

A Compositional Deadlock Analysis for Code Commits in Android Java

(Invited Talk)

Nikos Gorogiannis

Middlesex University, UK

`nikos.gorogiannis+www@gmail.com`

We report on a static deadlock analysis for Android Java applications under active development at Facebook.

The analysis targets an abstract language which approximates Java, by adding behaviours (non-deterministic iteration and branching) and by removing some others (only balanced re-entrant global locks, non-recursive procedure calls). We show that the existence of a deadlock in this abstract language is equivalent to a certain condition over the sets of so-called critical pairs of each program thread. These record, for all possible executions of the thread, which locks are currently held at the point when a fresh lock is acquired. Since the set of critical pairs of any program thread is finite and computable, the deadlock detection problem for our language is decidable, and in NP.

We then leverage these results to develop an open-source implementation of our analysis adapted to deal with real Java code. The core of the implementation is an algorithm which computes critical pairs in a compositional, abstract interpretation style, running in quasi-exponential time. Our analyser is built in the Infer verification framework and has been in deployment at Facebook for over two years; we report on developer adoption and impact.