Assurer la sécurité de véhicules autonomes démonstrateurs -l'approche Safety-Bag-

Manel Brini, Paul Crubillé, Benjamin Lussier, Walter Schon

Assurer la sécurité de véhicules autonomes démonstrateurs

-l'approche Safety-Bag-

Manel Brini, Paul Crubillé*, Benjamin Lussier*, Walter Schôn**
**Sorbonne université, Université de Technologie de Compiègne, CNRS,*
Heudiasyc UMR 7253, CS 60 319, 60 203 Compiègne Cedex.
Email: {manel.brini, paul.crubille, benjamin.lussier, walter.schon} @hds.utc.fr

RESUME

Cet article présente une étude de sûreté de fonctionnement menée sur les véhicules autonomes robotisés de l'équipe ASER dans le laboratoire Heudiasyc. Cette étude confirme que l'utilisation de ces véhicules comporte des risques importants lors d'expérimentations, et que l'insertion dans l'architecture du véhicule d'un composant *Safety-Bag* permet de réduire considérablement la gravité de ces risques.

ABSTRACT

This work presents a study concerning the dependability of autonomous vehicles within the ASER team in the Heudiasyc laboratory. This study confirms that the use of such vehicles involves significant risks during experimentations, and that the integration of the Safety-Bag component into the vehicle architecture can significantly reduce the level of those risks.

Mots clés: *Safety-Bag*, sûreté de fonctionnement, sécurité-innocuité, véhicule autonome, tolérance aux fautes.
Keywords: Safety-Bag, Dependability, Safety, Autonomous vehicle, Faults tolerance.

I. INTRODUCTION

Les véhicules autonomes sont des robots mobiles rapides et puissants, donc capables d'accumuler une grande quantité d'énergie, d'être source de danger pour leur opérateur ou leur environnement. Etant par ailleurs complexes et faisant appel à des logiciels d'intelligence artificielle basés sur des mécanismes déclaratifs, leur supervision, leur validation, et leur vérification sont difficiles[1]. Dans le cas de véhicules expérimentaux, l'évolution continue du système et des logiciels, ainsi que des étapes de validation moins longues et rigoureuses que pour des systèmes industrialisés, rendent le problème encore plus complexe. Pour les véhicules du laboratoire Heudiasyc, nous avons réalisé une étude de sûreté de fonctionnement en utilisant les techniques d'analyse de risques AMDEC et les arbres de défaillances [6][11]. Il en ressort que même avec un pilote formé à la reprise en main du véhicule, les risques d'accidents sont importants.

Nous avons alors étudié l'intégration d'un *"Safety-Bag" ou "Composant Indépendant de Sécurité-innocuité"* à l'architecture de contrôle-commande des véhicules. Ce *Safety-Bag* assure la supervision des règles de sécurité, particulièrement en regard du comportement de la partie applicative [8][9]. Il contrôle par exemple que cette application est vivante, et qu'elle respecte des conditions de vitesse et de vitesse de rotation définies, et dans le cas contraire il est capable d'alerter le pilote rapidement et de mettre le véhicule dans des conditions permettant la reprise en manuel dans de bonnes conditions [7][12]. Basé sur deux calculateurs embarqués, il intègre une redondance très partielle mais suffisante pour éviter qu'une défaillance unique de ses composants matériels mène à une situation catastrophique. L'étude de sûreté de fonctionnement [10] que nous avons effectuée pour un véhicule robotisé de type Fluence du laboratoire Heudiasyc est basée sur deux méthodes classiques: la méthode AMDEC et les arbres de défaillances.

II. ETUDE DU VEHICULE SANS SAFETY-BAG

1. Architecture du véhicule sans Safety-Bag

Nous présentons ici l'architecture du système de contrôle commande d'un véhicule robotisé sous la forme d'un schéma de déploiement UML. Ce schéma ne détaille pas la structure interne de l'application mais met en évidence le rôle du pilote d'essai, des capteurs et des actionneurs.

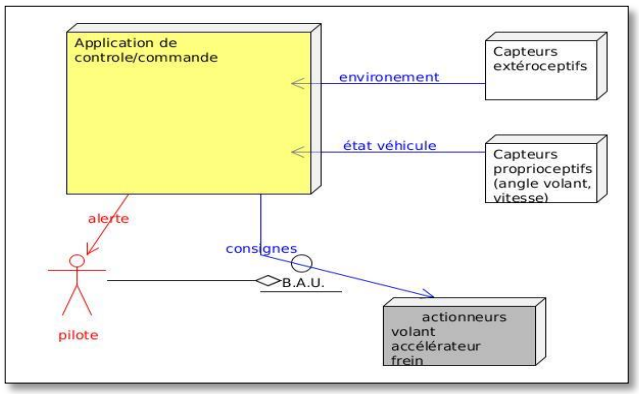

Figure1: Schéma de déploiement UML simplifié

Le pilote doit être capable de reprendre à tout instant le contrôle du véhicule au cas où une défaillance du contrôle robotisé aurait lieu. Le pilote doit ainsi surveiller en permanence la situation de conduite, le comportement du véhicule et les commandes appliquées. Sa capacité à réagir rapidement est critique pour limiter la gravité d'une défaillance. Cependant, le temps de réaction humain dans ce contexte difficile est de l'ordre de quelques secondes, à mettre en rapport avec les capacités du véhicule d'accélérer de 0 à 50km/h en 4 secondes. Des alertes explicites émises par l'application peuvent être d'une grande aide pour réduire ce délai de réaction et permettre une réaction plus rapide et mieux ciblée du pilote.

2. AMDEC du véhicule robotisé

a. Objectifs:

L'objectif est d'établir la liste des composants pouvant être à l'origine d'une défaillance catastrophique du contrôle autonome du véhicule. Une défaillance catastrophique est une défaillance susceptible de provoquer des blessures graves, des morts ou des dégâts matériels importants. Cette analyse permet ensuite de proposer les mesures de réduction des risques.

b. Composition:

L'AMDEC est composée de 2 analyses:

✓ L'AMDE qui est une démarche ascendante consistant à identifier au niveau d'un système les modes potentiels de défaillance de ses éléments, leurs causes et leurs effets.

✓ L'Evaluation de Criticité des modes de défaillance qui passe par la prise en considération de la probabilité d'occurrence du mode de défaillance, la possibilité de mettre en place une détection et de la gravité des conséquences.

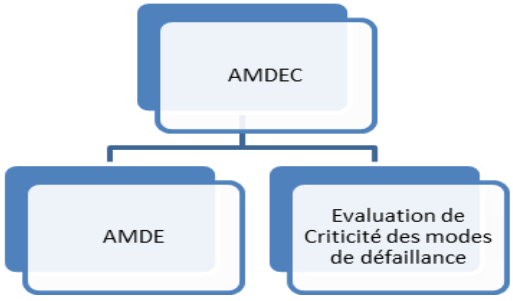

Figure2: Principe de la méthode d'analyse de risques AMDEC

c. Echelle de gravité:

L'*échelle de gravité* sert de notation pour la présentation de l'analyse AMDEC mais est également un guide pour les démarches de réduction des risques.

N° G	Echelle de gravité
0	Fonctionnement normal
1	Arrêt de l'expérimentation automatique **ou** nécessité de reprise en manuel en 10s
2	Commande maintenue devenant obsolète ==>**alerte** et nécessité de reprise en main en 2s
3	Commande non maintenue/ **alerte** et nécessité de reprise en manuel facile immédiate
4	Commande non maintenue/ nécessité de reprise en main facile immédiate **Sans** alerte
5	Commande aberrante maintenue avec conséquences catastrophiques

Figure3: Table des niveaux de gravité

d. L'AMDEC pour les véhicules Sans Safety-Bag:

L'analyse AMDEC se présente sous la forme d'un tableau détaillant pour chaque composant ses différents modes de défaillance et les conséquences associées, notamment leur gravité. Nous y détaillons également les moyens de détection et de correction possibles, ainsi que l'ordre de grandeur de leurs temps permettant d'éviter la défaillance. La figure 4 présente un extrait de l'analyse AMDEC pour le véhicule autonome concernant seulement deux composants et quelques causes possibles de leurs défaillances.
Dans un dossier de sécurité, l'analyse AMDEC doit être aussi complète que possible, et peut comprendre un très grand nombre de lignes suivant le niveau d'abstraction des composants étudiés.
Dans notre cas, nous considérons des dizaines d'éléments et un grand nombre des causes. Nous nous limitons ici à un exemple illustratif mais qui met en évidence que des situations à haut risques ont des probabilités d'occurrences élevées.

λ(t): est l'inverse du MTBF et il est difficile d'obtenir des valeurs précises sans réaliser un très grand nombre d'expérimentations. Nous avons pris des valeurs pessimistes en ce qui concerne les valeurs de défaillance matérielle, à partir de notre connaissance du système et de ses composants.
Pour les défaillances logicielles, les composants expérimentaux utilisés sont développés dans un contexte de recherche et ne pourront être testés que sommairement avant d'être intégrés dans le système complet. Nous n'avons pas donné d'estimation, mais la pratique nous conduit à croire qu'ils sont significativement élevés.

e. Résultats et interprétations:

Les défaillances des composants matériels ou les erreurs logicielles peuvent amener des tensions de commande sur l'accélérateur et sur le volant, menant très rapidement le véhicule dans un état ou toute tentative de reprise en main est illusoire. La perte des informations de vitesse ou d'angle volant par exemple peut, en l'absence de diagnostic, amener une application à commander une trajectoire également aberrante.

Eléments	**Causes**	**λ(t)**	**Effets**	**Moyens de détection**	**t/d (s)**	**Moyens d'action**	**t/a (s)**	**Gravité des conséquences**	
								G	**Commentaires**
Convertisseur D/A	-Panne D/A -Arrêt/blocage de calculateur de contrôle -Panne logicielle livelock	*-<10^-5* *-≈10^-3* *-???*	Accélération maintenue et couple sur la direction maintenue	Pilote d'essai observe un comportement anormal c.à.d. accélération incontrôlée + volant fou	2s	Reprise en manuel difficile voire impossible	+2s	5	Véhicule incontrôlable (La vitesse peut augmenter et la voiture tourner)
Capteurs du véhicule vitesse angle-volant	-Panne capteur de véhicule -Panne connectique artisanale	≈10^-5	Commande désactivée si le diagnostic est effectué	L'application si elle intègre un diagnostic	<0.5s	Pilote	<+2s	2/3	La vitesse reste faible et le pilote peut intervenir
		10^-3	Commande aberrante si non détectée	Le pilote	0.2s	Pilote	+2s	4/5	Véhicule incontrôlable

Figure 4 : Extrait de l'analyse AMDEC du véhicule sans *Safety-Bag*

III. ETUDE DU VEHICULE AVEC SAFETY-BAG

1. Architecture du véhicule avec Safety-Bag

Pour réduire les risques en cas de défaillance, nous insérons dans l'architecture un composant constitué de deux calculateurs: un "*Cube*" et un "*IGEP*". Ce composant "*Safety-Bag*" s'assure que les commandes produites par l'application respectent des règles de sécurité-innocuité avant de les transmettre aux actionneurs du véhicule.

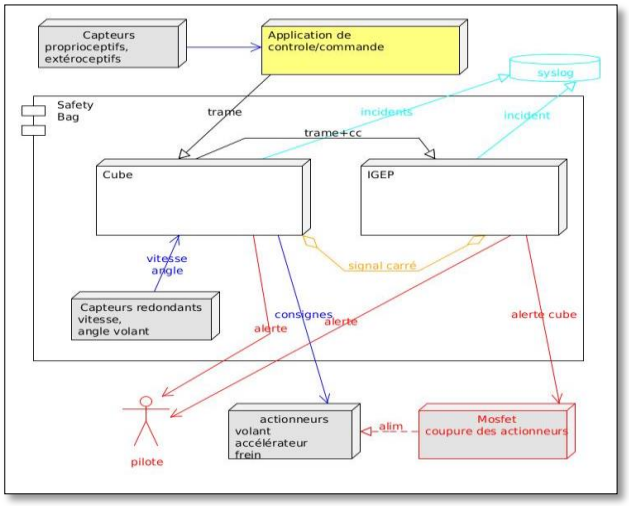

Figure5: Schéma de déploiement UML simplifié

Le "Cube" vérifie la vivacité de l'application de contrôle/commande en supervisant les fréquences d'envoi des commandes. Il utilise également des capteurs de vitesse et de position du volant différents de ceux de l'application pour vérifier le respect des règles de sécurité-innocuité et la crédibilité des informations utilisées par l'application. Les deux calculateurs se co-surveillent par l'échange d'un signal carré (*heart-bit*), dont la perte signifie la défaillance de l'autre calculateur et est remontée à l'opérateur. Si l'IGEP ne perçoit plus le signal émis par le Cube, il désactive les actionneurs du véhicule par l'intermédiaire d'un MOSFET (installé sur leur ligne d'alimentation), pour éviter que ces actionneurs ne continuent à envoyer des données aberrantes. Le Cube n'est plus supervisé et il demande l'arrêt de l'expérimentation au pilote, pour éviter que sa possible défaillance future ne soit pas détectée et ne conduise à une défaillance du véhicule. Par ces différentes méthodes, le *Safety-Bag* réduit la gravité de la plupart des défaillances de niveau 5 de notre système, à un niveau de gravité 3 (dont les défaillances présentées dans la Figure 4).

Accessoirement, le *Safety-Bag* offre également un système d'alerte redondant et unifié qui facilite la formation des pilotes afin de réduire leur temps de réaction, ainsi qu'un système de log regroupant des évènements concernant la sécurité-innocuité du véhicule.

2. Arbre des défaillances

Nous ne présenterons pas ici l'analyse AMDEC du véhicule robotisé avec *Safety-Bag*, mentionnée dans la section précédente, et nous limiterons à une illustration de l'utilisation des arbres de défaillances dans ce contexte.

a. Objectifs:

Un arbre de défaillance est une méthode déductive sous forme d'une représentation graphique qui permet de déterminer les causes d'une défaillance d'un système ou composant. Il s'agit de représenter les différents évènements nœuds ainsi que leurs liaisons (portes logiques ET ou OU). Par exemple, l'arbre de défaillances donné en figure 6 représente l'enchaînement d'un "Evènement Redouté" (une défaillance matérielle du *Safety Bag*) à travers des "Evènements intermédiaires" (tels que la non-mise à jour des calculateurs) jusqu'aux "Evènements élémentaires" tels que la défaillance du Cube).

b. Arbre de défaillance:

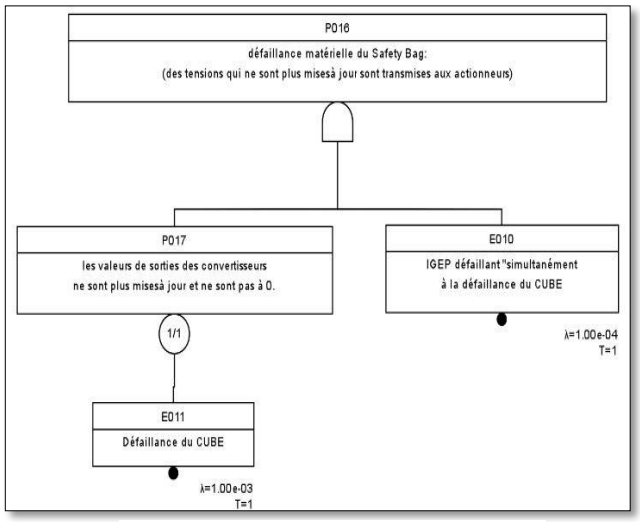

Figure6: Arbre de défaillances

d. Résultats et interprétations:

L'arbre étudie l'enchainement conduisant à *l'évènement redouté* défaillance matérielle du *Safety-Bag* due à la défaillance successive des deux calculateurs (Cube et IGEP) en moins de quelques millisecondes. Une défaillance aussi rapide du second calculateur empêcherait de diagnostiquer la défaillance du premier, provoquant une défaillance sans alerte du *Safety-Bag*, et des conséquences possibles de gravité de 5. La probabilité d'une telle succession d'événement est cependant extrêmement faible.

VI. CONCLUSION

D'après notre étude un *Safety-Bag* ou *Composant Indépendant de Sécurité* permet de réduire considérablement la dangerosité des véhicules autonomes expérimentaux. Cette approche détecte les dysfonctionnements de l'application de contrôle commande, tels que des blocages ou des commandes dangereuses (dépassement de vitesse limite longitudinale et latérale). Il produit également des actions de mise en état sûr, telles que le rejet de commandes dangereuses, ou un repli en mode manuel accompagné de signaux d'alerte pour pilote formé et entrainé.

Cependant, pour éviter l'introduction de nouvelles fautes dans le système, causant potentiellement des défaillances catastrophiques, le comportement du *Safety-Bag* doit rester simple et facile à valider, ce qui limite l'expressivité des règles de sécurité qu'il assure. Par exemple dans nos systèmes, la validation ou le contrôle de suivi de trajectoires n'est pas prévue à son niveau. De plus, la réduction de défaillance qu'il permet dans certains cas reporte la gestion de la défaillance sur le conducteur par le passage en mode manuel avec alarme. Ainsi, des conducteurs formés et vigilants restent indispensables dans les véhicules, mais le *Safety-Bag* permet de garantir un bien plus grande confiance dans leur sécurité.

V. REMERCIEMENTS

Ce travail a été réalisé et financé dans le cadre de l'EQUIPEX ROBOTEX. Il a été soutenu par le gouvernement français, à travers les programmes "Investissement d'avenir" gérés par l'Agence Nationale de la Recherche. (Références: ANR-10-EQPX).

IV. BIBLIOGRAPHIE

[1] Lussier, B. (2007). Tolérance aux fautes dans les systèmes autonomes.

[2] Blanquart, J., Fleury, S., &Hernek, M. (n.d.). Software Safety Supervision On-board Autonomous Spacecraft.

[3] David, P., &Guiochet, J. (2005). Etude et analyse de differents dispositifs externes de securite-innocuite de type safety bag.

[4] Baudin, É., Blanquart, J. P., Guiochet, J., & Powell, D. (2007). Independent Safety Systems for Autonomy.

[5] Powell, D., &Thévenod-Fosse, P. (2002, October). Dependability issues in ai-based autonomous systems for space applications. In 2nd IARP/IEEE-RAS Joint Workshop on Technical Challenge for Dependable Robots in Human Environments (pp.163-177).

[6] Mekki Mokhtar, A. (2012). Processus d'identification de proprietes de securite-innocuite verifiables en ligne pour des systemes autonomes critiques. Toulouse: Université de Toulouse.

[7] Mekki Mokhtar, A., Blanquart, J.-P., & Guiochet, J. (n.d.). Safety Trigger Conditions for Critical Autonomous Systems.

[8] Pace, C., & Seward, D. (n.d.). An approach to safety for a robotic excavator.1-7.

[9] Pace, C., Seward, D., &Sommerville, I. (n.d.). A Safety Integrated Architecturef or an AutonomousExcavator. IEEE, 1-10.

[10] Pietre-Cambacedes, L. (2010). Des relations entre surete et securite. Paris: Telecom ParisTech.

[11] Amina Mekki-Mokhtar. Processus d'identication de proprietes de securite-innocuiteveriables en ligne pour des systemes autonomes critiques. Robotics.Universite Paul Sabatier – Toulouse III, 2012.French.<tel-00766636>.

[12] D.Guiochet, J., Powell, ,Baudin, E., &Blanquart, J. P. (2008, May). Online safety monitoring using safety modes.In Workshop on Technical Challenges for Dependable Robots in Human Environments (pp.1-13).