International Journal of Enhanced Research in Science, Technology & Engineering
ISSN: 2319-7463, Vol. 5 Issue 6, June-2016

A hybrid Bee Colony Algorithm by Tabu Search for Multiple Knapsack Problems

Ruba Talal Ibrahim

Computer Science and Mathematics, Mosul University, Mosul, IRAQ

ABSTRACT

In this paper, an enhanced Hybrid artificial bee colony algorithm (HBCT) is proposed to solve combinatorial optimization problems like 0-1 Multidimensional knapsack problem . The aim of MKP is to find a subset of a given set of *n* objects in such a way that the total profit of the objects included in the subset is maximized, while the total weight of them does not exceed the capacity of the knapsack. Artificial bee colony algorithm is a new swarm intelligence technique inspired by foraging behavior of natural honey bee swarms.The solution search equation of ABC is good in an exploration but is poor inexploitation. So ,in the proposed algorithm first, the harmony search algorithm is used in creating the initial solutions. Thereafter, Tabu search is adopted to avoid falling into local optima.Performance of (HBCT) algorithm on MKP problem instances is compared with the other best heuristic techniques. Computational results show that new (HBCT) algorithm is better in many aspectslike solution quality,convergence which is very rapidly and controls the exploration-exploitation balance.

INTRODUCTION

Multidimensional Knapsack Problem is the classical combinatorial optimization problem and is the expansion of 0-1 knapsack problem, which has been proved to be NP-hard problem. The knapsack problems can be used in many applications for example : scheduling problems, project selection, cutting-stock problems, portfolio optimization and so on[1].The 0–1multipuleknapsack problem can be described as follows :number of items is n, thevalue(profit) of item j is$c_j > 0$, and there are m resources andconstraint $b_i > 0$, the amount of resources consumed i ofterm j is$a_{ij} \geq 0$. Variable x_j is 0 means item j not putting into knapsack,$x_j$is 1 means item j putting into knapsack :

$$\text{Maximize} \sum_{J=1}^{n} c_j\, x_j \qquad \ldots\ldots(1)$$

$$\text{Subject to} \quad \sum_{J=i}^{n} a_{ij}\, x_j \leq b_i\,, \quad i=1,\ldots\ldots,m \quad \ldots\ldots(2)$$

$$X_j \in (0,1)\,, \quad j = 1,2,\ldots\ldots\ldots\ldots\ldots,n \quad \ldots\ldots\ldots(3)$$

Through computing evolutionary methods ,it is necessary to reduce search space in order to let Multidimensional Knapsack Problem get optimalsolution in short time and limit space[2],so the observation on existing literature reveals that a lot of numbers of famous stochastic algorithms have been tested on the MKP.These include artificial fish swarm, accelerated Particle Swarm Optimization and ant colony optimization etc.

In 2015 ,a new simplified binary version of the artificial fish swarm algorithm is presented by Azad, Rocha and Fernandes, where binary string of 0/1 bits is applied on a point/fish. This paper uses crossover and mutation to create trail points in the different fish behavior that are randomly selected by using two user defined probability values[3].

In 2014 ,Zan and Jaros address the possibility of solving the MKP using a GPU accelerated Particle Swarm Optimisation (PSO). The paper aiming to estimate the negotiable performance benefit when using a highly optimized GPU code instead of an efficient multi-core CPU implementation[4].

In 2013,Dalian and Urumqi integrated the Estimation of Distribution Algorithm (EDA) and Particle Swarm Optimization (PSO) to enhance PSO algorithm[2].

In 2013 , Krzysztof Schiff describes in his PHD a new ant colony optimization algorithm for the discrete knapsack problem with a new heuristic pattern, based on the ratio of the square of the profit coefficient to the square of the weight coefficient of the original problem. The aim of new algorithm to choose objects that should be packed into the knapsack.Results of tests under new values of ant algorithm parameter's such as the number of cycles, the number of ants, the evaporation rate, and the load knapsack capacityshow that ant algorithm is better than other algorithms[5].

In 2011, Gong, Zhou and Luo proposed a hybrid artificially glowworm swarm optimization algorithm to solve multidimensional 0-1 knapsack problem[6].

In 2010,Koppaka and Hota produce(HQEA)quantum-inspired evolutionary algorithm, forsolving combinatorial optimization problems. it's combination ofQHW Remote Search and QHW Local Search - the quantum equivalents of classical mutation and local search[7].

In 2009, hybrid algorithm is created by combiningthe fundamental concepts of the PSO algorithm and selected features of evolutionary algorithms ,even though the algorithm is at the very early stage of development[8].

(ABC)algorithm , is a swarm based meta-heuristic algorithm that imitates a foraging behavior of honey bees. It can be observed that the preference on exploration at the cost of exploitation of ABC algorithm. Therefore, to maintain the proper balance between exploration and exploitation behavior of ABC, it is highly required to develop a local search approach in the basic ABC to exploit the search region. Many researches applied ABC algorithm to solve MKP for example , Shyam Sundar and Alok Singh produce an artificial bee colony (ABC) algorithm in such a way that grouping property of QMKP is preserved as far as possible while solving QMKP[9].Srikanth Pulikanti and Alok Singh proposed a new hybrid approach combining artificial bee colony algorithm with a greedy heuristic and a local search for the quadratic knapsack problem ,which is an extension of the 0/1 knapsack problem[10].Shima, Farokhi and Shokouhifar developed a hybrid probabilistic mutation scheme for searching the neighborhood of food sources. The proposed algorithm can guide the search space quickly and improve the local search ability[11].Wei et al. (2011) presented a novel ABC algorithm based on attraction pheromone for the multidimensional knapsack problems[12].

In this paper, a novel hybrid optimization algorithm (HBCT) is proposed to solve the 0–1 Mknapsack problems, which combined between (bee colony) and other search algorithmsto generate diversity in the initial solutions which required in initialization step ,as well as,controls the exploration-exploitation balance. the proposed algorithm can solve the 0–1 knapsack problem effectively, and has better convergence ability and higher computation precision. The performance of the algorithm is improved greatly in comparison with others algorithms.

Artificial Bee Colony (ABC) Algorithm

Swarm Intelligence belongs to Artificial Intelligence based on studyof actions of individuals in many decentralized systems[13].During this decade,various evolutionary algorithms and meta-heuristic has been usedvery much as search and optimization tools in many areas from science to engineering , industry, and others.Many applications that include the solution of optimization problems of high complexity which belonging to a special class of problems called NP-hard have been solved by various methods [13]. Meta-heuristic algorithms are now considered among the best tools must to find good solutions with a reasonable investment of resources.Some classical optimization algorithms characterized by their inflexibility to adapt thesolution algorithm to a given problem. So , more flexible and adaptable general purpose algorithms are needed to overcome above limitations. many swarm optimization algorithms were developed in the literature like genetic algorithms, ant colony ,practical swarm optimization(PSO) and tabu search.

Results has been shown that these algorithms can provide better solutions in comparison to classical algorithms. Artificial Bee Colony (ABC) is a relatively new member ofswarm intelligence, which is based meta-heuristic algorithm that was introduced by Karaboga in 2005[14] for optimizing numerical problems. It was inspired by the intelligent foraging behavior of honey bees. In ABC[14][15] we need to know three important components :food sources,employed and unemployedbees. The position Food source in MPK, represents solution to the optimization problem and the nectar amount of a food source corresponds to the similarity value of the associated solution(fitness value).employed bee search for new food sources within the neighborhood of the food source in their memory. Thereafter, they return to the hive with loads of nectar and with information about the food source, like (distance and direction from the hive). They tell unemployed bees(onlooker bee)about above information by dancing inside the hive. The second type isunemployed foraging bees.

Which consisting of two groups of bees: onlooker bees and scouts. Theonlooker bees wait in the hive and decide a food source to exploit depending on the information shared by theemployed bees. The employed bees whose food sources are abandoned are converted into scout bees. The scout beessearch in the environment for new food sources.A bee waiting on the hive for making decision to choose a food source is called an onlooker and a bee going to the food source visited by it previously is named an employed bee. A bee carrying out random search is called a scout. For every food source, there is only one employed bee. The number of employed bees is equal to the number of food sources around the hive. The employed bee whose food source is exhausted by the employed and onlooker bees becomes a scout.

The Pseudo code of ABC algorithm is given below [16]:

1. Initialize population with *nBee* random solutions.
2. Evaluate fitness of the population.
3. While (stopping criterion not met)

3.1. Select *nSite* sites for neighborhood search.

3.2. Determine the patch size (*ngh*).

3.3. Recruit *nep* bees for selected sites and evaluatefitnesses.

3.4. Select the representative bees from each patch.

3.5. Assign remaining bees to search randomly andevaluate theirfitnesses.

Fig1. The Pseudo code of ABC algorithm

Proposed ABC Algorithm

This study proposes a modified ABC algorithm that benefits from a variety of search strategies to balance exploration and exploitation.

1. Initialization

Initially, the knapsack is empty. The procedure will use Harmony search algorithm[17] to generate a set of feasible solutions(source foods),The size of thisset is same as number of employed bees and onlookers .these solutionsmust satisfy maximum profit while the total weight of them does not exceed the capacity of the knapsack.This process is repeated until it is not possible to add any more objects to the knapsack without violating any of the constraints.

(1) for j=1 to D do D:dimension of problem
(2) if rand(0, 1) $\leqslant$ HMCR then
(3)Choose a harmony x_i from HM randomly, $i \in$ [1..HMS]
(4)$x_j' = x_{i,j}$
(6) if rand(0, 1) $\leqslant$ PAR(t) then
(7)Generate a random integer number $k \in [1..D]$
//$best$ represents the index of the best harmony in the HM
(8)$x_j' = x_{\text{best},k}$
(9)end if
(10)else
(11)$x_j' = L_j + \text{rand}(0, 1) \cdot (U_j - L_j)$ // random selection ……..(4)
(12)end if
(13) end for

Fig2.Harmony search Algorithm

Where (HMS) denotes harmony memory size, (HMCR) denotes harmony memory considering rate and (PAR) denotes pitch adjusting rate . The procedure is done by :

(i) randomly pick a number between 0and 1. If the random number(item) is greater than the harmony memory consideration rate (HMCR), put item in knapsack . However, if the random number(item) is less than or equal to HMCR,pick another random number. Then compare this new random number with the pitch adjusting rate (PAR). If itis less than PAR, pick another random number. Otherwise, apply (4) equation .

(ii) Evaluate the fitness of the first item. Then store the first item in the harmony memory (HM).

(iii) Use the methods in (i) to generate the next item. Then compare the new item to thosein the HM. If the new item is better than the worst item in the HM, replace the worstitem with this new one. Otherwise, add this new item to the HM.

(iv) Repeat (iii) until the number of items in the HM is equal to the population size P.

2. Fitness function selection

According to the Multiple Knapsack problem, we select the fitness function in equation (1) .

3. Neighborhood search

An artificial employed modifies the position (solution) in her memory for finding a new food source by :

$V_{ij}=x_{ij}+\varphi_{ij}(x_{ij}+x_{kj})$ …………...(5)

Wherev_{ij} is a new solution , x_{ij} is an old solution ,$\varphi_{ij}$is a uniformly distributed real random number within the range [-1,1], k is the index of the solution chosen randomly from the colony (k = int(rand ∗SN) + 1), j = 1, . . .,D and D is thedimension of the problem.After producing $\vec{v}_i$, thisnew solution is compared to $\vec{x}_i$ solution and theemployed bee exploits the better source.If the nectar amount of this new food source is higher than that of its currently associated food source, then this employed bee moves to the new food source abandoning the old one, otherwise it continues with the old one. After completion of this process by all employed bees, they start sharing information about their food sources with onlookers.

4. Transition probability and roulette wheel selection

An onlooker bee chooses a food source depending on the transition probability (Roulette Wheel)associated with that food source, p_i, calculated by the following expression:

$P_i=\frac{f(x_i)}{\sum_{k=1}^{s} f(x_k)}$ ………………(6)

Where $f(x_i)$is fitness value of MKP which is equation (1), S is total number of food sources .
If the number of cycles is greater than a predetermined limit, the source is considered to be exhausted. The employed bee associated with the exhausted source becomes a scout and makes a random search inproblem domain.

5. Applying Tabu Search

Tabu search is a heuristic procedure for solving optimization problems, designed to guide other methods to escape the trap of local optimality[18].In this paper we need to enhance ABC algorithm by using Tabu search for avoiding fall in the local search .The proposed algorithm includes Tabu limited list for infeasible solutions. In every cycle ,checking process is required for Tabu list. If the solution is in Tabu list, then employed bee is neglected, otherwise retained .So, employed bee cannot select the solutions in Tabu list . Note : When the numbers of remaining employed bees do not achieve the number of population after passing through above process, then the employed bee with the highest fitness value should be duplicated some times until meeting the requirements .

Algorithm description

First: Initialize the population of solutions using Harmony search .
Second: Evaluate the population by formula (1).
Third : t = 1.
Fourth: Repeat:
Step4.1: Apply Tabu search for the employed bees and update Tabu list, then produce newsolutions $V_i$by formula (5) and evaluate them.
Step4.2: Apply the greedy selection process for the employed bees.
Step4.3: Calculate the transition probability $p_i$for the solutions X_i by formula (6).
Step4.4: Produce the new solutions $V_i$by formula (5) for the onlookers from the solutions $X_i$selected depending on $p_i$by roulette wheel selection and evaluate them.
Step4.5: Apply the greedy selection process for the onlookers.
Step4.6: Determine the abandoned solution for the scout, if it exists, then replace it with anew randomly produced solution X_i and update Tabu list.
Step4.7: Memorize the best solution f_bestachieved so far.
Step4.8: t = t + 1.
Fifth: Until (t >Tmax).
Sixth: Output optimal items in Knapsack

COMPUTATIONAL RESULTS:

In this section, the proposed (HBCT) algorithm, which is discussed in the previous section, is analyzed in detail using several benchmarking problems. The (HBCT) algorithm was coded using the C# language executed on a computer with 6.00GB RAM and an Intel Core5 Duo 2.50 GHz CPU and window 8.To assess the efficiency and performance of (HBCT), the described algorithm is applied on Two parts ofexperiments.In the first part of experiments, we have tested and compared our algorithm on 10 MKP instancesare chosen from paper[19].

The number of onlooker bees was equal to the number of employed bees = 50, and the number of scout bees was selected as five.

In Table1 The population size(colony size) is set to 50-500, while the maximum number of iterations is set to 1000. From Table 1, we can find that CPBA can get the optimal solution for all the ten problems. For other two algorithms, the performance of discreteABC(see Table 2)[20] is the best for four cases (Kp4,Kp6,Kp8,Kp10). And the PSO(see Table 3)[21] can also get the optimal solution for three cases(Kp4,Kp6,Kp10). So the optimization performance of (HBCT) is better than other methods and more stability.

Table1: Comparison results of each algorithm for 0–1 knapsack problems Kp1 to Kp10

Problem Name	No. Items	Algorithm	OptimalSolution
Kp1	10	PSO ABC HBCT	280 290 **294**
Kp2	20	PSO ABC HBCT	1000 1015 **1019**
Kp3	4	PSO ABC HBCT	30 35 **37**
Kp4	4	PSO ABC HBCT	**20** **20** **20**
Kp5	15	PSO ABC HBCT	450 455 **463**
Kp6	10	PSO ABC HBCT	**50** **50** **50**
Kp7	7	PSO ABC HBCT	99 104 **105**
Kp8	23	PSO ABC HBCT	9600 **9750** **9750**
Kp9	5	PSO ABC HBCT	125 127 129
Kp10	20	PSO ABC HBCT	**1023** **1023** **1023**

TABLE 2. PARAMETER SETTING FOR Discrete ABC

Parameters	Value
the colony size of	50–100
employed bees	50
onlooker bees	50
scout bees	1

volume	1,……99
value	100,……999

TABLE 3. Parameter Setting For PSO

PARAMETERS	value
Iteration max	500
No. of Particles	50
C1	
C2	2
W_max	1
W_min	-1
V_max	
PrimitiveVelocity of each Particle	0

In the second part of experiments, we have tested the HBCT algorithm on some big size MKP instances taken from OR-Library[22] named mknapcb1 and mknapcb4. We have used 5 tests of the benchmarks mknapcb1 (5.100) which have 5 constraints and 100 items, and we have used 5 tests of the benchmarks mknapcb4 (10.100) which have 10 constraints and 100 items. Our algorithm is executed 30 times on each large instance with a different random seed. The (HBCT) is executed for 5000 iterations on each instance of benchmark set .

TABLE 4. Experimental Results of MKP with mknapcb1 and mknapcb4 instances

Benchmark Name	Problem size	HBCT
mknapcb1	5.100.00	25220
	5.100.01	25200
	5.100.02	25340
	5.100.03	24888
	5.100.04	23999
mknapcb4	10.100.00	23911
	10.100.01	23756
	10.100.02	23234
	10.100.03	23001
	10.100.04	22988

Table 4 shows the experimental results of our HBCT with some hard instances of mknapcb1 and mknapcb4. Obtained results in table 4 appear that HBCT algorithm can find optimal solution effectively.

CONCLUSION

In this paper a new meta-stochastic method was applied for the 0-1Mknapsack problem. The(HBCT)in comparison with the above mentioned algorithms has a great improvement in the quality of solution, stability. In addition, it has been found that algorithm is able to find a good solution withbig size MKP instances and provide a good balance between exploration and exploitation.Simulation results show that (HBCT) has better performance than other algorithms .

REFERENCES

[1]. Kellerer H, Pferschy U, Pisinger D (2005) Knapsack problems. Springer, Berlin .

[2]. LIU Wen" Hybrid Particle Swarm Algorithm for Solving Multidimensional Knapsack Problem" Przegląd Elektrotechniczny, ISSN 0033-2097, R. 89 NR 1b/2013 .

[3]. Azad, Rocha ,Fernandes," Solving Large 0-1 Multidimensional Knapsack Problems by a New Simplified Binary Artificial Fish Swarm Algorithm "Journal of mathematical Modelling and Algorithms in Operations ResearchAugust 2015, Volume 14, Issue 3, pp 313-330 .

[4]. Drahoslav Zan and Jiri Jaros," Solving the Multidimensional Knapsack Problem using a CUDA accelerated PSO ",2014 IEEE Congress on Evolutionary Computation (CEC) July 6-11, 2014, Beijing, China .

[5]. KRZYSZTOF SCHIFF," ANT COLONY OPTIMIZATION ALGORITHMFOR THE 0-1 KNAPSACK PROBLEM" ,Ph. D. Krzysztof Schiff, e-mail: kschiff@pk.edu.pl, Department of Automatic Control and Information Technology, Faculty of Electrical and Computer Engineering, Cracow University of Technology.

[6]. Gong, Zhou , Luo," Hybrid Artificial Glowworm Swarm Optimization Algorithm for Solving Multi-dimensional Knapsack Problem" Science Direct , Elsevier Procedia Engineering 15 (2011) 2880 – 2884 .

[7]. Mahdabi, Abadi and Jalili," A Novel Quantum-Inspired Evolutionary Algorithm for Solving Combinatorial Optimization Problems ", *GECCO'09,* July 8–12, 2009, Montréal Québec, Canada. ACM 978-1-60558-325-9/09/07 .

[8]. Nam Fai Wan and Lars Nolle," SOLVING A MULTI-DIMENSIONAL KNAPSACK PROBLEM USING A HYBRID PARTICLE SWARM OPTIMIZATION ALGORITHM ",2009 , Proceedings 23rd European Conference on Modelling and Simulation ©ECMS Javier Otamendi, Andrzej Bargiela, José Luis Montes, Luis Miguel Doncel Pedrera (Editors).

[9]. Shyam Sundar and Alok Singh,"A Swarm Intelligence Approach to the Quadratic Multiple Knapsack Problem", Springer-Verlag Berlin Heidelberg, 2010 .

[10]. Srikanth Pulikanti and Alok Singh," An Artificial Bee Colony Algorithm for the Quadratic Knapsack Problem", Springer-Verlag Berlin Heidelberg 2009.

[11]. Shima, Farokhi and Shokouhifar,"A Novel Artificial Bee Colony Algorithm for the Knapsack Problem ", 978-1-4673-1448-0/12/$31.00 ©2012 IEEE .

[12]. Wei H, Ji J, Qin Y, Wang Y, Liu C (2011) A novel artificial bee colony algorithm based on attraction pheromone for the multidimensional knapsack problems. In: Deng H, Miao D, Lei J, Wang F (eds) Artificial intelligence and computational intelligence. Lecture notes in computer science, vol 7003. Springer, Berlin, pp 1–10 .

[13]. Marco Aurelio Sotelo-Figueroa, Rosario Baltazar, and Martín Carpio," Application of the Bee Swarm Optimization BSO to the Knapsack Problem", Springer-Verlag Berlin Heidelberg 2010, Soft Comp. for Recogn. Based on Biometrics, SCI 312, pp. 191–206.

[14]. D. Karaboga and B. Basturk, On the performance of artificial bee colony (ABC) algorithm, Applied Soft Computing, 8 (2008), 687-697 .

[15]. Thammano and Phu-ang," A Hybrid Artificial Bee Colony Algorithm with Local Search for Flexible Job-Shop Scheduling Problem " 2013 The Authors. Published by Elsevier B.V, Science Direct .

[16]. Papova Nhina Nhicolaievna, Le Van Thanh," Bee Colony Algorithm for the Multidimensional Knapsack Problem ", Proceedings of the International MultiConference of Engineers and Computer Scientists 2008 Vol I ,IMECS 2008, 19-21 March, 2008, Hong Kong .

[17]. K. S. Lee and Z. W. Geem, A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice, Computer Methods in Applied Mechanics and Engineering, 194 (2005), 3902-3933 .

[18]. D. Karaboga, B. Akay, Artificial Bee Colony (ABC), Harmony Search and Bees Algorithms on Numerical Optimization, IPROMS 2009 Innovative Production Machines and Systems Virtual Conference, Cardiff, UK.

[19]. Zou D, Gao L, Li S, Wu J (2011) Solving 0–1 knapsack problem by a novel global harmony search algorithm. Appl Soft Comput 11:1556–1564 .

[20]. Xin-hong WEI , Kai ZHANG,"" Discrete Artificial Bee Colony Algorithm for Multiple Knapsack Problems, International Journal of Advancements in Computing Technology(IJACT) Volume4, Number21,November 2012 .

[21]. Z. Li, N. Li, "A Novel Multi-Mutation Binary Particle Swarm Optimization for 0-1 Knapsack Problem", IEEE Chinese Control and Decision Conference (CCDC), pp. 3042-3047, 2009.

[22]. ORLibrary,J.E.Beasley,http://www.people.brunel.ac.uk./mastjjb/jeb/orlib/mknapinfo.html.