Hybrid Algorithm Based on HBMO and GRASP For Real-time Task Scheduling Problem Resolution

Yahyaoui Khadidja[1] , Debbat Fatima[2], Khelfi Mohamed Fayçal[3]

[1] **Department of Mathematics and Computing Science**
Faculty of Science and Technology University of Mascara, Algeria

[2] **Department of Mathematics and Computing Science**
Faculty of Science and Technology University of Mascara, Algeria

[3] **Laboratory of Researches in Industrial Computing and Networks**
Faculty of Sciences University of Oran Es-Senia, Algeria

Abstract

The problem of real-time scheduling is an NP-Hard problem where each task is characterized by temporal, preemptive and static periodicity constraints. It is therefore necessary to use metaheuristic methods to solve it. This research proposes a hybrid heuristic approach for further improving the quality of solutions. This approach applies Honey Bees Mating Optimization (HBMO) algorithm in combination with one of the best heuristics used to solve this problem Greedy Random Adaptive Search Procedure (GRASP). A key element in the success of this hybrid approach is the use simultaneously an intensified research on several areas of research space. The implementation of these algorithms has been subject to extensive tests. Several experiments are carried out on different problem instances. Results obtained show the advantages and efficiency of our approach.

***Keywords**: Real-time task scheduling, Optimization, Meta-heuristic, HBMO, GRASP*.

1. Introduction

Optimizing one objective over a set of constraints has been widely studied for many combinatorial optimization problems including scheduling problems. There exists a host of problems related to scheduling. The fundamental of this problem is, informally, to determine when and where to execute tasks of a program on a target computing platform. Scheduling is a hard problem that has been extensively studied in many aspects, theoretical as well as practical ones.

Real-time systems play an important role in our modern societies. A vast amount of work has been done in the area of real-time scheduling by both operations research and computer science communities. In real-time systems, all tasks are characterized by specific parameters such as Computing Time (c_i), Deadline (d_i), Priority (p_i), etc. In a given real-time system, the goal of a good scheduler is to schedule the system's tasks on a processor, so that every task is completed before the expiration of the task deadline (di) [1]. In order to obtain a task scheduling taking into account all imposed constraints, namely static periodicity, temporal and preemptive constraints, we have used a new approach based on the hybridization of several algorithms. It combines an HBMO algorithm, and the Greedy Randomized Adaptive Search Procedure (GRASP).

Our contribution consists in generating the HBMO initial population with GRASP and combining the global search represented by HBMO with some local search for the resolution of real-time task scheduling problems in order to intensify the search in promising zones detected by the HBMO exploration process. This hybridization is realized with methods which have been applied in isolation in the resolution of real-time task scheduling problems: TS, SA and GRASP [2][7].

This paper is organized in the following way. The related works are presented in the second section. The presentation and formulation of real-time tasks scheduling problems is described in Section 3. Section 4 presents the proposed Scheduling Approach. The resolution process of the scheduling real-time tasks problem is detailed in Section 5. The sixth section presents the simulation and experimental results arrived at. The paper finishes with the conclusion and recommendations for future research.

2. Related works

The task scheduling problem is defined as one of the many popular academic NP-hard problems. The use of metaheuristic methods shows their efficiency and effectiveness to solve these categories of complex problems [8]. In the literature, many metaheuristics have been proposed based on methods and approaches to task scheduling. The authors in [9][8] resolve the scheduling task problem through hybridization of Particle Swarm Optimization (PSO) with GA. In [9], E. G. Talbi et al. have hybridized Simulated Annealing (SA) with Genetic Algorithms (GA). In [10][11], the authors applied Bees

Colony Optimization (BCO) as a heuristic algorithm to solve the problem of static scheduling of independent tasks on identical machines. This strategy is based on the intelligent behavior of honey bees in the foraging process. Koudil et al. adapted the MBO algorithm to solve integrated partitioning/scheduling problems in co-design in [12]. This algorithm gives good results in terms of solution quality and execution time.

The HBMO algorithm was proposed by Abbass in [13]. Since then it has been used on a number of different applications such as multi-robot path-planning [14], large scale vehicle routing problems [15] , Multi-Objective distribution feeder reconfiguration problem [16], Examination Timetabling Problems, [17], the Load Frequency Control (LFC) problem in a restructured power system [18]. HBMO is also used as a new approach to estimate the state variables in distribution networks including distributed generators [19].

3. Real-time task scheduling problem formulation

In a real-time system, the goal of the scheduler is to schedule the system's tasks on a processor, so that every task is completed before the expiration of the deadline (***di***).

In this paper, a task is defined by a set of temporal constraints : computing time (***ci***), deadline (***di***), and priority (***pi***), as well as preemption and static periodicity constraints. Thus the main objective is to schedule a set of tasks : $i = \{1,2 \ldots n\}$, taking into consideration all the afore-mentioned imposed constraints. The objective function of our optimization problem is the aggregation of the following two objective functions:

- **The first objective function** f_{-1}: Minimization of not respected tasks constraints number (The sum of tasks priorities which are not able to be execute in time).
- **The second objective functio** f_2 : Minimization of the project delay.

$$f\_1(t+1) = \begin{cases} f_{-1}(t) + pi \quad if \quad Dproject > di \\ f\_1(t) \; otherwise \end{cases} \qquad (1)$$

$$f\_2(t+1) = \begin{cases} f\_2(t) + (Dproject - di) \\ \quad if \; Dproject > di \\ f\_2(t) \quad otherwise \end{cases} \qquad (2)$$

The global cost function is:

$$f(t) = f\_1(t) + f\_2(t) \qquad (3)$$

It yields

$$f(t+1) = \begin{cases} f(t) + pi + (Dproject - di) \\ \quad if \; Dproject > di \\ f(t+1) \; otherwise \end{cases} \qquad (4)$$

The following algorithm explains the process of the different function evaluations

```
Début
D_projet = 0 ; // Delay of project.
 f = 0, f_1 = 0, f_2 = 0 ;
Nb  = number of tasks ;
For  i = 0 To  Nb do
  If  (D project > di) then begin
                    f_1 = pi ;
                    f_2 = D_project – di ;
                    f = f_1 + f_2 ;
                    End
   Endif;
Endfor;
End.
```

4. Proposed scheduling approach

The proposed approach combines the Honey Bees Mating Optimization (HBMO) algorithm with the Greedy Randomized Adaptive Search Procedure (GRASP), and local searches: Simulated Annealing (SA) and Tabu Search (TS).

HBMO was found to outperform some better known algorithms. However, it has not been applied to real-time scheduling problems. A honey bees colony consists of the queen(s), drones, worker(s) and broods.

The Honey bees Mating Optimization algorithm mimics the natural mating behavior of the queen bee when she leaves the hive to mate with drones in the air. After each mating, the genetic pool of the queen is enhanced by adding sperm to her spermatheca. One or more heuristic workers are introduced to the original HBMO to improve the queen's genetic constitution. Before the mating flight begins, the queen is initialized with a certain amount of energy and only ends her mating flight when the energy level drops below a threshold level (which is close to zero) or when her spermatheca is full [13]. The probability of a drone mating with a queen obeys the following annealing function:

$$Prob(D) = e^{[-\Delta(f)/speed(t)]} \qquad (5)$$

where *Prob (D)* is the probability of adding the sperm of drone *D* to the spermatheca of the queen, that is, the probability of a successful mating, *Δ(f)* is the absolute difference between the fitness of *D* and the fitness of the queen and *Speed(t)* is the speed of the queen at a given time *t*. After each flight, the queen's speed and energy evolve according to the following equations:

$$Speed(t+1) = \propto \times \; Speed(t) \qquad (6)$$

$$Energy(t+1) = \propto \times Energy(t) \qquad (7)$$

where factor $\alpha \in (0,1)$ is the amount of speed and energy reduction after each flight. The workers are presented as heuristics whose functionality is to improve the broods produced during the mating process. To diversify the initial HBMO population and select the best solution as the queen, GRASP method is used.

To improve the broods produced during the queen's mating, we use two workers based on local search metaheuristics: Tabu Search (TS) and Simulated Annealing (SA).

5. Presentation of the resolution process

In this section, we present the approach adapted to resolve the issues addressed. To start with, a set of parameters must be defined:

- The bees population represents the set of scheduling plans of real-time tasks with temporal constraints,
- The queen represents the tasks best scheduling plan in the population generated by GRASP, randomly or hybridization of a percentage from each of the two populations obtained using the two aforementioned procedures and gradually improved by the implementation of the neighborhood generations iterative procedures (crossover in the HBMO) and two workers SA and TS.
- The drones make up the remaining task scheduling plans.

Figure 1 represents the adopted approach.

```
Initialization
Generate the initial population of the bees using
GRASP, Random or through Hybridization
Evaluate the fitness of the plan;
Select the best plan of the population of plans which represents
the queen;
Sperm: Size of the spermatheca;
E (t) and S (t): Energy and Speed in [0.5, 1];
α : factor of reduction of energy and speed in [0,1];
M: maximum number of mating flights;
For i=0 to M (mating flights)
    do while E(t) > 0 and Sperm is not full
        Select a drone
        if the drone passes the probabilistic condition
            Add sperm (plan) of the drone in the spermatheca
        Endif
    S(t+1) = α × S(t)
    E(t+1) = α × E(t)
    Enddo
    For j = 1 to Size of Spermatheca
    Select a sperm from the spermatheca;
    Generate a brood by crossover the queen's genotype with
the selected sperm;
    Improve the brood's fitness by applying the workers
(Tabu Search /Simulated Annealing)
    if the brood's fitness is better than the queen's fitness
  Replace the queen with the brood
        else Add the brood to the population of drones
        Endif
Replace the drones by the broods
    Enddo ( for)
 Enddo
 return The Queen (Best Solution Found)
```

Fig. 1 The adopted approach.

In the following part of this section, we explain the initial population generation and broods improvement stages.

5.1 Stage of initial population generation

In these following paragraphs, we present the process of generating initial population based on GRASP approach and hybridization. The process of hybridization is used for building the initial population of the HBMO algorithm. It combines several methods such as GRASP and the random procedure to generate the initial HBMO populations and we take a percentage of the population result from each procedure.

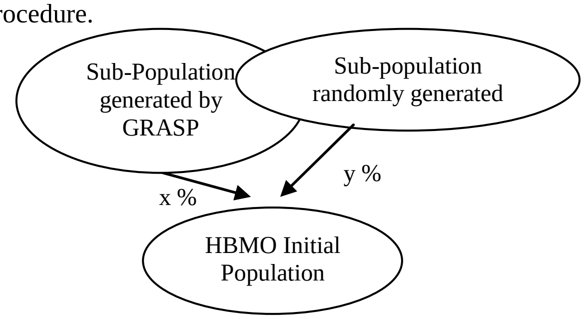

Fig. 2 Initial population generated by hybridization

The GRASP method is a multi-start or iterative process (Lin and Kernighan, 1973), in which each GRASP iteration consists of two phases, a construction phase, in which a feasible solution is produced, and a local search phase, in which a local optimum in the neighborhood of the constructed solution is sought. GRASP has proved its efficiency in research and computer science applications as well as in industrial applications. These include graph theory, quadratic and other assignment problems, location, layout, cutting, covering, clustering, packing, partitioning, routing, sequencing and scheduling, logic, manufacturing, transportation, telecommunications, electrical power systems, biology, VLSI design, drawing, and miscellaneous topics [5].

- ***Process of initial population Generation By GRASP Method***

```
Begin
    RCL: List of tasks Ti (n_i, ci, d_i, p_i)
    Choose one element from RLC to build the solution randomly.
    RCL ← GenerateTasks (s) //generate all the tasks Ti to be scheduled
    build Solution ()
    { s ← ∅;
    While s not completed {
    x← Choose Randomly (RCL) //not the tasks already in s
    s ← s U {x};}
     Return s
        } ;
     s*← build Solution ();
     Repeat
        s' ← Build Solution ();
        Ameliorate (s')// by local search
     If f(s') < f(s*) Then begin
                s* ← s';
                Insert s' in the population;
                 End;
     Endif
  Until the size of the population
  End.
```

The next point explains the improvement strategies for the brood solutions by using two heuristic methods (SA and TS).

5.2 Stage of broods improvement

The improvement process is realized by two methods: SA and TS as follows:

SA is based on the annealing of metals. The Metropolis algorithm, also known as the Metropolis rule of probability, is used to simulate annealing through a series of moves. During each move, the system has some probability of changing its current configuration to a worse one. SA is a generic algorithm which tends not to fall in a local minimum or maximum [6].

TS is a strategy for solving combinatorial optimization problems with applications in graph theory. It is an adaptive search procedure to combine with other methods, such as linear programming algorithms and specialized heuristics, to overcome the limitations of local optimality [7].

- ***Improvement process By Tabu Search (TS)***

```
Begin
    Initial solution s (brood which will be improved);
     Insert s in the tabu List ;
       Smax ← s. / /Smax : best solution
         While (Criterion of stop not checked) do
          Generate the neighborhood of the current solution by
           mutation/lag/a combination of the two operations.
            Select s' in this neighborhood although s' is not
    present in
           the tabu list.
               If F (s') < F (Smax) then begin Smax ← s'
                                  //minimize the cost function
                                Update the tabu list End;
               Endif;
           End of while;
         End.
```

- ***Improvement process By Simulated Annealing (SA)***

```
Begin
    Initial solution s (brood which will be improved);
       Put T ← f(s); // f (s) the fitness
       //do one mutation or two mutations or
       One mutation and one lag on the brood
       s'← Mute (s);
      s'← offset (s');
      If (f(s') < f(s)) then return s'
                  Else
        Generate a real number randomly r in [0, 1]
               If r < e^((f(s)-f(s'))/T) then return s'
                                    Else return s;
                  Endif;
       Endif ;
    End.
```

6. Simulation and experimental results

This part is devoted to the implementation and testing of the algorithmic methods developed above. JCreator version 4 has been used to implement the approach and jfreechart: version 1.0.13 for the realization of the graphs.

6.1 Adopted Approach Parameters

The parameters of the algorithm have been selected after thorough testing. A number of different values were tested and the ones selected are those that yielded the best results in both solution quality and computational time. The best values for these parameters appear in the following tables:

Table 1: Parameters and their values for the approach adopted

Algorithms	Parameters	Values
HBMO	Size of population	50
	Mating flight	100
	Spermatheca	6
	Speed	0.80
	Energy	0.70
	Alpha (α)	0.20
Simulated annealing (SA)	Initial Temperature	Fitness of brood
	Final Temperature	0
	Number of iterations	Until the amelioration or temperature=0
Greedy Randomized Adaptive Search Procedure (GRASP)	Size of the population	50
	Algorithm of amelioration	Local seach
Tabu Search (TS)	Dimension of the list	5
	Number of iterations	5
	Type of neighborhood	With mutation and lag

6.2 Simulation Results

In order to assess the effectiveness of the approach under study, it was applied to three cases: with 40 tasks, 60 tasks and 80 tasks, each task having its own temporal characteristics such as computing time (***ci***), deadline (***di***), and priority (***pi***).
In this section, we generate the initial population of the proposed algorithm randomly. After a series of executions, the best values for fitness and optimal execution time are summarized below:

Table2. Simulation results for objective function and computing time The table values show that in all three cases (40 tasks, 60 tasks and 80 tasks),

	HBMO-random	
	objective Function	Computing time (ms)
40 tasks	1815	1780
60 tasks	6382	1867
80 tasks	11799	2760

The following tables summarize the simulation results thus obtained.

Table 3: Objective Function values obtained

	HBMO-random	HBMO-GRASP	HBMO-HYBRIDIZATION
40 tasks	1815	1777	1755
60 tasks	6382	6448	6299
80 tasks	11799	11816	11539

Table 4: Execution time values obtained

	HBMO-random (ms)	HBMO-GRASP (ms)	HBMO-HYBRIDIZATION
40 tasks	1780	1658	1378
60 tasks	1867	1264	2306
80 tasks	2760	2091	2672

The best real-time task scheduling approach is to be selected taking into consideration the objective function and execution time criteria. We notice that HBMO-HYBRIDIZATION is better than the other two approaches regarding the objective function values.
In the area of execution time however, we notice that HBMO-HYBRIDIZATION is better than the other two approaches when the number of tasks is ≤ 40, while HBMO-GRASP yields very better results in the cases where the number of tasks exceeds 40.

To make a decision as to which approach is best, we calculate coefficient ω which represents the objective function standard deviation between the hybridization methods adopted.
The following table calculates ω in terms of the relative deviation from the optimum, that is

$$\omega = \frac{100*(C_{HBMO-GRASP} - C_{HBMO-HYBRIDIZATION})}{C_{HBMO-HYBRIDIZATION}},$$

Where $C_{HBMO\text{-}GRASP}$, $C_{HBMO\text{-}HYBRIDIZATION}$ are respectively the cost of the optimal solution obtained by two

hybridization processes namely HBMO-GRASP and HBMO-HYBRIDIZATION.
Tables 5 indicates the different relative deviation values of ω for all three instances.

Table 5: ω relative deviation values in different instances

	ω	Observations
40 tasks	0.0125	HYBRIDIZATION slightly better GRASP
60 tasks	0.0236	HYBRIDIZATION slightly better than GRASP
80 tasks	0.0240	HYBRIDIZATION slightly better than GRASP

As illustrated in the tables above, in all these complex cases involving a great number of tasks, HBMO-HYBRIDIZATION gives the best results.

7. CONCLUSION

In this paper, an approach is proposed to solve real-time task scheduling problems. This approach is based on the hybridization of several algorithms inspired from nature such as HBMO, Tabu Search and Simulated Annealing and GRASP algorithms. In the stage of generating initial solutions, the adopted approach uses random selection alone, GRASP alone, and hybridization of the two processes selecting a percentage of the population from each. In the stage of solution improvement by the workers, HBMO uses Tabu Search and Simulated Annealing algorithms. The proposed approach was tested on three real-time task scheduling instances with 40 tasks, 60 tasks and 80 tasks where each task has its own temporal constraints.
The relative deviation from the optimal solution is used as the standard to evaluate the quality of the solutions obtained. Results confirm the positive impact of using a hybrid strategy with regard to objective function quality and computing time in comparison with the hybridization algorithms object of this paper.
The results arrived at show that for complex real-time scheduling problems (with a number of tasks exceeding 40), the best approach is HBMO-HYBRIDIZATION. If the time factor is critical in scheduling, the best approach is HBMO-GRASP which compromises between cost, run-time and algorithm complexity.
In the future, we envisage the study of energy consumption which is a very significant constraint in real-time systems. We will also test these approaches on other NP-complete problems such as vehicle routing problems.

References

[1] M. Hamzeh, S. M, Fakhraie and C. Lucas, "Soft Real-Time Fuzzy Task Scheduling for Multiprocessor Systems", world Academy of science, Engineering and Technology vol. 28, 2007.

[2] A. Thesen, "Design and evaluation of tabu search algorithms formultiprocessor scheduling", Journal of Heuristics, vol. 4, No. 2, pp.141–160, 1998.

[3] Fleischer and Mark. "Simulated Annealing: Past, Present, and Future", Proceedings of the Winter Simulation Conference, Department of Engineering Management, Old Dominion University, Norfolk, VA. 1995

[4] S. Casey and J. Thompson. "GRASP the examination scheduling problem". In E. Burke and P. D. Causmaecker (Eds.), Volume 2740 of Lecture Notes in Computer Science, pp. 232–244. Springer, PATAT 2002.

[5] P. Festa, G. Mauricio and C. Resende, "effective application of grasp ", AT&T Labs Research Technical Report. 2009.

[6] Fleischer and Mark, " Simulated Annealing: Past, Present, and Future", Proceedings of the Winter Simulation Conference, Department of Engineering Management, Old Dominion University, Norfolk, VA, 1995.

[7] F. Glover, "Tabu Search part I", ORSA Journal on Computing, Vol. 1, No. 3,1989.

[8] E. G. Talbi, "Metaheuristics: From Design to Implementation", Wiley & Sons, Hoboken, New Jersey, 2009.

[9] E. G. Talbi and T. Muntean. "Hill-climbing, simulated annealing and genetic algorithms : a comparative study" . IEEE Computer Society Press. In International Conference on System Sciences HICSS: Task scheduling in parallel and distributed systems, Hawaii, pp. 565–573, Jan 1993.

[10] T. Davidovic, D.Ramljak, M.Selmic and D.Teodorovic , "Bees colony optimization for the p-center problem", Computers & Operations Research Vol. 38, pp. 1367–1376, 2011.

[11] T. Davidovic, M. Selmic and D.Teodorovic, "Scheduling Independent Tasks: Bees Colony Optimization Approach", 17th Mediterranean Conference on Control & Automation Makedonia Palace, Thessaloniki,
Greece June, 2009.

[12] M. Koudil, K. Benatchba, A. Tarabet and E.B, Sahraoui, "Using Artificial Bees to Solve Partitioning and Scheduling Problems in Codesign", Applied Mathematics and Computation, vol. 186, No. 2, pp. 1710-1722, 2007.

[13] H.A Abbass, "A Single Queen Single Worker Honey–Bees Approach to 3-SAT", The genetic and Evolutionary computation conference, GECCO, san Francisco, USA, 2001

[14] R.R. Sahoo, P.RakshitMd and T. Haidar, "Navigational Path Planning of Multi-Robot using Honey Bees Mating Optimization Algorithm (HBMO)", International Journal of Computer Applications (0975 – 8887), Vol. 27 ,No. 11, pp. 0975-8887, August 2011.

[15] Y. Marinakis, M. Marinaki and G. Dounias, "Honey Bees Mating Optimization algorithm for large scale vehicle routing problems", Nat Comput, vol. 9, pp. 5–27, 2010.

[16]T. Niknam ,"Multi-Objective distribution feeder reconfiguration problem In the radially distribution systems", Journal Expert Systems with Applications: An International Journal archive, vol. 38, No.3, March, 2011.

[17] N. Sabar, M. Ayob and G.Kendall, "Solving Examination Timetabling Problems using Honey bees Mating Optimization (ETP-HBMO) ", Multidisciplinary International Conference on Scheduling : Theory and Applications (MISTA), Dublin Ireland, 2009.

[18] O. Abedinia, M. Naderi and A. Ghasemi, "Robust LFC in deregulated environment: Fuzzy PID using HBMO", 10th International Conference Environment and Electronical Engineering (EEEIC), 2011.

[19] T. Niknam, J. Olamaie and Reza Khorshidi, "A Hybrid Algorithm Based on HBMO and Fuzzy Set for Multi-Objective Distribution Feeder Reconfiguration", World Applied Sciences Journal, vol. 4,No. 2, pp. 308-315, 2008.

Biographies

Yahyaoui Khadidja received the engineer degree in soft engineering and the M.Tech. degree in industrial computing from Oran university computer science department , Algeria, in 2000 and 2006, respectively. She is currently working toward the Ph.D. degree in Oran University, Algeria. She is currently a Research member at Mascara University Institute of sciences and Technology, Algeria. Her research interests include real time systems, scheduling and optimization.

Debbat Fatima received the engineer degree in instrumentation from the sciences and Technology university , Oran , Algeria in 1999 and the M.Tech. degree in telecommunication from spatial technical national center Arzew, Algeria in 2002. She received PhD degree in telecommunication in 2007 from Tlemcen University, Algeria. She is currently a Research member of the Telecommunication laboratory at Tlemcen University. Her main research interests include optimization, artificial intelligence, and telecommunication.

Khelfi Mohamed Fayçal Received Ph.D. degree in Automatic Control from Nancy University, France, in 1995. He is currently Professor at the Computer Science Department - Faculty of Sciences - University of Oran - Algeria. He is also a research member at the Laboratory of Research in Industrial Computing and Networks. His main research interests include Automatic Control, Industrial Computing, Robotics and Networks.