発表概要

入力空間とイベント空間を探索するJavaScriptコードの等価性検証

冨永 江奈[1,a)]　荒堀 喜貴[1]　権藤 克彦[1]

2018年11月1日発表

コードの等価性検証の技術は，書き換え前後のコードが同様に動作するか，同じインタフェースの異なる実装がまったく同じ動作をするか等を確認する場面において，非常に有用なものである．コードの等価性検証を行うツールには，C 言語の 2 つの関数の等価性の検証を行うツール UC-KLEE が存在する．UC-KLEE は遅延初期化に基づく記号実行により，動的データ構造操作も含めた正確な検査が可能である．しかし，UC-KLEE ではイベント処理の順序に起因するイベント空間を考慮していない．そのため，C 言語とは異なりイベントの発火順序が実行結果に大きく影響する JavaScript コードの等価性検証においては，UC-KLEE の単純拡張は不十分である．そこで，本発表では入力空間とイベント空間の両方の探索を行う，JavaScript における関数の等価性検証ツールのプロトタイプ実装を提案する．入力空間の探索はコンコリック実行によりできるだけ多くの実行パスを通すことで，イベント空間の探索はイベントの発火順序をランダムに操作することで行う．入力空間に加えてイベント空間も操作することで，ある特定の順序でイベントが発火したときにのみ通るパスの探索も可能となる．加えて，入力空間に加えてイベント空間を考慮する提案手法の等価性検査の有効性を定量的に示すことを目的とした評価実験を行った．これは，入力空間しか考慮しない従来の UC-KLEE の単純拡張を比較対象とし，いくつかの小規模テストケースを用いて入力空間とイベント空間の双方を考慮する提案手法の等価性検査の精度，効率を計測することにより行う．

Presentation Abstract

WIP: Towards Equivalence Verification of JavaScript Code

Ena Tominaga[1,a)]　Yoshitaka Arahori[1]　Katsuhiko Gondow[1]

Presented: November 1, 2018

The technique of equivalence verification of code is very useful in a situation where it is confirmed whether codes before and after reworking operate similarly, whether different implementations of the same interface perform exactly the same operation, and the like. UC-KLEE is a tool which verifies the equivalence of two functions in C language. UC-KLEE can perform accurate inspection including dynamic data structure operation by symbolic execution based on lazy initialization. However, UC-KLEE does not consider the event space due to the order of event processing. Therefore, unlike in C language, the simplicity extension of UC-KLEE is insufficient in equivalence verification of JavaScript code in which order of event occurrence greatly affects execution result. Therefore, in this presentation we propose a prototype implementation of a function equivalence verification tool in JavaScript which explore both input space and event space. Exploring the input space is performed by passing as many execution paths as possible through a concolic execution, and exploring the event space is performed by randomly manipulating the event firing sequence. By manipulating the event space in addition to the input space, it is also possible to search for a path that passes only when the event occurs in a specific order. In addition, we conducted an evaluation experiment aiming to quantitatively show the effectiveness of the equivalence check of the proposed method considering the event space in addition to the input space. To do this, we measure the accuracy and efficiency of the equivalence check of the proposed method considering both the input space and the event space by using several small scale test cases and compare simple extension of UC-KLEE, which only considers input space.

This is the abstract of an unrefereed presentation, and it should not preclude subsequent publication.

[1] 東京工業大学情報理工学院情報工学系
Department of Computer Science, School of Computing, Tokyo Institute of Technology, Meguro, Tokyo 152–8552, Japan
[a)] tominaga@sde.cs.titech.ac.jp