REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE MOHAMED BOUDIAF - M'SILA
FACULTE DES MATHEMATIQUES ET DE L'INFORMATIQUE
DEPARTEMENT D'INFORMATIQUE

MEMOIRE de fin d'étude

Présenté pour l'obtention du diplôme de MASTER

Domaine : Mathématiques et Informatique

Filière : Informatique

Spécialité : Réseaux

Par: Larbaoui Lotfi

SUJET

Outil d'aide à la localisation des erreurs dans les modèles PRISM

Soutenu publiquement le : 31/05/2016 devant le jury composé de :

	Université de M'sila	**Président**
Dr . Mustapha Bourahla	**Université de M'sila**	**Rapporteur**
	Université de M'sila	**Examinateur**
	Université de M'sila	**Examinateur**

Promotion : 2015 /20 16

Résumé

Dans ce mémoire, nous proposons un outil d'aide à la localisation des erreurs qui apparaissent lors de la vérification de systèmes probabilistes en utilisant la technique du *model checking probabiliste* avec l'outil PRISM. Le model checking probabiliste est une technique de vérification qui consiste à déterminer si un modèle probabiliste M vérifie une propriété donnée. Les modèles sont décrits par des systèmes de transitions tandis que la logique temporelle est utilisée comme langage de spécification des propriété .

Ce mémoire aborde pour la première fois une tâche du *vérification totalement automatique* des modèle PRISM , pour lequel un contre-exemple est disponible . Nous présentons les résultats des premières exprérimentations qui sont encourageants.

Mot-clefs :Vérification formelle , Le model checking probabiliste ,outil PRISM ,contre-exemple probabiliste , localisation d'erreurs .

Abstract

In this memory we offer tool to support errors location that occur during the verification of probabilistic systems using the technique of *model-checking* with PRISM tool. Probabilistic model-checking is a technique for verification that aims at determining whether a probabilistic model satisfies a given property.The models are described by transition systems while temporal logic is used as specification language for properties.

This memory addresses for the first time the task of *fully automatic verification* of PRISM model ,for which a counter-example is available . We present preliminary experimental results that are quite encouraging.

Keywords :Formel verification , Probabilistic model checking, PRISM model checker , probabilistic contre-example ,errors location

Table des matières

Conclusion générale

Les systèmes issus des technologies de l'information et de la communication font désormais partie intégrante de notre vie quotidienne. Internet n'est plus l'apanage de quelques spécialistes et des universitaires, il s'est répandu dans toutes les couches de la société.Des systéme du transport intélligent ,des systèmes intégrés comme les cartes de crédit, les téléphones portables et les téléviseurs intelligents sont autant de preuves de cette présence dans notre réalité quotidienne. Des techniques et des outils ont donc été conçus pour permettre aux concepteurs de faire la vérification de façon automatique. Dans ce mémoire, nous nous sommes intéressés à une technique de vérification formelle basée sur le modèle appelée évaluation de modèle (model-checking en anglais). L'évaluation de modèle probabiliste est une technique de vérification qui consiste à déterminer si un modèle probabiliste M vérifie une propriété donnée. Les modèles sont décrits par des systèmes de transitions appelés modèles tandis que la logique temporelle est utilisée comme langage de spécification des propriétés. L'évaluation de modèle exige aussi un algorithme d'évaluation de modèle qui se charge de déterminer si le système à l'étude vérifie ou non la propriété énoncée. Les algorithmes sont implémentés sous forme d'outil informatique pour l'évaluation de modèle appelés vérificateurs (model-checkers en anglais).

Il existe différentes façons de modéliser un système et différents types de modèles existent. Dans le cas du model-checking probabiliste, nous utilisons des modèles qui intègrent l'information sur la probabilité qu'une transition entre états se produise. Plusieurs outils ont été développés pour permettre de faire le model-checking probabiliste automatique. PRISM, qui est l'outil qui nous intéresse dans le cadre de ce travail, est un model-checker probabiliste qui supporte le model-checking pour différents types de modèles probabilistes comme les chaînes de

Markov à temps discret ou DTMC (de l'anglais *Discrete-Time Markov Chain*), les processus de décision de Markov ou MDP (de l'anglais *Markov Decision Process*), les chaînes de Markov à temps continu ou CTMC (de l'anglais *Continuous-Time Markov Chains*), etc. En entrée, l'outil prend deux paramètres, une description du modèle dans le langage PRISM et une liste de spécifications du modèle. Il détermine ensuite quels états du modèle satisfont chaque spécification donnée. Dans le cadre de ce travail,nous avons choisi d'analyser une réseaux véhiculaire particulier comme une étude du cas d'utilisation de notre outil .

Le sujet de ma maîtrise consiste à donne de l'aide à localisation des erreurs dans des modèles PRISM à partir un contre-exemple. Le problème du localisation des erreurs est difficile car le contre-exemple contient une quantités trés important des informations (plusieurs chemins) .Cependant, la génération de petites et indicatifs des contre-exemple ne suffit pas pour comprendre l'erreur.Pour résoudre ce problème il y a une technique de diagnostiquer des contre-exemple probabiliste par ICL qui a été explique brièvement dans le chapitre 4 . L'enjeu est de rendre automatique la preuve de telles technique, et d'exhiber des diagnostique lorsqu'une propriété est non satsfaite.

Les travaux futurs concernent à la fois une réflexion sur le raissonnement par par abduction probabiliste ICL (dans le cadre de l'analyse des contre-exemple probabilistes composées d'un grand nombre de chemins finis.) et l'optimisation du l'outil ELPMC de la construction des marquage de modèle erroné par raffinement des diagnostique pour fixer uniformément la correction des erreurs .

Bibliographie

[1] G. NORMAN et D. PARKER A. HINTON M. KWIATKOWSKA. « PRISM : A tool for automatic verification of probabilistic systems ». In : *12th International Conference on Tools and Algorithm for the Construction and Analysis of Systems (TACAS'06)* (2006).

[2] G. ACOSTA-MARUM. « Wave : A tutorial ». In : *IEEE Communications Magazine,* (May 2009).

[3] Vigyan SINGHAL et Robert BRAYTON ADNAN AZIZ Kumud SANWAL. « Model-checking continuous-time markov chains ». In : *ACM Transactions on Computational Logic* (2000).

[4] H. ALJAZZAR et S. LEUE. « Directed explicit state-space search in the generation of counterexamples for stochastic model checking ». In : *IEEE Transactions on Software Engineering* (2010).

[5] « ASTM International, Standard Specification for Telecommunications and Information Exchange Between Roadside and Vehicle Systems - 5 GHz Band Dedicated Short Range Communications (DSRC) Medium Access Control (MAC) and Physical Layer (PHY) Specifications ». In :

[6] A. et B. R. Haverkort BELL. « Untold horrors about steady-state probabilities : What reward-based measures won't tell about the equilibrium distribution ». In : *Formal Methods and Stochastic Models for Performance Evaluation.*

[7] Mustapha BOURAHLA. « Repairing Errors in PRISM Programs Using Probabilistic Abduction Reaso ». In : *Model and Data Engineering* (2015).

[8] H. Hermanns C. BAIER B. Haverkort et J.-P. KATOEN. « Model-checking algorithms for continuous-time Markov chains ». In : *IEEE Transactions on Software Engineering* (2003).

[9] Jon Crowcroft Marta Kwiatkowska Robin Milner Eammon O'Neill Tom Rodden Vladimiro Sassone DAN CHALMERS Matthew Chalmers et Morris SLOMAN. *Ubiquitous computing : Experience, design and science*. Rapp. tech. 2006.

[10] Edmund M. CLARKE et E. ALLEN EMERSON. « Design and synthesis of synchronization skeletons using branching-time temporal logic ». In : *Dexter KOZEN*.

[11] E. Allen EMERSON et Joseph SIFAKIs EDMUND M. CLARKE. « Model checking : algorithmic verification and debugging ». In : *Communications of the ACM* (2009).

[12] ETSI. *intelligent transport systems (its) ; european profile standard for the physical and medium access control layer of intelligent transport systems operating in the 5 ghz frequency band*. Rapp. tech. etsi std, 2009.

[13] Ansgar FEHNKER et Peng GAO. « Formal verification and simulation for performance analysis for probabilistic broadcast protocols ». In : *Proceedings of the 5th International Conference on Ad-Hoc, Mobile, and Wireless Networks (ADHOC-NOW 2006)*.

[14] D. Flitzanis G.F. MARIAS P. Georgiadis et K. MANDALAS. « Cooperation enforcement schemes for manets : A survey ». In : *Wireless Communications and Mobile Computing* (May 2006).

[15] Holger HERMANNS et Joost Pieter KATOEN GHRISTEL BAIER Boudewijn R. HAVERKORT. « Model-checking algorithms for continuous-time markov chains ». In : *IEEE Transactions on Software Engineering (TSE)* (2003).

[16] Jane HILLSTON. « Process algebras for quantitative analysis ». In : *LIOS* (2005).

[17] Jane HILLSTON. « Tuning systems : From composition to performance (the Needham lecture ». In : *The Computer Journal* (2005).

[18] *http ://www.anfr.fr/fr/l-anfr/organisation/le-cadre-europeen/ cept-ecc.html*. •.

[19] *http ://www.dcs.ed.ac.uk/pepa/*. 2016.

[20] « IEEE Recommended Practice on Software Reliability ». In : *IEEE STD 1633-2008* (Juin 2008).

[21] J. JAKUBIAK et Y. KOUCHERYAVY. « State of the art and research challenges for vanet ». In : *Consumer communications and networking conference (CCNC' 08)* (2008).

[22] SOMMER C. JOERER S. Dressler F. « Comparing Apples and Oranges ? Trends in IVC Simulations ». In : *9 th ACM International Workshop on Vehicular Internetworking (VANET 2012)* (2012).

[23] Jean-Pierre QUIELLE et JOSEPH SIFAKIS. « Spécification and verification of concurrent systems in cesar ». In : *Mariangiola DEZANI-CIANCAGLINI et Ugo MON- TANARI* (1982).

[24] Arne KESTING. « Microscopic Modeling of Human and Automated Driving : Towards Traffic-Adaptive Cruise Control ». Thèse de doct. Faculty of Traffic Sciences, 2008.

[25] R. KUMAR et M. DAVE. « A review of various vanet data dissemination protocols ». In : *International Journal of U- and E-Service, Science and Technology* (2012).

[26] G. NORMAN et D. PARKER M. KWIATKOWSKA. « PRISM 4.0 : Verification o probabilistic real-time systems ». In : *International Conference on Computer Aided Verification (CAV'I 1).*

[27] N.H.T.S.A. *Rapport mondial sur la prévention des traumatismes dus aux accidents de la circulation*. Rapp. tech. March 2005.

[28] *NS3 : Network Simulator 3 www.nsnam.org (2016).*

[29] *OMNeT++ : http ://www.omnetpp.org/ (2016).*

[30] Dr.Jody PAUL. « Testing and Debugging ». In : *http ://www.jodypaul.com/SWE/TD/TestDebug.html* (2016).

[31] Amir PNUELI. « The temporal logic of programs ». In : *In Proceedings of the 18t Annual Symposium on Foundations of Computer Science (FOCS'77)* (1977).

[32] Amir PNUELI. « Vérification engineering : A future profession (a. m. turing award lecture) ». In : *PODO* (1997).

[33] *Prism manual version 4.3.* 2016.

[34] *Requirements Engineering*. Springer, 2010.

[35] M.S. Mousavi S. YOUSEFI et M. FATHY. « Vehicular ad hoc networks (vanets) : challenges and perspectives ». In : *Proceedings of the 6th International Conference on ITS telecommunications* (2006).

[36] M. L. SICHITIU et M. KIHL. « Inter-vehicle communication systems : A survey ». In : *IEEE Communications Surveys and Tutorials* (2008).

[37] *Site internet de PRISM : http ://www.prismmodelchecker.org*. 2016.

[38] *Systems and Software Verification. Model-Checking Techniques and Tools*. Springer, 2001.

[39] *Vehicular networks : from theory to practice*. 2009.

[40] Eric Wong et VIDROHA DEBROY. *A survey of software fault localisation*. Rapp. tech. Département of computer science ,University of Texas aT Dallas, 2009.

[41] W Eric Wong et VIDROHA DEBROY. « Software Fault Localisation ». In : *Encyclopedia of Software Engineering* (2010).