

Réseaux de Petri Temporel à Priorité : Application à l'analyse d'ordonnançabilité des systèmes temps réel

Résumé :

L'appréhension de la complexité croissante des systèmes temps réel nécessite d'utiliser des techniques de conception basées sur un haut niveau d'abstraction. Un soin particulier doit être amené à la phase de haut niveau de modélisation. C'est à ce niveau en effet que les propriétés fonctionnelles et non fonctionnelles du système sont définies.

L'aspect critique attaché aux systèmes temps réel, qui résulte principalement des contraintes temporelles exigées, impose la nécessité d'appliquer des techniques d'analyse et de vérification à savoir les techniques formelles dédiées à l'analyse d'ordonnançabilité du système. Le fait de profiter suffisamment des ressources matérielles de la plateforme physique dans laquelle est implantée l'application ajoute encore une dimension au problème. Mes travaux de mastère s'inscrivent dans le cadre la vérification formelle de l'ordonnançabilité.

Après avoir fait le point sur les différentes méthodes classiques d'analyse d'ordonnançabilité, nous avons proposé comme première contribution une nouvelle démarche de vérification des aspects temporels, qui guide le concepteur à allouer les différentes tâches du système sur une architecture. En deuxième étape, nous avons suggéré une extension des Réseaux de Petri temporel. Cette solution introduit une nouvelle priorité sur les arcs.

Travaux réalisés :

La démarche suggérée requiert l'intégration d'un outil d'analyse des propriétés temporelles. La différence entre les outils existants réside au niveau de l'efficacité de détection de toutes les erreurs subtiles et la génération de tous les scénarios possibles. Nous proposons alors une extension des Réseaux de Petri qui vient répondre à ces deux besoins.

En effet, la représentation graphique des RdPs, appliquée à la vérification des systèmes temps réel, permet de supporter le parallélisme, la synchronisation, le partage des ressources qui sont des critères dont l'ordonnancement temps réels doit les tenir en compte. La représentation mathématique permet d'établir les équations d'états, à partir desquels il est possible d'apprécier les propriétés du modèle et de les comparer avec le comportement du système modélisé. Cependant, la représentation d'un problème d'analyse d'ordonnançabilité n'est pas triviale malgré les extensions courantes de formalisme. D'autre part, les systèmes temps réel

possèdent naturellement un caractère compositionnel et notamment en les analysant avec une ingénierie à base de modèles. La principale contribution du master porte sur l'extension des RdPs temporels. Les deux intentions majeures de cette extension sont d'apporter l'aspect déterminisme à travers l'intégration de la notion de priorité. La deuxième motivation est d'assurer une composition hiérarchique des objets temps réel afin de fournir une grande flexibilité de manipulation et d'échange d'informations avec les modèles UML/MARTE.

En effet, le formalisme proposé appelé PTPN (Priority Time Petri Nets) vise principalement à surmonter le problème de conflit de transitions tirables. Seule la transition qui a la plus grande priorité est franchie avec le nouveau formalisme. La sémantique du PTPN permet de supporter une stratégie d'ordonnancement à priorité fixe tel que le Rate Monotonic. Elle accélère la simulation des séquences de franchissement est par la suite expliquée. En fait, pour franchir un marquage, le PTPN offre une machine de franchissement appelée PTPN Firing Machine (PFM). L'entrée de la machine PFM est un marquage du réseau PTPN. Pour chaque entrée, la machine déclenche quatre événements (travaux) séquentiels. Au cours du premier travail et à partir d'un marquage de départ, PFM détermine la séquence des transitions franchissables F_{t_s} . F_{t_s} présente l'entrée du deuxième travail. Ce dernier consiste à extraire les transitions non valides de F_{t_s} pour que cette dernière ne présente que les transitions valides. Ces transitions sont franchissables et respectent leurs durées de tir.

L'analyse d'ordonnançabilité d'un système temps réel (STR) modélisé avec PTPN, produit un modèle dont la taille est très importante. La grande complexité des STR la rend difficile à comprendre et à gérer. Pour remédier aux problèmes d'hiérarchisation et d'interprétation des modèles décrits en PTPN, nous faisons recours aux méthodes de structuration des réseaux de Petri. En se basant sur la structure d'un STR, nous distinguons deux importantes classes d'objets. Une classe (Task) qui décrit les objets tâches de l'application temps réel. Une deuxième classe (Processor) présente les objets de type processeur considéré comme composante principale de l'architecture ou encore n'importe quelle ressource matérielle d'exécution. Le modèle considéré doté d'une sémantique bien définie, permet de définir le comportement dynamique de l'ensemble d'objets en combinant les comportements individuels décrits dans les classes et les protocoles qui permettent aux objets de collaborer. Quand il s'agit de l'analyse d'ordonnançabilité.

Rappelons que nous travaillons dans le cadre d'une méthodologie basée sur l'ingénierie dirigée par les modèles. Les modèles de départ sont à base du nouveau profil UML appelé MARTE. Toutefois, UML\Marthe ne propose pas d'un moyen de vérification des

incohérences entre les différents modèles. Nous ne pouvons pas, par exemple, vérifier si les comportements décrites par le diagramme états-transitions implémentent bien les interactions décrites par le diagramme de séquence. L'absence d'une sémantique bien déterminée rend la validation formelle du comportement indispensable. Il s'agit alors de traduire les modèles semi-formelles en composants PTPN. Nous détaillons les modèles sources qui nous serviront de base afin d'exprimer quels éléments seront traduits. Nous présentons également le processus d'intégration du PTPN dans un cadre IDM. Nous illustrons en conséquence la démarche de traduction des modèles. Nous finissons par l'illustration de quelques expérimentations.