CISC-499 Projects 2015–16

Kai Salomaa
School of Computing, Queen's University
ksalomaa [AT] cs [DOT] queensu [DOT] ca

1. **Converting machines to regular expressions**

   Grail is a symbolic computation environment for finite state automata, regular expressions and other language objects. Grail's `fmtore` function converts finite state machines to regular expressions using the state elimination algorithm. This algorithm tends to generate very large and redundant regular expressions, partly because it does no simplification. In this project you will develop and implement heuristics for simplifying regular expressions and apply these heuristics to the problem of converting finite state machines to regular expressions. The tasks can include the following:

   - investigate current work on simplification of regular expressions
   - study the expressions that tend to be generated in conversion of finite state machines, and characterize the types of simplification that would be useful
   - develop and implement a simplification heuristics for regular expressions
   - test the efficacy of your algorithms by using them in the conversion of finite state machines to regular expressions
   - (optional, if time permits) evaluate the cost of the simplification heuristics and develop a metric for deciding when to use them

   The project requires an understanding of the basics of finite automata and regular expressions. The amount of programming required is not large, but you should expect to run a significant number of simulations and other experiments.

2. **Template-guided recombination**

   Template-guided recombination (TGR) is a formal model for the gene descrambling process occurring in certain unicellular organisms (stichotrichous ciliates). The mechanism by which these genes are descrambled is of interest both as a biological process and as a model of natural computation.

   This project studies template guided recombination as an operation on strings and languages with a goal to better understand its computational capabilities. A goal is to implement a TGR simulator with *Grail*-like input/output format that can be used in experiments to study the complexity of the operation. *Grail* is a symbolic computation environment for finite-state machines and language objects.

   Optionally the project may include experiments on the state complexity of the TGR-operation.

3. **Other projects**

I have available some additional topics – please come to see me in my office or send me email. If you have your own idea for a project related to my research, please come to talk with me about it.