Multi-threaded software is ubiquitous, a lot of communication happens via global variables. Thread-modular analyses needed to avoid state explosion.

Approaches from literature

Miné’s style (e.g. [2])
- Propagate values from unlocks of a mutex to its locks, provided appropriate side-conditions are met
- Compute set of protecting mutexes always held when global is accessed
- Publish value on unlock of last protecting mutex
- Identify weaknesses and propose improved versions
- Principled soundness proofs for both styles of analyses
- Comparison
- Equally precise for 11/13 benchmarks
- For pfscan and yppbind: [2] less precise for 6% resp. 16% of globals

Vojdani’s style (e.g. [5, 6])
- Accumulate flow-insensitive information for globals during flow-sensitive analysis of locals
- Formulation of both styles in a common framework
- Increases with sophistication
- Relations between globals are likely to be mediated by mutexes
- Further finite abstractions (e.g. thread id)
- Tracking bigger clusters may be both more and less precise?!?
- For certain domains (e.g. Octagons), tracking subclusters of size ≤ 2 already yields maximum precision

Contributions in Non-Relational Setting

- Formulation of both styles in a common framework
- Equally precise for 11/13 benchmarks
- For pfscan and yppbind: [2] less precise for 6% resp. 16% of globals

Side-Effecting Equation Systems [1]

Accumulate flow-insensitive information for globals during flow-sensitive analysis of locals

$$(u, \text{unlock}(b)) \eta = \text{Let} \; \sigma' = \ldots \in \{(\mu g \rightarrow (\eta [u]) g \mid g \in \ldots), \sigma'\}$$

Side-Effects Contribution to $$\sigma'$$

Ingredients for More Precise Analyses

Consider further finite abstraction to exclude more reads

For each global $$g$$
- $$W_g$$: Set of locksets held when last writing to $$g$$
- $$P_g$$: Set of locksets held since last writing to $$g$$

For each mutex $$a$$
- $$L_a$$: Set of locksets held when last acquiring $$a$$
- $$V_a$$: Set of globals that must have been written locally since last acquiring $$a$$

References


