AIDE À LA CONCEPTION DISTRIBUÉE D'UN SYSTÈME DIAGNOSTICABLE

Pauline RIBOT*

Directeur(s) de thèse : Yannick Pencolé et Michel Combacau

Laboratoire d'accueil :
LAAS-CNRS
7, Avenue du Colonel Roche
31077 Toulouse Cedex 4

Établissement d'inscription :
Université Paul Sabatier
118, Route de Narbonne
31062 Toulouse Cedex 4

Résumé

Cet article s'intéresse au problème de diagnosticabilité de faute dans des systèmes à événements discrets (SED) dans un cadre distribué. Des travaux antérieurs présentent des méthodes permettant de vérifier si une faute est diagnosticable ou pas. De nos jours, la simple vérification de cette propriété n'est plus suffisante dûe à la complexité grandissante des nouveaux systèmes. Il devient nécessaire de déterminer les causes de la non-diagnosticabilité d'une faute et de proposer des solutions à la conception qui permettent de les éliminer. Nous introduisons le problème de la caractérisation automatique des retours sur conception du système distribué afin d'en améliorer son degré de diagnosticabilité. Cette caractérisation s'appuie sur un problème d'optimisation de coût.

Mots-clés

Diagnosticabilité, Recommandations de conception, Systèmes à événements discrets, Systèmes distribués.

1 INTRODUCTION

Nous nous intéressons au problème de diagnostic de faute dans un système à événements discrets distribué. Il s'agit de déterminer les occurrences d'événements de faute à partir d'observations et de connaissance du système. Ce problème est étudié depuis plusieurs années, [9], [2], [6]. Dans ces travaux, l'objectif est de modéliser le système et d'appliquer sur ce modèle des algorithmes de surveillance mais ces algorithmes ont tous la même faiblesse : aucun d'eux ne tient compte du problème de diagnosticabilité du système. Si une analyse de diagnosticabilité est réalisée sur le système, l'algorithme est alors plus efficace et moins coûteux car de nouvelles informations sont prises en compte avant de l'implémenter. Dans cet article, nous adoptons le point de vue suivant. Due à la compléxité grandissante des nouveaux systèmes et des nouveaux pré-requis tels que la maintenance, la fiabilité ou la sécurité, l'étude de diagnosticabilité doit être réalisée dès la conception du système afin de garantir leurs performances. La principale difficulté dans la conception d'un système distribué provient du fait qu'un tel système est généralement réalisé par plusieurs concepteurs qui ne s'occupent que d'une seule partie du système. L'intégration des différentes parties du système est alors très complexe et a un impact direct sur la difficulté à garantir l'objectif de diagnosticabilité du système entier.

Notre objectif principal est de déterminer les caractéristiques et les modifications qui peuvent être utiles pour les concepteurs afin d'améliorer et de garantir des objectifs de diagnosticabilité. Cette analyse repose sur la conception du système distribué (conception et intégration des composants) mais

*pribot@laas.fr

aussi sur la conception de l'architecture de surveillance qui s'occupe de diagnostiquer le système. Nous proposons de formaliser ce problème comme un problème d'optimisation de coût dans un cadre distribué. Nous montrons ensuite la relation qu'il existe entre optimiser le coût de conception du système dsitribué et optimiser le choix d'une architecture de surveillance appropriée.

Cet article est organisé de la manière suivante. La section 2 rappelle des notions sur le diagnostic de faute et la diagnosticabilité. La section 3 définit le problème d'optimisation de coût dans le but de garantir la diagnosticabilité d'un système à événements discrets distribué. La section 4 introduit une méthodologie qui détermine, pour un système donné spécifié a priori, un ensemble de pré-requis de conception pour la diagnosticabilité qui tient compte des caractéristiques sous-jacentes à un système distribué et qui minimise le coût d'implémentation.

2 NOTIONS

2.1 DÉFINITION DU FORMALISME DES SED

Notre étude se place dans le cadre des systèmes à événements discrets (SED) pour le diagnostic à base de modèles comme défini dans [9]. Ce cadre a été développé depuis plusieurs années et il est utilisé pour différents types d'application. De plus, un grand nombre de propriétés a déjà été démontré dans ce cadre.

Un système distribué Γ est un ensemble de n composants $\Gamma = \{\Gamma_1, \Gamma_2, \ldots, \Gamma_n\}$ qui interagissent et qui évoluent avec les occurrences d'événements. Ce système peut être modélisé par un ensemble d'automates dans lequel chaque automate représente le modèle d'un composant (i.e. un modèle local). Dans la suite, nous noterons indifféremment le système et son modèle Γ.

Définition 1 (Modèle local) *Un modèle local Γ_i est un automate $\Gamma_i = (Q_i, \Sigma_i, T_i, q_{0_i})$ où :*
- *Q_i est un ensemble fini d'états ;*
- *Σ_i est l'ensemble des événements ayant lieu sur Γ_i ;*
- *$T_i \subseteq Q_i \times \Sigma_i \times Q_i$ est l'ensemble des transitions ;*
- *q_{0_i} est l'état initial.*

Différents événements peuvent apparaître sur un composant. L'ensemble Σ_i est divisé en deux ensembles disjoints : Σ_{l_i}, l'ensemble des événements localisés sur un composant Γ_i et Σ_{c_i}, l'ensemble des événements interactifs qui permettent la communication entre les différents composants. Les événements locaux de Σ_{l_i} peuvent être observables ou non observables : $\Sigma_{l_i} \subseteq \Sigma_{o_i} \cup \Sigma_{uo_i}$, où Σ_{o_i} (resp. Σ_{uo_i}) est l'ensemble d'événements observables (resp. non observables). Σ_{f_i} représente l'ensemble des événements de faute intervenant sur le composant Γ_i qui doivent être diagnostiqués. L'algorithme de diagnostic repose sur la localisation des événements de faute intervenant sur les composants, nous supposons donc que $\Sigma_{f_i} \subseteq \Sigma_{l_i}$. À la différence du cadre défini dans [9], certains événements interactifs peuvent être observables : $\Sigma_{c_i} \subseteq \Sigma_{o_i} \cup \Sigma_{uo_i}$.

Un sous-système γ est un ensemble non vide de m composants du système, avec $m \leq n$. Le comportement du sous-système γ peut être explicitement modélisé par un automate $\|\gamma\|$ obtenu à partir de l'opération de synchronisation classique, notée $\|$, qui est le produit d'automates synchronisés sur les événements interactifs de γ.

2.2 DÉFINITION D'UN DIAGNOSTIQUEUR POUR UN SOUS-SYSTÈME

Le problème de diagnostic est identique à celui spécifié dans [7]. Un diagnostiqueur a pour but de diagnostiquer un type de faute. Soit F un événement de faute apparaîssant sur un composant Γ_i qui appartient à un sous-système γ, le diagnostiqueur d'un sous-système γ qui doit diagnostiquer F est noté Δ_γ. Le diagnostiqueur Δ_γ est une fonction qui, après l'observation d'une séquence d'observations σ émise par γ, fournit en temps réel une information de diagnostic $\Delta_\gamma(F, \sigma)$ qui correspond à l'un des trois types suivants.

- $\Delta_\gamma(F,\sigma) = F$-sûr : la faute F est apparue dans tous les comportements du sous-système γ qui sont cohérents avec la séquence d'observations σ.
- $\Delta_\gamma(F,\sigma) = F$-sain : aucun des comportements de γ cohérents avec la séquence d'observations σ ne contient la faute F.
- $\Delta_\gamma(F,\sigma) = F$-ambigu : certains comportements de γ qui sont cohérents avec la séquence d'observations σ contiennent la faute F, et d'autres pas.

Généralement le diagnostiqueur est défini à partir du modèle entier Γ comme dans [9]. Le résultat global de diagnostic pour une séquence d'observations σ est alors donné par $\{\Delta_\Gamma(F,\sigma),\ F \in \Sigma_f\}$.

2.3 DÉFINITION DE LA DIAGNOSTICABITLITÉ SUR UN SOUS-SYSTÈME

La diagnosticabilité est une propriété qui mesure la capacité du système de surveillance à diagnostiquer des fautes apparaîssant sur le système surveillé. Nous reformulons la définition de diagnosticabilité de [5].

Définition 2 (Diagnosticabilité locale) *Un événement de faute F est localement diagnosticable dans un sous-système γ si chacune de ses occurrences sur un composant de γ est toujours suivie par une séquence finie d'observations telle que le diagnostic de Δ_γ est F-sûr.*

Plusieurs outils permettent de vérifier la diagnosticabilité comme le diagnostiqueur global dans [9]. Notre définition de la diagnosticabilité est similaire à celle introduite par [9] mais elle s'applique à tout sous-système γ. La propriété suivante présente une relation entre la diagnosticabilité locale et globale. La diagnosticabilité globale correspond à la diagnosticabilité locale sur le système entier Γ.

Propriété 1 *Sous l'hypothèse d'observabilité équitable [1], si une faute F est localement diagnosticable sur un sous-système γ alors F est diagnosticable dans le système entier Γ.*

D'après cette propriété, il n'est donc pas nécessaire d'observer le système entier pour diagnostiquer une faute F intervenant sur un composant de Γ. Il peut être suffisant d'observer seulement un sous-système γ.

3 CONCEPTION POUR LA DIAGNOSTICABILITÉ

L'objectif est d'établir un retour à la conception sous la forme de pré-requis afin de garantir la diagnosticabilité des différentes parties du système distribué (l'analyse de diagnosticabilité est réalisée dès la conception). Ce problème est étroitement lié au problème d'assistance à la conception décrit dans [5].

Afin de garantir la diagnosticabilité du système, plusieurs modifications sur le modèle des sous-systèmes peuvent être considérées selon l'architecture de surveillance adoptée. Le premier type d'opération consiste à améliorer l'observabilité d'un sous-système γ en sélectionnant des types de capteurs (des capteurs d'événements locaux ou de communication, des capteurs intelligents pour l'acquisition active d'information comme dans [10]) et en optimisant leur positionnement sur les composants du sous-système et leur nombre. Les capteurs ajoutés sont supposés fiables et non bruités. Ce problème est similaire au problème de sélection de capteurs comme dans [1] ou [4]. Le second type d'opération modifie la structure du sous-système qui se traduit par une réorganisation des transitions du modèle de γ (ajout ou suppression d'une transition). Les différentes opérations possibles et leur signification physique sont énumérées dans [8]. Un coût est associé à chaque modification sur le système. Certaines modifications sur le système ne sont pas réalisables. Un coût infini est associé à de telles modifications, cela correspond par exemple à l'observation des événements de faute. Le but est de fournir des pré-requis aux concepteurs pour rendre le système diagnosticable en minimisant le coût

[1] L'observabilité équitable signifie que chaque composant émet toujours des observations après un temps fini (pas de famine d'observations).

des opérations sur le système, noté C_D.

Pour avoir un système diagnosticable, l'architecture de surveillance a besoin de récupérer les informations émises par les différents composants du système. L'accès à ces ressources d'informations induit un coût C_M pour le système de surveillance (coût pour récupérer des observations, coût algorithmique) lié au choix du type d'architecture de diagnostic (centralisée [9], distribuée, décentralisée [2] [6]). Le challenge est donc de déterminer un compromis entre les deux coûts C_D et C_M :

$$C_G = min \sum_{i=1}^{p} (C_{D_i} + C_{M_i}), \tag{1}$$

où p est le nombre de fautes qui peuvent apparaître dans le système et pour lesquelles la diagnosticabilité doit être garantie.

4 MÉTHODOLOGIE

Cette section introduit les propriétés de précision d'un diagnostic et de monotonie de la diagnosticabilité. Nous montrons comment ces propriétés peuvent aider à établir une méthodologie pour fournir des pré-requis et des coûts minimaux aux concepteurs en indiquant la partie du système qui doit être respécifiée et surveillée. Dans cette section, nous considérons qu'il existe déjà une spécification du système et que la seule modification possible est l'ajout d'événements observables.

4.1 PRÉCISION

Le diagnostic d'un sous-système γ est dit *précis* si son observation est suffisante pour fournir un diagnostic qui est globalement cohérent. Le système de surveillance n'a alors pas besoin d'information provenant des autres composants et nous avons la certitude d'avoir à tout instant un diagnostic local cohérent (égal) au diagnostic global. Soit Δ_γ le diagnostiqueur du sous-système γ et $\sigma_\gamma = P_{\Sigma_{o_\gamma}}(\sigma)$, la projection de la séquence $\sigma \in \Sigma_o^*$ sur les événements observables de γ. L'opération de projection peut être récursivement définie comme suit. Notons ϵ, la séquence vide dans Σ^*.

Définition 3 (Projection) *L'opération de projection $P_{\Sigma'} : \Sigma^* \rightarrow \Sigma'^*$ est telle que $P_{\Sigma'}(\epsilon) = \epsilon$ et pour tout $uv \in \Sigma^*$, $u \in \Sigma$,*

$$P_{\Sigma'}(uv) = \begin{cases} uP_{\Sigma'}(v) & \textit{si } u \in \Sigma' \\ P_{\Sigma'}(v) & \textit{sinon.} \end{cases}$$

Définition 4 (Précision) *Le diagnostic d'un sous-système γ est précis pour un événement de faute $F \in \Sigma_{f_\gamma}$ si et seulement si il existe le diagnostiqueur Δ_γ tel que*

$$\forall \sigma \in \Sigma_o^*,\ \Delta_\Gamma(F, \sigma) = \Delta_\gamma(F, \sigma_\gamma). \tag{2}$$

Indépendamment de la diagnosticabilité, il est très intéressant de trouver un sous-système dont le diagnostic est précis car c'est un moyen de borner le coût C_M, la surveillance pouvant être limitée à ce sous-système. Nous pouvons toujours trouver un sous-système dont le diagnostic est précis (le diagnostic du système global Γ étant toujours précis).

4.2 MONOTONIE DES PROPRIÉTÉS

La méthodologie repose sur la recherche d'un ensemble de modifications pour garantir deux propriétés : diagnosticabilité et précision. Il est donc important de savoir si chacune des deux propriétés peut être conservée après les modifications pour garantir l'autre.

Définition 5 (Monotonie) *Soit $Prop$ une application booléenne ($\forall X,\ Prop\ :\ P(X) \mapsto \{0,1\}$, où $P(X)$ est l'ensemble des parties de X), $Prop$ est monotone ssi*

$$\forall X,\ Y,\ X \subseteq Y \Rightarrow Prop(X) = Prop(Y). \tag{3}$$

Cette définition montre que pour deux ensembles X et Y, si X est inclus dans Y, toute propriété vérifiée par X est aussi vérifiée par Y. Dans notre cas X et Y sont deux ensembles d'événements observables ($X \subseteq \Sigma_o$, $Y \subseteq \Sigma_o$).

Propriété 2 *La diagnosticabilité est une propriété monotone*[2]*. La précision n'est pas une propriété monotone.*

La propriété de diagnosticabilité est toujours conservée en considérant de nouvelles observations alors que la propriété de précision peut être perdue par cet ajout d'information.

4.3 ALGORITHME

Cette section présente un algorithme qui sélectionne un sous-système et retourne des pré-requis Req sous la forme d'un ensemble de modifications à réaliser sur ce sous-système afin de le rendre diagnosticable à l'aide d'un diagnostiqueur précis avec un coût total C_G minimal (en considérant le coût de la surveillance C_M, le coût C_A pour la précision et le coût C_D pour la diagnosticabilité de la faute F intervenant dans le composant Γ_i).

Input : $F \in \Sigma_{f_i}, \Gamma = \{\Gamma_1, \ldots, \Gamma_n\}$
$C_G^0 \leftarrow \infty$; $k \leftarrow 1$; $Req = \emptyset$
repeat
  $C_G^k \leftarrow \infty$
  for all $\gamma \in subsystem(\Gamma_i,\ k)$ **do**
    $C_M \leftarrow Monitoring(\gamma)$; $C_D = 0$; $C_A = 0$;
    if $CheckDiagnosability(\gamma, F)$ **then**
      if $\neg CheckAccuracy(\gamma)$ **then**
        $(Req, C_A) \leftarrow MakeAccurate(\gamma)$
      end if
    else
      $(Req, C_A, C_D) \leftarrow MakeDiagnosable(\gamma, F) \wedge MakeAccurate(\gamma)$
    end if
    $C_G^\gamma \leftarrow C_M + C_A + C_D$
    $C_G^k \leftarrow \min(C_G^k, C_G^\gamma)$
  end for
  $k \leftarrow k + 1$
until $(C_G^{k-1} \geq C_G^{k-2}) \wedge (k \geq 2) \wedge (k \leq n)$
$C_G \leftarrow C_G^{k-2}$
Output : γ ; C_G ; Req

Comme la propriété de diagnosticabilité est monotone, nous préférons rendre le sous-système diagnosticable avant de considérer la précision. Si le sous-système est diagnosticable, la propriété de diagnosticabilité sera conservée par l'ajout d'événements observables pour rendre le diagnostic précis. Si le sous-système n'est pas diagnosticable, l'optimisation se fera directement sur le couple de coûts $C_D,\ C_A$. Dans l'algorithme proposé, l'expression $subsystem(\Gamma_i,\ k)$ représente l'ensemble des sous-systèmes composés de k composants qui contiennent Γ_i. La fonction $Monitoring$ induit un coût C_M pour la surveillance du sous-système qui dépend de l'implémentation de l'architecture

[2]Les preuves des propriétés de monotonie ont été omises par défaut de place dans l'article.

de diagnostic. Les fonctions $CheckingAccuracy$ et $CheckingDiagnosability$ sont booléennes. La première fonction applique un critère qui détermine si le diagnostic d'un sous-système est précis ou non [7] et la seconde fonction utilise un algorithme pour déterminer la diagnosticabilité d'un sous-système pour une faute F [5], [3]. Les fonctions $MakeDiagnosable$ et $MakeAccurate$ utilisent des techniques de placement de capteurs sur γ [1] [4] [11].

5 CONCLUSION

Cet article définit un cadre basé sur un formalisme classique de diagnostic à base de modèles afin d'étendre l'analyse de diagnosticabilité automatique et de fournir des pré-requis de conception pour un système dynamique distribué. Nous proposons de définir ce problème comme un problème d'optimisation de coût en tenant compte du coût de conception du système mais aussi des coûts liés à l'architecture de surveillance afin de minimiser les coûts d'intégration du système distribué. Nous présentons la propriété de précision qui permet d'isoler un sous-système sur lequel il est possible de concevoir une architecture de surveillance qui fournit un diagnostic aussi précis que possible. Dans cet article nous avons présenté un algorithme qui sélectionne un sous-système et indique les modifications de conception pour le rendre diagnosticable et précis avec un coût total C_G minimal.

Nos perpectives sont de développer en détails la méthodologie présentée en intégrant les différentes méthodes (testeurs de diagnosticabilité, sélecteurs de placement de capteurs) et de fournir automatiquement des pré-requis de conception pour des systèmes distribués.

Références

[1] R. DEBOUK, S. LAFORTUNE et D. TENEKETZIS. « On an Optimization Problem in Sensor Selection for Failure Diagnosis ». Dans *38th Conference on Decision and Control*, pages 4990–4995, Phoenix, Arizona, USA, December 1999.

[2] R. DEBOUK, S. LAFORTUNE et D. TENEKETZIS. « Coordinated Decentralized Protocols for Failure Diagnosis of Discrete Event Systems ». *JDEDS : Theory and Application*, 10(1–2) :33–86, 2002.

[3] S. JIANG, Z. HUANG, V. CHANDRA et R. KUMAR. « A Polynomial Time Algorithm for Diagnosability of Discrete Event Systems ». *IEEE Transactions on Automatic Control*, 46(8) :1318–1321, 2001.

[4] S. JIANG, R. KUMAR et H. E. GARCIA. « Optimal Sensor Selection for Discrete Event Systems Under Partial Observation ». *IEEE Transactions on Automatic Control*, 48 :369–381, March 2003.

[5] Y. PENCOLÉ. « Assistance for the Design of a Diagnosable Component-Based System ». Dans *17th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'05)*, pages 549–556, 14-16 Nov 2005.

[6] Y. PENCOLÉ et M.-O. CORDIER. « A formal framework for the decentralised diagnosis of large scale discrete event systems and its application to telecommunication networks ». *Artificial Intelligence*, 164 :121–170, May 2005.

[7] Y. PENCOLÉ, D. KAMENETSKY et A. SCHUMANN. « Towards low-cost diagnosis of component-based systems ». Dans *6th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Process*, Beijing, China, September 2006.

[8] P. RIBOT, Y. PENCOLÉ et M. COMBACAU. « Characterization of requirements and costs for the diagnosability of distributed discrete event systems ». Dans *5th Workshop on Advanced Control and Diagnosis (ACD'07)*, Grenoble, November 15-16 2007.

[9] M. SAMPATH, R. SENGUPTA, S. LAFORTUNE, K. SINNAMOHIDEEN et D. TENEKETZIS. « Diagnosability of Discrete Event System ». *IEEE Transactions on Automatic Control*, 40(9) :1555–1575, 1995.

[10] D. THORSLEY et D. TENEKETZIS. « Active Acquisition of Information for Diagnosis of Discrete-Event Systems ». Dans *42th Annual Allerton Conference on Communication, Control, and Computing*, University of Illinois, 2004.

[11] G. TORTA et P. TORASSO. « Computation of Minimal Sensor Sets from Precompiled Disciminability Relations ». Dans *18th International Worshop on Principles of Diagnosis (DX'07)*, pages 202–209, Nashville, TN, USA, May 2007.