

# Un cadre d'études pour une approche temps-réel de l'ingénierie des IHM à contraintes temporelles

*Francis JAMBON, Dominique GENIET*

LISI/ENSMA  
BP 40109, 86961 Furtutoscope cedex  
{Francis.Jambon, Dominique.Geniet}@ensma.fr  
<http://www.lisi.ensma.fr/ihm/>

*Philippe BRUN*

IHM CONSEIL  
18, av. du Général de Gaulle  
92130 Vanves  
[ph.brun@libertysurf.fr](mailto:ph.brun@libertysurf.fr)

## RÉSUMÉ

Cet article propose un cadre d'études pour une nouvelle approche du processus d'ingénierie des Interfaces Homme-Machine devant respecter des contraintes temporelles. Pour cette catégorie d'IHM, le respect des temps de réaction est un paramètre essentiel de leur utilisabilité, et plus encore pour certains domaines d'application, de la sécurité du système dans son ensemble. Après un panorama d'exemples tirés de divers domaines d'application, nous montrons que les solutions actuellement adoptées visent à obtenir une marge de sécurité raisonnable vis-à-vis des fautes temporelles, mais ne garantissent en aucun cas une sécurité totale. Nous définissons dans cet article la caractéristique d'Interface Homme-Machine Temps-Réel (IHM-TR). De plus, nous montrons que les propriétés de ces IHM-TR peuvent être assurées grâce à une approche temps-réel basée sur la spécification d'échéances et l'utilisation de méthodes de validation temporelles. En corollaire, nous proposons deux modélisations distinctes des systèmes interactifs selon qu'ils s'apparentent à des systèmes de pilotage ou de supervision.

**MOTS CLÉS :** Ingénierie des IHM, contraintes temporelles, validation temps-réel, systèmes à risques.

## INTRODUCTION

On assiste, depuis une dizaine d'années, à une utilisation extensive de l'informatique dans des systèmes interactifs où la maîtrise des temps de réaction est un élément clé de leur utilisabilité [3, 5, 6]. Dans ces systèmes, les contraintes temporelles liées à la transmission des informations entre l'utilisateur et le système interactif influent directement sur la qualité, l'efficacité, et même parfois la sûreté du système homme-machine dans son ensemble.

Plus précisément, les enjeux de la maîtrise des contraintes temporelles varient de manière significative selon les systèmes et leurs domaines d'application. Elles peuvent ainsi aller d'un simple inconfort pour l'utilisateur jusqu'à la destruction complète du système. Nous décrivons ci-après des exemples de systèmes interactifs où la maîtrise des contraintes temporelles est un élément primordial de leur utilisabilité.

## Exemples

Les interfaces multimédia et plus encore les interfaces multimodales diffusent vers l'utilisateur, ou acquièrent en provenance de celui-ci, une grande quantité de données qui doivent être traitées dans des délais contraints :

- L'application MATIS de réservation de billets d'avion proposé par L. Nigay [8] utilise la reconnaissance vocale pour renseigner les requêtes à la base de données des vols disponibles. Lorsque l'utilisateur utilise la voix, celui-ci peut également compléter, par d'autres moyens d'interaction, d'autres paramètres de la requête. Le système doit ainsi être capable d'effectuer la reconnaissance vocale quelle soient les actions de l'utilisateur sur le système, et donc la charge de la machine hôte. La solution retenue repose sur l'utilisation d'un processeur supplémentaire dédié au traitement du signal (DSP) et disposant d'une entrée de signal indépendante du bus de données du processeur principal.

Pour ce système interactif, le non respect de contraintes temporelles a pour conséquence une baisse de l'utilisabilité. Cependant, pour certaines classes de systèmes, un non respect de ces mêmes contraintes peut avoir des conséquences plus sérieuses, c'est le cas en particulier des systèmes à risques :

- Deux exemplaires de l'avion de chasse suédois "Gripen" se sont successivement écrasés [7]. La première fois, le système de contrôle de vol électronique a réagi trop tard aux ordres du pilote, lequel a perdu le contrôle de l'appareil. La seconde, ce même système a été saturé lors d'une manœuvre trop brusque du pilote sur les commandes et n'en a averti celui-ci que plusieurs secondes après, c'est-à-dire beaucoup trop tard. Dans cet exemple, le système de contrôle n'a pas été capable, dans certains cas particuliers, de réagir à temps pour maintenir la pilotabilité de l'appareil, avec pour conséquence une perte matérielle du système (le pilote s'étant fort heureusement éjecté à temps !).
- Le domaine de la supervision des lignes ferroviaires TGV, notamment pour la gestion de l'énergie et la signalisation, fait appel à de très nombreuses sources

d'information. Il n'est pas possible de temporiser ces notifications du terrain étant donné que le flux de données est constant et important (plusieurs milliers de données élémentaires) et qu'un retard dans l'affichage peut amener le contrôleur de signalisation à réagir trop tard, en cas d'alarme notamment.

Dans de tels systèmes, la solution retenue consiste à remonter les informations terrain au niveau de l'interface de supervision uniquement si les données ont suffisamment variées par rapport à l'état précédent : si le capteur ou l'automate envoie des données de type numérique, on ne prend en compte que les basculements d'états et si il envoie des données analogiques, on fixe un seuil minimum pour prendre en compte la nouvelle valeur.

- Dans le domaine du contrôle aérien, les consoles de visualisation doivent afficher de l'ordre de quelques milliers d'objets graphiques sur des écrans de 2000 x 2000 pixels. Cet affichage doit s'effectuer toutes les cinq secondes en prenant en compte l'augmentation possible du nombre de symboles (plus d'avions et/ou d'informations) mais aussi le temps nécessaire au traitement des interactions du contrôleur avec le système. Actuellement ces systèmes fonctionnent avec des solutions adaptées au cas par cas, en particulier pour optimiser le fonctionnement de la boîte à outils graphique utilisée.

### Solutions actuelles

Les concepteurs de systèmes interactifs font aujourd'hui appel à des solutions ad-hoc. Celles-ci permettent d'estimer, avec une marge de sécurité importante, que le système respectera les contraintes temporelles dans un certain nombre de situations préalablement identifiées. Les solutions utilisées font appel à ces deux approches :

- Un *surdimensionnement* des systèmes de traitement est souvent utilisé, par exemple par l'ajout d'un processeur supplémentaire ou par l'augmentation de la puissance de traitement du système. Cette solution est coûteuse car elle sous-utilise par principe les ressources disponibles.
- Une *optimisation* des processus de traitement est également utilisée, par exemple en re-concevant les bibliothèques graphiques ou en programmant à bas niveau le fonctionnement du système afin d'en contrôler plus finement les ressources. Cette solution rend la conception des systèmes longue et complexe, et est en opposition avec les règles de génie logiciel usuelles.

Ces deux approches ont pour but unique de réduire le temps d'exécution perceptible des traitements. Ceci afin de se garder une marge de sécurité importante, rendant le risque de non-respect des contraintes temporelles (nommées *fautes temporelles*) faible. Cependant, réduire

un risque ne signifie pas l'éliminer. Ainsi, rien ne prouve formellement, lors de la mise en route du système, qu'il fonctionnera correctement dans toutes les situations possibles. Au mieux, on dispose d'une probabilité de défaillance obtenue par des méthodes de calcul ou de test statistique.

Pourtant, de telles applications ne doivent être mises en exploitation que sous la garantie du risque zéro ou, pour les applications non critiques, une assurance de qualité ergonomique. Les exemples cités ci-dessus mettent en évidence des défaillances dues à des fautes temporelles, pour lesquelles nous nous proposons de mettre en place une méthodologie d'évitement a priori.

### INTERFACES HOMME-MACHINE TEMPS-RÉEL

Les contraintes temporelles que nous désirons assurer peuvent s'exprimer, du point de vue de l'utilisateur, sous la forme de propriétés ergonomiques dites externes [4]. Plus précisément, nous définissons une *Interface Homme-Machine Temps-Réel* (IHM-TR) comme un moyen dont dispose l'utilisateur pour appréhender et commander un système interactif et qui assure les propriétés suivantes :

- *L'IHM-TR doit être en mesure de mettre à disposition de l'utilisateur une image fiable de l'état du système et dont l'âge peut être borné.* Il s'agit ici de l'interprétation temporelle de la propriété d'observabilité (*observability*). Nous utiliserons désormais le terme d'observabilité temporelle. Cette propriété est prédominante dans l'exemple des consoles de visualisation du contrôle aérien.
- *L'IHM-TR doit permettre à l'utilisateur de se déplacer parmi les états du système avec des temps compatibles avec le fonctionnement de ce système.* Il s'agit de la propriété d'atteignabilité (*reachability*) vue du point de vue temporel. Nous utiliserons le terme d'atteignabilité temporelle. Cette propriété a fait défaut dans l'exemple de l'avion de chasse suédois.
- *L'IHM-TR doit implanter des dispositifs de commande compatibles avec les temps de réaction de l'utilisateur.* Cette propriété ne s'intéresse pas directement à la conduite du système comme les deux précédentes, mais plus précisément au fonctionnement de l'interface elle-même. Elle s'apparente à la propriété d'adaptation au rythme (*pace tolerance*) qui exprime le fait que la vitesse de réaction du système interactif doit être adaptée à celle de l'utilisateur et vice-versa : ni trop vite, ni trop lentement. C'est cette propriété qui doit être assurée dans l'exemple du système MATIS.

Ces propriétés définissent une IHM-TR. Cependant, le respect des contraintes temporelles intervient, bien que

dans une moindre mesure, sur d'autres propriétés. Par exemple, la capacité pour un système d'autoriser le dialogue à plusieurs fils (*multithreading*) ou la prédictibilité (*predictability*) des temps de réponse du système sont en partie sous la dépendance des contraintes temporelles.

Du point de vue de l'activité de conception, les propriétés évoquées ci-dessus, c'est-à-dire l'observabilité et l'atteignabilité temporelle ainsi que la tolérance au rythme peuvent s'implanter par des processus du système respectant ces deux types de contraintes :

- Il existe un temps borné entre l'événement déclencheur et la mise en fonction effective du processus chargé de le traiter. L'événement pouvant être du fait de l'utilisateur, d'un procédé ou propre à l'interface. Par exemple, les consignes d'un pilote doivent être envoyées aux vérins des surfaces de contrôle de l'avion dans un temps borné.
- Lorsqu'un processus est en fonctionnement, il doit être en mesure de fonctionner périodiquement à une certaine fréquence et à chaque période pendant une certaine durée. Par exemple, les écrans du contrôle aérien doivent être rafraîchis toutes les 5 secondes.

Ces contraintes des processus du système sont la condition nécessaire, mais aussi suffisante pour assurer le respect des propriétés ergonomiques que nous avons ciblées. Ces contraintes temporelles peuvent être rapprochées des préoccupations de la communauté de recherche en informatique Temps-Réel. Ainsi, dans la suite de cet article, nous montrons comment modéliser un système interactif comme un système temps-réel en proposant une distinction entre les systèmes interactifs apparentés au *pilotage* de ceux apparentés à la *supervision*.

### MODÉLISATION PROPOSÉE

Historiquement, le développement de l'informatique temps-réel a été motivé par l'automatisation de la conduite de procédés techniques. Dans ce contexte, l'ordinateur, appelé le *pilote*, se substitue à l'humain. Le pilotage consiste à partir des *signaux* émis par le *procédé* à agir sur celui-ci via les *commandes*, de façon à assurer une évolution correcte de son comportement compte tenu de contraintes temporelles (Cf. figure 1). La validation d'un système temps-réel (ou *ordonnançabilité*) consiste à assurer formellement que tous les processus pourront s'effectuer dans les délais spécifiés [2].

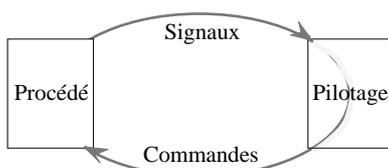


Figure 1 : La boucle de pilotage d'un système temps-réel

La validation d'une IHM-TR peut avoir deux motivations orthogonales. La première vise à assurer la capacité de l'homme à piloter un procédé par l'intermédiaire de son interface, par exemple un avion. Nous l'avons définie comme l'approche *pilotage*. La seconde vise à assurer que l'interface respecte un certain cahier des charges compatible avec les temps de réaction humains, c'est l'approche *supervision*. Bien entendu il existe des cas intermédiaires, en particulier certains systèmes combinant dans la même interface les deux approches.

### Approche « pilotage »

Cette approche vise à assurer que le système est globalement pilotable, c'est-à-dire que le procédé pourra être opérationnellement contrôlé par l'opérateur humain à partir des connaissances que nous avons sur les contraintes temporelles de ce procédé, les temps de réaction des capteurs et actionneurs (éventuellement informatisés), ainsi que les temps de réaction humains. Le domaine de validité de cette modélisation est très limité et se situe dans les cas où l'homme peut être considéré comme un composant fiable répondant dans un temps donné aux sollicitations du système. Seules les activités réflexes sont ainsi éligibles. Les données sur l'opérateur sont alors celles issues du modèle du processeur humain [1]. Cette modélisation est en somme un retour dans le passé, au sens où c'est l'homme qui se substitue à l'ordinateur pour contrôler le procédé.

Dans cette modélisation le pilote du procédé est l'opérateur humain, et la validation temporelle doit assurer que la boucle de pilotage intégrant l'homme est compatible avec les délais critiques du procédé. Dans l'exemple de l'avion de chasse suédois, cette validation aurait montré une impossibilité. Cependant, cette approche est limitée aux cas où le procédé est directement contrôlé par l'opérateur comme avec un automate (principe de consigne, erreur et correction). Dans de nombreux autres cas, l'opérateur ne contrôle pas directement le procédé, ou même il n'y a pas véritablement de procédé, l'utilisateur agit alors en superviseur.

### Approche « supervision »

A l'opposé de l'approche pilotage où le but est de valider le système dans sa globalité avec un opérateur réagissant avec des durées connues, l'approche supervision considère que les temps de réaction de l'homme sont difficilement mesurables car la conduite du système nécessite de la part de l'utilisateur des tâches de résolution de problème. C'est par exemple le cas du contrôle aérien. Il est ainsi irréaliste de vouloir valider un système dans son ensemble. On peut cependant chercher à garantir que les temps de réaction du système satisfont au cahier des charges du comportement humain. C'est le cas général.

Dans cette approche les rôles sont inversés au sens où le procédé à contrôler est l'utilisateur et le pilote de ce procédé est l'IHM-TR. Le système interactif doit alors implanter un certain nombre de processus destinées à gérer l'interaction. Ainsi, pour « contrôler » l'utilisateur le pilote temps-réel dispose d'actionneurs qui sont les affichages et de capteurs qui sont les dispositifs d'entrée de l'utilisateur. Parmi les processus, on dénombre :

- *Les processus périodiques* à durée constante : ce sont en sortie les processus d'affichage comme ceux utilisés pour les écrans du contrôle aérien. Dans la communauté temps-réel, ces processus périodiques sont bien connus et de nombreux résultats permettent d'en assurer l'ordonnancement.
- *Les processus aperiodiques* : ils représentent les interactions opportunistes de l'utilisateur sur le système ou les alarmes déclenchées par ce dernier. Ils sont par essence non prévisibles, mais leur nombre maximal à un instant donné peut être évalué. Par exemple pour l'utilisateur, une seule interaction à la souris est possible à un instant donné, il est donc inutile de prévoir un système supportant le déplacement de deux objets (non groupés) simultanément, sauf bien entendu s'il y a deux souris... Dans la communauté temps-réel, ces processus aperiodiques sont connus et il existe certains résultats permettant d'en assurer l'ordonnancement.
- *Les processus périodiques par rafales* : ce cas particulier est constitué des processus qui sont périodiques mais peuvent être déclenchés et terminés en cours d'interaction. Ce sont par exemple les systèmes de reconnaissance de la parole déclenchés à l'initiative de l'utilisateur. Dans la communauté temps-réel, de tels processus posent encore de nombreux problèmes d'ordonnancement.

#### CONCLUSION & AXES DE RECHERCHE

Dans cet article, nous avons montré que les solutions actuelles en matière de validation de systèmes dont l'IHM doit respecter des contraintes temporelles ne sont pas satisfaisantes. Pour palier à ce problème, nous proposons une approche temps-réel, dans laquelle l'interface homme-machine est un (ou un ensemble de) processus au même titre que les autres composants du système. L'application des méthodes de validation temps-réel permet d'assurer que les contraintes temporelles de l'IHM-TR seront toujours satisfaites, quoi qu'il arrive.

Cette approche met en lumière plusieurs pré-requis nécessaires à la mise au point d'un processus de développement systématique d'IHM-TR :

- Le développement d'une IHM-TR exige de la part du concepteur de spécifier précisément les contraintes temporelles critiques vis-à-vis des exi-

gences à la fois fonctionnelles mais aussi ergonomiques. Pour cela nous envisageons l'utilisation de formalismes tels les réseaux de Petri temporisés, les logiques temporelles ou le langage B événementiel.

- Les systèmes d'exploitation temps-réel ne disposent pas de boîtes à outils graphiques temporellement validées, c'est-à-dire aux routines desquelles on peut associer de manière garantie les caractéristiques temporelles. Cette validation s'applique au niveau de l'électronique des cartes graphiques, et au niveau du logiciel des boîtes à outils.
- Les processus temps-réel périodiques par rafales sont encore un sujet de recherche pour la communauté du temps-réel. Ce type de processus n'étant pas couramment utilisée dans les systèmes critiques de contrôle de procédé. On peut, bien sûr, les assimiler brutalement à des processus périodiques, mais cette option aurait pour effet de surcharger inutilement le système, et de rendre irréalistes les architectures matérielles ciblées.

#### REMERCIEMENTS

Les auteurs remercient S. Chatty et S. Maury, du Centre d'Étude de la Navigation Aérienne, pour les échanges fructueux concernant leurs retours d'expérience sur les problèmes liés à l'intégration du temps dans les interfaces homme-machine.

#### BIBLIOGRAPHIE

1. Card, S., Moran, T. et Newell, A. *The Psychology of Human-Computer Interaction*. Lawrence Erlbaum Associates, 1983.
2. Cottet, F., Delacroix, J., Kaiser, C. et Mammeri, Z. *Ordonnement temps réel*. Hermès science, 2000.
3. Fields, B., Wright, P. et Harrison, M. Temporal aspects of usability / Time, tasks and errors. *ACM SIGCHI Bulletin*. 28, 2 (April 1996).
4. Gram, C. et Cockton, G. *Design Principles for Interactive Software*. Chapman & Hall, 1996.
5. Johnson, C. The Challenge of Time. In *Proceedings of Design, Specification and Verification of Interactive Systems (DSV-IS'95)* (June 7-9, Bonas, France), Springer-Verlag/Wien, 1995, pp. 345-357.
6. Johnson, C. et Gray, P. Assessing the Impact of Time on User Interface design. *SIGCHI Bulletin*. 28, 2 (April 1996), pp. 33-35.
7. Neumann, P.G. *Computer-Related Risks*. Addison-Wesley, 1995.
8. Nigay, L. *Conception et Modélisation Logicielle des Systèmes Interactifs : Application aux Interfaces Multimodales*. Doctorat d'Université (PhD Thesis) : Université Joseph Fourier, 1994.