Project Proposal

Uday Shankar <us@andrew.cmu.edu>

Project Choice: Synchronous model checking.

Preliminary Thoughts: I will begin by making a "barebones" implementation using the closure algorithms to compute product and power automata, as these should prevent ridiculous blowups when they don't actually need to occur. Once this is done, I will need to do some experimentation to find out where the slowdowns occur, but I expect that they will essentially all be related to the size of the automaton representing the subformulas growing too large. My first attempt to fix this will be to implement and apply standard finite automaton minimization algorithms (which work because synchronous automata are just normal finite automata over a product alphabet), but this will involve a determinization step before hand (the minimization algorithms we studied require a DFA as input) so this could actually end up being too expensive. Perhaps I will only apply the minimization step after a computation that is guaranteed to produce a deterministic automaton as output (for instance processing a negation). After this, if I still have time, I will build a wrapper UI that tries to make the project easily demo-able. The UI will allow the user to write formulae and draw out the finite automata corresponding to the relations that they use, and then it will produce the truth judgement and maybe the intermediate automata produced by the algorithm as well.

The language of choice will be Standard ML, not because I particularly like the language, but because I think that the graph algorithms involved might benefit from the parallelism provided by some SML compilers, such as MLton.