CMPT 440 – Spring 2019: Quantum Finite Automata

John DeFalco
Due Date: 5/1/2019

Theoretical Background

Quantum finite automata, QFAs, are finite automata similar to DFAs and NFAs, that have been introduced with quantum mechanics to form a method of quantum computing. Quantum computing is more powerful then conventional computing because quantum mechanics allows a wider range of operations to be used, and it always faster than its conventional counterpart as QFAs require less states when compared to other finite automata (Ambainis (2011)).

QFAs are split into two categories, 1-way QFAs and 2-way QFAs. 1-way QFAs are able to recognize all regular languages, while 2-way QFAs are able to recognize all non-regular languages. The definition of a these finite automata is a 6-tuple as follows:

$$M = (Q, \Sigma, \delta, q_0, q_{acc}, q_{rej}) \tag{1}$$

Q is equal to the set of all states in the automata. Σ is the alphabet used within the language. δ is the transition function. q_0 is the starting state for the QFA. q_{acc} is equal to the set of accepting states, and q_{rej} is the set of rejecting states (Nayak (2012)).

Example

An example of the practical application of QFAs is for their efficiency over their conventional counterparts.

QFAs produce outcomes with much higher senses of probability and certainty when compared to other FAs (Gruska et al. (2014)). Meaning that the output, when compared to the output of say DFAs, arrives in such a way that contains proportionally smaller succinctness. This means when using a QFA instead of a DFA, the output will always be smaller and more efficient after going through the QFA.

References

A. Ambainis. Quantum finite automata. pages 9–13, 01 2011.

J. Gruska, D. Qiu, and S. Zheng. Potential of quantum finite automata with exact acceptance. *CoRR*, abs/1404.1689, 2014. URL `http://arxiv.org/abs/1404.1689`.

T. Nayak. Quantum finite automata, quantum pushdown automata quantum turing machine: A study. pages 3765–3769, 03 2012.