

Scrounging-Act Scheme of Bees for Generalized Assignment Problem

Dilip K¹

4th Sem M.Tech, Dept. of Computer Science
SJB Institute of Technology
Bangalore, India
dilipklenovo@gmail.com

Mohan Kumar K N²

Assistant Professor, Dept. of Computer Science
SJB Institute of Technology
Bangalore, India
mohan4183@gmail.com

Abstract — The Scrounging-Act Scheme of Bees is taken in to consideration for modeling, which reflects the population characteristics of Bees. The honey bees' natural conduct in food scrounging-Act is what the model tries to contrive. The model developed is then used to sensibly estimate solution for functional optimization. The approximation behind the model is based on how the honey bees situate food sources optimally using various mechanisms like waggle dance. Based on this approximation an attempt is made to demonstrate the performance of algorithm inspired by bee, i.e. Artificial Bee Colony on a NP-hard problem which is known as generalized assignment problem.

Also in market, genetic algorithm is easily implemented to get feasible solution for generalized assignment problem. Most recent and more advanced research on swarm intelligence shows that Bees' inspired algorithm has the capability to produce more accurate feasible solution than genetic algorithm or any other algorithms. Swarm intelligence is very difficult and complex to implement. No standardized and reusable module to solve generalized assignment problem has been developed. The motivation behind the project is that previous work has confronted that bee inspired algorithms have a very promising potential for modeling and solving complex optimization problems. But there is still a long way to go in order to fully utilize the potential of bee inspired algorithms. The main focus is on introducing modified version of ABC algorithm for optimization problems and develop standardized and reusable module to solve the same.

Keywords—Artificial Bee Colony (ABC), Combinatorial Optimization, Generalized Assignment Problem (GAP) NP-Hard Problem, Scrounging-Act.

I. INTRODUCTION

Swarm Intelligence (SI) is a research branch and is a part of Artificial Intelligence based on study of actions of individuals in various decentralized systems. It is an emerging arena in the field of optimization and the researchers have formulated various algorithms by mocking up the behaviors. It is a collective behavior of decentralized, self-organized systems, artificial or natural.

It is an SI systems mostly made up of a population of simple agents or factors interacting locally with one another and with their environment. The aspiration often comes from nature, peculiarly biological systems. The agents pursue very bare rules, and although there is no concentrated control structure prescribing how individual agents should conduct, local, and to a sealed degree random, such agents interactions lead to the egress of "intelligent" global behavior, strange to

the individual agents. Natural examples of Swarm Intelligence include ant colonies, bee colonies, animal herding, bird flocking, fish schooling, and bacterial growth. The Swarm Intelligence based optimization algorithm i.e. is inspired by the foraging behavior of honey bees has inspired the Bees Algorithm

Honey bee optimization algorithms constitute a new trend in the field of swarm intelligence. This algorithms class is based on the behavior of honeybees. The algorithms currently are based on either of two principles- foraging or mating. Foraging in honeybees is interesting, as the underlying decentralized decision-making processes enable a colony to balance exploitation of known food sources with exploration for new and potentially better-food sources in a dynamic environment. Algorithms based on foraging usually use artificial bees to search for solutions and thus associate solutions with food sources. The Bees Algorithm was originated to be used with continuous and combinatorial Function optimization problem.

The latest research reveals that the nature inspired population based optimization algorithms are capable of finding optimal and near optimal solutions to the difficult optimization problems. There are two classes of population based algorithms. They are evolutionary algorithms and swarm intelligence based algorithms.

The Artificial Bee Colony (ABC) algorithm is relatively new swarm intelligence based optimizer. It mimics the cooperative foraging behavior of a swarm of honey bees. ABC is simple and flexible as it employs less control parameters. ABC at the beginning was aimed by Karaboga in 2005 for optimizing multi-modal and multi-variable continuous functions. The exchange of information among bees is the most important happening in the establishment of collective knowledge. The most recent research has revealed some good properties of ABC. Particularly, the number of control parameters in ABC is fewer than that of other population-based algorithms, which makes it easier to be enforced. Meanwhile, the optimization performance of ABC is comparable and sometimes superior to the state-of-the-art meta- heuristics. Consequently, ABC has elicited much interest and has been successfully applied to different kinds of optimization problems. Few algorithms inspired by bees' behavior appeared during the last decade such as Bee System, BCO algorithm, E ABC algorithm, MBO, Bees algorithm, VBA algorithm, Interactive Artificial Bee Colony.

Any attempt to design algorithms or distributed problem-solving devices inspired by the collective behavior of social insect colonies and other animal societies: swarm intelligence.

II. RELEVANT WORK

[1] I.H. Osman, Heuristics for the generalized assignment problem: simulated annealing and tabu search approaches. This paper deals with solving the generalized assignment problem to obtain optimal or near optimal solution. He is using a set of GAP instances from the OR-library which are set of standard problems as dataset. A λ -generation mechanism is introduced in this paper. Different search strategies and parameter settings are looked in for the λ -generation descent, hybrid simulated annealing/tabu search and tabu search heuristic methods. The resulting algorithms showed beneficial outcome in solving few among five instances from the OR-Library. Unlike exact algorithms, this method was capable of solving only few difficult series instances. He believes that the searching pattern can be further optimized which he considers carefully for his future works. Kluwer Academic publishers, Boston, 1999, pp.459–471.

[2] M. Yagiura, T. Yamaguchi, T. Ibaraki, A variable-depth search algorithm for the generalized assignment problem. This paper deals with adaptively changing the size of neighborhood so that it can effectively traverse larger search space while keeping the amount of computational time reasonable. The benchmark instances called types C and D with up to $n=200$ and $m=20$ (tasks and agent respectively) taken from OR-Library (<http://mscmga.ms.ic.ac.uk/jeb/orlib/gapinfo.html>) and instances type E generated by themselves, which are available at <http://www.kuamp.kyoto.ac.jp/labs/or/members/yagiura/gap/> were used as dataset. A heuristic algorithm based on Variable Depth Search (VDS) for generalized assignment problem is proposed in this paper. Modified swift neighborhood and swap neighborhood were the methods introduced to adaptively change the size of neighborhood so that it can effectively traverse larger search space while keeping the amount of computational time reasonable. The proposed algorithm was capable of solving the problem effectively keeping the amount of computational time reasonable better than Tabu, Tabu-CSP, R&A and Genetic algorithms. For some series D instances, this method was not effective enough of obtaining the feasible solution and also the VDS is sensitive to the parameter α (the weight on the penalty term). Incorporating automatic tuning ability of α to their algorithm is thought for his future work.

[3] M. Yagiura, T. Ibaraki, F. Glover; An ejection chain approach for the generalized assignment problem. This paper deals with featuring an ejection chain approach on TS framework, which is embedded in a neighborhood construction to create more complex and powerful moves for solving generalized assignment problem. Five types of benchmark instances called types A, B, C, D, and E (Chu and Beasley (1997), Laguna et al.1995) were used as dataset for computational study. Sub-gradient Phase, Long Chain Neighborhood, Double shift neighborhood are the strategic moves method introduced to solve the problem. Adaptive mechanism is incorporated in Tabu-search for adjusting search parameters, to maintain a balance between visits to feasible and infeasible regions. Tabu-Search algorithm obtained optimal solutions for all the tested instances within a few seconds on small and medium problems compared to Exact Algorithm. As future research, they plan for further optimization of the

algorithm. This algorithm's outcome is compared with only exact algorithm. *Informatics Journal of Computing* 16 (2004) 131–151.

[4] H.R. Lourenco, D. Serra, Adaptive search heuristics for the generalized assignment problem. This paper deals with presenting hybrid algorithms consisting of adaptive construction heuristics and subsequent application of local search to solve the GAP. The test problems available from the OR library: <http://www.ms.ic.ac.uk/info.html> was used in their computational experiments. These adaptive heuristics are based on two meta-heuristic approaches to solve combinatorial optimization problems: the Greedy Randomized Adaptive Search Procedure (GRASP) and the *Max-Min* Ant System (MMAS), a specific algorithm falling into the framework of the Ant Colony Optimization meta-heuristic. The outcome was that on average the ASH+LS+TS performed better than other approaches for the above mentioned instances (test problems). As future research, they plan to apply the adaptive search heuristics based on the general framework to develop solution methods for other combinatorial optimization problem. *Mathware and Soft Computing* 9 (2002) 209–234.

[5] L. Alfandari, A. Plateau, P. Tolla, A path relinking algorithm for the generalized assignment problem. This paper deals with featuring path relinking approach, which is a mechanism for generating new solutions by combining two or more reference solutions for solving generalized assignment problem. It also features an ejection chain approach, which is embedded in a neighborhood construction to create more complex and powerful moves. Types A, B, C, D, E instances taken from: <http://mscmga.ms.ic.ac.uk/jeb/orlib/gapinfo.html>. and <http://www-or.amp.i.kyoto-u.ac.jp/~yagiura/gap/>. are used as dataset. The PREC algorithm's outcome in solving the problem is very effective and its performance is the best among the tested algorithms. Though the rules for the path relinking component are simple, their path relinking approach is efficient. This optimization approach is problem dependent to obtain the solution. As future research, they plan for further optimization of the algorithm. Kluwer Academic Publishers, Boston, 2005, pp. 1–17.

[6] Mutsunori Yagiura, Shinji Iwasaki, Toshihide Ibaraki, Fred Glover, A Very Large-Scale Neighborhood Search Algorithm for the Multi-Resource to generalized assignment problem. This paper deals with featuring a very large-scale neighborhood search, which is a mechanism of conducting the search with complex and powerful moves, where the resulting neighborhood is efficiently searched via the improvement graph. Five types of benchmark instances of GAP called types A, B, C, D and E were used as dataset. They proposed a tabu search algorithm in which a sophisticated neighborhood called the chained shift neighborhood is incorporated. It was confirmed through computational comparisons on benchmark instances that the method is effective, especially for type D and E instances, which are known to be very difficult.

[7] H. Feltl, G.R. Raidl, An improved hybrid genetic algorithm for the generalized assignment problem. This paper deals with finding near optimal solutions to the difficult optimization problems by motivation from nature. They are using Types A, B, C, D, E instances taken from OR libraries as dataset. GA is based on genetic science and natural selection and it attempts to simulate the phenomenon of natural evolution. The GA consists of five components. These are a

random number generator, a fitness evaluation unit and genetic operators for reproductions, cross over and mutation operations. New generations of solutions are produced with each phase of the algorithm. The comparative study performed on results obtained by GA had no significant variations compared to the existing methods except for few benchmark functions. Apart from the maximum evaluation number and the population size, a standard GA has three more control parameters (cross over, mutation rate, generation gap). In other terms; it requires too many parameters which is a limitation. A standard DE has at least two control parameters (crossover rate, scaling factor). Further optimization of algorithm to gain potential utilization is scheduled for future work. Proceedings of the 2004 ACM Symposium on Applied Computing, Nicosia, Cyprus, 2004, pp. 990–995.

The previous work has presented that bee inspired algorithms have a very promising potential for modeling and solving complex optimization problems. But there is still a long way to go in order to fully utilize the potential of bee inspired algorithms. With these yet to optimize for full potential utilization in modeling and solving complex optimization problems will be taken in to consideration for further studies.

Also In market, genetic algorithm is easily implemented to get feasible solution for generalized assignment problem. Most recent and more advanced research on swarm intelligence shows that Bees' inspired algorithm has the capability to produce more accurate feasible solution than genetic algorithm or any other algorithms. Swarm intelligence is very difficult and complex to implement. No standardized and reusable module to solve generalized assignment problem has been developed. Even though swarm intelligence has been proved to be better than genetic algorithm, commonly they have not been implemented because of complexity. This reason has made everyone in market to go on with genetic algorithm.

III. ANALYSIS

Generalized assignment problem is a combinatorial optimization problem and is a classic scheduling problem with many real-life applications. The main role in GAP is to assign a set of tasks to a set of agents with a minimum total cost. Each agent represents a single resource with limited capacity. Each task must be assigned to only one agent and it requires a certain amount of the resource of the agent. Primary concern on this problem occurred from its NP hard structure and the NP-completeness of proving that a solution is a feasible solution. Swarm intelligence is very difficult and complex to implement. No standardized and reusable module to solve generalized assignment problem has been developed. The former work has confronted that bee inspired algorithms have a very promising potential for modeling and solving complex optimization problems. But there is still a long way to go in order to fully utilize the potential of bee inspired algorithms.

The abstract bees' algorithm's methods are revised and modified to incorporate a number of features to obtain optimal and near optimal solutions for generalized assignment problem. Features like an ejection chain approach which is engrafted in a neighborhood construction to create more complex and powerful moves, adaptive mechanism which is embedded for adjusting search parameters, to maintain a balance between visits to feasible and infeasible regions automatic tuning ability of α through modified swift and swap neighborhood, adaptive

search heuristics are applied based on the general framework to develop solution methods. The gap between Research and Engineering is bridged. The abstract bees' algorithm in mathematical language is efficiently engineered into concrete program/module using the construct of the target programming language such that the efficiency of the algorithm is not compromised. The design and application are developed more efficiently in the target programming language along the technological constraints making sure that the quality in attaining the objective is reflected in the program. The module/program is developed using Object Oriented model and the cutting edge software engineering methods so that it can be easily integrated and quickly be used by others.

IV. PROPOSED MODEL

The Modules that I am using in my project are classified into 2 engines.

- The User Interface (UI) is simple and provides a user friendly feature with minimum security, and
- Swarm Intelligence Core (SI Core) is the main integral part of the project which has algorithm complexity in it and this can be used to find optimal solution to NP-hard problem.

Fig. 1 depicts the main modules and its orientation modeled to solve the combinatorial optimization problem.

The main modules of SI core engine are-

1. **SI Engine:** It is nothing but the coded program of Artificial Bee Colony Algorithm
2. **GRAH** (Greedy Randomized Adaptive Search Heuristic): It is used to construct initial Employed Bee Colony solutions.
3. **Neighborhood Search:** It is used to find more feasible solutions.
4. **Fitness Evaluation:** It is a function used to find the best optimal solution.

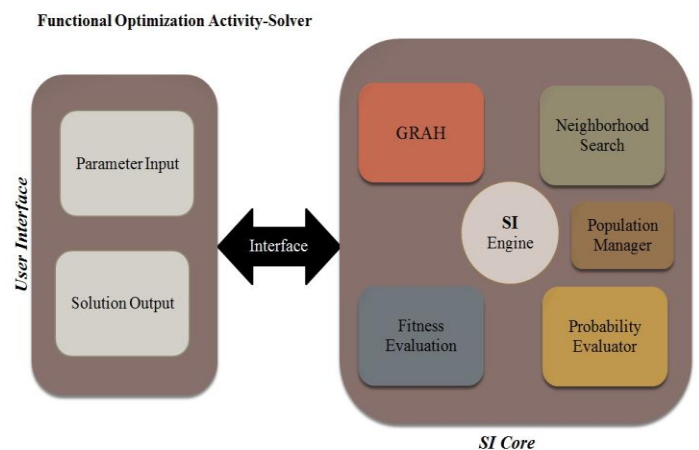


Fig. 1: System Modules and Description

V. MODIFIED BEES' ALGORITHM

Primarily Stimulus, The Data flow inside SI core engine can be easily perceivable utilizing the adopting Data Flow diagram. The food source whose nectar is abandoned or exhausted by the bees is replaced with a new food source by

the scouts. In the ABC algorithm this is simulated by randomly producing a position and replacing it with the abandoned one. If a position cannot be improved further through a predetermined number of cycles called limit then that food source is assumed to be abandoned. After each candidate source position v_{ij} is produced and then evaluated by the artificial bee, its performance is compared with that of x_{ij} . If the new food has equal or better nectar than the old source, it is replaced with the old one in the memory. Otherwise, the old one is retained. In other words, a greedy selection mechanism is employed as the selection operation between the old and the current food sources.

The main steps of the algorithm are given by

Step 1: Initialize the population of solutions & evaluate them.

Step 2: Produce new solutions for the employed bees, evaluate them and apply the greedy selection mechanism.

Step 3: Calculate the probabilities of the current sources with which they are preferred by the onlookers.

Step 4: Assign onlooker bees to employed bees according to probabilities, produce new solutions and apply the greedy selection mechanism.

Step 5: Stop the exploitation process of the sources abandoned by bees and send the scouts in the search area for discovering new food sources, randomly.

Step 6: Memorize the best food source found so far.

Step 7: If the termination condition is not satisfied, go to step 2, other-wise stop/halt the algorithm.

These main steps of the algorithm are adopted with in the SI core engine with integrated modules such as GRAH (Greedy Random Adaptive Search Heuristics), Neighborhood search, Population manager, Probability Evaluator, Fitness Evaluation to attain the reachability of features like an ejection chain approach engrafted in a neighborhood construction to create more complex and powerful moves, adaptive mechanism embedded for adjusting search parameters, to maintain a balance between visits to feasible and infeasible regions. These modules working together as an integral part of the SI core Engine leads to the automation of incorporating automatic tuning ability of α (parameter- the cost of using one unit of overloaded capacity). In other terms, it confronts the general framework to develop solution methods. The below diagram describes more generalized steps of how bees collectively behave in nature to aggregate nectar from rich food sources found in the neighborhood by their intelligent collective behavior performances.

Fig. 2 gives the abstract view of nature's perspective bee algorithm. In computer science and operations research, the Artificial Bee Colony (ABC) algorithm is an optimization algorithm based on the intelligent foraging behavior of honey bee swarm, proposed by Karaboga in 2005.

The process of the ABC algorithm is presented as follows

Step 1: Initialization: Spray ne percentage of the populations into the solution space randomly, and then calculate their fitness values, which are called the nectar amounts, where ne represents the ratio of employed bees to the total population. Once these populations are positioned into the solution space, they are called the employed bees.

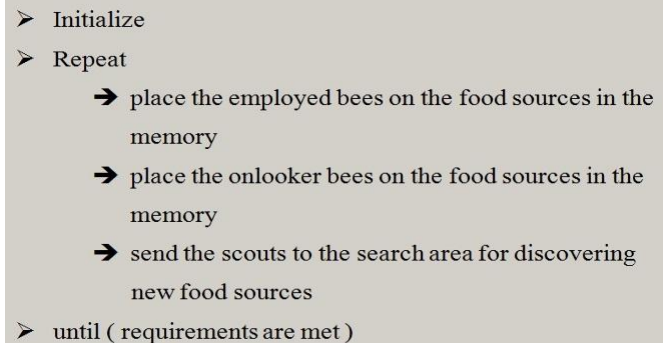


Fig. 2: majorly generalized nature's perspective bee algorithm

Step 2: Move the Onlookers: Calculate the probability of selecting a food source, select a food source to move to by roulette wheel selection for every onlooker bees and then determine the nectar amounts of them.

Step 3: Move the Scouts: If the fitness values of the employed bees do not be improved by a continuous predetermined number of iterations, which is called "Limit", those food sources are abandoned, and these employed bees become the scouts.

Step 4: Update the Best Food Source Found So Far: Memorize the best fitness value and the position, which are found by the bees.

Step 5: Termination Checking: Check if the amount of the iterations satisfies the termination condition. If the termination condition is satisfied, terminate the program and output the results, otherwise go back to the Step 2.

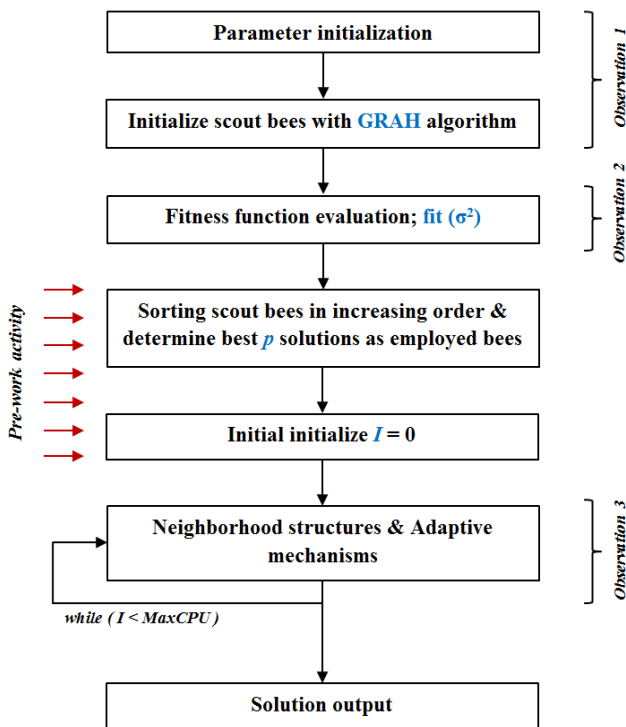


Fig. 3: pragmatic approach flow chart of bee algorithm

Pragmatic data flow

The produced Fig. 3 represents the actual practical data flow diagram of proposed modified bees' algorithm for generalized assignment problem. At the beginning, with keen observation parameter initialization and initialization of scout bees with GRAH algorithm is induced. Then, the fitness function evaluation is reckoned. Later on, scout bees are sorted in an increasing order and best p solutions out of it as employed bees are determined. Followed by, initial parameter setting is induced. Sorting of scout bees and determining best p solutions as employed bees are the pre-work activity performed. The outcome of the pre-work activity is probed in to Neighborhood structures and Adaptive mechanisms to attain the objective.

Neighborhood structures and Adaptive mechanisms are incorporated with features like an ejection chain approach engrafted in a neighborhood construction to create more complex and powerful moves, adaptive mechanism embedded for adjusting search parameters, to maintain a balance between visits to feasible and infeasible regions.

The more practical and observational framework flowchart of Bee algorithm obtained on refining. More refined and simpler framework for perceiving the bees algorithm's functioning is constituted. The adopted supportive pragmatic approach flow chart of bee algorithm represents the same. This flow chart fosters a pragmatic approach to problem solving of generalized assignment problem (GAP). This flowchart diagram design technique defers implementation details until later stages of design to preserve flexibility.

Approach for the solution may be tuned-up with these co-curricular activities:-

- A catalog of the collective behaviors of swarms or social insects is created then
- Model simulating how social insects collectively perform tasks is formulated
- This model is used as a basis upon which artificial variations can be developed
- Model parameters can be tuned within a biologically relevant range or by adding non-biological factors to the model

The abstract bees' algorithm's methods are revised and modified to incorporate a number of features to obtain optimal and near optimal solutions for generalized assignment problem. Features like an ejection chain approach is engrafted in a neighborhood construction to create more complex and powerful moves, adaptive mechanism is embedded for adjusting search parameters, to maintain a balance between visits to feasible and infeasible regions, automatic tuning ability of α through modified swift and swap neighborhood, adaptive search heuristics are applied based on the general framework to develop solution methods.

VI. EXPERIMENTATION

The proposed Bees' algorithm is coded and tested in a set of problems ranging from 5 agents-15 tasks to 10 agents-60 tasks. These test problems are publicly available from the www.OR-Library.com. The set of test problems can be divided into two groups: Gap1-Gap6/easy and Gap7-Gap12/difficult. Each problem set consists of 5 different problems with the same size, so there are $12 \times 5 = 60$ problems to solve. These set of problems are of maximization form of GAP and optimal values are known. Fig. 4 depicts the experimental results obtained after running the MBA.

problem	MBA	GA	ACO
	cost		
gapa1	1698(1)	1698(1)	1698(1)
gapa2	3235(1)	3235(1)	3235(1)
gapa3	1360(1)	1360(1)	1360(1)
gapa4	2623(1)	2623(1)	2623(1)
gapa5	1158(1)	1158(1)	1158(1)
gapa6	2339(1)	2339(1)	2341(2)
gapb1	1843(1)	1843(1)	1846(2)
gapb2	3552(1)	3553(2)	3561(3)
gapb3	1407(1)	1407(1)	1407(1)
gapb4	2827(1)	2829(2)	2849(3)
gapb5	1166(1)	1166(1)	1166(1)
gapb6	2339(1)	2340(2)	2350(3)
gapc1	1931(1)	1931(1)	1931(1)
gapc2	3456(1)	3457(2)	3464(3)
gapc3	1402(1)	1403(2)	1406(3)
gapc4	2806(1)	2807(2)	2825(3)
gapc5	1243(1)	1243(1)	1246(2)
gapc6	2392(1)	2396(2)	2411(3)
gapd1	6353(1)	6365(2)	6625(3)
gapd2	12744(1)	12767(2)	13197(3)
gapd3	6356(1)	6373(2)	6613(3)
gapd4	12442(1)	12536(2)	13024(3)
gapd5	6221(1)	6259(2)	6484(3)
gapd6	12276(1)	12386(2)	12951(3)

Fig. 4: Experimental results of MBA, GA and ACO

An example consisting of 3 agent and 6 task assignment problem available at www.OR-Library.com is implemented to

solve as a minimization problem. A Bee solution is represented as an array of tasks which contains the assignment of agents. Five runs for each problem are evaluated. Different algorithms that solved Gap1-Gap12 in the literature are determined for comparison. The values in Table represent the mean deviation from the optimal value for each problem set. Proposed algorithm found the optimal solutions in all five runs for all problem sets with previously defined parameters. As compared to other algorithms the proposed algorithm is unambiguously the best performer.

VII. DISCUSSION

The Experimentation results are taken into consideration and the respective graph representations are worked out. The below Fig. 5a and Fig. 5b represents graphs that symbolizes the Proposed Modified Bees' algorithm is unambiguously better performer than the other algorithms in the literature except for PREC (path relinking approach based on ejection chains) and TSEC (tabu search based on ejection chains). However, Literature suggests PREC and TSEC algorithms have high time complexity of CPU performance to solve the complex combinatorial problems (GAP problems).

On the other hand, the Proposed Modified Bees' Algorithm when compared with those algorithms that exist in market with their actual Standardization and Implementation, Firm belief is that the Proposed Modified Bes' Algorithm outperforms others.

This conclusion is based on the results obtained on solving the set of problems ranging from 5 agents-15 tasks to 10 agents-60 tasks. These test problems are publicly available from the www.OR-Library.com.

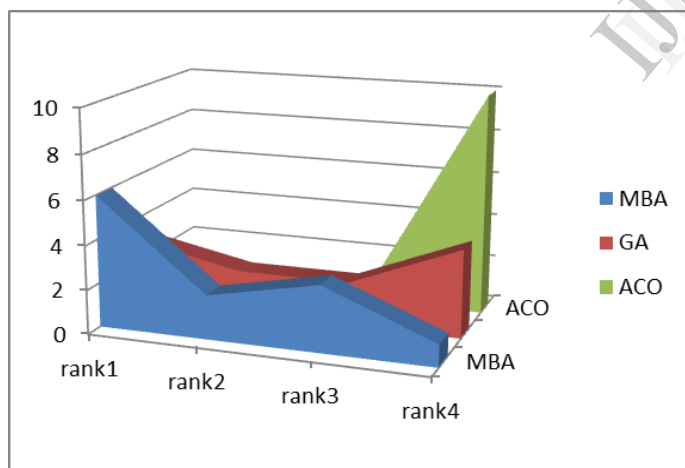


Fig. 5a: Solution quality based Ranking of algorithms

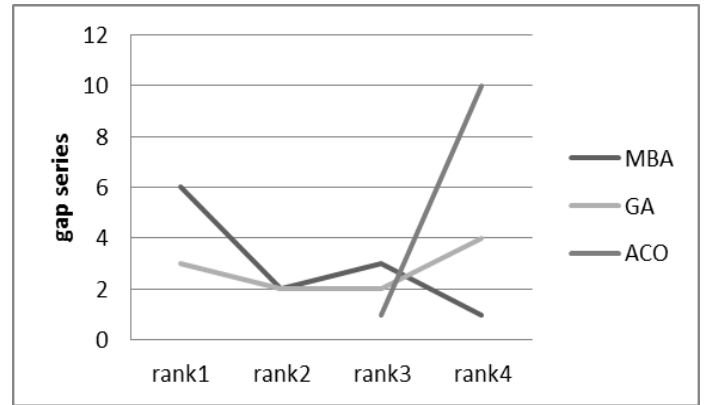


Fig. 5b: Outlook comparison of Ranked algorithms

The outcome obtained from the extensive computational study depicts that the proposed Bees' Algorithm has a big potential for solving complex assignment problems. In the present study no serious parameter optimization for the proposed Bees' Algorithm is carried out as this work is scheduled for future work.

VIII. CONCLUSION AND FUTURE SCOPE

The scrounging-act scheme of the bees is taken in to consideration for modeling and designing a purely object oriented reusable module to computationally obtain reasonably approximate solution for the Generalized Assignment Problem. A number of features is incorporated into the abstract bees' algorithm's methods to obtain optimal and near optimal solutions. Features like an ejection chain approach are engrafted in a neighborhood construction to create more complex and powerful moves, adaptive mechanism are embedded for adjusting search parameters, to maintain a balance between visits to feasible and infeasible regions, automatic tuning ability of α through modified swift and swap neighborhood structures, adaptive search heuristics are applied based on the general framework to develop solution methods. This revised and modified Bees' Algorithm is found very effective in solving small to medium sized generalized assignment problems and for solving larger size and tightly constrained generalized assignment problem. By performing a careful parameter optimization, I conceive that the performance of the proposed modified Bees' Algorithm can be meliorated further for larger size generalized assignment problems. This is scheduled as a future work.

ACKNOWLEDGMENT

I hereby thank my guide and all the authors of the papers mentioned below for their help and support.

REFERENCES

- [1] M. Savelsbergh, "A branch-and-price algorithm for the generalized assignment problem", *Operations Research* 45 (1997) 831–841. October, 2004.
- [2] I.H. Osman; "Heuristics for the generalized assignment problem": simulated annealing and tabu search approaches, *OR Spektrum* 17 (1995) 211–225.
- [3] M. Yagiura, T. Yamaguchi, T. Ibaraki, "A variable-depth search algorithm for the generalized assignment problem", in: S. Vob, S. Martello, I.H. Osman, C. Roucairol (Eds.), *Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization*, Kluwer Academic publishers, Boston, 1999, pp.459–471.
- [4] H.R. Lourenco, and D. Serra, "Adaptive search heuristics for the generalized assignment problem", *Mathware and Soft Computing* 9 (2002) 209–234.
- [5] L. Alfandari, A. Plateau, and P. Tolla, "A path relinking algorithm for the generalized assignment problem", in: M.G.C. Resende, J.D. Sousa (Eds.), *Metaheuristics: Computer Decision-Making*, Kluwer Academic Publishers, Boston, 2005, pp. 1–17.
- [6] Mutsunori Yagiura, and Shinji Iwasaki, "Toshihide Ibaraki, and Fred Glover; "A Very Large-Scale Neighborhood Search Algorithm for the Multi-Resource to generalized assignment problem".
- [7] H. Feltl, and G.R. Raidl, "An improved hybrid genetic algorithm for the generalized assignment problem", in: *Proceedings of the 2004 ACM Symposium on Applied Computing*, Nicosia, Cyprus, 2004, pp. 990–995.
- [8] M. Yagiura, T. Ibaraki, and F. Glover, "An ejection chain approach for the generalized assignment problem", *Inform Journal of Computing* 16 (2004) 131–151.
- [9] Basturk B., and Karaboga D., "An Artificial Bee Colony (ABC) Algorithm for Numeric Function Optimization", *IEEE Swarm Intelligence Symposium 2007*, Indiana, USA, 2007.
- [10] Hemant Nagpure, Rohit Raja, "The Applications Survey on Bee Colony Optimization". Hemant Nagpure et al, / (IJCSIT) *International Journal of Computer Science and Information Technologies*, Vol. 3 (5), 2012, 5137 – 5140.
- [11] Dervis Karaboga, Bahriye Akay, " algorithms simulating bee swarm intelligence" Springer Science, *Artificial Intelligence Revised*, 2009.
- [12] Pei-Wei TSai, Jeng-Shyang Pan, Bin-Yih Liao, and Shu-Chuan Chu, "Enhanced Artificial Bee Colony Optimization," *International Journal of Innovative Computing, Information and Control* Volume 5, Number 12, December 2009.

IJERT