National Multidisciplinary Conference on Emerging Trends, Opportunities and Challenges in Higher Education
International Journal of Scientific Research in Science and Technology
Print ISSN: 2395-6011 | Online ISSN: 2395-602X (www.ijsrst.com)

Emerging Trend of Meta-heuristic Solution Approaches for GAP

Dr. Harshalata Rohida
Smt. Narsamma Arts, Commerce and Science college, Kiran Nagar, Amravati, Maharashtra, India

ABSTRACT

The Generalized Assignment Problem (GAP) is to find a minimum-cost assignment of tasks to agents such that each task is assigned to exactly one agent and also each agent's resource capacity is honored. The GAP is a NP-hard problem which includes job scheduling, routing, loading for flexible manufacturing system and facility location. During the last three decades, most recent papers describe heuristic methods for generating good solutions due to the difficulty in solving hard GAPs to optimality. In this paper we mainly concentrate on some of the most recent Meta-heuristic solution approaches. We present some of the combinatorial optimization problem with similar structure to that of the GAP. We will also see in this collective material that this extra ordinary Meta-heuristic is very smooth to solve GAPs. Lastly, some real life applications will be discussed for more understanding of GAP.
Keywords: Generalized Assignment Problem (GAP), Meta-heuristic Approaches.

I. INTRODUCTION

The GAP is a well–known combinatorial optimization problem with various real-life applications. The g*eneralized assignment problem* (GAP) has been the subject of numerous research papers since 1975. The problem is to find a minimum-cost assignment of tasks (or jobs) to agents (or machines) such that each task is assigned to exactly one agent and also each agent's resource capacity is honored. GAP applications run the gamut from job scheduling (Balachandran 1972) to routing (Fisher and Jaikumar 1981) to loading for flexible manufacturing systems (Mazolla et al. 1989) to facility location (Ross and Soland 1977). A review of applications and algorithms (both exact and heuristic) appears in Cattrysse and Van Wassenhove (1992). Both the minimization and maximization type objective functions are used for the GAP. Consider a minimization problem Martello and Toth (1990) remarked that both of these cases are equivalent since it is possible to transform one problem to the other by using the negative of the objective function parameters. Ever since Ross and Soland (1975) have introduced the GAP, many papers have been published on this popular combinatorial optimization problem. After a brief introduction of the GAP applications, the authors have discussed the solution algorithms the GAP is known to be NP-hard (Sahni and Gonzalez, 1976) and even the problem of finding a feasible assignment is NP-Complete (Fisher and Jaikumar, 1981). Therefore its large instances are computationally intractable and as a result, there are several heuristic as well as metaheuristic solution approaches. A general overview of heuristic and metaheuristic approaches for the GAP can be found in Osman (1995) and Yagiura and Ibaraki (2004 , 2007).

Many applications of the GAP, from production planning to supply chain, from telecommunication to facility layout are cited. However, there are too many that we cannot enumerate all of them in this paper. We have also tried to shortly summarize some of the recent developments on the solution approaches for this problem.

II. MODEL FORMULATION

The GAP may be formulated as a 0–1 integer linear programming (ILP) model. Let n be the number of tasks to be assigned to m agents ($n \geq m$) and define $N = \{1,2,....,n\}$. We define the requisite data elements as follows:

c_{ij} = cost of task j being assigned to agent i

r_{ij} = amount of resource required for task j by agent i

b_i = resource units available to agent i.

The decision variables are defined as:

$$x_{ij} = \begin{cases} 1, & \text{if task } j \text{ is assigned to agent } i \\ 0, & \text{if not.} \end{cases}$$

The 0–1 ILP model may then be written as:

$$(P) \qquad \text{minimize} \qquad \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \qquad (1)$$

$$\text{subject to:} \qquad \sum_{j=1}^{n} r_{ij}x_{ij} \leq b_i, \qquad \forall i \qquad (2)$$

$$\sum_{i=1}^{m} x_{ij} = 1, \qquad \forall j \in N \qquad (3)$$

$$x_{ij} = 0 \text{ or } 1, \qquad \forall i, j. \qquad (4)$$

The objective function (1) is to minimize the total assignment cost of items to agents, while constraint (2) enforces the resource limitation for each agent. Constraint (3) ensures that each job is assigned to exactly one agent. Finally, constraints (4) enforce the integrality condition on the decision variables.

III. GAP TYPE PROBLEMS

In this section, we review some of the combinatorial optimization problems which have similar structure to that of the GAP. The relationship between the GAP and other combinatorial optimization problems have been discussed by some researchers. In 1997, Ferland has showed that the basic assignment structure is embedded in several combinatorial optimization problems such as the GAP, the Timetabling Problem, the Graph Coloring Problem and the Graph Partitioning Problem, hence the GAP can be considered as a generalization of the Assignment Problem (AP) (Kuhn, 1955). A general overview and discussion of assignment type problems and their generalizations have been presented by Ferland in 1997 . On the other side, the GAP has also been considered as the generalization of the Pairing Problem (Toktas, 2004), the Semi Matching Problem (Harvey et al. 2006) and the Parallel Machine Scheduling Problem (Shmoys and Tardos, 1993). Now, we consider the GAP as a special case of the Weighted Assignment Problem (WAP) (Ross and Zoltners, 1979), since the WAP generalizes both the constraints and variables of the GAP. The WAP is to find the optimal assignment of a set of tasks to a set of agents such that each task is performed by exactly one agent. Ross and Zoltners (1979) have discussed several problems as special cases of the WAP and also presented several other models equivalent to the WAP. They have introduced a generalization of the WAP: the Multi-Resource Weighted Assignment Problem. So, briefly we have discussed the solution algorithms and applications of the WAP and its variations. Let us define the binary variable x_{ijk} equal to 1 if and

only if task j is completed by agent i at performance level k. The WAP formulation as follow:

$$\text{WAP : } \min \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k \in K_{ij}} C_{ijk} x_{ijk} \quad (5)$$

$$\text{s.t.} \sum_{i=1}^{m} \sum_{k \in K_{ij}} x_{ijk} = 1 \qquad j = 1,\ldots,n \quad (6)$$

$$\alpha_i \leq \sum_{i=1}^{n} \sum_{k \in K_{ij}} r x_{ijk} x_{ijk} \leq b_i \qquad i = 1,\ldots,m \quad (7)$$

$$\sum_{i=1}^{m} \sum_{k \in K_{ij}} s_{ijk} x_{ijk} \begin{bmatrix} \leq \\ \geq \\ = \end{bmatrix} e_j \qquad j = 1,\ldots,n \quad (8)$$

$$x_{ijk} = \{0,1\} \qquad \forall i, j, k. \quad (9)$$

Constraints (6) ensure that each task *j* is performed by agent *i* at a certain performance level k and performance levels K_{ij} are fixed for each agent *i* and task *j*. Constraints (7) denote the resource consumption limits of each agent. The constraints (8) indicate the boundaries of the used resource when the job *j* is completed. It is clear that when there is no lower limit on the resource consumption (i.e. $a_i = 0$), there is no boundary on the used resource and there is only a single performance level for all task-agent pairs; the WAP becomes the GAP.

IV. SOLUTION PROCEDURES FOR THE GAP

The GAP is shown to be NP-Hard (Sahni and Gonzalez, 1976). Unless P = NP, there is no polynomial time algorithm which finds a feasible solution to an arbitrary instance of the GAP. Due to small sized instances in exact solution, several heuristic as well as metaheuristic approaches have been devised to overcome the limitations of exact methods. Metaheuristics intensively use neighborhood search methods to explore the solution space without necessarily improving the objective function and sometimes allowing infeasible moves. In this section, we look an overview some of the solution approaches for the GAP, many approaches, from state-of-the-art metaheuristics to variable neighborhood search algorithms and from exact solution procedures to simple heuristic algorithms, have been applied to this well known combinatorial optimization problem. As we know there are four solution procedures the GAP: Approximate solution approaches, Metaheuristic solutuion approaches, Relaxations and Exact approaches. Here, we mainly discuss the metaheuristic solution approaches.

4.1. Metaheuristic Solution Approaches

We will discuss some metaheuristic solution approaches which can be briefly described as follows.

4.1.1. Simulated Annealing

Simulated Annealing (SA) is a local optimization method for solving hard combinatorial optimization problems. Kirkpatrick *et al.* (1983), and Cerny (1985) showed how a modelfor simulating the annealing process of solids, as proposed by Metropolis et al. (1953), could be used for optimization problems. SA has been applied to many optimization problems such as locational analysis, molecular physics and chemistry, image processing, and job shop scheduling Eglese (1990). SA is to find a feasible solution of minimum cost function: For each optimization problem, there is a set *S* of feasible solutions, each solution *s* having a cost function *f*(*s*). Generally, SA algorithm begins with an initial solution (taking at random). A neighborhood of the solution is generated by some suitable mechanism and the change in the objective function is calculated. If a reduction in the objective function is found, the current solution is replaced by the neighborhood solution, otherwise, the neighborhood solution is accepted with a certain probability. The probability of accepting an uphill move is normally set to $exp(-\Delta/T)$ where *T* is a control parameter which corresponds to temperature in the analogy with physical annealing, and Δ is the change in the objective function value. The acceptance function shows that small increases in the objective function are more likely to be accepted than large increases.

T is high, most moves will be accepted, but as T approaches zero most uphill moves will be rejected. So, in SA, the algorithm starts with a relatively high temperature, to avoid being prematurely trapped at a local optimum. The algorithm attempts a certain number of moves at each temperature while the temperature parameter drops gradually.

4.1.2. Tabu Search

Tabu search was originally proposed by Glover, and since then these metaheuristics have been subject to extensive studies and applied to several optimization problems with great success. Tabu search can be treat as a intelligent search that uses memory to drive the search out of local optimal solutions and find good results. Our motivation to apply tabu search to GAP was the excellent results obtained by Laguna et al. for a multilevel generalized assignment problem. They also presented computational results to the GAP and were able to obtain always the optimal solution for test problems with 5 agents and 25 task in less than 1.30 seconds.

4.1.3. Genetic Algorithm

Genetic Algorithms are a family of computational models inspired by evolution. These algorithms encode a potential solution to a specific problem on a simple chromosome-like data structure. Genetic algorithms are often viewed as function optimizer, although the range of problems to which genetic algorithms have been applied are quite broad. An implementation of genetic algorithm begins with a population of (purely random) chromosomes. One evaluates these structures and allocated reproductive opportunities for which these chromosomes which represent a better solution to the target problem are given more chances to `reproduce' than those chromosomes which are poorer solutions. The 'goodness' of a solution is typically defined with respect to the current population.

Genetic algorithm are the main paradigm of evolutionary computing. GAs are inspired by Darwin's theory about evolution "survival of the fittest". Some important points should be needed:

- GAs are the ways of solving problems by mimicking processes nature uses i.e. Selection, Crosses over, Mutation and Accepting, to evolve a solution to a problem.
- GAs are adaptive heuristic search based on the evolutionary ideas of natural selection and genetic.
- GAs , although randomized, exploit historical information to direct the search into the region of better performance within the search space.
- GAs are intelligent exploitation of random search used in optimization problems.

4.1.4. Neural Networks

Neural networks (NN) are mainly adaptive learning processes that continuously update some weights until an acceptable, namely a feasible or close to feasible solution is reached. Neural networks has been used to solve combinatorial optimization problems. The study by Li and Luyuan (1991) is the very first attempt to solve the GAP using competition based neural networks. They have defined an $m \times n$ matrix where at each position there exists a neuron which compete to become active. In 2004 Monfared and Etemadi have tried four different methods to structure the energy function of the neural network: exterior penalty function method, augmented Lagrangian method, dual Lagrangian method and interior penalty function method. The augmented Lagrangian method could yield better solutions than the other methods with respect to integrality measure while maintaining feasibility and stability measures.

When dealing with different metaheuristic approaches, there is no clear answer for the question that which one is the best approach? So, we can compare the performance of metaheuristics by solution quality and speed.

V. REAL LIFE APPLICATIONS RELATED WITH THE GAP

We mention some of the real life application the GAP.

5.1. Production Planning Applications

There are many applications of the GAP in production planning. Dobson and Nambimadom (2001) have discussed the Batch Loading Problem (NP-Hard problem). Given a sequence of batches, the Batch Loading Problem (BLP) is to assign jobs to the batches. The BLP is a special case of the GAP with $a_j = r_{ij}$ where a_j is the volume of job j and $b = b_i$ where b is the capacity of the processor, $x_{ij} = 1$ if and only if job j is assigned to batch i. where, the cost function of the BLP is much more complicated and difficult than the GAP (Dobson and Nambimadom, 2001).

5.2. Telecommunication Applications

A telecommunication application of the GAP was mentioned in the paper of Bressoud et al. (2003) for the optimal configuration for Border Gateway Protocol (BGP). BGPs play a crucial role in the control of transit traffic flows from customers and providers. They have addressed the optimization problem of the routing traffic cost. Another telecommunication application which is regarded as an extension of the GAP, is the maximal covering code multiplexing access telecommunication networks with capacity constraints by Barbas and Marin (2004). The problem is to find the optimal assignment of terminals to base stations.

5.3. Transportation and Routing Applications

Ruland (1999) have discussed the transportation of patients between military hospitals in the United States. He solved the GAP to determine the assignment of patients to flights. The primary objective function is to minimize the patient inconvenience, namely the number of nights that patients remain overnight. The secondary objective is to minimize the flying time or route length. The GAP arises as a subproblem in solving the well-known VRP. Fisher and Jaikumar (1981) have proposed a generalized assignment approach for the solution of the VRP. Finally, they have solved the Traveling Salesman Problem (TSP) for each set of cities. In their method, the selection of seeds are also allowed manually to include human judgement and knowledge into the solution procedure. Baker and Sheasby (1999) have reported that the optimal solution of the GAP for determining seed position does not guarantee the best result for the VRP.

5.4. Scheduling Applications

Many applications of the GAP appear within the scheduling problems (Cattryse and Van Wassenhove, 1992). Employee scheduling, machine scheduling, multiprocessor taskscheduling, workforce planning, classroom scheduling batching etc. Many applications of the GAP appearing in resource scheduling have been discussed by Zimokha and Rubinstein (1988). One scheduling problem that includes the GAP as a subproblem arises in project networks. Drexl (1991) has addressed the Job Assignment Problem arising in nonpreemptive resource constrained project networks. The objective is to find a minimum cost assignment of jobs to a set of resources (labour) for a limited period of time.

VI. CONCLUSION

The GAP is a well known problem which is to find the optimal assignment of n items into m agents where each of which a fixed capacity availability. This research paper has discussed the different metaheuristic approaches for GAP. Metaheuristic has evolved rapidly in an attempt to find “close to optimal” solutions to industrial problems. A real life sugar cane optimization problems (Higgins 1999), which was implemented and delivered benefits to growers throughout the Australian sugar industry, required the solution of a generalized assignment problem with 150,000 jobs and 12 agents. Whilst existing metaheuristic approaches can applied to such large problems, the need to search a very large neighborhood gave size to the need of a more specialized heuristic. Different approaches keep efficient power to solve the large problems. We have shown the general structure of metaheuristic approaches. A real life applications of GAP can be viewed in the last section of the paper for which we can more comfort with metaheuristic approaches for GAP. These approaches can also be helped to solved problems in other area of mathematics.

VII. REFERENCES

[1]. Al-khedhairi A. (2008) Simulated Annealing Metaheuristic for Solving P-Median Problem. Int. J. Contemp. Math. Sciences, Vol. 3, 2008, no. 28, 1357 – 1365.

[2]. Beasley, D., Bull, D. R. and Martin, R. R. (1993), "An overview of genetic algorithms: Part 2, research topics",UniversityComputing, Vol. 15, pp. 170{181. citeseer.nj.nec.com/article/beasley93overview.html.

[3]. Burke Edmund K., Curtois Timothy, Rong Qu and Berghe Greet Vanden (2007) A Time Pre-defined Variable Depth Search for Nurse Rostering. Computer Science Technical Report No. NOTTCS-TR-2007-6.

[4]. Chakraborty R C (2010) Fundamental of Genetic Algorithms: AI course Lecture 39-40.

[5]. Chambers, L. (1995), Practical handbook of genetic algorithms: Applications, Vol.I, CRC Press, Boca Raton, Florida.

[6]. Chu, P.C. and Beasley, J.E. (1997) A genetic algorithm for the generalized assignment problem. Computers and Operations Research, 24 (1): 17–23.

[7]. Feltl Harald and Raidl R. (2004) An Improved Hybrid Genetic Algorithm for the Generalized Assignment Problem. SAC Nicosia, Cyprus Copyright 2004 ACM -158113-812-1/03/04.

[8]. Ferland, J.A. (1997) Generalized assignment-type problems a powerful modeling scheme. In: Burke, E.K. and Carter, M.W.(eds.). The Practice and Theory of Automated Timetabling II. Springer Lecture Notes in Computer Science, 1408, 53–77.

[9]. Ferland, J.A., Hertz, A., and Lavoie, A. (1996) An object-oriented methodology for solving assignment-type problems with neighborhood search techniques. Operations Research, 44 (2): 347–359.

[10]. Lourenço Helena Ramalhinho and Serra Daniel (1998) Adaptive Approach Heuristics For The Generalized Assignment Problem. Department of Economics and Management, Universitat Pompeu Fabra, R. Trias Fargas 25-27, 08005 Barcelona, Spain.