

Avis de Soutenance

Monsieur Anasse CHAFIK

Informatique et applications

Soutiendra publiquement ses travaux de thèse intitulés

Defeasible Temporal Logic pour la spécification et la vérification des systèmes tolérants aux exceptions.

dirigés par Monsieur Jean François CONDOTTA et Monsieur Ivan VARZINCZAK

Soutenance prévue le **vendredi 09 septembre 2022** à 10h00

Lieu : UFR des Sciences Jean Perrin Rue Jean Souvraz SP 18 F-62307 Lens Cedex

Salle : des thèses

Composition du jury proposé

M. Jean-François CONDOTTA	Université d'Artois	Directeur de thèse
M. Balbiani PHILIPPE	Université Toulouse 3	Rapporteur
Mme Laura GIORDANO	Università del Piemonte Orientale	Rapporteuse
M. Ivan VARZINCZAK	Université d'Artois	Co-directeur de thèse
Mme Fahima CHEIKH-ALILI	Université d'Artois	Examinatrice
M. Yakoub SALHI	Université d'Artois	Examineur
M. Yannick CHEVALIER	Université Toulouse 3	Examineur
Mme Souhila KACI	Université de Montpellier	Examinatrice

Résumé :

Les logiques temporelles sont des outils formels de spécification et de vérification des systèmes informatiques. Le succès de ces logiques est principalement dû à leur syntaxe élégante et à leur sémantique intuitive qui simplifient la représentation des propriétés évoluant au cours du temps. Différents outils ont été développés à partir de ces formalismes, notamment des approches par automates ou des méthodes des tableaux. Cependant, ces logiques restent limitées quand il s'agit de la modélisation et du raisonnement à propos des certains aspects réels des systèmes informatiques. En effet, les systèmes informatiques ne sont pas garantis sûrs à 100%, et les propriétés que l'on souhaite vérifier peuvent avoir des exceptions triviales et tolérables, ou au contraire, des exceptions devant être gérées avec soin afin de garantir la fiabilité globale du système. De même, le comportement attendu d'un système peut ne pas être correct pour toutes les exécutions possibles, mais plutôt pour ses exécutions les plus normales où les plus plausibles. Le raisonnement non-monotone est un domaine de recherche qui tente de capturer des modes de raisonnement révisables représentant avec précision le raisonnement du sens commun au delà du raisonnement déductif de la logique classique. De plus, il permet de raisonner avec des exceptions et le dynamisme de l'information. L'objectif principal de ce travail est d'intégrer des approches non monotones aux logiques temporelles pour mieux représenter le comportement de systèmes tolérants aux exceptions. Le formalisme présenté dans ce mémoire, appelé logique temporelle linéaire révisable, combine la syntaxe et la sémantique de LTL classique avec l'approche préférentielle sur des inférences conditionnelles. Sa syntaxe contient une version révisable des opérateurs temporels permettant d'exprimer des spécifications similaires à leurs contreparties classiques, mais d'une façon plus souple afin de considérer des points de temps admettant des exceptions lors des exécutions du système. LTL révisable étend les interprétations temporelles avec une relation de préférence permettant de nuancer la prise en compte des points temporels. Nous avons travaillé sur la décidabilité du problème de la satisfiabilité des formules du langage. Dans ce cadre, nous avons fait l'étude sur deux sous-langage de LTL révisable qui sont L1 et L*. La classe de complexité du problème de satisfiabilité pour les formules de L1 est NP-complet. Concernant L*, nous avons pu démontrer la décidabilité du problème pour une classe d'interprétations nommée state-dependent interprétations. Nous avons montré la propriété des modèles bornés pour les deux sous-langages. Grâce à ces propriétés, nous avons défini des procédures pour vérifier si une interprétation devinée d'une façon non-déterministe satisfait la formule d'entrée. Dans un deuxième temps, nous avons défini une méthode des tableaux pour L1 en s'appuyant sur l'approche one-pass tree-shaped tableau récemment proposée pour LTL classique. L'originalité dans notre approche est de montrer comment gérer la sémantique préférentielle dans LTL révisable, ainsi que comment générer les interprétations lorsque les formules sont satisfiables. Nous avons introduit une méthode correcte et complète qui pourra servir comme base pour l'exploration d'autres méthodes des tableaux pour cette logique.