DOWNLOAD PDF CLASSICAL AND QUANTUM COMPUTING WITH C AND JAVA SIMULATIONS

Chapter 1 : Simulation of a Quantum Algorithm on a Classical Computer

Classical and Quantum computing provides a self-contained, systematic and comprehensive introduction to all the subjects and techniques important in scientific computing. The style and presentation are readily accessible to undergraduates and graduates.

Jean-Francois Van Huele, Physics and Astronomy Quantum computation is a recent idea of how to use quantum mechanics to improve computer performance. Although a full quantum computer has not yet been built, a few algorithms have been developed which take advantage of different aspects of quantum mechanics. Two distinct quantum properties allow increased performance: All elementary exist in specific states solutions of quantum differential equations , or in a superposition of those states. A superposition is not an average; quantum particles can simultaneously exist in multiple states including being in two places at once. When the particle is measured, however, the measurement yields only one of those states, and the knowledge of what other states it was in is lost forever. For example, a photon can have either vertical or horizontal polarization, which are distinct quantum states. Most photons are in a superposition of both states, with specific probabilities of being measured in each state. When the photon is passed through a horizontal polarization filter, it will either pass through meaning it is measured to be horizontally polarized or it will not vertically polarized , dependent on the probability of measuring that state. Through physical interaction this photon can be entangled correlated with another so their polarizations must be opposite. In a quantum computer the memory register consists of many quantum particles that can be in one of two states, or a superposition of the two. By labeling one state as 0 and the other as 1, these particles take the place of bits in a classical computer. The different quantum bits, or qubits represent data, similar to a classical computer, but with a twist. While a classical register of three bits can represent any one of the numbers from 0 to 7, three entangled qubits can be in a super-position of any or all the possible values from 0 to 7. Furthermore, an operation performed on the quantum register would operate on each of the possible values simultaneously. This ability to perform exponentially many computations lasts until a measurement is made, so quantum algorithms are designed to raise the probabilities of correct answers, while lowering the probabilities of incorrect answers to zero. Just as with the entangled, polarized photons, the user gets only one of the values out of the computer while the other values are lost. A classical computer can quickly handle all of these calculations, except finding the period of f, which is done using the inefficient classical discrete fourier transform. Although the quantum discrete fourier transform is efficient, any classical simulation cannot be since classical computer bits can neither be entangled, nor represent a superposition of solutions. The quantum discrete fourier transform performs on single, or sets of 2, qubits at a time. Since the entangled qubits are in a superpostion of all possible solutions, the computer only needs to operate on each distinct set of qubits once. To simulate this my program uses an array containing the complex probability amplitude of each possible state represented by a specified number of qubits. Operations are performed by multiplying the array by matrices. The transformation is performed by a series of matrix multiplications which operate on each of the possible states. Given current memory constraints my simulation is presently capable of factoring numbers as high as Indications are that by changing my simulation and using for-loops rather than matrices I can significantly increase this possibility, but will not match classical performance. The real value of the simulation lies in the process of development and its ability to aid in further study of entanglement, rather than in its ability to factor numbers. In programming the simulation I had to understand every aspect of the algorithm, including learning various aspects of the number theory and the nature of the discrete fourier transform. Through this I know exactly where the shortcomings of the classical simulation are, and how quantum computation makes up for these. In the future the simulation will be used to look more closely at the properties of entanglement and how these affect the performance of a quantum computer. As it stands, my simulation follows the steps a quantum computer would to quickly factor a large number and can be used to study other aspects of quantum computation. Discrete logarithms and factoring.

Chapter 2 : Classical and Quantum Computing

Classical and Quantum Computing: with C++ and Java Simulations by Yorick Hardy, Willi H. Steeb This is a self-contained, systematic and comprehensive introduction to all the subjects and techniques important in scientific computing.

Chapter 3 : Software simulation of a quantum computer - Stack Overflow

An algorithm [48, 63, 77, 97,] is a precise description of how to solve a problem. For example algorithms can be used to describe how to add and subtract numbers or to prove theorems.

Chapter 4 : List of QC simulators | Quantiki

Get this from a library! Classical and Quantum Computing: with C++ and Java Simulations. [Yorick Hardy; Willi-Hans Steeb] -- Classical and Quantum computing provides a self-contained, systematic and comprehensive introduction to all the subjects and techniques important in scientific computing.