



HAL
open science

Analyse des dysfonctionnements des systèmes complexes en amont de la conception des IHM : apports, difficultés, et étude de cas

Faouzi Moussa, Christophe Kolski, Meriem Riahi

► To cite this version:

Faouzi Moussa, Christophe Kolski, Meriem Riahi. Analyse des dysfonctionnements des systèmes complexes en amont de la conception des IHM : apports, difficultés, et étude de cas. *Revue d'Interaction Homme Machine*, 2006, 7 (2), pp.79-111. hal-03390694

HAL Id: hal-03390694

<https://uphf.hal.science/hal-03390694v1>

Submitted on 4 Jul 2022

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.



Distributed under a Creative Commons Attribution 4.0 International License

Analyse des dysfonctionnements des systèmes complexes en amont de la conception des IHM : apports, difficultés, et étude de cas

Dysfunction analysis of complex systems with a view of HCI design: contributions, difficulties, and case study

Faouzi MOUSSA (1), Christophe KOLSKI (2), Meriem RIAHI (1)

(1) LIPP-Dép. Sciences de l'Informatique- Faculté des Sciences de Tunis, Tunisie
faouzi@gnet.tn

(2) LAMIH, Université de Valenciennes et du Hainaut-Cambrésis, France
Christophe.Kolski@univ-valenciennes.fr

Résumé. Dans de nombreux systèmes industriels complexes, l'analyse et la modélisation des dysfonctionnements forment une étape indispensable venant en amont des projets de conception des systèmes de supervision. Elle constitue potentiellement une source précieuse pour s'orienter vers la déduction des besoins informationnels des utilisateurs. Toutefois, la transition est considérée depuis longtemps comme difficile en IHM et a d'ailleurs été relativement peu étudiée sur des cas complexes. Cet article vise à apporter une première approche de solution. Une démarche globale d'analyse est proposée ; celle-ci considère explicitement les situations de dysfonctionnement. Elle est illustrée par un cas d'étude représentatif d'une situation industrielle.

Mots-clés. dysfonctionnements, besoins informationnels des utilisateurs, IHM, systèmes industriels.

Abstract. In industrial processes, the dysfunction modelling step is a must. It is performed in the beginning of the design of process control systems. Information carried out in this step offers a great opportunity to deduce the users' informational needs. However, going from the dysfunction modelling step to the user's informational needs deduction step still seems to be difficult and is often miss studied on complex cases. In this paper, we present a way of solution. A global analysis approach is proposed; it considers explicitly dysfunction modelling. An illustration is given too; it is representative of an industrial situation.

Key words. dysfunctioning, users' informational needs, HCI, industrial systems.

1 Introduction

De nombreuses démarches méthodologiques globales sont proposées dans la littérature, pour la conception et l'évaluation des systèmes interactifs (Lim et

Long, 1994 ; Bodart *et al.*, 1995 ; Palanque et Bastide, 1996). D'autres sont proposées dans le domaine spécifique des systèmes industriels complexes (Gilmore *et al.*, 1989 ; Millot et Debernard 1993 ; Abed, 2001 ; Moussa *et al.*, 2000 ; Lepreux *et al.*, 2003 ; Riahi, 2004), ces derniers constituant le cadre applicatif de cet article.

Dans ces systèmes industriels complexes caractérisés par des risques importants, qu'ils soient matériels, écologiques et/ou humains, inhérents aux dysfonctionnements, par la diversité des situations (normales, anormales), par le nombre de variables à considérer (souvent des centaines, parfois des milliers), par une forte dynamique liée aux interactions et liens d'influence entre variables, l'opérateur humain demeure indispensable en salle de contrôle. D'une part, l'intelligence humaine fait que l'opérateur humain est à même d'assurer une supervision intelligente sans cesse enrichie par l'expérience. D'autre part, la présence humaine s'avère nécessaire pour réagir face à des situations de dysfonctionnement imprévues (Kolski, 1997 ; Moray, 1997 ; Rasmussen, 1986 ; Reason, 1990). Partant de ce constat, les méthodes de conception générales en IHM s'avèrent insuffisantes, et en tout cas, elles s'intéressent rarement aux situations de dysfonctionnement.

Dès lors, la conception des IHM peut en particulier être utilement axée sur deux rôles prépondérants : (1) offrir à l'opérateur le moyen de superviser l'état de fonctionnement des principaux constituants du système, (2) préparer les moyens qui permettent (voire assistent) son intervention en cas de dysfonctionnement (visible, prévisible...). Le recours indispensable à la modélisation du procédé, particulièrement dans les applications industrielles complexes, ne semble pas être complètement assimilé et systématique, même si des recherches actives sont menées dans ce domaine.¹

Dans les salles de contrôle des systèmes critiques, l'utilisation de l'interface graphique (exclusivement ou partiellement) pour constituer le support de l'IHM s'est avérée d'un intérêt capital. En effet, il est bien connu désormais que les interfaces graphiques permettent des affichages diversifiés et évolutifs dans le temps (Daniellou, 1988 ; Lejon, 1991, Lambert, 1999, etc.). Bien conçues, elles doivent permettre aussi et surtout une interaction homme-machine facile et performante. Ainsi, la conception de celle-ci est devenue centrée sur l'utilisateur (appelé aussi opérateur humain (OH) dans l'article), dans la mesure où l'efficacité de l'IHM est très liée à la capacité de l'utilisateur à percevoir, interpréter les informations, raisonner sur celles-ci et à agir à travers l'interface pour piloter le système (De Keyser, 1988 ; Millot, 1988). Cette conception centrée autour de l'utilisateur (Norman, 1986) se doit d'intégrer les enseignements et les résultats de plusieurs disciplines :

- l'automatique et ses apports, entre autres, dans la modélisation du procédé, la surveillance, la coopération entre l'homme et la machine, etc. (Millot, 1988) ;

¹ A ce sujet, depuis les années 90, nous constatons un engouement progressif autour des approches de conception à base de modèles (Model Based Design/Development, (Szekeley, 1996)), s'accroissant suite à la proposition de l'approche MDA (Model Driven Architecture) par l'OMG et à l'Ingénierie Dirigée par les Modèles (IDM) (Favre et al., 2006). Cette classe d'approches implique forcément le recours aux modèles (procédé, tâche, interaction, etc.) dans le processus de conception. La méthode proposée dans l'article se situe dans ce courant, en accordant une large part à la modélisation.

- la psychologie cognitive permettant, entre autres, la modélisation de l'opérateur humain, l'analyse et la modélisation de la « tâche » et de « l'activité » de l'opérateur humain (Scapin et Bastien, 2001 ; Diaper et Stanton, 2004) ainsi que la détermination des besoins en information de celui-ci ;
- les langages de spécification et d'analyse permettant, selon la rigueur de leurs formalismes, de modéliser les tâches opérateurs, les objets de l'interface et leurs comportements, etc. (Jambon *et al.*, 2001). Les possibilités de vérification des propriétés des modèles établis sont ici des atouts appréciables (Cf. par exemple Palanque *et al.*, 1995 ; Guittet *et al.*, 1997) ;
- l'ergonomie cognitive, venant principalement en complément à la psychologie cognitive, pour la résolution des problèmes autour de la présentation de l'information à l'écran, la prise en charge des problèmes sur le plan du confort opérateur, etc. (Stewart et Travis, 2003) ;
- l'informatique dans la dimension la plus large du Génie logiciel et dans certaines spécialités sous-jacentes comme les Systèmes à Base de Connaissances, les systèmes de diagnostic et d'assistance, etc.

Toutes ces disciplines concourent à apporter des éléments de réponse à la problématique de conception des Systèmes Homme-Machine (SHM) d'une manière générale et des Interfaces Homme-Machine (IHM) en particulier. C'est pourquoi la méthode proposée dans l'article combine nécessairement des méthodes et modèles provenant de différents domaines.

Toutefois, soulignons que dans la majorité des approches de conception des IHM proposées, quand, dans le meilleur des cas, le procédé est modélisé, celui-ci n'est pas représenté en état de dysfonctionnement. De ce fait, les interfaces produites ne seront adaptées surtout qu'aux situations de fonctionnement normal et présentent, par conséquent, des lacunes au niveau de l'expression des besoins des opérateurs relatifs à ces situations peu ou mal analysées. L'occurrence d'un dysfonctionnement ramènera le système à une situation critique imprévisible où l'absence d'informations adéquates au contexte anormal rendra la tâche de l'opérateur plus difficile et son niveau de stress potentiellement plus important. Le besoin en assistance est alors indispensable. La méthode proposée dans l'article, prenant en considération explicitement les dysfonctionnements, trouve ici toute son importance.

Dans cet article, qui étend (Moussa *et al.*, 2005) et qui se base largement sur (Riahi, 2004 ; Moussa, 2005), nous commençons par exposer une brève synthèse bibliographique sur les méthodes et outils d'analyse et de modélisation des systèmes complexes en mode de fonctionnement normal, ou anormal (appelé aussi mode dégradé, ou de dysfonctionnement), vus comme des pré-requis nécessaires à la conception des IHM, ces méthodes et outils venant en amont de l'analyse des tâches humaines. Nous présentons, par la suite, une approche globale d'analyse et de conception des IHM, proposée pour faciliter la déduction des Besoins Informationnels des Opérateurs (BIO). Nous nous intéressons principalement dans l'article aux étapes suivies pour l'analyse du procédé à contrôler et l'analyse de ses différents modes de fonctionnement dans l'objectif de construire un modèle d'interaction homme-machine qui facilitera par la suite la déduction des besoins informationnels des opérateurs et la spécification de l'interface appropriée. Ces étapes sont détaillées et illustrées à travers une étude de cas représentative d'une situation industrielle.

2 Méthodes et techniques d'analyse et de spécification

Notre souci étant de couvrir, à travers une analyse et une modélisation du système, les aspects de fonctionnement normal mais aussi et surtout de dysfonctionnement dans l'objectif de mieux cerner les BIO à proposer, selon les différents contextes opérationnels, notre attention ne se porte plus, uniquement, sur le procédé, mais plus généralement sur le système homme-machine. La question qui se pose là est comment procéder à cette analyse et modélisation et quels méthodes et outils utiliser ? A ce sujet, nous passons en revue un ensemble de méthodes d'analyse et de spécification, et ceci sans souci d'exhaustivité, mais plutôt de représentativité.

2.1 Méthodes d'analyse fonctionnelle et/ou structurelle des systèmes

Avec les systèmes industriels de complexité croissante, l'analyse fonctionnelle et/ou structurelle d'une installation devient fastidieuse, voire impossible sans l'adoption d'une ou de plusieurs méthodes. Plusieurs types de méthodes existent dans la littérature.

Utilisées dans de nombreux domaines industriels depuis une trentaine d'années, les méthodes cartésiennes (aussi appelées structurées) permettent de procéder à un découpage fonctionnel, hiérarchique du système (telles que les méthodes SA, SADT, SA-RT, ...); elles sont très pratiques pour représenter des systèmes complexes, et pour faciliter la communication dans les équipes projet, SADT étant par exemple présenté comme « un langage pour communiquer des idées » (IGL Technology, 1989). Il est toutefois bien connu que de telles méthodes présentent des carences au niveau de la description fine de la composante dynamique du système, et ce malgré les propositions de SA-RT (Hatley et Pirbai, 1991) à ce sujet.

D'autres méthodes sont proposées pour l'analyse fonctionnelle permettant l'étude des relations d'influence entre les variables (telles que les graphes de fluence (Sinclair *et al.*, 1965) et la méthode MFM (Multilevel Flow Modelling) (Lind, 1990). Néanmoins, les graphes de fluence présentent un risque de schémas très complexes, difficiles à lire et à exploiter et la méthode MFM est encore trop peu utilisée en entreprise.

On trouve également les méthodes qualifiées de systémiques permettant des modélisations centrées sur les données et les traitements, telles que les méthodes MERISE (Tardieu *et al.*, 1991) et AXIAL (Pelleaumail, 1987), MERISE étant la plus utilisée dans les pays francophones. Par exemple, pour ce qui est de l'étude de l'existant et de l'analyse des besoins, la méthode MERISE propose d'intéressants modèles centrés d'une part sur les données et d'autre part sur les traitements (Nanci et Espinasse, 2001). Mais on considère que de telles méthodes sont mieux adaptées au domaine de l'informatique de gestion et des systèmes d'information qu'à celui des applications industrielles critiques.

Les méthodes d'analyse orientées objets (toutes basées sur UML actuellement suite à l'unification des trop nombreuses méthodes existantes, durant les années 90) (Jacobson *et al.*, 2000 ; cf. www.omg.org) nécessitent des extensions adaptées aux systèmes industriels complexes.

Si l'on évalue informellement ces classes de méthodes (en nous focalisant sur les méthodes les plus représentatives) selon plusieurs critères (degré de recouvrement des aspects structurel et fonctionnel de l'analyse, capacité de découpage hiérarchique du système, degré de visibilité de la circulation du flux d'information, degré de simplicité impliquant la facilité d'apprentissage de la

méthode, aspect graphique de la méthode, degré d'applicabilité pour le domaine concerné par notre étude à savoir les applications industrielles), la méthode SADT semble la plus adéquate. En effet, cette méthode, bien que développée en fin des années 70, reste parmi les méthodes d'analyse fonctionnelle les plus utilisées pour les systèmes industriels. Elle est particulièrement adaptée à l'analyse et la conception des systèmes. Elle est utilisée, ici, spécialement pour avoir une vue structurée sur le système global, qui est généralement complexe.

Cette décomposition SADT nous permettra par la suite d'étudier les sous-systèmes considérés comme « élémentaires », selon notre approche de décomposition et y appliquer les méthodes d'analyse de dysfonctionnements. Notons aussi que cette méthode est utilisée depuis de nombreuses années dans le domaine de l'interaction homme-machine (Abed *et al.*, 1992, 1995 ; Benaïssa, 1993 ; Abed, 2001 ; Ezzedine et Kolski, 2004).

2.2 Des méthodes adaptées à l'analyse des dysfonctionnements

Une analyse des dysfonctionnements doit déboucher principalement sur l'ensemble des variables pertinentes et significatives de l'ensemble des causes possibles d'un dysfonctionnement donné, afin de pouvoir aider l'opérateur humain en salle de contrôle à localiser la cause de la panne et à rétablir une situation de fonctionnement normal.

Pour modéliser le système en mode dégradé ou de dysfonctionnement, trois principales classes de méthodes peuvent être distinguées (analyse préliminaire des dangers, analyse et regroupement des pannes, analyse détaillée causes/effets de panne) ; pour une revue quasi-exhaustive et des descriptions détaillées des méthodes listées ci-dessous, voir (Fadier, 1990 ; Villemeur, 1992 ; Laprie *et al.*, 1995). Ces trois classes de méthodes sont successivement décrites.

Dans la première classe, on retrouve principalement la Méthode d'Analyse Préliminaire des Dangers (APD). Son principe consiste à identifier les dangers d'une installation industrielle et ses causes (entités dangereuses, situations dangereuses, accidents potentiels), évaluer la gravité des conséquences (liées aux situations dangereuses et accidents potentiels), déduire tous les moyens, toutes les actions correctrices permettant d'éliminer ou de maîtriser les situations dangereuses et accidents potentiels mis en évidence. La méthode APD a été utilisée la première fois vers les années 60 aux Etats Unis, dans le cadre de l'analyse de sécurité de missiles. Ensuite, elle a été formalisée par l'industrie aéronautique, puis généralisée à de nombreuses industries : chimie, nucléaire, aéronautique, etc. Il est généralement recommandé de l'appliquer dès les premières phases de conception du système à risques.

Dans la seconde classe, on retrouve les méthodes d'analyse et regroupement des pannes, telles AMDE, MPCR, MTV :

- La méthode **AMDE** (Analyse des Modes de Défaillances et de leurs Effets) fut employée à partir des années 60 dans le domaine de l'aéronautique pour l'analyse de la sécurité des avions. Cette méthode inductive permet de : (1) évaluer les effets de chaque mode de défaillance des composants d'un système sur les différentes fonctions du système, (2) identifier les modes de défaillances ayant des effets sur la disponibilité, la fiabilité, la maintenabilité ou la sécurité du système. L'AMDE constitue une analyse préliminaire qui doit généralement être complétée par l'utilisation d'autres méthodes pour l'identification des combinaisons de défaillances pertinentes. La principale critique faite pour cette approche est sa lourdeur qui vient du principe de son analyse faite pour tous les composants du système quelle que soit leur

importance. Notons la variante AMDEC qui ajoute à l'AMDE la caractérisation de la criticité de chaque défaillance. L'AMDE et l'AMDEC sont très largement exploitées dans l'industrie.

- La méthode **MCPR** (Méthode de Combinaisons des Pannes Résumées) vient compléter l'AMDE, qui ne permet généralement d'étudier que les simples défaillances, par l'étude de combinaisons de ces défaillances. En revanche, sa possible lourdeur invite à ne l'employer qu'à bon escient, selon (Villemeur, 1992).
- La méthode **MTV** (Méthode de la Table de Vérité) consiste à recenser toutes les combinaisons d'états (états de fonctionnement et états de pannes) des composants, les unes après les autres et à en étudier leurs effets ; son principe est fort simple mais il se révèle irréalisable pour l'analyse manuelle de grands systèmes, en raison d'un très grand nombre de combinaisons à considérer.

Dans la troisième classe, on retrouve des Méthodes d'analyse détaillée causes/effets d'une panne, telles MDS, AdD, MACQ, MDCC :

- Le but essentiel de **MDS** (Méthode de Diagrammes de Succès) est la représentation de la fonction du système par des diagrammes de blocs. Ses limites sont rapidement apparues au début des années 60. La recherche de voies nouvelles et plus fines d'analyse a conduit aux méthodes AMDE et AdD.
- La méthode **AdD** (Arbre des Défaillances), aussi appelée Méthode de l'Arbre des Causes (MAC), est née au début des années 60 dans les bureaux de la société Bell Telephone. Son but est de déterminer les diverses combinaisons possibles d'événements qui entraînent la réalisation d'un événement indésirable unique, puis la représentation graphique de ces combinaisons au moyen d'une structure arborescente. La principale limite de cette méthode réside dans le fait que l'événement indésirable à analyser doit être défini d'une manière précise. Cette méthode est très largement répandue dans de nombreux domaines industriels.
- **MACQ** (Méthode de l'Arbre des Conséquences) fut employée pour la première fois entre 72 et 75 dans l'évaluation du risque lié aux centrales nucléaires aux Etats-Unis. Dérivée de la méthode des arbres de décision, elle fut ensuite appelée "méthode des arbres d'événements" (Event Tree Method). C'est l'outil le plus fréquemment utilisé pour la caractérisation et la détermination des accidents potentiels. Elle permet d'identifier les différentes séquences d'événements possibles (acceptables et inacceptables), les représenter, et les évaluer de manière quantitative et qualitative.
- **MDCC** (Méthode du diagramme Causes-Conséquences) a été initialement élaborée par le laboratoire RISØ au Danemark (début des années 70). Elle a été utilisée comme aide à l'analyse de fiabilité et de risque des centrales nucléaires dans les pays scandinaves. Elle combine les principes utilisés par la méthode déductive AdD et MACQ qui est inductive. C'est une méthode difficile à utiliser pour l'analyse des systèmes trop complexes. Une fois le système analysé, un document de spécification fonctionnelle peut être établi, précisant la structuration du système, l'ensemble de ses variables "pertinentes", le principe de son fonctionnement normal ainsi que ses différents dysfonctionnements possibles. Pour chacun de ces

dysfonctionnements, un diagnostic sera élaboré précisant ses symptômes, ses effets et les procédures de recouvrement possibles.

Il existe bien entendu de nombreuses combinaisons possibles de méthodes. Une solution représentative possible, exploitant au moins une méthode de chaque classe, et que nous avons retenue, consiste à utiliser une combinaison des méthodes complémentaires suivantes :

1. commencer par la méthode APD pour identifier l'ensemble des états de fonctionnement du système et les dangers possibles (étape préliminaire qui peut être ignorée si le système n'est pas trop complexe) ;
2. lister chaque état de dysfonctionnement ; pour chaque état, appliquer la méthode AMDEC pour identifier les différents modes de défaillances ;
3. analyser plus finement ces défaillances par la méthode AdD ;
4. analyser éventuellement les séquences d'événements conséquences possibles des défaillances par la MACQ (selon les caractéristiques et la complexité du système à analyser) ;
5. ayant l'ensemble des pannes et leurs effets, appliquer, éventuellement (selon les caractéristiques et la complexité du système à analyser), la méthode MCPR pour déduire un ensemble de pannes regroupées selon leurs effets.

Rappelons ici que notre objectif est de mener une analyse des dysfonctionnements des systèmes, afin de pouvoir dégager l'ensemble des variables pertinentes et significatives pour chaque cas de fonctionnement du système ainsi que l'ensemble des causes possibles d'un dysfonctionnement donné, afin de pouvoir aider l'opérateur dans sa tâche. En gardant en tête cet objectif, on s'aperçoit que la solution précédente en terme de combinaison de méthodes est centrée sur les méthodes AMDEC et AdD qui sont considérées comme complémentaires, et toutes les deux largement utilisées dans les entreprises : l'AMDEC, en effet, permet d'établir la liste des dysfonctionnements possibles d'un système donné et l'AdD la complète en analysant finement ces défaillances et précisant leurs causes possibles. Pour le cas d'étude qui sera traité en section 4, notons que la complexité moyenne du système industriel considéré ne permettra d'exploiter qu'une combinaison de ces deux méthodes.

2.3 Des techniques de spécification au besoin en modélisation de la dynamique dans les systèmes homme-machine

Dans le processus global de conception des IHM, l'activité de spécification (procédé, tâche interactive de supervision, etc.) fait appel aux notations et formalismes pour supporter l'activité de conception dont l'objet est la modélisation du système à concevoir. Le formalisme est donc une convention de représentation des concepts du modèle (Barthet, 1988). La figure 1 présente à ce sujet une taxonomie des techniques de spécification faisant ressortir le champ d'application et l'apport des différentes familles d'approches de modélisation. Il est clair qu'un nombre très important de techniques est actuellement disponible. Le lecteur intéressé se référera à (Jambon *et al.*, 2001) pour une description plus précise des approches citées dans la figure.

A ce sujet, de nombreux auteurs depuis près d'une vingtaine d'années, mettent en avant l'intérêt de baser les démarches d'analyse et conception des systèmes homme-machine (SHM) ou les applications interactives pour les systèmes critiques, sur des méthodes formelles et semi-formelles : cf. les travaux

de Palanque et ses collègues au LIIHS (Toulouse), au LAAS (Toulouse également), ceux de Abed et ses collègues au LAMIH à Valenciennes, ou encore ceux plus récents menés autour de l'utilisation de méthodes de spécification formelles (telle B) en IHM au LISI à Poitiers ; voir aussi Johnson et Palanque (2004) pour des synthèses ou des travaux représentatifs à ce sujet.²

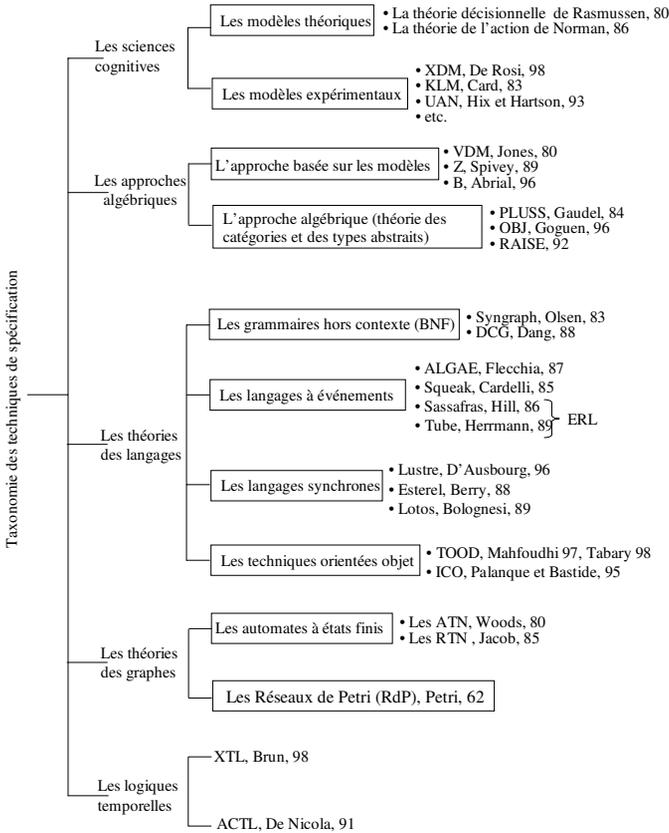


Figure 1. *Taxonomie des techniques de spécification* (inspirée de Jambon *et al.*, (2001))

Pour ce qui nous concerne, nous nous focalisons sur les méthodes et techniques permettant de venir en complément de celles envisagées en 2.1 et 2.2, visant à couvrir l'aspect dynamique du système, mais aussi de faciliter la modélisation dynamique des interactions homme-machine.

Notre choix se porte sur les Réseaux de Petri (RdP) (Petri, 1962 ; Petersen, 1981) considérés depuis longtemps comme très adéquats pour la modélisation de l'interaction homme-machine (Abed *et al.*, 1992 ; Palanque, 1992 ; Palanque et

² Dans la série de workshops DSV-IS (Design, Specification and Verification of Interactive Systems), de nombreux articles sont consacrés à ces aspects.

Bastide, 1995 ; Mahfoudi, 1997 ; Abed, 2001 ; Navarre, 2001 ; Gomes *et al.*, 2001...). Ce choix a été opéré suite à une étude bibliographique ayant mis en évidence l'aptitude de cette technique à répondre à des critères qui sont importants et applicables pour la spécification des IHM pour les procédés industriels complexes (Riahi, 2004). En effet, et pour anticiper sur la suite, précisons que les RdP disposent d'une définition formelle, ils offrent un grand pouvoir d'expression des aspects tels que le synchronisme, le parallélisme, ils sont exécutables, ils disposent de nombreuses techniques de vérification automatique des propriétés (borné, vivant, réinitialisable, etc.), il offrent une représentation graphique peu contraignante, etc. ; autant de critères adaptés aux contraintes de spécification et de génération automatique des interfaces graphiques pour les applications industrielles complexes et critiques.

Les RdP ont, d'ailleurs, toujours été efficaces dans le domaine de l'automatique qualifiée d'humaine (la conception des systèmes étant centrée sur l'homme ; cf. (Millot, 1988)) et ils deviennent de plus en plus utilisés dans celui de la conception d'interfaces homme-machine. Nous proposons, ci-dessous, certains critères ayant conduit à l'adoption des RdP pour nos travaux (plus de détails sont disponible dans Riahi (2004), cf. aussi Palanque (1992), Jambon *et al.* (2004)) :

- la puissance des RdP en terme de vérifiabilité des propriétés : la vérification sur modèle permet d'assurer, de manière certaine, les propriétés. Cette méthode exige que les propriétés soient au préalable formellement définies. Les réseaux de Petri permettent de s'assurer, en explorant le graphe des marquages, que l'utilisateur a bien accès à toutes les commandes du système (Palanque et Bastide, 1996) ou que celui-ci peut achever une action entreprise.
- le degré de prise en considération de l'aspect parallélisme : la concurrence dans la modélisation des tâches opérateur est un critère fort important. On peut modéliser la concurrence en utilisant le parallélisme vrai ou l'entrelacement. Les RdP, comme la logique temporelle XTL (Brun, 1998) ou la notation UAN (User Action Notation) (Hartson et Gray, 1992) sont des exemples qui remplissent cette propriété.
- le degré de la prise en considération de l'aspect synchronisme : implémenter la synchronisation des actions et des processus est indispensable dans la modélisation du dialogue homme-machine. Les RdP, de même que les langages synchrones Lotos (Turner, 1993) et Lustre (Halbwachs *et al.*, 1991) assurent ce genre de critère.
- Le séquençement des actions et les contraintes temporelles : ce critère regroupe cinq sous-critères (Balbo, 1994) permettant de préciser les relations entre les actions : la séquence entre deux tâches (A puis B) ; l'alternative (ou choix) entre deux tâches (A ou B) ; la composition de deux tâches dans un ordre quelconque (A & B) ; l'itération (A un certain nombre de fois) ; l'expression des délais entre les tâches. Les RdP temporisés offrent aussi des possibilités de modélisation à ce sujet.
- Les capacités génératrices : selon les objectifs méthodologiques auxquels on s'attache, une approche de conception peut aboutir sur des spécifications informelles inexploitable d'une manière automatisée, comme elle peut aboutir sur la génération automatique ou semi-automatique des IHM à partir des spécifications obtenues. Bien que souvent on se contente de l'utilisation d'une notation si elle est suffisamment expressive pour nos besoins, les

capacités génératrices restent la spécificité des formalismes. Par exemple, même si cela sort du cadre de l'étude, il peut être intéressant de souligner que les réseaux de Petri, dont le graphe des marquages fournit l'état courant du système, son évolution possible, ainsi que les étapes précédentes, permettent de générer de l'aide contextuelle en expliquant à l'utilisateur comment il est parvenu dans l'état courant, comment il peut continuer sa tâche et même lui donner le chemin optimal de réalisation comprenant le moins d'étapes possibles (Palanque et Bastide, 1997).

2.4 Conclusion sur les méthodes et techniques disponibles

Il existe énormément de méthodes et techniques, destinés à l'analyse et la modélisation des systèmes, de même qu'à la spécification. La littérature est très riche à ce sujet, et non nécessairement en rapport direct avec le domaine de l'interaction homme-machine. Certaines méthodes et techniques sont spécifiquement destinées à l'analyse des systèmes dans des situations de dysfonctionnement.

La difficulté réside dans le choix et la combinaison des méthodes et techniques, selon les caractéristiques et les objectifs du projet concerné ; nous avons fourni quelques critères à ce sujet.

Il s'agit dans la partie suivante de fournir une approche globale en terme d'analyse et de modélisation de système, avec une visée d'identification des besoins informationnels des opérateurs. Cette proposition sera illustrée ensuite en partie 4 par une étude de cas.

3 Approche globale proposée

Nous nous focalisons ici sur les premières étapes d'un cycle de conception-évaluation des IHM dans les systèmes industriels complexes.

Dans cet article, nous nous intéressons, plus précisément, aux étapes d'analyse 2, 3 et 4 ci-dessous selon une démarche facilitant la déduction des besoins informationnels des opérateurs (BIO). L'approche proposée identifie sept étapes principales : ³

1. Une première étape d'analyse préliminaire du processus et de son système de commande est nécessaire.
2. La seconde étape consiste à analyser le système homme-machine (SHM) en termes de processus, son système de commande et les tâches opérateur. Différentes techniques et méthodes sont utilisées à cet effet. Dans cette étape très importante, il s'agit de combiner un ensemble de méthodes décrites précédemment dans les sections 2.1 et 2.2 de l'article.
3. La troisième étape consiste en la modélisation du comportement de l'opérateur. Cette modélisation doit exprimer l'interaction entre l'opérateur et le système via l'interface homme-machine.

³ Pour d'autres propositions de démarches méthodologiques adaptées aux systèmes industriels complexes, il est possible de consulter par exemple Millot et Debernard (1993), Kolski (1997), Abed (2001), Abed et Ezzedine (1998) ou encore Gilmore et al. (1989). L'approche proposée dans cet article est généralement cohérente avec celles proposées par ces auteurs, tout en se focalisant sur les premières étapes d'un projet et en détaillant l'analyse profonde du système homme-machine.

4. La quatrième étape assure l'identification des BIO en termes d'objets de l'interface.
5. Une fois les objets de l'interface déduits, cette étape consiste à spécifier l'interface en termes de vues et objets graphiques.
6. Il est possible de tirer bénéfice de la technique formelle utilisée dans la spécification de l'interface pour vérifier les spécifications proposées, au niveau de cette étape.
7. La dernière étape de cette approche est dédiée à la production (qu'elle soit automatique, semi-automatique ou tout simplement manuelle) de l'interface homme-machine dédiée à la supervision du système.

Bien entendu, aussi bien en amont qu'en aval de la production de l'interface homme-machine, l'évaluation sous l'angle de l'utilité et de l'utilisabilité joue un rôle essentiel sur lequel nous ne reviendrons pas dans l'article, par manque de place. De nombreux ouvrages, chapitres de livres, thèses, articles dans des revues et congrès sont disponibles à ce sujet.

La **première étape**, réalisée en amont de l'approche, débouche sur un document rassemblant les données du procédé et ses différentes contraintes techniques et fonctionnelles. Cette étape est typique en ingénierie des systèmes et en génie automatique.

La **seconde étape**, quant à elle, consiste à conduire une analyse profonde, point de départ de la spécification de l'IHM. Cette étape est réalisée en coopération avec les concepteurs de l'installation industrielle et les opérateurs humains, futurs utilisateurs de l'interface homme-machine en salle de contrôle.

L'analyse du système homme-machine consiste, en effet, à identifier les sous-systèmes significatifs et importants à contrôler et à analyser leur comportement. L'objectif recherché de cette analyse est d'identifier les différentes vues (la plupart graphiques) appropriées pour le contrôle et la commande de ces sous-systèmes. L'objectif est de fournir à tout instant, une vue graphique par sous-système, reflétant son état de fonctionnement et regroupant l'ensemble des BIO à cet instant là. Ce travail peut s'effectuer en trois phases (figure 2).

- Phase 1 : procéder à une décomposition hiérarchique du SHM moyennant la méthode SADT identifiant un ensemble de sous-systèmes élémentaires plus clairs, faciles à étudier et à modéliser.
- Phase 2 : analyser les dysfonctionnements de chaque sous-système en appliquant la combinaison des méthodes AMDE et AdD.
- Phase 3 : identifier pour chaque sous-système et pour chaque contexte de fonctionnement, les tâches nécessaires de l'opérateur : l'analyse des différents contextes de fonctionnement du système étudié étant faite, il importe, à présent, d'identifier pour chacun des contextes préalablement identifiés (contexte de fonctionnement normal et contextes de fonctionnement anormaux) l'ensemble des tâches opérateur nécessaires pour le suivi et le contrôle. L'analyse de ces tâches permet de définir les actions à entreprendre par les opérateurs. Les paramètres de ces actions constitueront avec l'ensemble des variables d'entrée/sorties et les variables de contrôle (identifiées au moyen de SADT, AMDE et AdD) l'ensemble des besoins informationnels de l'opérateur (BIO).

En partant des modèles précédents conduisant à positionner les interventions humaines dans des situations aussi bien normales qu'anormales, la **troisième**

étape consiste à modéliser d'une manière formelle l'interaction homme-machine, en exploitant les réseaux de Petri.

Dans la **quatrième étape**, il s'agit de reprendre l'ensemble des réseaux de Petri modélisés précédemment, pour en étudier (situer) chaque indice suggérant une interaction homme-machine et mettre en conséquence en évidence l'ensemble des BIO nécessaires, prenant la forme plus particulièrement de variables d'information d'une part, de variables de commande d'autre part.

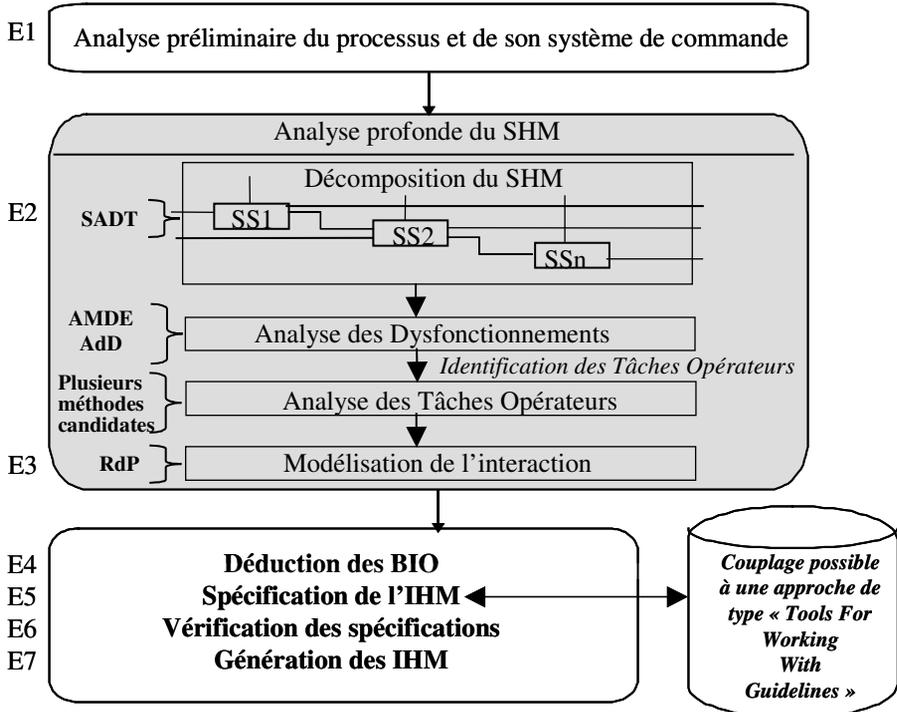


Figure 2. Analyse du Système Homme-Machine (SSi : sous-système i)

La spécification de l'interface proprement dite peut alors commencer (**cinquième étape**). Elle consiste à déduire pour l'ensemble des BIO identifiés, les objets d'interaction correspondants pour les différentes tâches en salle de contrôle. Notons que la prise en compte des critères relatifs à l'ergonomie des logiciels peut éventuellement être assurée à ce niveau avec le couplage à un outil d'aide, à base de connaissances, du type TFWWG (Tools For Working With Guidelines) si l'on se réfère au courant de recherche au niveau international animé en particulier par Jean-Vanderdonckt (Vanderdonckt, 1994, 1999 ; Moussa *et al.*, 2000 ; Vanderdonckt et Farenc, 2000 ; Bereikdar, 2004 ; Mariage, 2005).

L'utilisation des RdP pour la modélisation vise à préparer le terrain à des vérifications formelles et une validation *a priori* des interfaces au niveau de la **sixième étape** ; ce qui permet en principe de gagner un temps considérable dans le cycle de développement des interfaces.

Pour ce qui est de la production de l'IHM (**étape sept**), de nombreux langages et environnements de développement sont disponibles, qu'ils soient spécifiques ou non aux applications de supervision ; cf. à ce sujet Fekete et Girard (2001) pour un aperçu significatif.

Dans l'article, ces trois dernières étapes (cinq à sept), « classiques » en IHM, ne sont pas traitées ; nous ne nous focaliserons donc que sur les trois étapes précédentes (analyse du système en mettant en particulier l'accent sur l'analyse de ses dysfonctionnements, modélisation de l'interaction homme-machine, déduction des BIO). Celles-ci sont illustrées sur un cas représentatif d'un système industriel réel.

Notons aussi dès à présent que l'analyse de la tâche opérateur n'est pas traitée en profondeur ici, celle-ci faisant d'ailleurs l'objet d'un thème de recherche à part entière en IHM. A ce sujet, l'analyse des tâches humaines nécessite un haut niveau de connaissance et de savoir faire et une collaboration étroite et pluridisciplinaire entre les différents intervenants du processus de développement des applications interactives. Rappelons que plusieurs méthodes d'analyse et de modélisation de la tâche sont proposées dans la littérature ; voir par exemple Scapin et Bastien (2001), Diaper et Stanton (2004).

Nous supposons à cet effet que l'analyse de la tâche est accomplie en adoptant une méthode appropriée parmi celles-ci, de nombreuses méthodes étant candidates (MAD, HTA, CTT...). Le résultat de cette analyse permet de définir les actions relatives aux tâches de l'opérateur. Les paramètres de ces actions constitueront avec l'ensemble des variables d'entrée/sorties et les variables de contrôle (identifiées au moyen de SADT, AMDE et AdD), l'ensemble des besoins informationnels de l'opérateur (BIO).

4 Etude de cas

L'application étudiée concerne le cas d'un procédé industriel de fabrication de godets métalliques remplis par une solution chimique préparée préalablement, fig. 3.

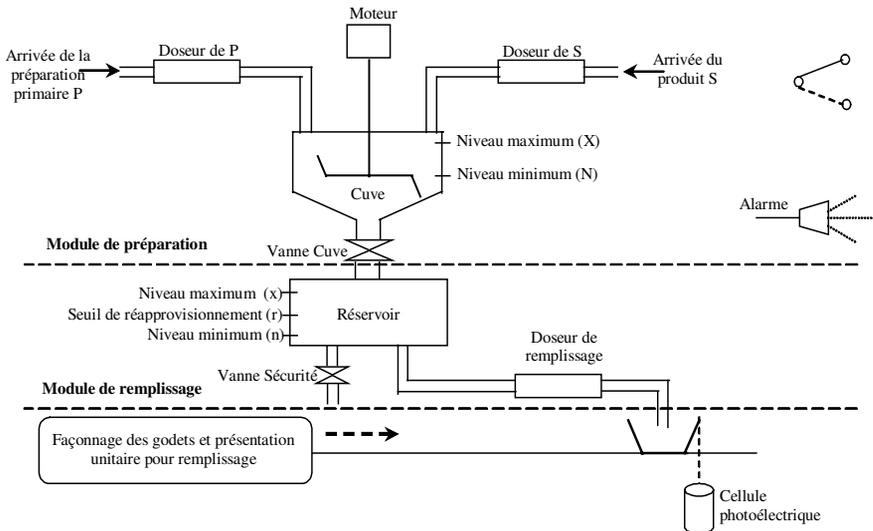


Figure 3. Procédé à superviser en salle de contrôle

Le processus simplifié de préparation comprend deux phases principales : (1) le façonnage des godets par emboutissage de pièces métalliques prédécoupées ; (2) le versement dans les godets d'une dose de solution chimique. La solution est obtenue à partir d'une préparation primaire P et d'un produit S. L'étude a porté sur cette deuxième phase qui met en œuvre deux modules : un module de préparation et un module de remplissage. La fonctionnalité principale de la partie étudiée de ce système consiste à préparer le mélange puis à en remplir les godets.

4.1 Analyse du système homme-machine

Décomposition fonctionnelle du système

La première étape de la démarche consiste à appliquer une méthode permettant une analyse fonctionnelle, par exemple SADT (comme dans Benaissa, 1993 ; Abed *et al.*, 1992, 1995 ; Abed et Ezzedine, 1998), pour décomposer le système et déceler les principaux sous-systèmes élémentaires pour les étudier. L'actigramme A0 (figure 4) présente à titre d'exemple la fonctionnalité principale "préparer mélange et remplir godets". Une décomposition de cette fonctionnalité peut révéler, dans l'actigramme A1, deux sous-fonctionnalités "préparer mélange" et "remplir godets" (figure 5).

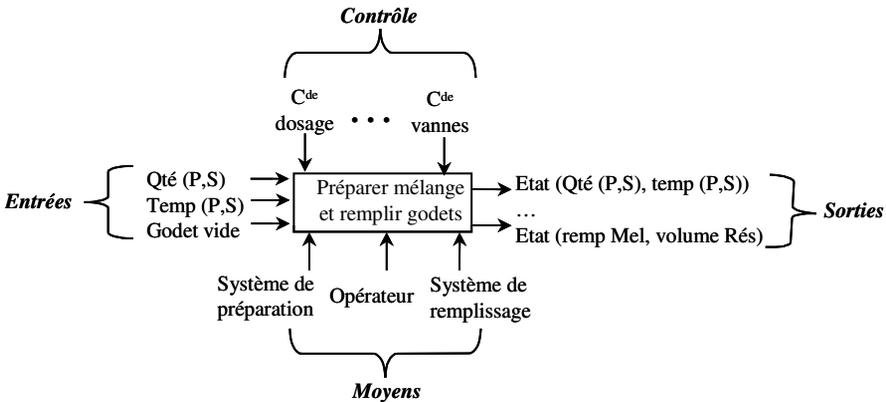


Figure 4. Décomposition du système par SADT (Actigramme A0)
Rem : pour des raisons de simplification, toutes les données ne sont visibles sur la figure 4

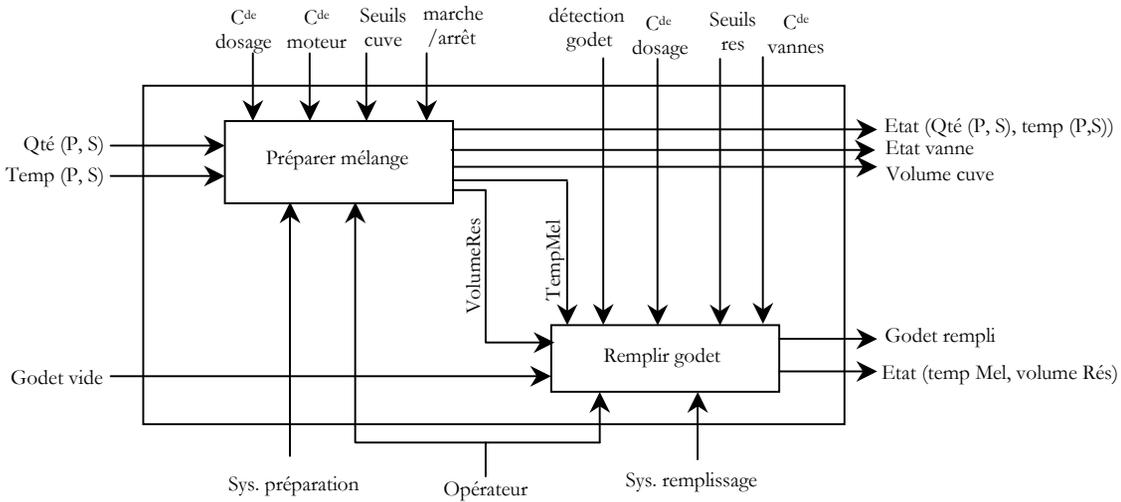


Figure 5. Décomposition du système par SADT (Actigramme A1)

Le système étudié étant simple, nous pouvons nous arrêter à ce premier niveau de décomposition et considérer les deux sous-systèmes élémentaires :⁴

- sous-système de préparation (S1)
- sous-système de remplissage (S2)

Analyse des dysfonctionnements du système.

- Analyse par la méthode AMDE

L'élaboration d'une AMDE demande une connaissance approfondie du fonctionnement normal et/ou dégradé du système étudié. Le recensement des modes de défaillances et de leurs causes éventuelles s'appuie sur l'expérience d'exploitation acquise pour des matériels similaires.

Les principales difficultés surgissent au niveau : de la définition des effets des défaillances (variations des paramètres, actions automatiques ou manuelles,...), de l'évaluation de ces effets.

Au cours de l'élaboration de l'AMDE, un contact permanent entre le(s) spécialiste(s) en sûreté de fonctionnement et les spécialistes des systèmes est nécessaire. Dans le cas de combinaisons des défaillances non prévues lors de la conception du procédé industriel, ce dialogue permet d'aboutir à l'élaboration d'hypothèses aussi réalistes que possible (Villemeur, 1992).⁵

⁴ Pour une meilleure compréhension, rappelons d'abord que dans SADT, une activité est représentée par un actigramme. Un actigramme peut posséder des données d'entrée situées en partie gauche (flèches entrantes), des données de sorties situées en partie droite (flèches sortantes), des données de contrôle situées au-dessus (flèches entrantes), des mécanismes (acteurs humains, logiciels, machines...) situés en-dessous (flèches entrantes). Chaque actigramme peut être détaillé à un niveau inférieur par d'autres actigrammes.

⁵ Au départ, notre tâche d'élaboration de l'AMDE n'était pas aisée dans la mesure où nous n'étions pas assistés par des spécialistes du domaine et où nous n'avions qu'une connaissance théorique sur ces types de méthodes. Nous nous sommes donc inspirés des documents

Pour notre cas d'application, l'AMDE a été élaborée pour les deux sous-systèmes mis en évidence avec la décomposition fonctionnelle précédente :

- Le premier sous-système (S1) est responsable de la fonctionnalité de préparation du mélange. Il est composé de deux doseurs (l'un pour la solution primaire P et l'autre pour le produit S), d'une cuve et d'un moteur pour mélanger la solution, d'une vanne-cuve permettant l'écoulement de la solution vers le réservoir et de cinq capteurs (pour mesurer les températures des entrées P et S ainsi que le volume courant dans la cuve et contrôler l'atteinte des niveaux seuil minimum et seuil maximum).
- Le deuxième sous-système (S2) est responsable de la fonctionnalité de remplissage des godets. Il est composé d'un réservoir, d'une vanne de sécurité permettant l'évacuation du mélange vers l'extérieur, d'un doseur du mélange pour le remplissage et d'un capteur de température de la solution de remplissage.

L'élaboration de l'AMDE consiste donc à considérer ces deux sous-systèmes et étudier pour chacun d'eux, les modes de défaillances de leurs composants, les causes possibles, ainsi que les effets de ces défaillances sur le système, les moyens de détection et les éventuelles actions de l'opérateur. Le résultat de cette analyse produit un tableau renseignant sur chaque composant, ses modes de défaillances, les Causes possibles, les effets sur le sous-système concerné, les effets sur tout le système, les moyens de détection et les actions de l'Opérateur. On présente ici (tableau 1) un extrait de ce résultat.

Le lecteur intéressé trouvera dans (Riahi, 2004) l'ensemble des tableaux AMDE élaborés sur cette étude de cas.

disponibles dans les Traités des Techniques de l'Ingénieur (Ridoux, 1999 ; Zwingelstein, 1999) pour bien comprendre comment traiter des cas d'applications industrielles réelles, étudier leurs différents composants et chercher les modes de défaillances possibles pour chaque composant.

| Composant | Modes de défaillances | Causes possibles | Effets sur S1 | Effets sur tout le système | Détection | Actions de l'opérateur |
|------------|---------------------------------|---|---|--|---|------------------------|
| Cuve | Absence seuil min dans la cuve | Blocage vanne cuve ouverte Fuite conduite du doseur de P (ou du doseur de S) Bouchage conduite du doseur de P (ou du doseur de S) Fuite cuve Fuite conduite cuve-vanne cuve | arrêt moteur arrêt système préparation | Absence seuil min dans le réservoir Arrêt système | Out.Alarme MinCuve = 0 | Fermeture vanne cuve |
| | Atteinte seuil max dans la cuve | Blocage vanne cuve fermée Bouchage conduite cuve-vanne cuve | arrêt moteur arrêt système préparation | Arrêt système | Out.Alarme MaxCuve = 1 | Ouverture vanne cuve |
| Vanne Cuve | Refus Ouverture | Défaillance mécanique (DM) Défaillance électrique (DE) | Arrêt moteur niveau max cuve atteint | Absence niveau min réservoir Arrêt système | Out.VanCuve <> Cd.VanCuve | |
| | Refus Fermeture | Défaillance mécanique (DM) Défaillance électrique (DE) | Arrêt moteur Absence seuil min cuve | | Out.VanCuve <> Cd.VanCuve | |
| | Réponse intempestive | Défaillance mécanique (DM) Défaillance électrique (DE) | Arrêt moteur Absence seuil min cuve | Arrêt système Absence niveau min réservoir | Out.Alarme MaxRes = 1 et Out.VanCuve = 1 Cd.VanCuve = 1 et Out.VanCuve = 0 | |

Tableau 1. Extrait des résultats de l'analyse AMDE pour le sous-système relatif à la préparation du mélange

On remarque entre autre que, pour la colonne relative aux effets sur tout le système, on préconise, pour certaines défaillances, un « arrêt système » et non un « arrêt système de remplissage » : on analyse ici le système de préparation et sa défaillance peut engendrer l'arrêt du système global y compris bien sûr le sous-système de remplissage ; ce qui nous intéresse ici est la mise en évidence de l'évènement redouté « arrêt système ».

- Elaboration de l'arbre des défaillances (AdD)

Rappelons que la méthode de l'arbre des défaillances (AdD) suppose les événements indésirables du système à étudier comme étant connus, et procède à une analyse plus fine de ces événements en précisant leurs causes éventuelles et ce, en se référant aux résultats de l'analyse AMDE menée précédemment.

Dans la pratique, là aussi, la tâche la plus délicate consiste à trouver « l'évènement redouté principal ». Il faut remonter tout le tableau AMDE et revoir les effets sur le système et le principe de fonctionnement pour déduire quel est l'évènement redouté principal et à quoi celui-ci est dû. Cet événement peut être dû soit à l'arrêt du sous-système de préparation (S1), soit à l'arrêt du sous-système de remplissage (S2). Il faut donc analyser de près chacun de ces deux événements indésirables : (E1) : arrêt du sous-système de préparation (S1) et (E2) : arrêt du sous-système de remplissage (S2). Chacun de ces 2 événements (E1 et E2) renvoie à un état de dysfonctionnement relatif (D1 et D2), que l'opérateur doit restituer. Les tâches de correction relatives à ces deux états de dysfonctionnement sont analysées et définies dans l'étape suivante.

L'élaboration de l'AdD permet à ce niveau, de venir en aide au diagnostic des causes éventuelles. La figure 6 présente un extrait du résultat du développement de l'arbre des défaillances relatif à l'évènement redouté du système étudié : "arrêt système".

L'AdD permet d'identifier les combinaisons possibles d'événements élémentaires qui causent ces dysfonctionnements. Ces événements élémentaires reviennent, en fait, à des défaillances des composants matériels du système. Les principales défaillances discernées par l'AdD sont les suivantes : problème de dosage de la solution primaire P, problème de dosage du produit S, perturbation du niveau du réservoir, perturbation du niveau de la cuve, problème d'évacuation de la solution du réservoir, problème de remplissage du godet, arrêt du moteur, défaillances des capteurs (de température et de volume), problèmes au niveau des vannes (vanne cuve et vanne sécurité).

Le lecteur intéressé trouvera dans (Riahi, 2004) l'ensemble des arbres de décision élaborés sur cette étude de cas.

Analyse de la tâche opérateur

Une fois le fonctionnement du système étudié, les dysfonctionnements possibles analysés, l'étape qui suit consiste à analyser la tâche de l'opérateur humain face à ces différents états de fonctionnement du système (fonctionnement normal et fonctionnements anormaux).

En se référant à Rasmussen (1986), Hoc et Amalberti (1995), en cas de fonctionnement normal, l'opérateur a besoin de constater le bon état de marche en question et de surveiller les dérives éventuelles. Cet état de bon fonctionnement peut être reflété au niveau de l'interface par une représentation d'un ensemble de variables pertinentes synthétisant l'état courant du système.

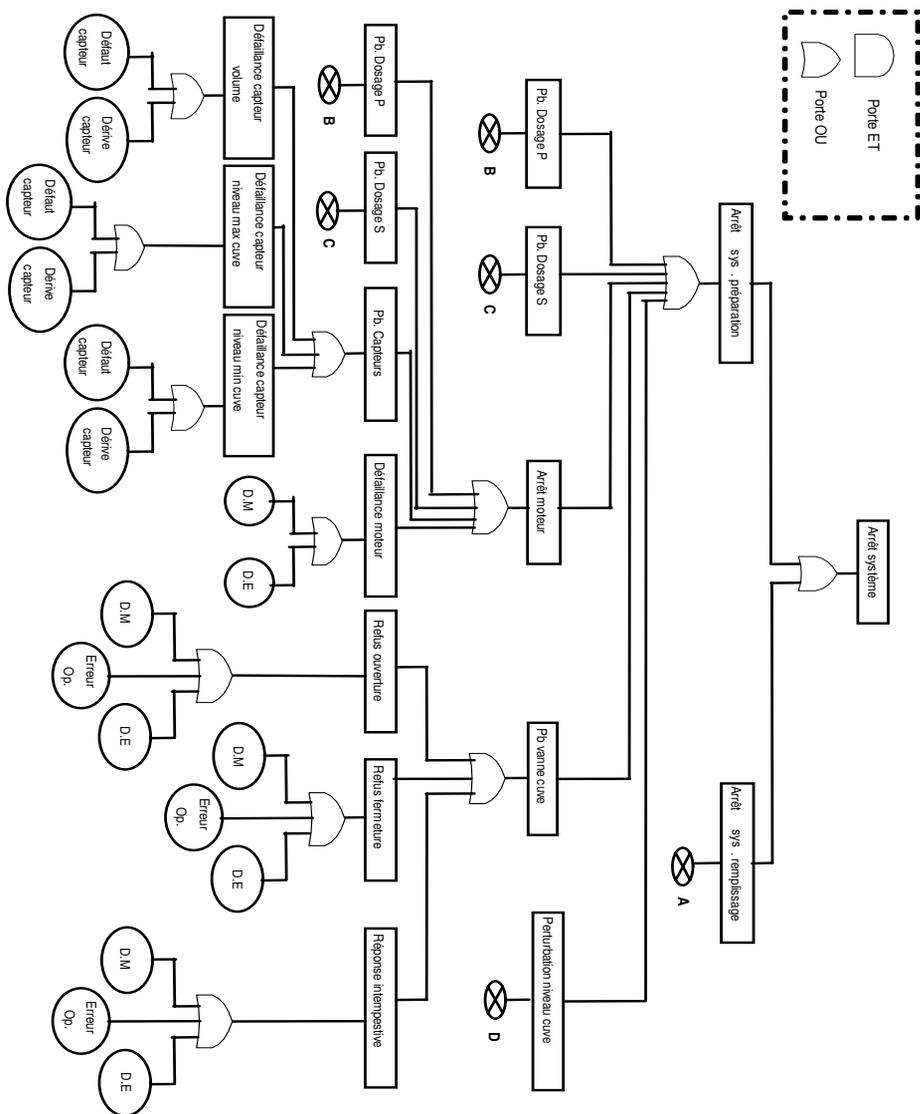


Figure 6. Elaboration de l'AdD pour l'événement redouté "arrêt système" (avec D.E. : Défaillance électrique, D.M. : Défaillance mécanique)

Dans la réalité, les variables représentatives de l'état de bon fonctionnement d'un procédé industriel sont identifiées avec les experts de celui-ci ; l'expérience montre que souvent quelques variables peuvent donner un bon aperçu de l'état du procédé (De Keyser, 1980 ; Taborin, 1989). Pour notre exemple d'application, en simulant nécessairement une telle analyse, on part sur l'hypothèse que l'état de fonctionnement normal peut être représenté par l'ensemble des informations suivantes :

- le volume courant dans la cuve avec les niveaux min et max (Out.VolCuve, Cs.MinCuve, Cs.MaxCuve),
- l'état du moteur (Out.Moteur),
- le volume courant dans le réservoir avec les niveaux min et max (Out.VolRes, Cs.MinRes, Cs.MaxRes),
- la température de la solution mélange dans le réservoir (Out.TempMel).

Cet ensemble de variables constitue les besoins informationnels de l'opérateur (BIO) en cas de fonctionnement normal. En cas de dysfonctionnement, l'opérateur doit être averti du mauvais fonctionnement par des signaux d'alarme et un ensemble d'informations en termes de variables représentatives au niveau de l'interface résumant la situation en cours. L'opérateur doit alors réagir le plus rapidement possible et intervenir sur un certain nombre de variables de commande, exécutant sa tâche de correction pour ramener le système en état de fonctionnement normal (Rasmussen, 1986 ; Hoc, 1996). Cet ensemble de signaux d'alarme, d'information et de commande constitue les BIO pour la situation courante et dépendra du cas du dysfonctionnement en question et de la tâche de correction de l'opérateur. L'ensemble de BIO, pour le cas d'une telle situation, ne pourra donc être défini qu'après avoir mené une analyse détaillée de la tâche opérateur.

Une analyse préliminaire pour notre cas d'application révèle deux tâches humaines principales : une tâche de surveillance en cas de fonctionnement normal, et une tâche d'analyse et de reprise en cas de dysfonctionnement du système. Cette dernière doit être détaillée relativement à chaque événement 'élémentaire' mis en évidence par l'arbre des défaillances produit par l'AdD. Pour chacun de ces événements élémentaires, l'opérateur doit accomplir une suite d'actions sur les composantes matérielles du système afin de restituer son bon état de fonctionnement. Cette étude a pour objectif de construire un modèle de la tâche opérateur en termes d'actions élémentaires (différents modèles de tâche bien connus en IHM sont décrits dans (Diaper et Stanton, 2004) ; rappelons aussi que les aspects de modélisation de tâche ne sont pas traités dans cet article). Les actions de l'opérateur sont déduites à partir du tableau AMDE et de l'arbre AdD. Une manière de déduire la modélisation de l'interaction homme-machine à partir de cette analyse est expliquée dans le paragraphe suivant.

Modélisation de l'interaction homme-machine

Une fois l'analyse des dysfonctionnements possibles du système effectuée, l'arbre des défaillances élaboré et les tâches opérateur d'intervention analysées, l'étape suivante de la démarche consiste à élaborer un modèle de l'interaction homme-machine en fonction de l'évolution de l'état du système et des interventions de l'opérateur humain afin de pouvoir déduire par la suite l'ensemble des BIO relatifs à chaque état. Cette modélisation peut être basée sur une composition de structures élémentaires de RdP (Palanque et Bastide, 1997 ; Riahi *et al.*, 2000 ; Khelil, 2001 ; Moussa *et al.*, 2002). Les RdP proposés ici sont les RdPI (Réseaux de Petri interprétés) (Moalla, 1985) ; ils ont été choisis dans la mesure où ils permettent une validation formelle, non traitée dans cet article.

Ce type de réseaux introduit les notions d'événement et de condition ainsi que la notion d'action. En effet, une condition de passage (C_j), un événement de déclenchement (Ev_j) et une action éventuelle (A_j) sont associés à chaque transition T_j d'un RdPI. Pour modéliser l'interaction homme-machine moyennant les RdPI, nous avons convenu d'utiliser les places pour représenter « l'état du comportement de l'opérateur » vis-à-vis de l'évolution du système

(Khelil, 2001). Nous considérons qu'une tâche opérateur est composée d'un ensemble organisé d'actions élémentaires. La structure modélisant une action élémentaire (par ex. une action de régulation de température) est donnée en figure 7.

La construction du modèle global de l'interaction homme-machine s'effectue à partir de structures de base modélisant les différentes actions élémentaires de l'opérateur. Elle se base sur l'application de différentes opérations de composition. Son principe réside dans le fait que la construction du modèle de la tâche n'utilise que des structures et des règles de composition définies. Cela est important pour assurer par avance de bonnes propriétés au modèle obtenu. Toutes les actions de l'opérateur (élémentaires ou non) sont ordonnées selon des compositions typiques : séquentielle, parallèle, alternative, de choix, itérative ou de fermeture (Khelil, 2001). Cette technique de modélisation, que nous ne détaillerons pas ici, nous permet de déduire les BIO en fonction des changements de contexte de fonctionnement du système industriel.

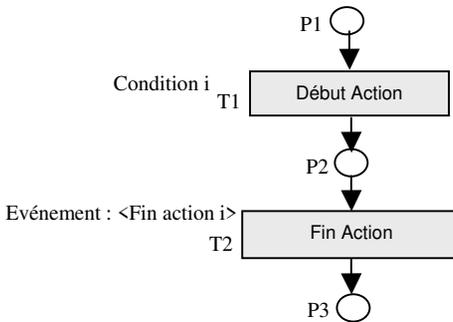


Figure 7. Structure modélisant l'état d'un opérateur face à une action élémentaire

L'algorithme de passage de l'AdD (précisant pour chaque défaillance les causes possibles) et du modèle de la tâche opérateur (devant préciser les actions élémentaires de l'opérateur face à une situation de dysfonctionnement) au modèle RdP, est décrit globalement ci-dessous.

Algorithme Déduire-modèle-RdP : /* pour un dysfonctionnement donné */

Début :

Étape 1 : parcourir l'arbre AdD jusqu'à localiser le nœud $n_{i,j}$ correspondant à l'événement redouté relatif à cet état de dysfonctionnement

Étape 2 : Traduire la sous-arborescence ayant pour racine le nœud $n_{i,j}$ en un RdP conformément aux règles suivantes :

- 1- une branche OU est traduite par une composition choix inclusive de RdP (*Composition ou inclusif*)
- 2- une branche ET est traduite par une composition parallèle de RdP
- 3- une branche OU exclusif est traduite par une composition alternative de RdP

Étape 3 : Formuler les conditions au niveau des différentes transitions du RdP, en fonction des formules de détection associées aux feuilles de la sous-arborescence (nous faisons ici appel à la fonction formuler-conditions(nœud $n_{i,j}$))

Fin

Fonction formuler-conditions(nœud n_{i,j})

Début

Si nœud n_{i,j} = feuille

Alors formuler-conditions \leftarrow formule de détection

/* cette condition sera associée à la transition en entrée de la structure élémentaire relative à l'action opérateur de correction de l'événement indésirable élémentaire */

Sinon

Selon (branche(nœud n_{i,j})) **faire**

Cas ET : /* composition parallèle */

formuler-conditions \leftarrow \wedge (formuler-conditions (nœud n_{i+1,k}))

pour k = 1 L

(L étant la largeur de la branche)

/* la condition sera associée à la transition en entrée de la composition parallèle */

Cas OU Inclusif : /* composition ou inclusif */

Pour k de 1 à L **faire**

c_k \leftarrow formuler-conditions (nœud n_{i+1,k})

FinPour

/* chaque condition c_k sera associée à une des 2 transitions sorties de P_k et sa négation sera associée à l'autre transition */

formuler-conditions \leftarrow true

Cas Ou Exclusif : /*composition alternative */

Pour k de 1 à L **faire**

c_k \leftarrow formuler-conditions (nœud n_{i+1,k})

FinPour

Pour k de 1 à L **faire**

Affecter comme condition à transition T_k la conjonction :

$\wedge(c_i : i \neq k) \wedge c_k$

FinPour

formuler-conditions \leftarrow true

FinSelon

FinSi

Fin

Le modèle de l'interaction homme-machine déduit pour le cas de cette application est présenté dans la figure 8. La place P0 modélise un état de supervision du système en fonctionnement normal. P1 modélise l'état de dysfonctionnement : "atteinte du seuil max du réservoir" et P2 modélise l'état de dysfonctionnement "température non conforme".

Les transitions T1 et T2 modélisent donc la détection de ces dysfonctionnements moyennant les conditions qui leur sont affectées (c01 et c02) : (1) "atteinte du seuil max du réservoir" (la condition c01 est "CS.maxRes =1"), (2) "température non conforme" (la condition c02 est "Out.tempMel > 35°C"). Le bloc SRP2 représente le sous réseau qui modélise l'interaction homme-machine face au dysfonctionnement 2. L'autre partie du réseau développée à partir de T3 modélise l'interaction pour le cas du dysfonctionnement 1. Ce bloc est déduit de l'AdD conformément à l'algorithme de passage expliqué ci-dessus. Il est construit à partir d'une composition "ou inclusive" de trois sous réseaux, chacun d'eux modélisant l'interaction face à l'une des causes possibles de cet état de dysfonctionnement.

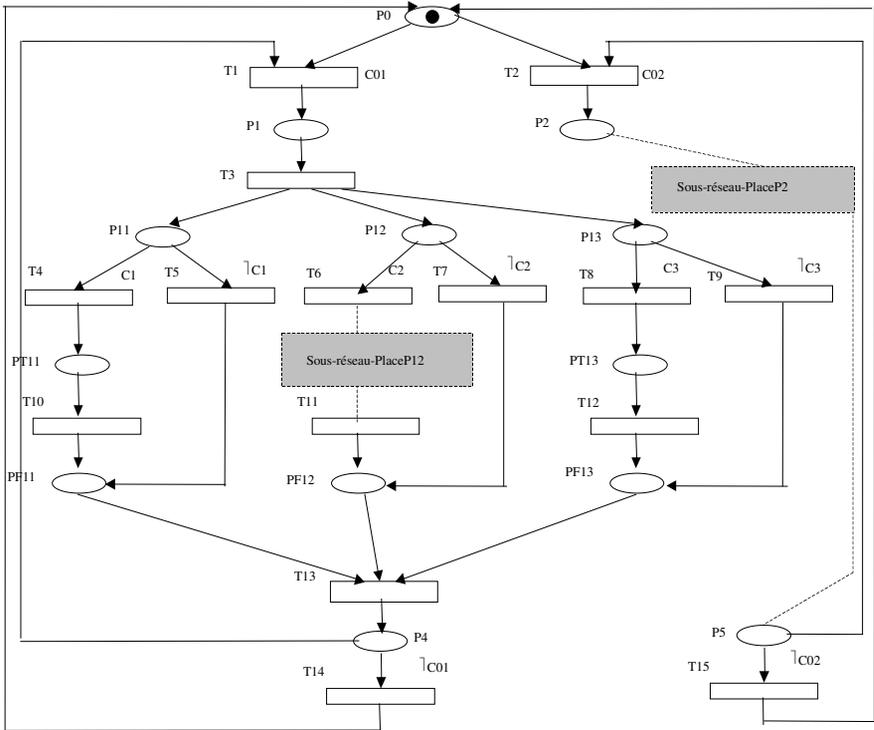


Figure 8. *Modèle de l'interaction homme-machine déduit de l'AdD*

A partir de l'AdD, nous pouvons distinguer trois causes possibles du dysfonctionnement "atteinte seuil max du réservoir" : une défaillance du capteur de mesure, un problème d'évacuation, et un problème de bouchage de conduites du réservoir.

Ces trois causes possibles sont modélisées respectivement par les places P11, P12 et P13. Pour les première et troisième causes, les analyses effectuées par l'AdD renvoient à des événements élémentaires et par la suite à des actions élémentaires de l'opérateur pour changer le capteur défectueux ou réparer les conduites bouchées. Chacune de ces actions est modélisée par une structure élémentaire (bloc [T4, PT11 et T10] pour la première action et bloc [T8, PT13 et T12] pour la deuxième). Pour la cause "problème d'évacuation" modélisée par P12, c'est un événement composé dû à une défaillance du doseur du mélange avec un blocage de la vanne sécurité en état fermée, ou un bouchage de la conduite réservoir-doseur mélange avec un blocage de la vanne sécurité en état fermée, ou un blocage de la vanne cuve en état ouverte avec un blocage de la vanne sécurité en état fermée.

Le sous réseau « Sous-réseau-PlaceP12 » (figure 9) modélise cette éventualité. Pour chacun de ces trois cas possibles, une action élémentaire de l'opérateur est nécessaire. Elle consisterait à débloquer la vanne de sécurité de l'état fermée et changer le doseur, ou déboucher la conduite réservoir-doseur ou la changer ou alors débloquer la vanne cuve de l'état ouverte.

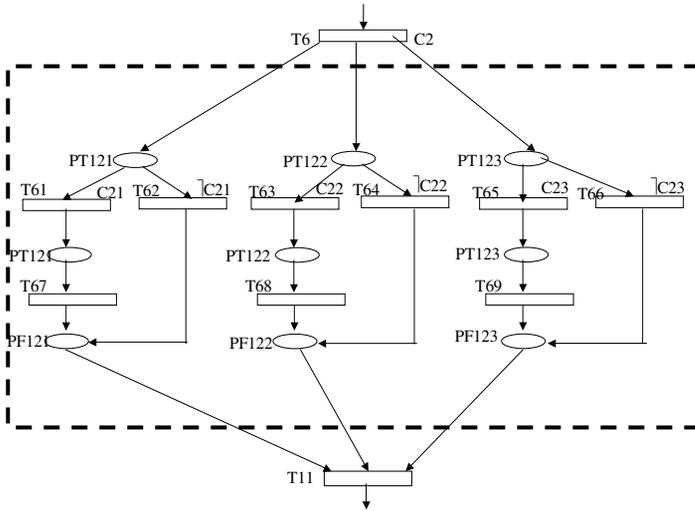


Figure 9. *Le sous réseau « sous-réseau-PlaceP12 »*

4.2 Dédution des BIO

Rappelons d'abord que l'opérateur a besoin d'identifier instantanément l'état de fonctionnement du procédé qu'il contrôle. Cette information lui est transmise à travers différents objets au niveau de l'interface (messages, valeurs, graphiques, alarmes, etc.). Ces objets sont, en fait, reliés à des variables d'état du système. Il est donc nécessaire d'identifier l'ensemble des variables d'information appropriées à chaque état du système. Ces variables d'information découlent directement de l'analyse du SHM menée auparavant. De plus, et dans le but d'accomplir ses tâches de correction, l'opérateur a à intervenir et agir à travers des variables de commande suite à l'apparition des dysfonctionnements. L'interface doit donc présenter dans de telles situations, l'ensemble des objets de commande (appelés aussi objets d'interaction en IHM) qui permettent à l'opérateur d'exécuter ses actions et commander le processus. L'ensemble de ces variables d'information et de commande constitue les BIO relatifs à chaque état. Ces variables présentées à l'opérateur lui permettent d'accomplir ses tâches de correction.

Nous expliquons ici la manière de déduire ces BIO à travers les précédentes analyses et le modèle de l'interaction homme-machine établi au niveau de l'étape précédente de la démarche.

Considérons principalement les places PT11, PT13, PT121, PT122 et PT123, qui représentent des états relatifs à la présence des événements particuliers en attente de l'accomplissement des tâches de correction. La liste des BIO associés à chacun de ces états est :

- Pour la place PT11, l'opérateur a besoin de savoir les seuils min et max et le volume actuel ; ceci est décrit par l'ensemble des variables : x , n , $Cs.MaxRes$, $Out.VolRes$.
- Pour PT13, il a besoin de savoir les seuils min et max, le volume actuel et les états des vannes. Ceci est décrit par : x , n , $Cs.MaxRes$, $Out.VolRes$.

- Pour la PT121, il a besoin de savoir les seuils min et max et le volume actuel, mais aussi les états des vannes et la quantité dosée et il aura à agir sur les variables de commande de dosage et d'ouverture de la vanne réservoir. Ceci est décrit par : x , n , Cs.MaxRes, Out.VolRes, Out.VanSecurite, Out.VanCuve, Out.QteMel, Cd.DebDosMel, Cd.OuvVanRes.
- Pour PT123, l'ensemble des BIO est décrit par : x , n , Cs.MaxRes, Out.VolRes, Out.VanSecurite, Cd.OuvVanRes.
- Pour PT122, l'ensemble des BIO est décrit par : x , n , Cs.MaxRes, Out.VolRes, Out.VanSecurite, Cd.OuvVanRes, Cd.OuvVanCuve.

Bien entendu, dans la réalité, ces choix doivent être validés avec les opérateurs humains et les spécialistes de l'installation industrielle.

Une fois les BIO identifiés (c'est-à-dire l'ensemble des variables d'information et de commande formé), l'étape suivante consiste à spécifier les objets graphiques relatifs à ces BIO :

- à chaque variable d'information, on associe un objet d'information (appelés aussi objets de présentation en IHM),
- à chaque variable de commande, on associe un objet de commande (ou d'interaction).

Pour notre cas d'étude, il est pertinent de générer trois vues principales :

- une vue de supervision pour l'état de fonctionnement normal,
- une vue de contrôle et commande pour l'état de dysfonctionnement 1, et
- une vue de contrôle et commande pour l'état de dysfonctionnement 2.

Pour chacune de ces vues, on y trouvera des objets graphiques représentant les BIO associés aux états du modèle de l'interaction couverts par ces vues. Ainsi, aux objets d'information et de consigne, il s'agira d'associer des zones de texte et/ou des barres-graphes principalement pour les informations sur le volume du réservoir. Pour les variables de commande, des boutons de commande pourraient être attribués. Rappelons que ce travail important de spécification et de conception ne fait pas l'objet de cet article ; mais on peut de nouveau souligner qu'un grand intérêt de la méthode proposée est qu'on dispose pour ce travail d'éléments provenant explicitement, aussi bien des situations normales que des situations de dysfonctionnement, au contraire des méthodes classiques en conception d'IHM.

5 Discussion

Dans cet article nous nous sommes intéressés particulièrement à deux aspects très importants voir cruciaux rentrant dans le processus global de conception et de génération des IHM, à savoir (i) la prise en considération de l'analyse des dysfonctionnements et (ii) l'aide à l'identification des Besoins Informationnels des opérateurs (BIO). Ce sont deux aspects qui, en dépit d'une littérature abondante de manière générale en interaction homme-machine, n'ont pas bénéficié, à notre avis, de suffisamment de réflexion ni d'expérimentation, ou tout simplement de description, vis-à-vis des applications de type système industriel complexe.

D'un coté, il n'est plus à démontrer aujourd'hui (certaines études ayant été menées il y a plus de trente ans), qu'au moment où apparaît un dysfonctionnement l'opérateur peut se retrouver dans une situation de stress, de surcharge mentale, et présenter, dès lors, des besoins particuliers en information et en assistance. Or force est de constater, que dans la majorité des approches étudiées, l'analyse du procédé en dysfonctionnement n'étant pas prise en considération, il devient difficile voire impossible de répondre systématiquement et de façon satisfaisante à ses besoins particuliers.

D'un autre coté, dans le souci de proposer une approche globale supportée à terme par des outils informatisés, il devient impératif de veiller à ce que toutes les étapes de l'approche en question « s'emboîtent », en matière d'entrée, de sortie et de traitement, à la manière d'un « LEGO® » pour former un tout intégré offrant des potentialités en matière d'automatisation. Ici également, la littérature, à notre connaissance, ne mentionne pas d'approches offrant des réponses claires quant à l'informatisation du passage délicat de l'analyse du système homme-machine à la déduction des besoins informationnels des opérateurs.

Les premiers éléments de réponse que nous apportons aux deux points susmentionnés, tirent, selon notre humble avis, leurs forces du fait qu'elles s'appuient sur des méthodes et des outils puissants (tout en étant basés sur des concepts de modélisation relativement simples) ayant déjà fait leurs preuves dans plus d'un domaine et dans le cadre de nombreux projets. Nous parlons de méthodes d'analyse et conception comme SADT, AMDEC ou AdD ou encore d'outils de modélisation formels comme les RdP. Bien sûr, les choix opérés sont à discuter et certains verront dans d'autres méthodes et outils d'éventuels meilleurs candidats. Seulement, au delà de la qualité intrinsèque que présenterait telle méthode ou tel outil, notre objectif, rappelons-le, était d'aboutir à une approche globale, homogène et offrant des étapes s'enchaînant les unes aux autres et permettant l'informatisation.

Le travail de recherche, de réflexion et d'ingénierie mené autour de ces méthodes et outils, dans l'objectif de proposer une méthodologie globale « intégrée » de conception et de génération d'IHM, a abouti à l'approche présentée et illustrée dans cet article.

En terme d'analyse du système en mode de dysfonctionnement, nous avons constaté le degré de rigueur et la potentialité en terme de quasi exhaustivité de l'analyse qu'offre la combinaison des deux méthodes bien connues AMDEC/AdD. Nous avons souligné plus haut la difficulté rencontrée lors de l'application de cette analyse ; mais rappelons que dans un processus pluridisciplinaire, comme celui de la conception des IHM dans les systèmes industriels complexes, une telle étape serait certainement affectée à des spécialistes en fiabilité qui n'auraient en principe que peu de gêne dans sa réalisation. De plus, selon nous, un algorithme du type de celui dénommé « Déduire-modèle-RdP », présenté précédemment, contribue à assurer ensuite le passage « automatisé » d'arbres AdD vers des RdPI modélisant certains aspects de l'interaction homme-machine ; cet algorithme est loin d'être parfait, il pourrait être étendu, adapté, mais il est représentatif d'une voie à suivre.

En terme de modélisation du système homme-machine, le recours aux RdP (dans notre cas les RdP interprétés) permet potentiellement de gagner en rigueur en termes de modélisation et de validation des modèles obtenus (dans la lignée de travaux déjà menés en IHM en exploitant les RdP, cf. les travaux menés par Palanque et ses collègues (Palanque *et al.*, 1995)). Le recours aux RdPI a permis aussi et surtout de faciliter (de quasiment automatiser, même si pour notre étude

de cas ce passage a été simulé manuellement) l'étape de déduction en temps réel des BIO (depuis les RdPI) et ce selon différents contextes de fonctionnement (normal ou anormal).

Ainsi, pour revenir à notre réflexion de départ, en cas de dysfonctionnement, si les résultats des premières étapes d'analyse et de modélisation ont été judicieusement exploités lors de la conception de l'IHM, l'opérateur humain devrait disposer quasiment instantanément d'une IHM offrant les informations nécessaires et pertinentes par rapport au contexte de fonctionnement en question, ce qui devrait faciliter ainsi grandement sa tâche.

6 Conclusion

Nous pensons que dans les méthodes de conception d'IHM, il faudrait mettre plus en avant l'analyse et la modélisation des situations anormales, aspect indispensable dans certains domaines d'application critiques et dangereux pour la sécurité et/ou l'environnement comme c'est le cas des applications industrielles complexes (chimie, transport...). Cet article a concerné une proposition d'une combinaison de méthodes parmi N possibles, assurant une analyse du système homme-machine dans les différentes situations de fonctionnement possibles (normales et anormales) dans le but de faciliter la déduction des besoins informationnels des opérateurs afin de concevoir l'interface appropriée.

Nos travaux de recherche se poursuivent dans cet axe de recherche et de nombreuses perspectives sont d'ores et déjà envisagées.

Nous projetons, tout d'abord, d'étudier encore plus l'articulation entre les différentes méthodes proposées dans cette démarche et d'améliorer les algorithmes actuels en étudiant des outils automatiques ou semi-automatiques d'aide à la détermination des BIO.

Il serait aussi intéressant d'appliquer la démarche sur des cas d'application de plus en plus complexes, impliquant des tâches collectives, des processus parallèles, des aspects multi-utilisateurs spécialisés possédant nécessairement différents points de vue (Enselme et David, 1994), etc. Dans ce cadre, nous nous intéressons aux potentialités des services web pour la conception de nouvelles applications de supervision destinées à des utilisateurs nomades en lien ou non avec les opérateurs de la salle de contrôle (Idoughi et Kolski, 2006a, 2006b). Dans le prolongement de ces travaux, la prise en compte de la plasticité des interfaces devrait aussi nécessiter une réflexion particulière (Thévenin et Coutaz, 1999 ; Calvary *et al.*, 2005).

Enfin, notre souhait est de généraliser la démarche à d'autres domaines d'application et d'adapter les choix des méthodes et algorithmes en fonction des caractéristiques du domaine.

Remerciements

Les auteurs remercient le Prof. M. Moalla pour ses remarques relatives à l'étude de cas, de même que les trois relecteurs anonymes de la revue RIHM pour leurs nombreuses remarques constructives et pertinentes. Le second auteur remercie également le FEDER et la région Nord-Pas de Calais pour leur soutien financier (projets TAC MIAOU et EUCUE).

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, Valenciennes, France.

Abed, M. (2001). *Méthodes et modèles formels et semi-formels pour la conception et l'évaluation des systèmes homme-machine*. Mémoire d'HDR, Valenciennes.

Abed, M., Bernard, J.M., Angué, J.C. (1992). A Process Method for Specification and the Design of Man-Machine Interface. *14th Annual international conference IEEE Engineering in medicine and biology society*, Lyon, France, November 2-4.

Abed, M., Ezzedine, H., Angué, J.C. (1995). Méthodologie d'analyse et de modélisation de tâches d'interaction Homme-Machine avec des outils de spécification. *Revue Européenne Diagnostic et Sûreté de fonctionnement*, vol. 5 n° 2, pp.159-180.

Abed, M. Ezzedine, H. (1998). Vers une démarche intégrée de conception-évaluation des systèmes Homme-Machine. *Journal of Decision Systems*, Vol. 7, pp. 147-175.

Abrial, J. R. (1996). *The B Book : Assigning Programs to Meanings*. Cambridge University Press.

Aït-Ameur, Y., Girard, P., Jambon, F. (1998). A Uniform approach for the Specification and Design of Interactive Systems: the B method. *Eurographics Workshop on Design, Specification, and Verification of Interactive Systems (DSV-IS'98)*, vol. Proceedings, P. Markopoulos and P. Johnson (Eds.), Abingdon, UK, 1998, pp. 333-352.

Balbo S. (1994). *Un pas vers l'évaluation automatique des interfaces homme-machine*. Thèse en informatique, Université Joseph Fourier, Grenoble 1, septembre.

Barthet M. F. (1988). *Logiciels Interactifs et Ergonomie – Modèles et méthodes de conception*. Dunod Informatique.

Benaïssa, M. (1993). *Une démarche de conception, réalisation et évaluation d'IHM : application au projet ferroviaire ASTREE*. Thèse de Doctorat en Automatique Humaine, Université de Valenciennes et du Hainaut-Cambrésis, décembre.

Bereikdar, A. (2004). *A methodology for automating web usability and accessibility evaluation by guideline*. Thèse de doctorat, Facultés Universitaires Notre-Dame de la Paix, Namur, Belgique.

Berry, G., Gonthier, G. (1988). *The Esterel Synchronous Programming Language : Design, Semantics, Implementation*. Technical report 842, INRIA.

Bodart, F., Hennebert, A.-M., Leheureux, J.-M., Vanderdonckt, J. (1995). A model-based approach to presentation: a continuum from task analysis to

prototype. In *Proceedings of Interactive Systems: Design, Specification and Verification*, Springer-Verlag, Berlin, p. 77-94.

Bolognesi, T., Brinksma, E. (1989). *The formal description technique LOTOS, Introduction to the ISO specification language LOTOS*. Elsevier Science Publishers.

Brun, P. (1998). *XTL : une logique temporelle pour la spécification formelle des systèmes interactifs*. Thèse en informatique, Université Paris XI, Orsay, Septembre.

Calvary, G., Daasi, O., Demeure, A., Coutaz, J. (2005). Des Widgets aux Comets pour la Plasticité des Systèmes Interactifs. *Revue d'Interaction Homme-Machine (RIHM)*, volume 6, N° 1, pp. 33-54.

Card, S.K., Moran, T.P and Newell, A. (1983). *The psychology of human-computer interaction*. Hillsdale, NJ. Erlbaum, 1983.

Cardelli, L., Pike, R. (1985). Squeak : A language for communicating with mice. *Computer graphics*, vol. 19. July.

D'Ausbourg, B., Durrieu, G. and Rocher, P. (1996). Deriving a formal model of interactive system from its UIL description in order to verify and to test its behaviour. In *Proceedings of DSV-IS'96*, Springer verlag, pp. 104-122.

Dang, W. (1988). Formal specification of interactive languages using Definite Clause Grammers. *Réunion internationale sur l'implantation de langage de programmation et la programmation logique*, Orléans, Mars.

Daniellou, F. (1986). *L'opérateur, la vanne, l'écran : l'ergonomie dans les salles de contrôle*. ANACT Collection Outils et Méthodes, Montrouge.

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.

De Keyser, V. (1988). De la contingence à la complexité : l'évolution des idées dans l'étude des processus continus. *Le Travail Humain*, 51 (1), pp. 1-18.

De Nicola, R. (1991). Action and State-based Logics for Process Algebras. *CONCUR '91, 2nd International Conference on Concurrency Theory, Amsterdam, The Netherlands, August 26-29, 1991, Proceedings*. Lecture Notes in Computer Science 527 Springer, ISBN 3-540-54430-5.

De Rosis, P. (1998). Formal Description and Evaluation of User Adapted Interfaces. *International Journal of Human-Computer Studies*, vol. 49, 1998, pp. 95-120.

Diaper, D., Stanton, N. (2004). *The handbook of task analysis for human-computer interaction*. Lawrence Erlbaum Associates.

Enselme, Y., David, B. (1994). Système de contrôle de procédés : principe d'une démarche de conception d'IHM. *Journées internationales sur les nouveautés en Génie Logiciel, JINGL*, 13-15 décembre, Paris.

- Ezzedine, H., Kolski, C. (2004). Démarche d'évaluation d'IHM dans les systèmes complexes, application à un poste de supervision du trafic ferroviaire. *Revue d'Interaction Homme-Machine*, vol. 5, pp. 91-122.
- Fadier, E. (1990). Fiabilité humaine : méthodes d'analyse et domaines d'application. In J. Leplat et G. De Terssac (Eds.), *Les facteurs humains de la fiabilité dans les systèmes complexes*. Octarés, Marseille.
- Favre, J.M., Estublier, J., Blay-Fornarino, M. (2006). *L'ingénierie dirigée par les modèles, au-delà du MDA*. Hermes, Paris.
- Fekete, J.D., Girard, P. (2001). *Environnements de développement de systèmes interactifs*. In Kolski C. (dir.), *Environnements évolués et évaluation de l'IHM, Interaction homme-machine pour les SI 2*, pp. 23-52, Hermes, Paris, 2001.
- Flechchia, M., Bergeron, R. D. (1987). Specifying complex dialogues in ALGEO. *Proceeding CHI and graphics interface, ACM*, New York.
- Gaudel M. C. (1984). A first introduction to Plus, *Alvey Workshop on Formal Specifications*. Swindon (GB), October.
- Gilmore, W.E., Gertman, D.I., Blackman, H.S. (1989). *User-computer interface in process control, a Human Factors Engineering Handbook*. Academic Press.
- Goguen, J.A., Malcolm, G. (1996). *Algebraic Semantics of Imperative Programs*. The MIT Press.
- Gomes, L., Barros, J.P., Coasta, A. (2001). Man-machine interface for real-time telecontrol based on Petri nets specification. In T. Bahill, F.Y. Wand (Eds.), *IEEE SMC 2001 Conference Proceedings (e-Systems, e-Man and e-Cybernetics)*, Arizona, USA: IEEE Press, pp. 1565-1570.
- Guittet, L., Girard, P., Pierra, G. (1997). *Dialogue verification using the EXPRESS language*. Eurographics Workshop on Design, Specification and Verification of Interactive Systems (DSV-IS'97), vol. Conference proceedings, edited by Harrison M.D. and Torres J.C. (Eds.), Granada, Spain, 1997, pp. 457-471.
- Halbwachs, N., Caspi, P., Raymond, P., Pilaud, D. (1991). Programmation et vérification des systèmes réactifs : le langage Lustre. *Technique et Science Informatiques (TSI)*, vol 10, n°2, 1991, pp. 139-158.
- Hatley, D.J., Pirbai, I.A. (1991). *Stratégies de spécification des systèmes temps réel*. Editions Masson, Paris.
- Hartson, H.R., Gray, P.D. (1992). Temporal aspects of tasks in the User Action Notation. *Human-Computer Interaction*, vol. 7, pp. 1-45.
- Herrmann, M., Hill, R. D. (1989). Abstraction and declarativeness in user interface development: the methodological basis of the Composite Object Architecture, *Proceedings IFIP' 89*, G.X. Ritter (ed), Elsevier Publishers B. V., North Holland.
- Hill, R.D. (1986). Supporting Concurrency. Communication and Synchronisation in Human-Computer Interaction - the Sassafras UIMS. *ACM transaction on Graphics*, vol. 5, n°3, p. 179-210, July.

- Hix, D., Hartson, H. R. (1993). *Developing user interface: Ensuring usability through Product process*. John Wiley Sons, New York.
- Hoc, J.M., Amalberti, R. (1995). Diagnosis: some theoretical questions raised by applied research. *Current Psychology of Cognition*, 14 (1), pp. 73-101.
- Hoc, J.M. (1996). *Supervision et contrôle de processus, la cognition en situation dynamique*. Grenoble, Presses Universitaires de Grenoble, Grenoble.
- Idoughi, D., Kolski, C. (2006a) Towards new Web Services based Supervisory Systems dedicated to Nomadic Operators. In *Proceedings of the 25th Edition of EAM'06, European Annual Conference on Human Decision-Making and Manual Control* (September 27-29, 2006, Valenciennes, France), Presses Universitaires de Valenciennes.
- Idoughi, D., Kolski, C. (2006b). Approches orientées services web de l'IHM de supervision: nouvelles solutions technologiques pour les ingénieurs et nouvelles problématiques pour les ergonomes ? In *ErgoIA 2006, "L'humain comme facteur de performance des systèmes complexes"* (Biarritz, France, 11-13 Octobre 2006), ESTIA & ESTIA. Innovation, Biarritz, pp. 111-118.
- IGL Technology (1989). *SADT, un langage pour communiquer*. Paris, Eyrolles.
- Jacob, R.J.K. (1985). A state transition diagram language for visual programming. *IEEE Computer*, vol. 18 (8), August 1985, pp. 51-59.
- Jacobson I., Booch G., Rumbaugh I. (2000). *Le processus unifié de développement logiciel*. Eyrolles.
- Jambon F., Brun Ph., Ait-Ameur Y. (2001). Spécification des systèmes interactifs. Dans Kolski C. (Ed.), *Analyse et Conception de l'IHM*. In *Interaction Homme-Machine pour les SI, Volume 1*. pp. 175-206. Paris: Éditions Hermes.
- Jambon, F., Ait-Ameur, Y., Breholée, B., Girard, P., Guittet, L. (2004). Formal Verification and Validation of Interactive Systems Specifications. In *Johnson, C., Palanque, P. (Eds.). Human Error, Safety and Systems Development 2004. Proceedings of HESSD 2004*.
- Johnson, C., Palanque, P. (Eds.) (2004). *Human Error, Safety and Systems Development 2004. Proceedings of HESSD 2004*. ISBN 1-4030-8152-9.
- Jones, C.B. (1980). *Software Development – A Rigorous Approach*. Prentice Hall.
- Khelil, N. (2001). *Modélisation sûre des interactions dans les interfaces homme-machine*. Mémoire de DEA, Faculté des Sciences de Tunis, octobre.
- Kolski, C. (1997). *Interfaces homme-machine, application aux systèmes industriels complexes* (2^{ème} édition). Éditions Hermès, Paris, 1997.
- Lambert, M. (1999). *Conception centrée sur l'homme d'un système de supervision avancé - application à un procédé de retraitement de combustibles nucléaires*. Thèse de l'Université de Valenciennes et du Hainaut-Cambrésis, septembre.

- Laprie, J.C. et collègues (1995). *Guide de la sûreté de fonctionnement*. Cépaduès Editions, Toulouse.
- Lejon, J.C., (1991). *L'évolution de la conduite sur SNCC : l'ergonomie des systèmes numériques de contrôle commande*. Editions ANACT, 1991.
- Lepreux, S., Abed, M., Kolski, C. (2003). A human-centred methodology applied to decision support system design and evaluation in a railway network context. *Cognition Technology and Work*, 5, pp. 248-271.
- Lim, K.Y., Long, J. (1994). *The Muse Method for Usability Engineering*, Cambridge University Press, Cambridge.
- Lind, M. (1990). *Representing goals and functions of complex systems: an introduction to Multilevel Flow Modelling*. RISO Laboratory, Denmark, Ref: 90-D-381. ISBN 87-87950-52-9, november.
- Mahfoudhi, A. (1997). *TOOD : Une méthodologie de description orientée objet des tâches utilisateur pour la spécification et la conception des IHM : Application au contrôle du trafic aérien*. Thèse de doctorat, Université de Valenciennes et du Hainaut-Cambrésis, France, 1997.
- Mariage, C. (2005). *MetroWeb : logiciel de support à l'évaluation de la qualité ergonomique des sites Web*. Thèse de doctorat, Université Catholique de Louvain, Belgique.
- Millot, P. (1988). *Supervision des procédés automatisés et ergonomie*, Éditions Hermes, Paris.
- Millot, P, Debernard, S. (1993). Men-machines cooperative organizations: methodological and practical attempts in air traffic control. In *Proceedings IEEE SMC*, Le Touquet, France, October 17-20.
- Moalla, M. (1985). Réseaux de Petri interprétés et Grafcet. *TSI*, 4 (1), pp. 17-30.
- Moray, J. (1997). Human factors in process control. In *Handbook of human factors and ergonomics*. G. Salvendy (Ed.), John Wiley & Sons, pp. 1944-1971.
- Moussa, F., Kolski, C., Riahi, M. (2000). A model based approach to semi-automated user interface generation for process control interactive applications. In *Interacting with Computers*, 12 (3), special issue Tools for Working With Guidelines (Part 2), pp. 279-292.
- Moussa, F., Riahi, M., Kolski, C., Moalla, M. (2002). Interpreted Petri Nets used for Human-Machine Dialogue Specification in Process Control: principles and application to the Ergo-Conceptor+ tool. *Integrated Computer-Aided Engineering*, 9, 87-98.
- Moussa, F., Kolski, C., Riahi, M. (2005). De la modélisation des dysfonctionnements d'un système complexe à la déduction des besoins informationnels des utilisateurs : une transition difficile en IHM. In *Proceedings of IHM 2005, International Conference Proceedings Series*, ACM Press, Toulouse, pp. 75-82, septembre.

- Moussa, F. (2005). *Vers une méthodologie globale de conception et de génération semi-automatique des IHM pour les systèmes critiques*. Mémoire d'HU (HDR), Tunis, Tunisie, Juin.
- Nanci, D., Espinasse, B. (2001). *Ingénierie des systèmes d'information : MERISE, 2ème génération*. Vuibert, Paris.
- Navarre, D. (2001). *Contribution à l'ingénierie en Interaction Homme-Machine, Une technique de description formelle et un environnement pour une modélisation et une exploitation synergiques des tâches et du système*. Thèse de l'Université de Toulouse 1, juillet.
- 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 N.J., Elbraum.
- Olsen, D. R., Dempsey, E. P. (1983). Syngraph : A graphical user interface generator. *ACM Transaction on Graphics*, July.
- Palanque, P., Bastide, R. (1995). Spécifications formelles pour l'ingénierie des interfaces homme-machine. *Technique et Science Informatiques (TSI)*, 14 (4), pp. 473-500.
- Palanque, P., Bastide, R., Senges, V. (1995). Validating interactive system design through the verification of formal task and system models. *6th IFIP Working Conference on Engineering for Human-Computer Interaction (EHCI'95)*, Grand Targhee Resort, Wyoming, U.S.A., 14-18 August.
- Palanque, Ph., Bastide, R. (1996). A design life-cycle for the formal design of interactive systems. In *Proceedings FAHCI'96BCS-FACS Workshop on the Formal Aspects of the Human-Computer Interface*. Sheffield, U.K., September.
- Palanque, Ph, Bastide, R. (1997). Synergistic modeling of tasks, system and users using formal specification techniques. *Interacting With Computers*, 9, 12, pp. 129-153.
- Pelleaumail, P. (1987). *La méthode Axial*. Editions d'organisation, Paris.
- Peterson, J.L. (1981). *Petri net theory and the modelling of systems*. Englewood cliffs, NJ: Prentice-Hall.
- Petri, C. (1962). *Kommunikation mit Automaten*. Ph. D. Dissertation, University of Bonn.
- RAISE (1992). *The RAISE Language Group. The RAISE Specification Language*, Prentice Hall.
- 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.
- Rasmussen, J. (1986). *Information processing and human-machine interaction, an approach to cognitive engineering*. Elsevier Science Publishing.
- Reason, J. (1990). *Human error*. Cambridge University Press, Cambridge.

- Riahi, M. (2004). *Contribution à l'élaboration d'une méthodologie de spécification, de vérification et de génération semi-automatique d'interfaces homme-machine : application à l'outil Ergo-Conceptor+*. Thèse de Doctorat, Valenciennes, septembre.
- Riahi, M., Moussa, F., Kolski, C., Moalla, M. (2000). Use of interpreted Petri nets for human-machine dialogue specification in process control. In *Proceedings ACIDCA'2000*, 22-24 March, Monastir, Tunisia.
- Ridoux, M. (1999). AMDEC- Moyen. *Techniques de l'ingénieur, L'entreprise industrielle*, vol AG1, AG-4-220, Avril, 1999.
- Sinclair, I.A.C., Sell, R.G., Beishon, R.J., Bainbridge L. (1965). Ergonomic study of L.D. Waste-heat boiler control room. *Journal Iron and steel inst.*, 204, pp. 434-442.
- Spivey, J. M. (1989). *The Z notation : a reference manual*. Prentice-Hall, Inc., Upper Saddle River, NJ, 1989.
- Stewart, T., Travis, D. (2003). Guidelines, standards, and guide styles. In Jacko J.A., Sears A. (Eds.), *The human-computer interaction handbook: fundamentals, evolving technologies and emerging applications*, Lawrence Erlbaum Associates, pp. 991-1005.
- Szekely, P. (1996). Retrospective and Challenges for Model-Based Interface Development, in: Bodart, F., Vanderdonckt, J. (eds.). In *Proceedings of the Eurographics Workshop, Design, Specification and Verification of Interactive Systems '96*, pp. 1-27, Springer.
- Tabary, D. and Abed, M. (1998). TOOD : An object-oriented methodology for describing user task in interface design and specification - An application to air traffic control. *La Lettre de l'Intelligence Artificielle*, vol 134-135-136, pp. 107-114.
- Taborin, V. (1989). *Coopération 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.
- Tardieu, H., Rochfeld, O., Colleti, R. (1991). *La méthode Merise, principes et outils, 2ème édition*. Editions d'Organisation (tome 1), Paris.
- Thévenin, D., Coutaz, J. (1999). Plasticity of user interfaces: framework and research agenda. *Proceedings of Interact'99 seventh IFIP Conference on Human-Computer Interaction*, Edinburgh, Scotland.
- Turner, K.J. (1993). *Using formal description techniques: An introduction to Estelle, Lotos and SDL*. Edition John.
- Vanderdonckt, J. (1994). *Guide ergonomique des interfaces homme-machine*. Presses Universitaires de Namur, Belgium.
- Vanderdonckt, J. (1999). Development milestones towards a tool for working with guidelines. *Interacting With Computers*, 12 (2), special issue Tools for Working With Guidelines (Part 1), pp. 81-118.

Vanderdonckt, J., Farenc, C. (Eds.) (2000). *Tools for Working With Guidelines TFWWG'2000*. Springer, London.

Villemeur, A. (1992). *Reliability, availability, maintainability and safety assessment* (vol.1 & 2). John Wiley & Sons.

Woods, W. A. (1980). Cascaded ATN grammars. *In Computational Linguistics*. Volume 6, Issue 1, January-March,1980, pp. 1-12, MIT Press, Cambridge, MA, USA.

Zwengelstein G. (1999). Sûreté de fonctionnement des systèmes industriels complexes. *Techniques de l'ingénieur, Traité informatique industrielle*, vol S3, S8-250, Septembre, 1999.