13ème Congrès AFRIF-AFIA de Reconnaissance des Formes et Intelligence Artificielle, Anger, France, 8-10Janvier 2002, pp 191-200

Diagnostic de pannes sur des systèmes à événements discrets : une approche à base de modèles symboliques

Fault Diagnosing in Discrete Event Systems: A symbolic State-Based Approach

Hervé Marchand[1] Laurence Rozé[2]
[1] INRIA Rennes
[2] INSA Rennes

IRISA, Campus universiaire de Beaulieu, F-35042 RENNES, France
hmarchan@irisa.fr, roze@irisa.fr

Résumé

Nous présentons une technique de supervision s'appuyant sur une représentation symbolique du modèle du système à superviser et de l'outil utilisé pour effectuer le diagnostic (le diagnostiqueur). Basé sur une approche polynomiale, nous utilisons pour représenter le modèle un automate de Moore symbolique qui permet de représenter des systèmes de grandes tailles. La partie combinatoire la plus importante de ce travail se situe au niveau de l'algorithme de construction du diagnostiqueur et a été diminuée par l'introduction d'une réduction symbolique du modèle (modèle quotient par rapport à une relation d'équivalence s'appuyant sur l'histoire du système). L'intérêt de cette réduction symbolique est grande puisqu'elle permet d'effectuer les calculs de la fonction de transition hors ligne lors d'un prétraitement, réduisant ainsi considérablement le travail devant être effectué en ligne.

Mots Clef

Diagnostic, méthodes symbolique, réduction de modèles.

Abstract

We here present a supervision technique based on a symbolic representation of the system to be supervised, as well as the tool to be used to perform the diagnosis : the diagnoser. Based on a polynomial approach, the system is modeled as a symbolic Moor automata allowing realistic system representation. The key point of this method is the use of symbolic reduction model techniques (quotient model w.r.t. a given equivalence relation) allowing the "off-line" computation of the diagnoser, hence reducing the complexity of the "on-line" computation .

Keywords

Diagnosis, symbolic methods, model reduction.

1 Introduction

Depuis de nombreuses années, des recherches ont été effectuées autour du problème du diagnostic, que ce soit pour des systèmes dynamiques, des réseaux de distribution ou des réseaux de télécommunications [16, 1, 15, 12, 3, 5]. La complexité des systèmes et les exigences croissantes en termes de performance et de fiabilité ont nécessité le développement de techniques de diagnostic systématiques.
Étant donné un système, de nombreux aspects du diagnostic peuvent être traités : la détection ou la localisation de pannes ou les deux aspects à la fois (cadre le plus classique du diagnostic), la surveillance du système. La détection revient simplement, à partir d'observations et de connaissances sur le système, à déterminer si ce dernier se comporte normalement ou non. La localisation nécessite de plus de localiser les composants défaillants et le type de défaillance s'étant produit. Par exemple, le diagnostic d'un système, e.g. une voiture, où le but final est la réparation, ne nécessite pas simplement de savoir si le système se comporte normalement ou non, mais de définir exactement quel composant est en panne pour pouvoir effectuer la réparation. Dans d'autres systèmes, tel que par exemple les réseaux de télécommunications, le but du diagnostic est simplement d'effectuer une surveillance : les composants peuvent tomber en panne puis revenir automatiquement en fonctionnement (c'est le cas par exemple d'une machine qui "*reboot*"). Le diagnostic consiste alors simplement à surveiller le réseau et à détecter les pannes persistantes.
Dans cet article, nous traitons le problème classique de diagnostic qui consiste à la fois à détecter et localiser une panne. Notre but est, dans un premier temps, de détecter qu'un problème s'est produit et, dans un second temps, de définir exactement quelle panne s'est produite. Nous nous plaçons dans un cadre où les pannes sont permanentes. Nous supposons de plus que les états du système sont non observables : seuls certains événements sont observables. En particulier aucune information n'est disponible sur l'état

initial du système. Le diagnostic consiste, à partir des événements observables, à détecter les pannes pouvant s'être produites.

Les travaux présentés dans cet article s'appuient sur la méthodologie présentée dans [8] et [16], qui consiste à transformer le modèle du système en un automate, appelé diagnostiqueur. Chaque état du diagnostiqueur contient de l'information sur l'état du système et sur les pannes s'étant produites pour arriver dans cet état. Le diagnostiqueur est construit à partir d'un modèle de tout le système, ce qui induit un grave problème de complexité pour de gros systèmes tels que les réseaux de télécommunications. Pour résoudre ce problème de taille du modèle, les solutions actuellement étudiées sont l'utilisation d'une approche décentralisée [3, 12, 5] ou l'utilisation d'une approche générique [15]. Dans ces approches, le système doit vérifier des propriétés spécifiques : il doit être décrit à l'aide d'un niveau structurel et comportemental. Le niveau structurel représente les composants et leurs interconnexions. Le niveau comportemental est constitué d'un ensemble de modèles. Chaque composant du niveau structurel est associé à un modèle du niveau comportemental. Le point fort de l'approche décentralisée est que le modèle de tout le système n'est jamais construit. Toutefois, une telle approche est inutilisable si le système ne peut être décomposé dans les deux niveaux structurel et comportemental. Pour l'approche générique, non seulement le modèle doit être décrit à l'aide des deux niveaux, mais en plus plusieurs composants du niveau structurel doivent être associés au même modèle comportemental.

Nous proposons une nouvelle méthodologie pour résoudre le problème de la taille du modèle en s'appuyant toujours sur un modèle global de tout le système. L'idée de base est d'utiliser un modèle et un diagnostiqueur symboliques. Dans cet article, nous nous appuyons sur la méthodologie proposée dans [8, 16]. Les transitions du diagnostiqueur sont étiquetées uniquement par les événements observables. Chaque état est associé au(x) panne(s) ayant du se produire pour atteindre cet état. Toutefois par rapport à [8], notre approche fournit des algorithmes symboliques génériques s'appuyant sur une représentation implicite du système, codée par une relation polynomiale, elle-même codée par des diagrammes de décision p-aires (p-DD), extension des diagrammes de décision binaires (BDD pour Binary Decision Diagrams [2]). Le but de cette représentation est de ne pas énumérer les états et les transitions. Un ensemble d'états sera toujours représenté par une fonction polynomiale caractérisant les éléments de cet ensemble, les calculs ensemblistes se faisant alors, non pas à partir des ensembles eux-mêmes, mais sur les fonctions polynomiales les caractérisant.

Notre approche est la suivante : le système sur lequel le diagnostic doit être effectué est modélisé par un automate de Moore symbolique. Les états sont caractérisés par des variables d'états, les événements (internes ou de sortie) par des variables événements. Seules les variables représentant les événements de sortie sont observables. Les pannes sont supposées permanentes et associées aux états. Le diagnostiqueur est automatiquement construit à partir du modèle du système. Ce dernier fournit une estimation de l'état du système. À partir de cette estimation, il est possible de déterminer la ou les pannes possibles du système. Deux approches sont proposées. Dans la première, le diagnostiqueur est dérivé "en-ligne", au fur et à mesure de l'arrivée d'observables. Dans la seconde approche, basé sur les travaux de [8] dans un cadre explicite, le diagnostiqueur est vu comme un quotient du modèle initial du système, à partir d'une relation d'équivalence qui prend en compte l'histoire du système. Les calculs permettant de calculer ce nouveau diagnostiqueur sont réalisées grâce à des techniques symboliques évitant l'énumération de l'espace d'états et donc une éventuelle explosion combinatoire.

2 Le modèle du système

Nous introduisons les modèles, les automates symboliques de Moore, sur lesquels des analyses symboliques seront réalisées. Mais dans un premier temps, nous présentons le modèle mathématique sur lequel est basé notre modèle.

2.1 Le modèle mathématique

Soit $\mathbb{Z}/_{p\mathbb{Z}} = \{0, 1, \ldots, p-1\}$, avec p premier, et Z un ensemble fini de k variables distinctes $Z_1, ..., Z_k$ [1] prenant leurs valeurs dans $\mathbb{Z}/_{p\mathbb{Z}}$. L'anneau [2] des polynômes en les variables $Z = (Z_1, \ldots Z_l)$ à coefficients dans $\mathbb{Z}/_{p\mathbb{Z}}$ sera noté $\mathbb{Z}/_{p\mathbb{Z}}[Z]$.

Étant donné un élément de $\mathbb{Z}/_{p\mathbb{Z}}[Z]$, $P(Z_1, Z_2, \ldots, Z_k)$ (noté $P(Z)$), il est possible de lui associer son ensemble de solutions $Sol(P) \subseteq (\mathbb{Z}/_{p\mathbb{Z}})^m$:

$$Sol(P) \stackrel{\text{def}}{=} \{(z_1, ..., z_k) \in (\mathbb{Z}/_{p\mathbb{Z}})^k | P(z_1, ..., z_k) = 0\} \quad (1)$$

On peut remarquer que dans $\mathbb{Z}/_{p\mathbb{Z}}[Z]$, pour tout $P(Z) \in \mathbb{Z}/_{p\mathbb{Z}}[Z]$, nous avons $Sol(P) = Sol(P + (Z_i^p - Z_i))$. Il est donc naturel d'introduire une abstraction modulo la $\equiv$-équivalence sur les polynômes, où $P_1 \equiv P_2$ signifie $Sol(P_1) = Sol(P_2)$. À cet effet, nous introduisons l'anneau quotient $A[Z] = \mathbb{Z}/_{p\mathbb{Z}}[Z]/_{<Z^p - Z>}$, où tous les polynômes $Z_i^p - Z_i$ sont "projetés" sur 0 (noté $Z^p - Z = 0$). $A[Z]$ peut être vu comme l'ensemble des fonctions polynomiales dont les coefficients sont dans $\mathbb{Z}/_{p\mathbb{Z}}$ et pour lesquelles le degré en chacune des variables est plus petit ou égal à $(p-1)$. Ceci est très important d'un point de vue algorithmique. De plus, [4] a montré comment calculer un représentant de $Sol(P)$, i.e. pour chaque classe d'équivalence $\equiv$, appelé *générateur canonique* (un tel polynôme

1. X, Y, etc sont des ensembles de variables définis de manière similaire. Par la suite X (resp. Y) représentera l'ensemble des variables d'états (resp. variables d'événements du système).

2. Un anneau commutatif R est donné par un ensemble d'éléments et deux opérations, + et *, toutes deux commutatives, distributives, associatives et closes dans R. + et * possèdent des éléments neutres, resp. 0 et 1 respectivement.

existe toujours). Par la suite, tout polynôme sera implicitement réduit modulo $\equiv$.

À partir de ce modèle mathématique, il est possible de définir sur les polynômes l'ensemble des opérations de base sur les ensembles associés (inclusion, intersection, union, complémentaire).

Propriété 1 [11] *Soient* $P_1, P_2, P \in \mathbb{Z}/_{p\mathbb{Z}}[Z]$.

- $Sol(P_1) \subseteq Sol(P_2)$ *si* $(1 - P_1^{p-1}) * P_2 \equiv 0$. *(*inclusion*)*
- $Sol(P1) \cap Sol(P_2) = Sol(P_1 \oplus P_2)$ *(*intersection*), où*

$$P_1 \oplus P_2 \stackrel{def}{=} (P_1^{p-1} + P_2^{p-1})^{p-1} \quad (2)$$

- $Sol(P_1) \cup Sol(P_2) = Sol(P_1 * P_2)$ *(*union*)*
- $(\mathbb{Z}/_{p\mathbb{Z}})^m \setminus Sol(P) = Sol(1 - P^{p-1})$ *(*complémentaire*).* ◇

Notations : Par la suite, on notera

- $\overline{P} = 1 - P^{p-1}$, i.e. $Sol(\overline{P}) = (\mathbb{Z}/_{p\mathbb{Z}})^m \setminus Sol(P)$.
- $(P_1 \Rightarrow P_2) = \overline{P_1} * P_2$ correspond à l'ensemble

$$\{z \in (\mathbb{Z}/_{p\mathbb{Z}})^k | P_1(z) = 0 \Rightarrow P_2(z) = 0\}.$$

Finalement, on introduit les *abstractions existentielle / universelle* des polynômes sur un ensemble de variables. Soit $P \in \mathbb{Z}/_{p\mathbb{Z}}[Z]$, on notera $\exists Z_i P$ le polynôme

$$\exists Z_i P \stackrel{\text{def}}{=} P|_{Z_i=1} * P|_{Z_i=2} * \ldots * P|_{Z_i=p}, \quad (3)$$

où $P|_{Z_i=v}$ est obtenu à partir de P en instanciant la variable Z_i par la valeur v. On obtient alors :

$$Sol(\exists Z_i P) = \{(z_1, ..., z_{i-1}, z_{i+1}, ..z_n) / \exists z_i \; P(z_1, ..., z_i, ..z_n) = 0\}.$$

De plus, quand $\tilde{Z} \subset Z$ est de la forme $\{Z_{i_1}, ..., Z_{i_r}\}$, on écrit $\exists \tilde{Z} P$ pour $\exists Z_{i_1} ... \exists Z_{i_r} P$. De manière similaire, il est possible de définir l'opération duale, basée sur l'élimination du quantificateur universel : $\forall Z_i P$ est calculé par

$$\forall Z_i P \stackrel{\text{def}}{=} P|_{Z_i=1} \oplus P|_{Z_i=2} \oplus \ldots \oplus P|_{Z_i=p} \quad (4)$$

Les solutions de $\forall Z_i P$ sont de la forme $(z_1, ..., z_{i-1}, z_{i+1}, ..., z_k)$ t.q. $\forall z_i, (z_1, ..., z_{i-1}, z_i, z_{i+1}, ..., z_k) \in Sol(P)$.

L'intérêt du modèle mathématique que nous venons de présenter réside dans le fait que nous manipulons les polynômes au lieu des ensembles des solutions de ces polynômes (C.f. Équation (1)), évitant ainsi l'énumération des solutions. De cette manière, il est possible de réaliser des opérations sur les ensembles tout en restant dans le domaine des polynômes (C.f. Propriété 1). De plus, cette approche polynomiale (et donc symbolique) a l'avantage de se situer à un niveau d'abstraction intéressant permettant la manipulation d'ensembles (d'états, d'événements) dans lequel les algorithmes peuvent entièrement être exprimés.

Remarque 1 *Une approche classique de ce formalisme est* $\mathbb{Z}/_{2\mathbb{Z}}[Z]$ *(i.e. les booléens). Dans ce cadre, les fonctions polynomiales représentent alors des prédicats; l'opération* $\oplus$ *correspond au* $\wedge$, $*$ *est le* $\vee$, *tandis que l'opération* $\overline{P}$ *correspond à la négation. Le cas* $p = 3$ *est utile pour encoder le fait que des signaux booléens (vus comme des variables) peuvent être présents ou non l'un par rapport à l'autre. Dans ce cadre, les trois valeurs possibles d'un signal booléen (i.e.* présent *et* vrai, présent *et* faux, *ou* absent*) sont codés par (présent et vrai* $\rightarrow 1$, *présent et faux* $\rightarrow -1$, *et absent* $\rightarrow 0$*). Cette approche a été mise en œuvre dans l'outil Sigali, permettant notamment la vérification ou la synthèse de contrôleurs sur des systèmes issus de programmes Signal [10]). Les cas* $p > 3$ *peuvent également s'avérer intéressants pour coder des systèmes avec buffers ou modélisés à l'aide de types énumérés [6].* ◇

Encodage des polynômes sous forme de p-DD.
Il s'avère que la meilleure implémentation connue (tant du point de vue des calculs que de la place mémoire) des polynômes sur $\mathbb{Z}/_{p\mathbb{Z}}$ est basée sur la décomposition de ceux-ci en fonction des polynômes de Lagrange, menant à une implémentation des polynômes sous forme de p-DD (p-ary Decision Diagram). Une approche classique est la décomposition de Shannon pour le cas $p = 2$ qui associe des BDDs (Binary Decision Diagrams) aux fonctions booléennes [2] . Pour le cas $p = 3$, les polynômes sont encodés sous forme de TDD (Ternary Decision Diagrams [11]), structure utilisée pour l'implémentation des polynômes dans Sigali. Une description plus détaillée des TDD est donnée en Annexe.

2.2 Modélisation du système

Les systèmes que nous étudions peuvent être décrits par des automates à états finis. Le changement d'état du système est lié à l'occurrence d'événements sur celui-ci. Tous les événements n'ont pas le même statut : certains sont observables, d'autres non (i.e leur occurrence n'est pas connue et ne conduit à aucune manifestation directement observable au niveau du système). Dans notre approche, le système sera vu comme une boîte noire. L'évolution interne du système (i.e. son état courant et les événements internes) n'est observable que par l'intermédiaire d'événements de sortie appelés *observations*.

Les Automates de Moore Symboliques.

Définition 1 *Un* Automate de Moore Symbolique *(AMS) est donné par une structure* $S = (X, X', Y, Y_o, I, \mathcal{T}, \mathcal{O})$ *où*

- $X = \{X_1, ..., X_n\}$ *et* $X' = \{X'_1, ..., X'_n\}$ *sont deux ensembles de* variables d'états *et codent respectivement l'état courant et suivant du système.*
- $Y = \{Y_1, ..., Y_l\}$ *est un vecteur de variables, appelées* variables événements, *qui servent à coder les événements internes du système.*

- $Y_o = \{Y_{o_1}, ..., Y_{o_k}\}$ *est un ensemble de variables codant les observations du système.*
- I *est une relation dans* $\mathbb{Z}/_{p\mathbb{Z}}[X]$ *qui caractérise les états initiaux du système.*
- $\mathcal{T}(X, Y, X') \in \mathbb{Z}/_{p\mathbb{Z}}[X, Y, X']$ *est une relation qui décrit les transitions légales du système.*
- $\mathcal{O}(X, Y_o)$ *est une relation caractérisant les observations du système en fonction de l'état courant du système. S'il y a une unique solution, cette relation sera vue comme une fonction* $Y_o = \mathcal{O}(X)$ [3] •

Le comportement d'un tel système S est le suivant : les états initiaux de S sont donnés par $Sol(I(X))$. Au cours de l'évolution, étant donnés un état $x \in (\mathbb{Z}/_{p\mathbb{Z}})^n$ accessible et un état cible $x' \in (\mathbb{Z}/_{p\mathbb{Z}})^n$, l'ensemble $Sol(\mathcal{T}(x, Y, x'))$ représente tous les événements admissibles entre l'état x et l'état x'. Ainsi, une transition entre x et x' via l'événement y est possible dès lors que le triplet (x, y, x') est solution de $\mathcal{T}$. De plus, lorsque le système atteint un état x, il "produit" une observation y_o, t.q. $y_o = \mathcal{O}(x)$. Par la suite pour dire que $\mathcal{T}(x, y, x') = 0$ et $\mathcal{O}(x') = y_o$, nous écrirons parfois :

$$x \xrightarrow{y} x' \quad (\mathcal{O}(x') = y_o)$$

avec $x = (x_1, ..., x_n)$, $x' == (x'_1, ..., x'_n)$, $y = (y_1, ..., y_l)$ et $y_o = (y_{o_1}, ..., y_{o_k})$.

Remarque 2 *Chaque variable d'états peut être vue comme un "modèle" abstrait d'un composant du système que l'on considère, les événements internes servant quant à eux à encoder les communications entre ces différents composants.* ◇

Dans la suite de cet article, les variables événements et d'états sont supposées être inobservables. Ainsi, la détection d'une panne (si elle se produit) ne pourra se faire que par l'intermédiaire des occurrences des observations.

Composition de systèmes.
Tout comme dans le cas explicite, la classe des automates de Moore symbolique est pourvue des opérations classiques entre systèmes de transitions. Parmi ces opérations, la *composition parallèle* et le *masquage d'événements internes* jouent un rôle important dans la spécification de systèmes complexes. La composition parallèle d'AMS impose la compatibilité de valeurs entre variables internes communes aux systèmes (d'un point de vue explicite, cette opération correspond à la composition parallèle synchrone comme définie en Signal [9] ou Lustre [7]). Toutefois, l'approche symbolique évite, dans une certaine mesure, l'explosion combinatoire potentielle due à la composition synchrone).
La composition parallèle, notée $S_1 \mid S_2$ peut se définir par

$$S_1 \mid S_2 \stackrel{\text{def}}{=} (X^1 \cup X^2, X'^1 \cup X'^2, Y, Y_o, \mathcal{T}_1 \oplus \mathcal{T}_2, \mathcal{O}_1 \oplus \mathcal{O}_2) \quad (5)$$

3. Dans ce cas, le polynôme $y_o - \mathcal{O}(X)$ correspondra à l'ensemble des états x t.q. $y_o = \mathcal{O}(X)$.

Le masquage d'événements consiste à s'abstraire de variables internes, i.e. à rendre interne des communications entre composants du système global. Le masquage d'événements vis-à-vis d'un sous-ensemble de variables internes, $Y_i \subseteq Y$, est obtenu en considérant le système

$$S \setminus \{Y_i\} \stackrel{\text{def}}{=} (X, X', Y \setminus \{Y_i\}, Y_o, \exists Y_i \mathcal{T}(X, Y, X'), \mathcal{O}) \quad (6)$$

D'autres types de composition, telle que la composition asynchrone (*interleaving*) ou encore la composition *fortement synchrone* [4] peuvent être également considérées. D'un point de vue abstraction, nous renvoyons à [13] pour plus de détails. Il est également possible de considérer des opérations permettant le séquencement de systèmes.

Modélisation des pannes.
On suppose que le modèle du système que nous considérons inclut le comportement global du système (avec en particulier un modèle des pannes pouvant s'y produire). $\mathcal{F} = \{N, Panne_1, \ldots Panne_l\}$ représente l'ensemble des modes de pannes du système, où N correspond au mode normal. Chaque mode de pannes $Panne_i$ correspond au même type de pannes pour un instrument, un capteur, etc (voir [16] pour plus de détails) [5]. Dans notre modèle, les pannes sont supposées être permanentes. De plus, comme le système global sur lequel le diagnostic doit s'opérer a été obtenu par composition de systèmes élémentaires, le système peut se retrouver à la fois dans plusieurs modes de pannes. Par la suite, on notera $\mathcal{K} = 2^{\mathcal{F}}$ l'ensemble des parties des modes de pannes. Ainsi, $\mathcal{K} = \{F_I, I \subseteq [0..l]\}$ t.q. $F_I = \{Panne_i, i \in I\}$ avec $Panne_0 = N$ et $F_o = \{Panne_0\}$.
On supposera de plus qu'un mode de pannes (ou un ensemble) est associé à un ensemble d'états. Ainsi, l'espace d'états $(\mathbb{Z}/_{p\mathbb{Z}})^n$ peut être partitionné relativement à $\mathcal{K}$. De manière symbolique, ce partitionnement est donné par un ensemble de polynômes $(P_I)_{I \subseteq [0...l]}$, t.q. :

$$\prod_{I \subseteq [0...l]} P_I(X) = 0. \quad (7)$$

En d'autres termes, $(\mathbb{Z}/_{p\mathbb{Z}})^n = \bigcup_{J \subseteq [0...l]} Sol(P_I)$. $x \in Sol(P_I)$ signifie que les pannes $F_I = \{Panne_i,\ i \in I\}$ se sont produites dans le système (si $x \in Sol(P_0)$, alors le système est en mode normal). (7) assure que tout état est soit en mode normal, soit dans un ensemble de modes de pannes. Du fait de l'hypothèse de pannes permanentes, il est facile de doter les polynômes $(P_J)_{J \subseteq [0...l]}$ d'un ordre partiel $\prec$ t.q. $P_I \prec P_J$ signifie que $F_I \subseteq F_J$. Basé sur cet ordre partiel, on peut voir qu'il existe un "chemin" entre deux états x et x' (t.q. $x \in Sol(P_I), x' \in Sol(P_J)$) seulement si $P_I \prec P_J$.
De plus, si le système est dans un état x, alors il ne peut se trouver dans deux ensembles de modes de pannes dif-

4. Cette composition correspond à l'intersection des comportements des deux systèmes.

5. À un niveau de détails moins élevé, les pannes peuvent être décrites par l'intermédiaire de variables d'états / événements.

férents. Ceci se traduit par $P_I(x) \oplus P_J(x) = 1, \forall I, J \subseteq [0..l]$ ou, en d'autres termes, $Sol(P_I) \cap Sol(P_J) = \emptyset$.
La Figure 1) résume ce codage des modes de pannes.

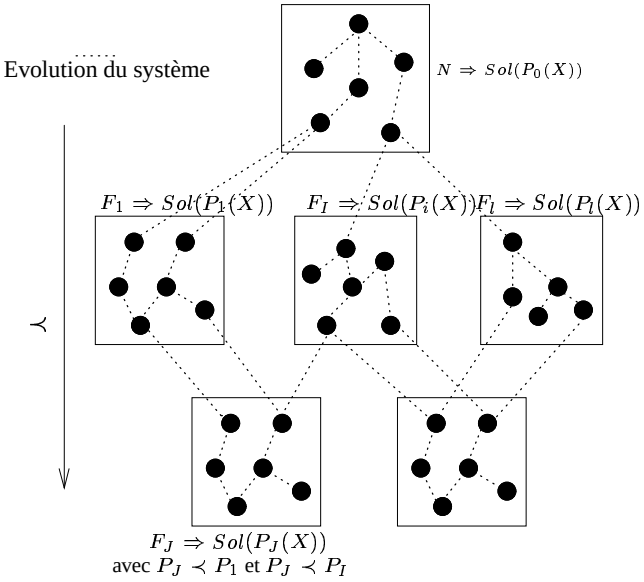


FIG. 1 – *Partition des états en fonction des pannes*

Pour clarifier les notations, on renomme les F_I, t.q. $\mathcal{K} = \{N, F_1, ..., F_m\}$ corresponde à l'ensemble des parties de modes de pannes et t.q. les $P_o, ..., P_m$ correspondent aux polynômes associés à ces parties. De plus, par abus de langage, on dira que le système est dans le mode de pannes F_i au lieu de dire que le système est dans l'ensemble des modes de pannes correspondant à l'ensemble F_i. Par la suite, nous aurons besoin de comparer le statut de deux états.

Proposition 1 *Deux états x et x' sont dans le même mode de pannes dès lors que $K_g(x, x') = 0$, avec*

$$K_g(X, X') = \prod_{i=0}^{m} P_i(X) \oplus P_i(X') \tag{8}$$

$\diamond$

On rappelle que $P_i(x) \oplus P_i(x') = 0$ si et seulement si $P_i(x) = 0$ et $P_i(x') = 0$. Ainsi $K_g(X, X') = 0$ dès lors qu'il existe un i permettant d'annuler $P_i(x) \oplus P_i(x')$. De plus, dans ce cas, du fait du partitionnement, il ne peut exister au plus qu'un seul i.

Estimation des Pannes.
Étant donné le caractère inobservable d'une partie des variables du système, nous devrons travailler sur une estimation de son état courant (caractérisé par un ensemble d'états possibles, dans lesquels le système pourra avoir évolué). D'un point de vue diagnostic, il faudra donc faire correspondre à cette estimation un ensemble de modes de pannes dans lequel le système pourra se trouver.

Plus formellement, nous introduisons une fonction $\overrightarrow{F}$ qui associe à chaque ensemble d'états E, décrit par un polynôme $P(X)$ (t.q. $Sol(P) = E$), une estimation des modes de pannes.

$$\begin{aligned} \overrightarrow{F} : \mathbb{Z}/_{p\mathbb{Z}}[X] &\longrightarrow (\mathbb{Z}/_{2\mathbb{Z}})^m \\ P(X) &\longmapsto \overrightarrow{F}(P) = \begin{bmatrix} \exists X\{P_0(X) \oplus P(X)\} \\ \cdots \\ \exists X\{P_i(X) \oplus P(X)\} \\ \cdots \\ \exists X\{P_m(X) \oplus P(X)\} \end{bmatrix} \end{aligned} \tag{9}$$

$\overrightarrow{F}(P)$ correspond à l'ensemble des pannes dans lesquelles le système peut se trouver sachant que celui-ci peut avoir évolué dans un des états de $Sol(P) = E$.
Intuitivement, à chaque polynôme P (et par conséquent à chaque ensemble d'états), on associe un vecteur de booléens $\overrightarrow{F}(P)$. Chaque composant de ce vecteur, e.g. $\overrightarrow{F}(P)_i$, est égal à 0 dès lors qu'il existe un état $x \in Sol(P)$, t.q. $x \in Sol(P_i)$. Cela signifie que le système peut se trouver dans le mode de pannes F_i.
De plus, un mode de pannes sera sûrement diagnostiqué dès lors qu'un unique composant de ce vecteur sera nul.

3 Le diagnostiqueur

Dans la section précédente, nous avons décrit le modèle du système : c'est un automate symbolique où les pannes sont associées aux états de l'automate. Les transitions incluent à la fois des variables observables et non observables. Toutefois, seules les variables observables sont disponibles pour la tâche de diagnostic. Une première méthode de diagnostic s'appuie sur des techniques de simulation, comme réalisé dans [1]. Mais ceci est combinatoirement coûteux pour effectuer du diagnostic en ligne.
Une autre solution consiste à construire "hors-ligne" un nouvel automate, appelé diagnostiqueur. Ses transitions se font seulement sur réception des observations du système et ses états incluent des informations sur les fautes pouvant s'être produites dans le système. Le diagnostic est alors réalisé "en ligne" en fonction de l'évolution de ce diagnostiqueur.

Dans notre approche, seules les variables d'observations sont visibles de l'extérieur. Ainsi la présence d'une panne dans le système ne peut se faire que par les séquences d'observations. Toutefois, nous supposons que seules les observations distinctes sont visibles. Ainsi une transition entre deux états ayant la même observation ne pourra être détectée par un acteur extérieur (e.g. le diagnostiqueur). En d'autres termes, supposons que la séquence d'observations suivante se produise : $y_{o_1}^{n_1} y_{o_2}^{n_2} \ldots y_{o_i}^{n_i}$, avec $y_{o_i} \neq y_{o_{i+1}}$. Alors, le diagnostiqueur observera seulement la séquence $y_{o_1} y_{o_2} \ldots y_{o_i}$ (voir Figure 2).

Calcul d'une nouvelle relation de transitions.
Dans un premier temps, nous introduisons une nouvelle re-

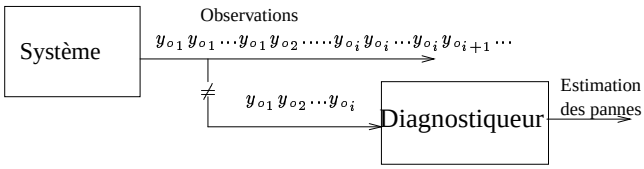


FIG. 2 – *Observation du système par le diagnostiqueur*

lation de transitions pour le système basé sur la distinction entre observations.

Définition 2 *Soient x et x' deux états du système, avec $x \neq x'$, alors $x \Longrightarrow x'$ si $\mathcal{O}(X) \neq \mathcal{O}(X')$ et soit $x \longrightarrow x'$ soit $\exists l \geq 2, x_2, \ldots x_{l-1}, y_1, \ldots y_{l-1}$, t.q.*

$$\overset{\mathcal{O}(x)}{\overset{\nearrow}{x}} \xrightarrow{y_1} \overset{\mathcal{O}(x)}{\overset{\nearrow}{x_2}} \xrightarrow{y_2} \ldots \xrightarrow{y_{l-2}} \overset{\mathcal{O}(x)}{\overset{\nearrow}{x_{l-1}}} \xrightarrow{y_{l-1}} \overset{\mathcal{O}(x')}{\overset{\nearrow}{x'}},$$

avec $\mathcal{O}(x) = \mathcal{O}(x_i), 1 \leq i \leq l-1$. •

Intuitivement, deux états x et x' seront en relation s'il existe une trajectoire dans le système partant de x et arrivant en x' sans distinction d'observations entre ces deux états.

Regardons maintenant comment la relation $\Longrightarrow$ peut être caractérisée sous forme d'un polynôme $R(X, X')$. Le calcul de R est réalisé en deux étapes.

Dans un premier temps, nous calculons l'ensemble des paires d'états (x, x') pour lesquelles il existe une trajectoire dans le système de la forme :

$$\overset{\mathcal{O}(x)}{\overset{\nearrow}{x}} \xrightarrow{y_1} \overset{\mathcal{O}(x)}{\overset{\nearrow}{x_2}} \xrightarrow{y_2} \ldots \xrightarrow{y_{l-2}} \overset{\mathcal{O}(x)}{\overset{\nearrow}{x_{l-1}}} \xrightarrow{y_{l-1}} \overset{\mathcal{O}(x)}{\overset{\nearrow}{x'}}, \qquad (10)$$

avec $\mathcal{O}(x) = \mathcal{O}(x_i) = \mathcal{O}(x'), \forall 1 \leq i \leq l-1$. Le calcul par point-fixe suivant permet d'obtenir cet ensemble de paires :

$$\mathcal{T}_{\mathcal{O}}(X, X') =$$

$$\begin{cases} \mathcal{T}_{\mathcal{O}_0}(X, X') = \exists Y \{\mathcal{T}(X, Y, X') \oplus (\mathcal{O}(X) - \mathcal{O}(X'))\} \\ \\ \mathcal{T}_{\mathcal{O}_{j+1}}(X, X') = \exists X'' \{\mathcal{T}_{\mathcal{O}_j}(X, X'') * \\ \qquad (\exists Y (\mathcal{T}(X'', Y, X') \oplus (\mathcal{O}(X) - \mathcal{O}(X'))))\} \end{cases}$$

Ce point-fixe s'arrête car les ensembles associés aux polynômes calculés à chaque itération induisent une suite d'ensembles croissante (l'espace d'états est fini). Il est alors facile de vérifier que si $\mathcal{T}_{\mathcal{O}}(x, x') = 0$, il existe dans S une trajectoire de la forme (10).

Partant de $\mathcal{T}_{\mathcal{O}}$, $\Longrightarrow$ est caractérisée par le polynôme R t.q.

$$R(X, X') = \exists X'' (\mathcal{T}_{\mathcal{O}}(X, X'') \oplus \exists Y (\mathcal{T}(X'', Y, X') \oplus (1 - (\mathcal{O}(X) - \mathcal{O}(X'))^{p-1}) \qquad (11)$$

Notons que $(1-(\mathcal{O}(X)-\mathcal{O}(X'))^{p-1}) = 0$ est vrai dès lors que $\mathcal{O}(X) \neq \mathcal{O}(X')$. On peut alors montrer la proposition suivante :

Proposition 2 $x \Longrightarrow x' \Leftrightarrow R(x, x') = 0$. ◇

Preuve : Si $R(x, x') = 0$ alors il existe x'' t.q. x et x'' soient reliés par une trajectoire de la forme (10) ($\mathcal{T}_{\mathcal{O}}(X, X'')$). Isl existe de plus un événement interne y faisant transiter le système de x'' à x' t.q. $\mathcal{O}(x') \neq \mathcal{O}(X'') = \mathcal{O}(x)$ (dernière partie de (11)). La réciproque est identique. ◇

Calcul du diagnostiqueur : une première approche.

En général, étant donnée une séquence d'observations, e.g. $s_o = y_{o_1} \ldots y_{o_i} \ldots$, l'état dans lequel se trouve potentiellement le système ne peut être déterminé de manière unique. L'idée du diagnostiqueur est de savoir, après chaque occurrence d'une observation, tous les états possibles dans lesquels le système peut potentiellement se trouver et, pour chacun de ces états, les pannes qui se sont produites pour atteindre cet ensemble d'états.

Dans [16, 3, 15] le diagnostiqueur est un automate. Ses transitions sont étiquetées par des événements observables. Ses états contiennent des informations à la fois sur l'état du système et sur les pannes s'étant produites pour arriver dans cet état. Ils sont de la forme $(etat, ens\_pannes)$ où $etat$ est un état possible du système et ens_pannes les pannes ayant du se produire pour arriver dans l'état $etat$. La tâche de diagnostic consiste alors à mettre à jour l'ensemble des états possibles du diagnostiqueur au fur et à mesure de l'arrivée d'événements observables.

Dans notre approche, les pannes sont directement associées aux états (par l'intermédiaire des variables d'états). Il est donc suffisant de ne conserver que l'ensemble des états possibles du système pour en déduire l'ensemble des modes de pannes dans lesquels se trouve potentiellement le système (i.e. en fonction de (9)). Ainsi, l'état du diagnostiqueur est caractérisé par un polynôme $\widehat{X}_i$ qui a pour solution l'ensemble des états possibles du système après i observations.

Donnons maintenant une définition plus formelle d'un diagnostiqueur. Il est construit incrémentalement comme suit :

$$\begin{cases} \widehat{X}_0(X) & = & 0 & (i) \\ \widehat{X}_1(X) & = & y_{o_1} - \mathcal{O}(X). & (ii) \\ \widehat{X}_{i+1}(X) & = & \mathcal{T}_d(\widehat{X}_i(X), y_{o_{i+1}}), & (iii) \end{cases} \qquad (12)$$

avec

$$\mathcal{T}_d(\widehat{X}_i, Y_o) = \exists X' \{(Y_o - \mathcal{O}(X)) \oplus R(X', X) \oplus \widehat{X}_i(X')\} \qquad (13)$$

Initialement, $\widehat{X}_0(X) = 0$ (i.e. tout l'espace d'états est solution); on suppose en effet que le diagnostiqueur peut être mis en marche alors même que le système l'était déjà. Après l'occurrence d'un événement y_{o_1}, le système peut alors avoir évolué dans un état x ayant y_{o_1} comme observation (C.f. (ii)). Supposons maintenant que l'état du diagnostiqueur soit donné par $\widehat{X}_i(X)$ et qu'il reçoive du système l'observation $y_{o_{i+1}}$. Alors, la nouvelle estimation (i.e. le nouvel état du diagnostiqueur) $\widehat{X}_{i+1}(X)$ est donnée par l'ensemble des états ayant $y_{o_{i+1}}$ comme observation et qui peuvent être atteints par au moins une trajectoire de la forme

$\Longrightarrow$ partant de $\widehat{X}_{i+1}(X)$ dont l'observation finale est $y_{o_{i+1}}$ (C.f. (iii) et l'équation (13)). Sachant que le système a évolué dans un des états de $\widehat{X}_{i+1}(X)$, l'estimation des pannes est calculée en fonction de (9) et est égale à $\overrightarrow{F}(\widehat{X}_{i+1}(X))$.

Toutefois, même si $\Longrightarrow$ (i.e. R) est calculé hors-ligne, de manière à obtenir l'estimation des états courants et l'estimation des pannes possibles, $\mathcal{T}_d$ doit être calculé après chaque occurrence d'un événement y_o (i.e. en ligne). Ce calcul est évidemment coûteux (et cela même si les calculs sont faits symboliquement). Pour résoudre ce problème, une solution serait de calculer hors-ligne la relation de transition du diagnostiqueur comme fait par [8], tout en préservant *l'aspect symbolique* de notre approche. C'est ce que nous présentons dans la section suivante.

Calcul du diagnostiqueur par réduction de modèle.

L'idée dans cette section est de calculer un modèle quotient du système modulo une relation d'équivalence $\sim$ sur les états du système qui préserve certaines caractéristiques importantes du modèle (même mode de pannes et même observation) [8].

Définition de $\sim$. Intuitivement, deux états seront dits équivalents vis à vis de $\sim$, dès lors qu'ils ont la même observation, qu'ils appartiennent au même mode de pannes, et si pour une séquence d'observations donnée les états atteints possèdent les mêmes caractéristiques. Ceci nous amène à la définition suivante de $\sim$.

Définition 3 *Soit x_1 et x_2 deux états du système, alors $x_1 \sim x_2$ si*

1. *pour tout événement y_o et pour tout état x'_1 admissible par $\Longrightarrow$ respectant y_o, i.e. tel que .*

$$x_1 \Longrightarrow x'_1 \wedge y_o = \mathcal{O}(x'_1),$$

il existe un état x'_2, t.q. $x_2 \Longrightarrow x'_2$, avec

$$y_o = \mathcal{O}(x'_2),\ \overrightarrow{F}(x'_1) = \overrightarrow{F}(x'_2) \text{ et } x'_1 \sim x'_2$$

2. *vice-versa.*

En d'autres termes, si deux états x et x' sont équivalents, alors la séquence d'observations produite à partir de l'un ou l'autre de ces états sera la même. Il sera également vrai que pour toutes les séquences d'observations possibles initialisées dans l'un ou l'autre de ces états, l'estimation des pannes sera identique.

Nous nous intéressons maintenant au calcul de la relation d'équivalence $\sim$. Elle peut être caractérisée par le polynôme $\mathcal{R}_\sim(X_1, X_2)$ obtenu par le calcul de point-fixe suivant :

$$\mathcal{R}_o(X_1, X_2) = K_g(X_1, X_2) \oplus (\mathcal{O}(X_1) - \mathcal{O}(X_2))$$

$$\mathcal{R}_{i+1}(X_1, X_2) = \begin{cases} \mathcal{R}_i(X_1, X_2) \\ \\ \oplus \forall Y_o \forall X'_1 [\{R(X_1, X'_1) \oplus (Y_o - \mathcal{O}(X'_1))\} \Rightarrow \\ \qquad \exists X'_2 \{R(X_2, X'_2) \oplus (Y_o - \mathcal{O}(X'_2)) \\ \qquad\quad \oplus \mathcal{R}_i(X'_1, X'_2) \oplus K_g(X'_1, X'_2)\}] \\ \\ \oplus \forall Y_o \forall X'_2 [\{R(X_2, X'_2) \oplus (Y_o - \mathcal{O}(X'_2))\} \Rightarrow \\ \qquad \exists X'_1 \{R(X_1, X'_1) \oplus (Y_o - \mathcal{O}(X'_1)) \\ \qquad\quad \oplus K_g(X'_1, X'_2) \oplus \ \mathcal{R}_i(X'_1, X'_2)\}] \end{cases} \quad (14)$$

Le calcul précédent termine et l'on peut montrer que

Proposition 3 $x_1 \sim x_2 \Leftrightarrow \mathcal{R}_\sim(x_1, x_2) = 0.$ $\diamond$

Nous renvoyons à [14] (Proposition 2 et Théorème 3) pour une preuve (équivalente) de cette proposition.

Calcul du modèle quotient. Soit S l'AMS modélisant le système et $\sim$ la relation d'équivalence sur les états du système calculée selon le point-fixe (14), et symboliquement représentée par $\mathcal{R}_\sim$. L'idée de la réduction est de considérer que tous les états équivalents peuvent être représentés par un seul et unique état (i.e. sa classe d'équivalence). Cette réduction s'opère en calculant le modèle quotient de S en fonction de $\sim$.
La première étape consiste à calculer les $\mathcal{R}_\sim$-classes du modèle quotient. À cet effet, nous introduisons "l" variables "fraîches" $Z = \{Z_1, ..., Z_l\}$, avec $l \leq n$ qui vont servir à coder le nouvel ensemble d'états (réduit). Dans un premier temps, nous introduisons un critère de fusion d'états vu comme une relation $\Phi \in \mathbb{Z}/_{p\mathbb{Z}}[X, Z]$ associée à $\sim$, telle que $\forall x \in (\mathbb{Z}/_{p\mathbb{Z}})^n$, $\Phi(x, Z) = 0$ a une solution unique et telle que $x_1 \sim x_2$ si et seulement si $\Phi(x_1, Z) = \Phi(x_2, Z)$. En d'autres termes, pour un état x donné la solution de $\Phi(x, Z)$ représente $[x]_{\mathcal{R}_\sim}$, i.e. la $\mathcal{R}_\sim$-classe de x.
Nous ne décrivons pas ici la manière dont Φ peut être symboliquement dérivée de $\mathcal{R}_\sim$. L'algorithme est basé sur une implémentation des polynômes sous forme de p-DD (nous renvoyons a [13] pour plus de détails). La figure 3 en donne une intuition.
Finalement, en supposant que Φ a déjà été calculée, la relation de transitions symbolique du diagnostiqueur peut être déterminée de la manière suivante

$$\mathcal{T}_d(Z, Y_o, Z') = \exists X \exists X' (\Phi(X, Z) \oplus \Phi(X', Z') \oplus (Y_o - \mathcal{O}(X')) \oplus \mathcal{R}(X, X')) \quad (15)$$

À cet AMS, nous associons une fonction de sortie délivrant le mode de pannes dans lequel le système se trouve.

$$\overrightarrow{F}(P(Z)) = [\exists X (\Phi(X, Z) \oplus F_i(X) \oplus P(Z))]_{i=[1..m]} \quad (16)$$

Par rapport à la première approche, on peut voir que la relation de transition est maintenant calculée hors-ligne. Outre

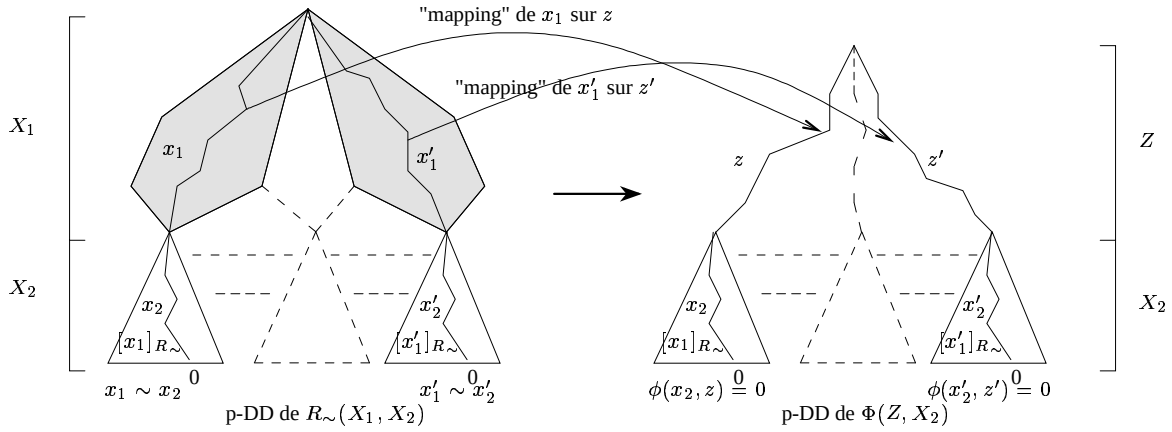


FIG. 3 – *Idée intuitive du calcul de* Φ

la réduction du modèle, le nombre d'états du diagnostiqueur a été potentiellement réduit (nous avons "confondu" des états ayant le même comportement). Toutefois, tous les problèmes n'ont pas disparu. En effet, le diagnostiqueur est par essence indéterministe (quand le diagnostiqueur reçoit une observation y_o, il peut y avoir plusieurs z pouvant être atteints). Ainsi des calculs en-ligne sont encore nécessaires (pour calculer par exemple l'estimation d'états). Par contre ces calculs sont beaucoup moins coûteux (parce que $\mathcal{T}_d$ a déjà été calculée). Ainsi l'estimation de l'état du diagnostiqueur est donnée par

$$\left\{ \begin{array}{ll} \widehat{Z}_0(Z) & = 0 \\ \widehat{Z}_1(Z) & = \exists X(\Phi(X,Z) \oplus (y_{o_1} - \mathcal{O}(X))). \\ \widehat{Z}_{i+1}(Z) & = \Delta(\exists Z(\widehat{Z}_i(Z) \oplus \mathcal{T}_d(Z, y_{o_{i+1}}, Z'))) \end{array} \right. \quad (17)$$

où Δ est une simple fonction de renommage d'un polynôme $P(Z')$ en le polynôme $P(Z)$.
L'estimation des pannes est donnée par la fonction $\overrightarrow{F}(\widehat{Z}_{i+1}(Z))$. Tout comme dans la première version du diagnostiqueur, un mode de pannes (ou un ensemble de modes de pannes) sera sûrement diagnostiqué lorsqu'un seul des composants du vecteur $\overrightarrow{F}(\widehat{Z}_{i+1}(Z))$ sera égal à zéro.

Théorème 1 *Le diagnostiqueur (12) et celui défini par (17) sont équivalents (dans le sens où, pour une séquence d'observations donnée, ils produiront la même estimation de pannes).*

Preuve : La preuve est due à la proposition 3, la correction de l'algorithme de réduction (la réduction permet de conserver pour une séquence d'observations donnée la même estimation des pannes et un futur commun) et au Théorème de [8] montrant le résultat dans un cadre explicite. ◇

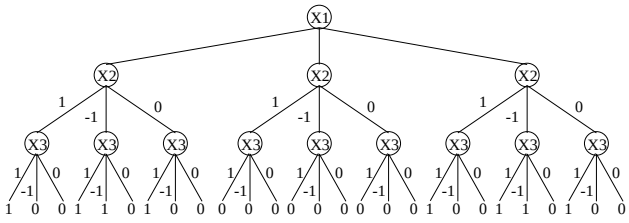
4 Conclusion

Basé sur [8], la technique de supervision présentée dans cet article s'appuie sur une représentation symbolique du modèle du système à superviser et de l'outil utilisé pour effectuer le diagnostic (le diagnostiqueur). La représentation polynomiale utilisée permet de représenter des systèmes de grandes tailles. La partie combinatoire la plus importante de ce travail se situe au niveau de l'algorithme de construction du diagnostiqueur et a été diminuée par l'introduction d'une réduction symbolique du modèle (modèle quotient par rapport à une relation d'équivalence s'appuyant sur l'histoire du système). L'intérêt de cette réduction symbolique est grande puisqu'elle permet d'effectuer les calculs de la fonction de transition hors ligne lors d'un prétaitement, réduisant ainsi considérablement le travail devant être effectué en ligne.

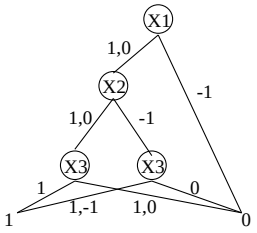
[16] et [8] ne présentent pas uniquement la construction de base du diagnostiqueur mais aussi une notion de diagnosticabilité. Ainsi un système est dit diagnosticable si toute panne peut être détectée après un nombre fini fixe n d'événements observés après l'occurrence de la panne. Des propriétés ont été présentées, permettant de vérifier, hors ligne, la diagnosticabilité ou non d'un système. Nous travaillons actuellement sur ces propriétés dans le cadre de notre approche symbolique. Le problème est donc dans un premier temps de définir une notion de diagnosticabilité (qui peut être la même que celle présentée dans [16] ou non) et, dans un deuxième temps, d'établir des propriétés permettant de vérifier de façon hors ligne si un système est diagnosticable ou non. Bien entendu ces propriétés doivent être vérifiées sur les modèles symboliques.

Une perspective est de lever l'hypothèse des pannes permanentes et de pouvoir aussi traiter les pannes transitoires. Une telle perspective est assez ambitieuse car elle ne permet plus de gérer des ensembles d'états mais nécessite de gérer des ensembles de trajectoires. Un tel travail est indispensable pour pouvoir utiliser une approche symbolique pour des systèmes tels que les réseaux de télécommunications. Enfin le traitement des pannes transitoires permettrait aussi de regarder comment utiliser les techniques de synthèse de contrôleurs dans le cadre du diagnostic. Lorsqu'une panne est détectée et localisée par le diagnostiqueur, le problème est d'activer un contrôleur qui va forcer le système à revenir dans un état normal. Le modèle sur lequel

le contrôleur doit agir est le diagnostiqueur lui-même. Il faudrait alors aussi étudier sous quelles conditions un tel système est contrôlable.



(a) Un example de graphe ternaire



(b) Le TDD réduit

FIG. 4 – *représentation de* $P(X_1, X_2, X_3)$

A Codage des polynômes ($p = 3$)

Les graphes de décisions ternaires (TDD pour Ternary Decision Diagrams) [4], un extension des graphes de décisions binaires (BDDs) [2], sont utilisés pour implémenter les polynômes avec toutes les opérations usuelles. Dans l'anneau quotient $A[X] = \mathbb{Z}/_{3\mathbb{Z}}[X]/_{<X^3-X>}$, pour chaque variable X_i,nous définissons trois polynômes

$$e_i^1 = -X_i^2 - X_i,\ e_i^2 = -X_i^2 + X_i,\ e_i^3 = 1 - X_i^2 \quad (18)$$

Ces polynômes ont les propriétés suivantes : $(e_i^\alpha)^2 = e_i^\alpha$ pour $\alpha = 1, 2, 3$, $e_i^\alpha e_i^\beta = 0$ pour tout $\alpha \neq \beta$, et $e_i^1 + e_i^2 + e_i^3 = 1$ (Dans $\mathbb{Z}/_{p\mathbb{Z}}$, cette famille de polynômes correspond aux polynômes de Lagrange).

Proposition 4 *Chaque $P(X) \in A[X]$ peut être décomposé d'une unique manière t.q. $P(X) = e_1^1 P_1 + e_2^1 P_2 + e_1^3 P_3$, où les polynômes $P_1,\ P_2,\ P_3$ ont la forme suivante : $P_1 = P(1, X_2, \ldots, X_n),\ P_2 = P(-1, X_1, \ldots, X_n)$, et $P_3 = P(0, X_1, \ldots, X_n,)$.* ⋄

Nous pouvons alors décomposer chaque polynôme en utilisant (18) pour chaque variable du polynôme : $e_1^{\alpha_1} \ldots e_n^{\alpha_n}$. Ainsi, étant donné un polynôme P de $A[X]$, et un ordre sur les variables $X_1 \prec X_2 \prec \ldots \prec X_n$, une h-expression de P est soit $P(X) = c_1 e_i^1 + c_2 e_i^2 + c_3 e_i^3$ où $c_i \in \mathbb{Z}/_{3\mathbb{Z}}$, soit $P(X) = e_i^1 P_1 + e_i^2 P_2 + e_i^3 P_3$ où les P_i sont des h-expressions avec des variables plus grandes que X_i.

Une h-expression peut également se voir comme un arbre ternaire. Ainsi, le polynôme suivant

$P(X_1, X_2, X_3) = X_1^2 X_2^2 X_3^2 - X_1^2 X_2^2 X_3 - X_1^2 X_2 X_3^2 + X_1^2 X_2 X_3 - X_1^2 X_3^2 - X_1^2 X_3 - X_1 X_2^2 X_3^2 + X_1 X_2^2 X_3 + X_1 X_2 X_3^2 - X_1 X_2 X_3 + X_1 X_3^2 + X_1 X_3 + X_2^2 X_3^2 - X_2^2 X_3 - X_2 X_3^2 + X_2 X_3 - X_3^2 - X_3.$

est représenté en Figure 4(a).

Deux idées mènent à une implémentation efficace des polynômes : la première est de réduire les h-expressions de la forme $P(X) = e_i^1 P_1 + e_i^2 P_2 + e_i^3 P_3$ en éliminant les idempotents e_i^α quand $P_1 = P_2 = P_3$ et en remplaçant la précédente expression par la valeur commune P_1. On peut de plus remarquer que plusieurs sous-graphes peuvent être identiques. L'idée est alors de ne les représenter qu'une seule fois. Nous obtenons ainsi une généralisation des graphes de Bryant [2]. Le graphe réduit du précédent exemple est donné en Figure 4(b).

Références

[1] P. Baroni, G. Lamperti, P. Pogliano, and M. Zanella. Diagnosis of active systems. In *Proceedings of the European Conference on Artificial Intelligence (ECAI)*, pages 274–278, 1998.

[2] R.E. Bryant. Graph-based algorithms for boolean function manipulations. *IEEE Transaction on Computers*, C-45(8):677–691, Août 1986.

[3] R. Debouk, S. Lafortune, and D. Teneketzis. Coordinated decentralized protocols for failure diagnosis of discrete-event systems. *Discrete Event Dynamic System : Theory and Applications*, 10(1/2):33–86, Janvier 2000.

[4] B. Dutertre. *Spécification et preuve de systèmes dynamiques*. Thèse, Université de Rennes I, IFSIC, Décembre 1992.

[5] E. Fabre, A. Benveniste, C. Jard, L. Ricker, and M. Smith. Distributed state reconstruction for discrete event systems. In *IEEE Control and Decision Conference (CDC)*, Sydney, Décembre 2000.

[6] J. Gunnarsson. *Symbolic Methods and Tools for Discrete Event Dynamic Systems*. PhD thesis, Linköping University, 1997.

[7] N. Halbwachs, P. Caspi, P. Raymond, and D. Pilaud. The synchronous data flow programming language LUSTRE. *Proceedings of the IEEE*, 79(9):1305–1320, Septembre 1991.

[8] S. Hashtrudi Zad. *Fault Diagnosis in discrete-event and hybrid systems*. PhD thesis, University of Toronto, Canada, September 1999.

[9] P. Le Guernic, T. Gautier, M. Le Borgne, and C. Le Maire. Programming real-time applications with signal. *Proceedings of the IEEE*, 79(9):1321–1336, Septembre 1991.

[10] H. Marchand, P. Bournai, M. Le Borgne, and P. Le Guernic. Synthesis of discrete-event controllers based on the signal environment. *Discrete Event Dynamic System : Theory and Applications*, 10(4):347–368, Octobre 2000.

[11] H. Marchand and M. Le Borgne. The supervisory control problem of discrete event systems using polynomial methods. Rapport de recherche 1271, Irisa, Octobre 1999.

[12] Y. Pencolé. Decentralized diagnoser approach: application to telecommunication networks. In *Proc. of 11th International Workshop on Principles of Diagnosis DX'00*, pages 185–192, 2000.

[13] S. Pinchinat and H. Marchand. Symbolic abstractions of automata. In *Proc of 5th Workshop on Discrete Event Systems, WODES 2000*, pages 39–48, Ghent, Belgium, Août 2000.

[14] S. Pinchinat, H. Marchand, and M. Le Borgne. Symbolic abstractions of automata and their application to the supervisory control problem. Rapport de recherche 1279, IRISA, Novembre 1999.

[15] L. Rozé and L. Cordier. Diagnosing discrete-event systems : an experiment in telecommunication networks. In *4th International Workshop on Discrete Event Systems*, pages 130–137, Cagliari, 1998.

[16] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. Teneketzis. Failure diagnosis using discrete event models. *Proceedings of the IEEE Transactions on Automatic Control*, 4(2):105–124, 1996.