SOBER: Statistical Model-based Bug Localization

C. Liu, X. Yan, J. Han, L. Fei, S. P. Midkiff

Motivations

- **Software Bugs and Dependability**
  - Ariane 5 explosion is due to “errors in the software of the inertial reference system”
  - Software faults cost the U.S. economy about $59.5 billion annually (NIST 2002)
- **Debug, debug and debug ...**
  - “50% of my company employees are testers, and the rest spends 50% of their time testing!”
    --Bill Gates, in 1995

Bug as Model Divergence

- **Program predicates**
  - Encode program execution
  - Embody program logic
- **Model divergence**

Indirect Calibration

- **Direct calibration is hard**
  - No prior knowledge of model
  - Not authorized for model assumption
- **Calibrate it indirectly**
  - Null hypothesis: $f(X|\theta_1) = f(X|\theta_2)$
  - Derive a statistic that conforms to normal distribution by Central Limit Theorem
  - Likelihood of the realized statistic represents the unlikelihood that $f(X|\theta_1) = f(X|\theta_2)$ hence the model divergence

Mass Experiment

- **Bug Benchmark**
  - 130 Known bugs
  - Mainly logic bugs
- **Quality Metrics**
  - Evaluated on dependence graph
  - Indicates extra human efforts
- **Compare with two algorithms**
  - Both claim state-of-the-art
  - The higher, the better

Conclusions

- **Software bugs and dependability**
- **SOBER can help developers localize software bugs for productivity.**
- **SOBER requires no information about program semantics.**
- **Data mining, machine learning, and statistical inference may find their ways into computer systems for performance and reliability.**

---