International Journal of u- and e- Service, Science and Technology
Vol.9, No. 7 (2016), pp.213-226
http://dx.doi.org/10.14257/ijunesst.2016.9.7.22

Improved NEH-Heuristic Job Scheduling for An Optimal System Using Meta-Heuristic GA–INSMG

Jeeva Rathanam G[1] and Dr. A. Rajaram[2]

[1]Research Scholar, [2]Associate Professor
[1,2]Department of Electronics and Communication Engineering
[1,2]SRS College of Engineering and Technology, Salem, Tamil Nadu, India
[1]jeevag@mitindia.edu, [2]gct143@gmail.com

Abstract

The application of simulation-optimization technique is merged with viable reality for computation of the scheduling process in a cloud environment. The optical systems are needed to be scheduled for requirement satisfaction of IOT (Internet of Things) services and demand services of QoS. Normally in the rule based scheduling algorithm are widely used to schedule the sequence process efficiently. The significance of environmental factors is simulated with meta-heuristics scheduling and processed under an accurate hypothesis of stochastic processing. In order to overcome the issues of scheduling a heuristic approach is proposed with the function to obtain a feasible high quality scheduling solution. The proposed heuristic algorithm, named INSMG, is performed and analyzed with the other heuristic algorithm. The analysis result shows the proposed algorithm efficiency and possible alternative for solving scheduling issues. This paper aims to provide a more possibilities of sequences, minimize makespan and solutions to solve the scheduling problems. Moreover the scheduling process is evaluated by the improved meta-heuristic GA for better schedules to prove the efficiency and better performances than the existing system.

Keywords: *Job Scheduling; NEH; Genetic Algorithm (GA); Meta-heuristic Approach; Optimal System; Makespan*

1. Introduction

In the past decades, the Flow shops Scheduling Problem were optimized to solve the problem and it processes the scheduling with jobs in the same order. However, the real-world scheduling in distributed environments are optimized in order to realize improved quality, lower management risks and makespan. Scheduling is processed with limited resources among sequential and parallel activities. The development of the scheduling is processed with efficient and effective scheduling approaches and technology. Solving the issues with an optimal solution consists with infinite number of possibilities. Typically, the process of jobs schedule task is complex. The issues of scheduling are solvable to find the impossible optimal solution without any enumerative algorithm. Computation times increases the problem size and to find exact solutions often using the dynamic programming algorithm. The difficulty is lead to find or develop the transmitting rules and heuristic methods, however; only special conditions apply to the function with the restricted points [1-12].

In the current competitive environment the efficient process of scheduling to meet the requirements with better performances is the main role in implementation. The issues of scheduling the jobs are executed according to the “priorities” and “capacity”. The effects of the scheduling characterize are used to specify the processing of jobs in the machines with the objectives and timing. The main focus on the environment is to optimize the system with same assumed sets which refer the scheduling flow [9]. The general flow-

ISSN: 2005-4246 IJUNESST

shop scheduling between the machines will not allow the changes of sequences in scheduling. In this flow shop, throughout the system the jobs sequences are maintained and constant time is set with limitation to process the available jobs at zero times. It consists with a makespan (Cmax) objective to optimize the job sequences in the machines with same set and represent the optimal solution to provide minimum makespan. It assumes the problems for m>2.

The scheduling of jobs is well-organized to operate the functions with enable the sequences requirement and availability to optimize. The job shop scheduling problem (JSP) is managing the issues of optimization hardly and manual constructing of schedules the optical issues are time consuming and complex task. It integrates the flow of the sequence process of job scheduling with rule based and enhanced the scheduling of previous process. In scheduling stage it provides the solution with a wide range of makespan.

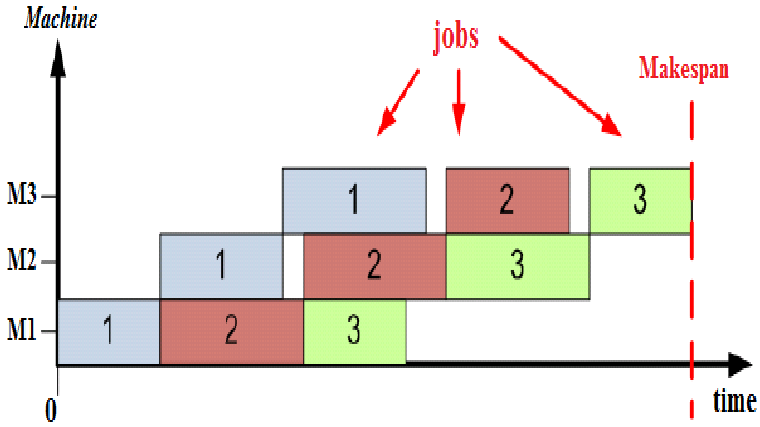

Figure 1. Scheduling Problem with 3 Jobs and 3 Machines

The genetic algorithm simulates the scheduling – optimization process in nature and it is a meta-heuristic method for saving and maintain the appropriate sequence and scheduling issues to evaluate the schedules [7]. The job sequences are same for all machines for the sequences, process to provide average completion time (*i.e.*, makespan) and it is non-deterministic polynomial-time (NP) hard problem. To overcome the problems various heuristic algorithms have been proposed, it builds possible solutions for solving it. Normally the solution qualities are not satisfied the requirements and process very fast. In the Nawaz-Enscore-Ham (NEH) heuristic states the sequences with minimum solution to optimize and try to improve the global or suboptimal optimum. Figure 1 shows the simple job scheduling problem with three jobs and three machines.

In this paper, the research work aims at designing the system which uses, an efficient, flexible optimization and scheduling with more possibilities solution for job scheduling in the cloud. It provides an optimal solution among the scheduling – optimization is processed in a simple manner. Through services the job sequence scheduling is performed with more possibilities and accessed in the cloud service. In this work for sequence process and also to reduce the time, makespan and increases the possibilities the meta-heuristic GA is implemented with the improved NEH heuristic job scheduling. It performed with efficient process and provides better performances in optimizing the optimal system.

In this paper, the strategies of scheduling in enterprise system are developed to optimize the time and makespan factors. The rest of this paper is structured as follows. The literature survey of scheduling – optimization problems and computation process were discussed in Section II. The problem statements models are presented in Section III.

The proposed work INSMG and the scheduling model algorithm are discussed in Section IV. The tentative and simulation results with the performance analysis were illuminating in Section V. Finally, in Section VI the research work is concluded its proposed work analysis and future work.

2. Related Works

In this section, a literature survey is done about the scheduling-optimization problem and related processes with solution in the cloud environment. Scheduling system is one of the optimal processes of the system in cloud computing, which is beneficial for all enterprises and industries, improving the quality of services in optimal scheduling with efficient process with minimum makespan. Novel heuristic scheduling algorithms are implemented to detect the diversity and better solution of candidates dynamically by low-level heuristic. To evaluate state-of-the-art scheduling algorithms HHSA significantly reduce the task scheduling makespan [3]. It is emerging the simulation-optimization techniques in the application of green internet computing to reduce the expected time required for completing the process with the assumption of processing time [16]. By meta-heuristic the consumption of energy and time are improved and efficient in environmental perspective with illustrating the potential savings [1] and [7-11].

The heuristic approach is constructive process of solution for the permutation flow-shop problem [11]. As per the approach the Johnson’s rule is applied to give initial solution. The MOD scheduled and evaluated with the NEH, CDS, Gupta’s and Palmer’s Slope Index algorithm. The datasets proposed by Taillard is consists of scheduling which have 120 benchmark problems. From the results the MOD process is solving the jobs and machine in a short interval time in flow-shop scheduling problems. The MOD algorithm approach shows the substantial enhancements when compare with the NEH heuristic. As a good results obtained from the realistic analysis of Taillard product of 120 benchmark problems. It provides an optimal solution in a reasonable time for the benchmark issues [6].

In the presented study, based on the rule the job sequence dependence is performed on the scheduling issues. The pair-splitting strategy is implemented to find the makespan minimal. Based on the involvement of the multi-jobs the strategy results are evaluated to provide better performance and have 6.88% averagely for all Taillard problems. In cellular scheduling problems is optimized to provide less makespan and setup the times for the sequence dependency of the cell line flow. Minimizing make span (Cmax) provides the result of the speed and rate of the output with improvement. The optimized problem is solved by using the local search technique, Genetic Algorithm (GA) and Ant Colony Optimization (ACO). This approach is validated by the results with the comparisons of a tailor-made heuristic algorithm [5-9].

In parallel machines, each stage of the scheduling process may be unrelated. As per the order it executes the operation and releases the specific dates, but it will not permit the pre-emption of the jobs. It determines the minimum tardy jobs and the makespan convex combination. Several dispatching rules and heuristic approaches are generated the job sequences with GA in dissimilar parallel machines and provide the performances with flexible access [7]. The approach is comparatively related with each other in a set and tested up to 20 stages with 50 jobs [12].

Flow shop Scheduling regulates the jobs sequence optimally in machines with the same set of orders. It constraint the jobs schedule one at a time and it will not make the machine idle when availability of jobs in processing. NEH (Nawaz, Enscore, Ham) Algorithm provides efficient processing with minimum makespan for Flow shop Scheduling Problems PFSP. It modified the NEH algorithm in order to improve the solution quality with its complexity as same as the real process of scheduling algorithm [4] and [15].

The job shop scheduling problem (JSP) is a multifaceted and time-consuming task of solving the issues. In scheduling the game designed with the players is not fully involved in the scheduling procedures. It provides the design of the game to enable the creation and solution of the players on scheduling issues [2]. It optimized by playing games via sense-making experiences. It provides the traditional board to play the game with graphic user interface by manipulating blocks and the players can appropriately process in a trial-and-error manner for scheduling. Moreover, it evaluated the scheduling process by generating GA and scheduling with a makespan. As a result, both are served to schedule the game playing simultaneously. If the problem relatively smaller in size then most of the procedure of scheduling not enough in experience. It includes all possibilities at the end and adopts the strategies for better search process.

Genetic algorithms are applied to optimization issues as a heuristic approach to provide satisfaction of optimization benchmark. According to the limitations on jobs route to characterize each job by the same technological route. It includes additional constraints processing at each stage like setup time, processing times and release times and genetic process is performed like criteria of termination, crossover and mutation. Also, it emphasizes the differences of the stages with their criteria problems [14]. It includes the package program which is developed for solving the scheduling issues with GA at the Otto-von-Guericke-University Magdeburg on "'LiSA - A Library of Scheduling Algorithms' developed".

In grid system, the capacity of solving problems at a same time with more nodes provides usefulness in the allocation of jobs for the resources with the facts. Roulette Wheel Selection Genetic Algorithm using Best Rank Power (PRRWSGA) scheduling will speed up the search process and convergence with less time using an MCT algorithm to achieve a high feasible scheduling solution quality. It is independent process and improves the search time than the standard GA and IRRWSGA [10].

Distributed Permutation Flow shop Scheduling Problem (DPFSP) is enhanced a new approach to deal with DPFSP through novel dispatching rule [8]. In new dispatching rule a group of jobs is inserted at a time to the factories instead one job. To validate it large instances are tested with intensive benchmark and perform in terms of trade-off between running time and quality of solution. In real world problem the PFSP generates the distribution randomly. Synthetic generation issues are alternate the performances rapidly with realistic characteristics. It provides less complex and fast process of stochastic algorithms for searching topology space.

An effective particle swarm optimization (PSO)-based memetic algorithm (MA) is minimizes the time completion in solving the optimization problem. It operates the local searching with abilities of exploration or exploration balance. It utilizes with the efficiency to perform manipulation by the rule based representation of random key for job permutations. It initializes the quality and diversity swarm generated by the optimization is performed with the probability and avoid the convergence of prematurity [13].

In this survey much of the discussions relate to work, which ensures the optimization services of scheduling in the cloud environment. The computation processes are ensuring dynamic services. The limitation with existing mechanism have performed with high makespan and limited solution for the issues. The proposed work overcomes such limitations with high performance, reduced makespan, time and provides more possibilities sequences as a solution for the scheduling problem in the cloud. It also ensures resilience against the limitation of scheduling.

3. Problem Statement

3.1. Problem Description

Assumed that the jobs and machines are optimized through the sequence of a given machine. Every job disintegrated into various levels of the machines and visits more than once. The duration of the processing time on the job is operated and performed with less makespan. The utility of the machine is implied by the time processing sequence. Each job has k ($k \geq 2$) of various operations in a chain with standard time processing t. In processing of jobs the machines will not idle unless there is no job to proceed and never break down the system in the scheduling period. The processes are immutable and operate in the unlimited space of jobs waiting in the system. In machines process the job scheduling is order is needed to be defined for the purposes of identifying the time differences in processing. The jobs are available at 0 times for processing in the machine.

3.2. System Model

The services on the cloud provides with the system representation of the environment as below.

- Meta - Heuristics: Enable efficient access to the same order of complexity and speed up the optimization process to define the efficient solution for the scheduling problem in the cloud.
- Cloud Server (CS): Cloud services provider (CSP) manages its access in the cloud environment.
- Improvement Meta-Heuristics: used improvement the performances by processing the system from the initial sequence to till the end of optimization improvement the function will continue the process.

3.3. Strategic Goals

It ensures the efficient of optimal system with certain phrases which are used to proceed with the content as follows.

- Flexibility: Enhance the improvement of the meta-heuristic approach of GA and scheduling process to lead the jobs on machines in an easy manner.
- Lightweight: Consume less time processing of system scheduling of jobs in in cloud.

3.4. Notation and Preliminaries

- J - Number of jobs {1, 2, …, n};
- M - Number of machines {1, 2, …, m};
- C_{max} - makespan;
- C_t - completion time;
- S_t - Starting time;
- O_{jm} - The operation of the jobs processing.
- m - Number of machines;
- n - Number of jobs processing at 0 time;
- l - Number of jobs layers;
- T_{lm} - Processing time of job layers;

- Z_{lm} - 1 precedes the operation else 0;
- T_{jm} - Processing time of n processing of jobs on m machine, n ∈ J and m ∈ M ;

The model of solving the issues is as follow:

Minimize C_{max}

Subject to $S_{lm} + T_{lm} \leq S_{l,m+1}$

$S_{lm}+ T_{lm} \leq S_{l+1,1}$

$M (1- Z_{lm}) + (S_{lm} - S_{ml}) \geq T_{lm}$

$M Z_{lm} + (S_{lm} - S_{ml}) \geq T_{lm}$

$S_{l,m} + T_{l,m} \leq C_{max}$

$C_{max} \leq 0, S_{ml} \geq 0; Z_{lm} = 0$ or 1;

The objective function to define the minimum makespan by the constraint sets as above is to satisfy the requirements of the heuristic scheduling process.

4. Improved NEH-Heuristic Job Scheduling Using Meta-Heuristic GA – INSMG

In cloud environment the optimal system is scheduling an optimal solution for improving the performances with less processing time. Relevant are engaged in computing the process of optimization of jobs to meet the requirements of the approach. It is easily reached via cloud around the world. The complex problem solutions are overcome in the proposed INSMG. It performed by generating the Meta heuristic NEH job scheduling based on the GA for efficient process in an easy manner and it is a parallel computing approach. It aims in providing the assurance of solving the problems and generate more possibility sequences with less provision time. By this method, a flexible access is provided in the cloud environment.

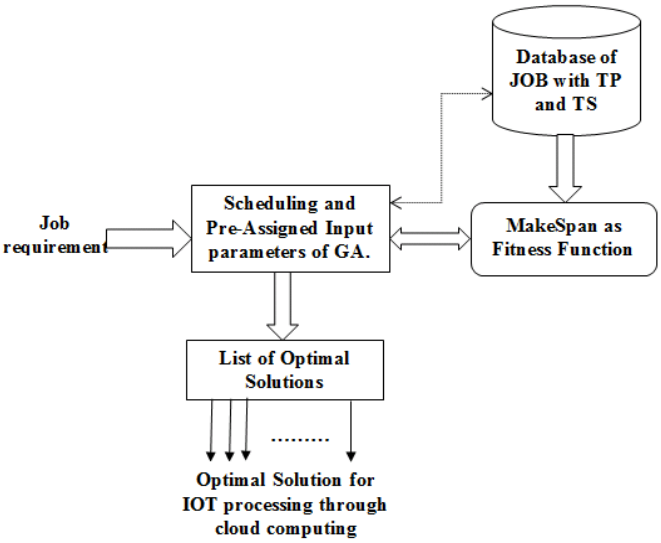
Database of JOB with TP and TS
Job requirement
Scheduling and Pre-Assigned Input parameters of GA.
MakeSpan as Fitness Function
List of Optimal Solutions
....
Optimal Solution for IOT processing through cloud computing

Figure 2. Block Diagram–Proposed Work (INSMG)

Figure 2 shows the block diagram of the proposed work for scheduling the jobs with best sequences and solution by finding the fitness function. Also, it is a process of content based service and diversity detecting. The proposed scheduling algorithm is performed to minimize the makespan process with better performance than the existing. The time complexity is reduced by the usage of GA but in proposed algorithm, it is used for providing more possibilities of sequences with less complexity process. It fetches all the job schedule sequences possibilities to choose the best solution for the issues in a best way.

In this section, Implementation and Analysis of the proposed method in cloud server is performed. For minimum makespan of the optimization sequence the fitness function is derived and performs with more possible objective function. For random representation of selection of solutions are beginning processes by GA. The genetic process of operating the scheduling sequences in order to improve the efficiency of solution quality and to reduce the makespan. It continues the process of operating the system until the satisfaction of the requirements and best selection of sequence. The procedure of the proposed algorithm of INSMG is performed as follow.

Step 1: Find the solution for the optimal issues by sequence process and setup the processing time using a heuristic approach from the sequence list (L).

While List-Sequence (L) is not empty do

- Choose a job from sequence list.

Step 2: The optimized problem is represented with the coding scheme for appropriate selection over the scheduling process based on the fitness value. Initialize the random process of optimization.

Step 3: Evaluate the fitness function.

Step 4: Sort list the time process of n jobs in non-increasing order.

Step 5: Assume the jobs f (f- number of jobs) from the list and form (f=5) 5! = 120 sequences with length 5 for each. The parameter p will select the sequences for processing from the more possible of the proposed algorithm.

Step 6: Set the list of jobs n = 10 on the list and insert the sequences at n positions. p (n − 1) - partial sequences of job, as (n × p) n-job sequences

Step 7: The best p is selected form the n × p sequences.

Step 8: Increment n by 1; if n > i, then that sequence is accepted as the best solution and update the solution. Then stop the process otherwise go to step 5.

Step 9: Remove the jobs after finding the solution from sequence list.

Step 10: End while and analysis the results.

The complexity of improved approach heuristic is enumerated the process with the (n (n+1) / 2)-1 and the total enumeration function. Hence, the proposed approach complexity is clearly stated as Θ (n^2).The total enumerations as f! + Σ, (n =5 to j) p * n; f! + p * Σ, (n=5 to j) n;

The objective of the minimize makespan of the sequence of n-job *i.e.* C_t (M_m, J_n). The computation of job process completion time is computed as:

$$C_t (M, J) = T_{jm} (M, J) \quad (1)$$

$$C_t (M_m, J) = C_t (M_m\text{-}1, J) + T_{jm}(M_m, J) \quad (2)$$

$$C_t (M, J_n) = C_t (M, J_n\text{-}1) + T_{jm} (M, J_n) \quad (3)$$

$$C_t (M_m, J_n) = \max \{C_t (M_m\text{-}1, J_n), C_t (M_m, J_n\text{-}1)\} + T_{jm} (M_m, J_n) \quad (4)$$

In this work, flexible and efficient access of optimization of job sequence provides more possibilities for reducing the selection and processing time to optimize the system. By this proposed system the makespan is reduced with better performance of scheduling. As per the equations the overall process of this research work takes place for efficient and effective solution process for the scheduling problems in a cloud environment.

5. Simulation Results and Performance Analysis

In cloud server the dynamic process of solving the issues of scheduling is performed by the proposed algorithm. In order to evaluate the performance of the INSMG (Efficiency, Makespan, Complexity and performance are included in it), to calculate this evaluate the system with various sequences and optimization solution for same set will be suitable for system analysis. The process and the machines are defined as Intel core I5, 4.00 GB of RAM and 500 GB of hard disk. The simulation results of job scheduling are obtained using MATLAB.

The analysis of the optimized solution of the proposed work is performed as per the sequences in simulating it. Before processing the jobs scheduling the inputs of the meta-heuristic GA is pre-assigned as follow:

Pre assigned GA inputs are:

Cross over Fraction =0.6;

Migration Fraction = 0.02;

Initial Population =20;

Maximum Generation =50;

Stall Time Limit =100;

The simulation results of the proposed scheduling – optimization algorithm is performed, the more possible sequence solution for the issues as per the job requirement and the selection of the sequence is based on the situation of the system.

Optimal solution for 3 jobs:

Total Number of Jobs =5

Number JOBs need to be processed =3

Optimal Solutions are:

Optimized Job Sequence=-J1 -J4 -J3

Optimized Job Sequence=-J4 -J1 -J3

Optimal solution for 4 jobs:

Total Number of Jobs =5

Number JOBs need to be processed =4

Optimal Solutions are:

Optimized Job Sequence=-J1 -J2 -J4 -J3

Optimized Job Sequence=-J2 -J1 -J4 -J3

Optimized Job Sequence=-J2 -J4 -J1 -J3

Optimized Job Sequence=-J4 -J2 -J1 -J3

Optimal solution for 5 jobs:

Total Number of Jobs =5

Number JOBs need to be processed = 5

Optimal Solutions are:

Optimized Job Sequence=-J1 -J2 -J4 -J5 -J3

Optimized Job Sequence=-J1 -J2 -J5 -J4 -J3

Optimized Job Sequence=-J1 -J4 -J2 -J5 -J3

Optimized Job Sequence=-J1 -J4 -J5 -J2 -J3

Optimized Job Sequence=-J1 -J5 -J2 -J4 -J3

Optimized Job Sequence=-J1 -J5 -J4 -J2 -J3

Optimized Job Sequence=-J2 -J1 -J4 -J5 -J3

Optimized Job Sequence=-J2 -J1 -J5 -J4 -J3

Optimized Job Sequence=-J2 -J4 -J1 -J5 -J3

Optimized Job Sequence=-J2 -J4 -J5 -J1 -J3

Optimized Job Sequence=-J2 -J5 -J1 -J4 -J3

Optimized Job Sequence=-J2 -J5 -J4 -J1 -J3

Optimized Job Sequence=-J4 -J1 -J2 -J5 -J3

Optimized Job Sequence=-J4 -J1 -J5 -J2 -J3

Optimized Job Sequence=-J4 -J2 -J1 -J5 -J3

Optimized Job Sequence=-J4 -J2 -J5 -J1 -J3

Optimized Job Sequence=-J4 -J5 -J1 -J2 -J3

Optimized Job Sequence=-J4 -J5 -J2 -J1 -J3

Optimized Job Sequence=-J5 -J1 -J2 -J4 -J3

Optimized Job Sequence=-J5 -J1 -J4 -J2 -J3

Optimized Job Sequence=-J5 -J2 -J1 -J4 -J3

Optimized Job Sequence=-J5 -J2 -J4 -J1 -J3

Optimized Job Sequence=-J5 -J4 -J1 -J2 -J3

Optimized Job Sequence=-J5 -J4 -J2 -J1 -J3

In this research work the complexity of the system is calculated to prove that the performance of this work provides an effective and efficient process in cloud server. The result shows, this research work performance time and minimization of makespan has been improved with optimal process. From the result of the INSMG experimental study, we have concluded this research work, as shown in performance analysis, also compared with the existing work. Figure 3 shows the analysis of processing time of the proposed system (makespan), which shows the performance improvement than the existing system.

Figure 4 shows the fitness values for scheduling the jobs and evaluated as per the factors of fitness like scores (mean, best and worst), selection function, individual best function and the best and mean fitness. Table I shows the time requirement for the job scheduling and the time differences depend on the nature of jobs. As mentioned in the block diagram the database specify with the job nature related details, which deals with the required time for the job selection time (TS) and processing time (TP). According to the function the fitness function is evaluated.

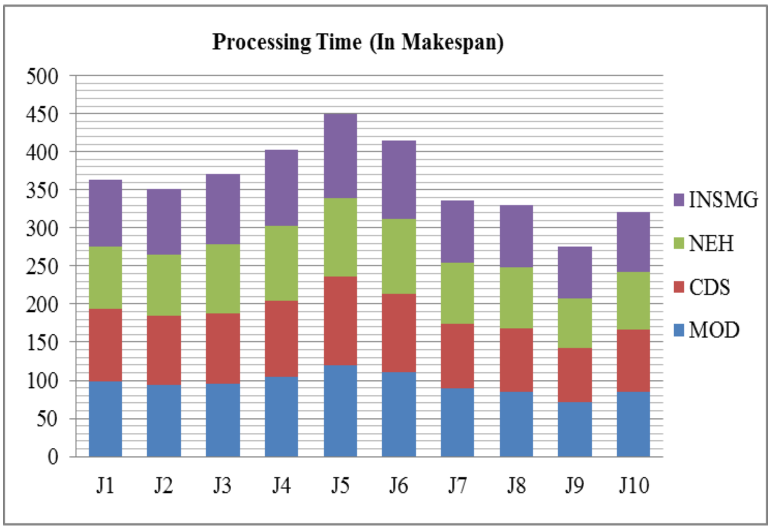

Figure 3. Analysis of Processing Time (In Makespan)

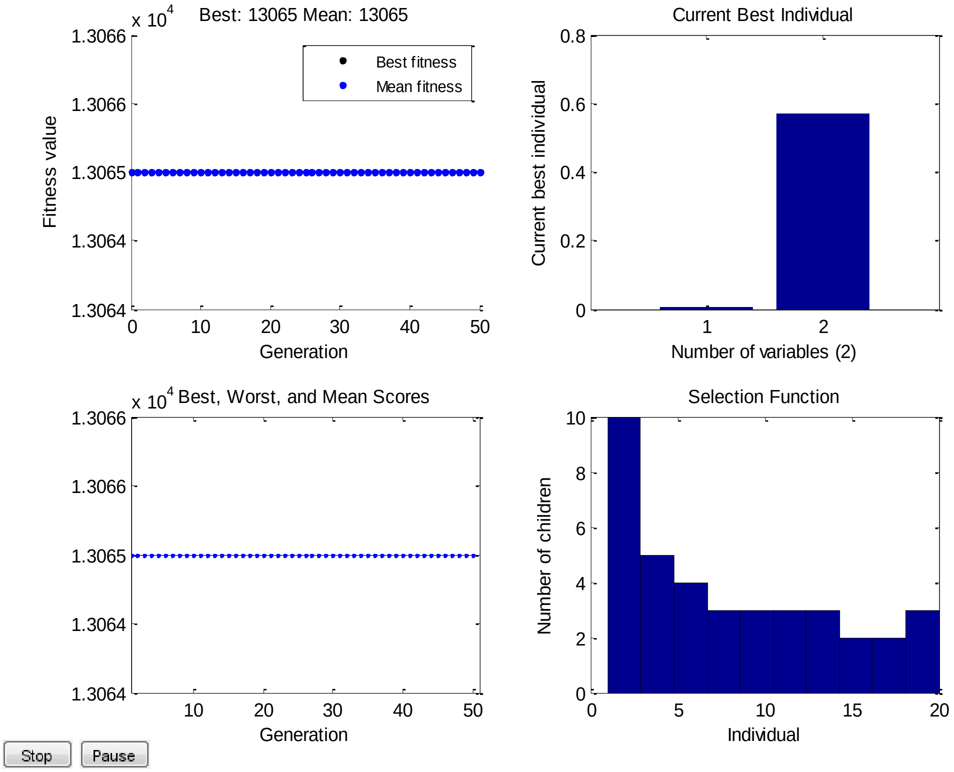

Figure 4. Fitness Function for the Proposed Schedule-Optimization System-INSMG

Table 1. Processing and Selection Time for Scheduling Jobs

Jobs	TS/TP	M1	M2	M3	M4	M5	M6	M7
J1	TS	120	260	261	30	180	0	27
	TP	340	140	143	57	64	3600	59
J2	TS	135	264	261	30	175	0	26
	TP	328	142	147	56	60	3600	59
J3	TS	120	260	259	33	175	0	26
	TP	335	147	147	56	64	3600	50
J4	TS	130	264	254	34	179	0	28
	TP	331	147	153	55	59	3600	61
J5	TS	132	259	267	32	180	0	29
	TP	337	142	159	51	62	3600	58

Hence this work aims at ensuring more possibilities of sequences of solution for scheduling problems. It also ensures the performance improvement with minimum scheduling-optimization time and makespan in an efficient manner. So the overall performance of the INSMG work provides efficiency, less time access, Complexity, Efficient process and Flexible access.

6. Conclusion and Future Work

In this work, proposed a service with an efficient scheduling of jobs in cloud service. The proposed optimal system will be reliable, easy to use and provides more sequence process for flexible and fast scheduling. This system is designed to provide less time or reducing the makespan with the optimized efficient process. The improved heuristic (INSMG) proposed to provide a better result than existing while considers the same complexity of algorithms. The proposed system generates more possible job sequences on machines with minimum makespan for heuristic processes of the optimal system to optimize the problem solution in an easy manner and fast process.

A streamlined, intuitive interface with optimization features of scheduling and sequences methodologies of the scheduling algorithm is utilized for better recitals. This research work provides a system in an easy manner improvised on time saving, flexible access and performance complexity.

Future work is planned to provide this approach with more possibilities of solution for the scheduling problem, which involving time setup based on the sequence-dependent. Also the approaches attempted the larger problems and try to find better strategies of pairs to optimize the system. The positive investigation results lead to a proposal of the research in the future. Still new ideas come forth as the field continues to progress further for scheduling the optimal system in a cloud environment.

References

[1] C. W. Tsai and J. Rodrigues, “Metaheuristic scheduling for cloud: A survey, “IEEE Systems Journal, vol. 8, no. 1, **(2014)**, pp. 279–297.

[2] K. J. Woo, “A Job Shop Scheduling Game with GA-based Evaluation”, An International Journal Applied Mathematics & Information Sciences, Appl. Math. Inf. Sci., vol. 8, no. 5, **(2014)**, pp. 2627-2634.

[3] C. W. Tsai, W. C. Huang, M. H. Chiang, M. C. Chiang and C. S. Yang, “A Hyper-Heuristic Scheduling Algorithm for Cloud”, IEEE Transactions on Cloud Computing, 2013, DOI 10.1109/TCC, 2315797, **(2014)**,

[4] E. Singhal, S. Singh and A. Dayma, “An Improved Heuristic for Permutation Flow Shop Scheduling (NEH ALGORITHM)”, International Journal Of Computational Engineering Research (ijceronline.com) vol. 2, no. 6, **(2012)** October.

[5] M. T. Taghavifard,” Scheduling Cellular Manufacturing Systems Using ACO and GA”, 48 International Journal of Applied Metaheuristic Computing, vol. 3, no. 1, **(2012)** January-March, pp. 48-64.

[6] P. Semančo and V. Modrák, “A Comparison of Constructive Heuristics with the Objective of Minimizing Makespan in the Flow-Shop Scheduling Problem”, Acta Polytechnica Hungarica, vol. 9, no. 5, **(2012)**.

[7] S. Kaur and A. Verma, “An Efficient Approach to Genetic Algorithm for Task Scheduling in Cloud Computing Environment”, International Journal of Information Technology and Computer Science (IJITCS), vol. 4, no. 10, September **(2012)**.

[8] J. Gao and R. Chen, “An NEH-based heuristic algorithm for distributed permutation flowshop scheduling problems”, Scientific Research and Essays, DOI: 10.5897/SRE10.1014, vol. 6, no. 14, **(2011)** July 18, pp. 3094-3100.

[9] P. D. Kaur and I. Chana, “Unfolding the distributed computing paradigm”, In: International Conference on Advances in Computer Engineering, **(2010)**, pp. 339-342.

[10] W. Abdulal, O. A. Jadaan, A. Jabas and S. Ramachandram, "Genetic algorithm for grid scheduling using best rank power", Nature & Biologically Inspired Computing, doi: 10.1109/NABIC.2009. 5393679, **(2009)** December 9-11, pp. 181-186.

[11] D. Laha and U. Chakraborty, “An efficient hybrid heuristic for makespan minimization in permutation flow shop scheduling,” The International Journal of Advanced Manufacturing Technology, vol. 44, no. 5-6, **(2009)**, pp. 559–569.

[12] J. N. Silva, L. Veiga and P. Ferreira, “Heuristics for Resource Allocation on Utility Computating Infrastructures”, In: 6th International Workshop on Middleware for Grid Computing, New York, **(2008)**

[13] B. Liu, L. Wang and Y. H. Jin, “An Effective PSO-Based Memetic Algorithm for Flow Shop Scheduling”, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B: CYBERNETICS”, DOI: 10.1109/TSMCB.2006.883272, vol. 37, no. 1, **(2007)** February.

[14] J. S. Chen, J. C. Hsien Pan and C. M. Lin, “A hybrid genetic algorithm for the re-entrant flow-shop scheduling problem” Multiprocessor Scheduling: Theory and Applications, Itech Education and Publishing, Vienna, Austria. ISBN 978-3-902613-02-8, pp. 436, **(2007)** December.

[15] J. P. Watson, L. Barbulescu, A. E. Howe and L. D. Whitley, “Algorithm Performance and Problem Structure for Flow-shop Scheduling”, American Association for Artificial Intelligence.

[16] G. Cabrera, H. Pérez-Roses, A. A. Juan and J. M. Marquès, “PROMOTING GREEN INTERNET COMPUTING THROUGHOUT SIMULATION OPTIMIZATION SCHEDULING ALGORITHMS”, Winter Simulation Conference, Proceedings of the, **(2013)**.

Authors

Jeeva Rathanam G, completed Diploma in Computer Science Engineering from Murugesan Institute of Technology, Salem, in 2007. She received the BE degree in Computer Science Engineering from Sengunthar College of Engineering For Women, Tiruchengode, in 2011 and currently pursuing M.S. (by Research) from Anna University, Chennai, India. Area of interest includes Big Data Analysis, Distributed and Parallel Computing, Cloud computing.

Rajaram A, received the BE degree in electronics and communication Engineering from the Govt.,College of Technology, Coimbatore, Anna University, Chennai, India, in 2006, the ME degree in electronics and communication engineering (Applied Electronics) from the Govt., college of Technology, Anna University, Chennai, India, in 2008 and he received the Ph.D. degree in electronics and communication engineering from the Anna University of Technology, Coimbatore, India in March 2011. He is currently working as an Associate Professor, ECE Department in SRS College of Engineering and Technology, Salem, India. His research interests include mobile adhoc networks, wireless communication networks (WiFi, WiMaxHighSlot GSM), novel VLSI NOC Design approaches to address issues such as low-power, cross-talk, hardware acceleration, Design issues includes OFDM MIMO and noise Suppression in MAI Systems, ASIC design, Control systems, Fuzzy logic and Networks, AI, Sensor Networks, Cloud Computing, Big data Analysis.