Generating Data Race Witnesses by an SMT-based Analysis

Mahmoud Said, Chao Wang, Zijiang Yang, and Karem Sakallah3

Motivations

Data races exhibit many different behaviors

A single problem can take weeks for programmers to identify.

Previous techniques don’t scale, none can reliably reproduce the data race

Solution

Use a trace and existing data race detection algorithms to give a set of potential data races.

A **maximal set** of possible traces by setting constraints on events in the trace

If there are recursive locks, the events are encoded differently

Another improvement is using FOL so SMT solvers enhance performance instead of search algorithms

Solution Cont.

$$\psi_\pi := \alpha_\pi \wedge \beta_\pi \wedge \gamma_\pi \wedge \rho_\pi$$

Partial Order

Write-Read Consistency - the value read by an event is always written by the most recent write:

1. e has no thread immediate write predecessor
2. e follows a write event e1 in its predecessor write

Synchronization consistency - Setting up a variable for objects for thread num and waitlist

Recursive-lock - two events both acquire with no release in between

Evaluation

Proof of Correctness by definition of the constraints

Eliminates bogus warnings

Efficiently Decidable subset of FOL

Yices SMT Solver - construct formula for all data races

Reduce possible data races siginificantly

Evaluation Cont.

Only up to Medium Traces

Not very feasible to use

Contribution

Improve existing data race detection algorithms to give all possible witnesses for data races

Improve runtime greatly by using FOL (no evidence given in paper)

Can be applied to future data race detection algorithms as well

Future Directions

Utilizing newer data race detection algorithms

Evolution aware

Questions

How come a large project wasn't used, it doesn't really seem that scalable in their claims since I'm sure people would want to use this for larger traces

Is it possible to pinpoint the more likely witnesses for data races?

What exactly do recursive locks do to a program other than using for constraints?