International Journal of Enhanced Research in Science Technology & Engineering, ISSN: 2319-7463
Vol. 2 Issue 12, December -2013, pp: (63-66), Available online at: www.erpublications.com

Solution of technical problems using Memetic Algorithm

Anu[1], Anil Saini[2]
[1]Mtech. Student, Department of Computer Science & Applications, K. U. Kurukshetra, INDIA
[2]Assistant Professor, Department of Computer Science & Applications, K. U. Kurukshetra, INDIA

Abstract: Evolutionary algorithms search on the basis of Darwinian evolution for solving simple problems. For complex problems local search is incorporated in evolutionary algorithm named Memetic algorithm, which greatly improve the efficiency of search. In this paper, we discuss the choice of the algorithm used for solving different problems based on several issues. Problems in the technical field sometimes need to reach a good solution in time instead of reaching optimal solution. For some undetermined problem evolutionary algorithm can be useful.

Keywords: evolutionary algorithm, Genetic algorithm, Memetic algorithm.

I. Introduction

Evolutionary algorithms(EA) are a class of a search and optimization techniques used for solving different problems in many engineering disciplines beyond computer science. EAs use the benefits of the biological evolution (natural selection) to find the optimal solution. Set of potential solution candidates, which undergo a simulated evolutionary process, served as the basic element called chromosome. The evolution of chromosomes due to the action of crossover, mutation and natural selection are simulated through computer code. Memetic algorithms (MAs) are extensions of evolutionary algorithms that apply separate processes to refine individuals, for example improving their fitness by hill climbing. Advantages of efficient heuristics incorporating domain knowledge and population based search are used to enhance solution. In the concept of heuristic optimization a meme is taken to represent a learning or development strategy. Memetic algorithms mimic the cultural revolution, memes are modified during an individual's life time but not genes value. The method is based on a population of agents and successfully find the approximate solution of NP optimization problems, operational research and optimization, automatic programming, machine and robot learning. Evolutionary algorithms are also applied to study and optimize of models of economics, ecologies, immune system, social system and the interaction between evolution and learning.

Genetic algorithms (GA) provide a way to employ evolution based on variation and selection. Evolve a population (multiset) of candidate solutions using the concepts of survival of the fitness, variation and inheritance. In this paper Memetic algorithm can be defined as genetic algorithm that include non-genetic local search to improve genotypes. Memetic algorithm can blend the functioning of genetic algorithm with several heuristic search techniques like simulated annealing, tabu search. Some of the principle issues are

- Is the problem suitable for EAs
- Choice of algorithm type

At this point it has to be stated clearly, that EAs are not generic problem solvers. If the optimization problem can be solved by a specialized method, than in general with this method better results can be achieved. Therefore evolutionary algorithms should only be used in cases where no specialized algorithms exist.

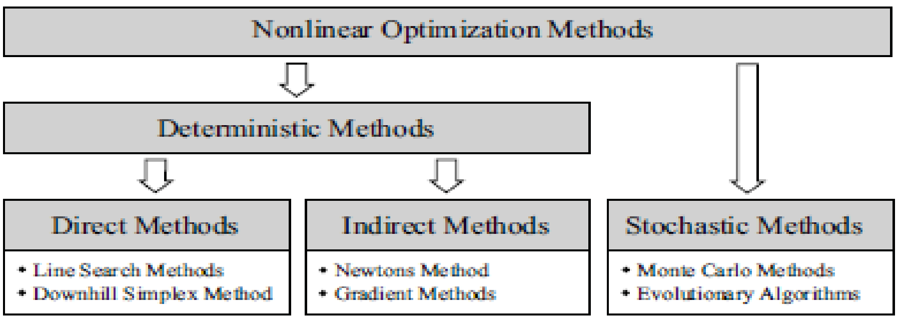

Figure 1: Nonlinear Optimization Method

II. Optimization problems

Optimization is a mathematical discipline that concerns the finding of minima and maxima of functions, subject to constraints. Optimization comprises a wide variety of techniques from Operations Research, artificial intelligence and is used to improve business processes in practically all industries. Minimize f(x): Where f(x) is the objective function and x is a real variable and S is feasible solution space. The purpose of an optimization algorithm is to find a solution x, for which the function f(x) is minimum.

$$F(x) = \min \{f(x) \mid x \in S\} \quad [1]$$

There are two different types of optimal points are:

- Local Optimal point if no point in the neighbourhood has a function value smaller than $f(x^*)$.
- A point or solution x^{**} is said to be a global optimal point, if no point in the entire search space has a function value smaller than $f(x^{**})$.

The constraints are some functional relationships among the design variables and other design parameters satisfying certain physical phenomenon and certain resource limitations [2].

The optimization algorithms are classified into a number of groups:

I. Single-variable optimization algorithms are classified into two categories :

i. Direct methods
ii. Gradient based methods

Direct methods do not use any derivative information of the objective function; only objective function values are used to guide the search process. However, gradient-based methods use derivative information (first and/or second order) to guide the search process. Although engineering optimization problems are multi-variable, single-variable optimization algorithms are mainly used as unidirectional search methods in multivariable optimization algorithms.

II. Multi-variable optimization algorithms demonstrate how the search for the optimum point progresses in multiple dimensions. Depending on whether the gradient information is used or not used, these algorithms are also classified into direct and gradient-based techniques.

III. Continuous or discrete based on the optimization parameters having real or discrete value.

IV. Specialized optimization algorithms: Two of these algorithms-integer programming and geometric programming-are often used in engineering design problems. Integer programming methods can solve optimization problems with integer design variables. Geometric programming methods solve optimization problems with objective functions and constraints written in a special form.

The general structure of an iterative optimization method:

1) Initialization: During the initialization process the optimization variables xj, $j = 1, \ldots, n$ get an adequate initial value.

2) Iteration code: By the iteration rules a sequence xi, $i = 1, 2, \ldots, n$ of the variable vector is generated, which cause an improvement in respect to the objective function in each iteration step i.

3) Abort criteria: The abort criteria exist in form of an in equation which is used to decide about the termination or further iteration steps.

The different methods of nonlinear optimization algorithms consists of two classes deterministic and stochastic methods. Furthermore deterministic methods are subdivided into direct and indirect methods. While direct methods use heuristics to determine the search direction, whereas indirect methods work on the basis of partial derivatives of the objective function and therefore require differentiability of the objective function. Stochastic methods are based on the usage of random numbers and probabilities. The benefit of these methods is, that they don’t demand an objective function which is continuous and differentiable. In figure 1 the most important nonlinear optimization methods are shown.

III. Evolutionary algorithms

Evolutionary algorithms as a general term for evolutionary programming, evolutionary strategies, and genetic algorithm have been applied in various domains of search and optimization. In 1990, genetic programming was established based on the work of Holland and Dejong [3], [4]. These algorithms mainly uses recombination operator instead of mutation operator. It is similar to biological evolution where mutations also appear with minor probability.

For optimization problems in the technical physical field Evolutionary strategies were developed by RECHENBERG and SCHWEFEL [5], [6]. Optimization process mainly uses mutation, while the recombination plays a minor role. The evolutionary programming is based on the work FOGEL, OWENS and WALSH [7]. Evolutionary programming uses only the mutation operator as biological variation. Genetic programming was developed in combination with genetic algorithms mainly by KOZA [8]. In this method the recombination is the primary and the mutation the secondary operator, just like in genetic algorithms. The name memetic algorithm leads back to the concept of meme developed by DAWKINS [9].

HYBRID Genetic Algorithm also creates an initial population randomly and searches the fitness landscape. Recombination and mutation will usually produce solutions that are outside this space of local optima (and can thus be regarded as "damaged") but a local optimiser can then "repair" such solutions to produce final children that lie within this subspace, yielding a memetic algorithm[10],[11]. Both the genetic structure of an individual and its associated fitness value are modified to reflect the changes in phenotype structure as a result of performing local search. The local search process drags solutions in the direction of local optima. These improvements go on accumulating over all generation resulting in a larger improvement in the total performance [12], [13]. After the local refinement, the best result is taken back in main algorithm. It strengthens the algorithm for convergence of both uni model and multi model function.Memetic Algorithms can fulfill these criteria better than traditional Evolutionary Algorithms. However the simple inclusion of a given local search method is not enough to increase the competence of the underlying EA[14].

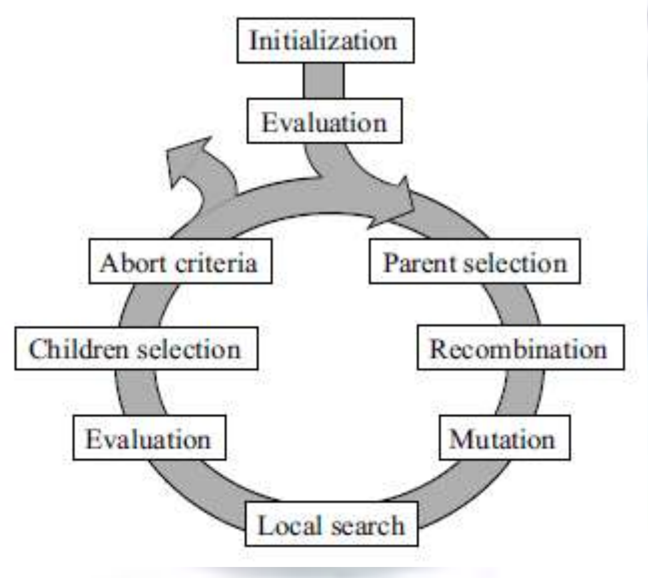

Figure 2: Evolutionary Cycle of memetic Algorithm

Some general conclusion can be drawn to help threaded to adapt an EA for a concrete optimization problem:

1. For continuous optimization genetic algorithms, evolutionary strategies or memetic algorithms should be used. Otherwise in case of discrete optimization evolutionary or genetic programming should be used.

2. In case of continuous optimization (population size: 20) a memetic algorithm with using tournament selection and simulated annealing. It could be a useful start configuration for problem adaptation.

3. The population size should be as low as possible and as high as necessary. With upper population size a better solution and a faster speed of convergence is expected but more computing time is required (linear increase).

4. The choice of the mutation operator should be considered very carefully. Dependent on the optimization problem the mutation operator can have different shapes. At the beginning of the optimization a large step size is profitable to make large steps through the search space. At the end of the optimization a small mutation step size is better.

Conclusion

General EAs can be used for many problems, but they might not always be the best choice. The adaptation of an EA to the concrete problem can cost a lots of time. In objective function continuous and differentiable derivative based methods such as the Newton method or the Gradient method might provide a faster solution, if the objective function does not fulfill this requirements or a complex objection function is given, the EAs may reach good optimization results according to a global optimization.

References

[1]. K. Weicker, 2002, "Evolution¨are Algorithmen", 1st ed. Teubner Verlag.
[2]. Aaquil Bunglowala, Dr. B. M. Singhi, 2008, "A Solution to combinatorial Optimization Problem using Memetic Algorithms", International Journal Of Computer Science and Applications Vol. 1, No. 3, ISSN 0974-1003.
[3]. Holland J., 1975, "Adaptation in Natural and artificial Systems", the university of Michigan.
[4]. K. A. D. jong,1975, "an analysis of the behaviour of a class of genetic adaptive systems", Ph. D dissertation. University of Michigan.
[5]. I. Rechenberg, Evolutions strategie. Optimierung technischer Systemenach Prinzipien der biologischen Evolution, 1st ed. Frommann Holzboog, 1973.
[6]. H.-P. Schwefel, "Evolutionsstrategie und numerische optimierung, "Ph.D. dissertation, Technische Universitt Berlin, 1975.
[7]. D. B. Fogel, "Evolving artifical intelligence," Ph.D. dissertation, University of California, San Diego, 1992.
[8]. J. R. Koza, Genetic programming: A paradigm for genetically breeding populations of computer programs to solve problems. Stanford University Computer Science Department, 1990.
[9]. R. Dawkins, The selfish gene, new ed. Oxford New York: Oxford University Press, 1989.
[10]. Natalio Krasnogor and Jim Smith,2005 "A Tutorial for Competent Memetic Algorithms:"Model, Taxonomy and Design Issues", ISSN : 1089-778X .IEEE
[11]. J. Digalakis and K. Margaritis, 2004, "Performance Comparison of Memetic Algorithms" Appl. Math. Comput 158.
[12]. T. Elmihoub, A. A. Hopgood, L. Nolle and A. Battersby, "Performance of Hybrid Genetic Algorithms Incorporating Local Search", 2004.
[13]. Nicholas J. Radcliffe and Patrick D. Surry, "Formal Memetic Algorithms", in "Evolutionary Computing: AISB Workshop", Ed: T.C. Fogarty, Springer-Verlag LNCS 865, pp1-16, 1994.
[14]. Kurt A. Hacker, John Eddy and Kemper E. Lewis, 2002, "Efficient Global optimization using Hybrid Genetic Algorithms", American Institute of Aeronautics and Astronautics, Inc.
[15]. Tarek A. El-Mihoub, Adrian A. Hopgood, Lars Nolle, Alan Battersby, "Hybrid Genetic Algorithms: A Review", Engineering Letters, 13,ISSN: 1816-093X, 2006.