

Analyse de la Propagation basée sur les Graphes Logiciels et les Données Synthétiques

Titre original: Propagation Analysis based on Software Graphs and Synthetic Data

Vincenzo Musco

Les programmes sont omniprésents dans notre vie quotidienne : dans les ordinateurs et les téléphones, mais aussi les réfrigérateurs, les avions... Le principal acteur dans le processus de création de ces programmes est l'homme. Aussi minutieux qu'ils puissent être, les hommes sont connus pour faire des erreurs involontaires. Ainsi, une fois finie la phase déjà difficile d'écriture du programme, ils doivent faire face à la phase de maintenance au cours de laquelle ils doivent gérer les erreurs qu'ils ont commises.

Ainsi, pour aider les développeurs, un grand nombre d'outils existent, certains toujours en développement, d'autres sont intégrés dans l'IDE (débugage, suite de tests, refactoring...). Plusieurs de ces outils sont manuels tandis que d'autres proposent une assistance automatique. Pour être efficaces, les outils automatisés doivent capturer du mieux possible la façon dont le programme est structuré et fonctionne. Dans ce but, une structure de données particulièrement bien adaptée est le graphe, matérialisant la façon dont les concepts sont liés les uns aux autres. Utiliser de telles structures de données pour proposer des outils d'assistance aux développeurs est prometteur.

Dans cette thèse, nous nous concentrons sur des outils basés sur une représentation sous forme de graphe du programme. Nous nous intéressons à deux problèmes majeurs : l'analyse d'impact au changement (CIA) et la localisation de fautes (FL). Le premier se concentre sur la détermination d'impacts d'un changement potentiel qui peut être publié par le développeur, tandis que le second identifie une faute à partir de ce qui s'est passé durant l'exécution du programme. Les deux concepts sont complémentaires : le premier s'occupe du côté ante-mortem du problème où on veut identifier une faute avant que l'erreur ne se produise, tandis que le deuxième travaille sur le côté post-mortem où des erreurs réelles sont analysées pour définir une façon de retourner à leurs sources.

Dans cette thèse, nous faisons face à deux problèmes principaux : (i) le manque de méthodologie d'évaluation systématique ou de framework pour évaluer la performance des techniques d'analyse d'impact au changement et (ii) les techniques les plus courantes de la localisation de fautes se concentrent sur un ensemble spécifique d'éléments rapportés par leur approche sans tenir compte de la façon dont ils dépendent les uns des autres à travers le programme dans son entièreté.

Dans cette thèse, nous avons pour but de trouver des solutions à ces problèmes. Nous présentons quatre contributions pour répondre aux deux problèmes présentés. Les deux premières contributions ciblent principalement l'analyse d'impact au changement tandis que la troisième travaille sur la localisation de fautes. La dernière contribution est une possible application pour des travaux futurs. Pour résumer, cette thèse explore les causes et les conséquences des erreurs dans les programmes informatiques en proposant des outils basés sur les graphes.