Optimisation du diagnostic de systèmes probabilistes

1 Motivations

De nos jours, les systèmes cyber-physiques sont omniprésents. De par l'aspect critique de certaines taches qu'ils remplissent, une défaillance peut entrainer des pertes économiques et humaines importantes. Par conséquent, il est devenu important de vérifier précisément et automatiquement si un système réel suit correctement sa spécification.

Le domaine des "Méthodes formelles" vise à réaliser cette vérification en traduisant d'abord le système réel et sa spécification en un modèle mathématique et un ensemble de propriétés formelles. Le *diagnostic* [5, 2] est une des propriétés classiquement étudiées. Celle-ci vise à détecter rapidement des comportements fautifs dans le système. L'étude du diagnostic de modèles incluant des comportements imprévisibles, stochastiques (tels que les chaîns de Markov[3, 4, 1]), est cependant encore limité.

2 Sujet

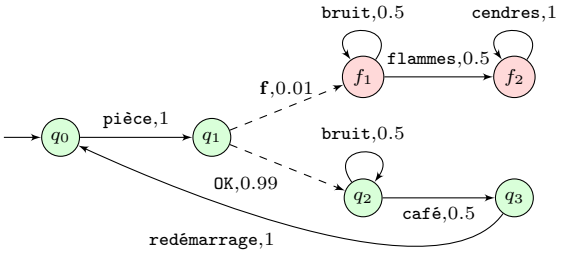
Dans un système du type automate, le passage d'un état à un autre se fait via une action. L'observation de ces systèmes suppose en général que certaines actions sont visibles pour un observateur, et d'autres sont inobservables. Si on distingue parmi ces actions inobservables une action particulière appelée "faute", le but du diagnostic est, en regardant le comportement du système (au travers des actions observables uniquement) de dire si l'exécution en cours contient une faute ou pas. Cette problématique est particulièrement cruciale dans les systèmes critiques (tels que la surveillance d'une centrale nucléaire) où seule une partie des actions est visible (par exemple via des données de capteurs) et où l'on veut utiliser cette information partielle pour savoir si une erreur s'est produite ou non, et éviter les conséquences qu'une telle faute non corrigée à temps pourrait engendrer.

Dans le cadre non probabiliste, un système est diagnosticable si, quelle que soit l'exécution observée, on peut au bout d'un moment conclure si cette exécution contient une faute ou non. Les travaux dans ce domaine se concentrent entre autres sur le fait de déterminer si un système est diagnosticable ou non, et de construire un diagnostiqueur le cas échéant.

Si la littérature est assez riche sur la diagnosticabilité de systèmes non probabilistes, il y reste un champ assez large de propriétés à étudier dans le cadre probabiliste, qui est le but de ce stage.

Dans ces systèmes, à chaque étape le choix entre plusieurs actions se fait aléatoirement, par exemple avec proba 1/2 je fais l'action a, 1/4 l'action b et 1/4 la faute (inobservable) **f**. On peut alors définir une mesure de probabilité sur l'ensemble des chemins. Dans ce cadre, le problème de diagnosticabilité devient : a-t-on dans notre système la diagnosticabilité (classique) sur "quasi" tous les chemins.

Cette distinction est utile car certains systèmes, comme l'exemple de machine à café ci-après, ne sont pas diagnosticables au sens classique. En effet, si la faute **f** a lieu, on peut ensuite boucler indéfiniment sur l'état suivant via l'action `bruit`. Tant que l'on n'observe que cette action on ne sait pas si on a pris l'action **f** ou l'action `OK` (inobservables toutes les deux). Si l'on introduit des probabilités dans le système, par exemple 1/2 sur la transition `bruit` et 1/2 sur l'autre transition, le chemin qui consiste à boucler sur `bruit` à l'infini a une probabilité 0. Or tous les autres chemins (qui ont donc soit une action `flamme` soit `café`) sont observables. La version probabiliste de ce système est donc diagnosticable.



Il existe différentes autres notions de diagnosticabilité sur les systèmes probabilistes, et on peut également s'intéresser à voir comment diagnostiquer un système à moindre coût. Autrement dit, si on définit un coût pour chaque type d'action (je dois payer 2 pour observer les a, 1 pour observer les b), quel est le coût minimum à payer pour rendre un système donné diagnosticable ?

Le sujet de stage est volontairement large pour pouvoir s'adapter à l'avancée du ou de la stagiaire et aux intérêts qu'il ou elle va développer durant l'avancée du stage.

3 Compétences attendues

Le ou la stagiaire devra avoir de bonnes connaissances sur les automates finis, des connaissances en vérification formelle et être intéressé.e pour travailler avec des systèmes probabilistes.

References

[1] The pomdp page. URL: `https://www.pomdp.org/`.

[2] N. Bertrand, S. Haddad, and E. Lefaucheux. A tale of two diagnoses in probabilistic systems. *Inf. Comput.*, 269, 2019.

[3] Wai-Ki Ching, Ximin Huang, Michael K. Ng, and Tak Kuen Siu. *Markov Chains: Models, Algorithms and Applications*. Springer Publishing Company, Incorporated, 2nd edition, 2013.

[4] Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., USA, 1st edition, 1994.

[5] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. Teneketzis. Diagnosability of discrete-event systems. *IEEE Transactions on Automatic Control*, 40(9):1555–1575, 1995.