THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par : *l'Université Toulouse 3 Paul Sabatier (UT3 Paul Sabatier)*

Présentée et soutenue le *21/12/2017* **par :**

Jordy Ruiz

Détermination de propriétés de flot de données pour améliorer les estimations de temps d'exécution pire-cas

JURY

Philippe CLAUSS	Professeur d'Université	Rapporteur
Nicolas HALBWACHS	Directeur de Recherche	Rapporteur
Sandrine BLAZY	Professeur d'Université	Examinateur
Jean-Paul BODEVEIX	Professeur d'Université	Examinateur
Christine ROCHANGE	Professeur d'Université	Directeur de thèse
Hugues CASSÉ	Maître de Conférences	Co-directeur de thèse

École doctorale et spécialité :
MITT : Domaine STIC : Réseaux, Télécoms, Systèmes et Architecture
Unité de Recherche :
Institut de Recherche en Informatique de Toulouse (UMR 5505)
Directeur(s) de Thèse :
Christine ROCHANGE et *Hugues CASSÉ*
Rapporteurs :
Philippe CLAUSS et *Nicolas HALBWACHS*

Lookup of data flow properties to improve worst-case execution time estimations

Jordy Ruiz

Abstract

The search for an upper bound of the execution time of a program is an essential part of the verification of real-time critical systems. The execution times of the programs of such systems generally vary a lot, and it is difficult, or impossible, to predict the range of the possible times. Instead, it is better to look for an approximation of the *Worst-Case Execution Time*.

A crucial requirement of this estimate is that it must be *safe*, that is, it must be guaranteed above the real WCET. Because we are looking to prove that the system in question terminates reasonably quickly, an overapproximation is the only acceptable form of approximation.

The guarantee of such a safety property could not sensibly be done without static analysis, as a result based on a battery of tests could not be safe without an exhaustive handling of test cases. Furthermore, in the absence of a certified compiler (and technique for the safe transfer of properties to the binaries), the extraction of properties must be done directly on binary code to warrant their soundness.

However, this approximation comes with a cost : an important pessimism, the gap between the estimated WCET and the real WCET, would lead to superfluous extra costs in hardware in order for the system to respect the imposed timing requirements. It is therefore important to improve the *precision* of the WCET by reducing this gap, while maintaining the safety property, as such that it is low enough to not lead to immoderate costs.

A major cause of overestimation is the inclusion of semantically impossible paths, said *infeasible paths*, in the WCET computation. This is due to the use of the *Implicit Path Enumeration Technique* (IPET), which works on an superset of the possible execution paths. When the *Worst-Case Execution Path* (WCEP), corresponding to the estimated WCET, is infeasible, the precision of that estimation is negatively affected.

In order to deal with this loss of precision, this thesis proposes an infeasible paths detection technique, enabling the improvement of the precision of static analyses (namely for WCET estimation) by notifying them of the infeasibility of some paths of the program. This information is then passed as data flow properties, formatted in the FFX portable annotation language, and allowing the communication of the results of our infeasible path analysis to other analyses.

The methods hereafter presented are included in the OTAWA framework, developed in TRACES team at the IRIT lab. They themselves make use of approximations in order to represent the possible states of the machine in various program points. These approximations are *abstractions* maintained throughout the analysis, and which validity is ensured by *abstract interpretation* tools. They enable us to represent the set of states for a family of execution paths up to a given program point in an efficient – yet safe – way, and to detect the potential program points associated to an empty set of possible states, signalling one (or several) infeasible path(s).

As the end goal of the developed analysis, the detection of such cases is made possible by the use of *Satisfiability Modulo Theory* (SMT) solvers. Those solvers are notably able to determine the satisfiability of a set of contraints, which we deduct from the abstract states. If a set of constraints, derived from a conjonction of predicates, is unsatisfiable, then there exists no valuation of the machine variables that match a possible execution case, and thus the associated infeasible paths are infeasible.

The efficiency of this technique is asserted by a series of experiments on various benchmarks suites, some of which widely recognized in the domain of static WCET, some others derived from actual industrial applications. Heuristics are set up in order to soften the complexity of the analysis, especially for the larger applications. The detected infeasible paths are injected as *Integer Linear Programming* (ILP) linear data flow constraints in the final computation for the WCET estimation in OTAWA. Depending on the analysed program, this can result in a reduction of the estimated WCET, thereby improving its precision.

Keywords : real-time systems, WCET, static analysis, abstract interpretation, SMT, infeasible paths, machine language.