Master Informatique

“Preuve mécanisée de l’algorithme de Tarjan”

Rapport
en vue de la validation de
l’UE Initiation à la recherche

2019-2020

Étudiants : Angela Ipseiz
Maxime Nicolas
Matthieu Olejniczak

Encadrant : Monsieur MERZ

Décharge de responsabilités

L'Université de Lorraine n'entend donner ni approbation ni improbation aux opinions émises dans ce rapport, ces opinions devant être considérées comme propres à leur auteur.

Remerciements

Sommaire

Introduction

« Nous dépendons de plus en plus des logiciels, et il est important qu'ils fonctionnent de mieux en mieux. »[(1)] déclare Leslie Lamport, pionnier de l'algorithme distribué et du concept de preuve algorithmique. En effet, les logiciels sont de plus en plus présents dans notre quotidien et pour certains, notre sécurité en dépend, comme les logiciels utilisés dans l'aviation ou le ferroviaire. De nos jours, un programme incertain, c'est-à-dire avec au moins une condition d'utilisation qui rend le résultat incorrect, peut engendrer d'énormes conséquences financières, humaines,.... C'est pourquoi, il est impératif de réaliser des **preuves d'algorithmes**, notamment pour ceux qui sont liés directement à la sécurité, comme celles qui ont été faites pour l'automatisation de la ligne de métro 14 à Paris, ou le respect de la vie privée que doivent garantir les entreprises du Cloud.

Différents types d'algorithmes peuvent nécessiter une preuve. En effet, il existe des algorithmes impliquant des **graphes** qui sont au coeur de nombreux problèmes actuels. Par exemple, la résolution d'un sudoku peut être transformée en problème 2-SAT qui détermine s'il existe une solution possible grâce à des graphes. Il est des graphes, des structures de données représentées par un ensemble de sommets et d'arêtes. Un graphe est **orienté** si ses arêtes ne sont parcourables que dans un seul sens dont la direction est représentée par une flèche. Le problème 2-SAT utilise la décomposition en **composantes fortement connexes**. On appelle composantes fortement connexes un sous-graphe maximal G' d'un graphe orienté G tel que, pour tout couple (u,v) de noeuds de G,' il existe un chemin de u à v. Ainsi, un graphe fortement connexe est un graphe formé d'une seule composante fortement connexe.

Pour les identifier, on peut utiliser différents algorithmes tels que l'algorithme de Kosaraju et l'**algorithme de Tarjan**, tous deux fondés sur un algorithme de parcours en profondeur en temps linéaire. La principale différence entre ces deux algorithmes est que celui de Tarjan identifie les composantes fortement connexes en parcourant une seule fois le graphe au lieu de deux pour celui de Kosaraju. Nous voulons ici nous intéresser à l'algorithme de Tarjan : Est-ce que l'algorithme de Tarjan nous donne l'ensemble des composantes fortement connexes quelque soit le graphe orienté sur lequel on l'applique ?

Pour répondre à cette problématique, nous regarderons dans un premier temps le bon fonctionnement de l'algorithme par le biais de l'outil TLA+. Ensuite, nous nous intéresserons à sa preuve. Enfin, nous étudierons les problèmes que nous avons rencontrés et comment nous les avons réglés.

Algorithme de Tarjan

Fonctionnement de l’algorithme

- explication de l’algorithme

Correction de l’algorithme

- TLA+
- Model Checking

Preuve de l’algorithme de Tarjan

Outils employés

- TLAPS
- invariant

Utilisation de ces outils à l’algorithme de Tarjan

- explication de la preuve

Problèmes et solutions apportées

Transcription de l'algorithme en TLA+

- Problème de transcription
- PlusCal

Preuve de l'algorithme

- difficultés pour les parties non déterministes
- difficultés pour les parties reposants sur le fonctionnement de la pile

Conclusion

Il arrive de croire que le programme que l'on a créé fonctionne, alors que ce n'est pas le cas. Pour pouvoir passer de croyance à certitude, on utilise des logiciels spécialisés pour nous aider à connaître la correction de ce que l'on a conçu. Mais savoir que quelque chose fonctionne ne suffit pas, il faut pouvoir le prouver. Ici, nous avons pu voir à l'aide d'un model checker que l'algorithme fonctionnait sur des exemples de graphes. Puis, avec l'assistance d'un prouveur, nous avons pu avoir la certitude que l'algorithme de Tarjan pouvait s'appliquer à n'importe quel graphe orienté pour en identifier toutes les composantes fortements connexes.

Évidemment, l'algorithme de Tarjan n'est qu'un cas d'étude ici. La méthode que nous avons utilisé peut être appliquée à n'importe quel algorithme qui nécessite d'être prouvé. En effet, si nous développons un algorithme qui va être utilisé pour gérer le trafic ferroviaire, il est nécessaire de s'assurer qu'il fonctionne dans n'importe quelle situation. Le domaine de la preuve est un domaine émergeant sur lequel de grandes entreprises comme Amazon investissent de plus en plus, c'est pourquoi on peut espérer un jour savoir pertinemment ce que l'on fait.

Annexes

Bibliographie

« Ça ne sert à rien de commencer à programmer si l'algorithme lui-même est mauvais » Leslie Lamport à propos de TLA+.
(1)https://www.lemonde.fr/sciences/article/2014/06/16/leslie-lamport-un-informaticien-dans-les-nuages_4439171_1650684.html

https://fr.wikipedia.org/wiki/Qualit%C3%A9_logicielle
https://fr.wikipedia.org/wiki/Graphe_orient%C3%A9

http://zanotti.univ-tln.fr/ALGO/I31/Complexite.html#def:successeur

Déclaration sur l’honneur contre le plagiat

Je soussigné(e),

Ipseiz Angela

Régulièrement inscrit à l’Université de Lorraine

N° de carte d’étudiant : 31608993

Année universitaire : 2019-2020

Niveau d’études : M

Parcours : Informatique

N° UE : 811

Certifie qu’il s’agit d’un travail original et que toutes les sources utilisées ont été indiquées dans leur totalité. Je certifie, de surcroît, que je n’ai ni recopié ni utilisé des idées ou des formulations tirées d’un ouvrage, article ou mémoire, en version imprimée ou électronique, sans mentionner précisément leur origine et que les citations intégrales sont signalées entre guillemets.

Conformément à la loi, le non-respect de ces dispositions me rend passible de poursuites devant la commission disciplinaire et les tribunaux de la République Française.

Fait à Vandœuvre-lès-Nancy, le……..

Signature :

Déclaration sur l'honneur contre le plagiat

Je soussigné(e),

Nicolas Maxime

Régulièrement inscrit à l'Université de Lorraine

N° de carte d'étudiant : 31509056

Année universitaire : 2019-2020

Niveau d'études : M

Parcours : Informatique

N° UE : 811

Certifie qu'il s'agit d'un travail original et que toutes les sources utilisées ont été indiquées dans leur totalité. Je certifie, de surcroît, que je n'ai ni recopié ni utilisé des idées ou des formulations tirées d'un ouvrage, article ou mémoire, en version imprimée ou électronique, sans mentionner précisément leur origine et que les citations intégrales sont signalées entre guillemets.

Conformément à la loi, le non-respect de ces dispositions me rend passible de poursuites devant la commission disciplinaire et les tribunaux de la République Française.

Fait à , le……..

Signature :

Déclaration sur l'honneur contre le plagiat

Je soussigné(e),

Olejniczak Matthieu

Régulièrement inscrit à l'Université de Lorraine

N° de carte d'étudiant : 31506421

Année universitaire : 2019-2020

Niveau d'études : M

Parcours : Informatique

N° UE : 811

Certifie qu'il s'agit d'un travail original et que toutes les sources utilisées ont été indiquées dans leur totalité. Je certifie, de surcroît, que je n'ai ni recopié ni utilisé des idées ou des formulations tirées d'un ouvrage, article ou mémoire, en version imprimée ou électronique, sans mentionner précisément leur origine et que les citations intégrales sont signalées entre guillemets.

Conformément à la loi, le non-respect de ces dispositions me rend passible de poursuites devant la commission disciplinaire et les tribunaux de la République Française.

Fait à Vandœuvre-lès-Nancy, le……..

Signature :

(Quatrième de couverture)