International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-6, Issue-6, June 2018 E-ISSN: 2347-2693

Multiprocessor Scheduling using Krill Herd Algorithm (KHA)

S.K. Nayak[1*], C.S. Panda[2], S.K. Padhy [3]

[1] Computer Science and Engineering, Centurion University of Technology and Management, Bhubaneswar, India
[2] Department of Computer Science and Application, Sambalpur University, Burla, India
[3] Department of MCA, Veer Surendra Sai University of Technology, Burla, India

[*]*Corresponding Author: nayaksasmita484@gmail.com, Tel.: 08895396822*

Available online at: www.ijcseonline.org

Accepted: 03/June/2018, Published: 30/June/2018

Abstract— This paper manages the issue of Multiprocessor scheduling Problem is one of the most challenging problems in distributed computing system. Many researchers solved the multiprocessor scheduling problem as static. But in this paper uses the dynamic multiprocessor scheduling problem which is an advanced area. Dynamic allocation strategies can be connected to huge arrangements of genuine applications that can be planned in a way that takes into account deterministic execution. In the first place, here defines the Dynamic Multiprocessor scheduling, which is an optimization problem, after that it optimizes the execution time of various tasks assigned to the processors with a Krill Herd Algorithm (KHA). In recent times, a robust meta-heuristic optimization algorithm, known as Krill Herd, which is used for global optimization to enhance the execution of the multiprocessor scheduling problem but other traditional algorithms stuck in local optimization. In this paper with the end goal of comparison, contemporary methodologies utilizing Genetic Algorithm (GA), Bacteria Foraging Optimization (BFO) and Genetic based Bacteria Foraging (GBF) found in the literature. Here, it demonstrates the better performance of Krill Herd Algorithm with the above mentioned methods by simulation process.

Keywords— Multiprocessor scheduling, Optimization problem, Krill Herd Algorithm (KHA)

I. INTRODUCTION

The Multiprocessor scheduling problem is an NP-hard problem [1,2,3,4]. In this paper, we exhibit another task assignment algorithm that is based on the Krill Herd Algorithm (KHA). The Multiprocessor scheduling problem can also be called as task scheduling algorithm. It is categorized into two problems: Static and Dynamic. Here we have considered the dynamic multiprocessor scheduling problem. The static multiprocessor scheduling knows the information about the execution time at compile time whereas in dynamic multiprocessor scheduling knows all the information at runtime only. Some conventional algorithms from literature optimize makespan and some optimizes flow time, whereas this paper optimizes total execution time. Compared with the traditional methods of multiprocessor scheduling, another method with an optimized algorithm based on KHA is proposed in this paper. This paper discovers the impacts of processor limit and task set on execution time of computation through simulation. A simulations result shows the minimum execution time using KHA.

The rest of the paper is sorted out as takes after, Section I contains the introduction to multiprocessor scheduling, the related work and background of problem statement illustrates in Section II. The working of KHA i.e. methodology is described in Section III, Section IV depicts the results and discussion i.e. simulation and the conclusion, future work is managed in Section V.

II. RELATED WORK

A. Related work

In traditional methods utilized as a part of optimization are deterministic, fast, and provide accurate results yet regularly have a tendency to get fixed on local optima [5]. The first evolutionary-based method was the genetic algorithms (GAs) [6]. GAs was created taking into account the 'survival of the fittest' which is a Darwinian principle and the natural process of evolution through reproduction. In light of its showed capacity to get near-optimum solutions from large problems, the technique of Genetic Algorithms has been utilized as a part of numerous applications in science and engineering [7,8]. In spite of their advantages, Genetic Algorithms may oblige time-consuming process for a near optimum value for developing. Additionally, all issues cannot give themselves good to a result with Genetic Algorithms [9]. However, Krill herd algorithm is utilized to diminish processing time and enhance the value of results, especially to abstain from being caught in local optima. After GA, many other nature-inspired meta-heuristic algorithms have appeared, for example, differential evolution (DE) [10,11,12], particle swarm optimization (PSO) [13,14,15], genetic programming (GP) [16,17], biogeography-based optimization (BBO) [18,19], bat algorithm (BA) [20,21], cuckoo search (CS) algorithm [22,23,24], firefly algorithm (FA) [25,26,27,28], and all the more as of late, in nature the krill herd (KH) algorithm [29] is in view of simulating the grouping or herding activities of krill individuals.

The ACOSS (Ant colony optimization (ACO) - scatter search (SS) algorithm), is a local search approach utilized to get the improved result for resource- constrained multiprocessor scheduling problem [1,30]. However, no SS algorithm is utilized to get the improved solution of the multiprocessor scheduling problem in the proposed algorithm.

In PSO an individual called as particle. It is based on special management of memory to optimize the objective function by iteratively enhancing a swarm of solution vectors. Every particle is adjusted by alluding to the memory of individual and best data of swarm's [31]. However, our proposed algorithm KHA optimizes the execution time of the task assignment problem instead of the management of memory in PSO.

B. Problem statement

This paper deliberates the allocation of task to the different processor with the accompanying situation. The framework comprises of an arrangement of tasks (B) and different processors (A) having distinctive memory and resources performed on diverse processor experiences distinctive execution time [1]. The communication links are thought to be indistinguishable, but communication cost among two tasks will be experienced when executed on diverse processors. A task will make utilization of the resources from its execution processor. The goal is to get the minimum total execution time came across by assigning of tasks. This area talks about the proposed dynamic multiprocessor scheduling using KHA. Table 1 demonstrates a descriptive example contains five tasks and four processors [1]. Each row and column represents the processors and the tasks respectively. From table the pair [A2, B4] =1 suggests that task B4 is allotted to processor A2 that is 1 and [A4, B3] =0 suggests that the task B3 is not allotted to processor A4 that is 0.

The Krill Herd Algorithm is utilized for the dynamic multiprocessor scheduling is as per the following:

- The proposed method begins with an initial population called as Krills.
- The Krills are created in light of the specified population size, the number of processors and number of tasks utilized.

Table 1. A krill representation of task assignment

	B_1	B_2	B_3	B_4	B_5
A_1	1	1	0	1	0
A_2	0	0	0	1	1
A_3	0	0	0	0	1
A_4	1	0	0	1	1

- At first Krills are generated randomly and the good Krill is figured which chooses the goodness of the schedule.
- For each krill individual, the fitness function is characterized as its separations from food and highest density of the swarm [29].
- The time-dependent location of an individual krill can be find out by using below mentioned key activities [29]:
  (i) Movement affected by other krill individuals,
  (ii) Foraging motion
  (iii) Random physical diffusion.
- The best individual is chosen as a best Krill.
- The herding of the krill individuals is a multi-objective process but here we have used a single objective function including two main goals [29]:
  i. Increasing krill density
  ii. Reaching food.
- The increasing density is the Density-dependent attraction of krill and the high food concentration area is finding food. Finally these are utilized as goals which lead to the krill to group all over the place of global minima.
- In this procedure, if a krill individual searches for highest density and food then the krill shifts to the best solution. i.e. the value of objective function will be less when nearer the separation of krill individuals to the high density and food.
- The above process is rehashed for the maximum number of iterations indicated and after that a best solution is acquired.
- When another task appears, it is differentiated and the tasks that are in the waiting queue and another schedule is obtained.
- Accordingly the grouping continues changing with time taking into account the entry of new tasks.

The objective function computes the total execution and the fitness function computes the average of the total execution time of the set of tasks allocated to the processors. $fit\_fun(A_i)$ is a fitness function of A_i processor. It calculates the value of the assigning of task by using (1) [1,32].

$$fit\_fun(A_i) = (1/makespan) \times \max(utilization) \quad (1)$$

The average utilization is computed found on the particular execution of the processor. The utilization of the individual processor is given by (2) [1,32],

$$utilization(A_i) = Finish\_time(A_i)/makespan \quad (2)$$

The average processor utilization is evaluated by dividing the sum of all processors utilization with the total no. of processors i.e. n. At the point when the average processor utilization is optimized, then avoid the processors being unused for long time. The $Objective\ fun$ can be found out using (3). It computes the average of the total execution time of the tasks assigned to the processors [1,32].

$$Objective\ fun = \min\left\{\frac{\sum_{i=1}^{n} fit\_fun(A_i)}{n}\right\} \quad (3)$$

The objective is the minimization of $Objective\ fun$ mentioned in (3). The value clearly indicates the optimum schedule along with the balance in the processor utilization.

III. METHODOLOGY OF MULTIPROCESSOR SCHEDULING USING KHA

KHA considered as a new meta-heuristic swarm intelligence optimization technique to deal with optimization problems [29]. It is established on the model of the grouping of the krill swarms because of particular biological and environmental energized technique. The main systems described are spoken about to the [29]:

- feeding capability
- improved reproduction,
- safety from predators
- Environmental situations

The preeminent species of sea animal is Antarctic krill [29] and [33]. When predators for example penguins, seabirds or seals attack krill, predators eliminate krill individually which consequences in decreasing the krill density while increasing density and discovering territories of high food absorptions are utilized as objectives which lastly direct the krill to group over the global minima. In the same way as other different methods, KHA has begun with creating random krill individuals from the search space and after that calculating them. In Genetic Algorithm and PSO algorithms the arrays known as "Chromosome" and "Particle Position" respectively forms the individuals holding values of parameters [34] whereas in KHA, each array is called "Krill Individual", with dimension $N_{Pop} \times N_{Var}$. In other words N_{Pop} numbers form the Krill matrix for a N_{Var} dimensional optimization problem which can be created by using (4) as:

$$krill_{Matrice} = \begin{bmatrix} x_1^1 & x_2^1 & \cdots & x_{N_{Var}}^1 \\ x_1^2 & x_2^2 & \cdots & x_{N_{Var}}^2 \\ \cdots & \cdots & \cdots & \cdots \\ x_1^{N_{Pop}} & x_2^{N_{Pop}} & \cdots & x_{N_{Var}}^{N_{Pop}} \end{bmatrix} \quad (4)$$

Where, N_{Pop} is no. of Krill individuals and N_{Var} is the no. of variables.

The position of krill in two-dimensional surface changes with each iteration. This process considered with three major events as follows:

1. Movement influenced by other krill individuals,
2. Foraging motion
3. Random physical diffusion.

In KHA, the searching space can be find out using the above three operations. Let M_i denotes the movement affected by other krill individuals, FM_i denotes foraging motion and PD_i denotes physical diffusion of the i^{th} krill. The Lagrangian model is used for searching spaces of arbitrary dimensionality to an n-dimensional decision space as shown in (5).

$$\frac{dY_i}{dt} = M_i + FM_i + PD_i \quad (5)$$

A. *Movement affected by other krill individuals*

The movement of krill individual can be found using (6):

$$M_i^{new} = M^{\max} \alpha_i + \omega_n M_i^{old} \quad (6)$$

Where,

$M^{\max}$ is the maximum induced speed,

ω_n is the inertia weight of the motion induced in $[0,1]$,

M_i^{old} is the last motion induced, and

α_i is the direction of motion induced

In this process, the direction of movement affected or motion induced (α_i) is roughly computed using the target effect, local effect, and a repulsive effect as shown in (7).

$$\alpha_i = \alpha_i^{local} + \alpha_i^{target} \tag{7}$$

Where, α_i^{local} and α_i^{target} are local and target effect given by the nearby krill and the best krill individual respectively.

The mutual forces between individuals can be assumed by the result of the nearby krill in a krill movement individual which can also be called as an attractive/repulsive tendency determined as follows:

$$\alpha_i^{local} = \sum_{j=1}^{NN} \hat{F}_{i,j} \hat{Y}_{i,j} \tag{8}$$

$$\hat{F}_{i,j} = \frac{F_i - F_j}{F^{worst} - F^{best}} \tag{9}$$

$$\hat{Y}_{i,j} = \frac{Y_j - Y_i}{\left\| Y_j - Y_i \right\| + \varepsilon} \tag{10}$$

Where,

F^{best} and F^{worst} are the best and worst fitness values of krill individuals respectively,

F_i is the fitness value of the i^{th} krill individual ,

F_j is the fitness value of j^{th} neighbor individual for $j = 1, 2, \ldots, NN$,

Y is the related positions,

NN is the number of krill neighbors and

ε is a small positive number for avoiding the singularities i.e. to avoid zero coming about within the (10), in denominator.

Distinctive procedures can be utilized for picking the neighbor. For illustration, a neighborhood ratio as it may be characterized to get the no. of the nearby krill. By different heuristic methods a sensing distance (S_d) found out over a krill and the nearby krill ought to be found using the actual behavior of the krill individuals as shown in Figure (1).

Let $S_{d,i}$ is a sensing distance of the i^{th} krill and K is the no. of krill individuals. Finally, the neighbours of the krill are those krill individuals that are in range, i.e. in the area of a circle centred at the position of the i^{th} krill, and which has a radius for each iteration is equal to:

$$S_{d,i} = \frac{1}{5K} \sum_{j=1}^{K} \left\| Y_i - Y_j \right\| \tag{11}$$

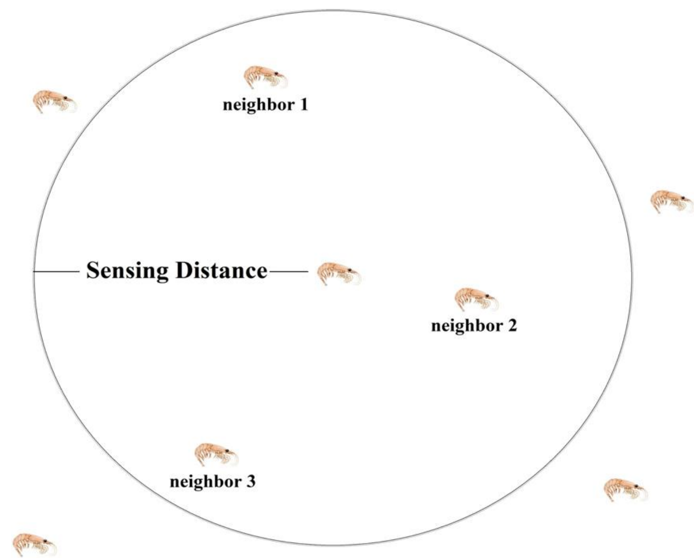

Figure 1. Diagram of the sensing distance over a krill

In the event that the separation of two krill individuals is below $S_{d,i}$ then they are supposed to become neighbors. The factor 5 in the denominator is exactly acquired [34].

The movement of krill is also dependent on the best individual location which leads to the global optima. The identified target vector of every krill individual is the lowermost fitness of a krill individual. Let C^{best} and α_i^{target} is the effective coefficient and the effect of the individual krill respectively with the best fitness on the i^{th} krill. Calculate α_i^{target} by (12):

$$\alpha_i^{target} = C^{best} \hat{F}_{i,best} \hat{Y}_{i,best} \tag{12}$$

C^{best} is the ratio of the individual impact with the best fitness function value for the i^{th} krill can be found out by using (13) [35]:

$$C^{best} = 2\left(\lambda + \frac{I}{I_{\max}} \right) \tag{13}$$

Where,

λ is a random number coming from the interval $[0,1]$,

I denotes the number of the current iteration, and

$I_{\max}$ indicates a maximum number of iterations.

B. Foraging Motion:

It is evaluated with the two fundamental components. They are: food location and the prior knowledge on the food location. For each i^{th} krill the foraging motion could be nearly created using (14) and (15):

$$FM_i = V_f \beta_i + \omega_f FM_i^{old} \tag{14}$$

$$\beta_i = \beta_i^{food} + \beta_i^{best} \tag{15}$$

Where,

V_f is the foraging speed i.e. it describes the speed of searching for food, and has been selected empirically. Its recommended value is 0.02.

ω_f denotes the inertia weight in range $[-1,1]$.

FM_i^{old} is the foraging motion,

β_i^{food} denotes food attraction

β_i^{best} represents effect of the best fitness of i^{th} krill.

The food attraction β_i^{food} is characterized to potentially draw the krill swarm to the global optima [29]. In general after some iteration the krills group over the global optima. The globality of the KH algorithm can be enhanced by considering it as a proficient global optimization method. Food attraction for each i^{th} krill is calculated using (16):

$$\beta_i^{food} = C^{food} \hat{F}_{i,food} \hat{Y}_{i,food} \tag{16}$$

Where, C^{food} represents a food coefficient which can be used to decreases the effect of food in the krill grouping throughout the time.

Equation (17) uses the variable J is same as I. On each j^{th} krill the influence of the impact of food location is determined by using (17) [29]:

$$C^{food} = 2 \times \left(1 - \frac{J}{J_{\max}}\right) \tag{17}$$

The location of food is the quantity that for KHA is defined on the basis of the distribution of the fitness function. The food effect is characterized as far as its location [29]. Initially we find the center of food and afterward try to formulate food attraction. In KHA, the effective center of food absorption is nearly computed by fitness distribution of krills, based on ‘‘center of mass’’ concept. For each iteration, the center of food is determined using (18):

$$Y^{food} = \frac{\sum_{i=1}^{K} F_i^{-1} Y_i}{\sum_{i=1}^{K} F_i^{-1}} \tag{18}$$

The effect of best fitness of the i^{th} krill individual β_i^{best} is hold by including best individual and its position can be determined as [29]:

$$\beta_i^{best} = \hat{F}_{i,best} \hat{Y}_{i,best} \tag{19}$$

Where, $\hat{F}_{i,best}$ is the best formerly visited location of the i^{th} krill.

C. Physical Diffusion:

It is a random process and the vector is defined using the maximum diffusion speed and the directional vector. Let $PD^{\max}$ denotes maximum diffusion speed and δ denotes random directional vector whose value coming from the interval $[-1,1]$. The physical diffusion PD_i can be found by using (20) as follows:

$$PD_i = PD^{\max} \delta \tag{20}$$

The less random the motion then betters the position of the krill. The outcomes of the foraging motion and the movement affected by other krill individuals slowly reduce by raising the iterations. Therefore, one more expression is included in (20) which reduces the random speed linearly with respect to time and makes on the base of a geometrical annealing schedule [29]:

$$PD_i = PD^{\max} \left(1 - \frac{I}{I_{\max}}\right) \delta \tag{21}$$

From the above mentioned motions, location of each krill becomes nearer to the global fitness. The movement affected by other krills and foraging motion contain two global and two local policies and the parallel work of these policies makes KHA powerful [34]. It has an attractive effect when the correlated fitness value of each of the effective factors ($F_j, F^{best}, \hat{F}_{i,best}, \hat{F}_{i,food}$) is better (less) than the fitness of the i^{th} krill; else, it has a repulsive effect [29]. In other words a better fitness is more useful on the movement of i^{th} krill.

This process makes a random search in KHA. Thus the location vector of a krill in the gap from t to $t+\Delta t$ can be found as:

$$Y_i(t+\Delta t) = Y_i(t) + \Delta t \frac{d Y_i}{dt} \tag{22}$$

Where Δt denotes scaling factor for the speed of the search of the solution or search space, and is defined as:

$$\Delta t = C_t \sum_{l=1}^{K} (UB_l - LB_l) \tag{23}$$

Where LB_l and UB_l are lower bound and upper bounds of the l^{th} variables ($l = 1, 2, \ldots, K$), respectively. Search space is the absolute value of $UB_l - LB_l$. The value of C_t is in the interval $[0, 2]$.

D. Genetic operators

The final stage of the main iteration in KHA is the use of genetic operators [35] such as crossover operator and mutation operator.

1) *Crossover operator:* It is initially utilized as a part of Genetic algorithm as a real technique for global optimization which uses an adaptive vectorized crossover method [29]. This operator performs the operation with crossover probability (Cr). There are two ways the operator carried out:
   - Binomial
   - Exponential

In case of binomial method the crossover performs on each of the variables. The l^{th} component of Y_i is $Y_{i,l}$, is manipulated by generating random numbers in the range [0, 1] :

$$Y_{i,l} = \begin{cases} Y_{r,l} & \gamma_{i,l} < Cr \\ Y_{i,l} & otherwise \end{cases} \tag{24}$$

$$Cr = 0.2\, \hat{F}_{i,best} \tag{25}$$

Where, γ a random number from the interval is $[0,1]$ generated according to the uniform distribution and $r \in \{1,2,\ldots,i-1,i+1,\ldots,K\}$ denotes a random index. Utilizing the new crossover probability, the global best value of crossover probability is 0. The crossover probability is increments when the fitness value is decreases. In this approach the crossover operator is acting on a single individual.

2) *Mutation Operator:* The essential part of evolutionary algorithm is the Mutation process. It is contained by a mutation probability (Mu). It modifies the m^{th} coordinate of the i^{th} krill in accordance with the formula [29]:

$$Y_{i,m} = \begin{cases} Y_{gbest,m} + \mu(Y_{p,m} - Y_{q,m}) & for\ \gamma \le Mu \\ Y_{i,m} & for\ \gamma > Mu \end{cases} \tag{26}$$

$$Mu = 0.05 / \hat{F}_{i,best} \tag{27}$$

Where, $p, q \in \{1,2,\ldots,i-1,i+1,\ldots,K\}$ and μ is a random no. from the interval is $[0,1]$.

E. Algorithm for KHA

Generally the KHA can be performed by the following steps:

Step 1: Initialize the parameters i.e. N_{Pop}, $M^{\max}$, ω_n, ε, $I_{\max}$, V_f, ω_f, $PD^{\max}$, C_t

Step 2: Create an initial population randomly in the solution or search space.

Step 3: For each krill estimate the fitness function in line with its position.

Step 4: For each iteration continue the procedure,

for t=1 to N_{Pop} do

for i = 1 to K **do**

Generate Solution $(Y_i(t))$

Evaluate and update best solutions

end for

Save best individual

Sort population of krills

for $i = 1$ to K **do**

Perform motion calculation and genetic operators:

$M_i \leftarrow$ Motion induced or Movement affected by other individuals

$FM_i \leftarrow$ Foraging Motion

$PD_i \leftarrow$ Random Physical diffusion

Crossover

Mutation

Update the position of krill

Update Solution $(Y_i(t))$

Evaluate and update best solutions

end for

Save best individual $Y_i(t)$

Stop condition $\leftarrow$ Check stop condition ()

$t \leftarrow t+1$

Repeat and go to step 4 until stop condition = **false**

return best individual $Y_i(t)$

End

IV. RESULTS AND DISCUSSION

So as to assess the execution of the proposed technique utilizing KHA, we have replicated the execution of GA, BFO, GBF based procedures with the end goal of correlation. In this paper, we have taken the criteria as minimizing the execution time with the simulation process by utilizing MATLAB.

For simulations the parameters set at: $g = 20$, $N_{Pop} = 100$, $M^{\max} = 0.01$, $\omega_n = 0.9$, $\varepsilon = 0.1$, $I_{\max} = N_{Pop}$, $V_f = 0.02$, $\omega_f = 0.9$, $PD^{\max} = 0.005$, $C_t = 0.5$. Where, g symbolizes the no. of Krill and set to value 20. Randomly picked no. of krill and assigned in an array $'c'$, with 4 distinct processors and 5 distinct tasks. If a task is allotted to processor, the value will be '1'; else, the value is 0. The total no. of tasks allotted to processor is generated using 'c' which is the no. of krill i.e. 'K'. Here K=8 i.e. K is the number of krill individuals.

The no. of population is chosen like '100'. KHA allowed repeating the procedure for 100 times in simulation to find the best fitness value in each iteration. Here, best means the smallest value of Y_i taking into account the fact that a minimum of a function is to be located. For every cycle, the KHA has the capacity to get the global minima in every case with high level of exactness. The performance of KHA algorithm after each 25th iteration can be shown in Figure (2).

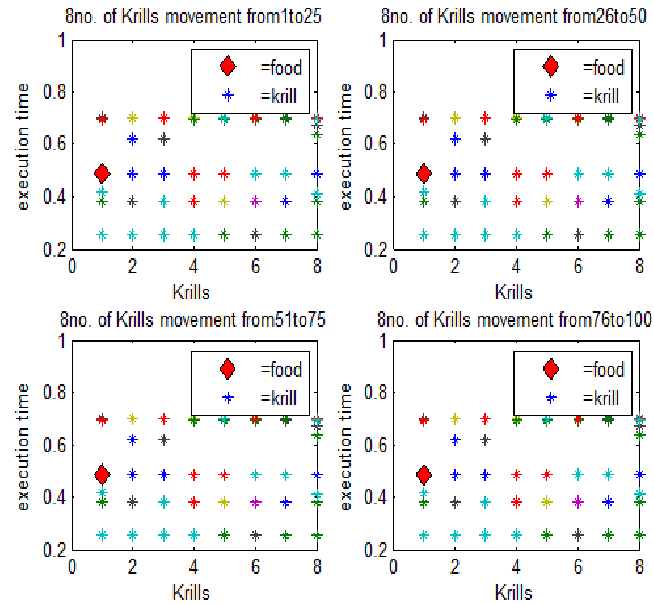

Figure 2. Performance of Krill Movements after each 25th iteration.

In this case the minimum execution time is 0.2553 which is found out by using KHA. The allocation of individual krills for 100 iterations is shown in Figure (3).

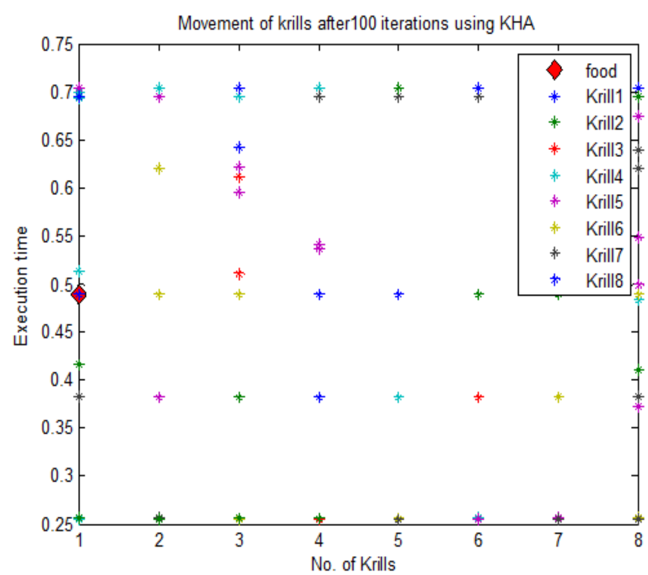

Figure 3. Allocation of Krills for 100 iterations using KHA

Figure (4) to (11) shows the execution time of individual krill movements for 100 iterations using KHA as shown in below diagrams.

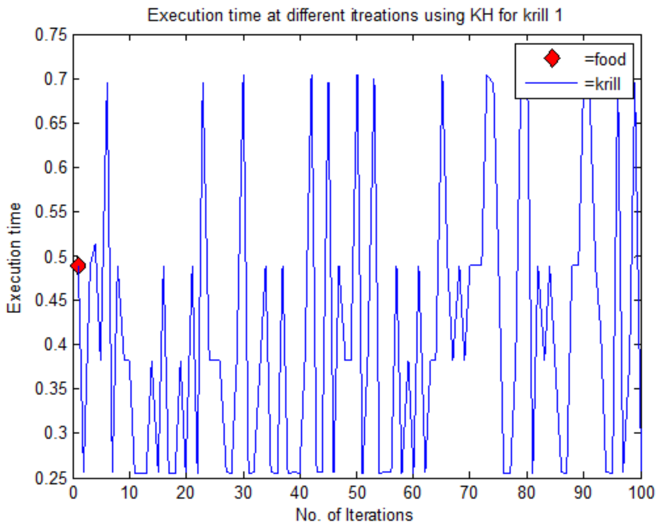

Figure 4. Performance of Krill 1 for 100 iterations

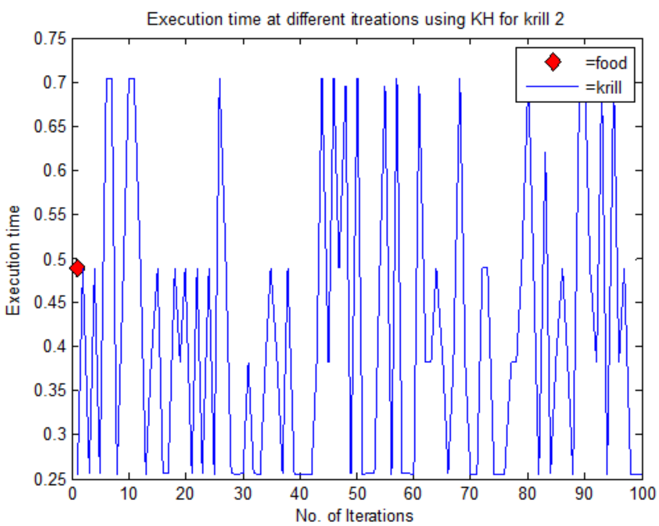

Figure 5. Performance of Krill 2 for 100 iterations

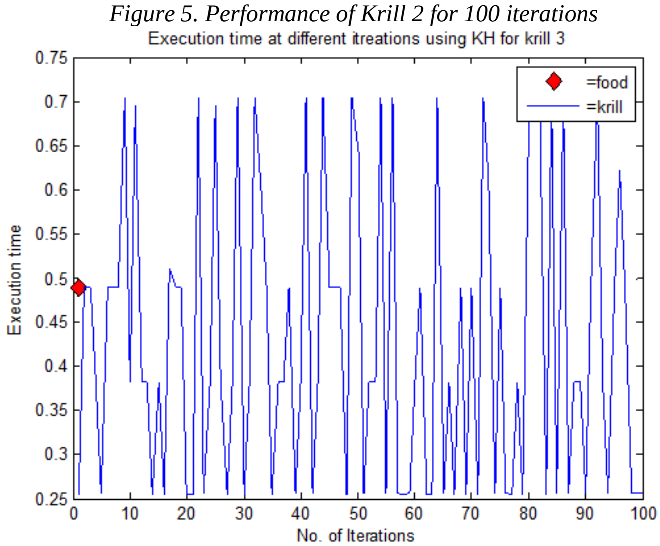

Figure 6. Performance of Krill 3 for 100 iterations

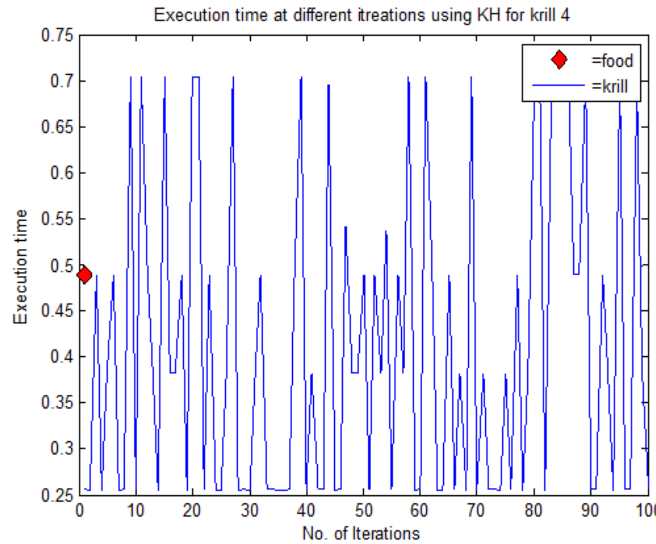

Figure 7. Performance of Krill 4 for 100 iterations

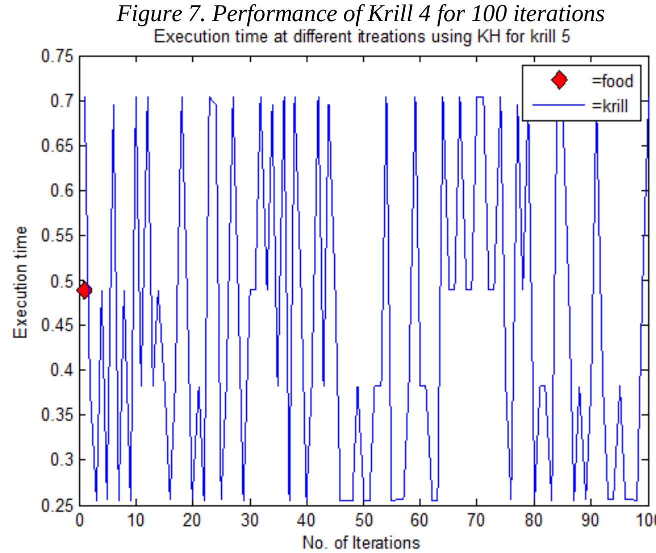

Figure 8. Performance of Krill 5 for 100 iterations

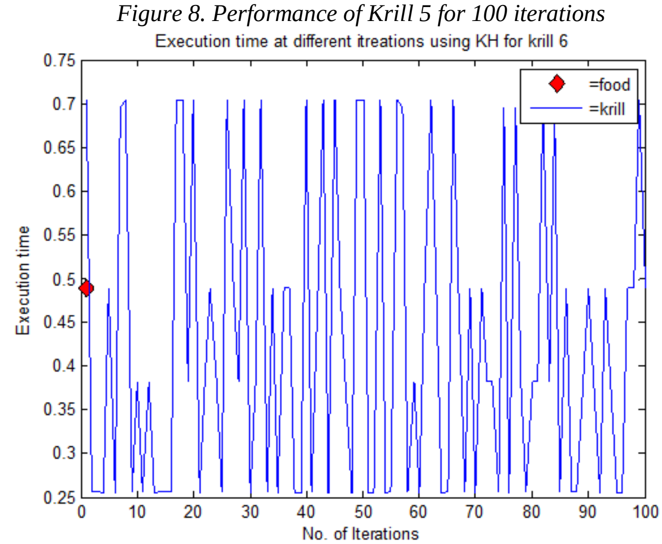

Figure 9: Performance of Krill 6 for 100 iterations

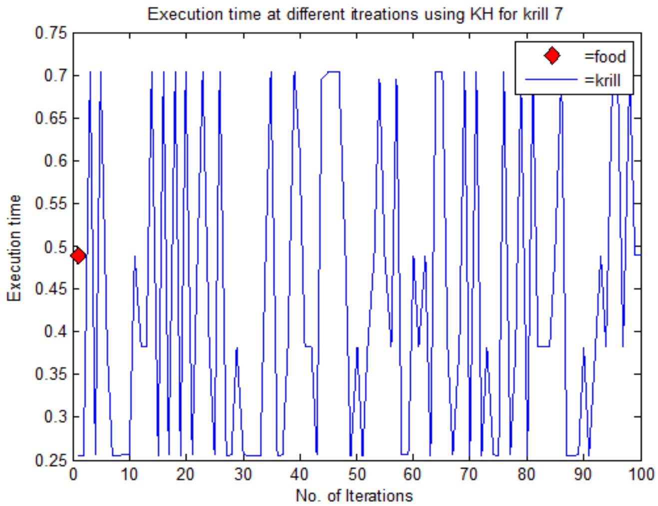

Figure 10: Performance of Krill 7 for 100 iterations

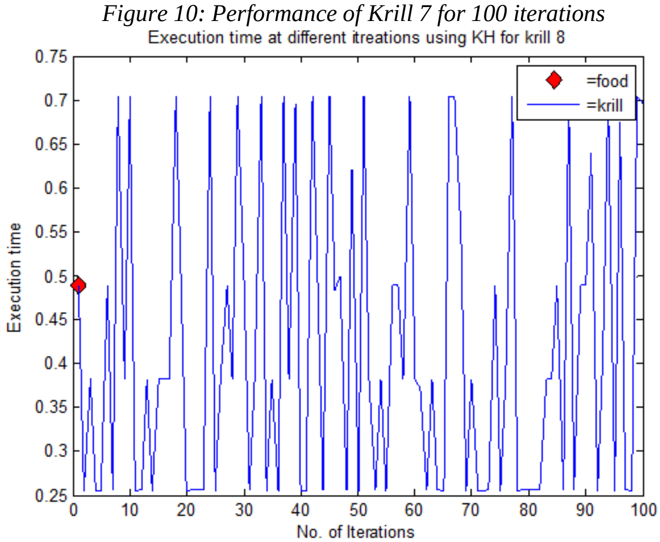

Figure 11: Performance of Krill 8 for 100 iterations

Table 2 shows the creation of data for 100 iteration outcome from GA, BFO, GBF and KHA. The bold values of Table 2 represents the minimum value of respective methodology.

Table 2. Performance of GA, BFO, GBF and KHA with execution time and No. of iteration as variables

Iteration No.	GA	BFO	GBF	KHA
10	59.0000	2.3879	**1.0002**	0.7040
20	41.0000	2.2417	1.0003	**0.2553**
30	24.0000	2.2873	1.0003	0.4885
40	7.0000	2.0342	1.0002	0.2553
50	7.0000	1.9173	1.0003	0.2553
60	7.0000	2.1818	1.0003	0.3824
70	**4.9615**	**1.7331**	1.0003	0.3824
80	4.9615	1.7702	1.0002	0.2566
90	7.0000	1.8326	1.0003	0.4885
100	7.0000	1.9398	1.0002	0.6945

The division of objective function for 100 iterations of *GA, BFO, GBF and KHA* is shown in Figure (12), which results that KHA will produce the better performance as compared with the other mentioned techniques. In other words, from the outcomes, we study that the proposed KHA accomplished fundamentally better performance on all values over the *GA, BFO, GBF*.

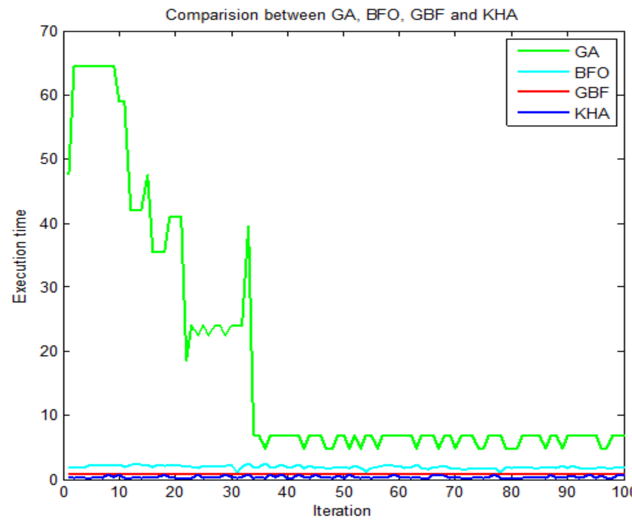

Figure 12: Performance of GA, BFO, GBF, KHA for 100 iterations

V. CONCLUSION AND FUTURE SCOPE

In this paper proposed a KHA is utilized for allocating a task to a processor in the multiprocessor scheduling problem. As obvious from the graphical and experimental results, the proposed KHA performed exceptionally well. In future, our work will be extended by performing with new advanced methods.

REFERENCES

[1] Sasmita Kumari Nayak, Sasmita Kumari Padhy, Siba Prasada Panigrahi, "A Novel algorithm for dynamic task scheduling", Future Generation Computer Systems, 2012, Volume 28, Issue 5, Pages 709-717.

[2] Fatma A. Omara, Mona M. Arafa, Genetic algorithms for multiprocessor scheduling problem, Journal of Parallel and Distributed Computing 70 (1) (2010) 13–22.

[3] Orhan Engin, Gülşad Ceran, Mustafa K. Yilmaz, An efficient genetic algorithm for hybrid flow shop scheduling with multiprocessor task problems, Applied Soft Computing 11 (3) (2011) 3056–3065.

[4] Savaş Balin, Non-identical parallel machine scheduling using genetic algorithm, Expert Systems with Applications 38 (6) (2011) 6814–6821.

[5] Tzu-Chiang Chiang, Po-Yin Chang, and Yueh-Min Huang, "Multi-Processor Tasks with Resource and Timing Constraints

Using Particle Swarm Optimization", *IJCSNS International Journal of Computer Science and Network Security*, Vol.6 No.4 (2006), pp. 71-77.

[6] Holland J. Adaptation in natural and artificial systems. Ann Arbor, MI: University of Michigan Press; 1975.

[7] Al-Tabtabai H, Alex PA. Using genetic algorithms to solve optimization problems in construction. Eng Constr Archit Manage 1999;6(2):121–32.

[8] Grierson DE, Khajehpour S. Method for conceptual design applied to office buildings. J Comput Civil Eng 2002;16(2):83–103.

[9] Joglekar A, Tungare M. Genetic algorithms and their use in the design of evolvable hardware. http://www.manastungare.com/ articles/genetic/genetic-algorithms.pdf; 2003, accessed on May 20, 2004, 15 p.

[10] Storn R, Price K (1997) Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces. J Global Optim 11(4):341–359

[11] Gandomi AH, Yang X-S, Talatahari S, Deb S (2012) Coupled eagle strategy and differential evolution for unconstrained and constrained global optimization. Comput Math Appl 63(1): 191–200. doi:10.1016/j.camwa.2011.11.010

[12] Khazraee S, Jahanmiri A, Ghorayshi S (2011) Model reduction and optimization of reactive batch distillation based on the adaptive neuro-fuzzy inference system and differential evolution. Neural Comput Appl 20(2):239–248. doi:10.1007/s00521-010-0364-x

[13] Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceeding of the IEEE International Conference on Neural Networks, Perth, Australia, pp. 1942–1948

[14] Chen D, Zhao C, Zhang H (2011) An improved cooperative particle swarm optimization and its application. Neural Comput Appl 20(2):171–182. doi:10.1007/s00521-010-0503-4

[15] Talatahari S, Kheirollahi M, Farahmandpour C, Gandomi AH (2012) A multi-stage particle swarm for optimum design of truss structures. Neural Comput Appl. doi:10.1007/s00521-012-1072-5

[16] Gandomi AH, Alavi AH (2012) A new multi-gene genetic programming approach to nonlinear system modeling. Part II: Geotechnical and Earthquake Engineering Problems. Neural Comput Appl 21 (1):189–201

[17] Gandomi AH, Alavi AH (2011) Multi-stage genetic programming: a new strategy to nonlinear system modeling. Inf Sci 181(23):5227–5239. doi:10.1016/j.ins.2011.07.026

[18] Simon D (2008) Biogeography-based optimization. IEEE Trans Evolut Comput 12(6):702–713

[19] Wang G, Guo L, Duan H, Liu L, Wang H (2012) Dynamic deployment of wireless sensor networks by biogeography based optimization algorithm. J Sens Actuat Netw 1(2):86–96. doi: 10.3390/jsan1020086

[20] Yang XS, Gandomi AH (2012) Bat algorithm: a novel approach for global engineering optimization. Eng Comput 29(5):464–483

[21] Gandomi AH, Yang X-S, Alavi AH, Talatahari S (2012) Bat algorithm for constrained optimization tasks. Neural Comput Appl. doi:10.1007/s00521-012-1028-9

[22] Gandomi AH, Yang X-S, Alavi AH (2012) Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems. Eng Comput Ger. doi:10.1007/s00366-011-0241-y

[23] Gandomi AH, Talatahari S, Yang XS, Deb S (2012) Design optimization of truss structures using cuckoo search algorithm. Struct Des Tall Spec. doi:10.1002/tal.1033

[24] Wang G, Guo L, Duan H, Wang H, Liu L, ShaoM(2012) A hybrid meta-heuristic DE/CS algorithm for UCAV three-dimension path planning. Sci World J 2012:1–11. doi:10.1100/2012/583973

[25] Yang X-S, Sadat Hosseini SS, Gandomi AH (2012) Firefly algorithm for solving non-convex economic dispatch problems with valve loading effect. Appl Soft Comput 12(3):1180–1186. doi:10.1016/j.asoc.2011.09.017

[26] Gandomi AH, Yang X-S, Alavi AH (2011) Mixed variable structural optimization using firefly algorithm. Comput Struct 89(23–24):2325–2336. doi:10.1016/j.compstruc.2011.08.002

[27] Talatahari S, Gandomi AH, Yun GJ (2012) Optimum design of tower structures using Firefly Algorithm. Struct Des Tall Spec

[28] Wang G, Guo L, Duan H, Liu L, Wang H (2012) A modified firefly algorithm for UCAV path planning. Int J Hybrid Inf Technol 5(3):123–144

[29] Gandomi AH, Alavi AH (2012) Krill Herd: a new bio-inspired optimization algorithm. Commun Nonlinear Sci Numer Simulat 17(12):4831–4845. doi:10.1016/j.cnsns.2012.05.010

[30] Wang Chen, Yan-jun Shi, Hong-fei Teng, Xiao-ping Lan, Li-chen Hu, An efficient hybrid algorithm for resource-constrained project scheduling, Information Sciences 180 (6) (2010) 1031–1039.

[31] Peng-Yeng YinT, Shiuh-Sheng Yu, Pei-Pei Wang, Yi-Te Wang, "A hybrid particle swarm optimization algorithm for optimal task assignment in distributed systems", science direct, Computer Standards & Interfaces 28 (2006) 441–450

[32] S.N. Sivanandam. "Dynamic task scheduling with load balancing using parallel orthogonal particle swarm optimisation", International Journal of Bio-Inspired Computation, 2009.

[33] Hofmann EE, Haskell AGE, Klinck JM, Lascara CM. Lagrangian modelling studies of Antarctic krill (Euphasia superba) swarm formation. ICES J Mar Sci 2004;61:617–31.

[34] Mani Ashouri, Seyed Mehdi Hosseini, "Application of Krill herd and Water cycle algorithms on Dynamic Economic Load Dispatch Problem", IJIEEB, 2014, (4), pp: 12-19.

[35] Wang, G., Guo, L., Wang, H., Duan, H., Liu, L., Li, J.: Incorporating mutation scheme into krill herd algorithm for global numerical optimization. Neural Comput & Applic 24, 853-871 (2014).

Authors Profile

Mrs. Sasmita Kumari Nayak, received her B.Tech in IT from BPUT, Odisha, India in 2005 and M.Tech in IT from CET, Bhubaneswar, Odisha, India in 2010. Presently she is pursuing her Ph. D reoceanrch under the guidance of Dr. Chandra Sekhar Panda and Dr. Sasmita Kumari padhy. She is working as an Assistant Professor at Computer Science Engineering Department of CUTM (Centurion University of Technology and management), Bhubaneswar, Odisha, India. Her research interest includes Grid Computing, Multiprocessor scheduling, Soft and Evolutionary Computing.

Mr. Chandra Sekhar Panda, had his M.C.A. from OUAT, Bhubaneswar, Odisha, India and Ph.D from Sambalpur University Odisha, India respectively. Presently he is working with Sambalpur University, Odisha, India. His research interests include Programming Languages, Machine Vision, Image Processing, soft computing.

Mrs. Sasmita kumara padhy, had her M.C.A. from BPUT, Odisha, India, M.Tech from Berhampur Univesity Odisha, India and Ph.D from Utkal University Odisha, India respectively. She is presently working with VSSUT, Odisha, India. Her research interests include signal processing, soft and evolutionary computing and multiprocessor scheduling.

