Surveillance des processus dynamiques par automates hybrides linéaires

MOHAMED FATHI KAROUI[12], HASSANE ALLA[1], ABDERRAZAK CHATTI[2]

[1]Gipsa-Lab, Département d'Automatique,
961, Rue de la houille blanche – BP- 46, 38402, Saint Martin d'hères
Mohamed_fathi_karouil@yahoo.fr, Hassane.Alla@gipsa-lab.inpg.fr

[2]Institut National des Sciences Appliquées et de Technologie
Centre Urbain Nord BP 676 - 1080 Tunis Cedex.
abderrazak.chatti@insat.rnu.tn

Résumé — **Dans ce papier, nous développons une méthode de surveillance des processus dynamiques basée sur les automates hybrides linéaires. Cette méthode tient compte du comportement du système ainsi que de l'évolution de ses paramètres. A cause de dysfonctionnements la dynamique du système peut varier entre plusieurs modes. La surveillance consiste à partir d'une durée globale de fonctionnement du système à garantir à tout instant que cette durée est respectée. La connaissance de l'espace atteignable réalisant cet objectif permettra de détecter au plus tôt le non respect de cette durée.**

Mots-clés— **Surveillance, automate hybride linéaire, processus dynamiques.**

I. INTRODUCTION

Le contexte économique actuel a rendu incontournable les exigences de performance, de fiabilité et de sécurité des systèmes réels. Devant la complexité croissante de ces systèmes il est nécessaire d'établir un mécanisme de surveillance qui signale le plus tôt possible à l'opérateur les écarts détectés par rapport au comportement nominal prévu [1] [4] [12]

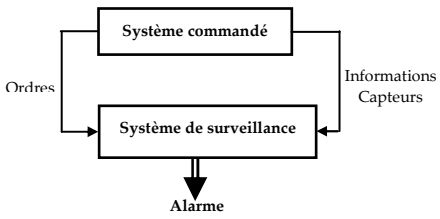

Fig. 1. Modèle de surveillance d'un système commandé

Comme c'est souvent le cas entre la théorie et la pratique il y a souvent une divergence, ces systèmes en fonctionnement réel ne se conforment pas à leurs actions prédéfinies. En effet un dysfonctionnement peut surgir, les causes de ce mauvais fonctionnement sont diverses ; l'occurrence d'un défaut, une mauvaise décision du système de commande ou une instrumentation erronée des capteurs [9].

Le problème de base consiste à étudier un système (Figure 2) comme pouvant évoluer dans plusieurs modes de fonctionnement différents : le premier est le mode initial où les paramètres sont initialisés à zéro, le deuxième est le mode normal où on a une évolution avec les paramètres nominaux préétablis. A partir du mode normal, plusieurs modes de dysfonctionnements sont possibles, chacun ayant une dynamique distincte.

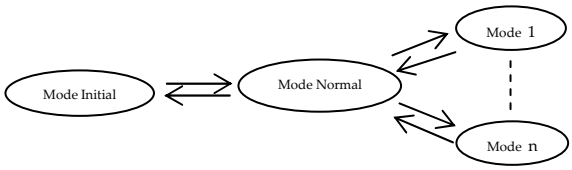

Fig. 2. Différents états du système

La dynamique d'un système peut être représentée de différentes manières. Cependant plus cette dynamique est complexe, plus les possibilités d'analyse formelle sont faibles. Il s'agit donc de trouver un compromis entre la qualité du résultat et la complexité du modèle. Pour notre part, nous considérons une vision événementielle dans laquelle nous injectons un comportement continu. Dans les premiers travaux déjà réalisés [2] les dynamiques sont données par des horloges qui peuvent être arrêtées. Il a alors été possible de synthétiser un système de surveillance pour des processus interruptibles. Nous considérons ici des dynamiques données par des équations du type dx/dt = constante. Cela va permettre d'augmenter la classe de systèmes étudiés. La surveillance consiste alors à partir d'une durée globale de fonctionnement du système (durée définie par un intervalle représentant une incertitude) à garantir à tout instant que cette durée est respectée. La connaissance de l'espace atteignable réalisant cet objectif permettra de détecter au plus tôt le non respect de cette durée.

II. PRESENTATION INTUITIVE DE L'APPROCHE

Pour bien illustrer le problème nous considérons le système suivant (figure 3). Un atelier de collage composé d'un poste P1 où il y a un dépôt de colle qui est effectué sur une pièce. Un convoyeur va transporter cette pièce vers le poste P2 où il y aura l'opération de collage. Le convoyeur débute sa tache suite à l'ordre d du contrôleur (après que la colle ait été

déposée sur la pièce). L'évènement b survient quand la pièce atteigne le poste P2. Le tapis roulant à deux fonctions ; la première c'est de transporter la pièce de P1 à P2 et la deuxième c'est qu'il sert de retardateur. En effet le trajet de la pièce est déterminé afin que la colle soit prête pour être utilisée. Le convoyeur a une vitesse nominale V_0 qui définit l'état normal, avec cette vitesse il devrait porter la pièce à destination en 8 u.t. (unités de temps). La durée acceptable du trajet est comprise dans l'intervalle [7, 8] u.t. Cet intervalle correspond à la période de temps durant laquelle la colle est exploitable. Un capteur logique produit les évènements c_i et r_i, ces évènements indiquent un changement de vitesse du convoyeur.

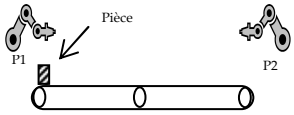

Fig. 3. Atelier de collage

Les techniques actuelles de surveillance provenant des systèmes à événements discrets (SED) détectent le mode défaillant si la durée finale du trajet n'est pas comprise dans cet intervalle. Nous proposons une approche plus précise pour détecter la défaillance au plus tôt et ce en calculant l'espace des états atteignables.

Pour présenter notre approche, on construit le modèle de surveillance de l'exemple illustré par la figure 3. Dés que le contrôleur donne l'ordre *d* deux compteurs *x* et *y* sont initialisés (Figure 4). Le compteur *x* mesure le temps écoulé depuis l'instant de la mise en route du convoyeur jusqu'à son arrêt, la dynamique de *x* prendra comme valeur :

$\dot{x} = 1$; Le convoyeur est en marche.

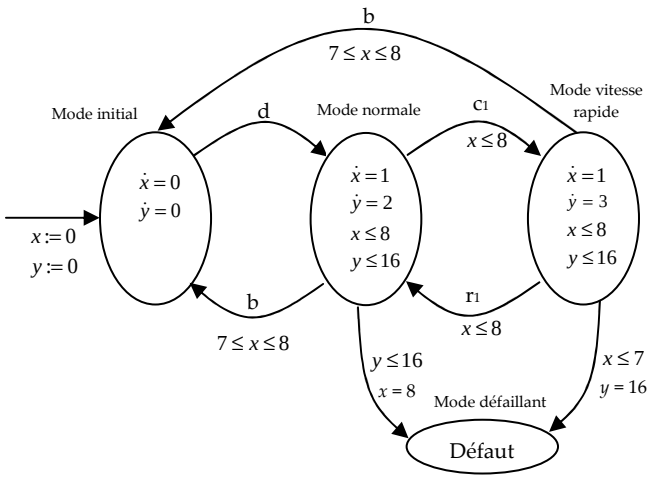

Fig. 4. Modèle proposé de la surveillance du convoyeur

L'arrêt du convoyeur est indiqué par l'occurrence de l'évènement b. Lorsque b a lieu, la valeur de *x* indique si la tache a été exécutée pendant la durée acceptable [7, 8]. Dans un fonctionnement normal la pièce se déplace sur le convoyeur et arrive au poste P2 en 8 u.t.

On définit le mode normale par l'état où les dynamiques caractérisants les variables *x* et *y* soient conforment aux valeurs prédéfinit avant le début de l'exécution. Dans notre cas la dynamique de *x* vaut 1 et la dynamique de *y* vaut 2. Tout écart de ces deux valeurs entraînera l'apparition d'un autre mode fonctionnement.

Lors du transport de la pièce du poste P1 au poste P2, la vitesse du convoyeur et donc celle de la pièce peut s'écarter de la vitesse nominale à cause de dysfonctionnements. Nous considérons dans cet exemple un changement vers une vitesse supérieure à la vitesse nominale. Ce changement de vitesse impliquera un changement d'état du système de l'état normal vers le mode 1. Ce changement doit être observable.

La figure 5 illustre l'espace des états atteignables par l'automate. Tant que les paramètres (x,y) respectent cet espace le système pourra respecter ces contraintes temporelles et positionnelles. Par ailleurs si ces paramètres dépassent cet espace une alarme se déclenche aussitôt.

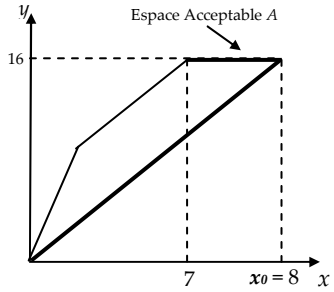

Fig. 5 Espace des états atteignables

Le compteur *y* indique la position de la pièce sur le convoyeur, la longueur du trajet est égale à 16m, lors d'une exécution normale lorsque l'événement *b* se présente on aura *y* = 16. La valeur de *y* indiquera si la pièce est arrivée au poste P2 ou pas. La dynamique de *y* reflète la vitesse de la pièce, autrement dit :

- $\dot{y} = 2$ quand la pièce évolue avec une vitesse nominale égale à 2 m/s.

- $\dot{y} = 3$ lorsque la pièce à une vitesse de 3 m/s.

On définit le mode vitesse rapide par le changement de la dynamique de *y* vers $\dot{y} = 3$.

Quand l'évènement *b* a lieu, la valeur de *x* indique si la pièce a été amenée vers le poste P2 dans l'intervalle acceptable [7, 8]. Normalement le convoyeur mettra 8 u.t. pour faire ce trajet. Au cours du trajet le convoyeur peut changer de vitesse, cette mutation est indiqué par l'évènement c_1, la pièce évoluera dans ce cas avec une vitesse supérieure à la vitesse nominale (v = 3 m/s), c'est le passage au mode 1, dans cet état la dynamique de *x* ne change pas par contre celle de *y* sera égale à 3. La condition associée à ce changement d'état et que $x \le 8$, puisque le changement de dynamique peut avoir lieu tant qu'on ne dépasse pas l'intervalle acceptable (avant alarme). Le retour à l'état normal est indiqué par l'évènement r_1, la dynamique de y sera de nouveau égale à 2,

la garde associée à cet évènement est $x \leq 8$. Les évènements c_1 et r_1 sont commandables.

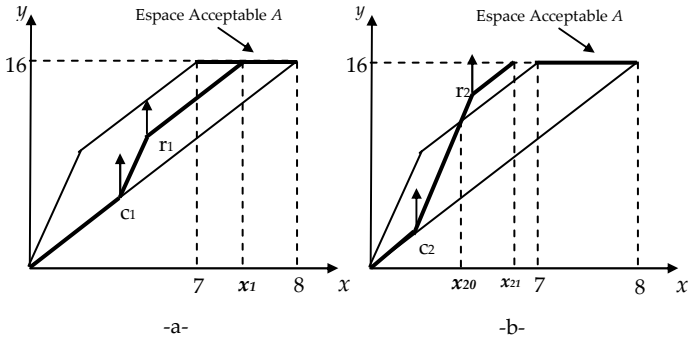

Fig. 6 : (a) trajectoire 1, (b) trajectoire 2

Un retour du mode 1 à l'état initial est possible si les conditions suivantes sont satisfaites ; $7 \leq x \leq 8$ et $y = 16$. Cela signifie que la tache a été effectuée et dans les délais.

Le passage du mode normale au mode défaut est régi par la contrainte suivante, l'horloge x indique que le temps maximal autorisé est atteint mais la pièce n'est pas encore arrivé au bout du convoyeur c'est-à-dire au poste P2. Le passage du mode vitesse rapide au mode défaut est régi par la contrainte suivante ; la pièce a atteint le poste P2 mais avec un temps inférieur à la durée minimale autorisé.

Notre objectif est de déterminer toutes les trajectoires possibles qui permettent d'atteindre le poste P2 tout en respectant les contraintes citées ci-dessus. Cela équivaut à déterminer l'espace d'état des variables (x, y) qui satisfait les conditions de terminaison correcte de la tâche.

La figure 6 illustre deux trajectoires : la première (Figure 6.a) définie par les évènements (c_1, r_1) la pièce atteint le poste P2 à x_1 qui est compris dans l'intervalle acceptable [7, 8]. Pour la deuxième trajectoire (Figure 6.b) qui est définie par les évènements (c_2, r_2) la pièce atteint le poste P2 à x_{21} qui n'appartient pas à l'espace des états atteignables. Le système de surveillance déclenche une alarme à l'instant x_{20}, car même si le système commandé retourne vers le mode de fonctionnement normal, la pièce n'atteindra pas le poste P2 dans l'intervalle acceptable. Ainsi il n'est pas nécessaire d'attendre l'instant x_{21} pour déclencher l'alarme.

III. SYNTHESE DE LA METHODE DE SURVEILLANCE

A. Comportement du système

On considère un système de surveillance comprenant plusieurs modes de fonctionnement : mode initial, mode normal et les modes de dysfonctionnement i. Chaque mode est défini par une dynamique distincte.

Dans la figure 7 le mode i est représenté de manière générique. On définit les variables x et y qui sont des paramètres observables, elles sont dans le système de surveillance et décrivent l'état du système. La variable x représente la durée d'exécution totale du processus et la variable y reflète l'état d'avancement du processus. Pour que l'exécution du processus soit effectuée correctement, il faut :

1) La variable x doit être comprise dans l'intervalle [α, β] (durée acceptable),

2) la tâche a été effectuée $y = \delta$ (valeur finale δ).

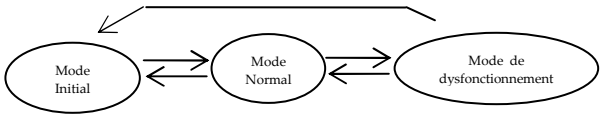

Fig. 7. Système à trois états

Durant son évolution le processus peut basculer du mode normal vers le mode défaillant i selon l'occurrence des évènements (c_i, r_i). Ces événements sont supposés être observables. Un retour à l'état initial est possible à partir du mode i si le processus a terminé son exécution tout en respectant les contraintes sur les variables x et y. Le comportement du processus peut être illustré par le chronogramme de la figure 8 :

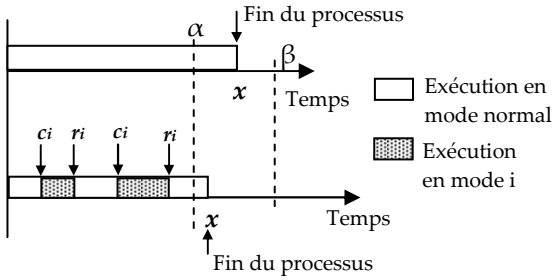

Fig. 8. Comportement du processus

B. Outils et techniques de modélisation

1. Automate hybride linéaire

Il y a plusieurs outils pour la modélisation des systèmes à événements discrets dynamiques dont les automates et les réseaux de Petri [5] Nous avons choisi l'automate hybride linéaire a cause de ca capacité d'analyse formelle [4] [8] [9].

Un automate hybride est un modèle formel qui opère par alternance de pas continus, ou les variables d'état et le temps évoluent de façon linéaire, et de pas discrets ou plusieurs transitions discrètes et instantanées peuvent êtres franchies. Ainsi, les changements discrets sont décrits par un automate d'état fini et la dynamique continue par un ensemble de variables et d'équations continues.

Un automate hybride linéaire est un 7-uplet H = (L, X, A, Σ, *dif*, *Inv*, l_0) [10] où :

- L : ensemble fini de sommets (appelés aussi localités, situations);
- X : ensemble fini de variables réelles (vecteur d'état à composantes continues) ;
- A : est un ensemble fini de transitions. a = (l, g, σ, R, l') ∈ A, avec l le sommet source, g la garde, σ

l'évènement associé, R l'affectation et l' le somment destination.

- Σ : ensemble fini d'étiquettes (i.e. ensemble d'actions événementielles liées aux franchissements de transitions)
- *Dif* : fonction associant à chaque sommet $l \in L$ un ensemble de comportements continus (appelés aussi activités) *Dif*(*l*) :

$$\left.\frac{dx_i(u)}{du}\right|_t = x_i(t) = cste_l$$

- *Inv* : fonction associant à chaque sommet $l \in L$ un invariant inv(*l*) (un prédicat sur les variables ; collection d'espaces de définition du vecteur d'état associés à chaque sommet);
- $l_0 \in L$: sommet initial.

Nous présentons dans la suite quelques techniques que nous emploierons afin de calculer les espaces des états atteignables [11].

L'état d'un automate est défini par la paire (*L*, *E*), *L* est le sommet où se trouve l'automate et *E* représente l'espace de temps de cet état. Quant le système atteint le sommet L_n les compteurs actifs ont plusieurs valeurs, l'ensemble de ces valeurs définissent l'espace de temps du sommet L_n : E_n^a .

Un automate a deux possibilités d'évolution à partir du sommet L_n :
- Rester dans le même sommet tandis que le temps s'écoule, l'espace d'état atteignable de cette évolution est appelé successeur continu Suc_t.

$$E_n = Suc_t(E_n^a)$$

- En franchissant la transition $a = (L_n, g_{n,n+1}, \sigma, R, L_{n+1}) \in A$, l'ensemble des états atteignables depuis tout état E_n est appelé successeur discret de la région E_n : Suc_d.

$$Suc_d(E_n) = E_n \wedge g_{n+1} \wedge R$$

Cette méthode d'analyse est appelée analyse en avant. On définit aussi la méthode d'analyse en arrière qui permet de calculer les prédécesseurs de chaque région. La notion de prédécesseur continu est duale à celle de successeur continu. Tout état *E* depuis lequel on peut atteindre un état donné *Q* en laissant le temps s'écouler tout en restant dans le même sommet est un prédécesseur continu de cet état.

On note : $E = Pre_t(Q)$

De même que la notion de prédécesseur discret est duale à celle de successeur discret.

2. Modélisation du processus

Pour modéliser le comportement du processus de la figure 7, nous avons recours à un modèle d'automate à trois états plus le mode Alarme. Pour simplifier la présentation on ne considère ici qu'un seul mode de dysfonctionnement. La généralisation de l'approche est une de nos perspectives de recherche.

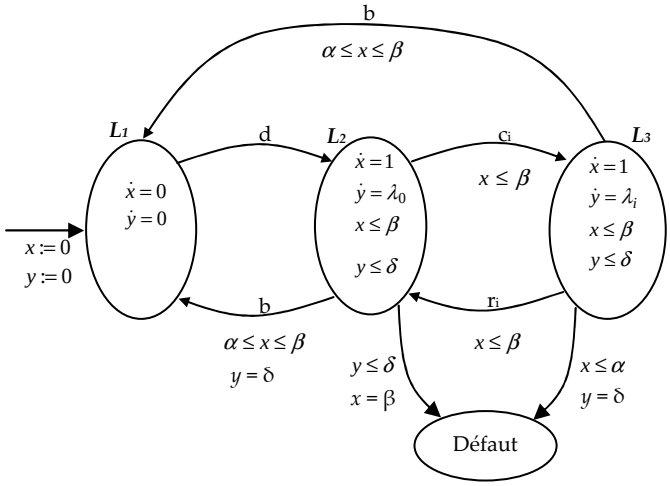

Fig. 9. Modèle de l'automate représentant le processus

Soit $H = (\Gamma, L_1, X, \Sigma_i, A, Inv, Dif)$, l'automate du processus dynamique où :

- $\Gamma = \{L_1, L_2, L3\}$;
- L_1 est le sommet de départ ;
- $X = \{x, y\}$;
- $\Sigma_i = \{b, d, c_i, r_i\}$;
- $A = \{a_1, a_2, a_3, a_4, a_5\}$
- $Inv(L_2) = Inv(L_3) = \{x \le \beta, y \le \delta\}$;
- $Dif(L_1)(x) = 0$, $Dif(L_2)(x) = Dif(L_3)(x) = 1$, $Dif(L_1)(y) = 0$, $Dif(L_2)(y) = \lambda_0$, $Dif(L_3)(y) = \lambda_i$

La dynamique du mode défaillant est $\lambda_i \in [\lambda_{min}, \lambda_{max}]$, avec $\lambda_{max} \ge \lambda_{min} \ge 0$. Le fait de supposer que les dynamiques sont positives est un choix résultant de notre expérience, les processus physiques peuvent s'exécuter plus rapidement ou plus lentement mais rarement en inversant leur sens d'exécution. Prenons l'exemple d'une vanne qui laisse passer un liquide, selon l'ouverture de la vanne le débit peut augmenter ou diminuer. Cependant il n'y a aucune contrainte théorique qui empêche d'avoir des dynamiques négatives.

Les évènements *d* et *b* représentent respectivement le début et la fin d'exécution du processus. Les évènements c_i et r_i sont représentatifs du mode défaillant, en effet l'évènement c_i amène le processus vers le mode défaillant et l'évènement r_i le ramène vers le mode normal.

La variable *x* indique le temps depuis le début d'exécution et la variable *y* reflète l'état du système commandé.

Dans cet automate nous avons les transitions suivantes :

- $a_1 : L_1 \xrightarrow{d} L_2$, représente le début d'exécution de l'état initial vers le mode normal, lié à l'évènement *d* émanant du contrôleur du système.
- $a_2 : L_2 \xrightarrow{s_i} L_3$, représente la migration vers le mode i, suite à l'occurrence de l'évènement c_i, la garde de cette transition est $g_{2,3} = x \le \beta$.

- $a_3 : L_3 \xrightarrow{r_i} L_2$, représente le retour au mode i, suite à l'occurrence de l'évènement r_i, la garde de cette transition est $g_{3,2} = g_{2,3} = x \leq \beta$.

- $a_4 : L_2 \xrightarrow{b} L_1$, représente le retour à l'état initial à partir du mode normale, c'est-à-dire la fin d'exécution du processus qui est indiquée par l'évènement *b*, la garde de cette transition est $g_{2,1} = \alpha \leq x \leq \beta$ *et* $y = \delta$.

- $a_5 : L_3 \xrightarrow{b} L_1$, représente le retour à l'état initial à partir du mode, c'est-à-dire la fin d'exécution du processus qui est indiquée par l'évènement b, la garde de cette transition est $g_{3,1} = \alpha \leq x \leq \beta$ et $y = \delta$.

3. Espace des états atteignables

- Analyse en avant

L'analyse en avant permet de calculer toutes les trajectoires possibles du système, y compris celles qui aboutissent à une alarme.

Les horloges sont initialisées au départ, l'espace de temps à l'entrée du sommet L_2 est $E_2^a = \{x = y = 0\}$, l'évolution de l'état (L_2, E_2^a) est déterminée en utilisant l'analyse en avant. Soit E_2 le successeur discret de la région E_2^a.

On note $E_2 = \text{Suc}_d\ (E_2^a)$. Ce calcul est fait en utilisant le logiciel PHAVER [6].

Dans notre cas on obtient :

$$0 \leq \frac{\lambda_i . x - y}{\lambda_i - \lambda_0} \leq \beta \ \wedge \ y \leq \delta \ \wedge \ 0 \leq x \leq \beta$$

L'espace E_2 résultant de ce calcul est illustré par la figure 9. Cet espace regroupe tous les ensembles atteignables des valeurs *x* et *y* dans L_2 et L_3. Cela veut dire toutes les trajectoires possibles dans ces sommets. Certaines de ces trajectoires sont acceptables car elles permettent de vérifier la garde $g_{1,3}$, mais ce n'est pas le cas de toutes les trajectoires. Notre objectif est de caractériser l'espace atteignable qui permet au système d'atteindre ces objectifs tout en respectant ces contraintes.

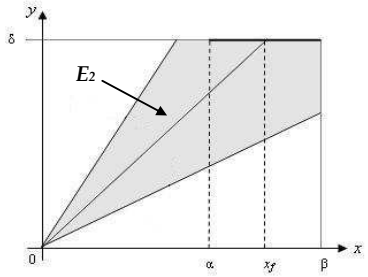

Fig. 10. Espace d'état calculé par l'analyse en avant (E_2)

- Analyse en arrière

Pour éliminer les trajectoires qui ne correspondent pas à une exécution correcte, on effectue une analyse arrière en partant de la région D qui représente le comportement du processus exécuté correctement, c'est-à-dire qui vérifie les contraintes finales décrites par les inégalités :

$$\alpha \leq x \leq \beta \text{ et } y = \delta.$$

Nous calculons l'espace E'_2 qui nous permettra d'atteindre la région D par la méthode d'analyse arrière. Elle revient à inverser l'automate et à faire une analyse avant.
L'espace E'_2 (Figure 11) est décrit par l'inégalité suivante :

$$\delta - \lambda_0 . \beta \leq y - \lambda_0 . x \leq \delta - \lambda_0 \alpha$$

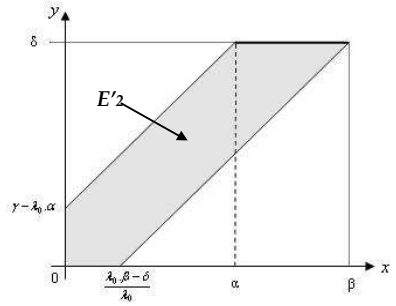

Figure 11. Espace d'état calculé par l'analyse en arrière(E'_2)

Pour obtenir l'espace caractérisant l'évolution correcte du système, on calcule la région *E* qui est l'intersection des régions E_2 et E'_2, (Figure 12)

$$E = E_2 \cap E'_2$$

Tout écart de l'espace *E*, entraînera immédiatement le déclenchement d'une alarme.

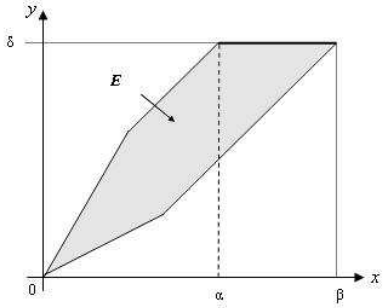

Figure 12. Espace des états du processus commandé (E)

IV. CONCLUSION

Dans cet article nous avons proposé un système de surveillance d'un système commandé, le modèle de surveillance de ce système est un automate hybride linéaire. Il tient compte des changements dynamiques qui peuvent apparaître au cours de l'exécution du processus tout en gardant une dominante événementielle. Les sommets de l'automate représentent les différentes dynamiques que peut avoir le système commandé, le passage entre les modes de fonctionnent est synchronisé par des évènements liés à ces diverses dynamiques. Le comportement autorisé du système est contrôlé par deux variables sur lesquelles sont appliquées des contraintes, ces contraintes exprimées par des inégalités

définissent l'espace acceptable d'évolution du système commandé. Dans le modèle proposé nous ne considérons qu'un seul changement dynamique au cours de l'exécution du processus. Dans nos travaux futurs, il s'agira de généraliser et de formaliser cette approche dans le cas d'un nombre quelconque de modes défaillants. Il s'agira également d'appliquer ces résultats dans le cas de systèmes réels pour prouver toute la pertinence de cette nouvelle approche.

RÉFÉRENCES

[1] A.allahham and Alla, H. Monitoring of timed discrete events systems: Application to manufacturing systems. *In The 32nd Annual conference of IEEE Industrial Electronics Society*, Paris, 2006.

[2] A.allahham and Alla, H. Monitoring of timed discrete events systems with interrups. *IEEE TASE 2009*

[3] Cassez, F. and Larsen, K. (2000). The impressive power of stopwatch. Number 1877, pages 38 - 152. Lecture Notes in *Computer Science, Springer-Verlag*.

[4] Cocquempot, V., Mezyani, T. E., and Staroswieckiy, M. (2004). Fault detection and isolation for hybrid systems using structured parity residuals. In Proceeding of *Asian Control Conference*, ASCC'04.

[5] David R. and Alla H., Du grafcet au réseaux de Petri. Hermes, Paris 1995.

[6] Frehse, G. (2005). Phaver: Algorithmic verification of hybrid systems past hytech. In Proceedings of the Fifth International Workshop on Hybrid Systems : Computation and Control, pages 258{273.

[7] G.Gardey and O.H.Roux and O.H. Roux, "State space computation and analysis of time Petri nets," Theory and Practice of Logic Programming, Special Issue on Specification Analysis and Verification of Reactive Systems, 2006.

[8] Henzinger, T. (1996). The theory of hybrid automata. In proceeding of the 11th Annual IEEE Symposium on Logic in Computer Science, LICS'96, pages 278{292.

[9] Huang, Z., Chandra, V., Jiang, S., and Kumar, R. (1996). Modeling discrete event systems with faults using rules based modeling formalism. Mathematical Modeling of Systems.

[10] R.Alur, C. Courcoubetis, N. Halbwachs, T. A. Henzingerd, P. H. Hod, X. Nicollin, A. Olivero, J. Sifakis and S. Yovine. The algorithmic analysis of hybrid systems. Theoretical Computer Science, 138(1), 1995.

[11] SAVA, A. (2001). Sur la synthèse de la commande des systèmes à événements discrets temporises. Thèse de doctorat de l'Institut National de Polytechnique de Grenoble, INPG.

[12] Tripakis, S. (2002). Fault diagnosis for timed automata. Proceeding 7th International Symposium on Formal Techniques in Real-Time and Fault Tolerant Systems (FTRTFT'02), 2791 of Lecture Notes in Computer Science: 205 - 224.