Implementation and Application of Automata in String Processing*

Gonzalo Navarro

Department of Computer Science
Univiversity of Chile
gnavarro@dcc.uchile.cl

Automata have been enormously successful in matching different types of complex patterns on sequences, with applications in many areas, from text retrieval to bioinformatics, from multimedia databases to signal processing. In general terms, the process to match a complex pattern is (1) design a NFA that recognizes the pattern; (2) slightly modify it to recognize any string ending with the pattern; (3) convert it into a DFA; (4) feed it with the sequence, signaling the endpoints of a pattern occurrence each time the DFA reaches a final state. Alternatively one can omit step (2) and backtrack with the DFA on the suffix tree of the sequence, which leads to sublinear-time complex pattern matching in many relevant cases. This process, as it is well-known, has a potential problem in stage (3), because the DFA can be of exponential size. Rather than being a theoretical reservation, the problem does arise in a number of real-life situations.

Bit-parallelism is a technique that helps circumvent this problem in many practical cases. It allows carrying out several operations in parallel on the bits of a computer word. By mapping NFA states to bits, bit-parallelism allows one to simulate the NFA behavior efficiently without converting it to deterministic. We show how bit-parallelism can be applied in many problems where the NFA has a regular structure, which allows us simulating it using typical processor instructions on machine words. Moreover, we show that even on general regular expressions, without any particular structure, bit-parallelism allows one to reduce the space requirement of the DFA. In general, the bit-parallel algorithm on the NFA is simpler to program and more space and time efficient than the one based on the DFA.

We show the use of bit-parallelism for exact pattern matching, for allowing optional and repeatable characters, classes of characters and bounded-length gaps, and for general regular expressions. The paradigm is flexible enough to permit combining any of those searches with approximate matching, where the occurrence can be at a limited edit distance to a string of the language denoted by the automaton. We then show applications of these ideas to natural language processing, where the text is seen as a sequence of words, and bit-parallelism allows flexibility in the matching at the word level, for example allowing missing or spurious words.

* Partially funded by the Millennium Institute for Cell Dynamics and Biotechnology (ICDB), Grant ICM P05-001-F, Mideplan, Chile.

S. Maneth (Ed.): CIAA 2009, LNCS 5642, p. 1, 2009.