
発表概要

ランダムテストプログラムによる Java JIT コンパイラ検証システム

吉川 隆英[†] 志村 浩也[†] 小沢 年弘[†]

Java 言語はサーバから携帯機器まで幅広い分野で使われるようになってきている。この処理系の動作検証を行うために、Sun は Technology Compatibility Kits (TCK) を出している。しかしこれは個々のバイトコード命令を対象としたものであり、JIT (Just-in-Time) コンパイラの動作検証を行うテストとしては不十分である。JIT の動作検証を十分に行うためには、数多くの命令列の並びを網羅したテストプログラムを実行し、その動作を検証することが望ましい。この膨大な組合せを網羅した多量のテストプログラムを一々人手で作ることは、非現実的である。そこで今回抽象実行系の機能を包摂したランダムテストプログラム生成器を作成し、1) バイトコードベリファイアを通り、かつ無限ループに陥らないランダムなバイトコード列を、抽象実行しながら逐次生成していき、2) 実行経過を反映した実行結果を出力させるコードを追加し、3) そのクラスファイルを検査対象処理系と他の比較対照用処理系とで実行、4) 両者の実行結果を比較することによって検査対象処理系の動作正当性検証を自動的に行うシステムの開発を行った。本発表では、まずこのランダムテストプログラム生成手法と、それを用いた動作検証手法を述べ、次いで生成されたランダムテストプログラムの性質（実行時間のばらつき、命令の網羅性）の評価を行う。

Random Test Program Generator for Java JIT Compiler Verification System

TAKAHIDE YOSHIKAWA,[†] KOUYA SHIMURA[†] and TOSHIHIRO OZAWA[†]

Java has been used widely, from servers to cellular phones. To verify its runtime systems, Sun Micro Systems provides “Technology Compatibility Kits (TCK)”. But this checks each instructions individually, thus this is insufficient for a Java JIT (Just-in-Time) Compiler. Various possible sequences of bytecodes must be test for a JIT compiler. It is infeasible to make those enormous test programs manually. Therefore, we developed a random test program generator, with an abstract execution engine, and a JIT compiler verification system. This system automatically 1) does abstract execution and randomly generates valid bytecodes, which can pass a bytecode verification and never loops infinitely; 2) inserts codes, which output a result representing its execution progress; 3) runs this generated test program on the target JIT compiler and a reference system; 4) compares both results. In this presentation, we describe how to generate random test programs, and how to test JIT compilers in practical use, and then we evaluate a quality of generated programs (dispersion of execution times and comprehensiveness of each instructions).

(平成 15 年 1 月 23 日発表)

[†] 株式会社富士通研究所
Fujitsu Laboratories LTD.