

CSci 780: Advanced Software Engineering

Some Project Ideas

- Apply automated analysis to existing open source projects
- Implement Dawson Engler's approach to finding bugs
- Write a formal specification of a voting system.
- Write a program using design by contract, ESC Java, etc.
- Write a specification of a protocol in Alloy and analyze it with respect to certain theorems
- Do an experiment to characterize the scalability of TestEra.
- Write a formal specification of a traffic simulator.

- Do specification-derived assertions outperform programmer-derived assertions? We already have a system that has both. Jennifer Haddock-Schatz worked on this, but the work has some problems: (1) potential bias in the way that faults were injected, (2) not enough analysis of the costs, (3) not enough description of the process of mapping specs to assertions. Find a real system with real faults and a specification, and redo the experiment, addressing the concerns above. We have the Nova solver system, which can fit the bill.
- Does bounded exhaustive testing find bugs as effectively as coverage-based testing? Ashwin Mundra worked on this for his Master's, but unfortunately removed faults during the experiment instead of leaving them in, so we don't know if BET could have found them. (He tried to see if BET could find *additional* faults, when he should have first verified that it could find the ones that coverage-based testing found.) Redo the experiment. - Perform BET on an NIA system for aircraft collision avoidance. They say they have proven the algorithm correct, but is the implementation correct?