ISSN:0975-9646

Shweta Gupta et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (3) , 2014, 4313-4316

Greedy Estimation of Distributed Algorithm to Solve Bounded knapsack Problem

Shweta Gupta[#1], Devesh Batra[#2], Pragya Verma[#3]

[#1]*Indian Institute of Technology (IIT)*
Roorkee, India
[#2]*Stanford University*
CA-94305, USA
[#3]*IEEE, USA*

Abstract— This paper develops a new approach to find solution to the Bounded Knapsack problem (BKP).The Knapsack problem is a combinatorial optimization problem where the aim is to maximize the profits of objects in a knapsack without exceeding its capacity. BKP is a generalization of 0/1 knapsack problem in which multiple instances of distinct items but a single knapsack is taken. Real life applications of this problem include cryptography, finance, etc. This paper proposes an estimation of distribution algorithm (EDA) using greedy operator approach to find solution to the BKP. EDA is stochastic optimization technique that explores the space of potential solutions by modelling probabilistic models of promising candidate solutions.

Index Terms— ***Bounded Knapsack, Greedy Algorithm, Estimation of Distribution Algorithm, Combinatorial Problem, Optimization***

1. INTRODUCTION

The Knapsack Problem is an example of a combinatorial optimization problem, which seeks for a best solution from among many other solutions. It is concerned with a knapsack that has positive integer volume (or capacity). It is NP-hard [9, p.65].Researchers have worked on different approaches to solve this combinatorial problem like dynamic programming, brute force, branch and bound, memory functions, genetic algorithms. Though genetic algorithms are found to be the quickest among all, but EDAs are proved to be giving even faster results than GA. These methods are not suitable to solve the practical problems because these techniques fail to solve larger problems within a practical limited time. It is also a problem which offers *many* practical applications in computer science, operations research, and management science.

Bounded Knapsack Problem (BKP) is concerned with a knapsack that has positive integer volume (or capacity) C. There are n distinct items that may potentially be placed in the knapsack. Item i has a positive integer weight wi and positive integer profit pi. In addition, there are Qi copies of item i available, where quantity Qi is a positive integer .Let Yi represents number of copies of item i which are to be placed into the knapsack. The objective is to choose a subset of given items so as to maximize the corresponding profit sum without exceeding the capacity of the knapsack.

EDAs belong to the class of evolutionary algorithms. Estimation of distribution algorithms (EDAs), sometimes called probabilistic model-building genetic algorithms, are stochastic optimization methods that guide the search for the optimum by building probabilistic models of favourable candidate solutions. The main difference between EDAs and evolutionary algorithms is that evolutionary algorithms produces new candidate solutions using an implicit distribution defined by variation operators like crossover, mutation, whereas EDAs use an explicit probability distribution model such as Bayesian network, a multivariate normal distribution etc. EDAs can be used to solve optimization problems like other conventional evolutionary algorithms.

The rest of the paper is organized as follows: next section deals with the related work followed by the section III which contains theoretical procedure of EDA. Section IV discusses proposed greedy estimation of distribution algorithm to solve BKP. Section V summarizes the results obtained from the proposed algorithm and the comparison with the results obtained from the genetic algorithm. Finally, section VI is composed of concluding remarks.

2. RELATED WORK

In [1] a greedy genetic algorithm is proposed to solve the bounded knapsack problem. Experimental results are used to prove the validity and feasibility of the algorithm.

[2] Presents a comparative study of brute force, dynamic programming, memory functions, branch and bound, greedy, and genetic algorithms in terms of memory and time requirements. On the basis of experimental results it is observed that genetic algorithm and dynamic programming are the most promising approaches.

In [3] a hybrid estimation of distribution algorithm (MOHEDA) for solving the multiobjective 0/1 knapsack problem (MOKP) is presented. Further, Local search based on weighted sum method is proposed, and random repair method (RRM) is used to handle the constraints. With the help of experimental results it is observed that MOHEDA outperforms several other state-of-the-art algorithms.

In [4] a hybrid genetic algorithm based in local search is described. Local optimisation is not explicitly performed but it is embedded in the exploration of a search metaspace. Upon compared with other GA-based approaches and an exact technique (a branch & bound algorithm), this algorithm exhibits a better overall performance in both cases.

3. ESTIMATION OF DISTRIBUTION ALGORITHM(EDA)

Estimation of distribution algorithms are relatively new branch to the evolutionary algorithms. They are stochastic optimization techniques that explore the space of potential solutions by modelling explicit probabilistic models of promising candidate solutions. Complex problems with larger domain have been solved using EDA because of its model-based approach to optimization. EDAs typically work with a population of candidate solutions to the problem, starting with the population generated according to the uniform distribution over all feasible solutions. *Fitness* function is computed to rank the population with the higher the value of fitness function the better the ranking. From this ranked population, a subset of the most favorable solutions is selected by the selection operator. Different selection operator can be used namely truncation selection with threshold τ=50%, which selects the 50% best solutions. The algorithm then proceeds by constructing a probabilistic model which estimates the probability distribution of the selected solutions. Following the model construction, new solutions are generated by sampling the distribution encoded by this model. The process is repeated until some termination criteria is met which can be either a solution of sufficient quality is reached or the number of generations fixed initially reaches.

EDA pseudo code

- Step 1: Generate an initial population P_0 of M individuals uniformly at random in the search space
- Step 2: Repeat steps 3-5 for generations l=1, 2, … until some stopping criteria met
- Step 3: Select N<=M individuals from P_{l-1} according to a selection method
- Step 4: Estimate the probability distribution $p_l(x)$ of an individual being among the selected individuals
- Step 5: Sample M individuals (the new population) from $p_l(x)$

The distinguishing step of the EDAs from many other metaheuristics is the construction of the model that attempts to build the probability distribution of the promising solutions. This is not a trivial task as the goal is not to perfectly represent the population of promising solutions, but instead to represent a more general distribution that captures the features of the selected solutions that make these solutions better than other candidate solutions. In addition, we have to ensure that the model can be built and sampled in an efficient manner.

4. GREEDY ESTIMATION OF DISTRIBUTION ALGORITHM(EDA)

The flowchart of the EDA using greedy operator is shown in the figure 1. Greedy operator ensures the feasibility of the solutions. Thus, hybrid EDA is used to find solution to the BKP.

Each chromosome is associated with a weight, profit and the number of copies

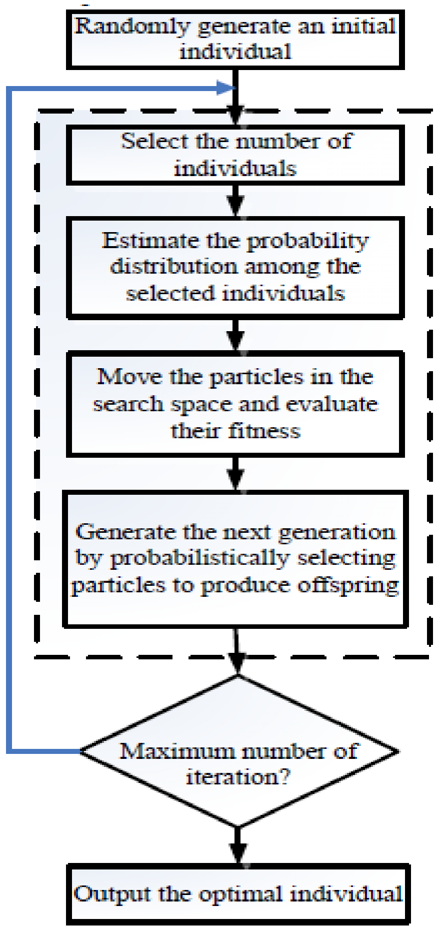

Figure 1: Flowchart for Estimation of Distributed Algorithm

4.1. Initialization of population

Binary bit pattern string is used to solve using the EDA. The size of each chromosome is same as the number of different items available i.e. n. Each chromosome is represented as

$$x=(x_1,x_2\ldots\ldots x_n) \text{ where } x_i=0 \text{ or } 1$$

x_i value represent whether item is selected to be included in knapsack with xi=1 represents the inclusion of the i^{th} item in the knapsack. Population is initialized randomly. Population size is dependent upon solution space of the problem statement. It is kept as 150 in this experiment.

4.2 Calculation of Fitness Function

Fitness function is calculated to score each chromosome. It is assumed that the all the available c_k copies of the k^{th} item with value 1 have been included in the knapsack. Total weight (W) and profit (P) is calculated for each chromosome using equation (4.1) and (4.2).

W =wkxkck (4.1)

P= pkxkck (4.2)

Where w_k is the weight of the single copy of the k^{th} item, c_k is the maximum no. of available of copies of the k^{th} item and p_k is the profit of single copy of the the k^{th} item. Maximum capacity of the knapsack (C_{max}) decided the feasibility of the candidate solutions. With this aspect, chromosomes satisfying the constraint ($W_{\leq} C_{max)}$ are said to be feasible solutions. Greedy operator transforms unfeasible solutions to the feasible solutions by excluding some of the items from the knapsack in a manner to maximize the fitness function F. Therefore, it is not necessary that all the available maximum copies of the items must be included into the knapsack.

4.3 Greedy Transform

Below is the algorithm for the greedy transform:
Initially generated binary string (x) as candidate solution, profit item (p), weight item (w) and maximum no. of copies available(c) are input to the algorithm and it outputs the feasible solutions by employing greedy transform. Profit of the items in each chromosome i.e. pk= (p1,p2….pn) is sorted into the descending order. Correspondingly, weight and no. of maximum copies available of the items are arranged with respect to the new sorted profit array.

-For all items, xi=1
-set C_{total}=0, item copies= [k_1,k_2…..k_n],k_i=0
-for i=1 to n

- If Ct_{otal} +$w_i \leq C_{max}$, then
- For j=1 to ci
  if Ct_{otal} +$w_i \leq C_{max}$, then
  set $Ct_{otal=}$ Ct_{otal} + wi
  ki=ki+1

- Else
  - xi=0

4.4 Selection Operator

There are different selection strategies used namely Roulette wheel, breeding pool selection, Boltzmann selection etc[1].In this experiment, truncation selection with threshold $\tau = 50\%$, which selects the 50% best solutions is used.

4.5 Modeling and Sampling

EDA replaces use of operator crossover with learning and sampling probabilistic model. Model: a probability vector l= (l1,...,ln) where li = probability of 1 in position i.Learning and sampling of probabilistic model consists of two steps:
1. Learn p: compute the proportion of 1 in each position
2. Sample p: Sample 1 in position i with probability pi.
Thus, after sampling based on the probabilistic model, new generation is formed. These steps make the performance of EDAs faster and more efficient than Genetic Algorithm**.**

4.6 MUTATION

Mutation is the process of inversion of bit from 0 to 1 or 1 to 0. It helps to overcome the problem of trapping the solution to local optima. Thus, it helps in enhancing the fitness function and provided dynamism to the algorithm In mutation process a sequence of chromosome size of random numbers (values between 0 and 1) is generated for each chromosome. The parity of the bit in chromosome, having value less than the mutation probability (mp), is inverted. As an example,

Parent1: **[10101011]**
If random sequence having 3rd and 5th value greater than the mutation probability ,then would be
Children: **[10000011]**

5. EXPERIMENTAL RESULTS

Experimental results were performed on Intel ® Core ™ i5-2430M CPU @2.40 GHz and 4.00 GB RAM. MATLAB 7.6.0 (R2008a) was used for simulation. It is observed that the value of knapsack increases with the increase in number of iterations as shown in figure 2.

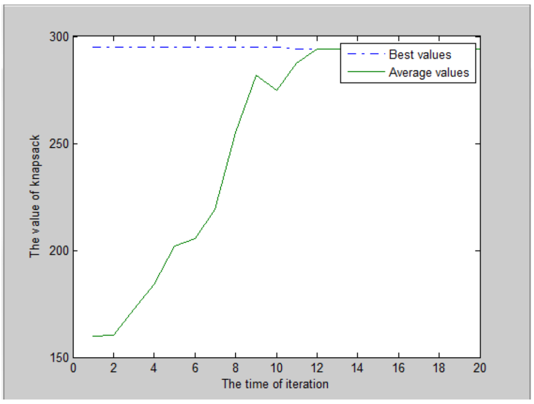

Figure 2: Iterative process for best and average values

Further, it was also observed that as the number of iterations increases, the average value approaches the best value and after certain number of iterations the two coincide.

6. CONCLUSION

In this paper, estimation of distribution algorithm using greedy operator approach was used to solve bounded knapsack problem (BKP).Greedy operator enhanced the feasibility and searching ability of the algorithm. Learning and sampling of the probabilistic model was used instead of crossover operator of Genetic algorithm. Results obtained from EDA are found to be more efficient and faster than the results obtained from existing algorithms.

REFERENCES

[1] Kaystha, Sarsij, and Suneeta Agarwal. "Greedy genetic algorithm to Bounded Knapsack Problem." Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on. Vol. 6. IEEE, 2010.
[2] Hristakeva, Maya, and Dipti Shrestha. "Different Approaches to Solve the 0/1 Knapsack Problem." Retrieved November 3 (2004): 2012.
[3] Li, Hui, et al. "Hybrid estimation of distribution algorithm for multiobjective knapsack problem." Evolutionary Computation in Combinatorial Optimization. Springer Berlin Heidelberg, 2004. 145-154.
[4] Cotta, Carlos, and José M. Troya. "A hybrid genetic algorithm for the 0–1 multiple knapsack problem." Artificial neural nets and genetic algorithms. Springer Vienna, 1998.
[5] Julstrom, Bryant A. "Greedy, genetic, and greedy genetic algorithms for the quadratic knapsack problem." Proceedings of the 2005 conference on Genetic and evolutionary computation. ACM, 2005.

[6] Khuri, Sami, Thomas Bäck, and Jörg Heitkötter. "The zero/one multiple knapsack problem and genetic algorithms." Proceedings of the 1994 ACM symposium on Applied computing. ACM, 1994.

[7] Fukunaga, Alex S. "A new grouping genetic algorithm for the multiple knapsack problem." Evolutionary Computation, 2008. CEC 2008.(IEEE World Congress on Computational Intelligence). IEEE Congress on. IEEE, 2008.

[8] Miller, Brad L., and Michael J. Shaw. "Genetic algorithms with dynamic niche sharing for multimodal function optimization." Evolutionary Computation, 1996., Proceedings of IEEE International Conference on. IEEE, 1996.

[9] Winston, Wayne L., and Jeffrey B. Goldberg. "Operations research: applications and algorithms." (1994).

[10] Knowles, Joshua D., and David W. Corne. "M-PAES: A memetic algorithm for multiobjective optimization." Evolutionary Computation, 2000. Proceedings of the 2000 Congress on. Vol. 1. IEEE, 2000.

[11] Zhang, Qingfu, et al. "Hybrid estimation of distribution algorithm for global optimization." Engineering computations 21.1 (2004): 91-107.

[12] Ishibuchi, Hisao, and Tadahiko Murata. "Multi-objective genetic local search algorithm." Evolutionary Computation, 1996., Proceedings of IEEE International Conference on. IEEE, 1996.

[13] Saeys, Yvan, et al. "Fast feature selection using a simple estimation of distribution algorithm: a case study on splice site prediction." Bioinformatics19.suppl 2 (2003): ii179-ii188

.