

STUDY IN SOLUTION OF INTRACTABLE PROBLEMS THROUGH SOFT COMPUTING

KUMAR SHEKHAR SINGH

Asst. Professor's Department of Computer Science & Applications, Gaya College, Gaya, Bihar

Abstract - In theoretical computer science, solution of computational problems require suitable algorithms. In this context an important question arises how & when an algorithm provides a satisfactory solution.

The efficiency of an algorithm is measured on the basis of time complexity & space complexity. Worst case complexity of an algorithm is the largest no. of machine cycles or operations needed to solve a given problem.

A problem that is solvable using an algorithm with polynomial worst case complexity is called TRACTABLE i.e. solution can be found in polynomial time.

The situation is much worse for a problem that cannot be solved using algorithm with worst case polynomial complexity. Such a problem will be called INTRACTABLE.

Many problems in practically diverse fields such as Industries Humanities, Inventory & supply chain Management, Medicine etc. instead of looking, exact and precise solution of a problems, approximation is sought for i.e. agreed on approximate solutions.

Soft computing refers to a variety computational techniques which are applied in situations where imprecision, approximation uncertainly and partial truth is tolerable.

The present paper relates to some of the important soft computing methods being implemented, towards solution of intractable problems.

INTRODUCTION

The present paper is divided into three sections and subsequent subsections.

The **first section** relates to intractability of computational problems.

The **second section** is related to scope and implementations of three soft computing methods.

- a) Artificial Neural networks
- b) Genetic Algorithm
- c) Fuzzy Logic

The **third and final section** presents the conclusion of the study.

SECTION 1 - COMPUTABILITY

The time complexity algorithm can be expressed in no. of operations used by an algorithm when the input size is fixed.

An algorithm has polynomial complexity if it has time complexity $O(n^b)$

Where b an integer, $b \geq 1$

An algorithm has exponential complexity if it has time complexity $O(b^n)$

Where $b > 1$

An algorithm has factorial complexity if it has time complexity

$O(n!)$

1.1 NP & NP complete problems

Solvable problems are believed to have the property that algorithm with polynomial worst case complexity solves them but solution if know can be checked/verified in polynomial time. Such problems are tractable and said to be of class non-deterministic

polynomial (NP). There is also an important class of problems with property that if any of these problems can be solved by a polynomial worst case time algorithm then all the problems in the class can be solved by polynomial worst case time algorithms. Such class of problems is NP complete class.

1.2 Complexity & Big - O

The big - O estimate of time complexity of an algorithm expresses how the time required to solve can change as input grows in size. In practice the best estimate that can be used is shown.

However the big - O estimates of time complexity cannot be directly translated into machine time used. Reason for that is big O estimate function (fn) is $O(g(n))$ in where $f(n)$ is complexity and $g(n)$ is reference function such that

$$C_1 g(n) \leq f(n) \leq C_2 g(n)$$

C_1 & C_2 are constants

So without knowing constants C_1 & C_2 in the equality this estimate cannot be used to determine a lower bound & upper bound in the no. of operations used in worst case.

1.3 Technological Improvements

It is important to know that how long a computer will be needed for solve a problem for example if an algorithm requires 10 hours, it may be worthwhile to spend the complete time. But if an algorithm requires 10 billion years to solve a problem it would be unreasonable to use the resources.

One of the important phenomenon of modern technology is the tremendous increase in speed & memory space of computers. Parallel processing also

increases the processing time in which sequences are processed in parallel.

SECTION 2: SOFT COMPUTING

2.1 Introduction

The term soft computing was introduced by Professor Lotfi Zadeh [1] (University of California, Berkeley) with objective of exploiting the tolerance for imprecision uncertainty and partial truth, intractability, Robustness, solutions cost and better rapport with reality. The ultimate goal is to emulate the human (Bio) mind as closely as possible to solve complex problems. It includes diverse fields such as Neural Networks, Genetic algorithm & Fuzzy logic. An important thing about the constituents of soft computing is that they are complementary to conventional computational techniques offering their own advantages to solve Similar to otherwise intractable/Unsolvable problems.

2.2 Artificial Neural Networks (ANNs) & Intractability

ANNs are computational approach modeled on the basis of interconnection of human neurons for optimal solutions.

The earliest working in this area was Mc. Culloch & Pitts [2] in 1943 was called threshold Logic, Donald Hebb [3] created the law of Synaptic strengthening a learning mechanism called unsupervised learning. Frank Rosenblatt [4] created. Perceptron a pattern matching algorithm based on Linear Separability.

ANN's can be used in certain intractable domain such as pattern matching & classification, optimization, approximation etc. These tasks are very difficult for traditional algorithms which include precise computational operations.

The main property of ANN is to its capability to learn or training. To training means that ANN adopts itself to a stimulus by making proper parameter adjustments resulting in desired results.

2.3 Genetic Algorithm (GAs) & Intractability

In 1975 Holland (5) put forward the idea of GAs and laid the foundation for further studies & research. In soft computing GAs are adaptive random search algorithms based on evolutionary ideas of natural selection and genetics. GA exploit the idea of Random search for solution of complex problems. From the solution point of view - GAs use data to direct the search into region where probability of finding a solution is better. Robustness of GAs comes from the fact that they do not break or degenerate even input are varied due to some noise GAs search for best solution among member of feasible solution is among various fitness (solution) points a single suitable point is selected.

GA's offer great benefits in searching large state space, multimodal search space or even n dimensional search space.

2.4 Fuzzy Logic (FL) & intractability

The concept of fuzzy logics was conceived by Lotfi Zadeh. It is an organized method for dealing with imprecise data. The data are called Fuzzy sets. It is a way of processing data by allowing partial set members rather than crisp membership/non-membership.

Fuzzy logic reasons that for processing there is not always needed precise & numerical information inputs, processing can be performed even with imprecise inputs. Suitable feed back may be designed to accept noisy in precise input and they would be much more effective and perhaps easier to implement.

Fuzzy logic is basically a multivalued logic that allows intermediate values to be defined between conventional evaluation such as Y/N, T/F, B/W etc. In this way a attempt is made to apply a more human like way of thinking in programming of computers. From solution point of view it is a problem solving control system methodology that lends itself implementation in systems ranging from simple small embedded microcontrollers to large. networked multichannel PC or workstation based data acquisition and control systems. It can be implemented in HW, SW or combination of both. It provides a simple way to arrive to definite conclusion upon. vague, ambiguous, imprecise, noisy or missing input.

SECTION 3: CONCLUSION

Soft computing techniques are based on biological systems. Purpose it to traverse search space of possible answer to a problem.

Soft computing provides an alternate approach of problem solution. Efficient algorithms & technological/hardware implementation contribute to the solution of complex problems. However some class of problems in which imprecision, partial truth can also be considered as solution, are candidates of consideration. Soft computing is an alternative approach this domain.

REFERENCES

- [1] Zadeh, LA (1965) 'Fuzzy Sets : Information & Control', University of California, Berkeley.
- [2] Mc. Culloch & Pitts (1943) 'A logical calculus of the ideals immanent in neurons activity' *Buttelin of Mathematical Bio physics*.
- [3] Hebb, Donald (1947) 'The organization of Behaviours'
- [4] Rosenblatt, Frank (1957) 'Perceptron Algorithm'
- [5] Holland, John (1957) 'Adaption in Natural & Artificial System' MIT Press.