Realization of N-state variables

Chandra Shekhar Kumar*

August 29, 2003

Abstract

In this paper, the process for realization of the mathematical representation of any variable having N > 2 states in the computing domain has been extended. This might lead to solve computationally intensive problems, mainly searching & sorting algorithms in more elegant way and fast.

1 Introduction

1.0.1 History

The architecture of today's computing machines, what we call computers, are based on the general concept of Turing Machines, the concept being introduced by Alan Turing. The generic concept behind this was Boolean Algebra, conceived by George Boole. Turing machines are the outcome of the mapping of the variables having 2-states, either 0 or 1, popularly known as bits. Quantum computers introduced the notion of qubits, which stands for mapping of the variables having 3-states. Several notions of representations has been introduced since then, but most of them still are based upon vector-arithmetic, pointer-arithmetic, integer-arithmetic etc. The main cause of concern with all of them is being sequential in nature. So the very term, parallel computing or multi-tasking are still sequential in nature because the execution of atomic operations are intrinsically sequential.

1.0.2 N-state concept

The idea was originally proposed by George Boole. The basic concept is: a variable having N-states, where N is any natural number. Fuzzy logic has tried to simulate this by introducing probabilistic measure with each state. But the very representation of this probabilistic measure in this scenario has been under dispute because this is based on linear allocation of [0..1] to each state for a variable at a given time. So again , this leads to execution of the atomic operations being sequential in nature.

*email: cskumar9@rediffmail.com

1.0.3 Extension of the representation

In Quantum physics, the object is in a state, which is a superposition of many states. The famous example is: Schrodinger's Cat Paradox. The cat is sealed inside a box. Then a poisonous capsule is injected inside the box. But we still donot know whether the cat is alive or dead. There is no way to know the state of the cat until we break the box. Breaking the box is equivalent to breaking the superpostion of the states, and only after that we are in a position to say whether the cat is alive or dead.
The same concept could be extended to computers. The box above is same as computer. Anything withing that is in a superposition of many states, call it N. So the processes, be it atomic ones, can have N-states at any time. The only issue here is to know the state from outside this box, computer. But there is a trick, one level of indirection is required to achieve the same. For the sake of simplicity of the representation, we can assume that the box, computer, is reflexive in nature, i.e. , it can know about itself. So here, we are not breaking the superposition. So, the computer is responsible for revealing the many states of any process running underneath.

1.1 Remarks and Conclusion

Now the onus is on the shoulders of the theoretical computer scientist with mathematical bent to come with the logical mapping in such a way so that the present representation is a special case of this more general one. For the sake of the simplicity, the mathematical rigorous treatment has been avoided in this paper. For details, the authors could be contacted at the given email-address.

References

An investigation into the Laws of Thought, on Which are founded the Mathematical Theories of Logic and Probabilities, George Boole, 1954.