ISSN:0975-9646

Kratika Chandra et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (4) , 2014, 5368-5373

Firefly Algorithm to Solve Two Dimensional Bin Packing Problem

Kratika Chandra , Sudhir Singh

Department of Computer Science and Engineering, U.P.T.U.

Kanpur Institute of Technology, Kanpur, India.

***Abstract*— The Firefly algorithm is one of the several nature inspired algorithms that have been developed in the recent past and is inspired from the flashing behaviour of the fireflies. The field of nature inspired computing and optimization techniques have evolved to solve the difficult optimization problems in diverse fields of engineering, science and technology. Several nature inspired algorithms have been proposed to solve NP hard optimization problems. Bin packing Problem is NP hard optimization problem. Firefly algorithm inspired by the flashing behaviour of fireflies The flashing behaviour of the fireflies is to attract other fireflies in the group for mating. The less bright firefly will be attracted by the brighter one. As all the fireflies are assumed to be unisexual, each firefly is attracted to the other. In this paper, Firefly Algorithm (FFA) is used to solve Two Dimensional Bin packing problem. And further compare it with First-Fit and Best-Fit algorithms. 2-BPP is to pack a set of items with different weights orthogonally into a minimum number of bins, minimizing bin space wastage and execution time.**

The performance of these algorithms was evaluated by testing on three classes of benchmark Data sets which was taken from OR Library. The results have shown that the firefly and best fit is approximately same and well rather than first fit for most of the problem cases.

***Keywords*—Firefly Algorithm, Bin packing problem, first fit algorithm, Best fit algorithm**

I. INTRODUCTION

Bin packing problem is a multi-disciplinary classic combinational optimization problem. It is computationally non-deterministic polynomial-time (NP) hard. In a Two Dimensional bin packing problem objects of different volumes must be orthogonally packed into finite number of bins of same volume. In the real world, the critical issue is to make efficient use of time and space. [6, 10].

Firefly has been first proposed Yang in 2007 [8]. The firefly algorithm is a meta-heuristic optimization algorithm that follows the flashing behavior of fireflies. The brightness of the fireflies is affiliated with the objective function under consideration. The algorithm has been formulated with three main assumptions [8]:

i) All fireflies are unisexual, eliminating the possibility of attraction based on gender that is, each firefly will be attracted by all other fireflies.

ii) Attraction is dependent on the amount of brightness that is a less brighter firefly is attracted to a brighter one.

iii) The brightness of the firefly is equivalent to the objective function.

In this research work 2-BPP was solved by using Firefly algorithm and compare it with First- fit and Best-fit algorithms.

II. BIN PACKING PROBLEM

The Bin Packing problem can be formulated as follows:

$$\text{Minimize } \sum_{j=1}^{n} y_j \quad (1)$$

subject to constraints

$$\sum_{i=1}^{m} w_i x_{ij} \le c y_j \qquad \forall j \in \{1, \dots n\} \quad (2)$$

$$\sum_{j=1}^{n} x_{ij} = 1 \qquad \forall i \in \{1, \dots n\} \quad (3)$$

$$x_{ij} \in \{0, 1\} \qquad \forall i \in \{1, \dots n\}, \forall j \in \{1, \dots n\} \quad (4)$$

$$y_i \in \{0, 1\} \qquad \forall j \in \{1, .. n\} \quad (5)$$

With, y_i= 1 if the bin i is used; else 0

X_{ij}= 1 if the item j is stocked in bin i.

Notations

c: Bin capacity,

i,j: Index of items and bins respectively

w_i: Weight or size of item i

m: minimal number of bins

Equation (1) states the objective function to minimize the number of bins assigned. Equation (2) assures constraint that the bin capacity is not exceeded by the sum of item sizes assigned to this bin. It is often called the capacity constraint. Equation (3) ensures that every item is packed to exactly one bin. Equation (4) and (5) determine that all variables are binary [2, 4].

Firefly algorithm , First- fit and Best-fit algorithms were used to solve Bin packing problem.

A. Firefly Algorithm

Flashing behaviour of the fireflies is unique to the kind of species they belong to and varies from one type of species to the other. The aim of such flashing behaviour can be either of attracting mates or for attracting prey. The flashing light can be formulated to mimic the objective function under consideration and formulate a new optimization algorithm.

The attractiveness of different fireflies leads to the movement of firefly towards the other. Luminescence or light intensity is defined as the amount of light energy transmitted and it varies with the distance. The attractiveness of the fireflies varies with the brightness which is in turn related to the objective function in the mathematical domain. The intensity decreases with the increase in distance, and hence, a given firefly will be attracted to a firefly that is close to it even though it is less bright than a farther but brighter firefly. The intensity of light is known to vary inversely with the square of increasing distance or radius given by [11]:

$$I=I_0 e^{-\gamma r}$$

where I_0 is the original light intensity and γ is the light absorption coefficient.

On the attractiveness of the FFA the main form of attractiveness function or β(r) can be any monotonically decreasing functions such as the following generalized form of

$$\beta(r) = \beta_0 e^{-\gamma r^2}$$

where r or r_{ij} is the distance between the ith and jth of two fireflies. B_0 is the attractiveness at r = 0 and γ is a fixed light absorption coefficient. The distance between any two fireflies i and j at x_i and x_j is the Cartesian distance as follows:

$$r_{ij}=\|x_i-x_j\|_2= \sqrt{\sum_{k=1}^{d}\left(x_{i,k}-x_{j,k}\right)}$$

where $x_{i,k}$ is the k-th component of the i-th firefly (xi). The movement of a firefly, i is attracted to another more attractive (brighter) firefly j, is determined by

$$x_i=x_i+ \beta_0 e^{-\gamma r^2 ij} (x_j-x_i) + \alpha*(\text{rand}-0.5)$$

where the 2nd term is due to the attraction while the 3rd term is the randomization with α being the randomization parameter. Rand is a random number generator uniformly distributed in the range of [0, 1]. In this paper, initially 5 firefly take action and the iterations are 15. In the each iteration 5 different fireflies pack the number of items with different weights and its corresponding capacity from different positions and produce different solutions. Each firefly calculates the light intensity based on the objective function of the BPP and also based on the distance. The attractiveness of a firefly is determined by its brightness and varies according to the distance. The distance of each firefly is calculated based on the starting position which it starts packing the items. In this research work the distance between fireflies is calculated by using Cartesian distance. The movement of one firefly is attracted to another more attractive (brighter) firefly [1, 5].

The following section describes the pseudo code [21].

1. Define an initialize parameters $f(x)$, $x = (x_1, ..., x_d)$
2. Generate initial population of fireflies x_i (i = 1, 2, ..., n)
3. Determine light intensity for x_i by calculating $f(x_i)$
4. Define light absorption coefficient γ
5. **While** t < Maximum Generation
6. Make a copy of the generated firefly population for move function
7. **For** i = 1 : n all n fireflies
8. **For** j = 1 : i all n fireflies
9. **If** ($Ij > Ii$),
10. Move fireflies i and j according to attractiveness Evaluating new solutions and updating light intensity for next iteration
11. **End if**
12. **End for** j
13. **End for** i
14. Sorting the fireflies to find the present best
15. **End while**
16. Begin post process on best results obtained

TABLE I

PARAMETERS & DEFINITION

Parameter	Notation in Algorithm
Brightness	Objective function
Alpha (α)	Randomization parameter
Beta (β)	Attractiveness
Gamma (γ)	Absorption coefficient
Number of generations	Iterations
Number of fireflies	Population
Dimension	Problem dimension

B. First- fit Algorithm

First-Fit algorithm (FFA) contains all empty bins open. It places the next item in the lowest numbered bin in which the item fits. A new bin is opened, If it does not fit in any bin, First-Fit tries to pack every item in these bins before opening a new one with more execution time. In this algorithm the rule followed is:

First place an item in the first bin, called lowest indexed bin into which it will fit, i.e., if there is any partially filled bin then place the item in the lowest indexed bin otherwise, start a new bin [9].

The following section describes the pseudo code

Procedure First-Fit ()

Begin

for All objects i = 1, 2, . . . , n do
for All bins j = 1, 2, . . . do
if Object i fits in bin j then
Pack objects i in bin j.
Break the loop and pack the next object
end if
end for
if Object i did not fit in any available bin then
  Create new bin and pack object i
  end if
  end for
  end procedure

C. Best- fit Algorithm

Best-Fit (BFA) is simple and behaves well in practice. Best-Fit (BF) is the best known algorithm for Bin Packing problem. It picks the bin with the least amount of free space in which it can still hold the current element. This algorithm tries to choose the fullest bin possible with enough space each time an item is assigned. All unoccupied bins are kept open until the end. It places the next item in the bin whose current contents is the largest, but should not exceed its capacity. If it does not fit in any bin, new bin is opened [7].

The following section describes the pseudo code

Procedure Best-Fit ()

begin

for All objects i = 1, 2, . . . , n do
for All bins j = 1, 2, . . . do
if Object i fits in bin j then
Calculate remaining capacity after the object has been added
end if
end for

Pack object i in bin j, where j is the bin with minimum remaining capacity after adding the object (i.e. the object "fits
best")
If no such bin exists, open a new one and add the objects
end for
end procedure

III. IMPLEMENTATION RESULTS

The results obtained from FireFly Algorithm were compared with First-Fit and Best-Fit algorithm. Experimental results are discussed in this section.

A. Two dimensional Bin Packing test Problem

In this research work three classes of benchmark Data set 1, Data set 2 and Data set 3 for BPP-2 are classified as Easy, Medium and Hard classes instances. The benchmark in the easy class instance, n (number of items) ranges from 50 – 500 and c (bin capacity) ranges from 100-150. In the medium class instance n ranges from 50 – 500 and c equals 1000 and in hard class instances n equals 1000 and c equals 100000 [12].

B. Implementation Results of Easy class Instances

In this research work, 15 instances of easy class were solved by using First-Fit, Best-Fit and FireFly Algorithms. The results of three algorithms were compared and analysed. FireFly Algorithm gave optimal solutions for 3 instances out of 15 with minimum bin used. Best-Fit algorithm gave optimal solutions for remaining 12 instances with minimum bin used, its value is approximately same when compared to FireFly Algorithm. Whereas, First-Fit algorithm was found to be used more bin and more free space. It could not achieve optimal solutions. While comparing to the three techniques, FireFly technique have shown good performance for 3 instances out of 15 instances in comparisons to best fit algorithm and for remaining its approximately same. It was found from the results that both FireFly and Best-Fit algorithm are efficient in terms of bin used and free space when compare to First-Fit algorithm. The computational results are given in Table 2 and the Figure 1 and Figure 2 shows the graphical representation of Table 2.

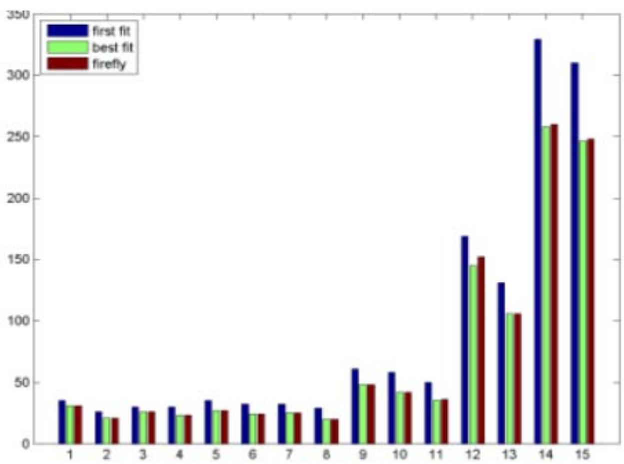

Fig. 1 number of bins used for easy class

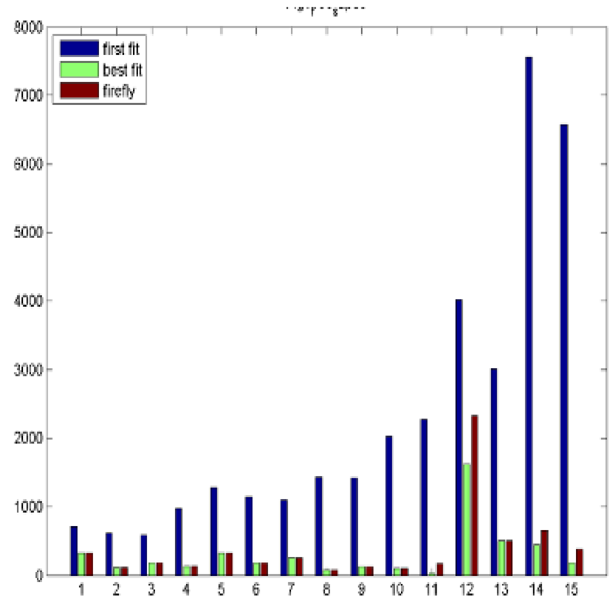

Fig. 2 free space for easy class

C. Implementation Results of Medium class Instances

In medium class, 20 instances were solved. FireFly Algorithm gave optimal solutions for 5 instances with minimum bin used and free space and for remaining 15 instances it obtained optimal solution as same as Best-Fit algorithm gave optimal solutions with minimum bin used and free space . Whereas, First-Fit algorithm was found to be used more bin and free space as compare to best fit and firefly algorithm.

It was found from the results that FireFly technique have shown good performance for 5 instances out of 20 instances when compare to other two techniques. Both FireFly and Best-Fit algorithm are efficient in terms of bin used and free space when compare to First-Fit algorithm. The computational results are given in Table 3. Figure 3 and Figure 4 and shows the graphical representation of Table 3.

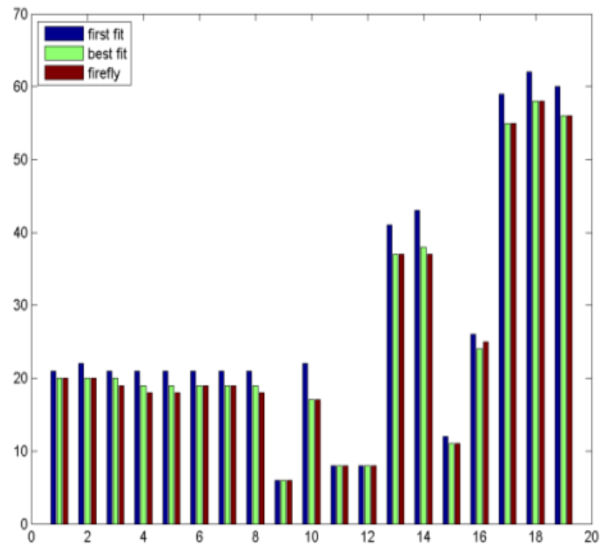

Fig. 3 number of bins used for medium class

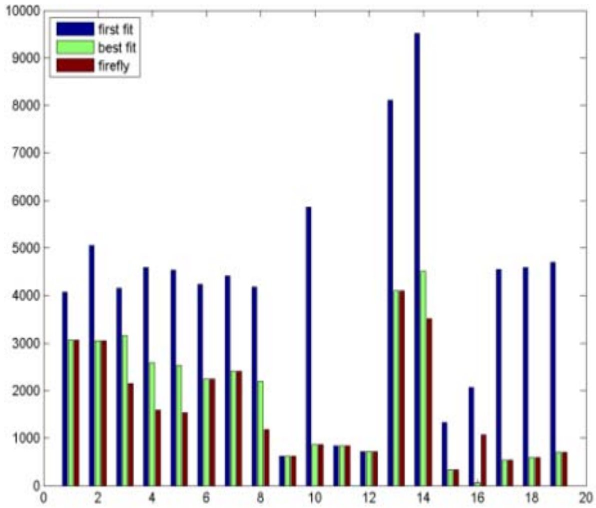

Fig. 4 free space for medium class

D. Implementation Results of Hard class Instances

Hard class instances of Bin Packing problem finds more difficult and challenging to solve. In hard class, 10 instances were solved. It was found from the results that best fit and firefly algorithms have obtained near optimal solutions. FireFly Algorithm and best fit algorithm shows good performance for all the instances when compared to First-Fit techniques. The computational results are given in Table 4. Figure 5 and Figure 6 shows the graphical representation of Table 4.

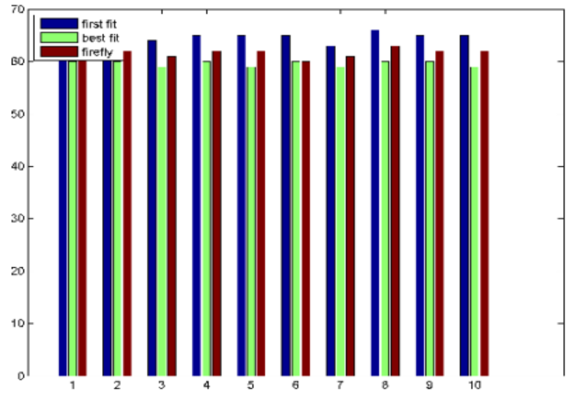

Fig. 5 number of bins used for hard class

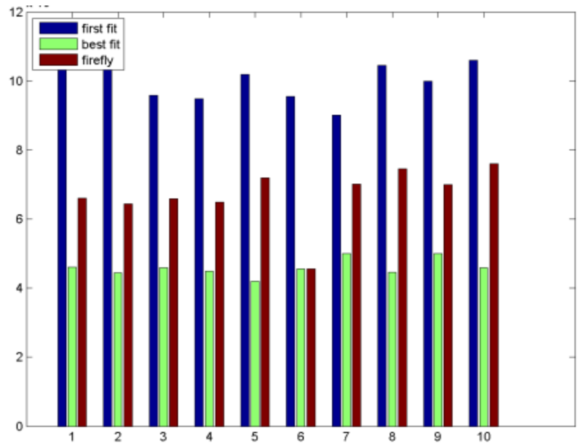

Fig. 6 free space for hard class

TABLE 2
COMPARATIVE RESULT OF EASY CLASS BPP

Ins name	N	C	opt	First fit		Best fit		firefly	
				Used bins	Free space	Used bins	Free space	Used bins	Free space
N1C1W1B	50	100	**31**	35	718	31	318	31	318
N1C1W1C	50	100	**21**	26	616	21	**116**	21	116
N1C1W1N	50	100	**26**	30	579	26	179	26	179
N1C2W1H	50	120	**23**	30	977	23	137	23	137
N1C2W1_I	50	120	**27**	35	1280	27	320	27	320
N1C2W1K	50	120	**23**	32	1144	24	184	**23**	184
N1C2W1L	50	120	**25**	32	1093	25	253	25	253
N1C3W1F	50	150	**19**	29	1435	20	85	**19**	85
N2C1W1A	100	100	**48**	61	1428	48	128	48	**128**
N2C2W1J	100	120	**42**	58	2029	42	109	42	109
N2C3W1A	100	150	**35**	50	2274	**35**	24	36	174
N3C1W4S	200	100	**145**	169	4025	145	1625	145	2325
N3C1W1A	200	100	**106**	131	3012	106	512	106	512
N4C1W1L	500	100	**257**	329	7557	258	457	**257**	657
N4C1W1M	500	100	**246**	310	6578	**246**	178	248	378

TABLE 3
COMPARATIVE RESULT OF MEDIUM CLASS BPP

Ins name	N	C	opt	First fit		Best fit		firefly	
				Used bins	Free space	Used bins	Free space	Used bins	Free space
N1W1B1R0	50	1000	**20**	21	4066	20	3066	20	3066
N1W1B1R1	50	1000	**20**	22	5054	20	3054	20	3054
N1W1B1R3	50	1000	**19**	21	4155	20	3155	**19**	**2155**
N1W1B1R4	50	1000	**18**	21	4590	19	2590	**18**	**1590**
N1W1B1R5	50	1000	**18**	21	4534	19	2534	**18**	**1534**
N1W1B1R6	50	1000	**19**	21	4239	19	2239	19	2239
N1W1B1R7	50	1000	**19**	21	4411	19	2411	19	2411
N1W1B1R8	50	1000	**18**	21	4184	19	2184	**18**	**1184**
N1W4B1R8	50	1000	**6**	6	624	6	624	6	624
N1W1B3R0	50	1000	**17**	22	5866	17	866	17	866
N1W1B3R7	50	1000	**19**	26	8069	19	1069	19	1069
N1W3B2R0	50	1000	**8**	8	842	8	842	8	842
N1W3B3R4	50	1000	**8**	8	715	8	715	8	715
N2W1B1R0	100	1000	**37**	41	8106	37	4106	37	4106
N2W1B1R6	100	1000	**37**	43	9511	38	4511	**37**	**3511**
N2W4B2R1	100	1000	**11**	12	1331	11	331	11	331
N3W4B3R0	200	1000	**24**	26	2068	24	1068	25	1068
N4W4B3R4	500	1000	**55**	59	4543	55	543	55	543
N4W4B3R6	500	1000	**58**	62	4595	58	595	58	595
N4W4B3R9	500	1000	**56**	60	4696	56	696	56	696

TABLE 4
COMPARATIVE RESULT OF HARD CLASS BPP

Ins name	N	C	opt	First fit		Best fit		firefly	
				Used bins	Free space	Used bins	Free space	Used bins	Free space
HARD1	200	10000	**60**	66	1061159	60	461159	62	661159
HARD2	200	10000	**60**	66	1044214	60	444214	62	644214
HARD3	200	10000	**59**	64	958397	59	458397	61	658397
HARD4	200	10000	**60**	65	949303	60	449303	62	649303
HARD5	200	10000	**59**	65	1019247	59	419247	62	719247
HARD6	200	10000	**60**	65	955534	60	455534	60	455534
HARD7	200	10000	**59**	63	900807	59	500807	61	700807
HARD8	200	10000	**60**	66	104602	60	446029	63	746029
HARD9	200	10000	**60**	65	999542	60	499542	62	699542
HARD0	200	10000	**58**	65	1059718	59	459714	58	659714

IV. CONCLUSIONS

In this research work, FireFly Algorithm (FFA), Best fit algorithm and First-Fit algorithm were implemented for Two Dimensional BPP with fixed sized bins. These algorithms were tested on standard benchmark problem instances. The results obtained by the FireFly Algorithm shows good performance for most of the problem cases when compared to First-Fit algorithm and approximately same when compared to Best fit algorithm results. This work can be extended by modifying the parameter settings for FireFly Algorithm to improve the results. FireFly Algorithm can be effectively modified for other types of combinatorial optimization problems. FireFly Algorithm can also be applied to solve three dimensional Bin Packing problem.

REFERENCES

[1] R.yesodha & T.Amudha , " *Effectiveness of FireFly Algorithm in solving Bin Packing Problem*", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 5, May 2013

[2] Abdesslem Layeb & Seriel Rayene Boussalia, "*A Novel Quantum Inspired Cuckoo Search Algorithm for Bin Packing Problem*", I.J. Information Technology and Computer Science, 5, 58-67 2012.

[3] Khadwilard, A., Chansombat, S., Thepphakorn, T., Chainate, W., & Pongcharoen, P. (2012). *Application of Firefly Algorithm and its parameter setting for job shop scheduling*. 49-58.

[4] Doina Bein, Wolfgang Bein and Swathi Venigella, "*Cloud Storage and Online Bin Packing*", accepted at Intelligent Distributed Computing Conference (IDC 2011), The Delft, Netherlands, October 5-7, 2011.

[5] Aphirak Khadwilard, Sirikarn Chansombat, Thatchai Thepphakorn, Peeraya Thapatsuwan, Warattapop Chainate and Pupong Pongcharoen, "*Investigation of Firefly Algorithm Parameter Setting for Solving Job Shop Scheduling Problems*", 2011.

[6] Prof. Dr. Karsten Weihe and Prof. Dr. Matthias Müller-Hannemann, "*Solution Techniques for specific Bin Packing Problems with Applications to Assembly Line Optimization*", 2008.

[7] Swathi Venigella, "*Cloud Storage and Online Bin Packing*", May 2008.

[8] Yang, X. S. (2010). *Nature-inspired metaheuristic algorithms*. Luniver Press

[9] Ullas Mohan, "*Bio Inspired Computing*", Division Of computer science, SOE, CUSAT, July 2008

[10] Zhihua Cui, Zhongzhi Shi and Janos Abonyi, "*New trends on bio-inspired computation", Transactions of the Institute of Measurement and Control* 34(6) 665– 667, 2011.

[11] Yang, X. S. (2009). *Firefly algorithms for multimodal optimization*. In *Stochastic algorithms: foundations and applications* Springer Berlin Heidelberg.

[12] http://www.wiwi.uni-jena.de/Entscheidung/binpp