogue, très cohérents, d'un réseau très riche en information, dans lequel l'utilisateur peut naviguer librement et sans être contraint par le système.

Dans ce type d'application, deux décideurs ont été prévus : l'utilisateur en tant que demandeur d'un renseignement, et "l'intermédiaire" qui se charge de reformuler les demandes de l'utilisateur sous une forme compatible avec le système de gestion de la base de données de la bibliothèque.

Si l'utilisateur demande à accéder, par exemple, aux livres de science-fiction, il lui est possible d'identifier l'emplacement exact de ce type d'ouvrage, de sélectionner un livre et de lire ses caractéristiques sans quitter son écran de contrôle et avec une facilité adaptée à chaque utilisateur. Pour cela, l'interface propose à l'utilisateur un

certain nombre de stratégies de recherche lui permettant de naviguer artificiellement dans la bibliothèque. Il dispose d'un réseau d'images, illustrant l'ensemble de la bibliothèque, lui permettant de commencer sa recherche du hall de la bibliothèque. L'utilisateur peut ainsi choisir, dans un premier temps, la salle qui traite du domaine de la science-fiction. Dès qu'il accède à cette salle, dans laquelle il peut se déplacer comme s'il y était réellement, il peut consulter les différents livres, qui sont placés **virtuellement** devant lui. Il faut noter que, tout au long de sa recherche, un historique des différents tâches qu'il a effectuées reste présent en haut de son écran, Figure II.17.

Il faut remarquer que l'exemple de ce type d'interface utilise des techniques hypermédias.

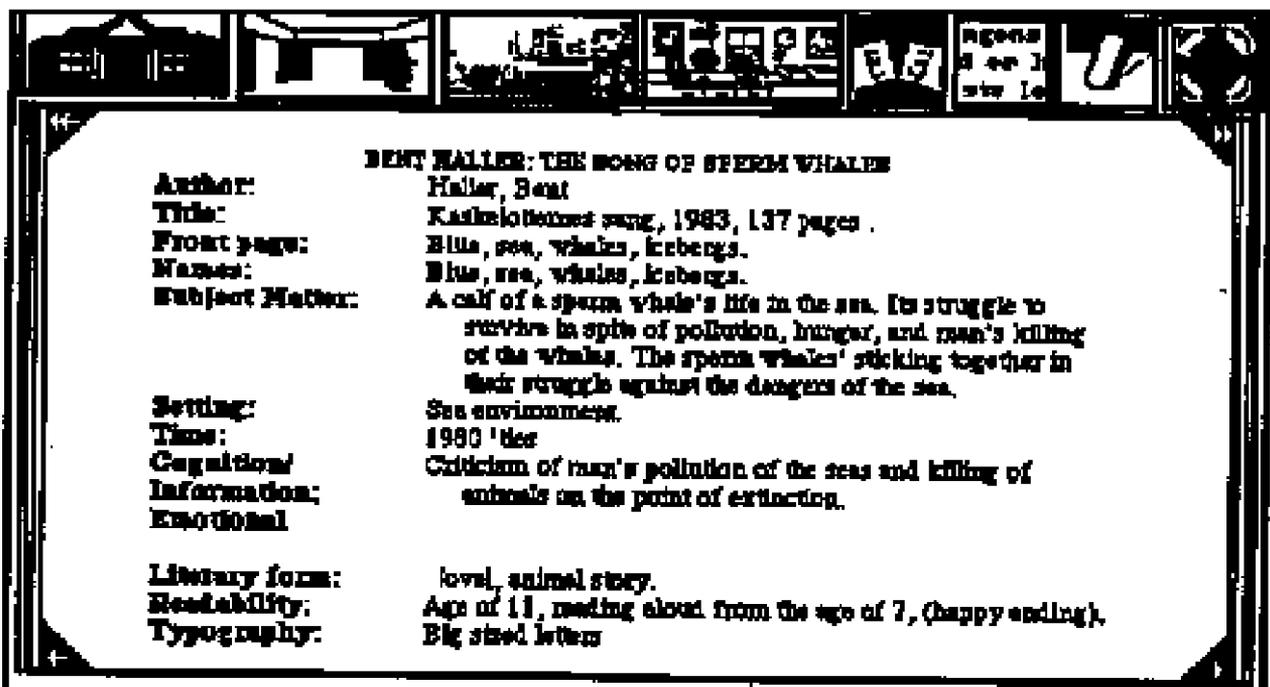


Figure II.17 : Page-écran de description d'un livre, rappelant l'ensemble du trajet et des actions effectuées pour arriver à celui-ci

VII.4. DISCUSSION

Le notion d'interface écologique, que l'on rencontre de plus en plus dans la littérature, fait allusion à une "philosophie" prenant une grande part dans les travaux de RASMUSSEN et VICENTE. Elle peut être décrite comme un essai pour rendre l'interface transparente vis-à-vis de l'utilisateur humain, afin de garder le niveau de charge mentale de l'utilisateur constant et éviter ainsi toute surcharge de travail et certains risques d'erreurs qui en découlent.

En résumé, ce type de système doit s'adapter aux tâches cognitives de l'utilisateur, de sorte que les informations qu'il perçoit lui permettent d'utiliser le plus souvent le niveau le plus bas dans la classification de tâche cognitives de l'utilisateur humain (RASMUSSEN, 1983) : ce niveau correspond aux activités quasi-reflexes, de la même façon que lorsque nous sommes en train de jouer à un jeu vidéo.

De nombreuses recherches restent encore à effectuer pour parvenir à adapter le concept d'interface écologique au contrôle de procédé industriel.

VIII. LES INTERFACES HYPERMEDIA

VIII.1. PRINCIPE

Bien que les concepts de base aient déjà été en 1945 décrits par BUSH², c'est Ted NELSON, en 1960, qui invente le mot "*Hypertexte*", qu'il définit comme "l'association d'un langage naturel avec les possibilités qu'a l'ordinateur d'établir des liaisons interactives et des affichages dynamiques d'un texte non linéaire". A l'origine des systèmes hypermédias, on retrouve le projet **Xanadu** (NELSON, 1965), dont le but est de définir les bases d'un réseau planétaire permettant le rassemblement de l'ensemble des connaissances humaines dans un vaste document. Ce document doit être accessible à "l'homme de la rue" à tout moment, en se référant à la pensée intuitive et sans se plier aux modes d'accès de l'ordinateur, comme dans un livre, mais avec des illustrations visuelles et sonores. L'exploitation de ce document se fait selon les besoins de l'utilisateur, ses connaissances, son expérience, ses désirs, son imagination, etc.

Quand l'accent est mis pour un système d'information sur des "médias" autres que le texte, en rajoutant le son et/ou l'image, on parle alors de système hypermédia ou encore multimédia (NIELSEN, 1990). La possibilité d'inclure, dans un hypertexte, de l'animation, des images vidéo, des sons, etc., n'en change pas les principes de base. Par contre, ceux-ci sont plus difficiles à mettre en œuvre et se heurtent à des problèmes techniques et conceptuels.

VIII.2. STRUCTURE GENERALE

Les composants d'un système hypermédia sont les *nœuds* et les *liens*. Les nœuds sont des contenants d'information. Ceux-ci sont considérés, par l'utilisateur qui les crée, comme des entités sémantiques distinctes les unes des autres. Un lien, quant à lui, représente une relation entre des nœuds ou entre les informations qu'ils contiennent.

² Douglas Engelbart a été en fait le premier scientifique influencé par les concepts de BUSH. Ses travaux à l'Institut de Recherche de Stanford ont porté notamment sur la réalisation d'un système d'information visant à accroître les capacités de l'intelligence humaine (ENGELBART, 1963).

Un système hypermédia est particulier, par sa structure, car si les commandes d'une interface classique sont souvent placées dans des menus, forçant ainsi l'utilisateur à les mémoriser, pour un système hypermédia elles sont placées directement dans le texte ou les graphiques, constituant ainsi deux sortes de structures dans le réseau de nœuds et de liens :

- La structure primaire, induite par l'organisation de l'ensemble des informations, correspondant à la structure logique du document.
- Les structures secondaires, prenant en compte les liens de référence entre les nœuds, constituent ainsi l'ensemble des stratégies de navigation dans le document.

La particularité de la structure d'un système hypermédia permet à son utilisateur le choix d'une ou de plusieurs stratégies de navigation parmi les informations. A cet effet, VAN DYKE PARUNAK (1989) propose un ensemble de stratégies de navigation utilisables dans un système hypermédia, dont certaines sont visibles sur la figure II.18 :

- Une stratégie linéaire, où l'opérateur passe d'un nœud à l'autre comme on tourne les pages d'un livre.
- Une stratégie en anneau, identique à la précédente, sauf que dans ce cas, la fin du document correspond à son début.
- Une structure hiérarchique, où l'opérateur peut cerner progressivement l'information qu'il recherche. C'est le principe utilisé dans une bibliothèque, où le lecteur choisit le domaine qui l'intéresse, puis le thème, la matière, la rubrique jusqu'à obtenir le détail sur l'information qu'il souhaite.
- Une structure en hypercube, où les nœuds forment une maille dans laquelle l'opérateur peut passer facilement d'un nœud à un autre.
- etc.

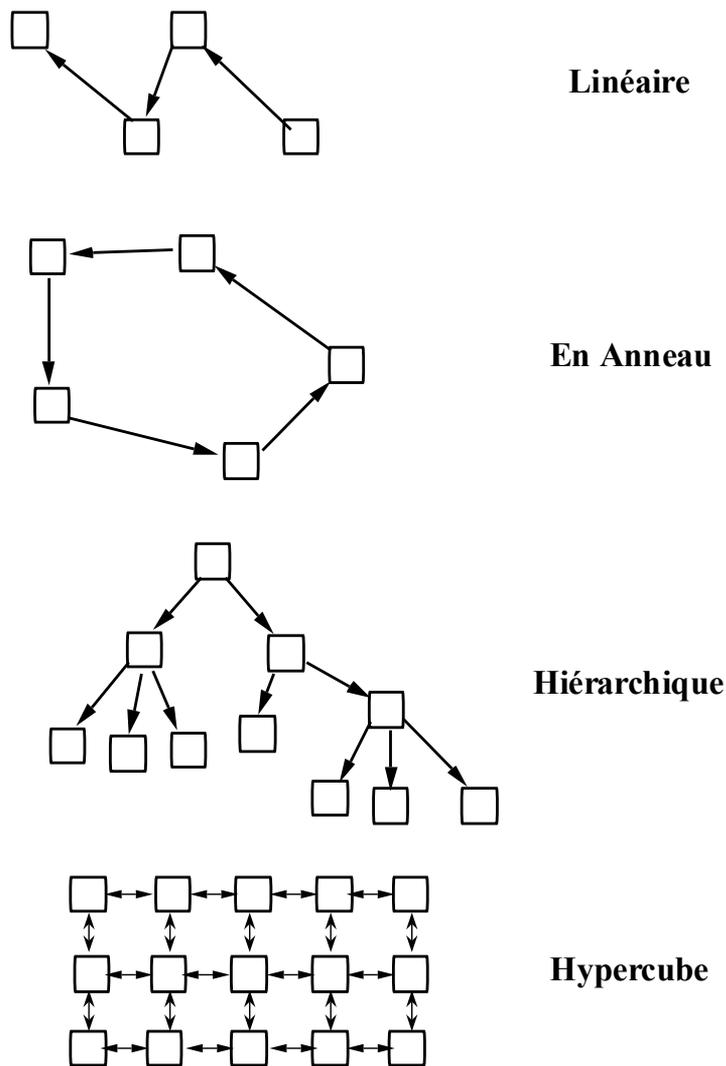


Figure II.18 : Exemples de stratégies de navigation dans un système hypermédia

L'utilisateur d'un système hypermédia possède globalement deux méthodes pour explorer un document :

- La première méthode consiste en l'utilisation des liens, lorsque l'opérateur sait où se trouve l'information qu'il cherche et désire en connaître la teneur, comme c'est le cas par exemple lors de l'utilisation d'un dictionnaire.
- La seconde méthode consiste en la spécification par l'opérateur de l'information qu'il recherche afin de la localiser, comme c'est le cas pour l'utilisation de l'index d'un livre.

Enfin il faut souligner qu'un système hypermédia offre généralement à ses utilisateurs un *browser* correspondant à un schéma synoptique du cheminement dans

le système. En effet, si le document possède un grand nombre de nœuds et de liens, l'opérateur peut se sentir désorienté, car il est projeté dans un hyper-espace et peut ressentir le besoin de visualiser sa position par rapport à la structure globale du document, ainsi que le chemin parcouru.

VIII.3. EXEMPLE D'INTERFACE HYPERMEDIA

Bien que des outils hypermédias soient de plus en plus utilisés pour la spécification et le maquetage des interfaces homme-machine (voir JACOB, 1985 ; MINCH, 1989 ou WELLS, 1989), le plus grand nombre d'applications hypermédia se trouve dans le domaine de la gestion de documents de plusieurs types : Textes, graphiques, images, sons, etc.

Par exemple, NoteCards (HALASZ et al., 1987) est un système hypermédia fonctionnant avec InterLisp sur une machine LISP. L'idée de ses concepteurs est de proposer aux utilisateurs des fonctionnalités génériques qu'ils peuvent personnaliser et adapter à leurs besoins. Pour cela, ils fournissent une interface de programmation qui connaît actuellement beaucoup de succès. D'après ses concepteurs, NoteCards semble être adapté au domaine de l'acquisition des connaissances, dans la mesure où il permet à un expert de décrire sa connaissance sous la forme de réseaux de concepts. A ce sujet, le lecteur se référera au système d'acquisition de connaissances réalisé par Mc ALEESE (1990) à l'aide de NoteCards. Quelques exemples de concepts et de liens proposés par NoteCards sont présentés dans le tableau II.1 : avec l'ensemble de ces concepts et liens, l'expert peut transcrire plus facilement ses connaissances et établir explicitement des liens entre ses granules.

Tableau II.1 : Exemple de concepts et de liens utilisables avec l'outils NoteCards

Concepts (nœuds)	Liens
<ul style="list-style-type: none">• Procédés• Propriétés• Produits• Agents	<ul style="list-style-type: none">• étant• montrant• ayant• incluant• utilisant• causant

Des systèmes hypermédias sont déjà exploités dans de nombreux domaines aussi variés que celui de l'aide à la décision (MINCH, 1989), les livres électroniques (VERCOUSTRE, 1989), la création d'environnement de programmation (GOLDSTEIN et BOBROW, 1987) ou comme l'acquisition de connaissances (WELLS, 1989). Dans le domaine de l'assistance à l'opérateur, citons le projet CID (Computer Integrated Documentation) à la NASA visant à réaliser des systèmes hypermédias collaborant avec des opérateurs humains pour résoudre des problèmes opérationnels (BOY, 1990).

VIII.4. DISCUSSION

les systèmes hypermédias vivent leurs premières années. Mais, avec le développement des technologies de traitement du signal, au niveau du son et de l'image, ceux-ci ne peuvent que s'améliorer. Il existe de nombreux systèmes opérationnels. Les plus connus sont certainement NoteCards (déjà cité), Intermedia (YANKELOVICH et al., 1985), Guide (BROWN, 1987) et surtout Hypercard qui a eu le mérite de faire découvrir l'hypertexte au "grand public". Le lecteur trouvera dans l'article de BEGORAY (1990) une synthèse sur les systèmes actuels.

On les retrouvera sans doute bientôt dans les salles de contrôle de procédé industriel (KOLSKI, MILLOT, 1990 ; ALTY, BERGAN, 1992). En effet, leur utilisation peut être envisagée pour la navigation parmi des représentations des composants du procédé, plus réalistes que ne le permettent les synoptiques actuels. Une centralisation de modes opératoires accompagnés de conseils sous forme vidéo sera aussi sans doute bientôt possible.

La conception et l'évaluation de tels systèmes en salle de contrôle nécessiteront des recherches spécifiques.

CONCLUSION

L'évolution technologique au niveau des interfaces homme-machine a fait un bond considérable, un des rôles d'une interface étant de combler les lacunes qui peuvent exister entre l'homme et la machine (CARD, 1988 ; HANCOCK et CHIGNELL, 1987 ; 1988). La plupart des systèmes de dialogue, présentés dans ce chapitre ont un objectif commun, qui est de **s'adapter** soit à l'opérateur humain soit au procédé. Cependant, on retrouve pour chacun d'eux, un mode d'adaptation particulier, tableau II.2 :

Tableau II.2 : Synthèse sur les systèmes de dialogue et leurs styles d'adaptation

Type de système de dialogue	Type(s) d'adaptation
Le système hypermédia	à l'inspiration et aux stratégies de recherche de l'opérateur humain quand celui-ci cherche à s'informer
L'interface flexible	au système à superviser et aux préférences individuelles des opérateurs dans les styles d'interaction
L'interface tolérante aux erreurs humaines	aux actions de l'opérateur humain
L'interface adaptative	aux traitements cognitifs de l'opérateur humain et aux erreurs de celui-ci
L'interface experte	aux connaissances de l'opérateur humain
L'interface écologique	aux ressources cognitives les plus profondes de l'opérateur humain
L'opérateur assistant	aux tâches cognitives de l'opérateur humain
L'agent intelligent	aux comportements de l'opérateur humain, tels que ses croyances, ses intentions ou ses désirs

Comme nous l'avons vu dans ce chapitre, l'opérateur assistant peut être potentiellement caractérisé comme une généralisation de l'interface adaptative, figure II.19. L'interface adaptative est une généralisation de l'interface experte qui elle-même est potentiellement un sur-ensemble des interfaces flexibles et tolérantes aux erreurs humaines.

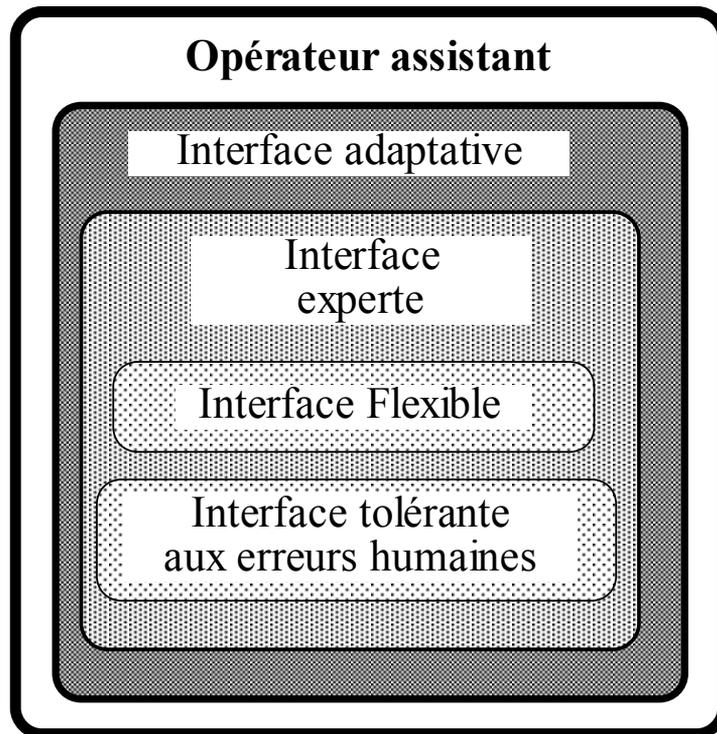


Figure II.19 : Généralisation des systèmes de dialogue

En résumé, un système de dialogue idéal devrait regrouper les différents concepts décrits dans ce chapitre. Cependant, sa conception hériterait alors des problèmes de chaque type de système de dialogue.

Actuellement, des recherches tendent à caractériser un système de dialogue, se rapprochant de l'interface idéale, comme une **interface "intelligente"**.

ROUSE et al. (1987, 1988) présentent une architecture fonctionnelle d'un système d'aide incluant une interface "intelligente", qui comprend un certain nombre de fonctionnalités. Celle-ci est issue de l'expérience de son équipe de recherche dans le domaine de la communication homme-machine. Cette architecture est basée sur :

- un système se chargeant de traiter les erreurs de l'opérateur humain ;

- un système se chargeant d'adapter l'aide à chaque opérateur ;
- un modèle de l'opérateur se chargeant de la gestion des intentions, des ressources et des performances de l'opérateur humain.

D'autres chercheurs caractérisent une interface "intelligente" comme le type d'interface qui intègre des outils d'aide pouvant minimiser la distance cognitive entre le modèle que l'opérateur se fait de la tâche et l'apparence que cette tâche possède à travers le système informatique, quand celle-ci est exécutée (HANCOCK et CHIGNELL, 1989). Selon le point de vue de CHIGNELL et HANCOCK (1989), "une interface intelligente est une entité intelligente qui joue le rôle de médiateur entre deux ou plusieurs agents interactifs, qui possèdent une compréhension incomplète des connaissances ou de la forme de communication de l'un ou de l'autre".

Puisque l'interface "intelligente" est par définition un système "intelligent", elle hérite de toutes les contraintes de ces systèmes. De ce fait, elle doit être conceptuellement dirigée par les buts à remplir, pour faciliter la focalisation sur le problème, mais doit être aussi dirigée par les données afin d'alerter l'opérateur des événements non perçus (NORMAN, 1979). Elle doit être autonome, en ce sens qu'elle doit être capable de surveiller ses actions, ses buts recherchés ainsi que ses intentions.

Notre contribution réside dans le développement d'une interface "intelligente". Celle-ci a été appelée Module Décisionnel d'Imagerie (M.D.I.), puisqu'elle se ramène à un module informatique capable de décider du contenu des images à présenter à l'opérateur. Ainsi, nous proposons d'utiliser un système expert pour assurer les interactions entre le procédé, l'opérateur humain et les différents outils d'assistance dont il dispose (TENDJAOUI et al. 1990). Notre approche se propose de résoudre les problèmes d'interaction entre l'homme et la machine en répondant aux trois questions suivantes : "QUOI ?", "QUAND ?" et "COMMENT ?", c'est à dire : quelles informations l'interface doit-elle présenter à l'opérateur, quand les lui présenter selon les modes de fonctionnement du procédé et sous quelle forme les lui présenter. Par la réponse à ces questions, notre approche vise à s'adapter, notamment, aux tâches cognitives de l'opérateur, à son expérience et à ses ressources.

Le M.D.I. est une interface de type adaptative, figure II.20. Cependant, si nous associons au M.D.I. l'ensemble des fonctionnalités d'assistance dont doit disposer l'opérateur, il sera très proche du système de dialogue de type "Opérateur assistant".

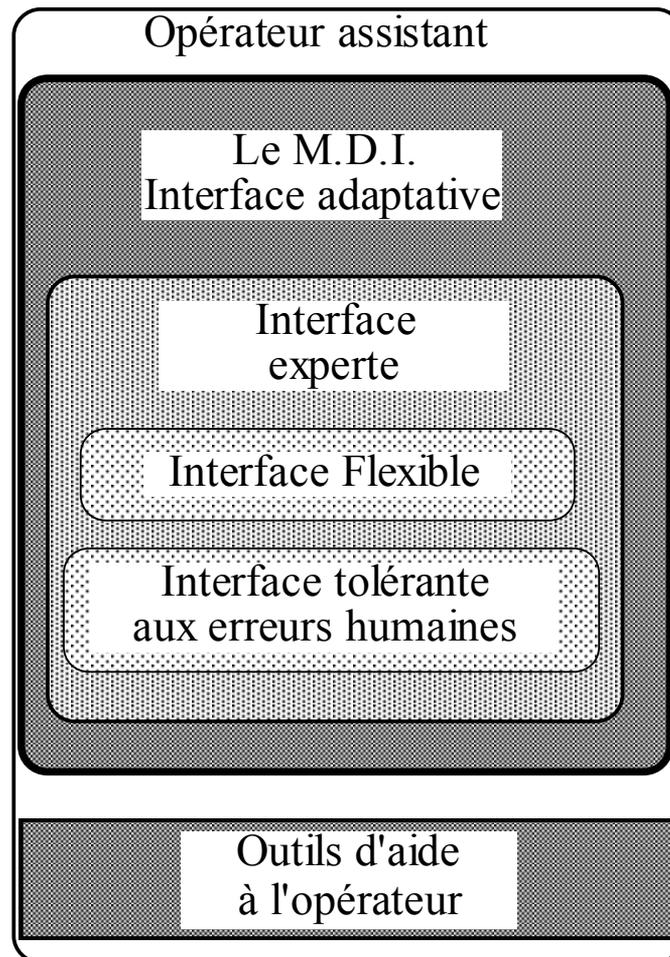


Figure II.20 : Généralisation des systèmes de dialogue

Le chapitre suivant est consacré justement à la présentation du principe et des caractéristiques du M.D.I.

Chapitre III

UNE APPROCHE D'INTERFACE "INTELLIGENTE" : LE MODULE DECISIONNEL D'IMAGERIE (Le M.D.I.)

CHAPITRE III : UNE APPROCHE D'INTERFACE "INTELLIGENTE" : LE MODULE DECISIONNEL D'IMAGERIE (LE M.D.I.)

INTRODUCTION

Nos travaux de recherche ont pour origine l'étude et la réalisation de l'interface entre un système expert temps réel d'aide à la décision et des opérateurs humains, dans le cadre d'un projet national appelé ALLIANCE auquel le LAIH avait participé (TABORIN, POULAIN, MILLOT, 1986 ; 1988 ; 1990). Le travail mené, les suggestions et les remarques soulevées dans le cadre de ce projet nous ont poussé à améliorer l'approche utilisée et à nous orienter vers une interface dite "intelligente", basée sur certains principes recensés dans le chapitre II.

Cette interface "intelligente" est construite autour d'un module informatique appelé "Module Décisionnel d'Imagerie" (M.D.I.). Celui-ci a pour objectif de résoudre un ensemble de problèmes liés aux interactions entre l'homme et la machine selon les trois aspects ergonomiques suivants : (1) QUOI présenter à l'opérateur, (2) QUAND lui présenter l'information et (3) COMMENT la lui présenter. Nous utilisons pour cela une approche d'Intelligence Artificielle (TENDJAOUI et al., 1990) pour assurer les interactions entre l'opérateur humain, le procédé et les différents outils d'assistance dont il dispose. Le M.D.I. peut être considéré comme une interface avancée située à la frontière entre l'interface "adaptative" au sens d'EDMONDS (1981), l'interface "flexible" au sens de WILLIGES et al. (1987) et l'interface "experte" au sens de d'ALTY et GUIDA (1985). En effet, le M.D.I. vise à s'accorder (i) avec le premier type d'interface quand à la prise en compte des traitements cognitifs de l'opérateur humain et des erreurs de celui-ci, (ii) avec le second, pour sa capacité à s'adapter au système à superviser ainsi qu'aux préférences individuelles des opérateurs dans les styles d'interaction et enfin (iii) avec le troisième, en ce qui concerne l'intégration de connaissances relatives au procédé et à l'utilisateur du système.

Dans ce chapitre, après avoir résumé les travaux antécédents au M.D.I., nous situons celui-ci dans la boucle de supervision, afin de mettre en évidence les différents systèmes avec lesquels il est en relation. Par la suite, les objectifs du

M.D.I. sont présentés de même que ses concepts principaux : le "QUOI", le "QUAND" et le "COMMENT". Enfin, une architecture fonctionnelle intégrant le M.D.I. est décrite dans ce chapitre.

I. TRAVAUX ANTECEDENTS AU M.D.I.

Le M.D.I. constitue la suite des travaux développés dans le cadre de l'étude et de la réalisation d'une interface homme-machine lors du projet ALLIANCE. Ce projet avait pour objectif la réalisation d'un système expert temps réel d'assistance à l'opérateur pour la conduite de procédés continus (DE LAPPARENT, 1988), dans la salle de contrôle du Centre d'essais Technologique situé à Cadarache. Ce site permet de tester les échangeurs de chaleur sodium-eau pour les réacteurs nucléaires. Ce projet a servi de support d'application aux recherches menées au LAIH sur les interfaces entre opérateur et système d'Intelligence Artificielle temps réel.

L'interface développée lors de ce projet est capable d'apporter une aide à l'opérateur pendant les deux situations de fonctionnement possibles du procédé : le fonctionnement normal et le fonctionnement anormal. Des vues spécifiques ont été réalisées en fonction des besoins informationnels de l'opérateur dans chacune de ces deux situations, et de l'aide que le système expert est susceptible d'apporter (TABORIN, POULAIN, MILLOT, 1986 ; 1988 ; 1990 ; TABORIN 1989).

Pendant le fonctionnement normal, les vues disponibles permettent à l'opérateur d'effectuer une tâche de surveillance. Cette tâche consiste à évaluer continuellement l'état du procédé de façon à détecter rapidement l'apparition d'un défaut. La surveillance se fait sur un ensemble restreint de variables (sept), les plus significatives de l'état de fonctionnement du procédé, définies par consultation de l'expert. Une vue synthétique regroupant ces sept variables, appelée vue ETOILE a été réalisée, selon le principe défini par CÆKIN (1969) et repris par WOODS et al. (1981), figure III.1.

La vue étoile comprend les consignes et les six seuils d'alarme de chaque variable (seuil haut, seuil très haut, seuil d'alarme haute, seuil bas, seuil très bas, seuil d'alarme basse), disposés respectivement sur sept rayons régulièrement espacés d'un même cercle. Les valeurs des consignes et des seuils ont été normées de façon à ce qu'elles figurent sur des cercles concentriques. La valeur de chaque variable est représentée par un point sur le rayon dont la position est modifiée en fonction de

l'évolution temporelle de l'amplitude de la variable. Les points symbolisant chaque valeur de variable sont reliés entre eux par des segments de droite de telle façon qu'en fonctionnement normal, où les variables sont égales à leurs consignes respectives, la figure soit un polygone régulier, figure III.1(A). Lorsqu'une variable s'écarte de sa consigne, le polygone se déforme indiquant un dysfonctionnement, figure III.1(B).

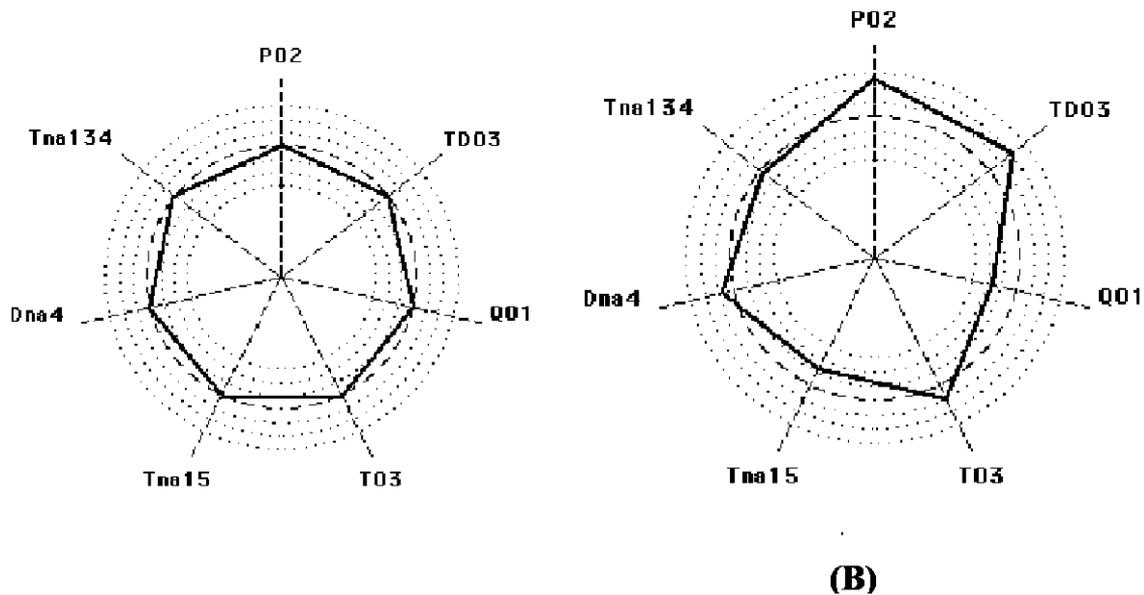


Figure III.1 : La vue Etoile

Cette déformation du polygone permet d'avertir l'opérateur de l'apparition d'un défaut. Il doit ensuite effectuer une tâche de diagnostic puis de correction. Après la détection du défaut, le système d'Intelligence Artificielle effectue un raisonnement aboutissant à la définition d'une procédure de reprise et de conseils d'actions qui sont, automatiquement, affichés dans une fenêtre de l'écran, selon l'exemple présenté en figure III.2.

Conseils de PREVENTION	
Variable	: Dna4
Action	: Upem03
Sens	: -1
Avant Tp	+ 2:20
Variable	: Tna15
Action	: C-P02
Sens	: +1
Avant Tp	+ 3:30

Figure III.2 : Conseils d'actions

Ces conseils d'actions sont affichés dans une fenêtre de l'écran et présentent :

- la variable à corriger (sur cet exemple, deux conseils sont donnés concernant respectivement les variable Dna4 et Tna15),
- commande à actionner pour y parvenir (Upem03 et C-P02),
- le sens et l'amplitude de la correction, (-1 et +1), et
- le délai d'intervention maximum pour effectuer la correction (2 mn 20 s et 3 mn 30 s).

Parallèlement, l'opérateur aura pu effectuer son propre raisonnement aboutissant lui aussi à ses propres actions, où deux situations sont prévues :

- l'opérateur est en accord avec les conseils d'actions proposés, et il n'a plus alors qu'à agir sur le procédé. Le système expert le conforte donc dans sa décision;

- l'opérateur n'est pas d'accord avec les conseils proposés, ou encore il n'a pas réussi à définir une procédure, et dans ces deux cas il souhaite des explications. A cet effet, deux niveaux de justification sont mis à sa disposition par l'interface homme-machine :

- Le premier niveau de justification répond au désir d'explications de l'opérateur sur les conseils d'actions. Dans ce cas, il peut accéder à une vue, appelée "vue de propagation", qui présente à l'opérateur le diagnostic du défaut. Sur un schéma simplifié de l'installation, sont affichées les variables importantes du procédé, reliées entre elles par des arcs orientés symbolisant la propagation du défaut depuis son origine jusqu'aux variables concernées par les conseils de prévention, selon le principe visible sur la figure III.3. Si l'opérateur demande des explications sur les propagations prévues, ou s'il veut disposer d'informations complémentaires notamment sur l'amplitude de la propagation, il peut alors accéder à un deuxième niveau de justification décrit ci-après.

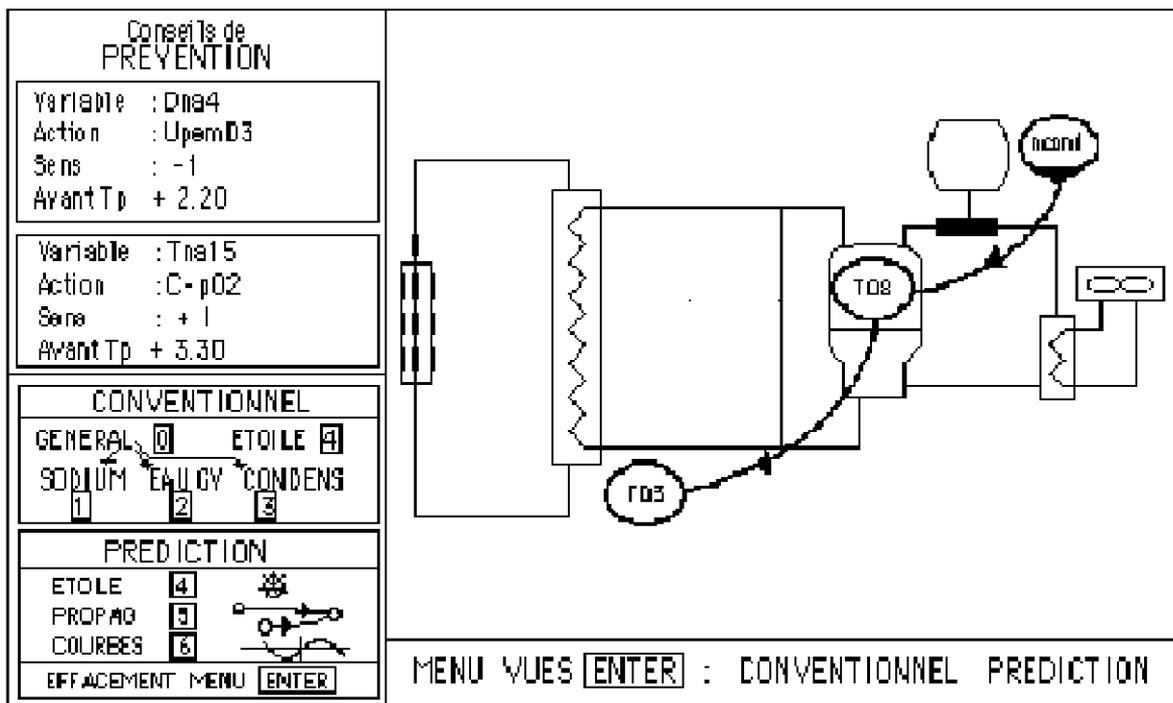


Figure III.3 : Premier niveau de justification : la vue de "propagation"

- Le deuxième niveau de justification se présente sous la forme d'une vue de type "courbe" présentant les variables les plus significatives de la prédiction faite par le système expert. La présentation d'une variable englobe en fait un ensemble de trois courbes, figure III.4. La première est l'historique, qui représente l'évolution passée de la variable. La deuxième courbe est la courbe de prédiction. Cette courbe représente la prédiction faite sur la variable depuis l'instant T_p . La troisième courbe est la courbe de suivi et correspond à l'évolution réelle de la variable entre deux prédictions. Elle est affichée pour permettre à l'opérateur de vérifier l'effet de sa correction.

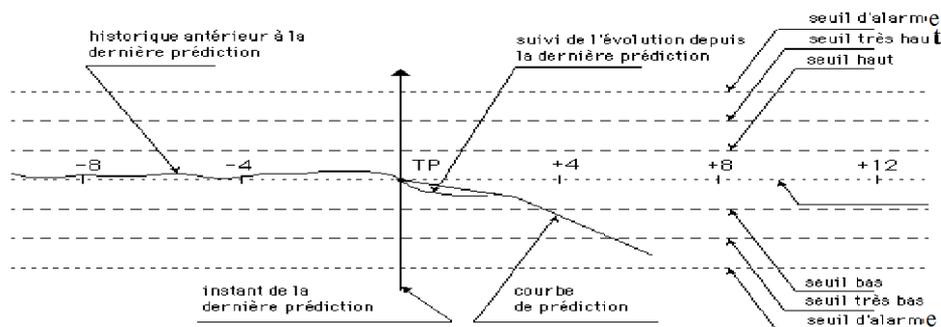


Figure III.4 : Deuxième niveau de justification : la vue "courbes"

L'étude de l'interface homme-machine, dans le cadre de ce projet, a permis de mettre en œuvre un **modèle de coopération** entre l'opérateur et le(s) module(s) d'aide à la décision, figure III.5. Ce modèle met en parallèle, d'une part les étapes de résolution de problème humaine, proposées par RASMUSSEN (1980), et les images correspondant à ces étapes d'autre part. Pendant le fonctionnement normal du procédé, le système expert ne délivre aucune information. Cependant, afin d'aider l'opérateur à effectuer les tâches de surveillance, la vue étoile lui est proposée. En cas de fonctionnement anormal, la vue étoile se déforme, et le système expert effectue un raisonnement qui aboutit à l'affichage des deux conseils d'actions les plus urgents. La présentation de ces conseils constitue l'étape de vérification. L'évolution prévue sur les sept variables constituant l'étoile, est alors décrite en accéléré, à partir d'une animation. S'il y a conflit décisionnel entre le système expert et l'opérateur humain ou si celui-ci souhaite des informations supplémentaires, il peut alors appeler les deux niveaux de justification :

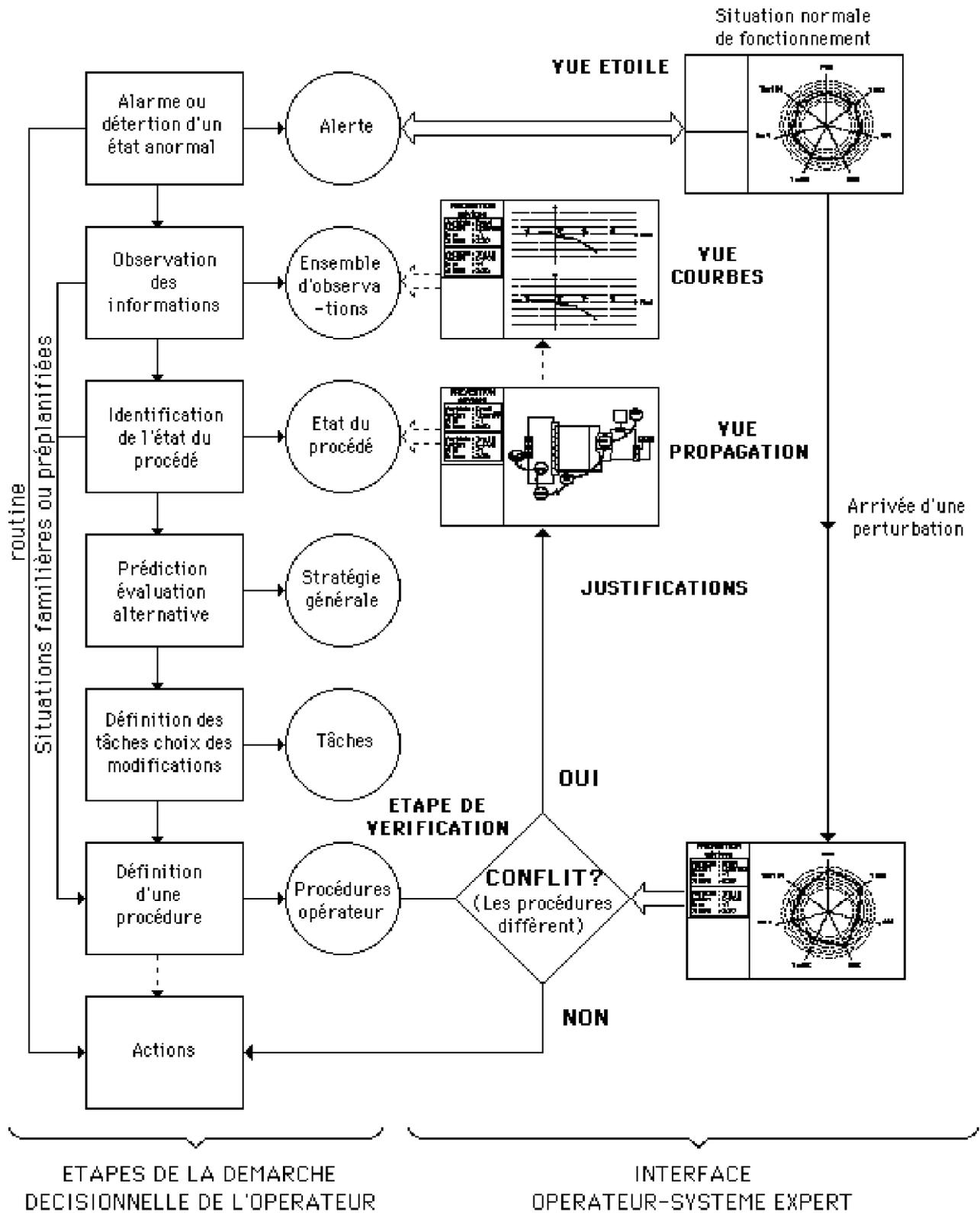


Figure III.5 : Le modèle de coopération homme-machine du synoptique ALLIANCE (TABORIN, 1989)

- **1er niveau** : La vue propagation est affichée, en remplacement de l'étoile et permet à l'opérateur d'évaluer l'étendue de la propagation prévue. L'objectif est de faciliter, à partir d'une vue très symbolique, le diagnostic rapide de l'origine de la perturbation en présentant l'ensemble des variables impliquées.
- **2ème niveau** : L'opérateur peut ensuite appeler la vue courbe affichant l'évolution des variables les plus significatives. L'objectif de cette vue est de lui permettre d'évaluer le bien fondé de la prédiction donnée et d'observer l'amplitude de cette prédiction sur les variables affichées.

L'interface homme-machine a fait l'objet d'une évaluation ergonomique préalable en laboratoire (KOLSKI, 1989 ; TABORIN, 1989) et avec les opérateurs. Ainsi, dans la mesure du possible, il a été tenu compte du fait que l'opérateur a un rôle important à tenir lors du développement de l'interface (BERRY et BROADBENT, 1986), alors que, dans la plupart des réalisations pour lesquelles il est pourtant le principal utilisateur, sa participation est étrangement rare. On peut souligner ainsi que si le concepteur de l'interface connaît bien les différents modes possibles de présentation des informations, il ne sait pas toujours vraiment ce que l'utilisateur en attend. Sur ce point, plusieurs remarques ont été formulées par les opérateurs lors de ce projet. Par exemple :

- Une des exigences premières de l'opérateur est de pouvoir voir "vivre son procédé" au travers des vues développées. Ainsi dans certains cas, il suggère que le système "force" l'affichage d'une vue même si cela peut paraître paradoxal. Bien souvent, il justifie ce souhait par le besoin d'être averti rapidement de l'apparition d'un nouvel événement important (TABORIN, 1989).
- De plus, il faut s'attacher à représenter de façon simple les informations. Notamment, il n'est pas toujours judicieux de rechercher une représentation proche de l'image physique du procédé, car bien souvent cela ne permet pas de présenter au mieux une information. Par exemple, dans le dessin en coupe d'une cuve, l'animation d'une surface colorée symbolisant le volume de liquide intérieur permet difficilement de représenter l'évolution du niveau de liquide, même en utilisant des dégradés de couleur. Dans ce cas, il est préférable d'utiliser, d'après les opérateurs, une représentation sous forme de courbe.

- Il faut aider l'opérateur à évaluer rapidement la gravité d'une situation. A cet effet, il est par exemple souvent utile de prévoir des seuils de "pré-alarmes" sur certaines variables. Concernant la notion de gravité, celle-ci sera traitée en détail dans le chapitre IV, paragraphe II.4. Dans ce paragraphe, nous proposerons une fonction de gravité dépendant de trois paramètres : le degré d'insécurité du système, le degré d'anormalité de la production et le degré d'urgence.

L'interface homme-machine développée lors de ce projet est basée sur un modèle, dit **implicite**, de coopération, dans la mesure où le modèle de raisonnement humain utilisé est un modèle de représentation qui n'est pas explicitement implanté dans l'interface. De ce fait, le concepteur ne peut "qu'espérer" a priori que l'opérateur accède à certaines vues graphiques en fonction de la situation, anormale ou normale, de fonctionnement du procédé. Cependant, ces vues ne sont pas utilisées de manière systématique par l'opérateur, et conformément à ce qu'elles sont sensées apporter à celui-ci d'un point de vue ergonomique.

Dans le but d'optimiser leur utilisation par l'opérateur, une approche d'interface "intelligente" mise en œuvre sous forme d'un modèle de coopération homme-machine représenté **explicitement** à l'aide de règles de production. Ce modèle est géré par un système expert de dialogue "évolué" et est appelé M.D.I., pour Module Décisionnel d'Imagerie (TABORIN et al., 1987). Sa localisation dans le système de supervision, ainsi que ses objectifs, font l'objet de la partie suivante.

II. LE MODULE DECISIONNEL D'IMAGERIE (M.D.I.)

Le Module Décisionnel d'Imagerie est un système "intelligent" de gestion de l'imagerie en dynamique. Cependant, afin de clarifier ses objectifs, il importe dans un premier temps de localiser un tel système dans la boucle de supervision d'un procédé industriel complexe.

II.1. LOCALISATION DU M.D.I. DANS LA BOUCLE DE SUPERVISION

L'intégration du M.D.I. dans la chaîne de supervision est présentée en figure III.6 : le calculateur de supervision centralise l'ensemble des données instrumentées du procédé. Ces données sont accessibles au(x) module(s) informatique(s) d'aide à la décision et au Module Décisionnel d'Imagerie. A partir de ces données, le(s) module(s) d'aide à la décision déduit (déduisent) des informations telles que des prédictions, des diagnostics ou des procédures de reprise. Ces informations sont alors transmises au M.D.I. qui sélectionne celles qui doivent être présentées à l'opérateur.

Le M.D.I. doit transmettre les informations considérées comme pertinentes à la console d'affichage graphique, utilisée par l'opérateur. Ainsi, la console doit être "esclave" du M.D.I., et doit être capable de modifier dynamiquement son affichage en fonction des décisions prises par le Module Décisionnel d'Imagerie. De plus, l'opérateur étant susceptible de demander des informations complémentaires ou des données au module d'aide à la décision, il est nécessaire de prévoir une commande de l'opérateur vers le Module Décisionnel d'Imagerie. Le M.D.I. est donc lié au(x) module(s) d'aide à la décision par les données et les informations qu'il doit gérer, mais il est aussi un composant indépendant de par la nature particulière des règles contenues dans ses bases de connaissances.

Ainsi, deux intégrations sont possibles pour l'implantation physique du module : la première possibilité est de l'intégrer sous la forme d'une tâche particulière au sein du système d'assistance complet et la deuxième consiste à utiliser un calculateur spécifique chargé des aspects graphiques, intégrant le M.D.I. s'occupant de la gestion de l'imagerie (TENDJAOUI, 1988 ; TABORIN, 1989).

Les objectifs du M.D.I. sont décrits précisément dans la partie suivante.

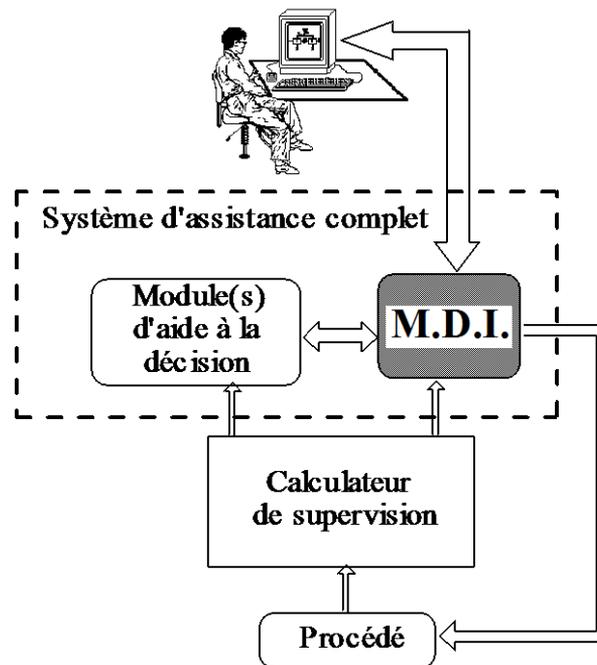


Figure III.6 : Localisation du M.D.I.

II.2. LES OBJECTIFS DU MODULE DECISIONNEL D'IMAGERIE

Le M.D.I. doit permettre des échanges d'information entre l'opérateur (ou un groupe d'opérateurs), un procédé et un ou plusieurs module(s) d'aide à la décision. Ainsi, les objectifs du M.D.I. sont (KOLSKI et al., 1990) :

- de **sélectionner** les informations à afficher sur l'écran compte tenu du contexte opérationnel du procédé et des besoins informationnels de l'opérateur, afin de permettre à l'opérateur d'effectuer la surveillance du procédé et de définir les actions correctrices éventuelles suite à l'apparition d'un dysfonctionnement ;
- **d'intégrer** à cette imagerie de supervision les conseils de correction fournis par le(s) module(s) d'aide à la décision lors des diagnostics et des reprises de défauts, en permettant aux modules de justifier leur raisonnement, afin de prévenir des conflits entre ceux-ci et l'opérateur humain ;
- de **définir une présentation ergonomique** de ces informations pour en faciliter la perception par l'opérateur humain.

Afin de réaliser ces objectifs, le module doit filtrer, à l'aide de bases de connaissances centralisant les besoins informationnels de l'opérateur dans les différents contextes opérationnels du procédé, les informations issues des modules d'aide à la décision, ainsi que les données fournies par le calculateur de supervision.

L'intégration du Module Décisionnel d'Imagerie dans la chaîne de supervision d'un procédé industriel demande notamment de procéder à une analyse de la démarche cognitive de l'opérateur vis-à-vis des différentes tâches qu'il doit exécuter, de façon à non seulement présenter les informations dont il a besoin mais aussi à adapter les modes de représentation appropriés à ces informations, dans le but de favoriser une coopération entre le(s) module(s) d'aide à la décision et l'opérateur considéré ici comme le "décideur final". Ce point sera plus discuté dans le chapitre IV.

Connaissant les objectifs du M.D.I., la partie suivante présente les concepts de base que le M.D.I. doit considérer, afin d'atteindre ceux-ci.

III. LES CONCEPTS DU MODULE DECISIONNEL D'IMAGERIE

Le M.D.I. doit être autonome et capable de s'adapter aux besoins des opérateurs. Il doit donc intégrer des connaissances sur : (i) les différents contextes opératoires du système à superviser, (ii) les opérateurs qui vont utiliser l'interface pour superviser le système, (iii) les tâches cognitives et sensori-motrices des ces opérateurs.

Ces connaissances doivent permettre de répondre aux trois problèmes ergonomiques suivants, figure III.7 :

- (i) **QUOI** présenter à l'opérateur, c'est à dire : quelles sont les informations pertinentes et utiles à afficher compte tenu des limites de perception humaines et aussi de la taille réduite des écrans. Nous considérons que le "QUOI" intègre le "Pourquoi", par l'utilisation de niveaux de justification de l'information présentée.
- (ii) **QUAND** lui présenter les informations sélectionnées, notamment en tenant compte du contexte opérationnel du procédé.

(iii) **COMMENT** les lui présenter, c'est à dire : quels sont les modes de représentation les mieux adaptés du point de vue ergonomique.

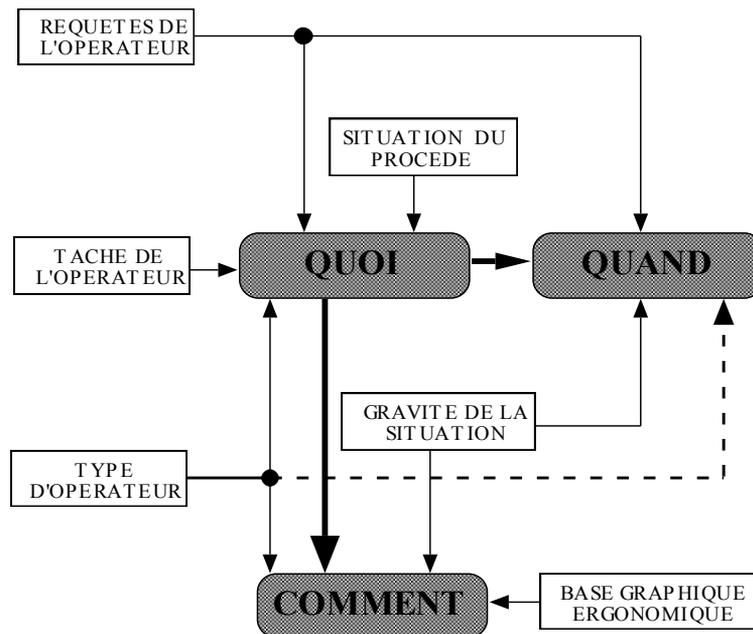


Figure III.7 : Les Concepts "QUOI / QUAND / COMMENT" du M.D.I.
(KOLSKI et al., 1990 ; TENDJAOUI et al., 1991B)

Ces problèmes ergonomiques sont discutés successivement ci-après.

III.1. LE PROBLEME DU "QUOI"

Le "quoi présenter à l'opérateur" représente l'ensemble des informations dont l'opérateur humain doit disposer, sur son écran, afin de mener à bien ses tâches de contrôle et de supervision. Il constitue ainsi l'ensemble des besoins informationnels de l'opérateur, pour une situation de fonctionnement donnée, lors de l'exécution d'une tâche particulière, pour une fonction et un opérateur donnés et en réponse à une requête particulière.

Ainsi, la réponse à la question du "**QUOI**" doit prendre en compte un ensemble de paramètres de fonctionnement du système homme-machine, figure III.7. Nous pouvons citer particulièrement :

- Les requêtes de l'opérateur : Elles correspondent à l'ensemble des informations dont l'opérateur peut avoir besoin à un moment donné de l'exécution de sa tâche. Ces informations sont demandées explicitement par

l'opérateur. Cependant, leur forme peut être différente selon le contexte considéré. Si l'opérateur, à un moment de l'exécution de sa tâche de conduite, réclame un type particulier d'information, tel que l'état d'une variable ou des justifications de conseils d'actions, alors le M.D.I. doit répondre à cette requête.

- Le type d'opérateur : Ce paramètre traduit les différences qui peuvent exister entre les utilisateurs du système. Ces différences peuvent se ressentir au niveau des fonctions attribués aux opérateurs humains (technicien de maintenance, rondier, chef quart, opérateur de la salle de contrôle), de leurs expériences respectives dans cette fonction, ou de toute autre considération professionnelle. Selon les différences d'expérience des opérateurs, si un défaut est prévu par le module d'aide à la décision, qui propose alors un plan d'actions pour le corriger, l'opérateur peut être d'accord avec ce plan et n'aura qu'à l'exécuter. Cependant, il peut ne pas être d'accord, et dans ce cas, il peut demander des justifications sur le plan proposé. Suivant le niveau d'expérience des opérateurs, le niveau de détail des justifications varie alors. Ainsi, un opérateur débutant aura tendance à demander plus de détails qu'un opérateur expérimenté. Le M.D.I devra en tenir compte.
- La tâche de l'opérateur par rapport aux différentes situations de conduite du procédé : Ce paramètre traduit la nécessité de savoir ce que l'opérateur humain est en train d'effectuer, connaissant la situation dans laquelle fonctionne le procédé. En effet, pour répondre aux interrogations de l'opérateur lors de l'exécution d'une de ses tâches, le M.D.I. doit connaître à quelle niveau de l'exécution de sa tâche l'opérateur se trouve, et ceci afin de sélectionner et de lui préparer les informations les plus utiles pour lui à ce moment. Dans ce but, le M.D.I. doit se baser sur la connaissance de la situation de fonctionnement du procédé. A titre d'exemple, la figure III.8 présente les différents types d'interventions du M.D.I., ainsi que les types de messages et de vues demandables par l'opérateur en fonction des états possibles de fonctionnement du procédé. Par exemple, en situation de démarrage, l'opérateur a besoin de conseils d'aide au démarrage, alors que, suite à une correction, il a besoin d'informations de suivi de certaines variables que le M.D.I. peut alors lui sélectionner automatiquement. De même, quand la situation est anormale, des alarmes doivent être automatiquement sélectionnées et hiérarchiquement affichées, etc.

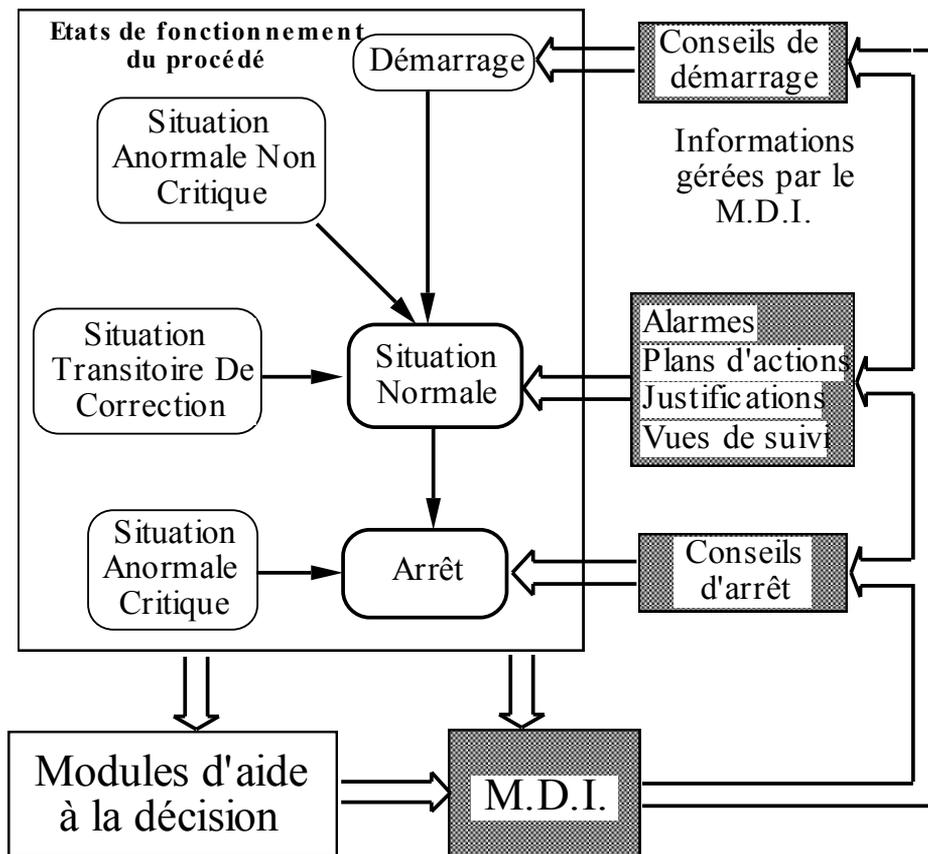


Figure III.8 : Exemples d'aides à apporter à l'opérateur en fonction des différentes situations de conduite du système

III.2. LE PROBLEME DU "QUAND"

Le "quand présenter l'information à l'opérateur" concerne le choix de l'instant où l'information à présenter à l'opérateur est la plus utile. Cette question (**QUAND**) est importante, car elle peut contribuer à gérer la charge de travail de l'opérateur, en modulant les flux d'informations provenant des modules d'aides à la décision, en fonction de la gravité de la situation du système homme-machine, figure III.9. Selon l'information à afficher et les requêtes de l'opérateur, le M.D.I. doit évaluer la gravité de la situation afin de déterminer si l'information doit être affichée instantanément ou ultérieurement.

Une telle démarche doit également permettre à l'opérateur, dans certains cas, d'entretenir ses connaissances et éventuellement de les enrichir. Par exemple, si suite à l'apparition d'un défaut, les plans de correction proposés par le module d'aide à la décision ne sont pas encore affichés, l'opérateur doit élaborer ses propres plans sans

attendre ceux du module d'aide à la décision. Dans ces conditions, l'opérateur entretient ses propres connaissances et évite en particulier le problème, "de la confiance aveugle", qui consiste pour lui à attendre et à appliquer automatiquement les plans proposés par le module d'aide sans les vérifier. Ceci peut le mener à perdre certaines de ses connaissances. Par conséquent, il ne sera plus capable dans certaines situations d'établir un plan de correction, si le module d'aide est défaillant.

En résumé, le problème du "**QUAND**" dépend de l'information à présenter (le "**QUOI**"), des requêtes de l'opérateur ainsi que de la gravité de la situation du système, figure III.7. L'évaluation de la gravité de la situation sera liée par exemple aux risques de violation des contraintes de sécurité et de production.

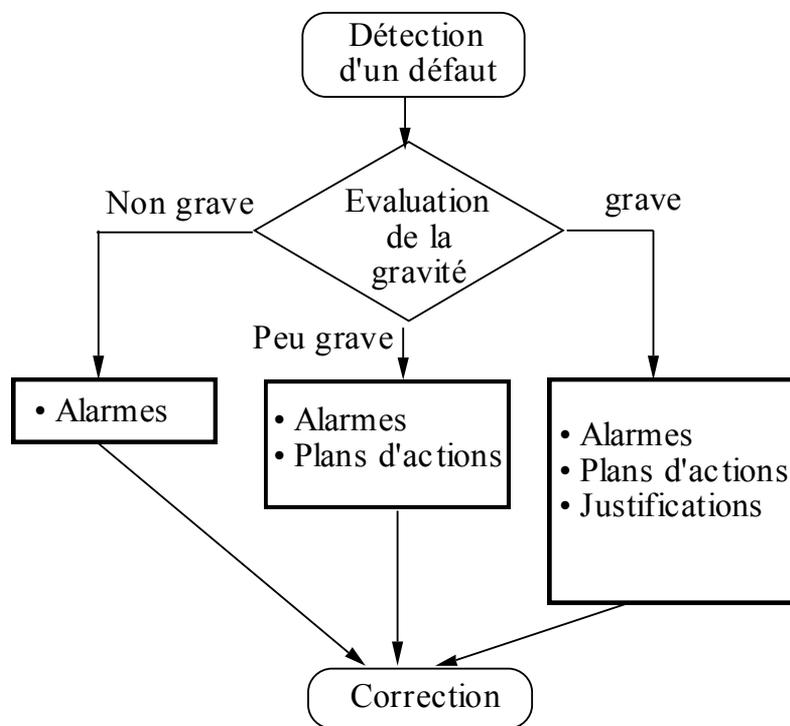


Figure. III.9 : Exemple de gestion automatique des messages d'aide en fonction de la gravité de la situation pour un opérateur novice

III.3. LE PROBLEME DU "COMMENT"

La réponse à la question "COMMENT présenter l'information" prend une grande importance. En effet, il importe de présenter l'information en fonction de formats ergonomiques pré-définis évalués a priori. Ceux-ci peuvent être centralisés dans une base graphique ergonomique, figure III.7. Si plusieurs modes de représentation sont utilisables avec la même efficacité dans un même contexte opérationnel, l'opérateur doit pouvoir **configurer** son interface, selon ses préférences et habitudes, avant la réalisation effective de ses tâches sur le procédé.

De plus, d'une représentation à une autre, l'information peut perdre son importance et sa pertinence. Ainsi, de nombreuses études ont montré l'efficacité du choix des modes de représentation (DE KEYSER, 1980 ; KOLSKI, 1989 ; TABORIN, 1989, etc.). A titre d'exemple, voici quelques définitions simples du "COMMENT" :

- Pour visualiser l'évolution d'une variable, une représentation sous forme de courbe est appropriée. En effet, une telle représentation permet d'informer l'opérateur sur la tendance de la variable. De plus, elle lui fournit un historique de celle-ci ;
- si la situation est grave, la couleur rouge pourra être utilisée pour attirer l'opérateur sur cette gravité ;
- les informations les plus importantes sont disposées en haut et à droite de l'écran ;
- si la situation de fonctionnement se dégrade et que l'opérateur n'arrive pas à être d'accord avec le plan de correction proposé par le module d'aide sans demander les justifications, celles-ci lui sont automatiquement proposées dans une fenêtre secondaire ;
- etc.

Il importe maintenant d'intégrer les connaissances de type "QUOI", "QUAND" et "COMMENT" dans une architecture de système d'assistance complet.

III.4. INTEGRATION DU "QUOI / QUAND / COMMENT" DANS UNE ARCHITECTURE DE SYSTEME D'ASSISTANCE

L'intégration du "QUOI", du "QUAND" et du "COMMENT" dans le système d'assistance complet, doit se conformer à des contraintes fonctionnelles particulières.

Après avoir présenté les objectifs et les concepts de base du M.D.I., il est maintenant possible de proposer une architecture fonctionnelle d'un système d'assistance intégrant le M.D.I., figure III.10.

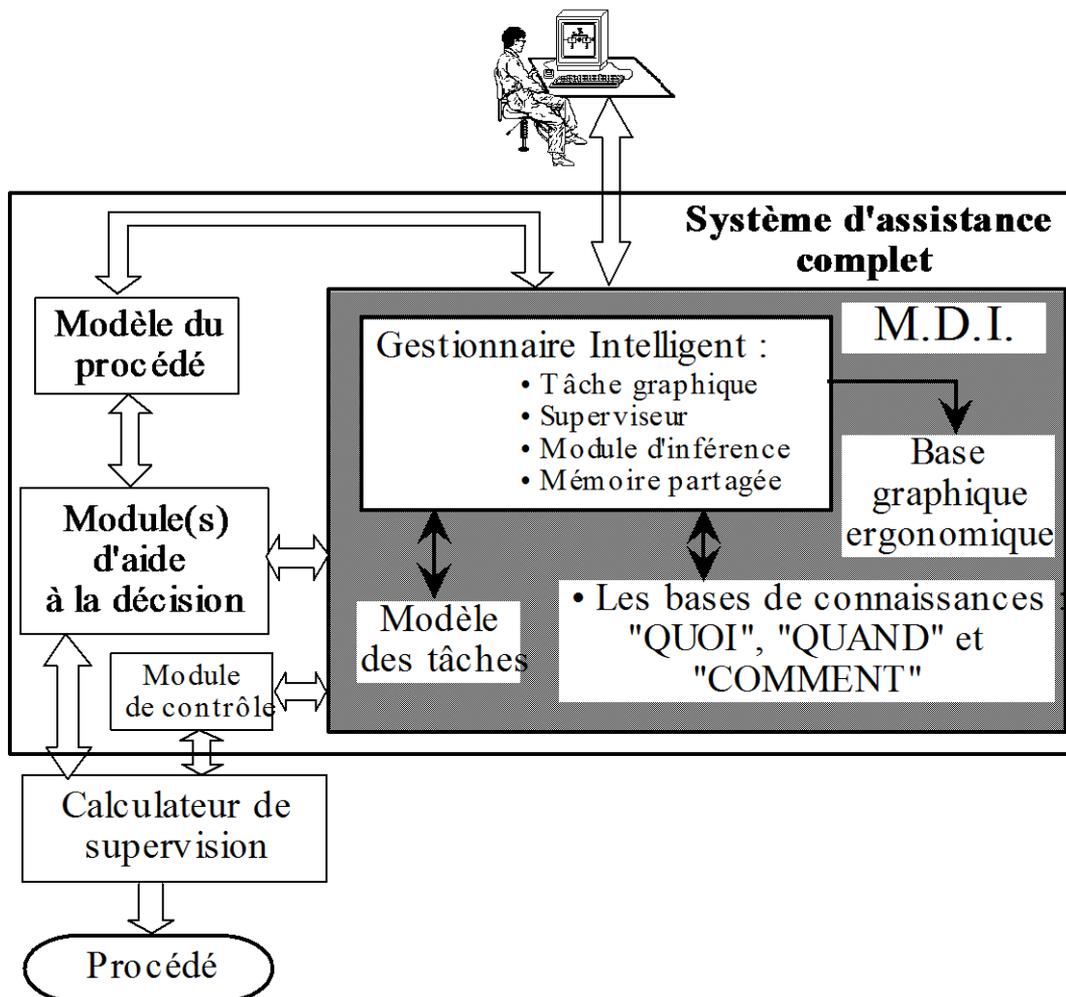


Figure III.10 : Architecture fonctionnelle du système d'assistance intégrant le M.D.I. (KOLSKI et al., 1992)

Au vu du chapitre II, il est bien évident que de nombreuses architectures peuvent être proposées. Celle que nous proposons a pour avantage de considérer **explicitement** les 3 concepts "QUOI", "QUAND" et "COMMENT" sous la forme de bases de connaissances. Le regroupement de ces 3 bases de connaissances peut être considéré comme correspondant à un modèle (simplifié) de l'opérateur dans la mesure où ces bases visent l'adaptation à ses besoins.

En s'appuyant sur un (ou plusieurs) modèle(s) du procédé, le(s) module(s) d'aide à la décision doit (doivent) pouvoir proposer de nouvelles informations (prédiction, diagnostic, conseils d'actions, justifications...). Cet ensemble d'information doit être disponible au niveau du M.D.I., grâce notamment à une zone mémoire commune, de type blackboard (NII, 1986). Le M.D.I. doit alors baser sa sélection de l'information à présenter à l'opérateur, dans un contexte particulier, à la fois sur un modèle des tâches que l'opérateur doit effectuer, et sur un modèle contenant des informations sur celui-ci, appelé "modèle de l'opérateur".

Le modèle des tâches de l'opérateur est composé de l'ensemble des tâches (assistées par le système d'assistance) recensées lors de l'analyse des tâches, alors que le modèle de l'opérateur intègre l'ensemble des 3 bases de connaissance requises pour assurer la gestion du "QUOI", du "QUAND" et du "COMMENT". Comme nous le décrirons dans le chapitre suivant, les décisions à prendre en matière de "QUOI", "QUAND" et "COMMENT" seront décrites à l'aide de règles de production structurées selon des arbres de décisions, suivant les principes désormais couramment utilisés dans les systèmes experts. A ce sujet, notons qu'il existe sur le marché un ensemble de générateurs de systèmes experts utilisant des principes de l'Apprentissage Symbolique Automatique³ (ASA) permettant, à partir d'un ensemble d'exemples, de générer automatiquement un arbre de décisions optimisé, tels EXPERT-EASE (Intelligent Terminals Ltd), RULE MASTER (Radian Corporation) ou 1stCLASS (Programs in Motion Inc). Notons que dans le cadre de cette recherche, nous avons utilisé l'outil 1stCLASS (chapitre IV, paragraphe IV.4).

En résumé, l'ensemble des connaissances requises par le M.D.I. portent notamment sur :

- les différents contextes opérationnels du procédé à superviser, incluant les modes de fonctionnements normaux et anormaux de l'installation ; ceux-ci

³ Pour de plus amples informations sur l'Apprentissage Symbolique Automatique, le lecteur se référera par exemple à BARR et al. (82), KODRATOFF (86) ou DUPAS (90).

sont modélisés à l'aide d'une fonction de gravité détaillée dans le chapitre IV, paragraphe II.4.

- les caractéristiques des différents opérateurs humains, leur niveau d'expérience, leur fonction, leurs préférences dans les modes et les styles d'interaction ;
- les tâches "assistées" des opérateurs ;
- les connaissances ergonomiques et les modes de représentation susceptibles d'être sélectionnés et affichés.

En plus de ces connaissances, le M.D.I. doit être capable de synthétiser les informations venant du procédé, afin notamment de pouvoir évaluer certaines connaissances utiles, telles que la situation de fonctionnement du système, ainsi que sa gravité. Chacun de ces points sera traité dans le dernier chapitre.

CONCLUSION

Ce chapitre a été consacré à la description du M.D.I., cœur d'une approche d'interface "intelligente" développée en laboratoire. Les travaux qui ont permis d'arriver au M.D.I. ont pour origine la réalisation d'une interface homme-machine entre opérateur(s) et module(s) d'aide à la décision implanté dans une salle de contrôle. Ces travaux ont permis de mettre en évidence et de développer un modèle implicite de coopération entre homme et système expert d'assistance.

L'avènement de nouvelles techniques en Intelligence Artificielle, dans le domaine de la conception d'interfaces graphiques et au niveau de l'analyse des tâches et des activités de l'opérateur, contribue actuellement à l'amélioration des interfaces homme-machine dans les salles de contrôle de procédés industriels. Dans ces conditions, notre approche d'interface "intelligente" peut bénéficier des progrès entrepris dans ces domaines afin de parvenir à des modèles explicites de coopération homme-machine. Cette approche conduit au concept de M.D.I., en visant à répondre dans chaque situation aux trois questions ergonomiques : QUOI présenter, QUAND et COMMENT présenter l'information.

Après avoir décrit les objectifs du M.D.I. et sa localisation dans le système de supervision, ce chapitre a proposé les différents concepts fondamentaux qui le définissent. Afin d'intégrer le M.D.I. dans le système d'assistance complet, une architecture fonctionnelle a ensuite été proposée intégrant trois modèles : un modèle du procédé, un modèle des tâches assistées de l'opérateur et un modèle de l'opérateur.

Dans le chapitre suivant, nous présentons une plate-forme expérimentale intégrant une interface "intelligente" de type M.D.I.

CHAPITRE IV

REALISATION D'UNE PLATE-FORME EXPERIMENTALE INTEGRANT UNE INTERFACE "INTELLIGENTE" DE TYPE M.D.I.

Chapitre IV réalisation d'une plate-forme expérimentale intégrant une interface "intelligente" de type M.D.I.

INTRODUCTION

Dans le chapitre précédent, nous avons énoncé les principes d'une interface "intelligente" appelée Module Décisionnel d'Imagerie (M.D.I.). Le présent chapitre est donc consacré à la mise en œuvre des concepts, présentés dans le chapitre précédent, pour la réalisation de cette interface "intelligente". Le cadre de cette mise en œuvre est un procédé simulé en laboratoire. Ce procédé consiste en un simulateur simplifié du circuit de refroidissement d'une centrale nucléaire. L'hypothèse de départ est que l'opérateur humain doit disposer d'un système d'assistance intégrant une interface de type M.D.I., capable :

- d'**anticiper** la détection d'alarmes grâce à son module de prédiction ;
- de **proposer** un plan d'actions pour rétablir la situation de fonctionnement du procédé ;
- de **justifier** l'état de la situation de fonctionnement du procédé et des différents plans d'actions qu'il a proposés.

Dans une première partie, ce chapitre présente globalement la plate-forme expérimentale réalisée.

Dans la deuxième partie, le système homme-machine dans lequel vient s'intégrer le M.D.I. est décrit. Ainsi, le procédé simulé est d'abord présenté. Ensuite, des hypothèses sont faites sur les limites, les ressources et les tâches prescrites de l'opérateur humain. Puis, les tâches humaines sont particulièrement analysées.

Dans la troisième partie, chacun des outils informatiques d'aide à l'opérateur, intégrés à la plate-forme, est décrit. Ces outils sont au nombre de quatre : un module de prédiction, un module de traitement des alarmes, un générateur de plans d'actions, et un générateur de justifications de plans d'actions.

Enfin le Module Décisionnel d'Imagerie est décrit dans la dernière partie, et particulièrement le principe de l'enchaînement entre les vues, la réalisation d'une bibliothèque graphique ergonomique et enfin celle d'un gestionnaire "intelligent" de l'imagerie.

I. PRESENTATION GLOBALE DE LA PLATE-FORME EXPERIMENTALE

La plate-forme expérimentale, développée au laboratoire sur ordinateur VAX sous VMS, est composée d'un ensemble de modules informatiques partageant une zone mémoire commune, figure IV.1. On distingue principalement :

- le simulateur du procédé ; celui-ci sera abordé dans la partie II de ce chapitre.
- Plusieurs fonctionnalités d'assistance à l'opérateur (décrites dans la partie III de ce chapitre) :
 - un module de prédiction ;
 - un module de traitement des alarmes ;
 - un générateur de plans d'actions ;
 - un générateur de justifications de plans d'actions.
- Le Module Décisionnel d'Imagerie (décrit dans la partie IV de ce chapitre), qui intègre :
 - un ensemble de bases de connaissances répondant aux trois problèmes ergonomiques "QUOI", "QUAND" et "COMMENT" ;
 - un moteur d'inférence chargé d'exploiter les règles contenues dans les bases de connaissances ;
 - un module graphique chargé de gérer et d'animer en permanence les différentes vues de l'interface à partir des conclusions du M.D.I. concernant le "QUOI", le "QUAND" et le "COMMENT" ;
 - une base de données sur les opérateurs. Dans l'état actuel de la plate-forme, cette base contient essentiellement des données relatives aux

expérimentations : nom du sujet, hypothèses de départ sur son expérience (novice, expérimenté ou expert), liste des scénarios de défauts utilisés, noms des fichiers gérés lors de l'expérimentation.

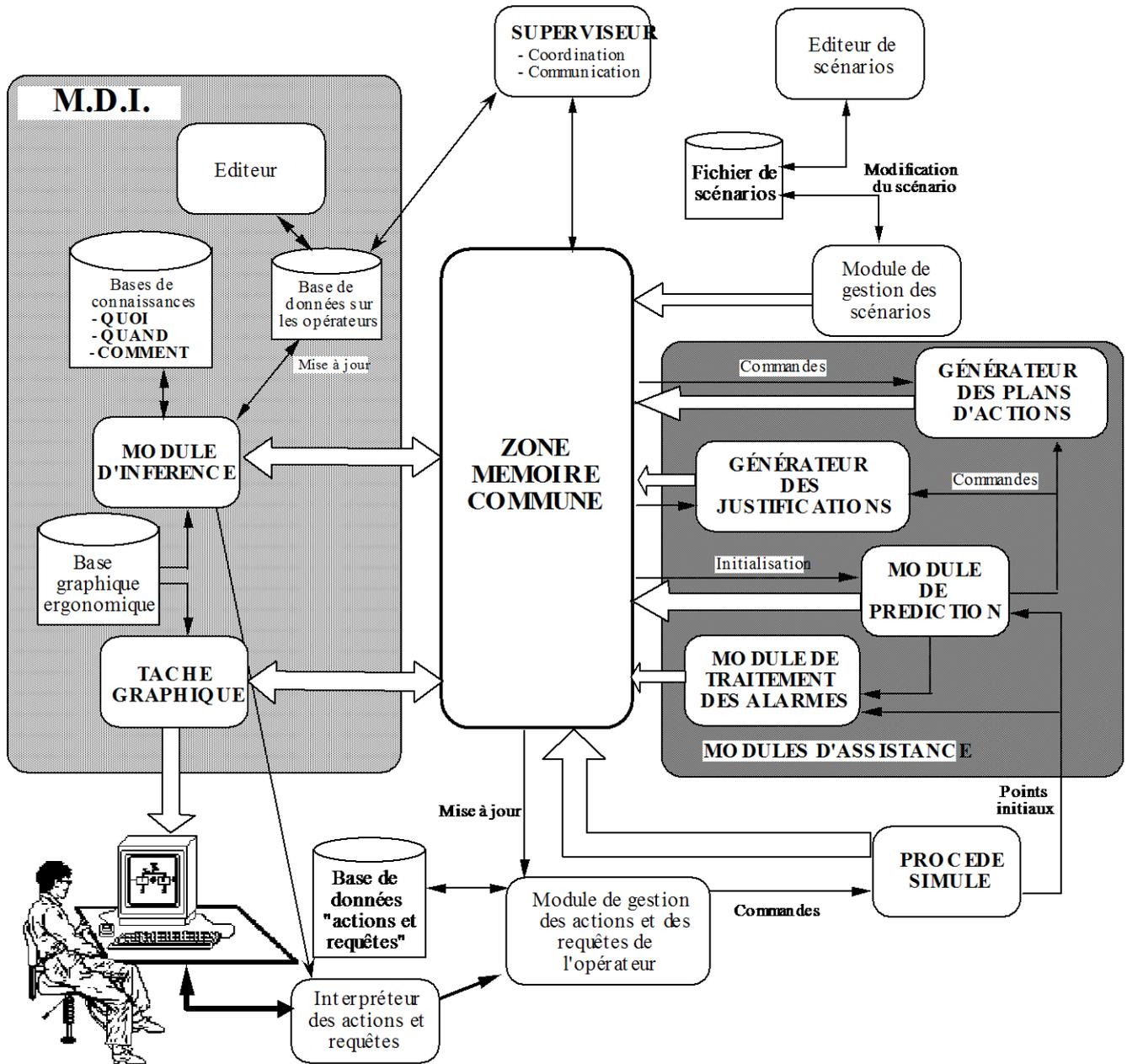


Figure IV.1 : Architecture de la plate-forme expérimentale (TENDJAOUI et al., 1991B)

- un superviseur gérant la coordination et les communications à travers la zone mémoire commune ; celui-ci ne sera pas décrit dans ce chapitre dans la mesure où il se contente d'assurer les échanges de données.
- un module capable de gérer des scénarios de pannes. Dans un but expérimental, chaque scénario peut être décrit à l'aide d'un éditeur spécialisé ; ce module sera décrit à travers un exemple dans le chapitre V.
- un module de gestion des actions et des requêtes de l'opérateur : ce module sauvegarde après interprétation celles-ci dans des fichiers. Pour les mêmes raisons que pour le superviseur, une description détaillée de ce module de gestion n'est pas indispensable.

Les modules principaux présents dans la figure IV.1 seront détaillés tout au long de ce chapitre.

Auparavant, il est important de décrire plus précisément le système homme-machine, tel qu'il est simulé dans la plate-forme expérimentale.

II. LE SYSTEME HOMME-MACHINE SIMULE

La plate-forme intègre un procédé simulé en laboratoire. Celui-ci simule le fonctionnement simplifié du circuit de refroidissement d'une centrale nucléaire (figure IV.2). Il est ainsi constitué de deux circuits principaux : un circuit primaire, appelé aussi circuit Sodium et un circuit secondaire, appelé aussi circuit Eau. Ces deux circuits communiquent entre eux grâce à un générateur de vapeur, considéré dans notre cas comme un échangeur de chaleur. De plus, parce que la production d'électricité est simulée par des fuites de vapeur d'eau à la sortie du générateur de vapeur, un circuit de réapprovisionnement en eau est considéré aussi.

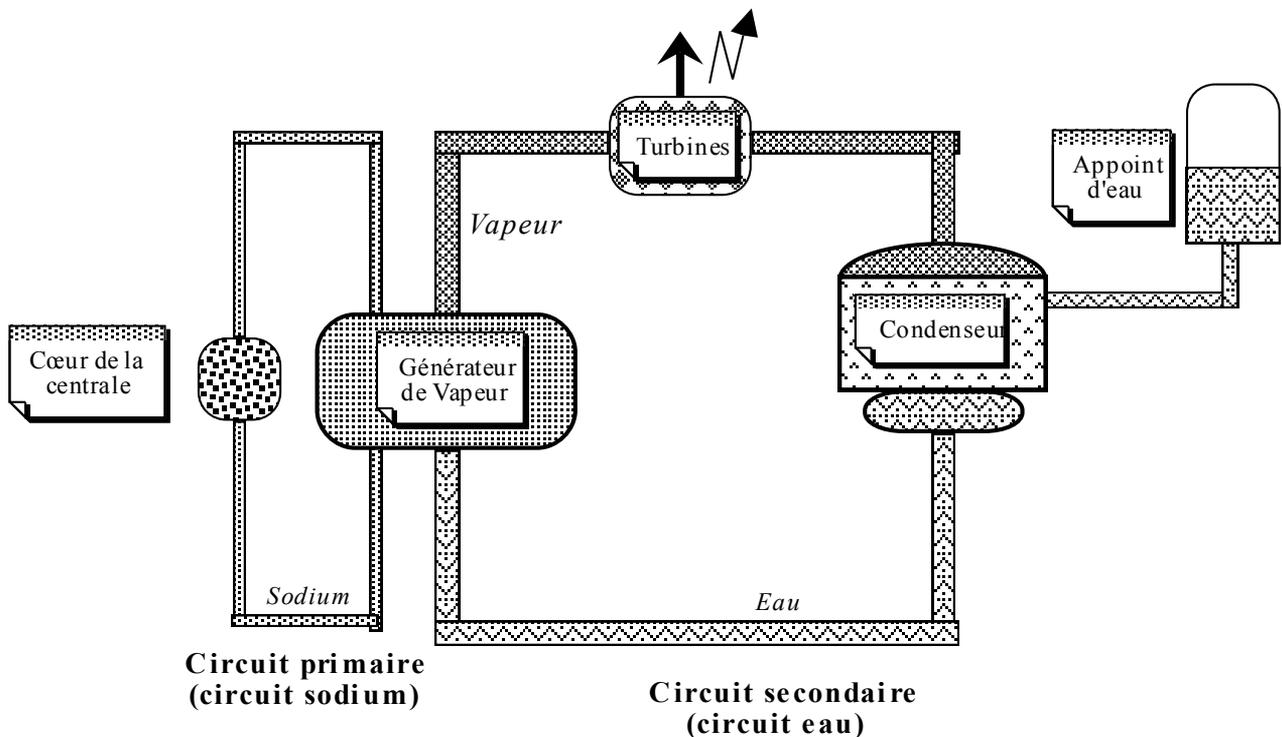


Figure IV.2 : Synoptique du procédé simulé

Ce procédé simulé est développé grâce au langage de programmation C et à certains principes de modélisation qualitative qui seront décrits plus précisément dans le paragraphe III.1. La simulation s'appuie sur un modèle de type graphe de fluence et se fait en quatre étapes :

- recenser l'ensemble des variables du procédé et construire un réseau de propagation entre elles,

- lier les variables les unes aux autres en suivant l'influence de chaque variable sur le réseau,
- affecter à chaque lien quatre paramètres (retard, temps de réponse, gain et sens de propagation),
- à chaque fois qu'une variable est perturbée, propager la perturbation dans le réseau en utilisant les quatre paramètres précédents.

Chacune des valeurs associées à une variable du procédé simulé est accessible par les autres modules par l'intermédiaire de la zone mémoire commune.

L'analyse, dans le paragraphe suivant, de ce procédé nous permettra progressivement de mettre en évidence des besoins en matière d'assistance à l'opérateur.

II.1. MODELISATION DU PROCEDE

A partir du procédé, il s'agit dans ce premier temps d'identifier et de dégager les contraintes techniques (dynamique, sécurité...) selon les différents contextes de fonctionnement envisagés.

Comme nous l'avons précisé, il simule le fonctionnement simplifié du circuit de refroidissement d'une centrale nucléaire. A ce sujet, l'ensemble des variables principales et des différentes relations qui existent entre celles-ci, est visible sur la figure IV.3.

Ce procédé étant simulé, il est bien entendu nécessaire de se baser sur un ensemble d'hypothèses concernant son fonctionnement. Ainsi, on suppose que les variables retenues pour le contrôle et la supervision de ce procédé sont au nombre de 16, et sont regroupées en 5 classes (voir le tableau IV.1) :

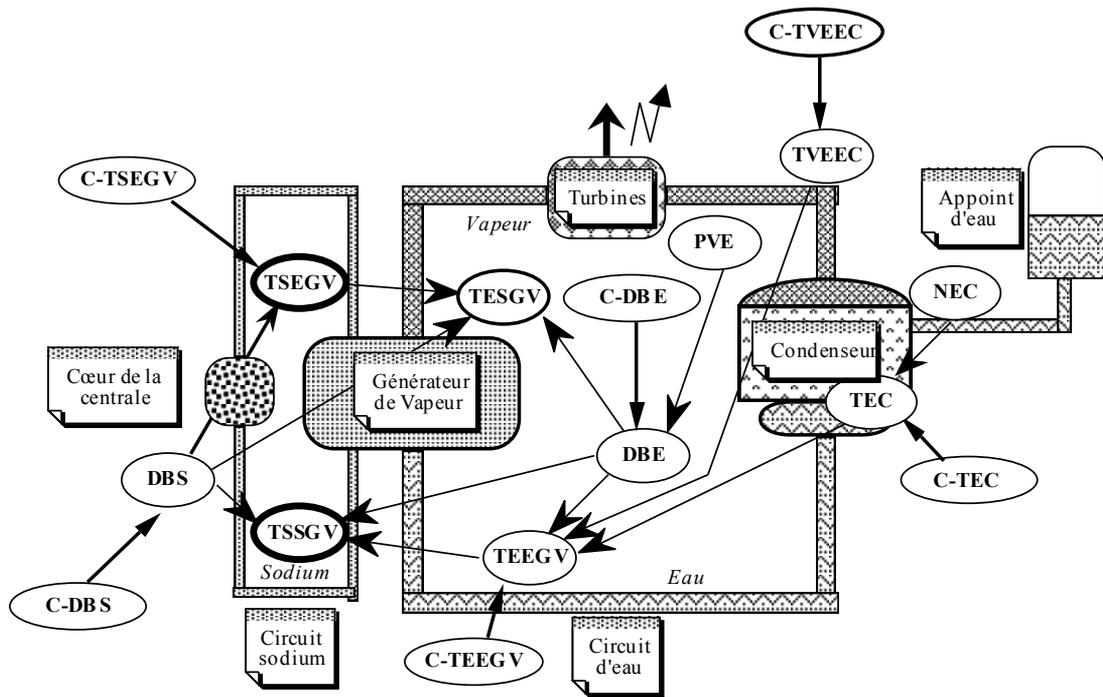


Figure IV.3 : Localisation et influence des variables dans le procédé simulé

Tableau IV.1 : Ensemble des variables du procédé simulé

Variable	Signification	Classe de la variable
TSEGV TSSGV	Température du Sodium à l'entrée du Générateur de vapeur Température du Sodium à la sortie du Générateur de vapeur	Sécurité
TESGV	Température de l'eau à la sortie du Générateur de vapeur	Production
C- TEEGV C- TSEGV C-DBS	Consigne de la température de l'eau à l'entrée du générateur de vapeur Consigne température du Sodium à l'entrée du Générateur de vapeur Consigne du débit du sodium	Commande
C-TEC C- TVEEC C-DBE NEC	Consigne de la température de l'eau dans le condenseur Consigne de la température de la vapeur d'eau à l'entrée du condenseur Consigne du débit d'eau Consigne du niveau d'eau dans le condenseur	Régulation automatique
PVE TEEGV TEC TVEEC DBE DBS	Pression de la vapeur d'eau Température de l'eau à l'entrée du générateur de vapeur Température de l'eau dans le condenseur Température de la vapeur d'eau à l'entrée du condenseur Débit d'eau Débit du Sodium	Conduite et Supervision

- la première classe regroupe deux variables traduisant un aspect de **sécurité** du système ;
- la seconde classe considère une variable traduisant un aspect de **production** ;
- la troisième classe regroupe trois variables de commande sur lesquelles l'opérateur peut agir pour contrôler le procédé ;
- la quatrième classe regroupe quatre variables sujettes à des perturbations et dont la régulation est assurée par des automatismes ;
- la dernière classe regroupe six variables internes de conduite, utilisable pour la supervision.

Pour simuler le procédé, nous avons construit un modèle qualitatif simple, figure IV.4. Celui-ci est composé de 16 nœuds (pour les 16 variables du procédé), reliés entre eux par des arcs. Si une perturbation arrive sur un des nœuds, elle sera propagée dans les arcs qui partent du nœud. A chaque arc est associé une fonction de transfert du premier ordre, représentée par :

- N° : numéro de l'arc (entre 1 et 20),
- Ret : Retard pur (en secondes),
- Trep : Temps de réponse (en secondes),
- Gain : le gain (Faible, Fort ou Moyen), qui est configurable. Dans notre cas, nous avons choisi la configuration suivante :

- Fort := 0,8
- Moyen := 0,5
- Faible := 0,2

A chaque nœud est associé le nom de la variable (par exemple TSEGV) et sa valeur nominale (par exemple 400 °C).

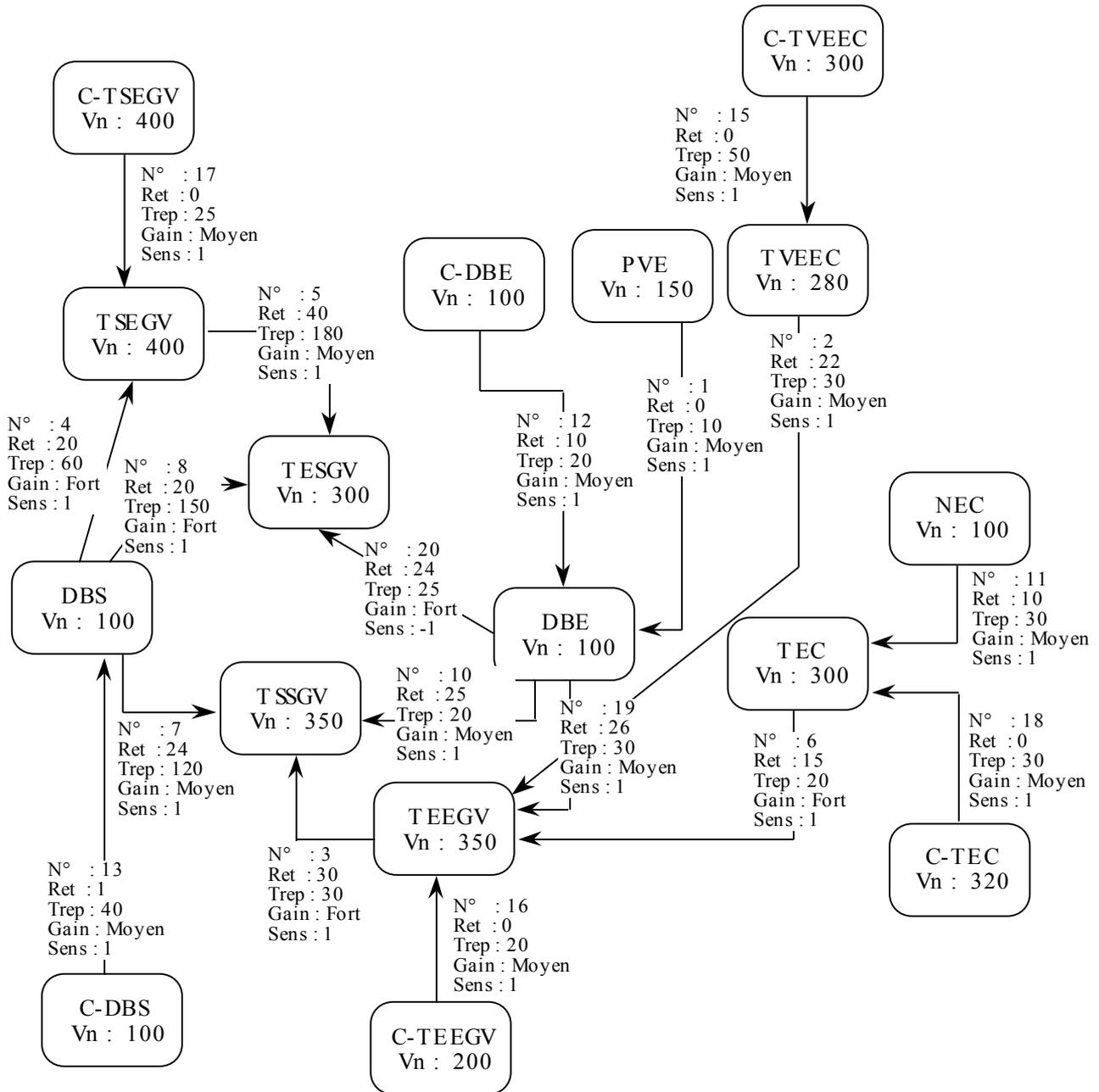


Figure IV.4 : Configuration du modèle qualitatif

D'autres hypothèses sont décrites dans le paragraphe suivant.

II.2. HYPOTHESES SUR LES LIMITES, LES RESSOURCES REELLES ET LES TACHES PRESCRITES DE L'OPERATEUR

Dans le cas du procédé considéré précédemment, on suppose que les tâches prescrites de l'opérateur consistent principalement à : (i) assurer la sécurité du système en évitant des dysfonctionnements du circuit primaire, aussi appelé circuit sodium, et (ii) assurer une production constante. Le fonctionnement normal peut donc en fait être caractérisé par la satisfaction de ces deux buts.

Pour le premier point, l'opérateur doit surveiller en permanence l'évolution des variables de la classe sécurité, afin de les garder dans une zone de sécurité pré-définie. Pour le second point, il doit surveiller l'évolution de la variable **TESGV**, traduisant la température de la vapeur d'eau à la sortie du générateur de vapeur. Cette variable simule la quantité d'électricité produite et donne ainsi une information sur la production du système.

Si l'opérateur assure ces deux points, le fonctionnement du procédé est qualifié de normal. En cas de dysfonctionnement, l'opérateur a pour tâche de compenser le défaut et de revenir à un point de fonctionnement normal. Dans les autres cas, l'opérateur a pour charge de surveiller et d'affiner les réglages des valeurs des variables du procédé, afin d'optimiser son fonctionnement.

Etant dans un contexte simulé, et ne disposant pas, pour l'instant, d'opérateur réel professionnel, nous avons fait principalement les hypothèses suivantes en ce qui concerne les limites et ressources réelles de l'opérateur :

- La seule fonction envisagée, pour l'instant, est celle de "contrôleur du procédé" (BAINBRIDGE, 1978). Nous avons donc écarté d'autres fonctions du type : "opérateur de maintenance", "ingénieur du procédé" ou encore "utilisateur occasionnel du système informatique".
- Nous supposons que l'opérateur fait de la résolution de problème pour compenser les défauts, et qu'il est habitué à utiliser un poste de travail comprenant un écran graphique, un clavier et une souris. Pour résoudre un problème, l'opérateur n'applique pas que des procédures pré-définies, il peut suivre chacun des trois comportements identifiés par RASMUSSEN (1983). A ce sujet, le but du M.D.I. sera de faire en sorte de l'orienter au maximum

vers un comportement basé sur l'habilité, en lui présentant les informations adéquates en fonction de la situation de fonctionnement du procédé (voir la figure III.9 dans le chapitre précédent).

- Nous supposons que les termes et les représentations graphiques utilisées pour l'interface sont usuels à l'opérateur. A cet effet, lors de la phase d'évaluation, des séances préliminaires d'apprentissage et d'entraînement auront à être effectuées.

Il s'agit maintenant de décrire plus précisément les tâches à effectuer, telles qu'elles sont susceptibles d'être assistées par le M.D.I.

II.3. LES TACHES DE L'OPERATEUR HUMAIN

Un modèle théorique des tâches assistées par le M.D.I. a été établi en laboratoire. Dans ce modèle, nous nous sommes limités aux tâches de l'opérateur dont l'identification est relativement aisée, figure IV.5.

Dans un premier temps, nous avons défini deux tâches facilement identifiables. La première tâche est celle qui accompagne la phase de démarrage du procédé appelée "tâche de démarrage". De la même façon, la deuxième tâche, appelée "tâche d'arrêt" est déclenchée par les procédures d'arrêt du procédé.

Si l'opérateur n'est pas en train de réaliser l'une des deux tâches précédentes et si aucune alarme n'est présente, nous supposons qu'il est en train d'accomplir une "tâche de surveillance et de compensation de défauts" afin d'optimiser le fonctionnement du procédé.

Enfin, nous avons regroupé l'ensemble des tâches décisionnelles de résolution de problème, définies dans le modèle de RASMUSSEN (1980 ; 1983), en une seule tâche principale appelée "tâche de résolution de problème". Celle-ci est identifiée grâce à l'apparition d'une alarme signalant l'apparition d'un défaut, qui suppose que l'opérateur va entrer dans une phase de résolution d'un problème. Cette alarme est générée grâce à la notion de gravité, décrite ci-après, capable de caractériser l'état de gravité de chaque mode de fonctionnement du procédé.

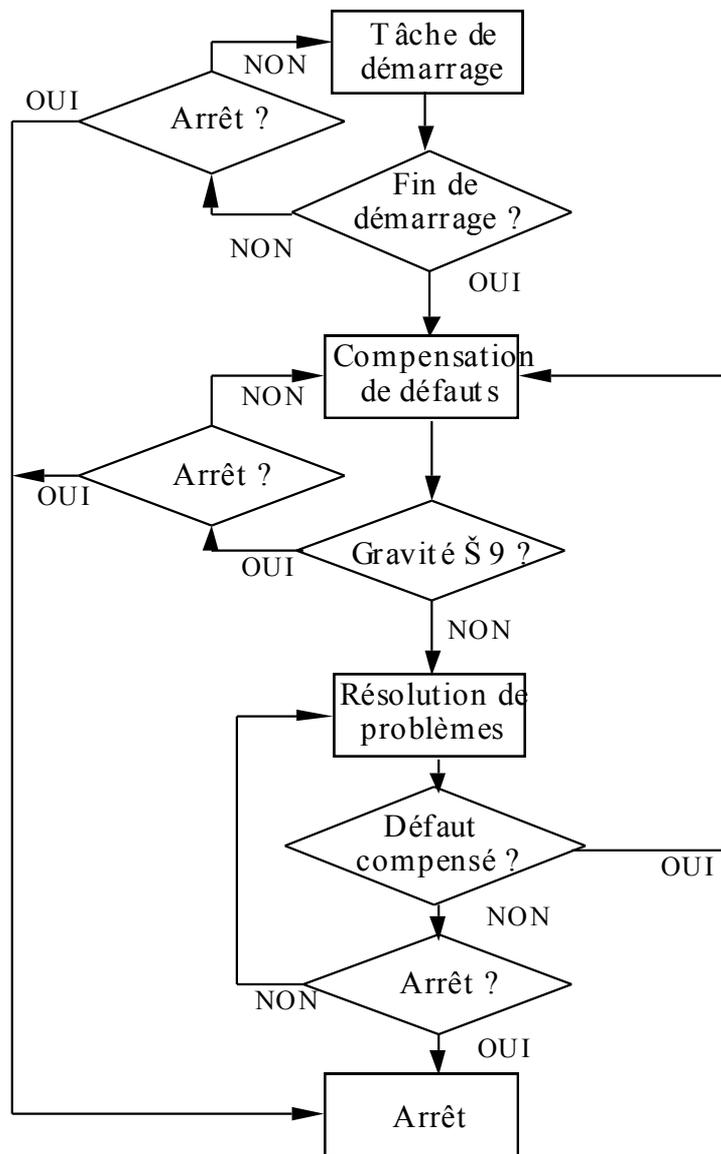


Figure IV.5 : Modèle des tâches assistées par le M.D.I.

II.4. LA NOTION DE GRAVITE

La notion de gravité de la situation est définie grâce à trois paramètres traduisant l'état de fonctionnement du système, tableau IV.2. Ces paramètres sont situés dans un ensemble de valeurs spécifiques. Chaque valeur est appelée "degré". Les différents paramètres sont définis comme suit :

- le degré d'insécurité du système traduit une échelle de gravité du système, prenant en compte de manière **explicite** les consignes de sécurité. Elle peut prendre trois valeurs : 0, 5 ou 10. La valeur 0 indique que la sécurité est

assurée, alors que la valeur 10 indique que la sécurité du système est menacée ;

- le degré d'anormalité de la production d'énergie traduit une échelle du taux de production en énergie de la centrale nucléaire. Ce paramètre prend en compte de manière **explicite** les consignes de production du système. Ce paramètre peut prendre l'une des trois valeurs : 0, 1 ou 2. La valeur 0 correspond à une production normale alors que la valeur 2 indique que la production ne respecte pas les consignes pré-établies ;
- le degré d'urgence exprime le délai imparti pour effectuer une compensation d'un défaut éventuel reste possible. Le degré d'urgence est quantifié par les valeurs : 0, 1 ou 2. La valeur 0 indique que l'opérateur dispose de "beaucoup de temps" pour agir sur le procédé et corriger la situation. Bien entendu, la notion de temps est en rapport direct avec la dynamique du procédé. En effet, une heure dans la vie d'un procédé lent n'est bien évidemment pas perçue de la même manière que dans un procédé rapide.

Tableau IV.2 : les valeurs associées aux notions de sécurité, production et d'urgence

Valeur	degré d'insécurité	Degré d'anormalité de la production	Degré d'urgence
0	Normale	Optimale	Long terme
1	-	Sous-optimale	Moyen terme -
2	-	Critique	Court terme -
5	Anormale	-	-
10	Critique	-	-

Grâce à cette classification des différents paramètres traduisant l'état de fonctionnement du système, nous pouvons facilement identifier 15 classes d'alarmes possibles. Chaque classe d'alarme est associée à un degré de gravité de la situation. Les niveaux de gravité sont classés dans une échelle allant de 0 à 14 (figure IV.6), et peuvent être calculés grâce à la fonction suivante :

$$\text{Degré (gravité)} := \text{Degré (insécurité)} + \text{Degré (production)} + \text{Degré (urgence)}$$

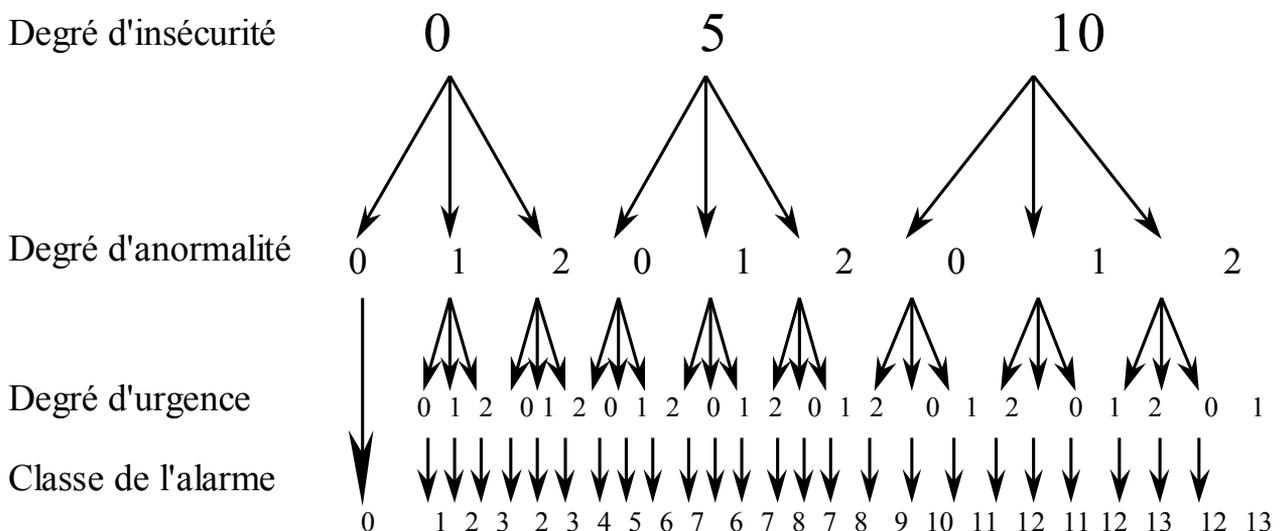


Figure IV.6 : Hiérarchisation des alarmes (TENDJAOUI et al., 1991C)

La classification précédente permet, dans un second temps, d'identifier un ensemble de situations de fonctionnement du procédé. Dans notre cas, les situations de fonctionnement identifiées ont été choisies au nombre de 5 : optimisée, normale, anormale, critique et irrécupérable (voir le tableau IV.3).

Nous tenons à signaler que cette formulation du degré de gravité, comme une agrégation des trois critères insécurité, anormalité de production et urgence a été choisie volontairement très simple dans notre étude, car l'objectif est ici de tester la faisabilité des concepts du M.D.I.

Ultérieurement, il conviendra d'approfondir cette formulation en s'appuyant sur des théories éprouvées comme la théorie de la décision multicritère (GANDIBLEUX et al., 1992) ou la théorie des sous-ensembles flous (GUERRA, 1991). Il faudra alors identifier ces critères de telle façon qu'ils correspondent à l'évaluation qu'un opérateur de supervision en fait réellement.

Tableau IV.3 : Identification de la situation de fonctionnement en fonction de la gravité.

Gravité : Hiérarchisation des Alarmes	Situation de fonctionnement du procédé
0	Optimisée
1 à 3	Normale
4 à 9	Anormale
10 à 12	Critique
13 à 14	Irrécupérable

Il est maintenant possible de décrire plus précisément les différentes fonctionnalités d'aide à l'opérateur, que le M.D.I. doit gérer.

III. LES OUTILS D'AIDE A L'OPERATEUR

Dans un but expérimental, afin de permettre à l'opérateur de mener à bien les différentes tâches qui lui sont attribuées, nous avons étudié les recommandations ergonomiques dans le domaine du contrôle de procédés complexes pour faire ressortir plusieurs besoins en assistance (VITTET, 1981 ; DANIELLOU, 1986 ; RASMUSSEN, 1986 ; TABORIN, 1989). Ainsi, on suppose que l'opérateur doit pouvoir disposer d'un ensemble de modules d'aide capables de lui fournir :

- des alarmes hiérarchisées ;
- les moyens de visualiser l'évolution future des variables ;
- des conseils d'actions en cas de dysfonctionnement ;
- des justifications de ces conseils d'action.

Ces aides sont fournies par plusieurs modules, écrits en C dans le cadre de cette recherche, décrits successivement ci-dessous. Il faut noter que les informations produites par ces différents modules ne sont pas affichées automatiquement mais leur affichage est contrôlé par le M.D.I.

III.1. LE MODULE DE PREDICTION

Puisque l'opérateur doit pouvoir disposer d'un moyen de visualiser l'évolution future des variables du procédé, nous avons repris le modèle de simulation du procédé, qui se base sur plusieurs travaux dans le domaine de la modélisation qualitative, pour l'utiliser comme modèle de prédiction. En effet, nous avons adopté les principes de la modélisation qualitative utilisées par CALOUD (1988) et FERRAY-BEAUMONT (1988). Le principe consiste à définir pour l'ensemble des 16 variables constituant le procédé un ensemble de paramètres caractéristiques de leur fonctionnement. Nous avons ainsi défini des relations qualitatives entre les différentes variables du procédé. Le résultat de cette opération est un réseau de propagation constitué de nœuds et d'arcs, où les nœuds correspondent aux différentes variables (16 en tout), et les arcs représentent l'effet d'une perturbation d'une variable sur une autre. Chaque arc possède 4 paramètres qui sont : (i) le gain, (ii) le temps de réponse, (iii) un retard pur, et (iv) le sens de de variation (+ ou -). Bien entendu, cette

modélisation s'inspire des modèles communément utilisés en Automatique. Son intérêt majeur consiste en la possibilité de propager informatiquement une perturbation d'une variable à une autre, les variables étant en relation par des arcs (voir le réseau en figure IV.4). De plus, et parce que certaines variables subissent les perturbations de plus d'une variable, nous avons appliqué les principes du théorème de superposition de petits mouvements. Ce principe considère, à un instant donné, le résultat de deux perturbations sur une variable comme la somme algébrique des deux.

Le module de prédiction est le premier à être activé. Il se déclenche à chaque fois qu'une variable régulée automatiquement ou contrôlée par l'opérateur subit une perturbation ou quand une des variables de commande change. Dans ce cas, le module de prédiction calcule les valeurs des variables pour les dix prochaines minutes et écrit ses résultats dans une zone mémoire commune, accessible aux autres modules.

III.2. LE MODULE DE TRAITEMENT DES ALARMES

Puisque l'opérateur doit pouvoir disposer d'alarmes hiérarchisées, le modèle de prédiction sera utilisé dans un premier temps pour détecter préalablement les dépassements de seuils des variables. Par la suite, et pour chaque variable, l'alarme sera classée grâce à la formule de hiérarchisation résultant de l'agrégation des gravités et présentée précédemment, figure IV.6. Ainsi, seules les alarmes dont la classe est la plus élevée seront susceptibles d'être présentées à l'opérateur.

Le module de traitement des alarmes est activé à chaque fois qu'une variable du procédé dépasse ou va dépasser le premier seuil d'alarme. Pour cela, il scrute en permanence les valeurs des variables pour les dix prochaines minutes.

III.3. LE GENERATEUR DE PLANS D'ACTIONS

Afin de permettre à l'opérateur de disposer de conseils d'actions en cas de dysfonctionnement, un certain nombre de plans d'actions théoriques ont été établis. Ces plans d'actions sont tirés d'un mini-réseau qualitatif extrait du réseau global du procédé. Ce mini-réseau ne tient compte que des variables des classes sécurité, production et commande, figure IV.7. Sur ce mini-réseau, nous avons propagé des perturbations et calculé les différents paramètres qualitatifs de ce mini-réseau. Parce

que le nombre de variables à surveiller est restreint, nous avons identifié l'ensemble des défauts possibles. Pour chaque défaut, nous avons établi un modèle de plan d'action de type : "Pour corriger la variable **X** il faut agir sur la variable **Y** de **Z** dans le sens (-,+) avant **T** secondes". Quelques exemples de plans d'actions détaillés sont donnés en annexe B.

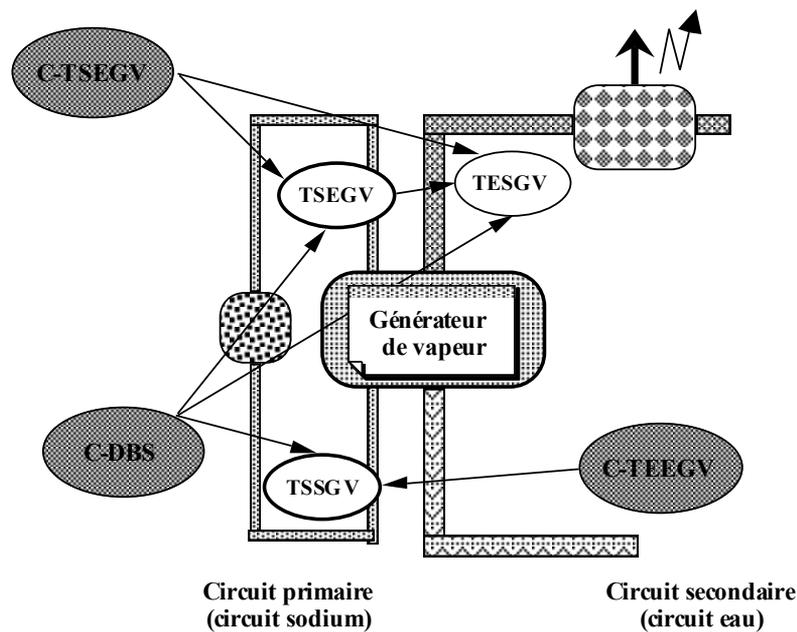


Figure IV.7 : Modèle du procédé permettant l'élaboration des plans d'actions

Le module de génération des plans d'actions est déclenché juste après que l'alarme soit classée. Il puise ses données dans la zone mémoire commune et propose des plans d'actions de correction pour chaque situation grâce au modèle défini précédemment.

III.4. LE GENERATEUR DE JUSTIFICATIONS

De la même façon que précédemment, nous avons modélisé les justifications de chaque défaut et son plan d'action par deux réseaux de variables. Le premier réseau, appelé "justification de surface", présente à l'opérateur la variable sur laquelle l'alarme s'est déclenchée ainsi que l'ensemble des causes possibles est relié à cette variable ; ceci permet de limiter le champ de recherche de la cause du défaut. Le deuxième réseau, appelé "justification profonde", représente la variable sur laquelle l'alarme s'est déclenchée et les causes les plus probables auxquelles elle est reliée

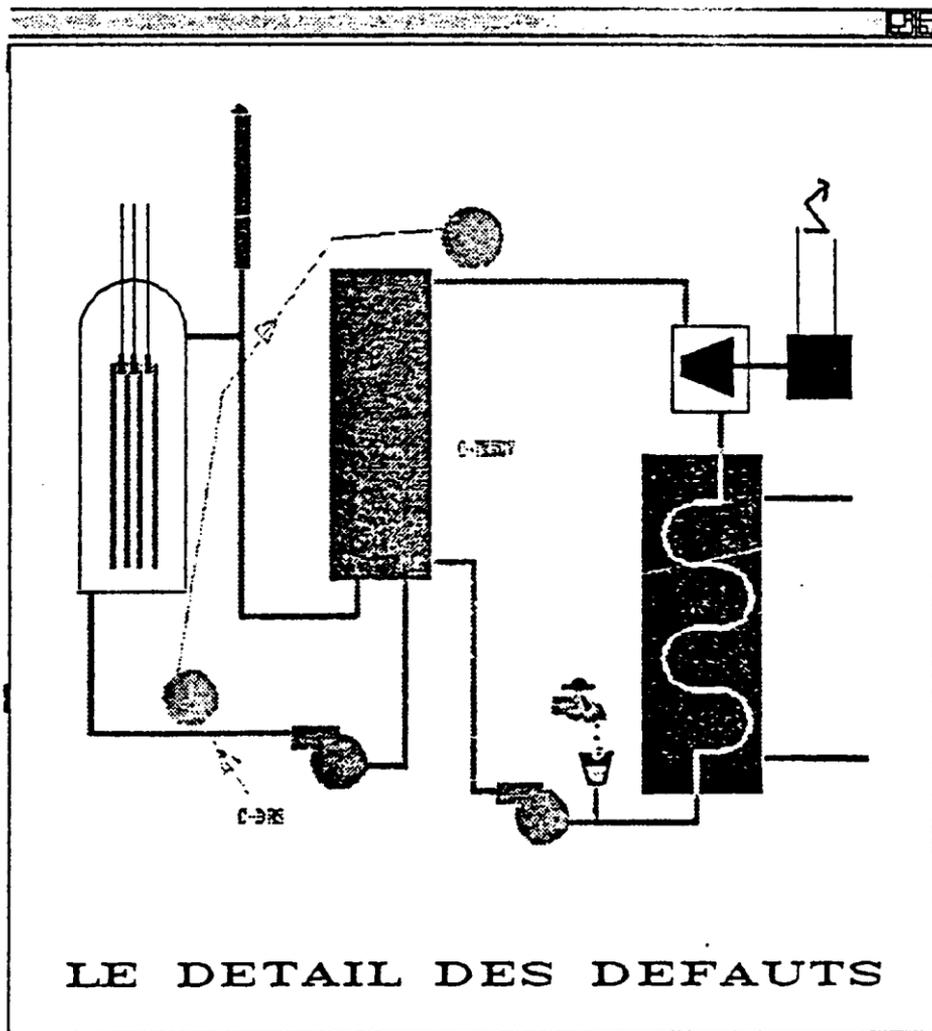


Figure IV.9 : Exemple de justification profonde

IV. LE MODULE DECISIONNEL D'IMAGERIE

Le Module Décisionnel d'Imagerie intègre principalement un ensemble de bases de connaissances répondant aux trois problèmes ergonomiques "QUOI", "QUAND" et "COMMENT", ainsi qu'un moteur d'inférence chargé d'exploiter les règles contenues dans les bases de connaissances.

Dans cette partie, nous expliquerons successivement les raisons du choix d'une structure d'arbres de décisions pour représenter les connaissances du M.D.I., puis les principes utilisés pour la composition et l'enchaînement des vues. Par la suite, nous décrirons la façon dont la bibliothèque graphique a été réalisée. Enfin, nous présenterons le gestionnaire "intelligent" d'imagerie à travers son fonctionnement et la méthode utilisée pour la construction de ses bases de connaissances.

IV.1. UTILISATION DES ARBRES DE DECISIONS POUR LA REPRESENTATION DES CONNAISSANCES

Le M.D.I. doit pouvoir composer et enchaîner "intelligemment" les vues en fonction du contexte du procédé, des tâches de l'opérateur, de son niveau d'expérience, de ses habitudes, de la situation de fonctionnement du procédé, etc. En résumé, il doit définir les **décisions** à prendre pour répondre aux questions de "QUOI" présenter et "QUAND" et "COMMENT" présenter cette ou ces information(s).

Pour cela, la technique des arbres de décisions a été utilisée pour construire les bases de connaissances du Module Décisionnel d'Imagerie. C'est une technique paraissant adaptée à la spécification de la composition et de l'enchaînement des vues graphiques, et qui a l'avantage d'être bien connue des concepteurs de systèmes IA.

Rappelons qu'un arbre de décisions est une représentation graphique de la succession des états de connaissance et de leurs opérateurs associés nécessaires à la résolution d'un problème donné ; cette structure est représentée par la figure IV.10.

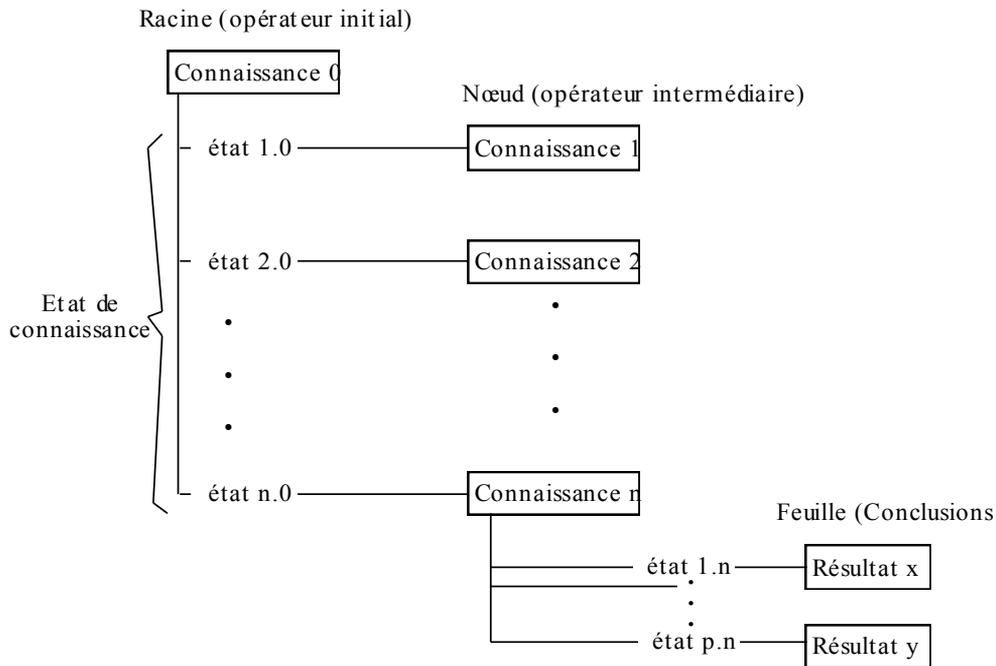


Figure IV.10 : Modèle de l'arbre de décisions

Chaque branche correspond à un état possible de la connaissance symbolisée par le nœud d'où elle part, et aboutit à un nœud représentant l'opérateur associé. L'opérateur initial est la racine de l'arbre de décisions et représente le point de départ du raisonnement, tandis que chaque nœud terminal (feuille) symbolise un résultat ou une conclusion. Dans un réseau d'arbres de décisions, tout nœud (initial, intermédiaire ou terminal) peut être lui-même un sous-arbre de décisions et non pas seulement un nœud simple : on parle alors de nœud générique. De plus, un tel réseau représente une synthèse entre les connaissances de l'expert et ses stratégies de résolution (NASSIET, 1987).

Un nœud générique intermédiaire symbolise un sous-but dont la démonstration est assurée par l'arbre de décisions associé, qui à son tour peut contenir des nœuds génériques intermédiaires. Un nœud générique terminal indique le chaînage vers un autre arbre de décisions. L'emploi des nœuds génériques permet la décomposition et la structuration de la connaissance en modules indépendants et plus facilement contrôlables qu'un arbre de décisions complet et unique.

A l'aide des arbres de décisions, il est possible d'établir une description structurelle et fonctionnelle de l'imagerie, en précisant particulièrement la composition des vues et leurs enchaînements. Comme nous l'avons souligné dans le

chapitre précédent, paragraphe III.4, il existe des outils basés sur des techniques d'Apprentissage symbolique Automatique qui sont capables de faciliter la création d'arbres de décisions, et ceci à partir d'une collection représentative d'exemples de décisions à prendre. Lors de cette recherche, nous avons utilisé l'outil 1stClass. Son utilisation sera plus précisée dans ce chapitre, paragraphe IV.4.

Avant de décrire, dans la fin de cette partie les bases de connaissances et le moteur d'inférence capable de les exploiter, il s'agit dans un premier temps d'expliquer les principes utilisés pour la composition et l'enchaînement des vues.

IV.2. PRINCIPES UTILISES POUR LA COMPOSITION ET L'ENCHAINEMENT DES VUES

Dans un premier temps, nous avons recensé le nombre de fenêtres que doit gérer l'interface et le contenu de chacune d'elles, figure IV.11. Ainsi, en nous inspirant des vues développées dans le cadre du projet ALLIANCE (Cf chapitre III), il a été convenu que l'opérateur doit pouvoir disposer :

- d'une fenêtre principale destinée à contenir les vues graphiques utilisant des modes de représentation graphiques de type "étoile", "courbes", "graphes de fluence", etc (voir le chapitre III) ; nous avons choisi la fenêtre C ;

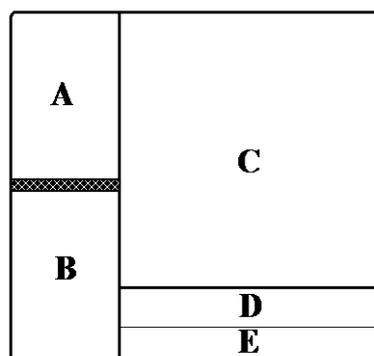


Figure IV.11 : La décomposition de l'écran en cinq zones

- d'une fenêtre destinée à contenir les informations importantes telles que les alarmes et les conseils de correction ; nous avons choisi la fenêtre A ;
- d'une fenêtre destinée à faciliter le contrôle du procédé ; nous avons choisi la fenêtre D ;

- d'une fenêtre destinée à contenir les commandes d'appel de vues (requêtes de l'opérateur) ; nous avons choisi la fenêtre E ;
- d'une fenêtre destinée à recevoir des affichages secondaires tels que les vues graphiques des justifications quand le M.D.I. les propose sans qu'elles soient demandées par l'opérateur ; nous avons choisi la fenêtre B.

L'enchaînement entre les vues peut être vu de deux manières : (i) la première peut être qualifiée "**d'absolue**" quand l'opérateur fait une requête pour l'affichage d'une information et (ii) la seconde de "**relative**" pour les informations dont l'affichage est contrôlé par le M.D.I.

Dans le premier cas où l'enchaînement est **absolu**, le M.D.I. traite la requête de l'opérateur, figure IV.12. Si la requête est une variable de commande, une fenêtre contenant un panneau de commande est alors affichée. Celle-ci ne disparaîtra que si l'opérateur confirme ou annule sa commande. Si la requête est une demande de vue courbe, une fenêtre présentant une liste hiérarchisée de variables est alors affichée. L'opérateur humain doit pouvoir à tout moment accéder à l'ensemble des vues. Cependant, si l'information qu'il souhaite consulter à un moment donné n'est pas disponible, il importe que les commandes qui doivent lui permettre d'y accéder le laissent supposer par leur représentation sur l'écran. En effet, l'opérateur doit savoir à tout moment si telle ou telle information est disponible, sans avoir à faire une requête. De plus, certaines informations, telles que les justifications de plans d'actions, sont très importantes dans certaines situations. Alors le M.D.I. doit être capable de les lui proposer en affichant la vue correspondante dans l'endroit réservé à la vue secondaire (zone B).

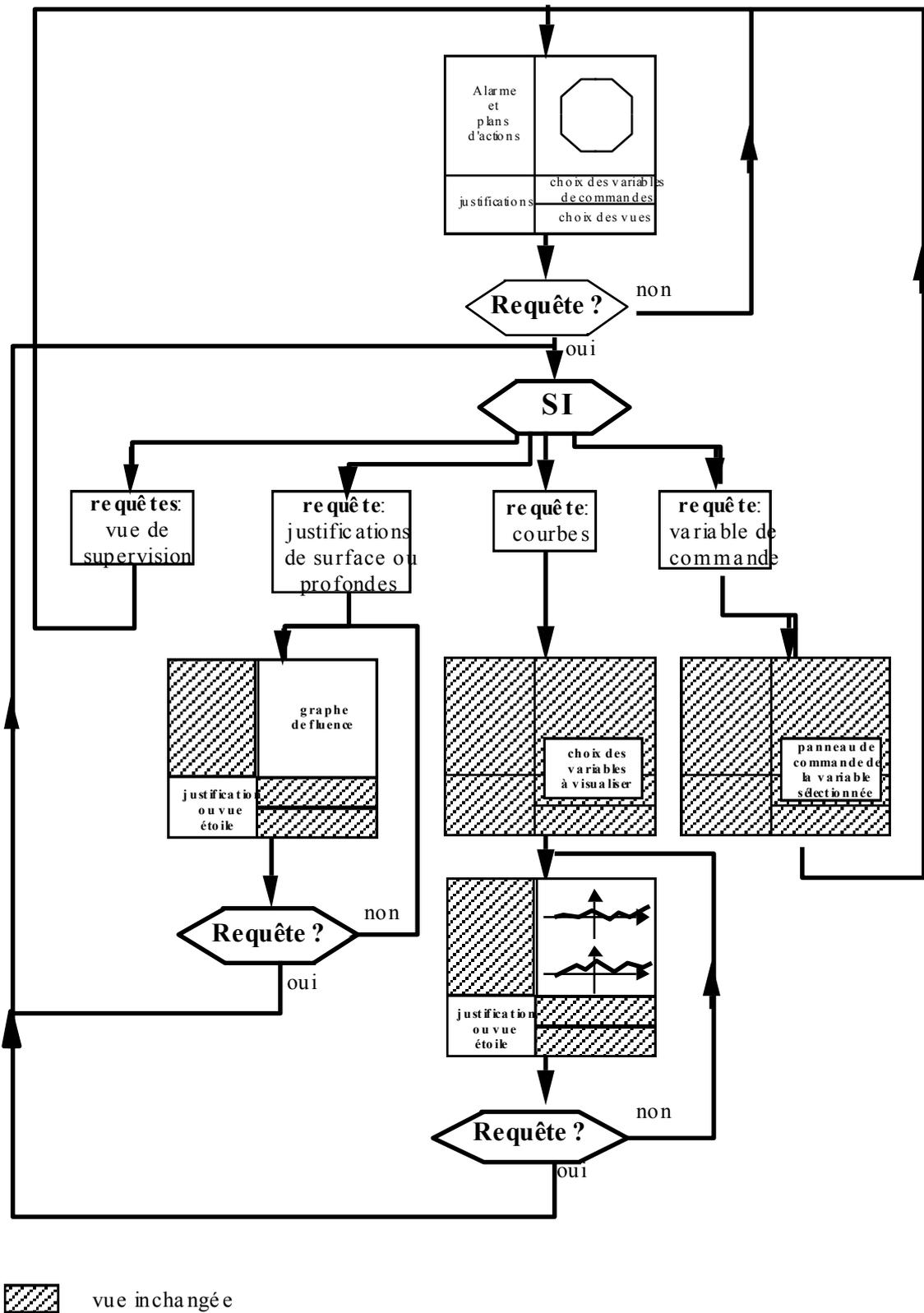


Figure IV.12 : Le traitement des requêtes de l'opérateur et les compositions de l'écran correspondantes

Dans le second cas d'enchaînement "**relatif**", l'affichage est fonction des connaissances centralisées dans les bases "QUOI", "QUAND" et "COMMENT". En effet, le M.D.I. évalue la situation et décide si une vue doit être affichée dans la fenêtre principale (fenêtre C), si l'opérateur la demande ou dans la fenêtre secondaire (fenêtre B) si le M.D.I. le décide. En effet, si la situation se dégrade, des justifications peuvent être affichées dans la fenêtre secondaire permettant à l'opérateur de vérifier le plus rapidement possible les conseils d'actions qui auront été déjà proposés. Les détails du gestionnaire, incluant le M.D.I., et qui pilote en "**relatif**" des vues sont présentés dans le paragraphe IV.4.

L'objectif final des enchaînements **absolu** et **relatif** est d'obtenir une coopération homme-machine permettant d'arriver le plus rapidement possible à un consensus entre l'opérateur et le système d'aide lors de la prise de décision et suite à l'apparition d'un défaut.

Pour l'ensemble des aspects graphiques gérés ou non par le M.D.I., il a été nécessaire de réaliser une bibliothèque graphique. Celle-ci est supposée a priori répondre à des critères ergonomiques de présentation d'information sur écran.

IV.3. REALISATION D'UNE BIBLIOTHEQUE GRAPHIQUE ERGONOMIQUE

Dans un premier temps, nous avons été amenés à choisir un outil graphique nous offrant la possibilité de créer facilement les différentes composantes graphiques telles que la vue étoile, les vues courbes, les graphes de fluence, les messages textuels. Nous avons choisi pour cela l'outil de DATAVIEWS (V.I. CORPORATION, 1988), qui tourne sur un grand éventail de stations de travail et en particulier sur la station de travail VS/3100. DATAVIEWS propose un ensemble de deux produits : DVDRAW et DVTOOLS.

DVDRAW est un éditeur graphique qui apporte des potentialités destinées à faciliter la réalisation d'interfaces graphiques, comme : l'appel d'une vue à partir d'une autre, l'utilisation de différents plans de dessin, l'utilisation de représentations graphiques pré-définies, appelées "objets", pour visualiser des variables ou du texte. DVDRAW permet aussi la création, la destruction ou le rafraîchissement, en "dynamique", d'objets graphiques (vues, fenêtres, objets géométriques, textes).

DVTOOLS est une bibliothèque de fonctions graphiques destinées à faciliter la gestion des différentes vues créées avec l'éditeur DVDRAW. Ces fonctions sont du type "afficher", "effacer", "déplacer", "changer" les attributs d'un objet ou d'un ensemble d'objets.

Enfin, DATAVIEWS a été choisi pour son ergonomie d'utilisation et la possibilité qu'il offre pour l'intégration de procédures graphiques dans un programme écrit en C ou en PASCAL.

Pour la spécification des différents modes de représentation et des attributs graphiques, nous nous sommes basés sur notre expérience dans le domaine de la conception d'interface, en harmonie avec la littérature actuelle dans ce domaine. Ces modes doivent répondre directement à la question "COMMENT présenter l'information". Dans notre cas, nous avons considéré quatre types de vues graphiques : (i) de supervision, (ii) de justification peu détaillée, (iii) de justification très détaillée et (iv) d'affichage de courbes. Des exemples de tels types de vues sont visibles en annexe D.

Le nombre de variables caractéristiques du procédé simulé en laboratoire est au nombre de sept. De ce fait, nous avons adopté pour la vue de supervision la représentation en vue étoile, sur laquelle est représenté en vert l'état courant des sept variables et en jaune, sous forme de scénario accéléré, leur évolution future, figure IV.13.

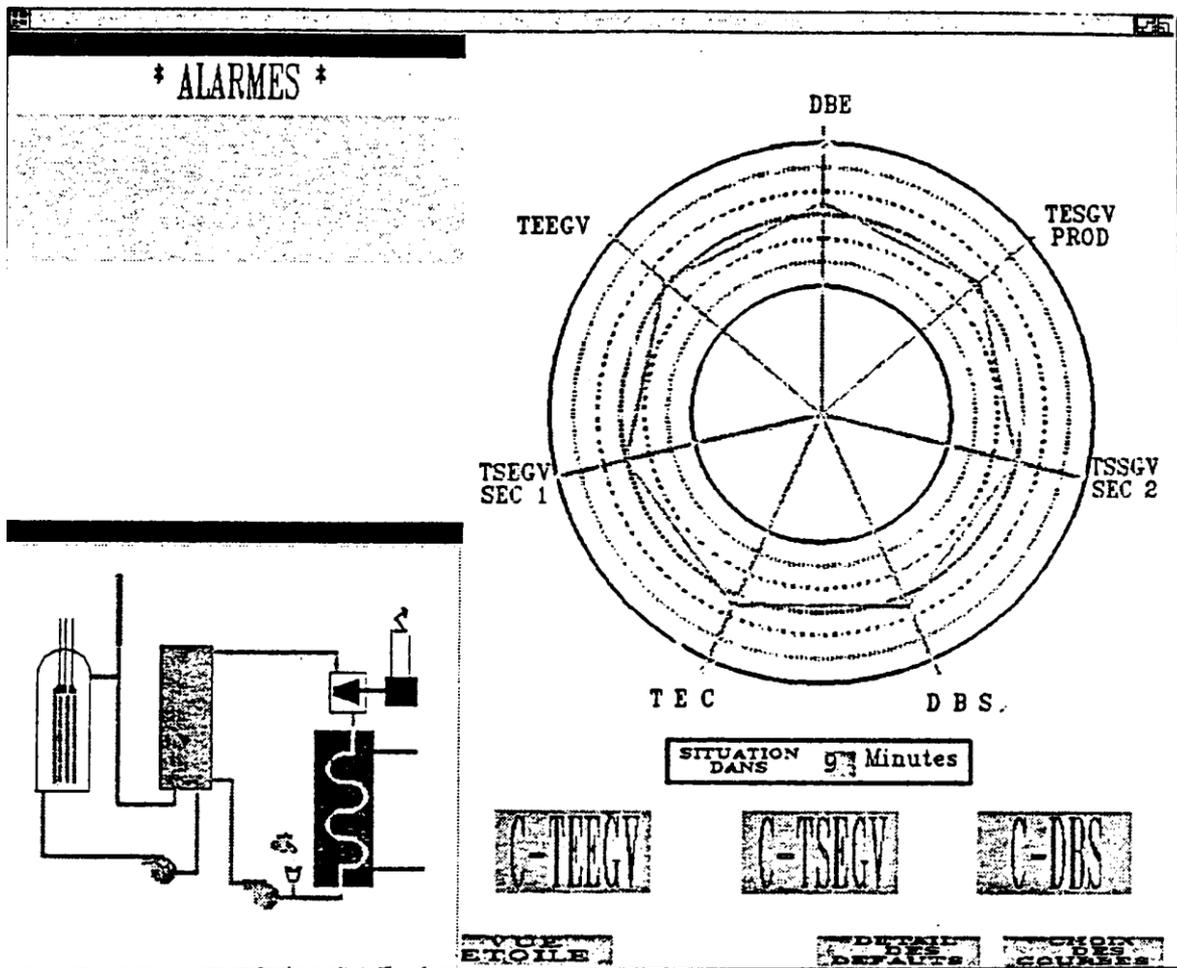


Figure IV.13 : Copie d'écran de la vue de supervision

Pour les vues de justification, nous avons adopté une représentation sous forme de graphe de fluence (Cf. figures IV.8 et IV.9 au paragraphe III.4.), combinée avec un codage de couleur pour considérer l'état de chaque variable dans le temps. Ainsi, nous avons adopté le codage suivant :

- si une variable de commande est loin de sa valeur de consigne, elle sera représentée dans un carré blanc, informant l'opérateur qu'une correction est en cours ou a été faite ;
- si une variable va dépasser le premier seuil d'alarme dans les minutes qui viennent, elle sera représentée dans un cercle jaune informant l'opérateur que c'est une information prédite ;
- si une variable a dépassé le premier seuil d'alarme, elle sera représentée dans un cercle vert, informant ainsi l'opérateur que c'est une information de l'état courant du procédé ;

- si une variable va dépasser le deuxième seuil d'alarme dans les minutes qui viennent, elle sera représentée dans un cercle orange informant l'opérateur que la situation va devenir grave dans les minutes qui viennent ;
- si une variable a dépassé le deuxième seuil d'alarme elle sera représentée dans un cercle rouge, informant ainsi l'opérateur que la situation actuelle est grave.

Pour les vues de type courbes, nous avons adopté une représentation qui autorise l'affichage d'un historique de 15 minutes et d'une prédiction de 15 minutes. De plus, l'opérateur peut sélectionner, suivant ses désirs, une, deux ou trois variables à visualiser sous forme de courbes. Ces variables sont, à chaque instant, hiérarchisées et présentées dans une liste suivant leur écart par rapport à leur valeur de consigne (Cf. l'exemple de vue D.7 en annexe D).

Cette bibliothèque graphique est pilotée par le gestionnaire "intelligent" de l'imagerie.

IV.4. REALISATION DU GESTIONNAIRE "INTELLIGENT" DE L'IMAGERIE

Ce paragraphe est consacré à la réalisation du gestionnaire "intelligent" de l'imagerie et à la construction des bases de connaissance, sur lesquelles le gestionnaire doit se baser pour déduire l'ensemble des informations à présenter à l'opérateur, en fonction des différents contextes de fonctionnement du procédé.

A ce niveau de la réalisation, nous avons à réaliser un système d'inférence capable de traiter et d'exploiter l'ensemble des connaissances relatives aux 3 problèmes : "QUOI", "QUAND" et "COMMENT". Le gestionnaire "intelligent" doit donc comporter :

- un moteur d'inférence ;
- une base de connaissance sur le "QUOI" ;
- une base de connaissance sur le "QUAND" ;
- une base de connaissance sur le "COMMENT".

Plusieurs faits multivalués peuvent facilement représenter les données manipulées par le moteur d'inférence.

La figure IV.14 explicite les entrées et les sorties d'informations utilisées par le moteur d'inférence du système. Celui-ci fonctionne en chaînage avant car on cherche à obtenir le plus de faits possibles : tous les faits déductibles sont déduits. Des faits représentent :

- l'état de fonctionnement du procédé tels que les faits : "Situation", "Gravité" et "Tâche opérateur" ;
- le type d'opérateur et ses requêtes éventuelles tels que : "Classe opérateur" et "Requête opérateur" ;
- l'état précédent de l'interface tel que : "QUOI précédent".

Ces faits font partie de la base de faits initiale. Les données nécessaires à la constitution de cette base sont fournies au système expert par un superviseur. Le moteur d'inférence utilise cette base de connaissance pour déduire de nouveaux faits : les faits QUOI, QUAND et COMMENT, qui correspondent à ce qu'il faut afficher, quand et comment l'afficher.

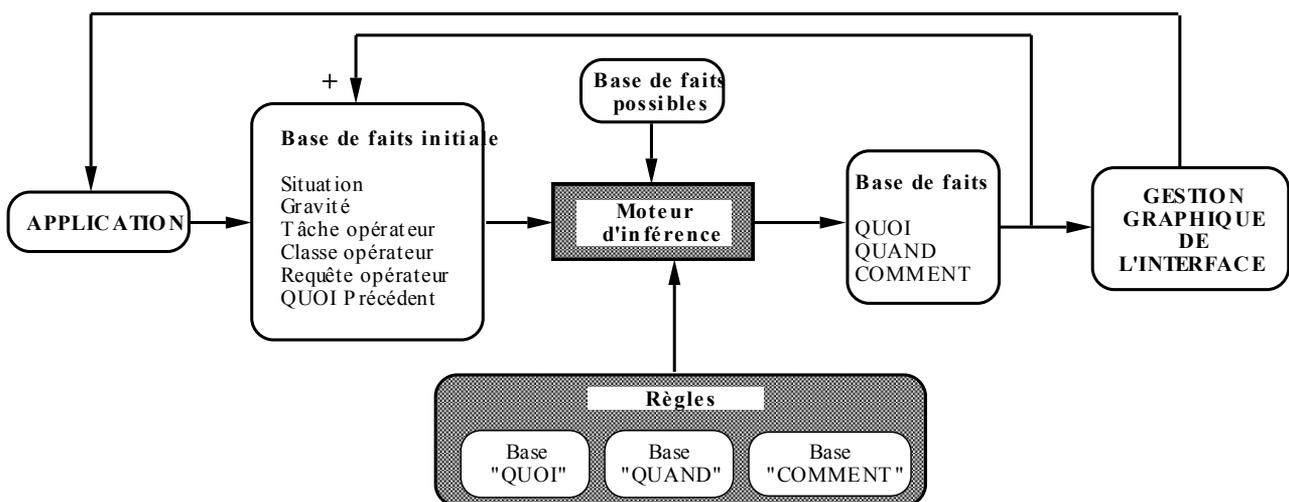


Figure IV.14 : Le moteur d'inférence du M.D.I.
(LE STRUGEON et al., 1992)

Nous avons choisi de développer notre propre moteur d'inférence à l'aide du langage C (LE STRUGEON, 1991). Pour déduire les faits QUOI, QUAND et COMMENT, qui correspondent à ce qu'il faut afficher, quand et comment l'afficher, le moteur d'inférence manipule neuf faits multivalués. Ces faits représentent l'état de fonctionnement du procédé (faits : SITUATION, GRAVITE et TACHE OPERATEUR), le type de l'opérateur et ses requêtes éventuelles (respectivement : CLASSE OPERATEUR et REQUETE OPERATEUR), et l'état précédent de l'interface (QUOI PRECEDENT). Le moteur commence par inférer sur le fait QUOI. Puis, en utilisant la Base de Faits augmentée de la (ou des) valeur(s) obtenue(s) pour le fait QUOI, les faits QUAND et COMMENT sont déduits. Afin de vérifier la validité des associations entre un fait et une liste de valeurs lors d'ajouts dans la Base de Faits, une base annexe est utilisée. Il s'agit de la "Base des Faits Possibles". Elle contient l'ensemble des faits acceptables ("possibles"), ainsi que pour chacun de ces faits, la liste des valeurs qui peuvent lui être associées.

Les règles manipulées par le moteur d'inférence sont créées, compilées et maintenues grâce à une méthode mise au point au laboratoire (TENDJAOUI et al., 1991A), figure IV.15. Cette méthode se compose de quatre étapes :

- La première étape conduit à un recensement de toutes les valeurs possibles liées à des critères de décision concernant la gestion de l'affichage par le M.D.I., par exemple : les degrés de gravité liés au procédé, les situations de fonctionnement, les besoins en outil d'aide liés aux tâches à accomplir par l'opérateur, etc. Ces informations définissent un ensemble de descripteurs caractérisant le système Homme-Machine. Un descripteur est un couple (attribut, valeur). Par exemple, à l'attribut "Situation de Fonctionnement du Procédé", peuvent être associées plusieurs valeurs, telles que : Normale, Anormale et Critique (modélisées précédemment).
- A partir de la base de faits possibles, nous avons mis en relation les critères de décision recensés avec les décisions possibles du M.D.I. Ces décisions concernent "QUOI présenter", "QUAND le présenter", "COMMENT le présenter". Une base d'exemples est générée lors de cette étape. La description des exemples est effectuée à l'aide de **9 attributs** relatifs au problème étudié. Chaque attribut possède un nombre fini de valeurs mutuellement exclusives.

- Sur la base d'exemples, nous utilisons des techniques issues de l'Apprentissage Symbolique Automatique pour générer des arbres de décision optimisés. Ces arbres sont générés à l'aide du système d'Intelligence Artificielle 1stClass. Cet outil se base sur l'algorithme ID3 (Interactive Dichotomizer 3), qui engendre à partir d'une collection d'exemples, appelée ensemble d'apprentissage, des procédures de classification exprimées sous la forme d'arbres de décisions (QUINLAN, 1979 ; 1983). Des exemples d'arbres de décisions générés à l'aide de 1stClass sont visibles en annexe A.

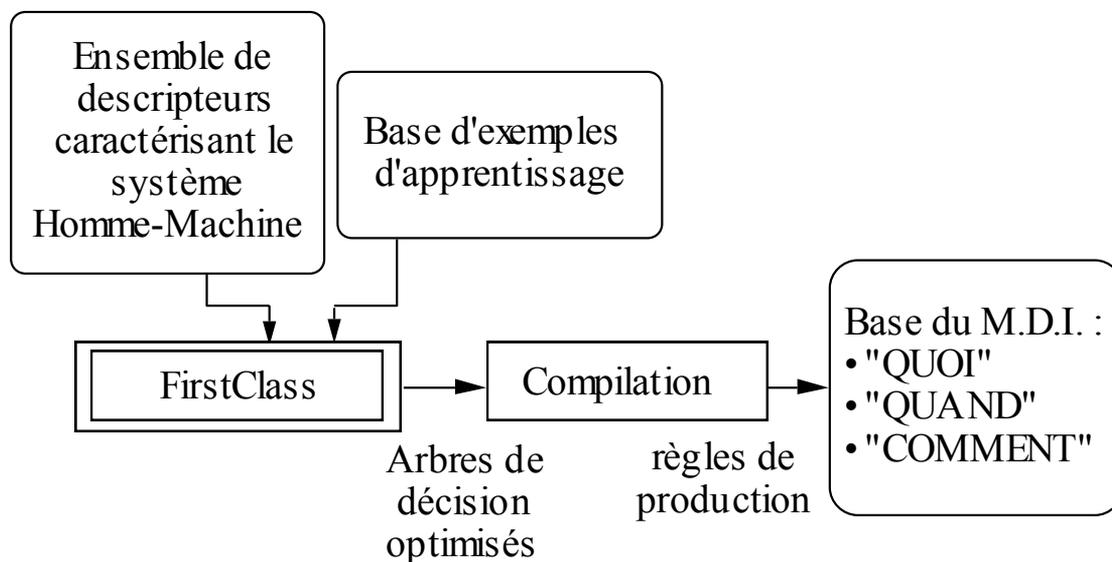


Figure IV.15 : Méthode de construction des règles

- La dernière étape a pour but la compilation de ces arbres sous un format exploitable par le M.D.I. Ainsi, ceux-ci sont transformés en règles de production utilisées par le moteur d'inférence décrit précédemment. L'arbre de décision est écrit dans un fichier de format ASCII (toujours par 1stClass). La technique utilisée est basée sur l'analyse syntaxique, qui a nécessité la définition d'une grammaire adéquate.

Le lecteur trouvera, comme exemple, douze règles de production issues des bases de connaissances du M.D.I. en annexe C.

CONCLUSION

Nous avons présenté dans ce chapitre une plate-forme expérimentale destinée à valider les concepts du Module Décisionnel d'Imagerie. Cette plate-forme est constituée principalement de plusieurs ensembles de modules : le Module Décisionnel d'Imagerie, l'outil d'assistance à l'opérateur intégrant un ensemble de fonctionnalités, des modules spécifiques permettant la gestion de la zone mémoire commune, et enfin un procédé simulé.

L'ensemble du développement de ces modules a été réalisé en langage C sur une station de travail de type VS 3100. Nous avons choisi, pour les aspects graphiques, l'outil DATAVIEWS qui propose un ensemble de deux produits : DVDRAW et DVTOOLS. Nous avons choisi de développer notre propre moteur d'inférence à l'aide du langage C. Les bases de connaissances du mécanisme d'inférence du M.D.I. ont été constituées par une méthode issue de l'Apprentissage Symbolique Automatique. Pour cela nous avons utilisé l'outil de développement de système expert 1stClass, qui engendre à partir d'une collection d'exemples des procédures de classification exprimées sous la forme d'arbres de décisions et transformables en règles de production.

Il faut souligner que, par rapport à l'ensemble de la plate-forme expérimentale, pour passer à une autre application que celle considérée dans ce chapitre, il suffirait d'adapter les modules spécifiques gérant les fonctionnalités d'assistance, ainsi que les trois bases de connaissances. Toutes les méthodes développées resteraient valables.

L'ensemble de la plate-forme expérimentale est opérationnel et une validation technique a été réalisée. Celle-ci fait l'objet du chapitre suivant.

Dans ce chapitre, nous proposerons également plusieurs perspectives de recherche contribuant à faire évoluer les principes de base du M.D.I.

CHAPITRE V

VALIDATION TECHNIQUE DE LA PLATE-FORME EXPERIMENTALE ET PERSPECTIVES

CHAPITRE V : VALIDATION TECHNIQUE DE LA PLATE-FORME EXPERIMENTALE ET PERSPECTIVES

INTRODUCTION

Dans le chapitre précédent, nous avons décrit les modules principaux de la plate-forme expérimentale qui devra permettre de valider les concepts d'interface "intelligente" énoncés dans le chapitre III. Rappelons que la plate-forme se compose principalement des modules suivants :

- le Module Décisionnel d'Imagerie,
- l'outil d'assistance à l'opérateur,
- les modules permettant la gestion de la zone mémoire commune,
- le procédé simulé.

Il s'agit d'évaluer et de valider dans un premier temps le fonctionnement de cette plate-forme expérimentale, et particulièrement les interactions entre les différents modules par l'intermédiaire de la zone mémoire commune. Il nous faudra vérifier, aussi, que le système complet d'assistance répond bien aux exigences techniques spécifiées dans le chapitre précédent. Cette phase de validation essentiellement technique fait l'objet de la première partie de ce chapitre.

Dans la deuxième partie de ce chapitre, nous présentons plusieurs perspectives de recherche. Celles-ci considèrent d'abord l'évaluation ergonomique du M.D.I. et de son apport vis-à-vis d'une interface "classique", où il importe de vérifier si l'utilisation d'une interface "intelligente" de type M.D.I. répond bien aux besoins d'assistance des opérateurs en matière. D'autres perspectives concernant l'évolution de certaines caractéristiques du M.D.I. sont proposées dans la dernière partie de ce chapitre.

I. PREMIERE VALIDATION TECHNIQUE DE LA PLATE-FORME EN LABORATOIRE

Afin de valider techniquement la plate-forme expérimentale et de vérifier que le système d'assistance, fourni aux opérateurs, répond bien aux exigences spécifiées a priori, nous avons établi un premier scénario de pannes. Celui-ci est d'abord expliqué ci-dessous. Ensuite, le déroulement d'une pré-expérimentation est décrit et fait l'objet de premiers commentaires.

Rappelons que le but de cette pré-expérimentation est de vérifier le fonctionnement de l'ensemble des modules constituant la plate-forme expérimentale.

I.1. DESCRIPTION DU SCENARIO UTILISE

Le scénario utilisé pour la validation technique est décrit dans le tableau V.1. Le scénario consiste à créer quatre défauts, à chaque fois espacés dans le temps de trois minutes. Les effets de chaque défaut, selon la configuration actuelle du simulateur, sont décrits ci-dessous.

Tableau V.1 : Description d'un scénario de panne

Date d'activation du défaut	N° et nom de la variable en défaut	Amplitude du défaut à créer en % de la Valeur Nominale	Premier(s) symptôme(s) engendrés	Liste des variables concernées par le défaut
+3 mn	11 : C-DBE	+50	TSSGV + TESGV -	3 : TEEGV 7 : TSSGV 8 : TESGV 9 : DBE
+6 mn	10 : NEC	+40	TSSGV +	3 : TEEGV 5 : TEC 7 : TSSGV
+9 mn	11 : C-DBE	-50	TSSGV - TESGV +	3 : TEEGV 7 : TSSGV 8 : TESGV 9 : DBE
+12 mn	10 : NEC	-40	TSSGV -	3 : TEEGV 5 : TEC 7 : TSSGV

Dans le cas du premier et du troisième défaut, les variables concernées sont présentées sur la figure V.1. Une augmentation (respectivement diminution) de la valeur de consigne du débit d'eau (C-DBE), va entraîner successivement les effets suivants :

- une augmentation (respectivement diminution) du débit d'eau dans le secondaire (DBE) ;
- puis, une augmentation (respectivement diminution) de la température de l'eau à l'entrée du générateur de vapeur (TEEGV) ;

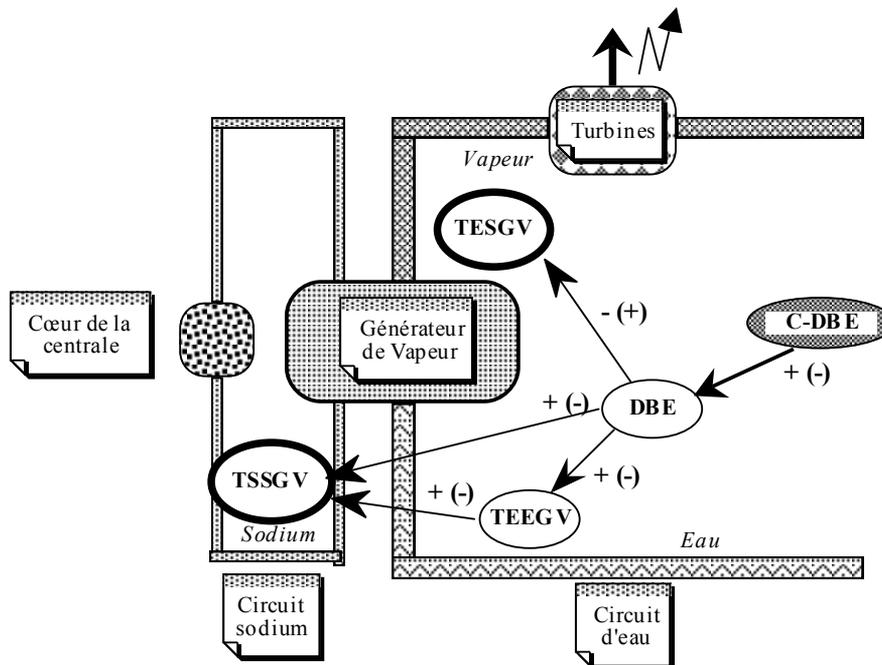


Figure V.1 : Symptôme(s), cause(s) et variables concernés par les défauts 1 et 3 du scénario.

- une augmentation (respectivement diminution) de la température de l'eau à la sortie du générateur de vapeur (TESGV) ;
- une diminution (respectivement augmentation) du gradient⁴ de température entre les deux circuits et par conséquent une augmentation (respectivement diminution) de la température du sodium à la sortie du générateur de vapeur (TSSGV).

⁴ Rappelons que toute **augmentation** du gradient de température entre le circuit primaire et le circuit secondaire entraîne une **diminution** de la température de sortie du primaire (le plus **chaud**) car il cède moins de chaleur au secondaire.

Dans le cas du deuxième et du quatrième défaut, les variables concernées sont présentées sur la figure V.2. Une augmentation (respectivement diminution) du niveau d'eau dans le condenseur (NEC), va entraîner successivement les effets suivants :

- une diminution (respectivement augmentation) de la température de l'eau dans le condenseur (TEC) ;

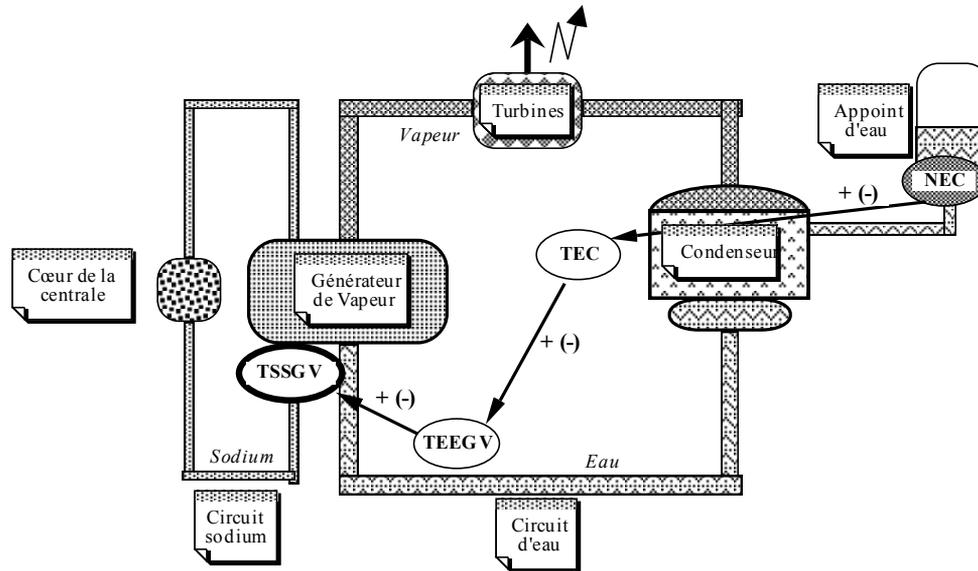


Figure V.2 : Symptôme(s), cause(s) et variables concernées par les défauts 2 et 4 du scénario.

- puis, une diminution (respectivement augmentation) de la température de l'eau à l'entrée du générateur de vapeur (TEEGV) ;
- une diminution (respectivement augmentation) du gradient de température entre les deux circuits et donc une augmentation (respectivement diminution) de la température du sodium à la sortie du générateur de vapeur (TSSGV).

Pour ce scénario, il a été demandé à un premier sujet d'utiliser la plate-forme, mais selon un mode particulier précisé ci-après.

I.2. DESCRIPTION DU DEROULEMENT DE LA PRE-EXPERIMENTATION

Dès que l'ensemble des modules est lancé, les variables du procédé sont mises à leur valeur de consigne de sorte que la situation de fonctionnement soit optimale.

Quelques moments plus tard (3 minutes dans le cas du scénario décrit dans le tableau V.1), le premier défaut apparaît.

Les données enregistrées lors de cette pré-expérimentation étaient : le niveau de gravité engendré par la situation de fonctionnement du procédé à un instant donné, le type de vue affichée dans la fenêtre principale, le type de vue affichée dans la fenêtre secondaire, et les actions du sujet (Cf paragraphe V.2 dans le chapitre IV et figures D.1 à D.5 dans l'annexe D).

Dans le but de pouvoir identifier l'activité du sujet, et ceci tout au long de l'expérimentation, nous lui avons demandé de sélectionner certains objets graphiques en les visant à l'écran grâce à l'utilisation de la souris. De cette manière, il est possible à chaque instant d'identifier et de mémoriser les six activités suivantes :

- détection d'une alarme : cette activité peut être identifiée si le sujet vise l'objet texte correspondant à l'alarme, et ceci dès son apparition ;
- détection d'une alarme et lecture d'un plan d'action : si le sujet vise l'objet texte contenant le plan d'action au moment de sa lecture ;
- recherche des causes possibles d'un défaut : si le sujet vise les objets graphiques contenus dans la vue de justification de surface et symbolisant les causes suspectées ;
- recherche de la cause d'un défaut : si le sujet vise l'(es) objet(s) graphique(s) contenue(s) dans la vue de justifications profondes et symbolisant la (les) cause(s) identifiée(s) ;
- correction d'un défaut : si le sujet confirme le changement de valeur de consigne de l'une des variables de commande ;
- suivi d'une action de correction : si, suite à une action de correction, le sujet n'est pas dans l'une des cinq activités précédentes, alors il est possible de conclure qu'il est en train de vérifier si sa correction rétablit bien les conditions de fonctionnement normales du procédé en utilisant notamment les informations de prédiction.

Les premières données obtenus lors de cette pré-expérimentation font l'objet du paragraphe suivant.

I.3. RESULTATS DE LA PRE-EXPERIMENTATION : DISCUSSION VIS-A-VIS DES DECISIONS PRISES PAR LE M.D.I.

L'évolution dans le temps des paramètres recueillis lors de la pré-expérimentation est résumée dans la figure V.3. On y retrouve :

- les décisions du M.D.I. représentées par la gestion de l'affichage des alarmes et des plans d'actions, ainsi que par le contenu de la fenêtre secondaire ;
- les réponses aux requêtes du sujet représentées par le contenu de la fenêtre principale ;
- l'activité du sujet ;
- l'évolution de la gravité de la situation (évaluée en permanence par le M.D.I.) engendrée par le fonctionnement du procédé tout au long de l'expérience.

Notre objectif sera ici **d'expliquer les décisions prises par le M.D.I.** au fur et à mesure des vingt minutes de cette pré-expérimentation, sans entrer bien sûr dans le détail de toutes les inférences.

I.3.1. Déroulement entre T0 et T1

Au début du scénario (entre T0 et T1), suite à la détection du premier défaut, le M.D.I. affiche une alarme, détectée instantanément par le sujet. De plus, il propose au sujet un ensemble de causes possibles, grâce aux justifications de surface qu'il affiche automatiquement dans la fenêtre secondaire. En effet, le M.D.I. répond aux questions "QUOI", "QUAND" et "COMMENT" de la manière suivante :

- "QUOI" : le M.D.I. déduit que le sujet a besoin de **plans d'actions** et de **justifications de surface** concernant ce premier défaut ;
- "QUAND" : parce que la situation n'est pas grave, le M.D.I. déduit que les plans d'actions doivent être affichés à **mi-échéance**, et que les justifications de surface sont à afficher **maintenant** ;
- "COMMENT" : le M.D.I. déduit que les justifications de surface doivent être affichées sous forme d'un **graphe de fluence**.

Dans ces conditions, le M.D.I. oriente ainsi le sujet dans sa recherche de la (les) cause(s) possible(s) du défaut, car la gravité engendrée par le fonctionnement du procédé n'est pas encore grande et par conséquent la situation n'est pas grave (voir la fonction de gravité expliquée dans le chapitre III, paragraphe II.4).

Pendant ce temps, le sujet agit de la façon suivante :

- il détecte l'alarme affichée dans la zone en haut et à gauche de l'écran ;
- par la suite, il cherche à trouver l'ensemble des causes possibles en s'aidant des justifications de surface affichées par le M.D.I. dans la fenêtre secondaire ;
- il exécute les actions de correction ;
- puis une nouvelle alarme est affichée par le M.D.I. peu après T1 correspondant à l'arrivée du deuxième défaut.

I.3.2. Déroulement entre T1 et T2

Dans la suite du scénario (entre T1 et T2), le M.D.I affiche simultanément l'alarme et le plan d'actions correspondant. Dans ce cas, la situation de fonctionnement du procédé est assez grave (comprise entre 6 et 8) et le sujet doit rétablir, le plus rapidement possible, le bon fonctionnement de l'installation. De la même façon que précédemment, le M.D.I. propose au sujet l'ensemble des causes possibles dans la fenêtre secondaire. Dans ce cas, le M.D.I. répond aux questions "QUOI", "QUAND" et "COMMENT" de la manière suivante :

- "QUOI" : le M.D.I. déduit que le sujet a besoin des **plans d'actions** et des **justifications de surface** ;

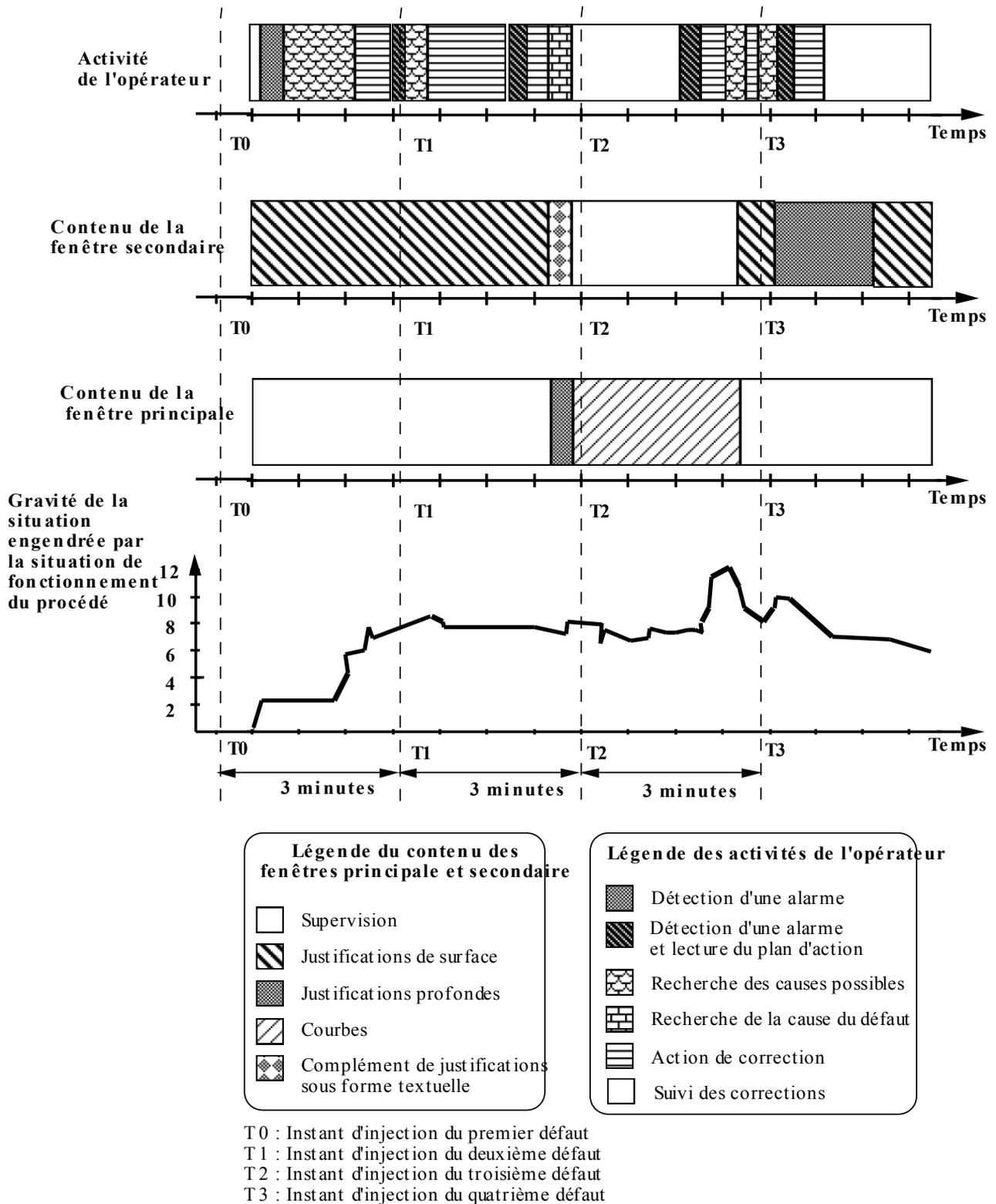


Figure V.3 : Résultats des expérimentations

- "QUAND" : parce que la situation est assez grave, le M.D.I. déduit que les plans d'actions et les justifications de surface doivent être affichés **maintenant** ;
- "COMMENT" : le M.D.I. déduit que les plans d'actions doivent être affichés sous forme **textuelle** au même format que les alarmes, et que les justifications de surface doivent être affichées sous forme d'un **graphe de fluence**.

Remarquons sur la figure V.3 que lors de l'arrivée du deuxième défaut, le comportement du sujet change car le niveau de gravité engendré par l'état de fonctionnement du procédé reste à un niveau élevé (environ 8). Le sujet a pu s'en apercevoir au niveau du plan d'actions proposé, sur lequel il est indiqué qu'il doit compenser le défaut dans un délai très court. Dans ce cas, le sujet agit de la façon suivante :

- dans un premier temps, il détecte l'alarme et lit le plan d'actions correspondant (affichés au même moment par le M.D.I.) ;
- par la suite, il cherche à trouver les causes possibles du défaut en s'aidant de la vue de justification de surface (présentée automatiquement par le M.D.I. dans la fenêtre secondaire), mais pendant ce temps, la gravité de la situation continue d'augmenter ;
- il exécute une partie du plan d'action proposé ;

A ce moment, le M.D.I. refait une évaluation de la situation et réactualise le plan proposé par l'affichage d'une **nouvelle alarme** et le **reste des actions du plan** proposé précédemment. Il répond aux questions "QUOI", "QUAND" et "COMMENT" de la même façon que précédemment. A partir de ce moment, le sujet agit de la façon suivante :

- il détecte une nouvelle alarme, qui a été affichée par le M.D.I. en même temps que le nouveau plan d'actions correspondant ;
- il exécute le plus rapidement possible le reste des actions du plan précédent, dans la mesure où il est en accord avec ce plan d'actions ;
- il demande des justifications profondes, qui lui sont affichées avec le maximum de détail dans la fenêtre principale, ces justifications étant

associées à des compléments d'explication présentés textuellement dans la fenêtre secondaire ;

- enfin, il demande la vue courbe pour visualiser l'historique des variables concernées par l'alarme afin de suivre l'évolution de celles-ci suite à ses corrections.

A l'instant T2, le troisième défaut survient.

I.3.3. Déroulement entre T2 et T3

Entre T2 et T3, le niveau de gravité engendré par le fonctionnement du procédé augmente brutalement alors que le sujet est en train de visualiser l'historique de certaines variables. Dans ce cas, notons que le M.D.I. a détecté le problème environ une minute avant son apparition, et a affiché une alarme et le plan d'actions correspondant, et laissé dans la fenêtre secondaire la vue de supervision. Le M.D.I. répond aux questions "QUOI", "QUAND" et "COMMENT" de la manière suivante :

- "QUOI" : le M.D.I. déduit que le sujet a besoin des **plans d'actions** car le principal but est de compenser le défaut et de baisser le niveau de gravité ; le M.D.I. ne propose pas de justifications au sujet car celui-ci n'a plus le temps de les consulter ;
- "QUAND" : parce que la situation est grave, le M.D.I. déduit que les plans d'actions doivent être affichés **maintenant** ;
- "COMMENT" : le M.D.I. déduit que les plans d'actions doivent être affichés sous forme **textuelle** au même format que les alarmes ;

Le sujet est en accord avec le plan d'actions proposé. Dans ce cas, le sujet agit de la façon suivante :

- il détecte une alarme et lit le plan d'actions correspondant ;
- Il exécute une partie du plan, en corrigeant d'abord la variable la plus prioritaire d'après le plan d'actions ;

A ce moment, le M.D.I. refait une évaluation de la situation et réactualise le plan proposé par l'affichage d'une nouvelle alarme et le reste des actions du plan proposé précédemment. Par la suite, le sujet agit de la façon suivante :

- il demande des justifications de surface pour localiser la cause du défaut ;
- Il exécute le reste du plan d'action proposé précédemment.

I.3.4. Déroulement à partir de T3

Pour la fin du scénario (à partir de T3), le M.D.I. prévoit l'imminence d'une situation très grave. Dans ce cas, il propose au sujet une alarme, le plan d'actions correspondant et affiche dans la fenêtre secondaire les justifications détaillées du plan proposé. Le M.D.I. répond aux questions "QUOI", "QUAND" et "COMMENT" de la manière suivante :

- "QUOI" : le M.D.I. déduit que le sujet a besoin des **plans d'actions** car le principal but est de compenser le défaut et de baisser le niveau de gravité. De plus, il propose l'affichage des **justifications profondes**, car la situation à ce moment n'est pas encore très grave.
- "QUAND" : parce que la situation risque de devenir très grave, le M.D.I. déduit que les plans d'actions et les justifications profondes doivent être affichés **maintenant** ;
- "COMMENT" : le M.D.I. déduit que les plans d'actions doivent être affichés sous forme **textuelle** et les justifications profondes sous forme de **graphe de fluence** ;

Dans ce cas, on peut constater que le sujet agit de la façon suivante :

- dans un premier temps, il détecte une alarme et lit le plan d'actions correspondant ;
- il exécute l'ensemble des actions du plan proposé car il dispose de tous les détails sur l'origine du défaut grâce à la vue de justifications profondes présentée automatiquement par le M.D.I. dans la fenêtre secondaire ;
- enfin, il suit simplement l'évolution des variables concernées par l'alarme pour remarquer que le fonctionnement du procédé redevient progressivement normal.

Ensuite, puisque le niveau de gravité engendré par la situation de fonctionnement du procédé est inférieur à 6 et continue à baisser (voir la figure V.3), le M.D.I. décide de remplacer dans la fenêtre secondaire la vue de justifications

profondes par une vue de justifications de surface, qui indique qu'il n'existe plus de causes de défauts possibles.

I.3.5. Premières conclusions

Au vu des données recueillies, il est possible de faire les premières interprétations suivantes sur le fonctionnement des différents modules de la plateforme, qu'il faudra bien entendu confirmer par d'autres pré-expérimentations :

- Le gestionnaire "intelligent" fonctionne. En effet, celui-ci a pu répondre aux trois questions "QUOI", "QUAND" et "COMMENT" en présentant automatiquement et de manière cohérente des informations dans la fenêtre secondaire. Des exemples d'affichages dans cette zone sont visibles en annexe D.
- Dans la mesure où le simulateur du procédé est opérationnel, il devient évident que le module de prédiction fonctionne, car il correspond tout simplement à une version accélérée du simulateur.
- Le module de traitement des alarmes a été testé également. Il est capable de proposer des alarmes hiérarchisées. Un exemple de page-écran est visible sur la figure V.4.
- Le module de génération des plans d'actions fonctionne (voir les exemples en annexe D). De plus, si le sujet n'exécute qu'une partie du plan proposé, un nouveau plan est recalculé automatiquement.
- Le module de justification peut proposer l'ensemble des causes possibles ainsi que l'ensemble des variables concernées par un défaut. Des exemples sont visibles en annexe D.

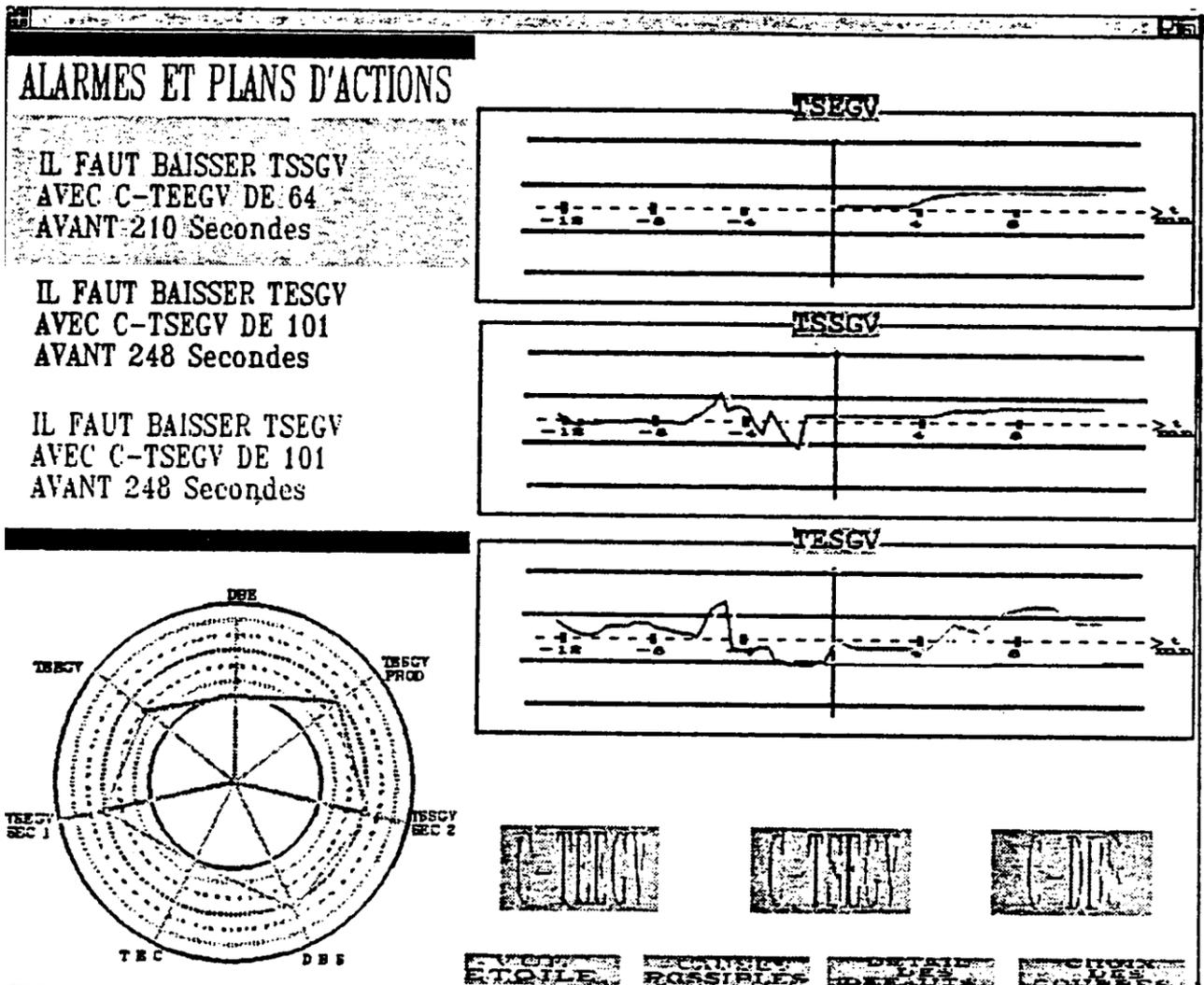


Figure V.4 : exemple de copie d'écran

La plate-forme expérimentale mise en œuvre paraît donc globalement opérationnelle. Il est maintenant possible d'entreprendre l'étape suivante consistant à évaluer ergonomiquement le M.D.I. Ceci n'a pas été réalisé dans le cadre de cette thèse, et fait donc l'objet de nos perspectives de recherche.

A ce sujet, l'ensemble de nos perspectives fait l'objet de la partie suivante.

II. PERSPECTIVES DE RECHERCHE

Le premier objectif visé par cette recherche était de proposer une première structure d'interface "intelligente" et d'intégrer celle-ci dans une plate-forme expérimentale en vue de son évaluation. L'objectif étant atteint, il importe maintenant d'évaluer ergonomiquement et de prévoir l'évaluation des concepts du M.D.I.. Nos perspectives de recherche sont à ce sujet successivement décrites.

II.1. EVALUATION ERGONOMIQUE DU M.D.I.

Nos perspectives de recherche à court terme consistent en l'évaluation ergonomique du M.D.I. à l'aide de la plate-forme expérimentale, et ceci dans le cadre d'une collaboration entre le L.A.I.H. et des laboratoires de Psychologie du Travail.

Cette phase aura pour objectif principal de vérifier que le système d'assistance fourni aux opérateurs répond à un ensemble de critères d'évaluation pré-établis. Ces critères pourront considérer les performances du système homme-machine, calculés en termes d'écart entre les objectifs du système et les résultats effectivement obtenus et/ou ergonomiques liés à l'utilisation du système de dialogue, du système d'assistance et à ses conséquences vis-à-vis de la réalisation des tâches dans les différents contextes opérationnels du procédé. Par exemple, l'évaluation portera sur l'adéquation des informations présentées aux besoins réels de l'opérateur pour réaliser ses tâches, la facilité d'accès aux différentes vues et aux fonctions de l'interface, l'enchaînement des vues en fonction de la démarche cognitive de l'opérateur, les temps d'accès à l'information, la charge de travail de l'opérateur, etc.

Il existe actuellement de nombreuses méthodes susceptibles de contribuer à l'évaluation des systèmes d'assistance à l'opérateur (voir WILSON et CORLETT, 1990 ; ABED, 1990 ; MILLOT, 1990 ; SENACH, 1990, etc). En situation réelle, le choix d'une ou plusieurs méthodes d'évaluation dépend de leur conditions d'application et notamment de disposer de la salle de contrôle et de l'équiper dans le cadre de cette évaluation. En effet, cette évaluation peut alors distraire les opérateurs, perturbant ainsi la production. C'est pourquoi il est souvent nécessaire d'évaluer d'abord le système homme-machine dans un contexte simulé (TABORIN, 1989). Ce sera de toute manière le cas des premières expérimentations que nous visons à mettre en place.

A ce sujet, d'après MILLOT (1990), il s'agit de souligner que la multiplicité des paramètres qu'il est nécessaire d'observer ou d'estimer est liée d'une part à la complexité du comportement humain et d'autre part, au fait que chacun de ces paramètres n'est pas assez représentatif individuellement et pas assez affiné pour caractériser à lui seul l'état du système "opérateur humain exécutant une tâche". Dans ces conditions, la mise en oeuvre de méthodes d'analyse d'activité s'imposera sans doute.

Enfin, il est également envisagé d'évaluer l'apport du M.D.I. par rapport à une interface "classique". L'expérience consistera à faire passer des groupes d'opérateurs face à une interface de type classique et/ou face au M.D.I. avec la même série de scénarios, pour superviser un système simulé, figure V.5. Lors de ces expériences, un ensemble de données objectives et subjectives pourront être mesurées, et après analyse et comparaison, pourront nous amener soit à changer certaines spécifications du M.D.I., soit à refaire de nouvelles expériences avec de nouveaux scénarios.

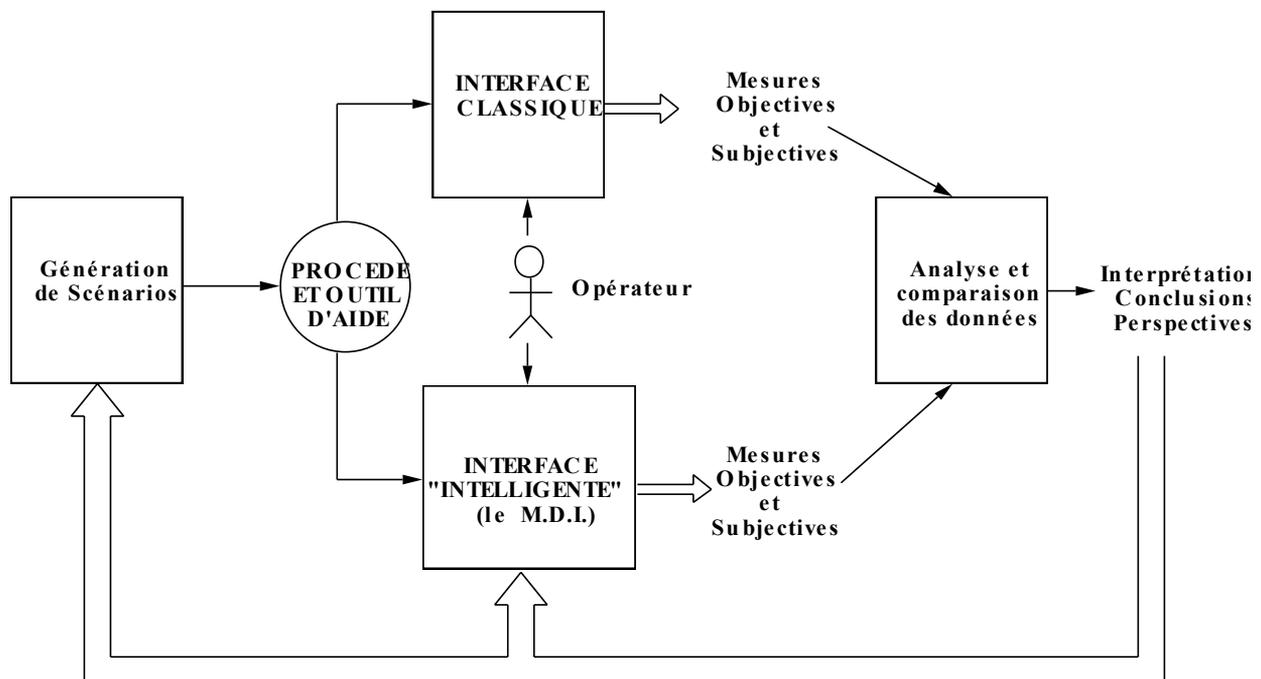


Figure V.5 : Comparaison entre interface "classique" et interface "intelligente"
(TENDJAOUI et al., 1991A)

L'évaluation ergonomique du M.D.I. à l'aide de la plate-forme expérimentale nous amènera à faire évoluer cette approche d'interface "intelligente". A ce sujet, un ensemble d'évolutions sont d'hors et déjà envisagées. Elles sont présentées ci-dessous.

II.2. EVOLUTION DU M.D.I.

En fonction des résultats issus de la phase précédente d'évaluation, il s'agira en particulier de reprendre chacun des descriptifs des concepts "QUOI", "QUAND" et "COMMENT" du M.D.I. (cf. le paragraphe III dans le chapitre III) et d'aller plus loin dans leur mise en œuvre informatique et leur utilisation.

Ces concepts sont rappelés sur la figure V.6, et successivement discutés :

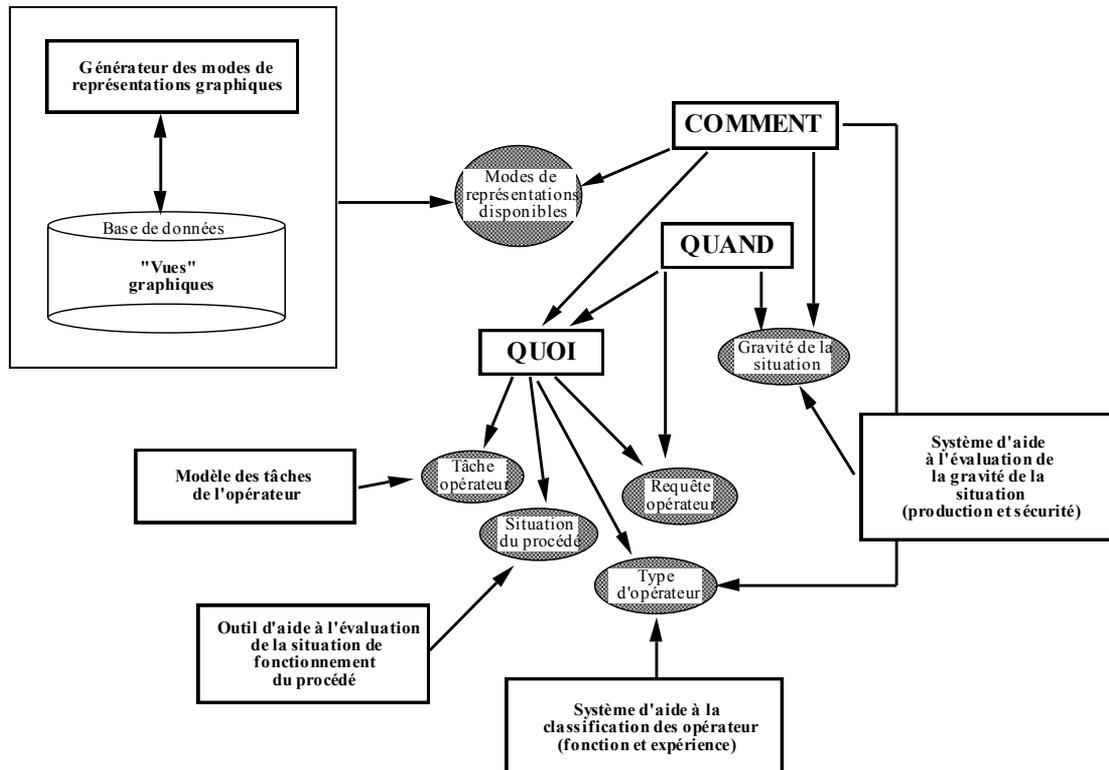


Figure V.6 : Descriptifs à faire évoluer

- la tâche en cours de réalisation par l'opérateur ;
- le type (classe) de l'opérateur ;
- les modes de présentation disponibles ;
- la situation de fonctionnement du procédé ;
- la gravité de la situation engendrée par le fonctionnement du procédé ;
- les requêtes de l'opérateur.

II.2.1. Le descriptif "tâche de l'opérateur"

Il semble évident que l'intégration d'un module capable d'identifier l'ensemble des tâches de l'opérateur humain serait d'un apport essentiel pour les performances du système global homme-machine. En effet, si le M.D.I. est capable d'identifier de façon assez précise la tâche de l'opérateur humain, il sera capable de mieux répondre aux questions "QUOI", "QUAND" et "COMMENT", car il disposerait d'une information beaucoup plus détaillée concernant les besoins de l'opérateur humain et les décisions que celui-ci doit prendre.

Un principe envisageable pour ce module serait d'analyser les actions de l'opérateur et l'état de fonctionnement du procédé durant une période qui resterait à définir, figure V.7. Ce module pourrait dans ce but utiliser des connaissances pré-établies pour identifier la tâche de l'opérateur.

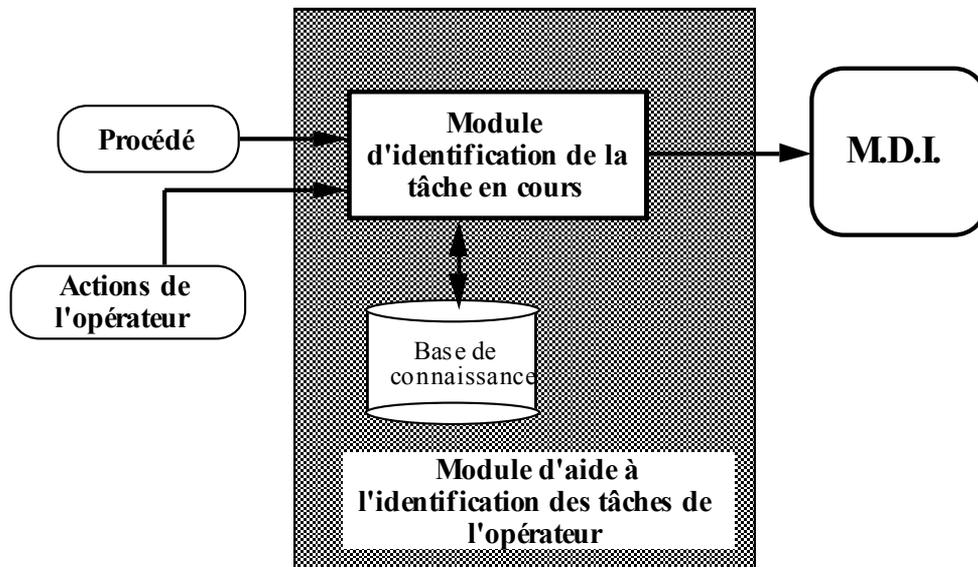


Figure V.7 : Module d'aide à l'identification de la tâche de l'opérateur

Les connaissances utilisées pourraient s'inspirer des résultats issus de la phase d'évaluation. Elles pourraient être du type suivant :

	Si Etat_Procédé = Anormal
et	Si Activité_Opérateur = détection d'une alarme
et	Si Actions_Opérateur = Correction
	Alors Tâche_Opérateur = Suivi d'une action de correction

voire même :

	Si Etat_Procédé = Anormal
et	Si Activité_Opérateur = détection d'une alarme
et	Si Actions_Opérateur = Aucune
puis	
	Si Etat_Procédé = Anormal
et	Si Activité_Opérateur = Recherche des causes du défaut
et	Si Action_Opérateur = Demande de justifications
	Alors Tâche_Opérateur = Elaboration d'un plan de correction

Dans ces deux exemples, le paramètre **puis** représenterait une succession dans les tâches, les actions ou les états de fonctionnement du procédé. Cependant, cette description des connaissances à mettre en œuvre doit bien entendu être approfondie.

II.2.2. Le descriptif "type d'opérateur"

Dans le cadre de notre étude, nous n'avons considéré qu'une seule fonction de l'opérateur (opérateur de la salle de contrôle) et que trois niveaux d'expertise de celui-ci : débutant, expérimenté et expert. Or, il serait plus intéressant d'élargir notre étude à un plus grand nombre de fonctions (technicien de maintenance, rondier, chef de quart, opérateur de la salle de contrôle) et éventuellement à plus de niveaux d'expertise.

Si, pour le premier point, il suffirait d'effectuer une identification a priori de la fonction de l'opérateur à partir du nom de celui-ci, le problème n'est pas aussi simple quand il s'agit d'évaluer l'évolution de son expérience tout au long de son utilisation du système d'assistance.

Pour résoudre partiellement ce problème, nous nous proposons d'intégrer au M.D.I. un module capable d'évaluer les performances de chaque opérateur et d'attribuer des points à celui-ci, figure V.8. L'attribution de ces points pourrait varier en fonction d'un ensemble de critères, tels que :

- le temps d'utilisation par l'opérateur du système,
- la moyenne du taux de production assuré par l'opérateur,
- les seuils de sécurité assurés par l'opérateur, etc.

Pour assurer cette fonctionnalité, le module considéré devrait gérer une base de données centralisant les performances de chaque opérateur selon des critères prédéfinis. Un problème crucial à résoudre est bien entendu celui du passage d'un niveau à l'autre. Ce passage doit nécessairement être validé, et donc validable, par les experts du système.

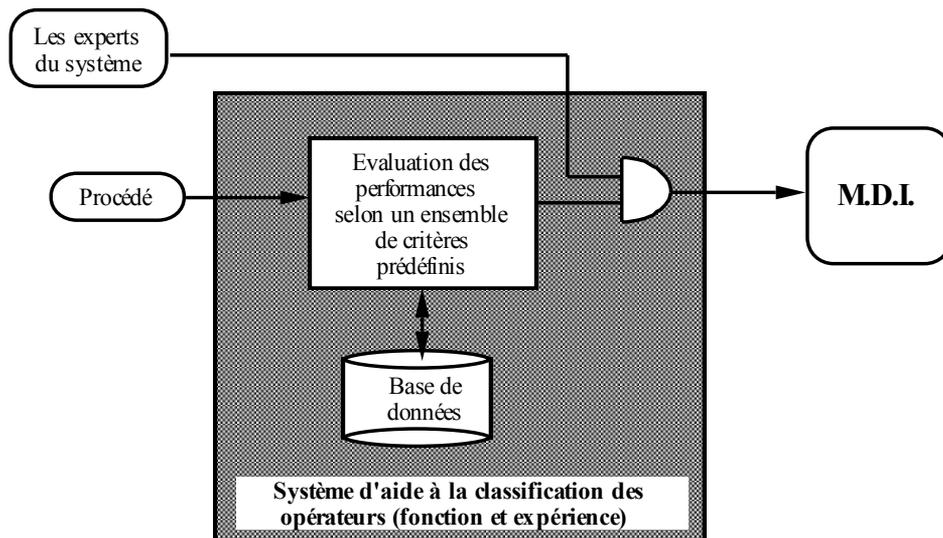


Figure V.8 : Module d'aide à la classification des opérateurs humains

II.2.3. Le descriptif "modes de présentation disponibles"

En raison de la multiplicité des valeurs des différents descriptifs caractérisant les réponses à la question "QUOI présenter à l'opérateur", il apparaît clairement que le nombre de possibilités de réponses aux questions "COMMENT présenter le QUOI" à l'opérateur peut devenir très important. Par conséquent, il importe d'associer au M.D.I. un module capable de générer des vues graphiques respectant a priori un ensemble de critères ergonomiques de présentation de l'information. Celui-ci devrait être capable de fournir au M.D.I. une base de données graphique, sous la forme d'un ensemble d'objets. Chaque objet de cette base serait alors associé à une description du comportement graphique en dynamique de celui-ci. A cet effet, un système envisagé pour générer les vues graphiques utilisées par le M.D.I. est ERGO-CONCEPTOR, décrit par MOUSSA (1992) et MOUSSA et KOLSKI (1992). Ce système est développé à l'aide du langage LELISP et de l'environnement de développement d'applications graphiques constitué de la boîte à outils AIDA et du générateur d'interfaces MASAI, sur station de travail de type VS 3100. ERGO-CONCEPTOR se compose de trois modules interconnectés, figure V.9 :

- Le premier module sert à décrire un procédé selon une méthodologie de description de procédés présentée dans la thèse de MOUSSA (1992). Ce module permet particulièrement la gestion des variables, de groupements fonctionnels et du réseau d'influence entre les différentes variables ou sous-systèmes du procédé à contrôler, à l'aide d'une interface spécifique. Cette description doit être guidée par les besoins informationnels des opérateurs.

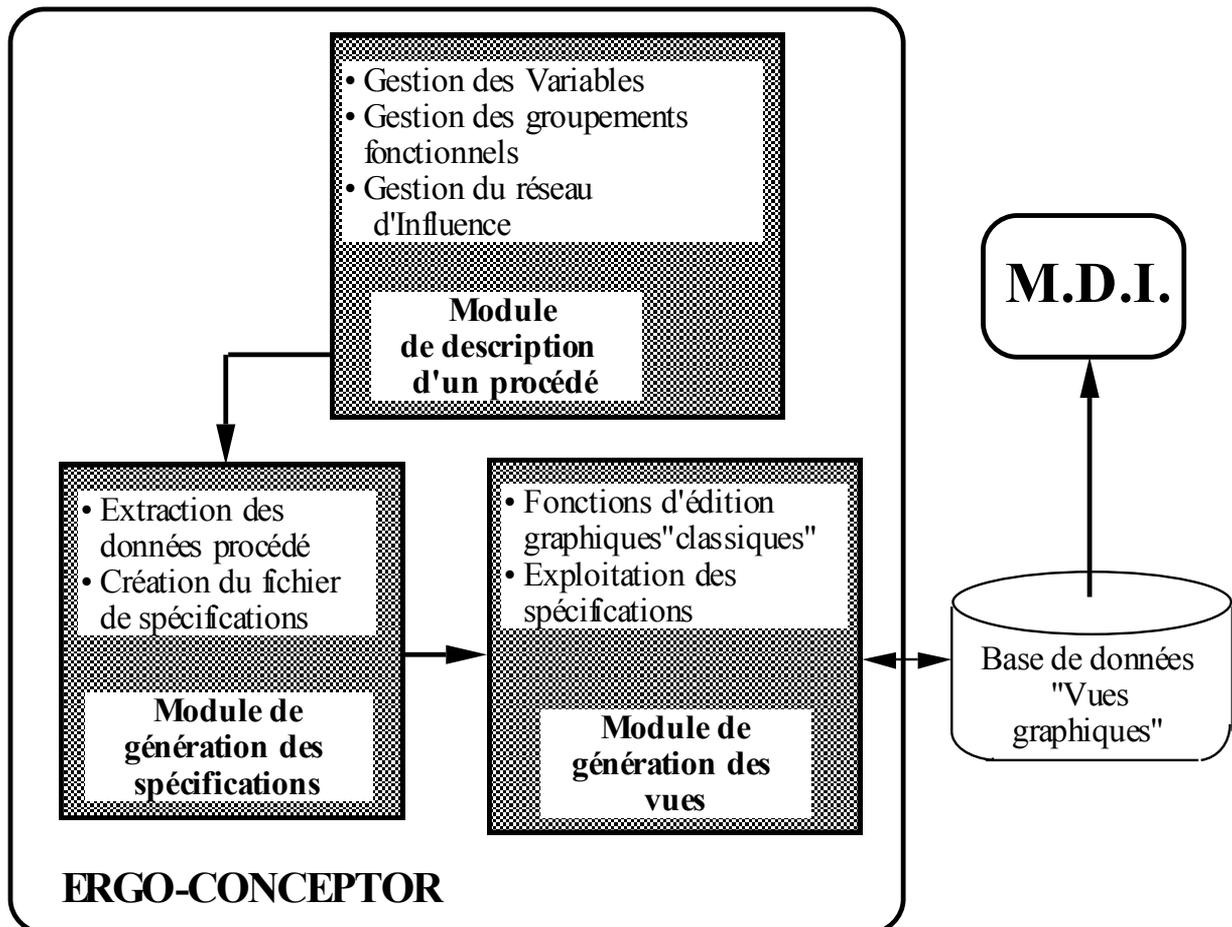


Figure V.9 : Connexion d'un générateur de vues graphiques au M.D.I.

- Le second module permet la génération automatique, à partir des résultats issus du module précédent, d'un fichier de spécifications d'interface en exploitant des connaissances ergonomiques stockées dans une base de connaissances. Ces connaissances ergonomiques concernent la structuration des vues, les modes de représentation et les attributs graphiques.
- Enfin, le dernier module assure la génération des vues graphiques, et ce en proposant au concepteur un éditeur lui permettant de créer interactivement les vues. Cet éditeur peut être dit "intelligent" du fait qu'il exploite les

spécifications déduites du module précédent. L'interface assure toutefois au concepteur des degrés de liberté lui permettant d'intervenir au niveau le plus bas de la conception des vues, à l'aide de fonctions d'édition graphique "classiques".

Cependant, pour assurer la connexion entre ERGO-CONCEPTOR et le M.D.I., il faudrait, dans un premier temps, harmoniser les outils graphiques utilisés par ces deux systèmes.

II.2.4. Les descriptifs "situation de fonctionnement du procédé" et "gravité de la situation du procédé"

Lors de notre étude, nous avons proposé un module capable d'estimer la situation de fonctionnement du procédé ainsi que la gravité de la situation. Nous avons défini ces deux descriptifs grâce à trois paramètres spécifiques traduisant l'état de fonctionnement du système. Ces paramètres sont le degré d'insécurité du système, le degré d'anormalité de la production d'énergie et le degré d'urgence.

Dans le cas plus général, il serait intéressant d'intégrer au M.D.I. un module d'aide, capable d'estimer "la gravité de la situation du procédé" et d'évaluer "la situation de fonctionnement" en fonction d'un ensemble d'autres paramètres tels que : un indice de sécurité, un indice de production et le temps disponible pour corriger le défaut, etc. La figure V.10 propose la structure d'un tel module, où l'utilisation de cinq modèles d'évaluation est envisagée :

- le premier modèle propose un indice de sécurité en évaluant, directement sur le procédé, le respect des contraintes de sécurité,
- le second modèle propose un indice de production en évaluant, directement sur le procédé, le respect des contraintes de production,
- le troisième modèle propose un indice sur le temps disponible pour corriger le défaut en évaluant les informations provenant du module de prédiction,
- le quatrième module évalue la situation de fonctionnement du procédé à partir des deux premiers indices,
- enfin le cinquième modèle évalue la gravité engendrée par l'état de fonctionnement du procédé en utilisant les trois premiers indices.

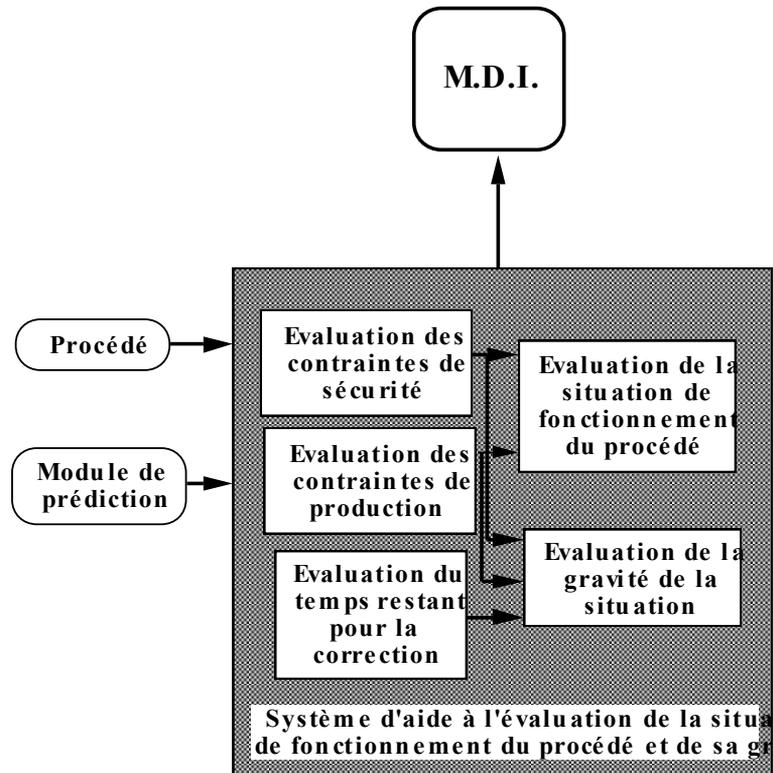


Figure V.10 : Module d'aide à l'évaluation de la situation de fonctionnement du procédé et de sa gravité

II.2.5. Le descriptif "Requête de l'opérateur"

Si les requêtes de l'opérateur concernant les demandes de vues ou d'informations particulières sont déjà prises en compte dans les traitements du M.D.I., il nous paraît important d'intégrer un module traitant les requêtes de type exécution d'une action ou d'un plan d'actions.

Ce module serait chargé de tester les actions de l'opérateur et de fournir un diagnostic sur l'état de fonctionnement du procédé qu'elles engendrent.

D'un côté, ce module, assimilable à un système tolérant aux erreurs humaines (voir le paragraphe IV dans le chapitre II), serait capable de détecter les actions de l'opérateur dont les conséquences affectent les contraintes de sécurité et de production.

D'un autre côté, un tel module serait intéressant d'un point de vue simulation puisque l'opérateur pourrait faire appel à ce module pour tester certains plans d'actions particuliers.

CONCLUSION

Dans la première partie de ce chapitre, nous avons décrit une première validation (essentiellement technique) du fonctionnement de la plate-forme expérimentale.

Nous pouvons en conclure que dans l'état actuel de la plate-forme expérimentale, la phase d'évaluation ergonomique peut maintenant commencer. Elle constitue une première perspective de nos travaux, et devra s'appuyer sur des compétences pluridisciplinaires, dans le cadre d'une collaboration entre le L.A.I.H. et des laboratoires de Psychologie du Travail.

Enfin, dans ce chapitre, nous avons présenté un ensemble de perspectives de recherche concernant l'évolution du M.D.I, et ceci au travers de six points : la tâche en cours de réalisation par l'opérateur, le type de celui-ci, les modes de représentation, la situation de fonctionnement du procédé, la gravité de la situation, et enfin les requêtes de l'opérateur. Pour chacun de ces points, de nouveaux principes ont été envisagés.

CONCLUSION GENERALE

Les travaux présentés dans ce mémoire s'intègrent dans le domaine de l'assistance à l'opérateur humain dans les salles de contrôle de procédés industriels complexes. C'est ainsi que, dans le premier chapitre, nous avons pu positionner les systèmes d'assistance dans l'architecture des systèmes de contrôle et de supervision de procédés industriels. Puis, nous avons discuté de l'évolution du rôle, des caractéristiques et des fonctions de l'opérateur humain en salle de contrôle. Il s'avère que les tâches humaines deviennent hautement cognitives et exigent des connaissances de plus en plus importantes.

Nous avons présenté par la suite un ensemble de solutions visant à faciliter les tâches de l'opérateur. Ces solutions consistent à intégrer dans le système homme-machine des systèmes informatiques d'assistance à l'opérateur humain. Par la suite, nous avons abordé la description d'un certain nombre de problèmes liés à l'élaboration de ces systèmes d'assistance. Ces problèmes concernent par exemple la modélisation du procédé, la prise en compte des tâches à réaliser par l'opérateur et surtout le mode de dialogue entre ces systèmes d'assistance et l'opérateur humain. L'une des solutions à ces problèmes et qui constitue le centre d'intérêt de nos travaux de recherche, vise l'étude, la composition et la mise en œuvre des interactions homme-machines au travers d'interfaces dites évoluées. Notre contribution s'insère dans cette nouvelle classe d'interface homme-machine. Elle vise la réalisation d'une interface "intelligente", basée sur des modèles et des connaissances sur les différents partenaires potentiels, en l'occurrence le procédé, le système d'assistance et l'opérateur humain.

Dans le second chapitre, nous avons parcouru et analysé un ensemble de systèmes de dialogue. Ceux-ci prennent la forme : d'interface flexible, tolérante aux erreurs humaines, experte, adaptative, d'opérateur assistant ou encore d'agent intelligent, d'interface écologique ou hypermédia. La plupart de ces systèmes de dialogue ont un objectif commun, qui est de **s'adapter** soit à l'opérateur humain, soit au procédé, etc. Cependant, on retrouve pour chacun d'eux, un mode d'adaptation particulier. Une conclusion que nous en avons retiré est qu'un système de dialogue idéal doit regrouper les différents concepts décrits dans ce chapitre. Cependant, sa conception va alors hériter des problèmes liés à chaque type de système de dialogue.

Notre contribution réside dans le développement d'une interface "intelligente" appelée Module Décisionnel d'Imagerie (M.D.I.). Ainsi, nous proposons d'utiliser un système expert pour assurer les interactions entre le procédé, l'opérateur humain et

les différents outils d'assistance dont il dispose. Notre approche se propose de résoudre les problèmes d'interaction entre l'homme et la machine en répondant aux trois questions suivantes : (1) Quoi présenter à l'opérateur, (2) Quand le présenter et (3) Comment le présenter. Par la réponse à ces questions, notre approche vise à s'adapter, notamment, aux tâches cognitives de l'opérateur, à son expérience et à ses ressources.

Dans le troisième chapitre, nous avons décrit les travaux qui ont permis d'arriver au M.D.I. Ceux-ci ont pour origine la réalisation d'une interface homme-machine entre opérateur(s) et module(s) d'aide à la décision implanté dans une salle de contrôle. Ces travaux ont permis de mettre en évidence et de développer un modèle de coopération implicite entre homme et système expert d'assistance. Mais, grâce à l'avènement de nouvelles techniques en Intelligence Artificielle, dans le domaine de la conception d'interfaces graphiques et au niveau de l'analyse des tâches et des activités de l'opérateur, notre approche d'interface "intelligente" peut bénéficier des progrès entrepris dans ces domaines afin de parvenir à une coopération homme-machine représentée **explicitement**, à l'aide de règles de production en se basant sur la possibilité de répondre dans chaque situation aux trois questions ergonomiques : QUOI présenter, QUAND et COMMENT présenter l'information. Dans la suite du chapitre, nous avons présenté les objectifs du M.D.I. et sa localisation dans le système de supervision, ainsi que les différents concepts fondamentaux qui le définissent. Enfin, nous avons proposé pour l'intégration du M.D.I. dans le système d'assistance complet, une architecture fonctionnelle intégrant trois modèles : un modèle du procédé, un modèle des tâches assistées de l'opérateur et un modèle de l'opérateur.

Dans le quatrième chapitre, dans un but d'évaluation à court terme des concepts du M.D.I., nous avons décrit la structure et les principaux éléments d'une plate-forme expérimentale intégrant un ensemble de modules d'assistance à l'opérateur et une interface "intelligente" de type M.D.I. Le procédé considéré est un simulateur du circuit de refroidissement d'une centrale nucléaire développé en langage C, utilisant des techniques de la modélisation qualitative. L'ensemble des modules réalisés dans ce cadre a nécessité l'emploi de plusieurs techniques et méthodes très variées, issues de : la modélisation qualitative, l'Apprentissage Symbolique Automatique, la génération de plans, la simulation numérique.

En analysant le système homme-machine ainsi constitué, il nous a été possible de mettre en évidence les tâches de l'opérateur et l'assistance dont il a besoin. C'est ainsi que l'ensemble des modules et modèles ont été mis en œuvre sur une station de travail de type VS3100 pour constituer la plate-forme expérimentale. Celle-ci est destinée à évaluer l'apport pour l'activité de l'opérateur et pour le système homme-machine, d'un système de dialogue de ce type.

Dans le cinquième chapitre, nous avons décrit, dans le cadre d'une première validation essentiellement technique, le fonctionnement de la plate-forme expérimentale. Celle-ci est maintenant prête pour une phase d'évaluation ergonomique.

Cette phase d'évaluation constitue, à nos yeux, un axe de recherche à part entière et s'adresse à une équipe pluridisciplinaire. Ces travaux d'évaluation entreront dans le cadre de collaborations entre le L.A.I.H. et d'autres disciplines comme la Psychologie du Travail.

Enfin, dans ce dernier chapitre, nous avons présenté d'autres perspectives de recherche visant à faire évoluer le Module Décisionnel d'Imagerie. Ces perspectives visent à le faire rapprocher de plus en plus de l'interface "intelligente idéale", si une telle interface existe...

De plus, il serait également intéressant d'étudier l'applicabilité des principes du Module Décisionnel d'Imagerie dans d'autres domaines que le contrôle de procédés industriels. Ainsi, certains descriptifs devraient être abordés différemment. Par exemple, en bureautique ou en utilisation de bases de données, les notions de gravité et de situations de fonctionnement n'auraient bien sûr pas la même portée que pour le contrôle de procédé. Dans d'autres domaines, tels que la télé-opération, la définition et l'implantation de descriptifs supplémentaires seraient éventuellement nécessaires. Des études spécifiques devraient alors être menées.

BIBLIOGRAPHIE

- ABED M., 1990.** Contribution à la modélisation de la tâche par outils de spécification exploitant les mouvements oculaires : Application à la conception et à l'évaluation des interfaces Homme-Machine. Thèse de Doctorat, Université de Valenciennes, France, Septembre 1990.
- ALTY J.L., BERGAN M., 1992.** The design of multimedia interfaces for process control. In Proc. of the 5th IFAC/IFIP/IFORS/IEA symposium on Analysis, Design and evaluation of Man-Machine systems. The Hague, The Netherlands, 9-11 June, 1992.
- ALTY J.L., GUIDA G., 1985.** The use of rule-based system technology for the design of man-machine systems. 2nd IFAC Conference on "Analysis and Evaluation of man-machine systems", Varese, Italy, pp. 46-56, September 1985.
- AMALBERTI R., 1986.** Erreurs humaines dans certains accidents aériens et perspectives d'aides intelligentes. Proceedings International Conference Human Machine Interaction and Artificial Intelligence in Aeronautics and Space, G. Boy (Ed.), CERT, Toulouse, 1986.
- AMALBERTI R., DEBLON F., 1992.** Cognitive modeling of fighter aircraft process control : a step towards an intelligent onboard assistance system. International Journal of Man-Machine Studies, 36 (5), pp. 639-773, 1992.
- AMALBERTI R., GRAU J.Y., VALOT C., 1991.** Assistance in process control : optimal systems versus human-like systems. Third European Conference Cognitive Science Approaches to Process Control, Cardiff, UK, September 2-6, 1991.
- ANDERSON J.R., BOYLE C.F., CORBETT A.T., LEWIS M.W., 1990.** Cognitive modeling and intelligent tutoring. In Artificial Intelligence, 42 , pp. 7-49, 1990.
- ANDOW P.K., 1980.** Real-time Analysis of Process Plant Alarms using a mini computer. Computer in Chemical Engineering. Vol 4, pp. 143-155, 1980.
- ASTRÖM K.J., 1991.** Intelligent Control. Proceeding of ECC'91, France, July 2-5, 1991.
- AUSSENAC N., 1989.** Conception d'une méthodologie et d'un outil d'acquisition de connaissances expertes. Thèse de Doctorat, Université Paul Sabatier de Toulouse, Octobre 1989.

- BAINBRIDGE L., 1978.** The process controller. In *The Analysis of Practical Skills*, Ed. W.T. Singleton, M.T.P., Press Limited, Lancaster, Lancaster, pp. 236-263.
- BARR A., COHEN P.R., FEIGENBAUM E.A., 1982.** *The Handbook of Artificial Intelligence*, vol. 3. W. Kaufmann (Ed.), Los Atlos, 1982.
- BEGORAY J.A., 1990.** An introduction to hypermedia issues, systems and application areas. *International Journal of Man-Machine Studies*, 33, pp. 121-147, 1990.
- BENKIRANE M., 1991.** Contribution à la méthodologie d'extraction de connaissances dans le domaine du diagnostic technique. Thèse de Doctorat, Université de Valenciennes, Mars 1991.
- BENKIRANE M., DUMAS P., HOURIEZ B. ET MILLOT P., 1990.** Aspects de la psychologie cognitive dans l'acquisition des connaissances. Convention I.A 90, Palais des Congrès, Paris, 15-18 Janvier 1990.
- BERRY D.C., BROADBENT D.E., 1986.** Expert Systems and the Man-Machine Interface. *Expert Systems*, vol 3, n°4, pp. 228-231, October 1986.
- BERSINI H., 1989.** La modélisation cognitive dans l'étude des systèmes homme-machines. Thèse de Doctorat, Université libre de Bruxelles, 1989.
- BOY G., 1986.** An Expert System for Fault Diagnosis in Orbital Refueling Operations. 24th Aerospace Science Meeting AIAA. 86, Reno, Nevada, January 1986.
- BOY G., 1988A.** Assistance à l'opérateur : une approche de l'intelligence artificielle. Editions Teknea, Octobre 1988.
- BOY G., 1988B.** Operator assistant systems. In *Cognitive Engineering in complex dynamic worlds*, edited by E. Hollnagel, G. Mancini, D.D. Woods, 1988.
- BOY G., 1990.** Advanced interaction media. International Conference Human Machine Interaction and Artificial Intelligence in Aeronautics and Space, Odysud, Toulouse-Blagnac, 26-28 September, 1990.
- BOY G., TESSIER C., 1985.** Cockpit analysis and assessment by the MESSAGE methodology. 2nd IFAC Conference on "Analysis and Evaluation of man-machine systems", Varese, Italy, pp. 46-56, Septembre 1985.

- BRAJNIK G., GUIDA G., TASSO C., 1988.** IR NLI II : Applying man-machine interaction and artificial intelligence concepts to information retrieval. In Proc. 11th International Conference Res. Development Information retrieval, Grenoble, France, 1988.
- BRAJNIK G., GUIDA G., TASSO C., 1990.** User Modeling in Expert Man-Machine Interface : A Case Study in Intelligent Information Retrieval. IEEE Transactions on Systems, Man and Cybernetics, Vol 20 (1), pp. 167-185, January/February, 1990.
- BROWN P.J.,1987.** Turning ideas into products : The guide system. In Hypertext '87 Papers, pp. 33-40, Chapel Hill, NC, 1987.
- BUSH T., 1945.** As we may think. Atlantic Monthly, 176, pp. 101-108, July, 1945.
- CACCIABUE P.C., DECORTIS F., DROSDOWICZ B., 1990.** Modélisation cognitive de l'activité d'un opérateur controlant un système complexe. Symposium sur la Psychologie du Travail et les Nouvelles Technologies. Université de Liège, 17-18 Mai 1990.
- CALOUD P., 1988.** Raisonement qualitatif : Application à l'aide à la supervision des procédés continus. Thèse de Doctorat, INPG Grenoble, France, Décembre 1988.
- CARD S.K., 1988.** Human Factors and Artificial Intelligence. In P.A. Hancock and M.H. Chignell (Eds). Intelligent Interfaces : Theory, Research and Design, Amsterdam, North Holland, 1988.
- CARD S.K., MORAN T.P. ET NEWELL A.,1983.** The Psychology of Human-Computer Interaction. Hillsdale, N.J. : Lawrence Erlbaum Associates, 1983.
- CHAIB-DRAA B., 1990.** Contribution à la résolution distribuée de problème : une approche basée sur les états intentionnels, Thèse de Doctorat, Université de Valenciennes, France, Juin 1990.
- CHAPOUILLE P., 1972.** La fiabilité. P.U.F., Collection Que Sais-Je, Paris, 1972.
- CHIGNELL M.H, HANCOCK, P.A., 1988.** Intelligent Interface Design. In : Handbook of Human Computer Interaction, M. Helander (ed.), 1988.
- CHIGNELL M.H, HANCOCK, P.A., 1989.** Human factors and Artificial Intelligence. In Intelligent Interfaces : Theory, Research and Design. Chignell & Hancock (Eds), North Holland, pp. 27-48, 1989.

- COEKIN J.A., 1969.** A versatile presentation of parameters for rapid recognition of total state. International Symposium on Man-Machine Systems, IEEE Conference Record 69 C58-MMS, September, 1969.
- DANIELLOU F., 1986.** L'opérateur, la vanne, l'écran, l'ergonomie des salles de contrôle. Edition de l'ANACT, Avril 1986.
- DECOSTER F., 1988.** Ergonomie et productique. Revue des conditions de travail, n° 33, Janvier, 1988.
- DE KEYSER V., 1980.** Etude sur la contribution que pourrait apporter l'ergonomie à la conception des systèmes de commande et d'alerte dans des industries de transformation. Luxembourg, rapport S/79/45, volumes 1 et 2, 1980.
- DE KEYSER V., 1987.** Dossier stratégique : interactions hommes-machines. Rapport CCE-DF XII n°FS2-0147-B (GDF/DSB), Université de l'Etat de Liège, 1987.
- DE KEYSER V., 1988.** Temporal decision making in complex environments. International Workshop on New Technology distributed decision making and responsibility, Bad Hombourg, May, 1988.
- DE LAPPARENT D., 1988.** Système d'aide à l'opérateur : Prédiction et prévention d'accidents en ligne. Journées d'études "Systèmes Experts et conduite de processus en ligne", Société des Electriciens et Electroniciens, Gif sur Yvette, Octobre, 1988.
- DEBERNARD S., VANDERHAEGEN F., MILLOT P., 1992.** An experimental investigation of dynamic allocation of tasks between air traffic controller and A.I. system. 5th IFAC/IFIP/IFORS/IEA Symposium of Analysis, Design and Evaluation of Man-Machine Systems, The Hague, The Netherlands, June 9-11, 1992.
- DE TERSSAC G., CHABAUD C., 1990.** Référentiel opératif commun et fiabilité. In J. Leplat et G. De Terssac. éditeurs, Les Facteurs humains de la fiabilité dans les systèmes complexes, Edition Octarés, Marseille, 1990.
- DUPAS R., 1990.** Apport des méthodes d'Apprentissage Symbolique Automatique pour l'aide à la maintenance industrielle. Thèse de Doctorat, L.A.I.H., Université de Valenciennes, Janvier 1990.
- EDMONDS E.A., 1981.** Adaptive Man-Computer Interfaces. In : M.J. Coombs and J.L. Alty (eds.), Computing skills and the user interface, London, Academic press, 1981.

- EMBREY D.E., 1986.** A systematic human error reduction and prediction approach. In : Proceeding of the international topical meeting on advances in human factor in nuclear power systems, Knoxville Tennessee, USA, 1986.
- ENGELBART D.C., 1963.** A conceptual framework for the augmentation of the man's intellect. In *Vistas in Information Handling*, vol. 1, Spartan Books, London.
- ESIA, 1992.** G2 : Outil de développement de systèmes experts temps réel, documentation technique. ESIA, Réseau Eurisys, St Quentin en Yvelines, 1992.
- FAVERGE J.M., 1972.** L'analyse du Travail. Dans : "Traité de Psychologie Appliquée", tome 3, pp. 5-60, Paris, PUF, 1972.
- FERAY-BEAUMONT S., 1988.** Modèle qualitatif de comportement pour un système d'aide à la supervision des procédés. Thèse de Doctorat, INPG Grenoble, France, Octobre, 1988.
- FININ T., KLEIN D., 1987.** The requirements of active systems. 7th Conference on Expert Systems and Their Applications, Avignon, p. 1199-1207, 1987.
- FISCHER G., 1987.** Making computers more useful and more usable. In G. Salvendy (Ed.), *Cognitive engineering in the design of human-computer interaction and expert systems*, pp. 97-104, Amsterdam, Elsevier, 1987.
- FMECA, 1967.** Design Analysis Procedure for Failure Mode, Effects and Criticality ANALYSIS (FMECA). Recommended Practice ARP 926, SAE Aerospace, September, 1967.
- FORBUS K.D., 1984** Qualitative process theory. "Artificial Intelligence", vol. 24 (1-3), pp. 85-168, 1984.
- FRANCIS J.C. ET LEITH R.R., 1985.** ARTIFACT: A real-time shell for intelligent feedback control. *Research and Development in Expert Systems*, M.A. Bramer(ed.), Cambridge University Press, pp. 151-162, 1985.
- GAMBIEZ F., KOLSKI C., TANG X., MILLOT P., 1991.** Man-Machine Interface for PREDEX, and Expert System shell for process control and supervision. In *Decision Support Systems and Qualitative Reasoning*, M.G. Singh and L. Travé-Massuyès (eds.), Elsevier Science Publishers B.V. (North Holland), pp. 341-346, 1991.

- GAMBIEZ F., TENDJAOUI M., KOLSKI C., MILLOT P., 1990.** Etude et mise en oeuvre d'un module d'interface pour l'outil PREDEX de développement de système expert de filtrage et de gestion d'alarmes pour les procédés continus. Rapport final établi dans le cadre de la convention LAIH/S2O, LAIH, Université de Valenciennes, Juin 1990.
- GANDIBLEUX X., LIBERT G., MILLOT P., 1992.** Multi-Criteria Decision Aid Model for an On-Line Decision Support in supervision : Problematic and Application Overview. Tenth International Conference on Multiple Criteria Decision Making Proc., Vol II, pp. 241-250, Taipei, Taiwan, R.O.C, July 19-24, 1992.
- GASSER L., 1991.** Social Conceptions of knowledge and actions : DAI foundation and open systems semantics. Artificial Intelligence, 1991.
- GOLDSTEIN I. ET BOBROW D., 1987.** A layered approach to software design interactive Programmig Environments. Ed. Barstow, Shorbe Sandewall, pp. 52-63, 1987.
- GUERRA T.M., 1991.** Analyse de données objectifo-subjectives : approche par la théorie des sous-ensembles flous. Thèse de Doctorat, Université de Valenciennes, Juillet 1991.
- GUIDA G., TASSO C., 1982.** NLI : A robust interface for natural language person-machine communication. International Journal of Man-Machine Studies, vol 17, pp. 417-433, 1982.
- GUILLEVIC C. 1990.** L'appropriation cognitive de l'outil : condition de la fiabilité dans les situations de transfert de technologies. Dans : Les Facteurs Humain de la fiabilité dans les Systèmes Complexes. Edition Octares Entreprise, 1990.
- HALASZ F.J., MORAN T.P., TRIGG R.H., 1987.** NoteCards in a nutshell. Proceedings of ACM Conference of human factors in computer systems, pp. 45-52, Toronto, April 1987.
- HANCOCK P.A., CHIGNELL M.H., 1987.** Adaptive Control in Human-Machine systems. In P.A. Hancock (ed.), Human Factors psychology, North holland, 1987.
- HANCOCK P.A., CHIGNELL M.H., 1988.** Mental workload dynamics in adaptive interface design. In IEEE Transactions on Systems, Man and Cybernetics, vol 18 (4), July/August, 1988.

- HANCOCK P.A., CHIGNELL M.M., 1989.** Intelligent interfaces : Theory, Research and Design. North-Holland, 1989
- HEFLEY, W.E., 1990.** Architecture for adaptable Human-Machine Interface. The second International Conference on "Human aspects of advanced manufacturing and hybrid automation", Honolulu, Hawaii, USA, August 12-16, 1990.
- HOC J.M., 1990.** La psychologie cognitive ergonomique face aux nouvelles techniques : quelques thèmes en émergence ou en renouvellement. Symposium sur la psychologie du travail et les nouvelles techniques, Liège, 17-18 Mai 1990.
- HOLLNAGEL E., 1989.** The design of fault tolerant systems : prevention is better than cure. 2nd European Meeting on cognitive science approaches to process control, Siena, Italy, October 24-27, 1989.
- HOUREZ B., BENKIRANE M., DUMAS P., 1988.** Réalisation d'un système expert d'aide au diagnostic des causes de défauts en coulée continue d'acier. Rapport de contrat SOLLAC-IRSID, Juillet, 1988.
- ILOG, 1989A.** AIDA : manuel d'utilisation. Société ILOG, Gentilly, 1989.
- ILOG, 1989B.** SMECI : manuel d'utilisation. Société ILOG, Gentilly, 1989.
- JAKOB F., 1985.** Application des Systèmes Experts au Contrôle Commande de Processus Industriels. Journées ORIA '85, Marseille, IIRIAM, 17-18 Juin, 1985.
- JERRAMS-SMITH J. 1989.** An attempt to incorporate expertise about users into an intelligent interface for Unix. International Journal of Man-Machine Studies, Vol. 31, pp. 269-292, 1989.
- JOHANNSEN G., 1982.** Man-Machine System, Introduction and Background. 1st IFAC Congress on Analysis, Design and Evaluation of Man-Machine Systems, Baden-Baden, September, 1982.
- KNÆUPER A., ROUSE W.B., 1985.** A rule-Based Model of Human problem-Solving Behaviour in dynamic Environment. IEEE Transactions on Systems, Man and Cybernetics, vol. 15 (6), November/December, 1985.
- KODRATOFF Y., 1986.** Leçons d'Apprentissage Symbolique Automatique. Collection Techniques Avancées de l'Informatique, Cepadues Editions, 1986.

- KOLSKI, C., 1989.** Contribution à l'ergonomie de conception des interfaces graphiques Homme-Machine dans les procédés industriels : application au système expert SYNOP. Thèse de Doctorat, Université de Valenciennes, Janvier 1989.
- KOLSKI C., MILLOT P., 1990.** Potential contribution of new hypermedia technologies for man-machine communication in industrial processes. NATO Advanced research Workshop "Structures of communication and intelligent Help for Hypermedia Courseware", Espinho, Portugal, 19-24 April, 1990.
- KOLSKI C., TENDJAOUI M., MILLOT P., 1990.** An "intelligent" interface approach. In Ergonomics of Hybrid Automated Systems II, W. Karwowski and M. Rahimi (Eds.), Elsevier Science Publishers B.V. (North Holland), pp. 587-594, 1990.
- KOLSKI, C., TENDJAOUI, M., MILLOT, P., 1992.** A process method for the design of "intelligent" man-machine interfaces : case study : "the Decisional Module of Imagery". in International Journal of Human factors in Manufacturing, Vol. 2 (2), pp. 155-175, 1992.
- LEJON C.J., 1991.** Rôle de l'homme et du système automatique dans la conduite des processus. R.G.E., N°5, pp. 27-31, Mai, 1991.
- LEPLAT J. , 1985.** Erreur humaine, fiabilité humaine dans le travail. Armand Colin Editeur, Paris, 1985.
- LEPLAT J., CUNY X., 1969.** Recherches sur le contrôle à distance. Le travail mental et automatisation. Etudes de physiologie et de psychologie du travail, N° 6, Luxembourg : CECA, 1969.
- LE STRUGEON E., 1991.** Contribution à la conception et et la réalisation d'une interface évoluée : le Module Décisionnel d'Imagerie. Rapport de DEA, Université de valenciennes, Juillet, 1991
- LE STRUGEON E.,TENDJAOUI M., KOLSKI C., 1992.** Knowledge specification and representation for an "intelligent" interface devoted to process monitoring and supervision. IFAC/IFIP/IMACS International Symposium on Artificial Intelligence in real-time control, Delft, The Netherlands, June 16-18, 1992.
- MALVACHE, 1990.** Système Homme-machine tolérant certaines erreurs humaines. Contrat de recherche en coopération transfrontalière N° 89.1780 avec les universités de Mons et Liège, Mars, 1990.

- MANDIAU R., CHAIB-DRAA B., LIBERT G., MILLOT P., 1991B.** Collaboration Work : A language-Action Perspective. In Proc. of First World Conference on the Fundamentals of Artificial Intelligence, pp. 323-334, Paris, 1-5 July 1991.
- MANDIAU R., KOLSKI C., CHAIB-DRAA B., MILLOT P., 1991C.** A new approach for the cooperation between human(s) and assistance system(s) : a system based on intentional states. World Congress on Expert Systems, Orlando, Florida, December 16-19, 1991.
- MANDIAU R., KOLSKI C., MILLOT P., 1991A.** Perspective of Human-Machine system modelling by logical approach : contribution of technics issued from Distributed Problem Solving. In Designing for Everyone I, Y. Queinnec and F. Daniellou (Eds.), Taylor and Francis, pp. 436-438, 1991.
- MARRAKCHI M., 1986.** Représentation des connaissances pour l'aide au diagnostic industriel. Application au Système Expert S.E.DIAG. Thèse de Docteur Ingénieur, Valenciennes, Mars 1986.
- MASSON M., DE KEYSER V., 1992.** Human errors learned from field study for the specification of an intelligent error prevention system. In S. Kumar (Ed.), Advances in industrial ergonomics and safety IV, pp. 1085-1092, Taylor & Francis, 1992.
- MC ALEESE R., 1990.** The acquisition and representation of domain specific knowledge using NoteCards. NATO Advanced research Workshop "Structures of communication and intelligent Help for Hypermedia Courseware", Espinho, Portugal, 19-24 April, 1990.
- MILLOT P., 1988.** Supervision des procédés automatisés et ergonomie. Editions Hermes, Paris, Décembre, 1988.
- MILLOT P., 1990.** Coopération Homme-Machine : Exemple de la téléopération. Journées du GR Automatique, Strasbourg, 17-19 Octobre, 1990.
- MINCH R.P., 1989.** Research Issues Involving Hypertext in Decision Support Systems. IEEE, pp. 630-638, 1989.
- MIT, 1987.** X-Toolkit library, C Language Interface, X protocol version 11, 1987.
- MOLICH R., NIELSEN J., 1990.** Improving a human-computer dialogue. Computing practice. 33 (3), March, 1990.

- MOORE R.L., HAWKINSON L.B., KNICKBOKER L.G., CHURCHMAN L.M., 1984.** A real-time Expert System for process control. the IEEE 1st conference on Artificial Intelligence Applications, pp. 569-576, 1984.
- MOUSSA F., 1992.** Contribution à la conception ergonomique des interfaces de supervision dans les procédés industriels : Application au système ERGO-CONCEPTOR. Thèse de Doctorat, Université de Valenciennes, Juillet, 1992.
- MOUSSA F., KOLSKI C., MILLOT P., 1992.** A formal methodology for ergonomic design of Man-Machine interfaces. 5th IFAC/IFIP/IFOR/IEA Symposium on Analysis, Design and Evaluation of Man-Machine Systems, The Hague, The Netherlands, June 9-11, 1992.
- MULLER M.J., 1988.** Disentangling application and presentation in distributed computing : architecture and protocole to enable a flexible common user interface. IEEE Transactions on Systems, Man and Cybernetics, 18 (4), July/August, 1988.
- NASSIET D., 1987.** Contribution à la méthodologie de développement des systèmes experts : application au domaine du diagnostic technique. Thèse de Docteur-Ingénieur, Université de Valenciennes, Novembre 1987.
- NELSON T., 1965.** A file structure for the complex, the changing and the indeterminate. Proceedings of the 20th National ACM Conference, pp. 84-100, 1990.
- NIELSEN, J., 1990.** Hypertext and Hypermedia. Academic Press, 1990.
- NII H.P., 1986.** Blackboard systems (part 1). Artificial Intelligence magazine, 7 (2), pp. 38-53, 1986.
- NORCIO A.F., STANLEY J., 1989.** Adaptive Human-Computer Interfaces : A literature survey and perspective. IEEE Transactions on Systems, Man and Cybernetics, 19 (2), March/April, 1989.
- NORMAN, D.A., 1979.** Analysis and Design of Intelligent systems. In F. Klix (ed.), Human and Artificial Intelligence, pp. 37-43, Amsterdam : North Holland, 1979.
- NORMAN, D.A., 1981.** Categorisation of action slips. Psychological Review, 88, pp.1-15, 1981.

- NORMAN D.A., 1986.** Cognitive engineering. In D.A. Norman & S.W Draper (Eds), User centred system design : new perspectives on human computer interaction, Hillsdale, NJ : Erlbaum.
- OCHANINE D.A., 1971.** L'homme dans les systèmes automatisés. Science du comportement, DUNOD, 1971.
- PEJTERSEN A.M., RASMUSSEN J.,** Intelligent systems for information retrieval and decision support in complex work situations. à paraître.
- QUINLAN J.R., 1979.** Discovering rules by induction from large collections of examples. In D. Mitchie (ed.) : Expert systems in microelectronic Age, Edinburgh University Press, 1979.
- QUINLAN J.R., 1983.** Learning efficient classification procedures and their application to chess end games. In Machine Learning : An artificial Intelligence Approach, Volume II, R.S. Michalski, J.G. Carbonnel and T.M. Mitchell (eds), London Academic Press, 1983.
- RASMUSSEN J., 1980.** The human as a system component. In H.T. Smith and T.R.G. Green (Eds) : Human Interaction with Computer, London Academic Press, 1980.
- RASMUSSEN J., 1982.** Human errors, a taxonomy for describing human malfunction in industrial installation. In : Journal of Occupational Accidents, 4, pp. 311-33, 1982.
- RASMUSSEN J., 1983.** Skills, rules and knowledge - signals, signs and symbols, and other distinctions in human performance models. IEEE Transactions on Systems, Man and Cybernetics, 13 (3), pp. 257-266, 1983.
- RASMUSSEN J., 1986.** Information processing and Human-Machine Interaction. An approach to cognitive engineering. North Holland series in system science and engineering, A.P. Sage (Ed.), 1986.
- RASMUSSEN J., VICENTE K.J., 1987.** Cognitive control of human activities and errors : implications for ecological interface design. RISO Technical Report M-2660, RISO National Laboratory, Roskilde, Danemark, September 1987.
- RASMUSSEN J., VICENTE K.J., 1989.** Coping with errors through system design : implication for ecological interface design. in International Journal of Man-Machine Studies, 31, pp. 517-534, 1989.

- ROSCOE S., 19810.** Cockpit workload, Residual attention and Pilot errors. Aviation Psychology, Iowa University Press, 1980.
- ROTH E.M., BENNETT K.B., WOODS D.D., 1988.** Human interaction with an "intelligent" machine. In Cognitive Engineering in complex dynamic worlds, edited by E. Hollnagel, G. Mancini, D.D. Woods, pp. 23- 69, 1988.
- ROUHET J.C., 1988.** Système homme-machine. R7610, Techniques de l'Ingénieur, 4, 1988.
- ROUSE S.H., ROUSE W.B., HAMMER J.M., 1982.** Design and evaluation of an onboard computer based in formation system for aircraft. IEEE Transactions on Systems, Man and Cybernetics, 12 (4), 1982.
- ROUSE W.B. 1973.** Predictive Aids for discret Decision Tasks with Input Uncertainly. IEEE Transactions on Systems, Man and Cybernetics, 3 (5), pp. 19-29, 1973.
- ROUSE W.B. 1980.** Systems Engineering Models of Human-Machine Interaction. New York - Oxford : North Holland Pub, 1980.
- ROUSE W.B., 1988.** Adaptive aiding for Human / Computer control. Human Factors, 30 (4), pp. 431-443, 1988.
- ROUSE W.B., GEDDES N.D., CURRY R.E., (1987, 1988).** An architecture for intelligent interface : Outline of an approach to supporting operators of complex system. Human-Computer interaction, 3 (2), pp. 87-122, 1987-1988.
- ROUSE W.B., MORRIS N.M., 1985.** Conceptual design of an human error tolerant interface for complex engineering systems. 2nd IFAC Conference on Analysis and Evaluation of man-machine systems, Varese, Italy, pp. 46-56, September, 1985.
- ROUSE W.B., ROUSE S.H.,1983.** Analysis and classification of human error. IEEE Transactions on Systems, Man and Cybernetics, 13, pp. 539-549, 1983.
- RUBIN K., JONES P.M., MITCHELL C.M., 1988.** OFMspert : Inference of operator intentions in supervisory control Using Blackboard architecture. IEEE Transactions on Systems, Man and Cybernetics, 18 (4), pp. 618-636, July/August, 1988.
- SACHS P.A., PATERSON A.M., TURNER M.H.M., 1986.** ESCORT : An expert system for complex operation in real-time. Expert System, 3 (1), pp. 22-29, January, 1986.

- SCAPIN D., 1986.** Guide ergonomique de conception des interfaces homme-machine. Rapport technique INRIA, n°77, Le Chesnay, 1986.
- SCHA R.J.H., 1988.** Natural Language Interface Systems. In Handbook of Human-Computer Interaction, M. Helander (ed.), North Holland, 1988.
- SEMINARA J.L., 1979.** Human Factors Methods for nuclear Control-room Design. EPRI NP 1118, Project 501-3, June, 1979.
- SENACH B., 1990.** Evaluation ergonomique des interfaces Homme-machine : une revue de la littérature. Rapport de recherche INRIA, Sophia Antipolis, n° 1180, Mars 1990.
- SHALIN V.L., LEVI K.R., PERSCHBACHER D.L., 1990.** Learning pilot tactics for an intelligent pilot's assistant. International Conference Human Machine Interaction and Artificial Intelligence in Aeronautics and Space, Odysud, Toulouse-Blagnac, 26-28 September, 1990.
- SHALIN V.L., PERSCHBACHER D.L., JAMAR P.G., 1988.** Intent recognition : an emerging technology. International Conference Human Machine Interaction and Artificial Intelligence in Aeronautics and Space, Odysud, Toulouse-Blagnac, 28-30 September, 1988.
- SHANNONE C., WEAVER W., 1949.** The Mathematical Theory of Communication. Urbana, IL : University of Illinois Press, 1949.
- SHERIDAN T.B., 1985.** Forty-Five Years of Man-Machine system : History and trends. 2nd IFAC Congress : Analysis, Design and Evaluation of Machine Systems, Varese, September, 1985.
- SPERANDIO J.C., 1988.** La psychologie du travail mental. Editions Masson, Paris, 1988.
- STASSEN H.G., 1986.** Human Supervisor Modelling : Some New Developments. Post NATO - ASI Conference on Intelligent Aids in Process Environment. Ispra, Italy, November 10-12, 1986.
- TABORIN V., 1989.** Cooperation entre Opérateur et Système d'Aide à la Décision pour la Conduite de Procédés Continus: Application à l'Interface Opérateur Système Expert du Projet ALLIANCE. Thèse de Doctorat, Université de Valenciennes, Mars, 1989.

- TABORIN V., GRZESIAK F., MILLOT P., WILLAEYS D., 1987.** Interface Opérateur- Système Expert pour l'aide à la conduite de procédés continus. 7ème journées Internationales : Les systèmes experts et leurs applications, Avignon, France, 1987.
- TABORIN V., POULAIN T., MILLOT, 1986, 1988, 1990.** ALLIANCE : Système de gestion d'alarmes utilisant les techniques de l'Intelligence Artificielle. Spécifications de l'interface Homme-Système expert. Rapports de contrat MRES-ADI en collaboration avec le CEA, LAG, ITMI, IIRIAM, SHELL RECHERCHE, EDF, SGN, Juillet et Novembre 1986, Mai 1988 et Janvier 1990.
- TENDJAOUI M., 1988.** Etude et réalisation d'un système d'aide à la conception d'interface entre opérateur et système expert. Rapport de DEA, Université de valenciennes, Juin, 1988.
- TENDJAOUI M., KOLSKI C., GAMBIEZ F, MILLOT P., 1991A.** Towards an "intelligent" Interface between Decision aid tool and operator in control room. In Designing for Everyone I, Y. Queinnec and F. Daniellou (Eds.), Taylor and Francis, pp. 652-654, 1991.
- TENDJAOUI, M., KOLSKI, C., MILLOT P., 1990.** Interaction between real-time aid expert system, intelligent interface and human operator. International Symposium Computational Intelligence 90 "Heterogeneous knowledge representation systems", September 24-28, 1990, Milano, Italy.
- TENDJAOUI, M., KOLSKI, C., MILLOT P., 1991B.** An "Intelligent" approach for Ergonomic Design of Man-Machine Interfaces in Process Control. In International Journal of Industrial Ergonomics, 8, pp. 345-361, 1991.
- TENDJAOUI M., KOLSKI C., MILLOT P., 1991C.** Knowledge based interface approach for real time aid expert system. IFAC/IMACS "SAFEPROCESS '91 Symposium, Baden-Baden, Germany, September 10-13, 1991.
- TESSIER C., 1984.** MESSAGE : un outil d'analyse ergonomique de la gestion de vol. Thèse de Docteur Ingénieur, ENSAE, Toulouse, Décembre, 1984.
- V.I. CORPORATION, 1988.** DATAVIEWS : Reference manual, version 6.0. Amherst Research Park, Amherst, Massachusetts 01002, U.S.A, 1988.
- VALOT C., DEBLON F., 1988.** AIDE : Towards human based models for rapid process control. 3rd IFAC congress on Analysis design and evaluation of man-machine systems, Oulu, Finland, June, 1988.

- VAN DER VEER G.C., 1989.** Individual differences and the user interface. *Ergonomics*, 32 (11), pp. 1431-1449, 1989.
- VAN DER VELDT, VAN DEN BOOMGAARD, 1986.** Predictive Information in the Control Room. Vth European Annual Manual : Conference on Human Decision Making and Manual Control, Berlin, June 1986.
- VAN DYKE PARUNAK H., 1989.** Hypermedia Topologies and user navigation. *Hypertext'89 Proceedings*, November, 1989.
- VERCOUSTRE A.M., 1989.** Edition structurée, approche hypertexte, compréhension et complémentarité. rapport de recherche RR-1052, INRIA, 1989.
- VITTET J., 1981.** Eléments pour une conception intégrée des salles de contrôle. Thèse de Docteur Ingénieur, Université Scientifique et Médicale de Grenoble, 1981.
- WAERN Y., 1989.** Cognitive aspects of computer supported tasks. John Wiley & Sons Ltd, 1989.
- WALSH P.A., LIM K.Y., LONG J.B., 1989.** JSD and the design of user interface software. *Ergonomics*, 32 (11), pp. 1483-1498, 1989.
- WANNER J.C., 1987.** Facteur humain et sécurité. Séminaire : Erreurs humaines et automatisation, Ecole Nationale de l'Aviation Civile, 19-20 Mai, 1987.
- WELLS T.L., 1989.** HypertText As Mean For Knowledge Acquisition, *Knowledge Acquisition Special Issue, Sigart Newsletter*, 8, pp. 136-138, April, 1989.
- WILLIGES R.C., WILLIGES B.H., ELKERTON J., 1987.** Software Interface Design. In G. Salvendy (ed.), *Handbook of Human factors*, pp. 1416-1449, Newyork, Wiley, 1987.
- WILSON J.R., CORLETT E.N. (EDS.), 1990.** Evaluation of human works : a practical ergonomics methodology. Taylor & Francis, 1990.
- WOODS D.D., WISE J.A., HANES L.F., 1981.** An evaluation of nuclear power plant safety parameter display systems. 25th Annual meeting Human Factors Society, pp. 110-114, 1981.
- YANKELOVICH N., MEYROWITZ N., VAN DAM A., 1985.** Reading and writing the electronic book. *IEEE Computer*, 1à, pp. 15-30, 1985.

ANNEXES

ANNEXE A : EXEMPLES D'ARBRES DE DECISION GENERES PAR L'OUTIL FIRST-CLASS

INDUCTION DE BASES DE CONNAISSANCES A L'AIDE DE L'ALGORITHME ID3

L'algorithme ID3 est utilisé pour engendrer des procédures de classification, exprimées sous la forme d'arbres de décisions, à partir d'une collection d'exemples appelée ensemble d'apprentissage.

La description des exemples est effectuée à l'aide de n attributs relatifs au problème étudié. Chaque attribut possède un nombre fini de valeurs mutuellement exclusives. Par exemple : l'attribut "SITUATION" possède 4 valeurs qui sont "OPTIMISE", "NORMALE", "ANORMALE" et "CRITIQUE".

De plus, chaque exemple est associé à un concept particulier, appelé encore "classe" de cet exemple. Dans notre cas, il existe trois ensembles de classes : l'ensemble des classes du "QUOI", l'ensemble des classes du "QUAND" et l'ensemble des classes du "COMMENT".

Le principe de cet algorithme repose sur la construction par itérations successives d'un arbre de décisions qui traduit la procédure de classification recherchée et dont les nœuds sont des attributs et les branches sont les valeurs correspondantes de cet attribut. Les feuilles de cet arbre contiennent les différentes classes possibles des exemples. Chaque itération procède au choix d'un attribut qui contribue à étendre l'arbre de décisions, et ce en partant de la racine constituée par le premier attribut choisi jusqu'aux feuilles de l'arbre.

L'attribut choisi à chaque itération est celui qui entraîne un gain d'information maximum. Le gain d'information est mesuré par la différence entre la quantité d'information contenue dans un nœud père avant son expansion et la quantité d'information contenue dans tous les nœuds fils qui découlent de l'attribut choisi. Le lecteur trouvera plus de détails sur cet algorithme dans QUINLAN (1983).

Nous présentons ici deux exemples d'arbres de décisions déduits par l'outil 1stClass (ces arbres ont été laissés au format 1stClass). Le premier exemple représente la base QUOI et le second représente la base QUAND.

Les valeurs utilisés pour chaque descriptif ont la signification suivante :

CLASSE :

1	Débutant
2	Expérimenté
3	Expert

GRAVITE :

0	Le niveau de gravité est égal à 0
1	Le niveau de gravité appartient à l'ensemble [1,3]
2	Le niveau de gravité appartient à l'ensemble]3,4]
3	Le niveau de gravité appartient à l'ensemble]4,7]
4	Le niveau de gravité appartient à l'ensemble]7,10]
5	Le niveau de gravité appartient à l'ensemble]10,12]
6	Le niveau de gravité est supérieur à 12

REQUETE :

NO_REQUEST	L'opérateur ne fait aucune requête
SUPERVISION	L'opérateur demande la vue de supervision
COURBES	L'opérateur demande les vues courbes
JUST_SURF	L'opérateur demande des justifications de surface
JUST_PROF	L'opérateur demande des justifications profondes

SITUATION :

OPTIMISEE	La situation de fonctionnement du procédé est optimisée
NORMALE	La situation de fonctionnement du procédé est normale
ANORMALE	La situation de fonctionnement du procédé est anormale
CRITIQUE	La situation de fonctionnement du procédé est critique
IRRECUPER	La situation de fonctionnement du procédé est irrécupérable

TACHE_OP (tâche de l'opérateur) :

TACHE_DEMARR	L'opérateur exécute une tâche de démarrage
---------------------	--

SURVE_OPTIM	L'opérateur surveille et optimise le fonctionnement du procédé
RESOLUTION	L'opérateur exécute une tâche de résolution de problème
TACHE_ARRET	L'opérateur exécute une tâche d'arrêt de l'installation

QUOI :

RIEN	L'inférence du QUOI ne donne aucun résultat
NON_DISPON	L'information n'est pas disponible
VARIABLES	Il faut présenter les vues courbes
VUE_ETOILE	Il faut afficher la vue étoile de supervision
JUST_SUR_FL	Il faut afficher les justifications de surface sous forme d'un graphe de fluence
JUS_SUR_FLT	Il faut afficher les justifications de surface sous forme d'un graphe de fluence accompagnées d'une explication textuelle
JUST_PRO_FL	Il faut afficher les justifications profondes sous forme d'un graphe de fluence
JUS_PRO_FLT	Il faut afficher les justifications profondes sous forme d'un graphe de fluence accompagné d'une explication textuelle
PLAN_ACTION	Il faut afficher les plans d'actions
GAME_OVER	Il faut aviser l'opérateur du déclenchement de la procédure d'arrêt d'urgence

QUAND

JAMAIS	L'information déduite par le QUOI ne sera jamais affichée
RETOUR	Refaire l'inférence pour le QUOI
MAINTENANT	L'information déduite par le QUOI doit être affichée maintenant
A_MI_ECHEAN	L'information déduite par le QUOI devra être affichée à mi-échéance
QUAND_REQUE	L'information déduite par le QUOI devra être affichée quand l'opérateur en fera la requête

COMMENT

PAS_DE_CHGT	Pas de changement
FIN	Arrêt de l'installation
VUE_ETOILE	Vue de supervision
VUE_COURBE	Fenêtre de sélection des variables afin d'alimenter les vues courbes

FLU_SUR	Le graphe de fluence des justifications de surface
FLU_TXT_SUR	Le graphe de fluence et le texte des justifications de surface
FLU_DET	Le graphe de fluence des justifications profondes
FLU_TXT_DET	Le graphe de fluence et le texte des justifications profondes
PL_ACTION	les plans d'actions accompagnant les alarmes
PL_ACT_DET	les plans d'actions détaillés accompagnant les alarmes

BASE "QUOI" 9:54 AM 1/20/1992

---- début ----
1: REQUETE??
2: NO_REQUEST:SITUATION??
3: OPTIMISEE:-----RIEN
4: NORMALE:-----PLAN_ACTION
5: ANORMALE:CLASSE??
6: 1:GRAVITE??
7: 0:-----RIEN
8: 1:-----RIEN
9: 2:-----JUST_SUR_FL
10: 3:-----JUS_SUR_FLT
11: 4:-----RIEN
12: 5:-----RIEN
13: 6:-----RIEN
14: 2:GRAVITE??
15: 0:-----RIEN
16: 1:-----RIEN
17: 2:-----PLAN_ACTION
18: 3:-----JUST_SUR_FL
19: 4:-----RIEN
20: 5:-----RIEN
21: 6:-----RIEN
22: 3:-----PLAN_ACTION
23: CRITIQUE:CLASSE??
24: 1:GRAVITE??
25: 0:-----RIEN
26: 1:-----RIEN
27: 2:-----RIEN
28: 3:-----RIEN
29: 4:-----JUST_PRO_FL
30: 5:-----JUS_PRO_FLT
31: 6:-----RIEN
32: 2:GRAVITE??
33: 0:-----RIEN
34: 1:-----RIEN
35: 2:-----RIEN
36: 3:-----RIEN
37: 4:-----JUS_SUR_FLT
38: 5:-----JUST_PRO_FL
39: 6:-----RIEN
40: 3:GRAVITE??
41: 0:-----RIEN
42: 1:-----RIEN
43: 2:-----RIEN
44: 3:-----RIEN
45: 4:-----JUST_SUR_FL
46: 5:-----JUST_PRO_FL
47: 6:-----RIEN
48: IRRECUPER:-----GAME_OVER
49: COURBES:-----VARIABLES

50: JUST_SURF:CLASSE??
51: 1:GRAVITE??
52: 0:-----RIEN
53: 1:-----JUST_SUR_FL
54: 2:-----JUS_SUR_FLT
55: 3:-----JUS_SUR_FLT
56: 4:-----JUS_SUR_FLT
57: 5:-----JUS_SUR_FLT
58: 6:-----RIEN
59: 2:GRAVITE??
60: 0:-----RIEN
61: 1:-----NON_DISPON
62: 2:-----JUST_SUR_FL
63: 3:-----JUST_SUR_FL
64: 4:-----JUS_SUR_FLT
65: 5:-----JUS_SUR_FLT
66: 6:-----RIEN
67: 3:-----NON_DISPON
68: JUST_PROF:GRAVITE??
69: 0:-----RIEN
70: 1:-----NON_DISPON
71: 2:-----NON_DISPON
72: 3:CLASSE??
73: 1:-----JUST_PRO_FL
74: 2:-----NON_DISPON
75: 3:-----JUST_PRO_FL
76: 4:CLASSE??
77: 1:-----JUS_PRO_FLT
78: 2:-----JUST_PRO_FL
79: 3:-----JUST_PRO_FL
80: 5:CLASSE??
81: 1:-----JUS_PRO_FLT
82: 2:-----JUS_PRO_FLT
83: 3:-----JUST_PRO_FL
84: 6:-----RIEN
85: SUPERVISION:-----VUE_ETOILE

BASE "QUAND" 9:58 AM 1/20/1992

---- début ----
1: REQUETE??
2: NO_REQUEST:QUOI??
3: NON_DISPON:-----JAMAIS
4: RIEN:-----JAMAIS
5: VARIABLES:-----JAMAIS
6: JUST_SUR_FL:-----MAINTENANT
7: JUST_PRO_FL:-----MAINTENANT
8: JUS_SUR_FLT:CLASSE??
9: 1:-----A_MI_ECHEAN
10: 2:-----QUAND_REQUE
11: 3:-----JAMAIS
12: JUS_PRO_FLT:CLASSE??
13: 1:-----A_MI_ECHEAN
14: 2:-----QUAND_REQUE
15: 3:-----JAMAIS
16: PLAN_ACTION:GRAVITE??
17: 0:-----JAMAIS
18: 1:CLASSE??
19: 1:-----MAINTENANT
20: 2:-----A_MI_ECHEAN
21: 3:-----A_MI_ECHEAN
22: 2:CLASSE??
23: 1:-----MAINTENANT
24: 2:-----MAINTENANT
25: 3:-----A_MI_ECHEAN
26: 3:-----MAINTENANT
27: 4:-----JAMAIS
28: 5:-----JAMAIS
29: 6:-----JAMAIS
30: GAME_OVER:-----MAINTENANT
31: VUE_ETOILE:-----JAMAIS
32: COURBES:-----MAINTENANT
33: JUST_SURF:-----MAINTENANT
34: JUST_PROF:-----MAINTENANT
35: SUPERVISION:-----MAINTENANT

ANNEXE B : EXEMPLES DE PLANS D' ACTIONS

Dans un premier temps, les variables sont classées par ordre d'importance comme dans le tableau B.1.

Tableau B.1 : Classement des 3 variables principales

Variable	Symbole	N°
Température de sodium à l'entrée du générateur de vapeur	TSEGV	4
Température de sodium à la sortie du générateur de vapeur	TSSGV	7
Température de l'eau à la sortie du générateur de vapeur	TESGV	8

Les plans d'actions sont créés selon le principe suivant :

- Il faut d'abord corriger la variable la plus importante, selon l'ordre donné dans le tableau B.1.
- Il s'agit ensuite de corriger éventuellement la deuxième variable puis la troisième.
- Il faut vérifier si la conséquence des corrections précédentes n'affecte pas les symptômes observés, auquel cas il faut majorer la correction correspondante.
- Il faut vérifier si la conséquence des corrections précédentes n'affecte pas d'autres variables, auquel cas il faut préparer une action de correction.
- Etc.

Il est possible selon cette démarche d'obtenir un ensemble de plan d'actions, dont trois exemples sont présentés sur la figure B.1.

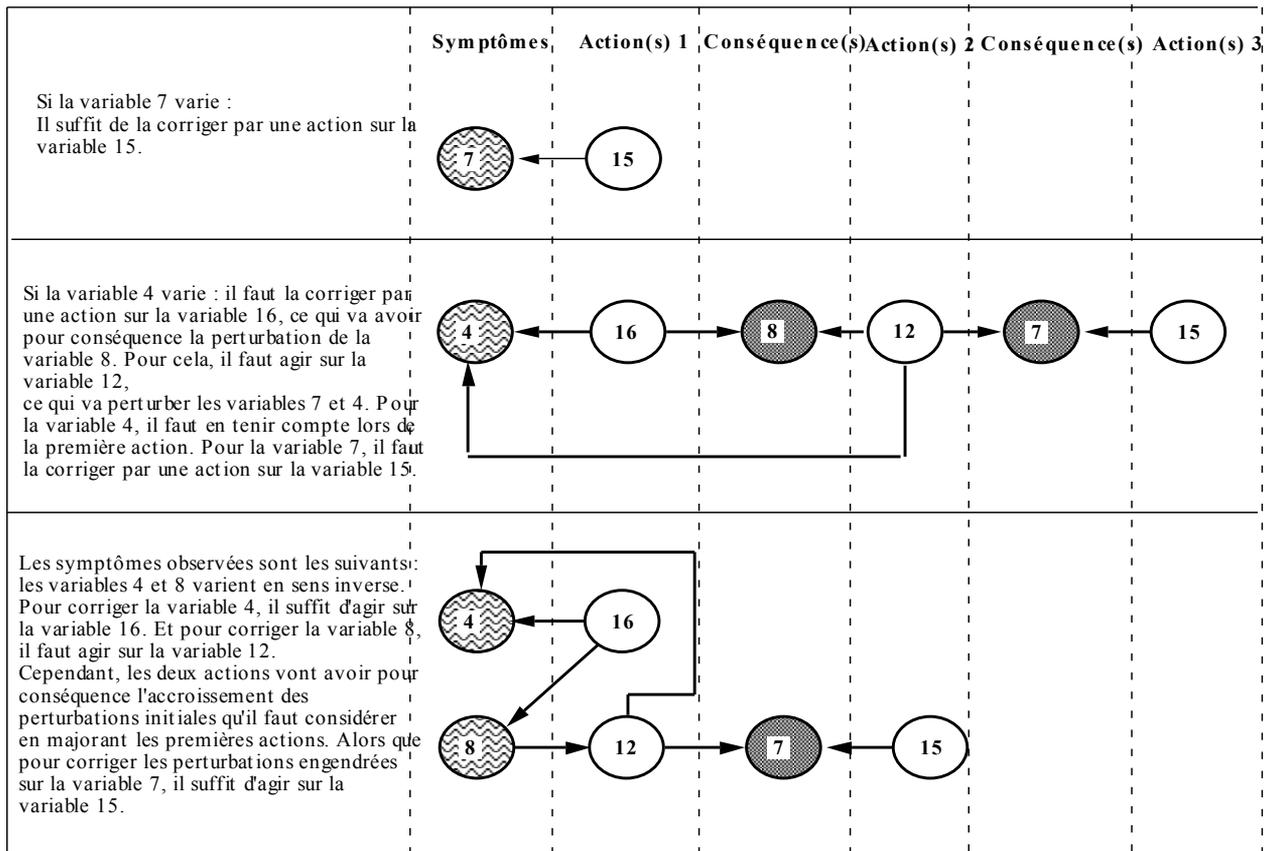


Figure B.1 : Trois exemples de plans d'actions

Rappelons que le procédé étant simple, l'ensemble des plans d'actions possibles a pu être recensé a priori par les concepteurs de la plate-forme expérimentale. Chaque plan d'actions a donc été établi cas par cas et placé dans une base de données accessible par le module de génération de plans d'actions (chapitre IV, paragraphe III.3). Par exemple, si la variable 7 est perturbée, alors le générateur de plans d'actions sélectionne automatiquement le premier plan visible sur la figure B.1, car seul ce plan a été prévu dans ce cas. En effet, à chaque type de défaut est toujours associé à un seul type de plan d'actions.

ANNEXE C : EXEMPLES DE REGLES DE PRODUCTION

Voici quelques règles de productions tirées des bases de connaissance du M.D.I.

5

SI :

GRAVITE = 2

CLASSE = 1

SITUATION = ANORMALE

REQUETE = NO_REQUEST

ALORS :

QUOI = JUST_SUR_FL

13

SI :

GRAVITE = 3

CLASSE = 2

SITUATION = ANORMALE

REQUETE = NO_REQUEST

ALORS :

QUOI = JUST_SUR_FL

30

SI :

GRAVITE = 5

CLASSE = 2

SITUATION = CRITIQUE

REQUETE = NO_REQUEST

ALORS :

QUOI = JUST_PRO_FL

36

SI :

GRAVITE = 4

CLASSE = 3

SITUATION = CRITIQUE

REQUETE = NO_REQUEST

ALORS :

37

SI :

GRAVITE = 5

CLASSE = 3

SITUATION = CRITIQUE

REQUETE = NO_REQUEST

ALORS :

QUOI = JUST_PRO_FL

64

SI :

CLASSE = 3

GRAVITE = 4

REQUETE = JUST_PROF

ALORS :

QUOI = JUST_PRO_FL

65
SI :
 CLASSE = 1
 GRAVITE = 5
 REQUETE = JUST_PROF
ALORS :
 QUOI = JUS_PRO_FLT

66
SI :
 CLASSE = 2
 GRAVITE = 5
 REQUETE = JUST_PROF
ALORS :
 QUOI = JUS_PRO_FLT

69
SI :
 REQUETE = SUPERVISION
ALORS :
 QUOI = VUE_ETOILE

82
SI :
 CLASSE = 1
 GRAVITE = 1
 QUOI = PLAN_ACTION
 REQUETE = NO_REQUEST
ALORS :
 QUAND = MAINTENANT

83
SI :
 CLASSE = 2
 GRAVITE = 1
 QUOI = PLAN_ACTION
 REQUETE = NO_REQUEST
ALORS :
 QUAND = A_MI_ECHEAN

84
SI :
 CLASSE = 3
 GRAVITE = 1
 QUOI = PLAN_ACTION
 REQUETE = NO_REQUEST
ALORS :
 QUAND = A_MI_ECHEAN

ANNEXE D : EXEMPLES DE COPIES D'ECRAN

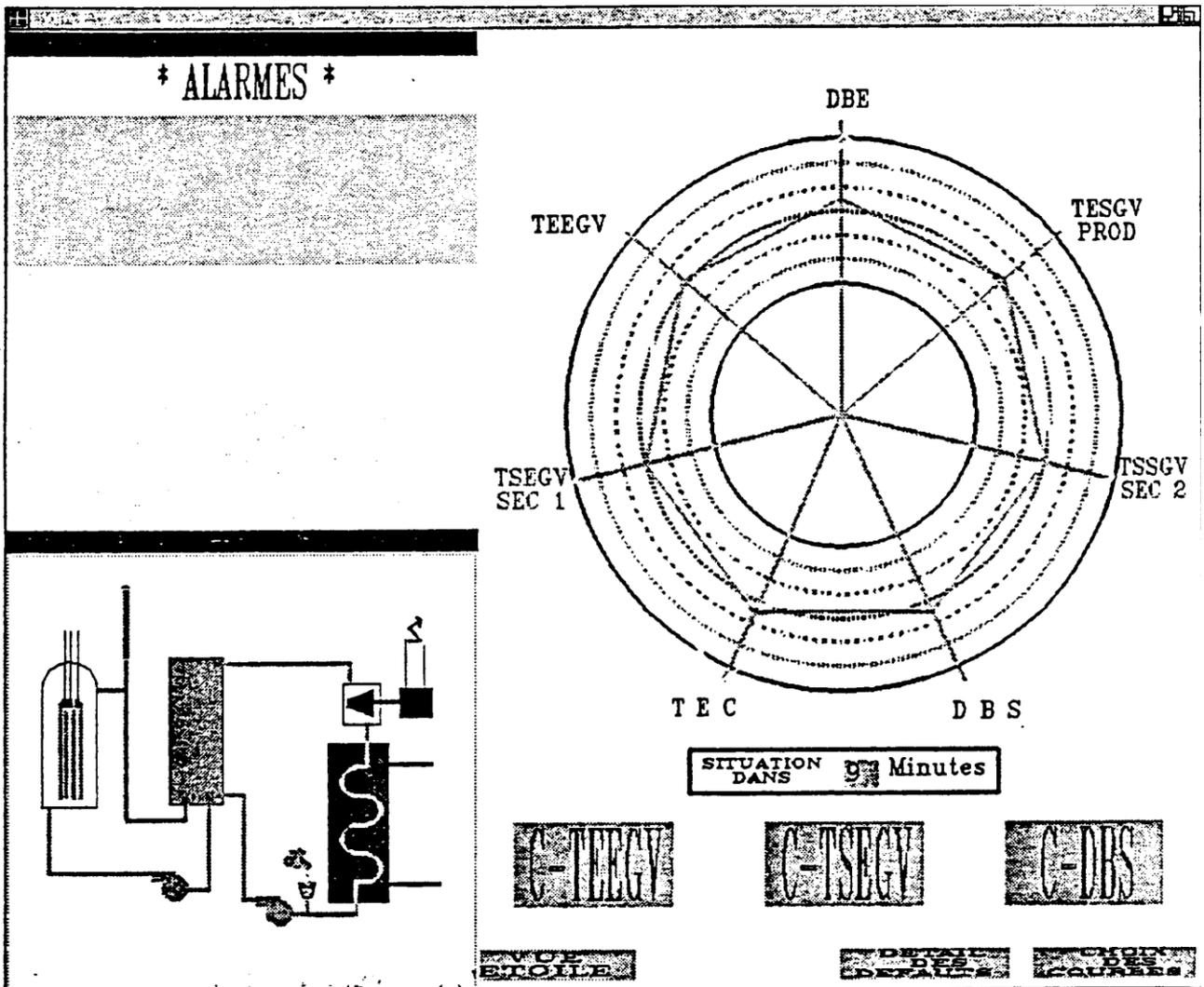


Figure D.1 : Exemple de copie d'écran représentant une vue de supervision et des justifications de surface

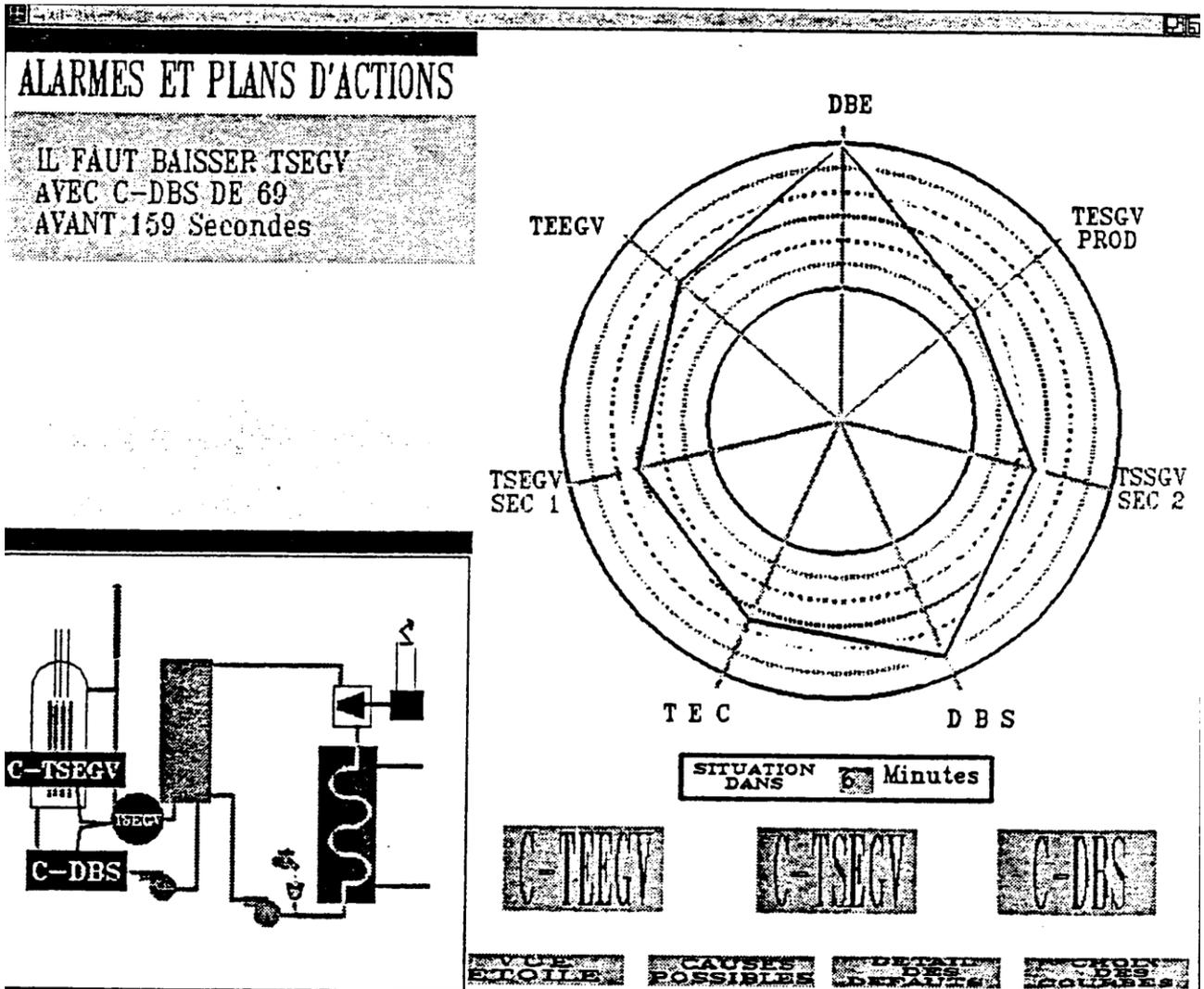


Figure D.2 : Exemple de copie d'écran représentant une vue de supervision, des justifications de surface et un plan d'action

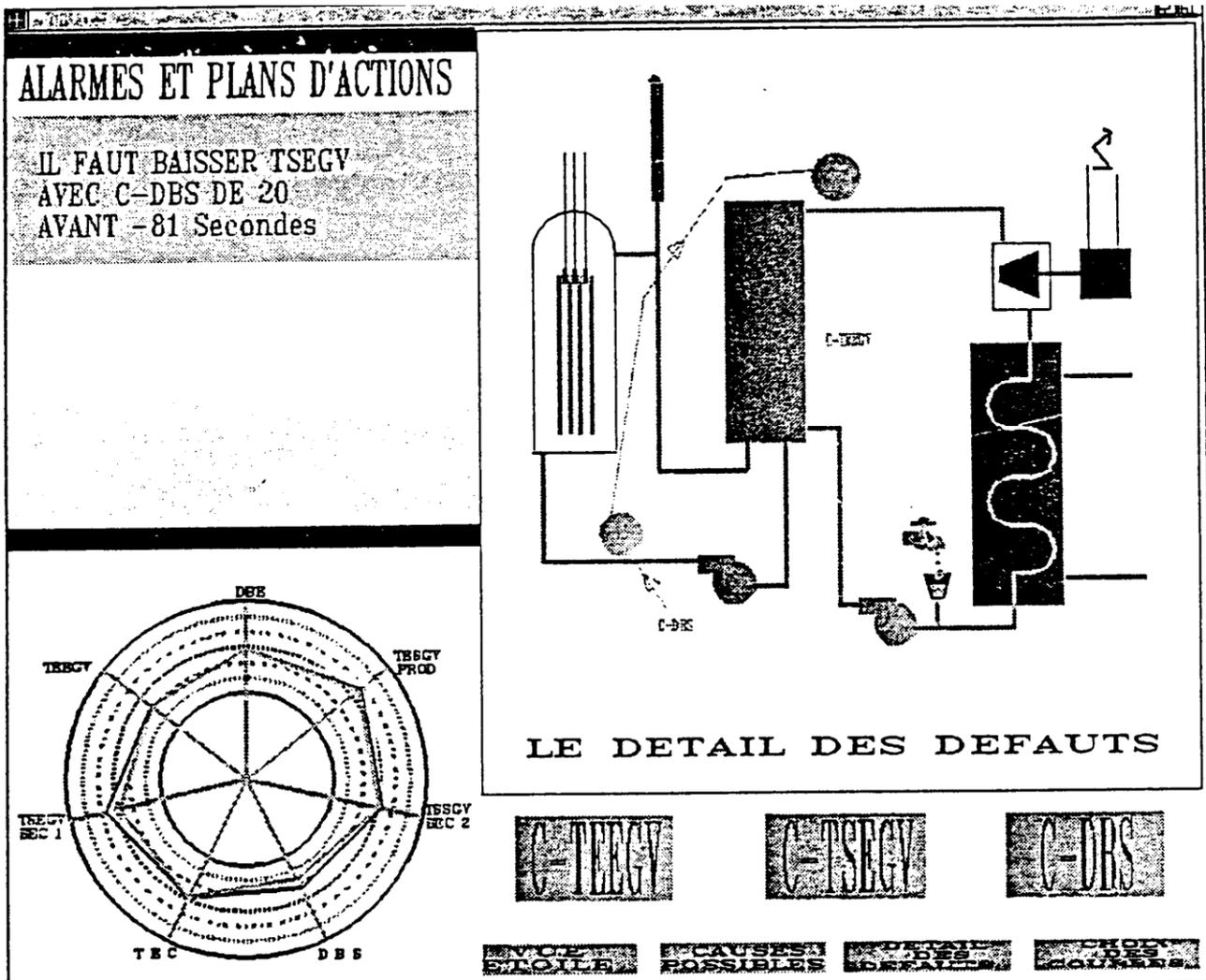


Figure D.3 : Exemple de copie d'écran représentant des justifications profondes, une vue de supervision et un plan d'action

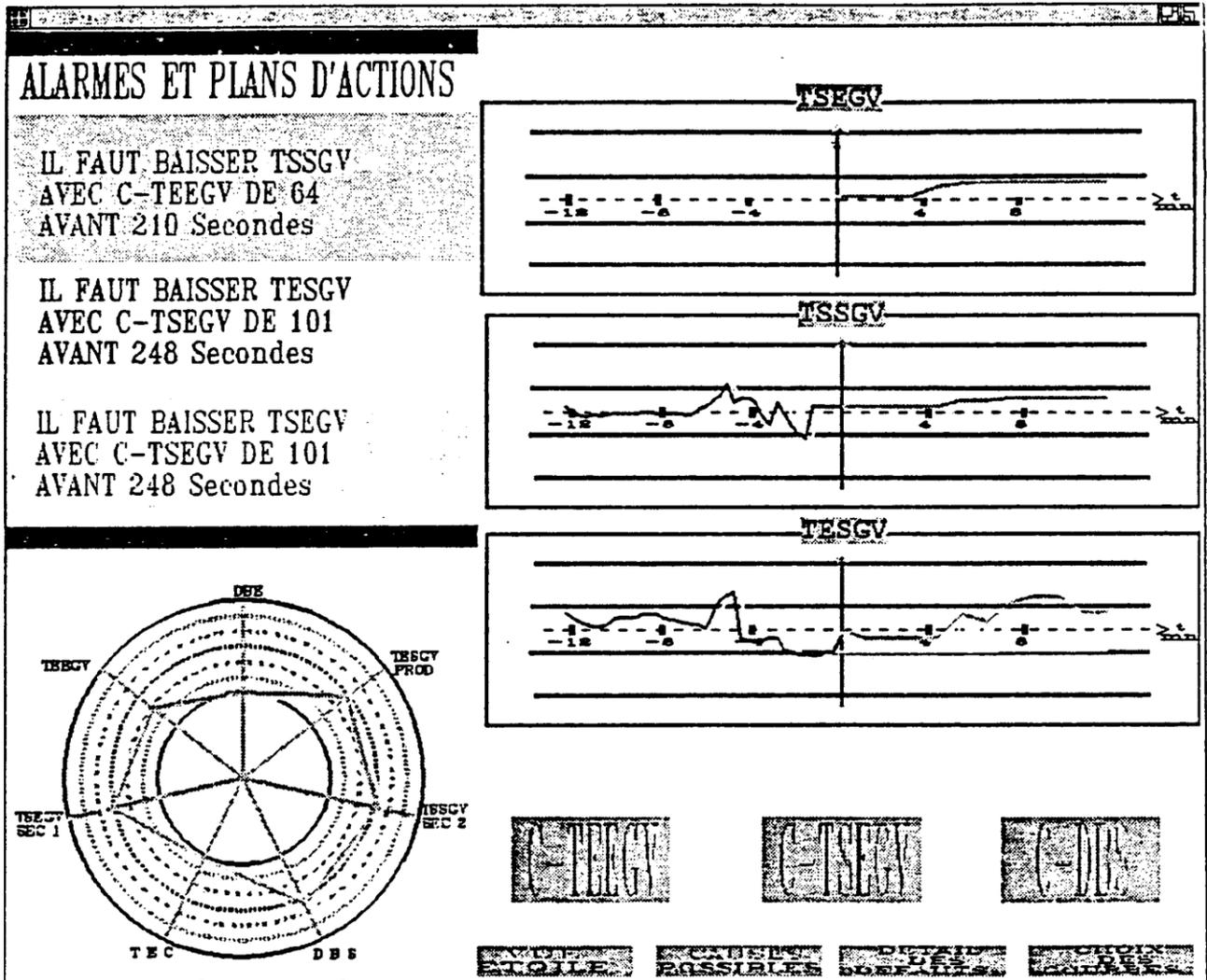


Figure D.4 : Exemple de copie d'écran représentant des courbes, une vue de supervision et trois plans d'action hiérarchisés

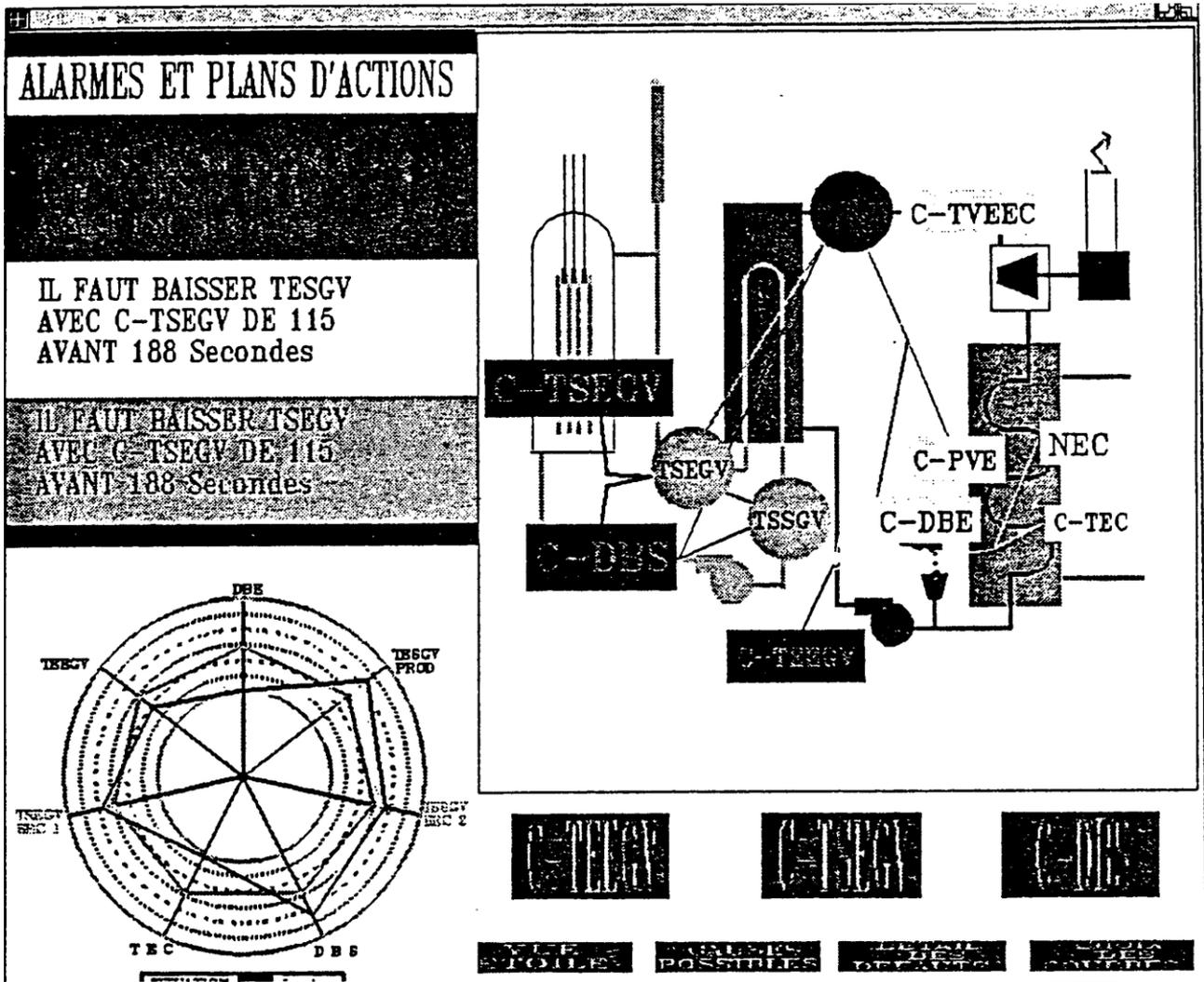


Figure D.5 : Exemple de copie d'écran représentant des justifications de surface, une vue de supervision et trois plans d'action hiérarchisés

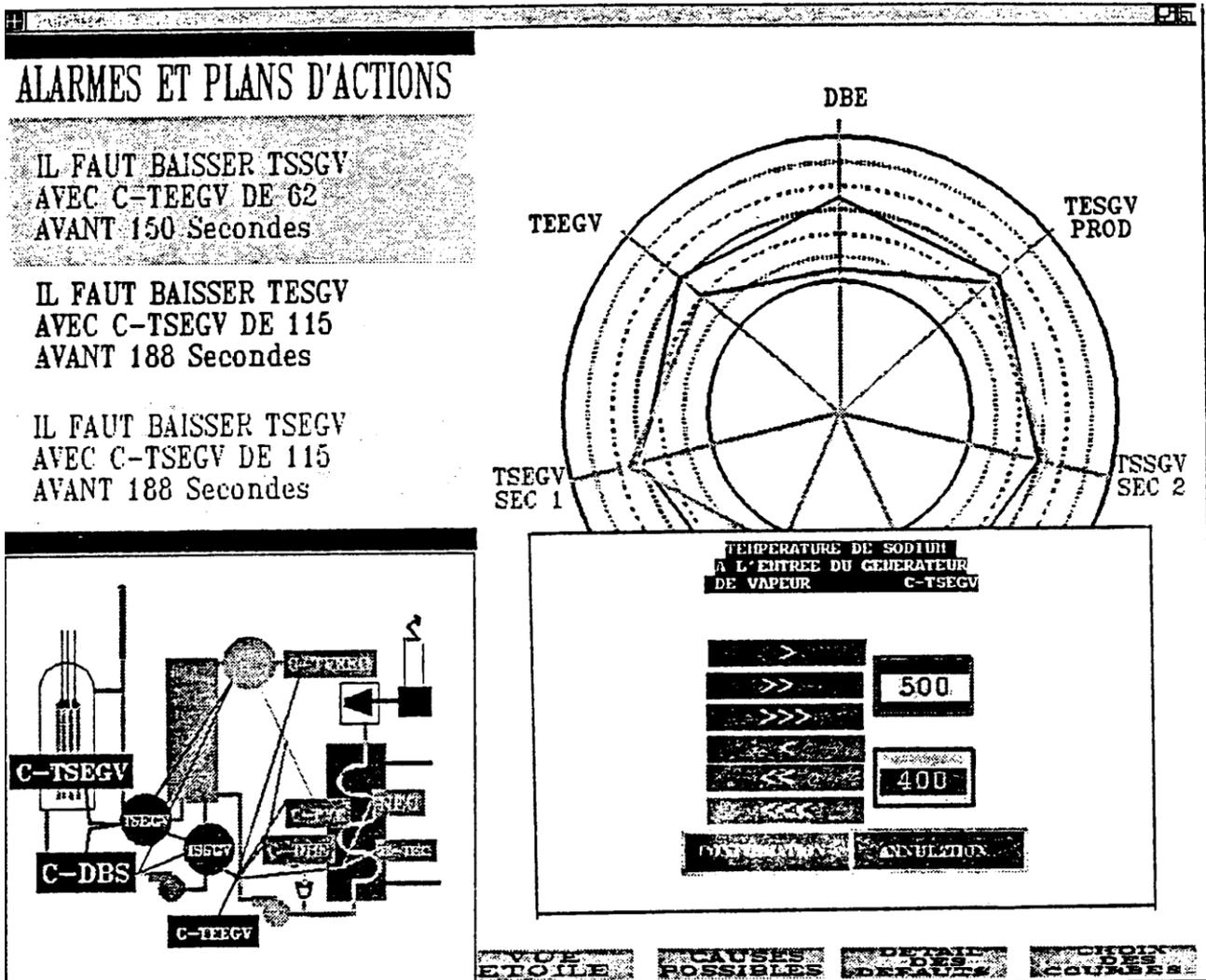


Figure D.6 : Exemple de copie d'écran la mise en œuvre d'une commande sur une variable

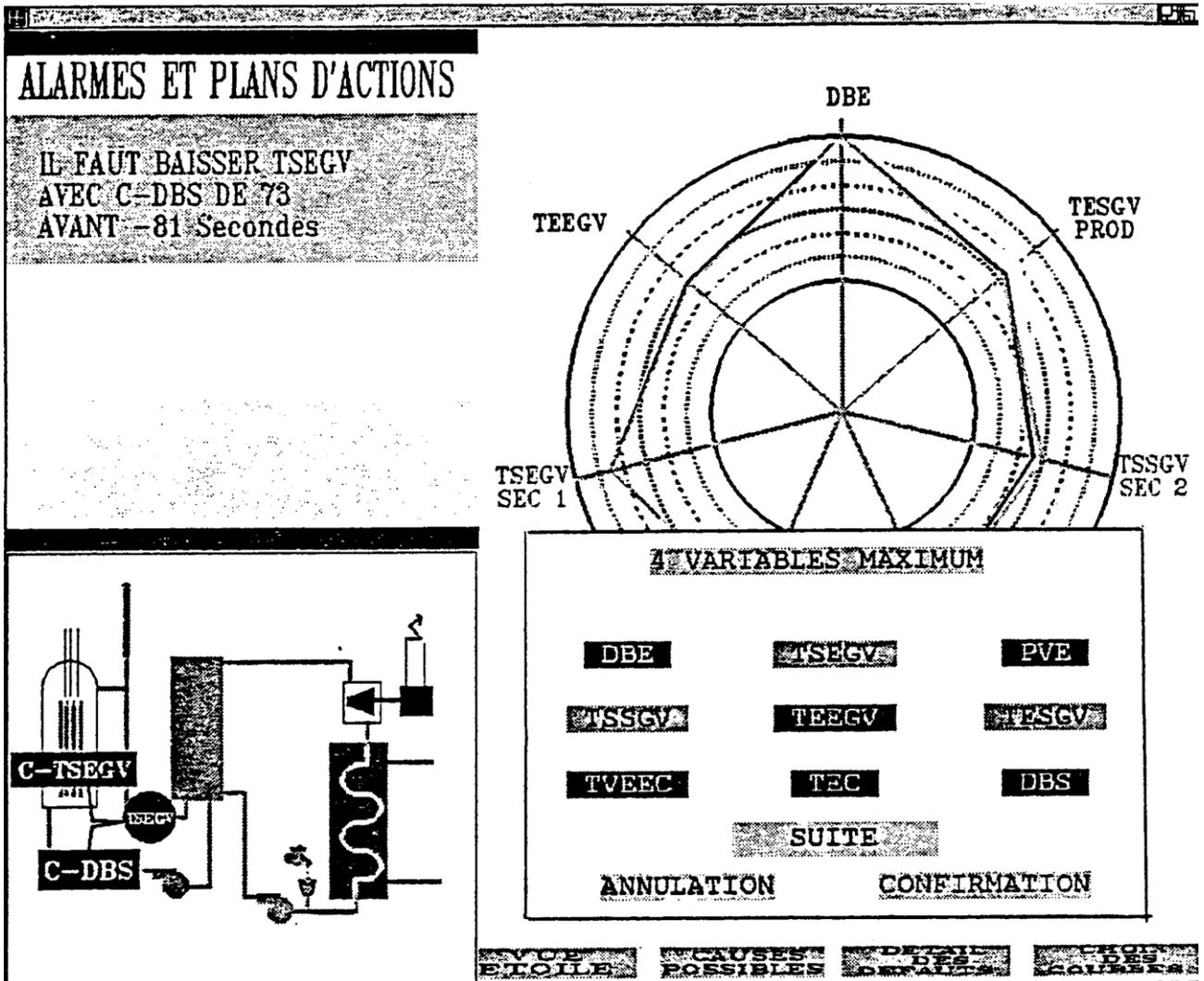


Figure D.7 : Exemple de copie d'écran représentant une liste hiérarchisée de variables à présenter sous forme de courbe