The University of York Department of Computer Science

Submitted in part fulfilment for the degree of MEng.

Quantum Algorithm Synthesis Workbench

Sam Ratcliff

3rd December 2010

Supervisor: John A Clark

Number of words = 8832, as counted by `wc -w`.
This includes the body of the report only.

Abstract

TO BE DONE

Contents

1 Introduction

2 Literature Review

2.1 Introduction to Quantum Computation

In 1980, Richard Feynmann noted 'it is impossible to represent the results of quantum mechanics with a classical universal device'[1]. This statement was a seed for interest in the field of Quantum Computation. The true power of quantum computation was not initially realised. With slow progress of research into both their implementation and algorithms, the energy behind the research started to decrease. It would take a discovery by Peter Shor(REFERENCE) to reignite the excitement surrounding the subject.

Quantum Computation is a mechanisim which uses the properties of quantum mechanics to perform computation. The classical computation model is the turing machine. This model is the basis of the Strong Church-Turing thesis defining what is and is not computationally possible.

/////QUOTE

'A (probabilistic) Turing machine can efficiently simulate any physically reasonable computer.' (REF SAM BAUSTEIN LECTURE NOTES)

/////END QUOTE

For this statement to be true it would require Quantum Computation to be efficiently simulated by classical computers. The realisation of Feynman indicates this is not so and implies the statement does not encompas the abilities of Quantum Computers.

In classical computers the computation is performed using the discrete values of 0 and 1. These values are indicated by +5V and 0V signals propergatating round circuits. A signal can only be 0 or 1, there is no in between value. Each signal can indicate the value of a single 'bit' of data. A combination of n bits can be used to represent a number from 0 to $2^n - 1$, an n-bit number. Classical computation works through the manipulation of these n-bit numbers.

The power of quantum computers come from the use of particles in superpositions as described by quantum mechanics. This can be seen as the partical being both a 0 and 1 simultaneously. The partical in this superposition represents a qubit, the quantum computers equivalent

to a classical computers bit. Just as classical computers manipulate bits to perform computation, quantum computers manipulate qubits and their superpositions to perform computation. The power of these superpositions is not obviously apparent.

It is not possible to observe the superposition of a particle. When observed the superposition 'collapses' to either logical 0 or 1. The probability the the superposition collapses to 0 is determined by the superposition's properties.

To write the state of a superposition it is usual to use the 'Ket' notation. A ket is mathematical notation which look as follows, $|a\rangle$. This is used to indicate a state, for example $|0\rangle$ is the state of a logical 0 whereas $|1\rangle$ is the state of logical 1. Using this notation the state of the superposition can be expressed.

As with all probablities, the overall probability of a superposition collapsing to any of the states it contains must equal 1. $\alpha|0\rangle + \beta|1\rangle$ is how the combination of the logical 0 and 1 states can be combined for a single particle. The probability of this state collapsing to 0 can be calculated by $|\alpha|^2$. Similarly, the probability of this state collapsing to 1 can be calculated by $|\beta|^2$. It follows that $\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$ is an equal superposition where the collapse to 0 is just as likely as collapsing to 1. This provides the first glimpse of where a single qubit has the ability to perform a function not currently possible on a classical computer. With n-qubits in the equal superposition, we have n binary values which have an equal probability of taking the value 0 as the value 1. With an ordering decided of these qubits, collapsing the superposition of each qubit will result in a binary value of length n. With all probabilities being $\frac{1}{2}$ this binary value takes a truely random value between 0 and $2^n - 1$. It is not possible to produce a truely random number using a classical computer.

A second indication of the power held within the idea of superposition becomes clear if we look at the n-qubits in their equal superposition. In 1935, Erwin Schrödinger proposed a thought experiment to explain the idea of superposition. Imagine a cat in a fully opaque box with a vile of poison. The vile may break at any time, a truely random variable. After sealing the box the state of the cat is not known. The cat could be alive if the vile has not broken but could just as likely be dead. Only by looking inside the box will the state of the cat be known. Until this time the cat could be thought of as both alive and dead at the same time. If we assign 'dead' to the state $|0\rangle$ and 'alive' to the state $|1\rangle$ the situation looks very similar to the state we have previously seen. Therefore, just

as the cat can be thought of as both dead and alive at the same time, a qubit in the superposition $\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$ can be thought of as both 0 and 1 at the same time. This leads to a very powerful property of quantum computers. With n classical bits, a single number in the range 0 to $2^n - 1$ can be expressed at any one time. With n quantum qubits, every number in the range 0 to $2^n - 1$ can be expressed at any one time. This effectively allows computation over the whole range of 2^n inputs to be carried out in parallel.

This parallism is very powerful and has been shown to enable the computation of problems classified as NP to be performed in polynomial time. This does however have a caveat. As mentioned previously the superposition cannot itself be observed or measured. When observed the superposition collapses with respect to the probabilities of states. This means that even though 2^n calculations can be performed in parallel, only a single answer can be observed.

Currently there are very few quantum algorithms known. The problems that have been the focus of quantum algorithm researchare those which, appear to be highly parallelisable and have only a single answer.

In 1994, Peter Shor astonished the computer science community with a factorisation algorithm to run on a quantum computer. This allowed the factorisation of interger numbers into their consitiuent primes in polynomial time (READ PAPER)

(READ PAPER)

–Problems with current algorithms

–Scalability

–Require many more Qubits than are available with current hardware implementations

–Current State of QC

–Latest hardware

–Latest algorithms

2.2 The Use of Evolutionary Computation in the Synthesis of Quantum Algorithms

2.3 The Focus of this Project

Bibliography

[1] R. Feynman and P. W. Shor, "Simulating physics with computers," *SIAM Journal on Computing*, vol. 26, pp. 1484 – 1509, 1982.