A Novel Bat Algorithm for Channel Allocation to Reduce FWM Crosstalk in WDM Systems

Ritu
Department of Electronics and Communication Engineering
Haryana Engineering College, Jagadhri, India

Shonak Bansal
Department of Electronics and Communication Engineering
PEC University of Technology, Sector-12, Chandigarh, India

Shweta Sharma
Department of Electronics and Communication Engineering
Haryana Engineering College, Jagadhri, India

ABSTRACT
In this paper on which work done is nature inspired algorithm named Bat algorithm. nature is good source for inspiration in life in different way. Even in many search, nature gives good example for optimization many complex problems in engineering fields. Bat algorithm is metaheuristic algorithm like particle swarm, firefly. This paper formulates on echolocation behavior to reduce the crosstalk like FWM in optical wavelength division multiplexing (WDM) system for solving channel allocation problems by using concept of OGR (Optimal Golomb ruler). The comparative study of simulation results obtained by proposed metaheuristic Bat algorithm demonstrates better and efficient generation of OGRs without the requirement of increasing total bandwidth of channel, unlike the two existing conventional algorithms i.e. Extended quadratic congruence (EQC) and Search algorithm (SA), in terms of ruler length and total channel bandwidth.

General Terms
Conventional computing, Four–wave mixing, Multi– objective, Nature–inspired, Metaheuristics Optimization.

Keywords
Channel allocation,Genetic algorithm, Metaheuristic Bat optimization algorithm, Optimal Golomb ruler, FWM (Four Wave Mixing), WDM (Wave Division Multiplexing).

1. INTRODUCTION
FWM interference (crosstalk) is most important source perform demotion of WDM channel system. In an attempt to reduce the FWM crosstalk in WDM system unequally spaced channel allocation (USCA) techniques have been studied in [1], [3] so that performance can be improve. In unequal channel spacing allocation techniques to make certain that no FWM signal will generated at any channel frequencies. However, USCA techniques have the drawback of increased of optical channel bandwidth requirement compared to equally spaced channel allocation (ESCA). So the frequencies between any two channel is different from any other channel in minimum
Operating bandwidth. In this paper (OGRs) optimal Golomb rulers concept used for unequally spaced bandwidth channel allocation without affecting total bandwidth to reduce FWM interference in WDM system.This USCA technique suppress the FWM crosstalk signals in the optical WDM systems without inducing additional cost in terms of channel bandwidth [6], [14]–[16].Golomb rulers drive a class of NP–complete problems [18]. Various search algorithms represent to solve Golomb ruler problem like exhaustive search [23] for higher order marks, constraint programming [21], local searches [22]. The success of ORGs problem solve by nature–inspired algorithms such as Genetic Algorithm [24]–[28], Tabu Search (TS) [27], Biogeography Based Optimization. (BBO) [28]–[30], Big Bang–Big Crunch (BB–BC) evolution theory [32], [33], and Firefly Algorithm (FA) [34]. Hence, nature–inspired algorithms seem to be very effective solutions for the NP–complete problems. In this paper, a new nature–inspired metaheuristic algorithm based on echolocation behaviours of bets namely Bat algorithm for finding the OGR sequences is being presented.

After this the paper represents the concept of Golomb rulers, Bat based channels allocation algorithm. Section 3 introduces with Bat nature–inspired optimization algorithm. Section 4 presents the problem formulation. Section 5 presents the simulation results and performance comparison of proposed algorithm and Section 6 presents the conclusion and future scope of the research.

2. GOLOMB RULERS
Golomb ruler refers to a set of positive integers named as *marks* and no distinct pairs of numbers from the set have the same difference [38]–[40]. The difference between the values of any two marks is called the *distance* between those marks. The difference between the largest and smallest number is referred to as the *length* of the ruler. The number of marks on a ruler is referred to as the *size* of the ruler. A *perfect Golomb ruler* measures all the non–negative integer distances from 0 to length *L* of the ruler [40]–[43]. An *optimal Golomb ruler* is the shortest length ruler for a given mark. There can be numerous different OGRs for a specific marks value. Figure 1 show an example of 6–marks non– optimal Golomb ruler having ruler length 17. The distance associated between each pair of marks is also shown in Figure 1. As clear from Figure 1 that the distance numbers 14 and 15 are missing so it is not a perfect Golomb ruler sequence.

3. BAT ALGORITHM
Bat algorithm, a meta–heuristic population–based optimization algorithm based on the theories of the echolocation behavior of bats.

Bat theory uses a type of sonar called echolocation, which used to detect prey, avoid obstracles and locate their food and insects in the dark [44]. Bats release a very loud sound pulse and these pulses strike the insects (objects) comes back as echo. Pulse comes back depend on the frequency *f* and velocity of sound in air [44].

$$\lambda = v/f \quad (1)$$

For simplicity, we now use the following approximate or idealized rules:

a. All bats use echolocation to sense distance, difference between food/prey and background barriers in some magical way.

b. Bats fly with vary velocity vi at position xi with a fixed frequency ƒmin, varying wavelength λ and loudness A0 to search for prey. Depends on their target a wavelength (or frequency) adjust of their emitted wavelength (or frequency) adjust the pulse rate r. Adjust the rate pf pulse emission r Є [0, 1], depending on the proximity of their target.

c. Although the loudness varies from a large (positive) A0 to a minimum constant value Amin.

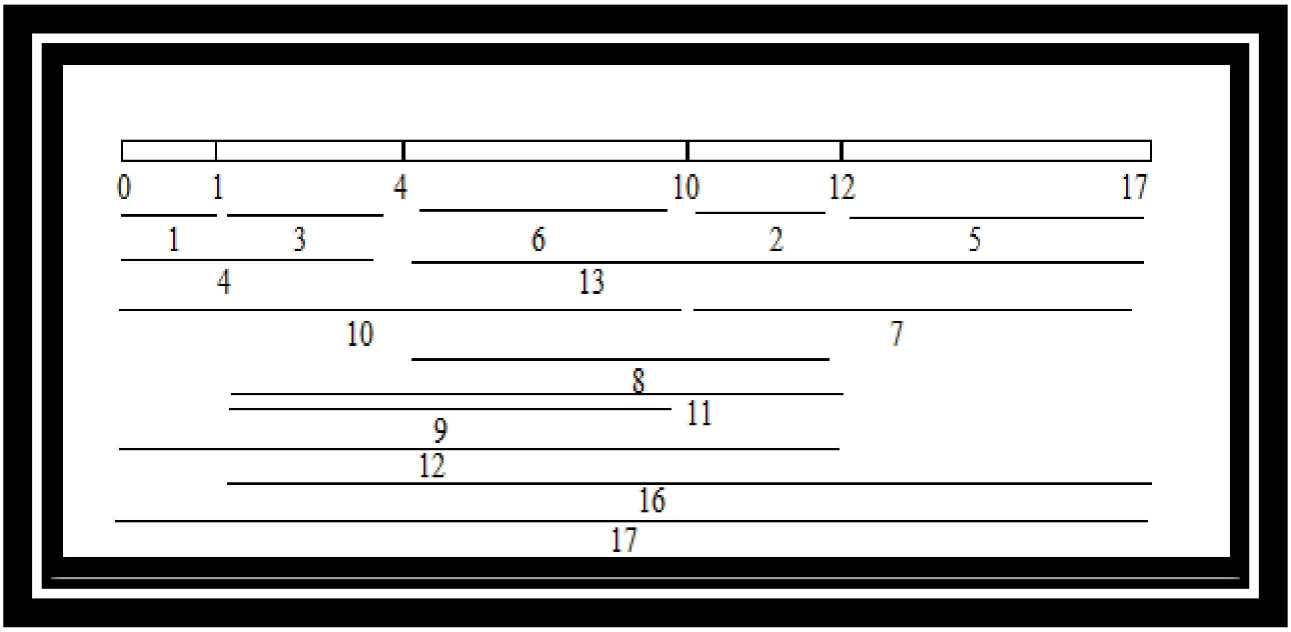


Fig1: A Non OGR having Ruler Length 17 with 6 Marks

In general the frequency lies in range [ƒmin, ƒmax].In simulations, bats positions xi and velocities vi are updated. The new solutions x_i^t and velocities v_i^t at time step t are given by [44]

$$fi = f\min + (f\max - f\min)\beta, \quad (2)$$
$$v_i^t = v_i^{t-1} + (x_i^t - x^*)fi, \quad (3)$$
$$x_i^t = x_i^{t-1} + v_i^t, \quad (4)$$

Here x* is the current global best location (solution) which is located after comparing all the solutions. Where β Є [0, 1] is a random vector drawn from a uniform distribution. Initially, each bat is randomly assigned a frequency which is drawn uniformly from [ƒmin, ƒmax].

In search part, once a solution is selected among the current best solutions, for each bat a new solution is generated locally using random process (5).

$$X_{new} = x_{old} + \epsilon A^t, \quad (5)$$

Where ϵ Є[−1, 1] is a random number, while $A^t = < A_i^t >$ is the average loudness of all the bats at this time step.

Bat algorithm can be considered as a balanced combination of the standard particle swarm optimization and the intensive local search controlled by the loudness and pulse rate. As the iteration proceed the loudness the loudness A_i and the pulse rate r_i update as equation (6). As know loudness usually decreases once a bat has found its prey, while the pulse rate increases. For simplicity, we can also use A0 = 1 and Amin = 0, assuming Amin = 0 means that a bat has just found the prey and temporarily stop emitting any sound. Now we have

$$A_i^{t+1} = \alpha A_i^t,\ r_i^{t+1} = r_i^0 [1 - \exp(-\gamma t)], \quad (6)$$

Where γ and α are constants. Their loudness and emission rates will be updated only if the new solutions are improved, which means that these bats are moving towards the optimal solution.

4. PROBLEM FORMULATION

If CS denoted the spacing between any pair of channels and the n is total number of channels is, then the objective is to optimize (minimize) the length of the ruler denoted as *RL*, which is given by the equation (7) [28]:

$$RL = \sum_{i=1}^{n-1} (CS)_i \quad (7)$$

Subject to $(CS)_i \neq (CS)_j$.

If each individual element is a Golomb ruler, the sum of all elements of an individual forms the total optical bandwidth of the channels. Thus, if an individual element is denoted as *IE* then the second objective is to minimize the total optical bandwidth *TBW* which is given by the equation (8):

$$TBW = \sum_{i=1}^{n} IE)_i \quad (8)$$

Subject to $(IE)_i \neq (IE)_j$.

Where $i, j = 1,2,..., n$ with $i \neq j$ are distinct in both equations (7) and (8).

The proposed pseudo–code for Bat algorithm to generate OGR sequences as unequally spaced channel– allocation in optical WDM system is shown in Figure 3

```
BatAlgorithm

Begin
/* BA parameter initialization */
            Define objective function f (X); X = (x1,…, xd)T,
            Generate initial bat population   xi (i = 1, 2,…n); Generate initial velocity vi and Define pulse frequency ƒi;
            Initialize Loudness coefficient A and pulse rate coefficient r;
/* End of BA parameter initialization */
            Generate maximum number of iteration;
While not T                                        /* T is a termination criterion */
       For t = 1: N
       For i = 1: n
          Generate new solutions by adjusting frequency;
          Updating velocities and locations/solutions [equations (2) to (4)];
      If (rand > r)                  Select a solution among the best solutions;
          Generate a local solution around the selected best solution;
     End If
         Generate a new solution by flying randomly;
     If (rand < Ai & f(xi) < f(x*))
         Accept the new solutions;
         Update pulse rate ri  (as increase) and  loudness Ai  (as reduce ) to improve best solutionedxx [equation (6)];
     End If
End for i                 End for t
          Rank the bats and find the current best x*;
End while
          Postprocess results and visualization;
End
```

Fig 2:Pseudo-code for Algorithm

```
Bat Algorithm to Generate OGRs
Begin
/* BA parameter initialization */
Define operating parameters for Bat algorithm
Initialize the number of channels (marks), lower and upper bound on the length of ruler;
While not Popsize                                            /* Popsize is the population size input by the user */
Generate a random set of Bat (integer population);      /* Number of integers in Bat is being equal to the number of channels */
Check Golombness of each Bat;
If Golombness is satisfied; Retain that Bat;
Else
Remove that particular Bat from the generated population;
End if
End while
         Compute the frequency (total bandwidth);                    /* Frequency represents the fitness (cost) value */
         Rank the Bat from best to worst based on fitness value;
/* End of BA parameter initialization */
While not T                                                                   /* T is a termination criterion */
/* Movement */              For t = 1: N
                             For i = 1: n
A:                Compute the velocity and location for N Bats and Check Golombness of  Bat;
              If Golombness is satisfied   then Check the pulse rate and loudness;
                 Else
                Retain the previous generated Bats and then go to A;
/* Previous generated Bat is being equal to the bat generated into the BA parameter initialization step */
           End if              End if
B:              Vary velocity v with frequency f;
                Determine the new position of each Bat and update frequency and position;
End for /* End for i */
End for /* End for t*/                                 /* End of Movement */
Rank the Bats from best to worst based on fitness value and find the current best;
End while
Display the optimal Golomb ruler sequences;                 End
```

Fig 3: Pseudo–code for Bat Algorithm to Generate OGR Sequences

5. SIMULATION RESULTS AND DISCUSSION

The proposed BAT algorithm has been written and verified in Matlab–7 language [53] under Windows 7 operating system to find OGR sequences means unequal spaced channel–allocation algorithm in optical WDM systems, the proposed. This section show the effectiveness of the proposed algorithm, its performance is being compared with known OGRs [15], [20], [38]–[42], [54]–[56], EQC, SA [2], [13], [24], GA [28], and BB–BC [31], [32] algorithms of generating unequal spaced WDM channel–allocation sequences.

5.1 Simulation Parameters for Bat Algorithm

To get optimal OGR as optical WDM channel–allocation, after a number of careful experimentation, the optimum values of BAT parameters finally been settled in this research is reported in Table 1.

A set of 18 trials for *n* = 3 to 20 are given in Table 2. With pulse rate and loudness parameters settings, the large numbers of sets of trials for various marks were conducted shown in table. Table 3. The performance of all the sets is nearly the same as given in Table 3.

Table 1. Simulation parameters fo Bat Algorithm

Parameter	Value
Pulse rate	**0.5**
loudness	**0.5**

5.2 Effect of Increasing Generations on Total Channel Bandwidth

In performance of Bat algorithm iterations has great effect as shown in Table 3 i.e. ruler length and total bandwidth gets optimized after a certain numbers of iterations. Total optical bandwidth of the sequence tends to decrease as number of iterations increase; it means that the rulers reach their optimum values after a certain number of iterations. This is the point where the results are optimum and no further improvement is seen, that is, we are approaching towards the optimal solution. By carefully observation, the paper fixed the iterations of 1000 for Bat algorithm. With these parameters values, a number of sets of trials for various order marks are conducted.

5.3 Comparison of BA with Previous Existing Algorithms in Terms of Ruler Length, Total Bandwidth and Average CPU Time

The aim to use Bat algorithm in this paper is to optimize the length of the ruler so as to conserve the total bandwidth occupied by by the channels in less computational time. Table 2 list the length of ruler (*RL*), total optical bandwidth (*TBW*) and average CPU time occupied by different sequences obtained by proposed algorithm for various channels *n* and its comparison with known OGRs [15], [20], [38]–[42], [54]–[56], EQC, SA [2], [13], [24], GA [28], and BB–BC [31], [32] algorithms.

EQC and SA is the conventional algorithms .The application of conventional algorithms is limited to prime powers [2], so Table 2. Show the length of ruler and total bandwidth for EQC and SA. The results obtained for 6 marks take few second to performed where as marks increase time is taken in minutes. Table 2. show that time taken by Bat algorithm is very small as compared to others non heuristic exhaustive searches. Comparing the simulation results obtained from Bat algorithm with known OGRs, EQC, SA, GA and BB-BC; it is perceived that there is a significant improvement with respect to the length of the ruler, the total bandwidth occupied and average CPU time that is, the results gets better. Figure 4 (a) and 4 (b) illustrates the graphical comparison of Bat algorithm to generate OGR sequences for optical WDM system with existing algorithms in terms of the length of the ruler and total optical bandwidth occupied by the various order mark values respectively, whereas Figure 5 illustrates the comparison of proposed Bat algorithm with GA and BB–BC algorithm in terms of average CPU time (in Sec.) for various order marks. So, it is concluded from Table 2, Figures 4 and 5 that the performance of proposed Bat algorithm is better than the existing algorithms.

6. CONCLUSION

This paper presented the aaplication of nature-inspired metaheuristic Bat optimization algorithm to search OGRs sequences needed for optical WDM systems. BA generate very efficiently the optimal Golomb ruler's sequence that provides the unequal channel–allocation in optical WDM systems to reduce the FWM crosstalk. The performance is being compared with the existing conventional i.e EQC and SA and nature– inspired algorithms in terms of the length of ruler, total optical channel bandwidth and average CPU time obtained by the different sequences.

The preliminary results indicate that proposed Bat algorithm appears to be most efficient algorithm to generate OGRs for optical WDM systems and outperforms the existing algorithms. As to see the complexity of realizing the unequal channel spacing, so real researches of algorithms does not show. In the future, in order for these algorithms to be of practical use, it is desired that the performance of the algorithms for higher order OGRs channels may be evaluated and may be used to provide unequal channel spacing in real optical WDM systems. Though this process will be very time consuming yet this needs be done for this work to be of some use in the field of communication engineering.

7. REFERENCES

[1] Kwong, W. C., and Yang, G. C. 1997. An Algebraic Approach to the Unequal-Spaced Channel-Allocation Problem in WDM Lightwave Systems, IEEE Transactions on Communications, Vol. 45, No.3, pp. 352–359.

[2] Saaid, N. M. 2010. Nonlinear Optical Effects Suppression Methods in WDM Systems with EDFAs: A Review, In Proceedings of the International Conference on Computer and Communication Engineering (ICCCE), Kuala Lumpur, Malaysia.

[3] Aggarwal, G. P. 2001. Nonlinear Fiber Optics, Edition, Academic Press, San Diego, CA.

[4] Thing, V. L. L., Shum, P., and Rao, M. K. (2004). Bandwidth–Efficient WDM Channel Allocation for Four-Wave Mixing-Effect Minimization, IEEE Transactions on Communications, Vol. 52, No. 12, pp. 2184 –2189.

[5] Forghieri, F., Tkach, R. W., Chraplyvy, A. R., and Marcuse, D. 1994. Reduction of Four–Wave Mixing Crosstalk in WDM Systems Using Unequally Spaced Channels. IEEE Photonics Technology Letters, Vol. 6, No. 6, pp. 754 –756.

[6] Babcock, W. C. 1953. Intermodulation interference in radio systems, Bell Systems Technical Journal, pp. 63–73.

[7] Sardesai, H. P. 1999. A Simple Channel Plan to Reduce Effects of Nonlinearities In Dense WDM Systems. Lasers and Electro–Optics, (23–28, May–1999), pp. 183– 184.

[8] Forghieri, F., Tkach, R. W., and Chraplyvy, A. R. 1995. WDM systems with unequally spaced channels. J. Lightwave Technol., Vol. 13, pp. 889–897.

[9] Hwang, B. and Tonguz, O. K. 1998. A Generalized Suboptimum Unequally Spaced Channel Allocation Technique—Part I: In IM/DD WDM systems. IEEE Trans. Commun., Vol. 46, pp. 1027–1037.

[10] Tonguz, O. K. and Hwang B. 1998. A Generalized Suboptimum Unequally Spaced Channel Allocation Technique—Part II: In coherent WDM systems. IEEE Trans. Commun., Vol. 46, pp. 1186–1193.

[11] Atkinson, M. D., Santoro, N., and Urrutia, J. 1986. Integer sets with distinct sums and differences and carrier frequency assignments for nonlinear repeaters. IEEE Trans. Commun., Vol. COM-34.

[12] Randhawa, R., Sohal, J. S. and Kaler, R. S. 2009. Optimum Algorithm for WDM Channel Allocation for Reducing Four-Wave Mixing Effects. Optik120, pp. 898–904.

[13] http://www.compunity.org/events/pastevents/ewomp204/jaillet_krajecki_pap_ew04.pdf.

[14] Bloom, G. S. and Golomb, S.W. 1977. Applications of Numbered Undirected Graphs. In Proceedings of the IEEE, Vol. 65, No. 4, pp. 562–570.

[15] Thing, V. L. L., Rao, M. K. and Shum, P. 2003. Fractional Optimal Golomb Ruler Based WDM Channel Allocation. In Proceedings of the 8th Opto–Electronics International Journal of Computer Applications (0975 – 8887) Volume 85 – No 9, January 201425and Communication Conference (OECC–2003), Vol. 23, pp. 631-632.

[16] Shearer, J. B. 1998. Some New Disjoint Golomb Rulers. IEEE Transactions on Information Theory, Vol. 44, No. 7, pp. 3151–3153.

[17] http://theinf1.informatik.unijena.de/teaching/ss10/oberseminar-ss10

[18] Robinson, J. P. 1979. Optimum Golomb Rulers. IEEE Transactions on Computers, Vol. C-28, No. 12, (December 1979), pp. 943–944.

[19] Shearer, J. B. 1990. Some New Optimum Golomb Rulers. IEEE Transactions on Information Theory. IT-36, pp. 183–184.

[20] Galinier, P., Jaumard, B., Morales, R. and Pesant, G. 2001. A constraint–Based Approach to the Golomb Ruler Problem. In Proceeding of 3rd International workshop on integration of AI and OR techniques (CP–AI–OR 2001).

[21] Leitao, T. 2004. Evolving the Maximum Segment Length of a Golomb Ruler. Genetic and Evolutionary Computation Conference, USA.

[22] Rankin, W. T. 1993. Optimal Golomb Rulers: An exhaustive parallel search implementation.

[23] M.S.thesis,DukeUniversity,Availableathttp://people.ee.duke.edu/~wrankin/golomb/golomb.htm.

[24] Shobhika. 2005. Generation of Golomb Ruler Sequences and Optimization Using Genetic Algorithm. M.Tech. Thesis, Department of Electronics and Communication Engineering, Thapar Institute of Engineering and Technology, Deemed University, Patiala.

[25] Soliday, S. W., Homaifar, A. and Lebby, G. L. 1995. Genetic Algorithm Approach to the Search for Golomb Rulers. In Proceedings of the Sixth International Conference on Genetic Algorithms (ICGA–95), Morgan Kaufmann, pp. 528–535.

[26] Robinson, J. P. 2000. Genetic Search for Golomb Arrays. IEEE Transactions on Information Theory, Vol. 46, No. 3, pp. 1170–1173.

[27] Ayari, N., Luong, T. V. and Jemai, A. 2010. A Hybrid Genetic Algorithm for Golomb Ruler Problem. In Proceeding of ACS/IEEE International Conference on Computer Systems and Applications (AICCSA 2010), pp.1–4.

[28] Bansal, S., 2014. Optimal Golomb Ruler Sequence Generation for FWM Crosstalk Elimination: Soft Computing Versus Conventional Approaches. Applied Soft Computing Journal (Elsevier), Vol. 22, pp. 443–457.

[29] Bansal, S., Kumar, S., Sharma, H. and Bhalla, P. 2011. Generation of Golomb Ruler Sequences and Optimization Using Biogeography Based Optimization. In Proceedings of 5th International Multi Conference on Intelligent Systems, Sustainable, New and Renewable Energy Technology and Nanotechnology (IISN–2011), Institute of Science and Technology Klawad, Haryana, pp 282–288.

[30] Bansal, S., Kumar, S., Sharma, H. and Bhalla, P. 2011. Golomb Ruler Sequences Optimization: A BBO Approach. International Journal of Computer Science and Information Security (IJCSIS), Pittsburgh, PA, USA, Vol. 9, No. 5, pp. 63–71.

[31] Kumar S., Bansal S. and Bhalla P. 2012. Optimal Golomb Ruler Sequence Generation for FWM Crosstalk Elimination: A BB–BC Approach. In Proceedings of 6th International Multi Conference on Intelligent Systems, Sustainable, New and Renewable Energy Technology and Nanotechnology (IISN–2012), Institute of Science and Technology Klawad–133105, Haryana, India, pp. 255–262.

[32] Bansal S., Kumar S. and Bhalla P. 2013. A Novel Approach to WDM Channel Allocation: Big Bang–Big Crunch Optimization. In the proceeding of Zonal Seminar on Emerging Trends in Embedded System Technologies (ETECH-2013) organized by The Institution of Electronics and Telecommunication Engineers (IETE), Chandigarh Centre, Chandigarh, pp. 80–81.

[33] Bansal, S. and Singh, K., 2014. A Novel Soft–Computing Algorithm for Channel Allocation in WDM Systems. International Journal of Computer Applications (IJCA), Vol. 85, No. 9, pp. 19–26.

[34] Bansal, S., Chauhan, R. and Kumar, P., 2014. A Cuckoo Search based WDM Channel Allocation Algorithm. International Journal of Computer Applications (IJCA), Vol. 96, No. 20, pp. 6–12.

[35] Jain, P., Bansal, S., Singh, A. K. and Gupta, N., 2015. Golomb Ruler Sequences Optimization for FWM Crosstalk Reduction: Multi–population Hybrid Flower Pollination Algorithm. Progress in Electromagnetics Research Symposium (PIERS), Prague, Czech Republic, pp. 2463–2467.

[36] Horn, J., Nafbliotis, N., and Goldberg, D. E. 1994. A Niched Pareto Genetic Algorithm for Multiobjective Optimization. Evolutionary Computation, 1994. IEEE World Congress on Computational Intelligence, Proceedings of the first IEEE Conference on, Vol. 1, pp 82-87.

[37] Yang, X.–S., Karamanoglu, M., and He., X. S. 2014. Flower Pollination Algorithm: A Novel Approach for Multi-objective Optimization. Engineering Optimization, Vol. 46, Issue 9, pp. 1222-1237, doi: 10.1080/0305215x.2013.832237.

[38] Dimitromanolakis, A. 2002. Analysis of the Golomb Ruler and the Sidon Set Problems, and Determination of Large, Near-Optimal Golomb Rulers. Master's Thesis, Department of Electronic and Computer Engineering, Technical University of Crete.

[39] Dollas, A., Rankin, W. T., and McCracken, D. 1998. A New Algorithm for Golomb Ruler Derivation and Proof of the 19 Mark Rulers. IEEE Transactions on Information Theory, Vol. 44, No. 1, pp. 379–382.

[40] Project OGR. http://www.distributed.net/OGR.

[41] Cotta, C., Dotu, I., Fernandez, Antonio J., and Hentenryck, Pascal V. 2007. Local Search-Based Hybrid Algorithms for Finding Golomb Rulers. Kluwer Academic Publishers, Boston, Vol. 12, Issue 3, pp. 263–291.

[42] http://mathworld.wolfram.com/PerfectRuler.html

[43] http://mathworld.wolfram.com/GolombRuler.html

[44] Afshar, M. H., and Motaei, I. 2011. Constrained Big Bang-Big Crunch Algorithm For Optimal Solution of Large Scale Reservoir Operation Problem, *International* Journal of Optimization In Civil Engineering, pp. 357–375.

[45] X.-S. Yang, A New Metaheuristic Bat-Inspired Algorithm, in: *Nature Inspired Cooperative Strategies for Optimization (NISCO 2010)* (Eds. J. R. Gonzalez et al.), Studies in Computational Intelligence, Springer Berlin, 284, Springer, 65-74 (2010).

[46] Ahmadi, S. and Sedighizadeh, M. 2014. An Efficient Hybrid Big Bang-Big Crunch Algorithm for Reconfiguration of Distribution System for Loss Reduction, in Conference and exhibition on Electricity Distribution, Vol 14-E-aaa-0000.

[47] Erol, O. K. and Eksin, I. 2006. A New Optimization Method: Big Bang–Big Crunch. Advances in Department, IBM Research. Engineering Software, Vol.37, pp. 106–111.

[48] Yesil, E. and Urbas, L. 2010. Big Bang–Big Crunch Learning Method for Fuzzy Cognitive Maps, World Academy of Science, Engineering and Technology 71, pp. 815–824.

[49] Price, K., Storn, R. and Lampinen. J. 2005. Differential Evolution–A Practical Approach to Global Optimization. Springer, Berlin, Germany.

[50] Storn, R., and Price, K. V. 1997. Differential Evolution—A Simple and Efficient Heuristic for Global Optimization Over Continuous Spaces. Journal of Global Optimization, Vol. 11, No. 4, pp. 341–359.

[51] http://in.mathworks.com/help/matlab/index.html. Shearer, J. B., 2001. Golomb Ruler Table. Mathematics

8. APPENDIX-A

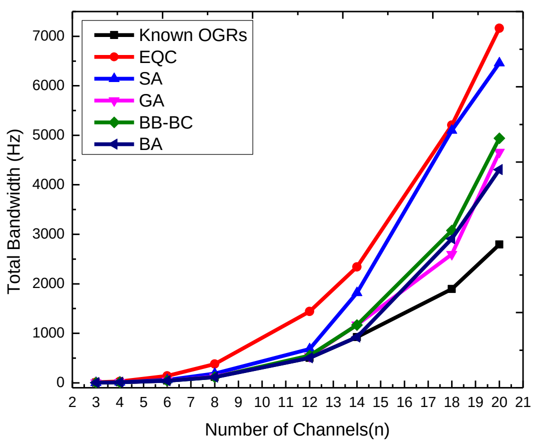
Table 2: Performance Comparison of proposed BAT Algorithm with Known OGR, EQC, SA, GA and BB–BC in terms of Ruler Length, Total Bandwidth, and Average CPU Time

n	**Known OGRs [15], [20], [38]–[42], [54]–[56]**		**ALGORITHMS**													
			Conventional Algorithms				**Existing Nature –Inspired Algorithms**						**Proposed Algorithm**			
			EQC [2], [13], [24]		**SA [2], [13], [24]**		**GA [28]**			**BB –BC [31], [32]**			**BA**			
	RL	***TBW (Hz)***	***RL***	***TBW (Hz)***	***RL***	***TBW (Hz)***	***RL***	***TBW (Hz)***	***Average* CPU time (Sec.)**	***RL***	***TBW (Hz)***	***Average CPU time (Sec.)***	***RL***	***TBW (Hz)***	***Average* CPU time *(Sec.)***
3	3	4	6	10	6	4	3	4	0.000	3	4	0.000	3	4	0.000
4	6	11	15	28	15	28	6 7	11	0.001	6 7	11	0.000	6 7	11	0.000
5	11	25 28	—	—	—	—	12 13	23 25 29	0.021	11 12	23 25	0.009	11 12 13	23 25	0.001
6	17	44 47 50	45	140	20	60	17 18 21	42 44 45	0.780	17 18	42 44	0.659	17 18	42 44	0.0512

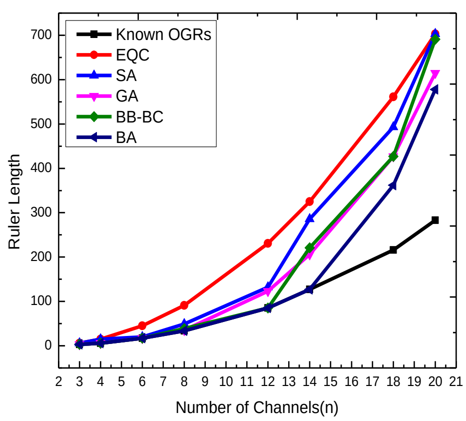
7	25	77 81 87 90 95	—	—	—	—	27 28 29 30 31 32	73 78 79 80 83 86 95	1.120	25 26 28 30	73 74 77 81	1.170	25 28	74 77 81 90	0.0870
8	34	117	91	378	49	189	35 41 42 45 46	121 126 128 129 131 133	1.241	39 41 42	113 118 119	1.210	34 39	113 117	0.1390
9	44	206	—	—	—	—	52 56 59 61 63 65	192 193 196 203 225	1.711	44 45 46 61	179 248 253 262	1.698	44	206	1.1770
10	55	249	—	—	—	—	75 76	283 287 301	5.499e+01	77	258	5.450e+01	55	249	3.120e+01
11	72	386 391	—	—	—	—	94 96	395 456	7.200e+02	72 105	377 490 456	6.990e+02	72	386	4.665e+02
12	85	503	231	1441	132	682	123 128 137	532 581 660	8.602e+02	85 91	550 580 613	7.981e+02	85	503	5.648e+02
13	106	660	—	—	—	—	203 241	1015 1048	1.070e+03	110 113	768 753	1.020e+03	106	660	8.436e+02
14	127	924	325	2340	286	1820	206 228 230	1172 1177 1285	1.028e+03	221	1166	1.021e+03	127	924	0.981e+03
15	151	1047	—	—	—	—	275 298	1634 1653	1.440e+03	267	1322	1.291e+03	151	1047	1.090e+03
16	177	1298	—	—	—	—	316	1985	1.680e+03	316	1985	1.450e+03	177	1298	1.158e+03
17	199	1661	—	—	—	—	355	2205	5.048e+04	369	2201	4.075e+04	199	1661	3.320e+03
18	216	1894	561	5203	493	5100	427 463	2599 3079	6.840e+04	427	3079	5.897e+04	427	3079	3.880e+04
19	246	2225	—	—	—	—	567 597	3432 5067	8.280e+04	584	4101	7.158e+04	467	3337	6.390e+04
20	283	2794	703	7163	703	6460	615 673 680 691	4660 4826 4905 4941	1.12428e+05	691	4941	1.0012e+05	578 615	4306 4660	7.110e+04

Table 3: Effect of Iterations on Total Bandwidth Generated by Proposed algorithm for Various Marks/channels

Iterations	*TOTAL BANDWIDTH* (Hz)												
	n=5	*n=6*	*n=7*	*n=8*	*n=9*	*n=10*	*n=11*	*n=12*	*n=13*	*n=14*	*n=16*	*n=19*	*n=20*
2	23	42	74	117	342	382	689	870	1156	2187	2519	5535	6435
50	23	42	73	113	206	279	593	711	923	2081	2519	5535	6435
100	23	42	73	113	176	249	498	682	904	2081	2484	5454	6435
150	23	42	73	113	176	249	386	520	886	1937	2407	5292	6435
200	23	42	73	113	176	249	386	503	823	1775	2347	4505	5361
250	23	42	73	113	176	249	386	503	794	1332	2149	4458	5361
350	23	42	73	113	176	249	386	503	660	1332	2149	4369	5055
400	23	42	73	113	176	249	386	503	660	1219	1985	4369	5055
500	23	42	73	113	176	249	386	503	660	1159	1958	4101	4941
600	23	42	73	113	176	249	386	503	660	924	1958	4101	4859
800	23	42	73	113	176	249	386	503	660	924	1298	3337	4660
1000	23	42	73	113	176	249	386	503	660	924	1298	3337	4306
1200	23	42	73	113	176	249	386	503	660	924	1298	3337	4306
1500	23	42	73	113	176	249	386	503	660	924	1298	3337	4306
1800	23	42	73	113	176	249	386	503	660	924	1298	3337	4306
2000	23	42	73	113	176	249	386	503	660	924	1298	3337	4306



(a)



(b)

Fig 4: The propose BAT algorithm exhibits the significant reduction in (a)) total occupied optical bandwidth and (b)ruler length in comparison to the existing algorithms i.e. Known OGR, EQC, SA, GA and BB-BC

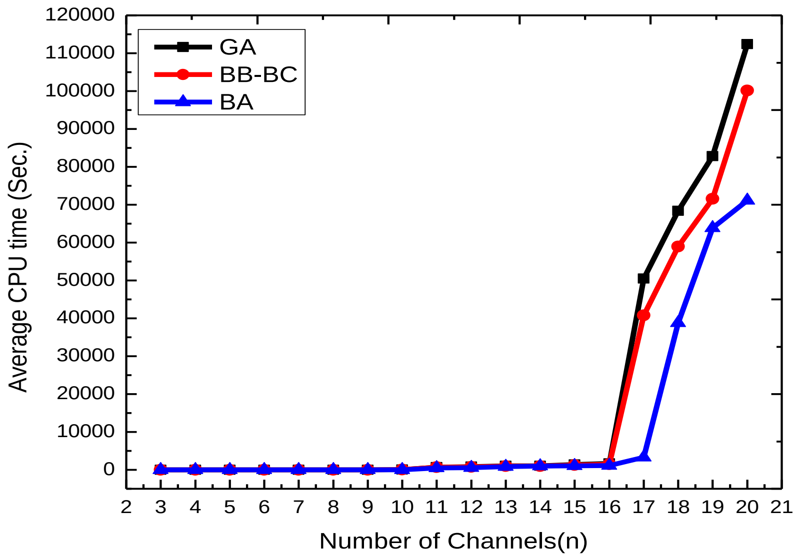


Fig 5: The proposed BAT algorithm exhibits the significant reduction in average CPU time in sec. in comparison to the existing algorithms i.e. GA and BB-BC

9. APPENDIX–B

The table below illustrates the optimal Golomb ruler sequences generated by the proposed BA for various marks:

Table 4. Optimal Golomb Ruler Sequences Generated By Proposed BA

n	*Length*	*Position of marks*
3	3	0 1 3
4	6	0 1 4 6
	7	0 1 3 7
5	11	0 1 4 9 11
	12	0 1 3 7 12
	13	0 1 4 6 13
6	17	0 1 4 10 12 17
	18	0 1 3 8 12 18
7	25	0 2 3 10 16 21 25
	26	0 1 7 9 12 22 26
	27	0 1 5 7 15 18 27
	28	0 1 3 8 12 22 28
8	34	0 1 4 9 15 22 32 34
	39	0 1 3 8 14 18 30 39
9	44	0 3 9 17 19 32 39 43 44
	47	1 2 4 11 17 22 36 44 48
	49	1 5 11 12 20 33 36 38 50
10	55	0 1 6 10 23 26 34 41 53 55
11	72	0 1 4 13 28 33 47 54 64 70 72
	72	0 1 9 19 24 31 52 56 58 69 72

	103	**1 3 4 12 17 24 34 49 53 77 104**
12	**85**	**0 2 6 24 29 40 43 55 68 75 76 85**
13	**106**	**0 7 8 17 21 36 47 63 69 81 101 104 106**
14	**127**	**0 5 28 38 41 49 50 68 75 92 107 121 123 127**
15	**151**	**0 6 7 15 28 40 51 75 89 92 94 121 131 147 151**
16	**177**	**0 1 4 11 26 32 56 68 76 115 117 134 150 163 168 177**
17	**369**	**2 5 6 14 21 32 49 54 108 110 180 190 222 247 253 337 371**
18	**445**	**0 1 3 17 29 35 71 98 102 122 147 160 212 235 256 295 338 445**
19	**467**	**3 6 25 26 51 53 58 66 104 135 139 153 243 277 319 348 402 459 470**
20	**578**	**4 8 22 27 44 47 103 110 118 131 168 180 319 354 363 364 405 432 525 582**