Complexities of Computation: A Survey Report

Anuj Kumar Jain\(^1\) and Nitin Kumar Jain\(^2\)

\(^1\) Research Scholar, M.Tech (CSE), MMEC Mullana (Ambala) Haryana, India email: jainanuj143@gmail.com
\(^2\) Master Student Department of Computer Science University of Leicester Leicester LE1-7RH United Kingdom email: nkj3@le.ac.uk

**Abstract**- Computation of real numbers has been a challenging task for many years. Because of its unique nature of infinity, it is considered as a very good area of research. This paper tries to convey the nature of the real numbers and the difficulty to compute them i.e. to approximate the value and some respective development processes related to the real numbers. While making a general calculation the approximation can go on and on, this still doesn’t give the exact value. Computer system’s memory is finite. Goal is to approximate the real numbers but the problem arises where to stop and which basis they are subjected for approximation on.

**Keywords:** Real vs. Computable numbers, Turing Computation Model, Quantum Computers

**PACS:** 31.15.B, 03.67.Lx

I. INTRODUCTION

Computation of numbers in computers leads to many memory related problems especially when dealing with real numbers. Even till now problems in finding a better way of using real numbers in computers are not fully solved. Lots of model is purposed for the computation like Turing’s model, Blum Shub Smale’s, Quantum computers which talk about carrying out the computation using the numbers. Talk about a computer which has memory larger than this its size, can process thousands of simultaneous inputs- can that really exist? Let’s have a look on these models.

II. REAL VERSUS COMPUTABLE NUMBER

It is very common to talk about the “computation” in daily life. In real time, speak about balancing a bank account, computing the kilometers per liter of petrol obtained by a motor bike, car or converting one currency to another like British pounds into U.S. dollars. Results generated from each of these operations require processing numbers in a particular way. The method which helps us to process the subjected numbers is formally termed as “algorithm”. Converting this common, however informal concept of taking out the computation into an exact, mathematical framework within which things about the properties of the algorithm can be proved- is the main challenge faced by scientists. System like the Turing machine is strictly interpreted as a model of the computational process–is the example of that computation [1]. Computing with real numbers can be a bit problematic task and by considering its unique property of infinite decimal representation it’s hard to evaluate where to approximate. Since the computer memory is finite, problem arises in the approximation. The main reason behind the invention of real number is to give the exact value and while computation these values cannot be approximated -which may result in producing a wrong result. Computers use floating point numbers to represent the real numbers. Most of the computers use 64 or 128 bit floating point numbers. Thus being finite it can only represent the approximation of real numbers. Floating point numbers are used to represent the real numbers as finite. Though real numbers are represented as finite, it cannot produce the correct output because it cannot be represented precisely. Also considering the factors such as size allocated to floating point numbers, there can be a loss of significant digits during calculations. Why real numbers cannot be represented precisely? Consider a real number 0.1 with base 10. 0.1 In base 16 is 0.199999999... 0.1 in base 2 is 0.00011000110001100011… In the above the number 0.1 is converted to an infinite number on base 16 and base 2. Thus computers using base 16 and base 2 cannot represent the number in base 10 [6], [7]. One of the greatest challenges that had occurred in the computation of real numbers is the Pentium bug. The Pentium bug occurred due to the fact that some of the least significant bit was unknowingly omitted by the processor during the computation of floating point numbers [8]. The earlier version of MS-Excel 2007 returns error when you are calculating some values near to 65,535. MS-Excel has capabilities to store floating point number to a limit of 15 digits. Multiplication of 77.1 with 850 returns 100,000 instead of 65,535. This is identified in still more calculations like 5.1*12850, 10.2*6425, 20.4*3212.5 etc. This error is caused because MS-Excel uses binary form
in order to carry out all the operations. There is no exact binary conversion for \((0.1)\) (periodical Binary fraction). It keeps on repeating and runs out of memory. So the bug rises due to approximation [9].

III. TURING’S COMPUTATION MODEL AND IDEAS

Algorithmic computation is the transformation of finite input in the finite time frame which can be determined by the start of the computation. Turing model has formed computer science. Turing thesis “A Turing machine can do everything that a computer can do”. Turing model requires input in advance, can only use finite time and memory for the computation and behavior constant as it doesn’t depend upon time. “Whenever there is an effective method of obtaining the values of a mathematical function, the function can be computed with the Turing machine”. Turing Church Thesis, A quantity is not computable if it is not computable on turing machine. Nobody else has opposed this theory till yet so we apply the same in computer science. It is applied over computing functions on integers, finite strings. However it doesn’t extend the other computation such as interactive computation. Turing contributed to cryptology and complexity theory by breaking ciphers. Turing’s test for Artificial intelligence “If a computer, on the basis of its written responses to the questions, could not be distinguished from a human respondent – then the computer is thinking and intelligent”. Super Turing computation includes all the computation which isn’t done by Turing machines and is more powerful way of doing computation than Turing machine algorithms. Three formal models of super Turing computation is Persistent Turing machine, \(\pi\)-calculus, \(\Sigma\)-calculus. Turing identified an unsolvable problem i.e. The Halting problem. His O-machine can solve halting problem but non-algorithmically. According to Garzon, Infinite number of discrete neurons can solve the halting problem. Three ways to solve the halting problem in \(\Sigma\)-calculus are interaction, evolution, infinity. Super Turing model extends Turing model to model the problems which are not modelled by Turing model. It requires knowledge of environment to solve the problems like flying airplane, driving home, artificial intelligence. Extending Turing machines to interactive models has solved many computational problems.

IV. QUANTUM COMPUTER AND RAPIDLY ACCELERATING COMPUTER (RAC)

Quantum computer processes data on quantum mechanics like entanglement and superposition. Quantum properties are used to represent the data and to operate on that data in the quantum computers. Factoring of large numbers is being the main challenge faced in the conventional computers which uses bits to compute. According to proposal, Quantum computer uses q-bits formally known as quantum bits to carry out the computation which can represent zero and one or may be quantum superposition of both of them. Quantum computer with \(n\) number of q-bits can have \(2^n\) different states of superposition at the same time while a conventional computer has just one state. Normal quantum system has two states which are spin down and spin up which is represented as binary 0 and 1[5], [14]. A conventional computer can’t process the exponential set of inputs at the same time while quantum computer can use quantum mechanics, computation to manipulate those-Which increases the speed of the computation to an exponential level. Quantum’s error correcting method encodes multiple bits using just one q-bit with combination of states. Factoring and approximations problems are addressed by quantum computer which denies the non-turning machine models. In other words quantum computer follows the turing theory and machine model. To check Turing Church Thesis, Ian Stewart half-jokingly came up with an Idea which he named as Rapidly-Accelerating computer (RAC). “His goal was to show exactly what it is about computing machines that gives rise to things like the unsolvability of the Halting Problem and un-computable numbers”. As assumption, in computation it takes a fixed, finite amount of time to carry out a step. But Stewart said that RAC takes a single second to cram infinite computational steps. According to him, RAC can calculate the incalculable. RAC can compute the un-computable and decide the un-decidable [2]. This is impossible in real as like Turing’s infinite memory machine. Computers have circuits and on logical gates, circuits we have Theoretical upper bound to the rate of information transfer (i.e. velocities). According to Einstein theorem “No material object can exceed the velocity of light” [1]. So we have to follow Turing’s limit. There are no devices to compute the un-computable and hence we call it computation of non-computable.

V. ACKNOWLEDGMENT

This article is inspired by the interest in real numbers and floating point arithmetic. Its aim is to understand the challenges faced while doing the computation. People generally face rounding off and approximation problems while doing any computing. It helps to understand why these errors happen. We’ll specially thank to our parents to motivate us and many other colleagues who helped us in presenting this paper.

REFERENCES


