

Energy-Aware Task Scheduling Using Hybrid Firefly-BAT (FFABAT) in Big Data

M. Senthilkumar

*School of Information Technology and Engineering, VIT University, Vellore, India
E-mail: mosenkum@gmail.com*

Abstract: *In modern times there is an increasing trend of applications for handling Big data. However, negotiating with the concepts of the Big data is an extremely difficult issue today. The MapReduce framework has been in focus recently for serious consideration. The aim of this study is to get the task-scheduling over Big data using Hadoop. Initially, we prioritize the tasks with the help of k-means clustering algorithm. Then, the MapReduce framework is employed. The available resource is optimally selected using optimization technique in map-phase. The proposed method uses the FireFly Algorithm and BAT algorithms (FFABAT) for choosing the optimal resource with minimum cost value. The bat-inspired algorithm is a meta-heuristic optimization method developed by Xin-She Yang (2010). This bat algorithm is established on the echo-location behaviour of micro-bats with variable pulse rates of emission and loudness. Finally, the tasks are scheduled with the optimal resource in reducer-phase and stored in the cloud. The performance of the algorithm is analysed, based on the total cost, time and memory utilization.*

Keywords: *Map reduce framework, task scheduling, firefly algorithm, BAT Algorithm, Hadoop, k-Means clustering, Hadoop, HDFS.*

1. Introduction

Today, we are all very active in social media, where we perceive exponential growth in the data being presented. The unimaginable magnitude in the growth of the data poses serious challenges of cutting-edge businesses, such as Google, Facebook, Twitter, LinkedIn, etc. The analytical representations for Hadoop with MapReduce workloads address this issue considering a small segment of the whole data set at a time. The capabilities of the MapReduce model under substantial workloads that process exponentially incremental data size, are considerable [1]. Apache produced Hadoop open source with a Java programming platform to support large volumes of data processing in a distributed environment. Hadoop is highly fault-tolerant with large quantities of data sets. Hadoop has name node, data node and task tracker performing the job execution in as many number of nodes as decided by the cluster [27]. The social media and their applications need to practice the Big data in a

proficient manner on regular basis for managing the large data. An efficient procedure to categorize the Big data from exhausting e-mail firefly is through a naive Bayes classifier. The proposed MapReduce framework method comprises two phases, the training phase and the testing phase. Primarily, the input Twitter records are given to the progression to select the appropriate feature for data classification. The traditional firefly algorithm is applied, and the optimized feature space is accepted for the finest fitting results [15].

Big data with scheduling becomes an active area for large scale research which is in progress on various scheduling protocols with optimization of the data. Hadoop framework is a very wide spread and largely used framework in the distributed data processing. Various scheduling algorithms of the MapReduce model, using Hadoop differ in design and behavior and are employed for handling several issues, such as data locality, awareness with resource, energy and time [8]. Hadoop mechanisms, based on a MapReduce model, deliver efficient processing of the Big data. Hadoop schedulers are precarious elements for providing desired performance levels. A scheduler allocates MapReduce tasks to the available resources in the Hadoop cluster. The performance of Hadoop scheduling schemes FIFO and Fair sharing are compared to the Classification and Optimization based Scheduler for Heterogeneous Hadoop (COSHH) scheduler. A hybrid method is introduced for selection of suitable scheduling algorithms for accessible and heterogeneous Hadoop systems with a number of requested jobs, considering the available resources [3].

The scheduling scheme is to minimize the deadline of jobs while processing a large volume of multimedia files, such as video and image files in MapReduce frameworks. The scheme is introduced to check the data locality for processing the allocated jobs within the limits of the deadline and to cogitate whether I/O load and requirements are fulfilled. If tasks are executed on a node with too much I/O load, the multimedia data from the replica node must be consumed to improve the processing speed of the job task. If the presented nodes are not originated due to the expected job completion time beyond the time limit, the job tasks in nodes, whose time limits are obtainable, are hesitated temporarily for reducing the job completion time. The performance results of the proposed scheme prove a gross reduction in the completion time by 13.8 % and an improvement of the deadline success ratio by 11%, compared to those of the existing schemes [9]. Cloud computing with resource scheduling of the nodes employ the firefly and a genetic algorithm to optimize the solution, which is generated initially. Using firefly with genetic algorithm demonstrates improved efficiency in performance of scheduling the resources in an optimized manner. The tasks are considered to shorten the time for completion of the task and to improve the overall processing capacity of the system [34].

The incredible quantity of energy consumption results in high operating costs, reduces the yield on investment and contributes towards energy-aware environment. However, there is a dire need for the energy-aware schedules based on resources spontaneously, when the rate of energy consumption becomes a major parameter. A Self-Optimization of Cloud Computing Energy-efficiency Resources (SOCCER) is introduced for energy-efficient scheduling of resources in data centers. The performance of the SOCCER has been evaluated in a real-time environment, and the

results demonstrate an increased level of performance of energy consumption and operate these resources in an optimal way [35]. Heuristic algorithms are generally suggested through stimulation from nature. For example, Particle Swarm Optimization (PSO) has been encouraged by the social behavior patterns. Bat algorithm is a heuristic algorithm suggested by Yang in 2010 and has been motivated by an echo-location, which directs the bats' activities during their flight and hunting even in complete darkness. EBA is introduced for local and global search features of bat algorithm for enhanced efficacy. EBA method is compared to the recent studies in the real-world complexities and is established to be a more efficient technique than the revisions appropriate to the other works of this sort of problems [6].

Firefly Algorithm has been designated as an anticipated bio-inspired method to perform load balancing and to decrease energy consumption in the data center. The results are compared to Particle Swarm Optimization Algorithm (PSO) and the energy disbursed in case of firefly algorithm is found to be less than the energy disbursed in PSO algorithm [16, 23]. Optimal solutions are offered through the bio-inspired algorithms in terms of time for the increasing challenges in data generation and applications of greater dimensions and highly tedious in nature. The efficacy of such specific algorithms as the particle swarm, ant colony optimization, artificial bee colony, bacterial foraging, cuckoo search, firefly algorithm, bat algorithm, genetic algorithm, flower pollination and artificial plant optimization algorithm, has been established. This review intends to pave the path for future studies to choose algorithms based on fitment [17].

Prediction and diagnosis of different diseases are two important mechanisms of health sciences. The cumulative expansion of science has controlled the use of decision-support systems to help surgeon's beneficial policies, such as the prediction and identification of breast cancer as the furthestmost common cancer concerning women, using artificial intelligent systems. A neural fuzzy adaptive system is used to acquire prediction and estimation. This proposed system in comparison with the other existing techniques has demonstrated better performance and high precision in prediction [18]. The growth of data has resulted in the evolution of the Big data environment and the MapReduce parallel programming model is introduced for handling the volume of data. The focus is on enhancement in utilization and energy saving through association, while MapReduce empowers a large-scale data analysis. The exodus of data and services provides more flexibility in surroundings, where data locality might not have a considerable impact. An energy efficiency assessment of Hadoop on physical and virtual clusters in different configurations is performed [31].

2. Related works

Big data is extremely valuable for creating productivity in businesses and taking an evolutionary step forward in scientific disciplines, which gives us several prospects of accomplishing great improvements in diverse fields. It is certain that all future struggles in business productivity and technologies to be adopted come together into the Big data explorations [20]. Here, FCO firefly colony optimization, inspired by an ant colony optimization algorithm, is introduced. The approach is distributed and

constructive meta-heuristic in nature to use positive feedback with greedy solutions, easily overcoming the low-quality solutions. BPP (Bin Packing Problem) is employed for measuring the performance and the results demonstrate the accomplishment of a feasible solution [7].

The commercial release of load dispatch problem has been defined and realized to deal with the optimization of these two differing objectives, namely, the minimization of cost and giving out of generating units. Several meta-heuristic optimization algorithms have been developed to address these issues. Dr. Xin-She Yang Firefly has introduced a technologically advanced novel meta-heuristic nature-inspired algorithm, namely FireFly Algorithm (FFA) in 2007 [10, 41], inspired by the blinking actions of fireflies. Artificial Bee Colony (ABC) Algorithm is another meta-heuristic optimization algorithm based on the intelligent behavior of honey bee swarm. Bat Algorithm (BA) is the most recently developed algorithm on the concept of the echo-location of bats [11, 22]. The validity of the proposed algorithm is authenticated through the results of a trade education carried out on six traditional structural optimization problems considered from the literature. The unique search features executed in FA are analyzed and their validity and applicability for future research work are discussed [28, 29]. BA is primarily based on the echo-location behavior of the bats and is substantiated after an exhaustive formulation and explanation of its operation, using eight familiar nonlinear engineering optimization processes for task optimization [12, 13].

Yet another novel meta-heuristic algorithm is formulated by fusing Levy flights with the search approach via the firefly algorithm. Numerical readings and results recommend that the proposed Levy-flight firefly algorithm is more efficient than the existing meta-heuristic algorithms [32, 33]. Big data analytics for smart energy management have the major aspects of power generation management, renewable energy management, quality management and collaborative operation, as well as Demand-Side Management (DSM) for serious consideration. The challenges of Big data in smart energy management in IT infrastructure include data gathering and domination, data combination and sharing, processing and analysis and so on [14]. The linear and dynamic programming methodologies are used to model the algorithm. The evaluation of the baseline shortest path algorithm with the smallest number of nodes traversed is done using the physical network topology of ISP [24].

The firefly algorithm emerges as a significant tool of Swarm Intelligence that has been applied in almost all areas of optimization. Problems from several areas have been effectively solved using the firefly algorithm with its variants. The original firefly algorithm needs to be modified in the context of its application for fixing diverse problems [1]. Swarm intelligence is emerging as a powerful technique for appropriate optimization. Bat algorithm has been cross-bred with discrepancy evolution strategies. A novel approach of applying bat algorithm with swarm intelligence is introduced. This hybridization was found to have considerably improved the original bat algorithm in addition to producing very encouraging results on standard benchmark functions [2]. Classification with bat algorithm for updating the weights of a Functional Link Artificial Neural Network (FLANN) classifier is proposed. The comparison of FLANN with PSO-FLANN proves that the proposed

classification technique is faster than FLANN and PSO-FLANN [5]. A data clustering technique used for data analysis is also used in many fields, such as data mining, image analysis and pattern recognition.

k-Means clustering is a common and simple approach for data clustering. An algorithm is then drawn-out to use k-Means clustering to refine centroids of the clusters. A new hybrid algorithm K-FA is introduced for enhanced efficacy. The experimental results have established the accuracy and capability of the K-FA algorithm for data clustering [4]. A nature inspired algorithm, when incorporated into a clustering algorithm, provides global optimal elucidation. The algorithms are evaluated through the parameters of the fitness function and CPU time per execution [19]. Particle Swarm Optimization is a mimic activities-based algorithm used to present the connectivity attitude in the centroid-based clustering algorithm. It provides optimum centroid for forwarding to better clusters. PSO is employed for discovery of initial centroids and K-means, for finding better clusters [30]. Tabu search with a hybrid bat is introduced to improve the performance obtained by applying a standard bat algorithm. The proposed algorithm transforms the phase of selection offering a new solution in the standard bat algorithm by the procedure of tabu search. The method is to maximize the density of the links [25]. The method is applied for extracting key features from each cardiac beat using an improved bat algorithm. This improved bat algorithm extracts the best features and then integrates them to the input of the classifier [26].

FireFly-Bat with optimized Rule-Based Fuzzy Logic (FFABAT-RBFL) prediction is designed to optimize the fuzzy rules and membership functions for classification of the data. FFABAT is an optimization algorithm which combines the optimization techniques of both bat and FireFly (FF) [21, 36]. Social Network-based Swarm Optimization algorithm (SNSO) is applied for analyzing the unconstrained single objective optimization problems. The execution of SNSO is associated with seven other swarm algorithms on twelve familiar benchmark functions [37]. Population-based algorithms integrated with data mining techniques and more efficient algorithms are proposed in order to understand better and to solve the real-time Big data analytics. The population-based algorithms are analyzed through the data analytics along with the optimization process to provide a critical entity in them [38]. A new scheme is presented to aid the scheduler in ascertaining the nodes on which laggards can be executed. The variations of the scheme make use of the resource utilization and network information of cluster nodes to find the most optimal node for scheduling the projected copy of a slow task. The analysis establishes a performance enhancement over 27% pertaining to the whole execution time in comparison to Hadoop Fair Scheduler (HFS) [39].

3. Proposed method

Task scheduling plays a vital role in Big data. Scheduling of jobs cannot be achieved on the basis of a single criteria but only in consideration of several rules and regulations, which can be comprehended as a settlement between users and service providers. The intention of this work is to improve the performance of task scheduling

in Big data. The main objective is to schedule the task over consistent machines in such a way that the execution time can be reduced to a manageable level for each task. In Big data resources, the cost of each task is dissimilar to the others and as a result scheduling of user tasks in Big data is not the same as in traditional scheduling methods. This work intends to further focus on schedule task groups in Big data, where the resources have dissimilar resource costs and computation performance. However, first, the tasks are prioritized based on their energy consumption rate. Here, the prioritization is achieved with the help of clustering techniques. The K-means clustering algorithm is adopted for task prioritization. Considering the priority, tasks are allocated optimally, based on the minimum cost and energy consumption rate of the available resources in the Hadoop cluster. The hybrid method of FireFly Algorithm (FA) with BAT (FF-BAT) is employed here for accomplishing the optimization.

4. Firefly algorithm

All fireflies are unisex in nature and one firefly is fascinated by the other fireflies despite their gender. Attractiveness is associated with their brightness which could decrease with the increase in the mutual distance. Thus, for any two flashing fireflies, the less bright one shifts towards the brighter firefly. This shifting of an exacting firefly to a brighter one is mostly at random. The brightness of a firefly is exaggerated or intensified by the landscape of the objective function to be optimized for achieving the performance of task scheduling.

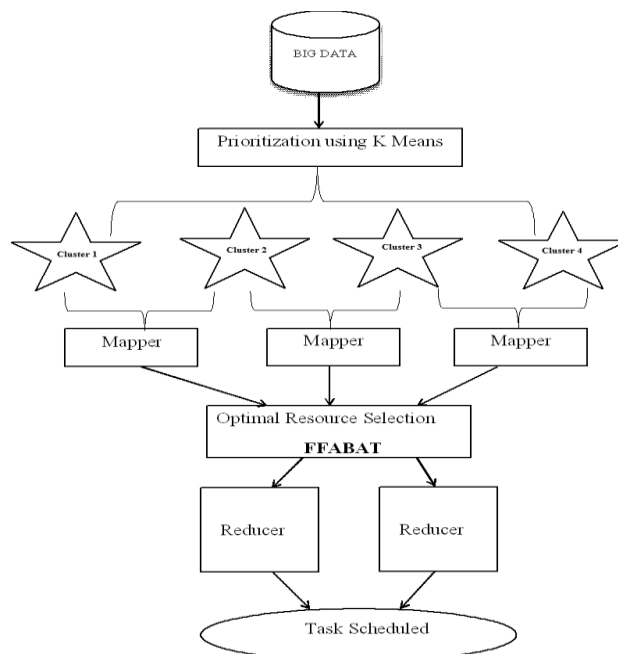


Fig. 1. Proposed work schedule of task Firefly with BAT

The implemented method has two main phases namely,

- Prioritization of task,
- Map-reduce framework.

4.1. Prioritization of task

Consider the dataset is D having the n number of attributes, $A = A_1, A_2, \dots, A_n$. First, we prioritize the dataset such as (P_1, P_2, \dots, P_n) . Here, the task prioritization is done using k-Means clustering algorithm for clustering the input data.

The parameter for clustering the task is to consider the level of energy consumption. Each data point reduces the mean squared Euclidean distance to its adjoining center. Now, we have a good mechanism to produce some Hadoop clusters by k-Means. However, by grouping the number of tasks based on the energy consumption level, k value can be fixed. The step-by-step procedure of k-Means clustering algorithm is demonstrated in the section below:

Step 1. First, initialize the result, then select the cluster value of k . Next, choose the centroid of k clusters in an arbitrary manner.

Step 2. Calculate the distance between the cluster centroids to respective object. Then, employ the Euclidean distance to locate the distance of object centroids.

$$(1) \quad \text{Euclidean distance} = \sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2}.$$

Step 3. Determine the **Euclidean** distance from each task (object) to each cluster. Then, choose each object to the adjacent cluster based on the minimum distance value.

Step 4. Update the centroid value.

Step 5. Again compute the distance of all tasks (objects) to the novel centroid value.

Step 6. Similar to Step 3. Allocate each task (object) on the basis of the minimum distance.

Step 7. Steps 4, 5 and 6 are continued until no task (object) moves clusters.

The general process of k-Means clustering algorithm is revealed in the flowchart Fig. 2.

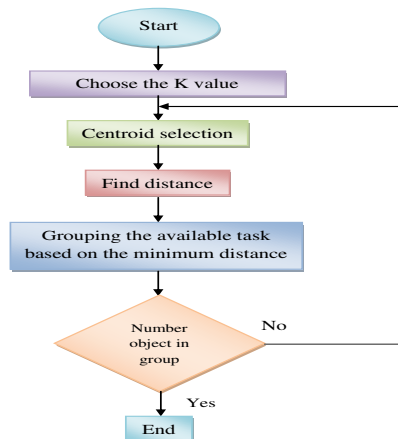


Fig. 2. The flowchart for k-Means clustering

4.2. MapReduce framework

Map the corresponding data with prioritization from k-Means clustering. The functionality of the Big data with clustering such application framework generates large volumes of transitional data sets. MapReduce framework is used for writing apps of large amounts of data from the large cluster. The work allocation of tasks to MapReduce framework proves better in the allotment of task scheduling of Hadoop cluster. The MapReduce framework consists of input data set, which is split into several blocks of data. Then, each block is handled in Map function in the map-phase, followed by reducing function in the reduce-phase, which produce the final results. MapReduce is used for the large computational work for easy re-execution of the failed tasks and parallelizing of the tasks.

Map. Job Tracker splits the map tasks and reduce tasks to idle Task Trackers. Map function sends the input data as HDFS format and performs map operation on the cluster. A set of transitional key-value pairs are generated during MapReduce phase. These results are written to disk.

Reduce. Reduce function receives transitional key-value pairs and perform reduce operation. It updates the final results to the output file. These output files written by the Reducers are then kept in Hadoop Distributed File System (HDFS) for the end user or for performing other MapReduce tasks.

4.3. Task scheduling

In task scheduling, the available resources are optimally selected based on the optimization technique. The optimization is done with the help of a hybrid technique using firefly algorithm with bat algorithm. Optimal resource is selected from the available results and the task is allocated to the corresponding resource with minimum cost. The detailed description of firefly algorithm and bat algorithm is explained below:

Proposed Algorithm FA with BAT (FFABAT)

01: Start the Hadoop cluster

02: Input the data, number no of nodes, data size

03: Initialize the task tracker data node and name node

04: Repeat for every new task

05: Firefly Algorithm

Initial population of fireflies: P

Objective function of $f(P)$,

Find out the light intensity of firefly N of at via

While (the termination criteria is not satisfied)

For $a= 1 : N$

For $b= 1 : N$

If

Then move Firefly x towards Firefly y

End if

Attractiveness varies with distance r via

Calculate new solutions and update light Intensity

06: BAT Algorithm

Allocate a fitness value to each task
 Choose the best and most fitting task
 End of b ;
 Check the ranges of the given solutions
 Update them as appropriate
 Rank the fireflies
 Find the current best based on BAT
 End for a ;
 End while
 Find the firefly with the highest light intensity among all fireflies
 End FA
 07: Evaluation of the solutions
 08: Until (the termination criteria is not satisfied)
 09: Results estimate
 10: End

5. Results and discussion

In the Big data Social networking, the data is taken as input for the experiment in a Hadoop environment with MapReduce framework. The scheduling scheme is applied to Big data in a series of experiments, performed on a PC with Windows 2008 server OS.

5.1. Experiment process and results

The following is the Big data scheduling of the task on MapReduce framework: First, the task is prioritized with k-Means clustering. Then FFABAT is applied to MapReduce framework in Hadoop cluster. The existing resource is optimally caused by k-Means clustering with a firefly algorithm and bat algorithm (FF-BAT) in Hadoop Map-phase. In the last stage, the tasks are scheduled with the optimal resource in reducer-phase and the results are updated in HDFS.

5.2. Performance evaluation by throughput

The performance evaluation of FFABAT with task scheduling is calculated based on throughput and job-execution time. The execution of MapReduce is restrained regarding various factors. Job execution time of each job is a major parameter since it is closely associated with power consumption. The throughput of the MapReduce is estimated by using the parameters of HDFS bytes read and HDFS bytes write for a particular time. The proposed model formula

$$\text{Throughput} = \frac{(\text{hdfs}_{\text{read}} + \text{hdfs}_{\text{write}})}{\text{Time}_{\text{jobExe}}} \text{ MB per 1 s,}$$

where $\text{hdfs}_{\text{read}}$ and $\text{hdfs}_{\text{write}}$ are the numbers of bytes read from and written to HDFS, respectively, by the MapReduce job. This denotes the overall IO data rate in MB. $\text{Time}_{\text{jobExe}}$ signifies the overall job execution time.

5.3. Energy analysis process

The power consumption of the proposed task scheduling with FFABAT is depicted in Table 1. This Table defines the modifications of the energy usage proposed on the number of data records which range from 256 MB to 2 GB. The data sets in our experiments are quite adequate. This makes it possible to evaluate the effectiveness of our approach regarding the volume of data. When 256 MB is used for Big data processing, energy is consumed for word count and data analysis at 102.1 W, 114.33 W, respectively. Energy usage increases corresponding to the increase in data size. Table 1 depicts the data size, IO transfer rate, the average power usage of data analysis and word count workloads. Table 2 displays the data size, IO transfer rate, job execution time, throughput of data analysis and the word count of workloads. The proposed task-scheduling is applied (FFABAT) over Big data on Hadoop environment with a series of experiments. The experiments have been performed on a PC with Windows Server Operating system at 2 GHz dual-core PC machine with 4 GB RAM running the 64-bit version. The experimental result of the proposed work FFABAT is illustrated in Table 1.

Table 1. Analysis of power usage by using FA& FA with BAT

Workload	IO rate (MB)	Power usage (W)	
		FA	FA+BAT
256	253.7	102.1	101.45
512	507.4	130.81	112.72
1024	1014.8	152.4	142.86
2018	2029.6	163.5	153.92
256	253.7	114.33	102.4
512	507.4	151.2	143.29
1024	1014.8	173.26	159.89
2018	2029.6	192.1	179.41

Table 2. Analysis of job execution time, throughput by using FA& FA with BAT

Workload	IO rate (MB)	Execution time (s)		Throughput (MB per 1 s)	
		FA	FA+BAT	FA	FA+BAT
256	253.7	65	49	25.64	26.86
512	507.4	91	74	27.15	29.4
1024	1014.8	108	97	31.53	34.8
2018	2029.6	132	121	34.29	38.9
256	253.7	85	72	24.61	25.41
512	507.4	126	113	25.28	27.46
1024	1014.8	154	148	26.63	28.91
2018	2029.6	168	151	27.05	31.21

The results of the recurring experiments accomplished on two MapReduce applications of increasing workloads, their configurations, and improvement in throughput are represented in Figs 3, 4 and 5.

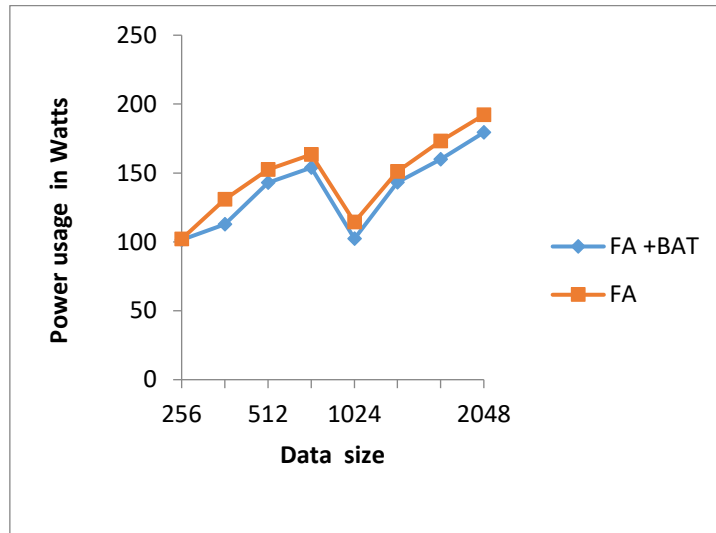


Fig. 3. Reduction of power usage for various workloads

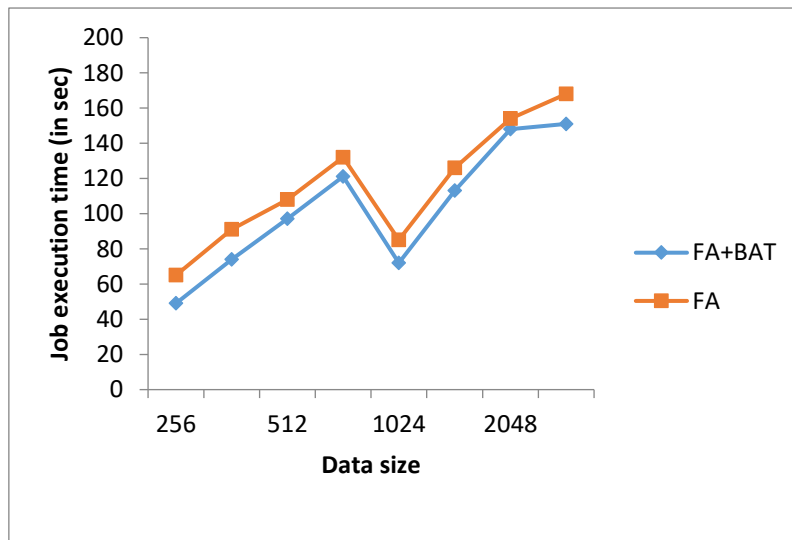


Fig. 4. Energy consumption rate of various workloads

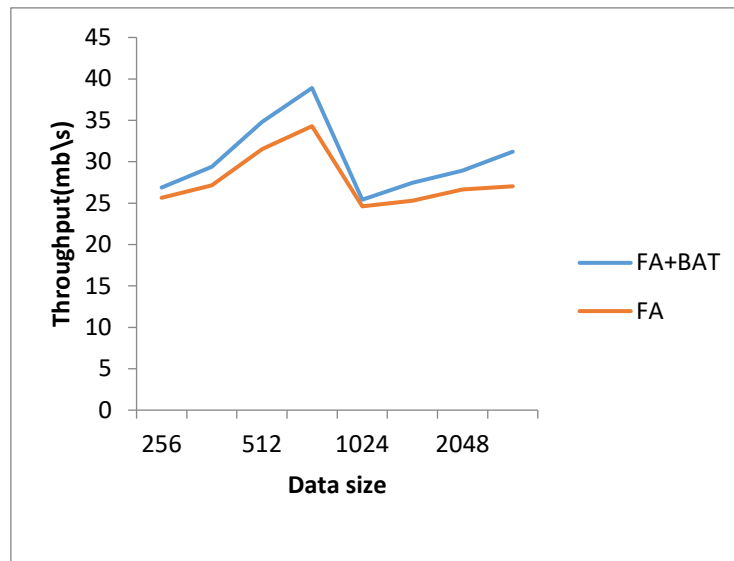


Fig. 5. Enhancement of throughput with increasing workloads

6. Conclusion

We have formulated a new model that can function on Big data in MapReduce framework. Initially, k-Means clustering is used for prioritization of the task. Then, the input is given to MapReduce framework in Hadoop cluster for execution of the jobs. In Hadoop map-phase, the existing resource is selected, using a firefly algorithm (FA) with bat algorithm optimally. Finally, the tasks are programmed with the optimal resource in Hadoop reducer-phase, and the results are updated in the HDFS. The performance of the proposed work is assessed through the power consumption and throughput. The size of the data, task execution time, energy consumption and throughput are considered for application of FA with BAT. The HDFS results are compared to FA with the hybrid algorithm (FFA with BAT). Energy optimization is achieved by using the hybrid algorithm FFA-BAT.

References

1. Fister, I., X. S. Yang, J. Brest. A Comprehensive Review of Firefly Algorithms. – *Swarm Evol. Comput.*, Vol. **13**, 2013, pp. 34-46.
2. Fister, I., D. Fister, X. S. Yang. A Hybrid Bat Algorithm. – *Elektroteh. Vestnik/Electrotechnical Rev.*, Vol. **80**, 2013, No 1-2, pp. 1-7.
3. Rasooli, A., D. G. Down. A Hybrid Scheduling Approach for Scalable Heterogeneous Hadoop Systems, 2012.
4. Hassanzadeh, T., M. R. Meybodi. A New Hybrid Approach for Data Clustering Using Firefly Algorithm and k-Means. – In: 16th CSI Int. Symp. Artif. Intell. Signal Process. (AISP'12), IEEE, 2012, No Aisp, pp. 7-11.
5. Mishra, S., K. Shaw, D. Mishra. A New Meta-Heuristic Bat Inspired Classification Approach for Microarray Data. – *Procedia Technol.*, Vol. **4**, 2012, pp. 802-806.
6. Yilmaz, S., E. U. Küçüksille. A New Modification Approach on Bat Algorithm for Solving Optimization Problems. – *Appl. Soft Comput. J.*, Vol. **28**, 2015, pp. 259-275.

7. Layeb, A., Z. Benayad. A Novel Firefly Algorithm Based Ant Colony Optimization for Solving Combinatorial Optimization Problems. – International Journal of Computer Science and Applications, Technomathematics Research Foundation, Vol. **11**, 2014, No 2, pp. 19-37.
8. Senthilkumar, M., Dr. P. Ilango. A Survey on Job Scheduling in Big Data. – Cybernetics and Information Technologies, Vol. **16**, 2016, No 3.
9. Bok, K., J. Hwang, J. Lim, Y. Kim, J. Yoo. An Efficient MapReduce Scheduling Scheme for Processing Large Multimedia Data. – In: *Multimed. Tools Appl.*, 2016.
10. Apostolopoulos, T., A. Vlachos. Application of the Firefly Algorithm for Solving the Economic Emissions Load Dispatch Problem. – *Int. J. Comb.*, Vol. **2011**, 2011, pp. 1-23.
11. Paper, C. Artificial Bee Colony. – *Firefly and Bat Algorithm in Unconstrained Optimization*, No October 2016.
12. Yang, X.-S., A. H. Gandomi. Bat Algorithm: A Novel Approach for Global Engineering Optimization. – *Eng. Comput. (Swansea, Wales)*, Vol. **29**, 2012, No 5, pp. 464-483.
13. Yang, X.-S., X. He. Bat Algorithm: Literature Review and Applications. – *Int. J. Bio-Inspired Comput.*, Vol. **5**, 2013, No 3, pp. 141-149.
14. Zhou, K., C. Fu, S. Yang. Big Data Driven Smart Energy Management: From Big Data to Big Insights. – *Renew. Sustain. Energy Rev.*, Vol. **56**, 2016, pp. 215-225.
15. Satish, K. V. R., N. P. Kavya. Big Data Processing with Harnessing Hadoop – MapReduce for Optimizing Analytical Workloads. – In: *Proc. 2014 Int. Conf. Contemp. Comput. Informatics, (IC3I'14)*, 2014, pp. 49-54.
16. Goyal, A., N. S. Chahal. Bio Inspired Approach for Load Balancing to Reduce Energy Consumption in Cloud Data Center. – *Int. Conf. Commun. Control Intell. Syst. CCIS 2015*, 2016, pp. 406-410.
17. Kar, A. K. Bio Inspired Computing – A Review of Algorithms and Scope of Applications. – *Expert Syst. Appl.*, Vol. **59**, 2016, No April, pp. 20-32.
18. Breast Cancer Detection Based on a Hybrid Approach of Firefly Algorithm and Intelligent Systems. – *Indian Journal of Fundamental and Applied Life Sciences*, Vol. **5**, 2015, pp. 468-472.
19. Agarwal, P., S. Mehta. Comparative Analysis of Nature Inspired Algorithms on Data Clustering. – In: *Proc. of 2015 IEEE Int. Conf. Res. Comput. Intell. Commun. Networks (ICRCICN'15)*, 2016, pp. 119-124.
20. Chen, C. L. P., C. Y. Zhang. Data-Intensive Applications, Challenges, Techniques and Technologies: A Survey on Big Data. – *Inf. Sci. (Ny)*, Vol. **275**, 2014, pp. 314-347.
21. Thippa Reddy, G., N. Khare. FFBAT-Optimized Rule Based Fuzzy Logic Classifier for Diabetes. – *Int. J. Eng. Res. Africa*, Vol. **24**, 2016, No October, pp. 137-152.
22. Yang, X. S., X. He. Firefly Algorithm: Recent Advances and Applications. – *Int. J. Swarm Intell.*, Vol. **1**, 2013, No 1, p. 36.
23. Yang, X.-S. Firefly Algorithms for Multimodal Optimization. – In: *Proc. of 5th Int. Conf. Stoch. Algorithms Found. Appl.*, 2009, pp. 169-178.
24. Baker, T., B. Al-Dawsari, H. Tawfik, D. Reid, Y. Ngoko. GreeDi: An Energy Efficient Routing Algorithm for Big Data on Cloud. – *Ad Hoc Networks*, Vol. **35**, 2015, pp. 83-96.
25. Hybrid Bat Algorithm for Overlapping Community Detection Overlapping Hybrid. – *IFAC-Paper Online*, 2016-Elsevier, No October, 2016.
26. Kora, P., S. R. Kalva. Improved Bat Algorithm for the Detection of Myocardial Infarction. – *Springerplus*, Vol. **4**, 2015, No 1, p. 666.
27. Senthilkumar, M., N. Manikandan, U. Senthilkumaran, R. Samy. Weather Data Analysis Using Hadoop. – *International Journal of Pharmacy and Technology*, Vol. **8**, 2016, No 4, pp. 21827-21834.
28. Gandomi, A. H., X. S. Yang, A. H. Alavi. Mixed Variable Structural Optimization Using Firefly Algorithm. – *Comput. Struct.*, Vol. **89**, 2011, No 23-24, pp. 2325-2336.
29. Parpinelli, R. S., H. S. Lopes. New Inspirations in Swarm Intelligence: A Survey. – *Int. J. Bio-Inspired Comput.*, Vol. **3**, 2011, No 1, p. 1.
30. Chaturbhuji, K. S. Parallel Clustering of Large Data Set on Hadoop Using Data Mining Techniques. – In: *Proc. of World Conference on Futuristic Trends in Research and Innovation for Social Welfare (Startup Conclave)*, IEEE, Coimbatore, India, 2016, pp. 5-8.

31. Feller, E., L. Ramakrishnan, C. Morin. Performance and Energy Efficiency of Big Data Applications in Cloud Environments: A Hadoop Case Study. – J. Parallel Distrib. Comput., Vol. **79-80**, 2015, pp. 80-89.
32. Yang, X. Research and Development in Intelligent Systems XXVI. – Development, Springer, 2010.
33. Overview, A., I. Applications, B. S. Mienda, A. Yahya, I. A. Galadima, M. S. Shamsir. Research Journal of Pharmaceutical, Biological and Chemical Sciences. – Res. J. Pharm. Biol. Chem. Sci., Vol. **5**, No 388, pp. 388-396.
34. Chen, J. Research on Resource Scheduling in Cloud Computing Based on Firefly Genetic Algorithm. – Int. J. of Grid and Distributed Computing, Vol. **9**, 2016, No 7, pp. 141-148.
35. Singh, S., I. Chana, M. Singh, R. Buyya. SOCCER: Self-Optimization of Energy-Efficient Cloud Resources. – Cluster Comput., Vol. **19**, 2016, No 4, pp. 1787-1800.
36. Satapathy, S., A. Naik. Social Group Optimization (SGO): A New Population Evolutionary Optimization Technique. – Complex Intell. Syst., Vol. **2**, 2016, No 3, pp. 173-203.
37. Liang, X., W. Li, P. Liu, Y. Zhang, A. A. Agbo. Social Network Based on Swarm Optimization Algorithm. – Semantic Scholar, 2015, pp. 360-365.
38. Cheng, S., B. Liu, T. O. Ting, Q. Qin, Y. Shi, K. Huang. Survey on Data Science with Population-Based Algorithms. – Big Data Anal., Vol. **1**, 2016, No 1, p. 3.
39. Brahmwar, M., M. Kumar, G. Sikkha. Tolhit – A Scheduling Algorithm for Hadoop Cluster. – Procedia Comput. Sci., Vol. **89**, 2016, pp. 203-208.
40. Senthikumar, M., P. Ilango. Big Data Optimization for Social Networking Tweet. – International Journal of Soft Computing, Vol. **11**, 2016, No 5, pp. 305-311.
<https://www.scopus.com/inward/record.uri?eid=2-s2.0-85011357358&doi=10.3923%2fijscmp.2016.305.311&partnerID=40&md5=aa1f3c8b978a2ec629ac50f180a2f3ac>
DOI: 10.3923/ijscmp.2016.305.311
41. Senthikumar, M., P. Ilango. Analysis of DNA Data Using Hadoop Distributed File System. – Research Journal of Pharmaceutical, Biological and Chemical Sciences, Vol. **7**, 2016, No 3, pp. 796-803.

Received 14.12.2017; Second Version 24.03.2018; Accepted 02.06.2018