A Review of No Wait Flow Shop Scheduling using GA Algorithm

Umesh Gupta*, Dr. Ashwani Kumar Dhingra**, Dr. Rahul Lodha***
*Research Scholar, Mewar University Gangrar Chittorgarh (India)
** Department of Mechanical Engineering, UIET, MD University, Rohtak,(India)
***Associate Professor, Mewar University, Chittorgarh (India)

Abstract- In this paper, we intend to work on no-wait flow shop scheduling problem with due date constraints. During this process, waiting time isn't allowed between sequential operations of jobs. When, the jobs completion have time limit; due time constraints are treated as hard constraints. The performance criterion considered here is make-span. The problem is strongly NP-hard. Based on various decision variables, distinct mathematical models are developed e.g. two constraint-programming models, three quadratic mixed integer programming model and a mixed integer-programming model. In addition, a unique modelling method is established for the problem. The new modelling method investigates the main features of the problem resulting in elimination of many illogical answers from given set of possibilities. The resulting propositions along with this modelling method are assimilated in an exact algorithm to obtain optimal results. In order to investigate performance of the newly developed mathematical model and compare it with the exact algorithm, many test cases are solved and the results are testified.Computational result is superior than our mathematical model.

I. INTRODUCTION

Flow look Programming has important applications in many industries. No-wait flow look programming could be a special approach with applications in concrete ware manufacturing, mechanical trade, chemical process and pharmaceutical process [1-4]. During a no-wait flow look, if we consider n jobs are running, then there are m operations performed with a predetermined interval. Simultaneously, a no-wait flow look programing downside need to follow certain constraints. The process of every job should be continuous. Which Means once the engagement starts on the first machine, it must be processed through all machines without any interruption and none pre-emption. Every machine can handle only one job at-a-time. Every job passes through each machine only once with zero release time. It means that all jobs are processed at time 0. The final processing time already includes the set-up time of every machine. In no-wait flow shop scheduling problem total flow time is considered as the most important performance measure as it can lead to rapid turn-around of jobs, stable utilization of resource, and minimization of in process inventory[1,5].

The main objective of this paper is to find a perfect job sequence that can minimize the total flow time. Though few exceptions have been observed, in most of the cases flow shop problems are proved to be NP-hard [6], For solving them heuristic procedures are the best. Since decades, heuristics procedures are mainly developed for solving the no-wait flow shop-scheduling problem. Rajendran and Chaudhuri [7] proposed a simple construction heuristic with two priority rules for total flow time objective while Fink and Voβ [8] used several simple construction heuristics such as cheapest insertion (Chins), nearest neighbour (NN), and pilot method (Pilot). To delineate the objective of the problem, we have consider a flow shop with two jobs and three machines: Where m1, m2 and m3 are three machines for sequencing the two jobs when no wait is acceptable. Let δ be the minimum difference between the start times of the two jobs. The value of δ be same for two consecutive jobs regardless of their positions in the sequence, as no wait is allowed for the jobs. Similarly, δ value of any two successive jobs can be obtained. Heuristic researches the principal relations for no-wait flow shop scheduling. Aldowaisan and Allahverdi [10, 11] in his study presents heuristic for no-wait flow shop-scheduling problem with m-machine constraints and setup times. Whereas Allahverdi and Aldowaisan [12, 13] study no-wait flow shop scheduling with different setup, three machines and sequence dependent additive changeover times. The Nawaz-Enscore-Ham (NEH) [14] anticipated the classic flow shop problem, which is considered the best constructive heuristic for makespan criterion [15]. Bertolissi, [16] in his research presented a heuristic that includes job-pair-comparison algorithm and job-insertion algorithm, which minimizes the total flow times. In addition, Gao [17] proposed a heuristic, SDH, for the no-wait flow shop scheduling with total flow time criterion. As evident from the literature study, in solving the no-wait flow shop-scheduling problem with total flow time criterion, the SDH and the Bertolissi heuristics proved to be better than NEH heuristic. Moving ahead, Laha and Sarin [18] proposed a modification of the method of raminan and Leisten [19] that can minimize total flow time in permutation flow shop. Ruiz and Allahverdi [20] in their research analysis presented several heuristics and local search methods based on genetic algorithm and iterated greedy procedures. Allahverdi and Aldowaisan [21], Framinan et al [22] and Li [23] have studied composite heuristic minimizing total flow time in flow shop and permutation flow shop scheduling problem. Amongst all the literature studied, we found that, in the composite heuristics, the iteration composite heuristic by Li [23] depicts the highest proficiency as it improves the solutions of the one-pass method by an iteration method.

II. LITERATURE REVIEW

Xiaojuan ZHAO, Jun TANG[1]:- M-machine no-hold up stream shop (NWFSS) issue is a NP-finish issue, and a novel way to deal with take care of this compose issue is exhibited in this paper. A nonstop stream shop issue with aggregate stream time as standard is considered. This paper expands the fake safe framework (AIS) approach by proposing another philosophy named as Psycho-Clonal calculation. Proposed calculation appreciates the kinds of AIS and Maslow's need chain of importance hypothesis to advance a Meta heuristic. Numerical reproduction with little and vast number of employments concerning mistake rate is accounted for. The outcomes got are contrasted and the other existing methodologies. Numerical reproduction has uncovered that outcomes got utilizing proposed calculation have significant improvement over others[24].GAO Kaizhou, PAN Quanke[2]:- This paper proposes a successful technique for the no-hold up stream shop planning improvement dependent on concordance seek (HS). Stream shop planning issue (FSSP) is an average NP-hard combinational streamlining. The motivation behind this paper is the aggregate stream time basis. Right off the bat, the HS-based streamlining component and system is displayed. Also, the aggregate stream time is computed by a novel technique. Thirdly, a biggest request esteem govern is utilized to change congruity in amicability memory from genuine vectors to work arrangement with the goal that the concordance look can be connected for FSSP. Ad libbing standard of new concordance is elaborated and high viable calculation parameters are set for advancement question. Finally, recreations and correlations show the proficiency, viability and vigor of agreement scan calculation for no-hold up FSSP[25]. Sagar U. Sapkal, Dipak Laha[3] as the no-wait flow shop scheduling problems are NP-hard, only heuristic procedures are the most suitable solutions for them, especially for big problems. In order to minimize total flow time criterion in no-wait flow shop scheduling problems, we propose a constructive heuristic. The proposed heuristic assumes that the priority of a job in the order calculated as the summation of its processing times on the bottleneck machine(s) in order to select the initial sequence of the jobs. Whereas the final sequence depends on the principle of job insertion for minimizing the total flow time. The results obtained from data analysis shows that the proposed heuristic significantly beats the existing heuristics, without affecting its computational CPU time [26]. Xiaoping Li, Qian Wang[4] introduced associate degree objective increment methodology for no-wait flow retailers with make-span decrease, and calculated make-span of a replacement schedule directly from its parent and figured out whether the new schedule is healthier than its parent or not. Specific make-span increments square measure adapted two basic operations in most heuristics, one analyzed insertion and second, pair-wise exchange. More commonly a composite heuristic supported make-span increment is planned. As it can be observed from the results, the proposal outperforms many effective algorithms like GR, SA2, and dominion. Additionally it takes negligible central processor time among the four algorithms compared, which clearly tells that the proposal is intriguing for large-scale no-wait flow retailers in sensible producing systems [27]. YOU-GEN LIU, XIA ZHU[5] observes that multi-objective flow-shop booking issues are increased considerably both in common as well as scholarly fields. In this paper, A half and half multi-objective hereditary calculation is proposed in this paper, which comprehend multi-objective no-hold up flow shop planning issues with a two prong approach, the make span and the aggregate stream time minimization. The projected result analysis utilizes standard of non-ruled arrangement along with elitist arrangements methodology. To make it effective, two successful multi-target detection systems on target growth are presented at every step in the document. Test results prove that the proposal beats the other three heuristics both on adequacy and on productivity [28] Weishi Shao, Dechang Pi[6] proposes a hybrid iterated ravenous (HIG) method for calculation to understand the assumed no-hold up stream shop booking issue (DNWFSP) through the makespan measure. The HIG mainly includes four segments, i.e. primer stage, development and pulverization, nearby pursuit, acknowledgment basis. In the initial stage, an adjusted NEH (Nawaz-Enscore-Ham) is suggested to create a favorable initial arrangement. In the region pursue stage, four adjacent techniques dependent on issue properties (i.e. embed move inside processing plant, embed move between industrial facilities, swap move between manufacturing plants) are used to promote observing capacity. The viability of the induction stage and adjacent detection strategy is calculated by numerical examination, and the correlations with the as of late distributed iterated covetous calculations show the high capability and pursuing capacity of the proposed HIG for measuring the DNWFSP [29]

III. NO-WAIT FLOW SHOP

A no-wait flow shop is a constrained flow shop scheduling problem in which each of the n jobs $\{J1…Jn\}$is processed sequentially on m machines without delay between adjacent machines, i.e. the start of a job must be delayed on the first machine when necessary so that the job need not wait for processing on the subsequent machines. Let $O_{i,j}$ be the operation of job j (j = 1,2,.. n) processed on machine I (i =

1,2,, m), ti1j be the processing time of 0°, , S1j and C1i be the starting time and finish time of Oi,j. Because a schedule z is a permutation of the n jobs, the permuted sequence can be denoted as in which is the i th (i=l,..,n) job in z To illustrate the following properties conveniently, two dummy jobs are introduced to denote the start and the end of z. The processing times of the two dummy jobs are zero and they are explicitly denoted as Os in z . The start dummy job 0 is regarded as J,,, and the end one as J[n+11 in z Therefore, The makespan of z is equal to Cm [n] which can be computed specifically, as follows. Figure 1 depicts two adjacent jobs Ji and J1 in a schedule, in which finish distance D1j denotes the distance between the finish times of the two jobs, i.e. the distance between the finish times of the adjacent operations on machine Mm. Assume J, starts to process on the first machine just as Ji finishes on the last machine (denoted as Jj in figure 1). Shift Jj1 left until there exists at least one machine M, (1 < s < m) on which the idle time between the two adjacent operations is zero, i.e. the starting time of 0s equals to the finish time of 0SX. This left shifting length is denoted as L1i.

IV. ALGORITHMS

A. Basic OF GA (genetic calculation): In a hereditary calculation, a populace of hopeful arrangements (called people, animals, or phenotypes) to an improvement issue is advanced toward better arrangements. Every competitor arrangement has an arrangement of properties (its chromosomes or genotype) which can be changed and modified; customarily, arrangements are spoken to in twofold as series of 1s, however different encodings are likewise conceivable. The development typically begins from a populace of arbitrarily produced people, and is an iterative procedure, with the populace in every cycle called an age. In every age, the wellness of each person in the populace is assessed; the wellness is generally the estimation of the target work in the enhancement issue being tackled. The more fit people are stochastically chosen from the present populace, and every individual's genome is changed (recombined and potentially haphazardly transformed) to shape another age. The new age of applicant arrangements is then utilized in the following emphasis of the calculation. Usually, the calculation ends when either a most extreme number of ages has been delivered, or a tasteful wellness level has been gone after the populace.

An average hereditary calculation requires:

1. a hereditary portrayal of the arrangement space.
2. a wellness capacity to assess the arrangement space.

A standard portrayal of every applicant arrangement is as a variety of bits. Varieties of different kinds and structures can be utilized in basically a similar way. The principle property that makes these hereditary portrayals helpful is that their parts are effectively adjusted because of their settled size, which encourages straightforward hybrid activities. Variable length portrayals may likewise be utilized, yet hybrid execution is more intricate for this situation. Tree-like portrayals are investigated in hereditary programming and diagram shape portrayals are investigated in transformative programming; a blend of both straight chromosomes and trees is investigated in quality articulation programming. Once the genetic representation and the fitness function are defined, a GA proceeds to initialize a population of solutions and then to improve it through repetitive application of the mutation, crossover, inversion and selection operators.

Applications of evolutionary computation to machine learning are referred to as genetic-based machine learning (GBML). Evolutionary computation (EC) techniques belong to the class of optimization tools, inspired by biological processes. The main idea of EC lies in the iterative modification of the population of individuals (candidate solutions of the problem – chromosomes) with selection and recombination procedures. Rule-based genetic algorithms (GA) are successfully applied to the solution of machine learning problems due to natural scalability, parallelization, noise resilience, flexibility of objective function, universality of computational scheme and possibility of using heuristics for data representations [Kovacs, 2010]. On the other hand, rule-based forecasting is able to take into account several time series at once and consider existing causal relationships in complex economic processes, which are significantly affected by various factors.

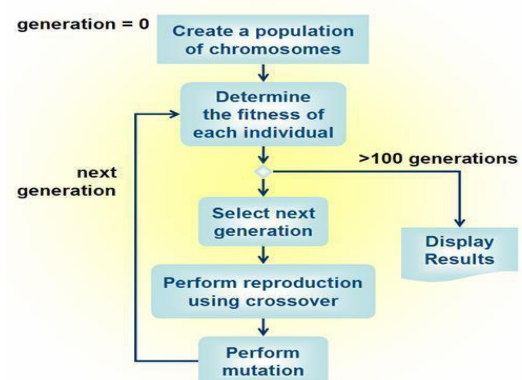

Fig.1: Genetic algorithm flow diagram.

Genetic algorithms (Gas) based on the principles of natural evolution. Due to its ease of applicability, numerous applications of genetic algorithms are found in the area of business, scenic, engineering, and forecasting problems. Now we will mention some basics of genetic algorithms.

B. Fuzzy Logic:- Fuzzy logic is a form of many-valued logic in which the truth values of variables may be any real number between 0 and 1. It is employed to handle the concept of partial truth, where the truth value may range between completely true and completely false. By contrast, in Boolean logic, the truth values of variables may only be the integer values 0 or 1. Furthermore, when linguistic variables are used, these degrees may be managed by specific (membership) functions.

The term fuzzy logic was introduced with the 1965 proposal of fuzzy set theory by Lotfi Zadeh. Fuzzy logic had however been studied since the 1920s, as infinite-valued logic—notably by Łukasiewicz and Tarski. Fuzzy logic has been applied to many fields, from control theory to artificial intelligence. Classical logic only permits conclusions which are either true or false. However, there are also propositions with variable answers, such as one might find when asking a group of people to identify a color. In such instances, the truth appears as the result of reasoning from inexact or partial knowledge in which the sampled answers are mapped on a spectrum. Humans and animals often operate using fuzzy evaluations in many everyday situations. In the case where someone is tossing an object into a container from a distance, the person does not compute exact values for the object weight, density, distance, direction, container height and width, and air resistance to determine the force and angle to toss the object. Instead the person instinctively applies quick "fuzzy" estimates, based upon previous experience, to determine what output values of force, direction and vertical angle to use to make the toss. Both degrees of truth and probabilities range between 0 and 1 and hence may seem similar at first, but fuzzy logic uses degrees of Fuzzy all input values into fuzzy membership functions. Truth as a mathematical model of vagueness, while probability is a mathematical model of ignorance.

V. CONCLUSSION

Generally, the essence of an algorithm to resolve practical problems lines in two aspects. One is how to effectively search the solution of object function. Another is how to effectively evaluate solution. These two aspects directly affect the effectively and efficiency of an algorithm. This paper proposed a heuristic for solving no-wait flow shop scheduling problem with the total flow time criterion. result experiments and comparisons demonstrated the efficiency of the Heuristic. In the future research, hybrid heuristic will be developed for no-wait flow shop scheduling problem and multi-object complex scheduling problems.

VI. REFERENCES

[1]. Rajendran C, A no-wait flowshop scheduling heuristic to minimize makespan. J Oper Res Soc, vol. 45, pp. 472–478,1994.

[2]. Hall NG, Sriskandarayah C, A survey of machine scheduling problems with blocking and no-wait in process. Oper Res, vol. 44, pp. 510–525, 1996.

[3]. Grabowski J, Pempera J, Sequencing of jobs in some production system. Eur J Oper Res, vol. 125, pp. 535–550, 2000.

[4]. Raaymakers W, Hoogeveen J, Scheduling multipurpose batch process industries with no-wait restrictions by simulated annealing. Eur J Oper Res, vol. 126, pp. 131–151, 2000.

[5]. French S. Sequenceing and scheduling: an instroduction to the mathmatics of the job-shop. Chinchester: Ellis Horwood; 1982.

[6]. Gonzalez T, Sahni S. Flow shop and job shop scheduling: complexity and approximation. Operations Research, vol. 26, pp. 36-52, 1978.

[7]. Rajendran C, Chaudhuri D, Heuristic algorithm for continuous flow shop problem. Nav Res Logist, vol. 37, pp. 695-705, 1990.

[8]. Andreas Fink, Stefan Voβ, Solving the continuous flow shop scheduling problem by metaheuristics. Eur J Oper Res, vol. 151, pp. 400-414, 2003.

[9]. Aldowaisan T, A new heuristic and dominance relations for no-wait flow shops with setups. Comput Oper Res, vol. 28, pp.563-584, 2001.

[10]. Aldowaisan T, Allahverdi A, Total flow time in no-wait flow shops with setup times. Comput Oper Res, vol. 25, pp.757-765, 1998.

[11]. Aldowaisan T, Allahverdi A, New heuristic for m-machine no-wait flow shop to minimize total completion time. Omega, vol. 32, pp. 345-352, 2004.

[12]. Allahverdi A, Aldowaisan T, No-wait and separete setup three-machine flow shop with total completion time criterion. Int Trans Oper Res vol. 7, pp. 245-264, 2000.

[13]. Allahverdi A, Aldowaisan T, Minimizing total completion time in nowait flow shop with sequence dependent additive changeover times. J Oper Res Soc, vol. 52, pp. 449-462, 2001.

[14]. Nawaz M., Enscore EEJ, Ham I. A Heuristic algorithm for the mmachine, n-job flow shop sequencing problem. Omega , vol. 11, pp.91- 95, 1983.

[15]. Kalczynski PJ, Kamburowski J, On the NEH heuristic for minimizing the makespan in permutation flow shop. Omega, vol. 35, pp.53-60, 2007.

[16]. Edy Bertolissi, Heuristic algorithm fot scheduling in the no-wait flow shop. J Mater Process Technol, vol. 107, pp. 459-465, 2000.

[17]. Gao Kai-zhou, Pan Quan-ke, Li Jun-qing, Discrete harmony search algorithm for the no-wait flow shop scheduling problem with total flow time criterion. Int J Adv Manf Technol, vol. 56, pp. 683-692, 2011.

[18]. Pan Quan-Ke, Tasgetiren MF, Liang YC. A discrete particle swarm optimization algorithm for the no-wait flowshop scheduling problem. Computers & Operations Research, vol. 35, pp.2807-2839, 2008.

[19]. Dipak Laha, Subhash C. Sarin, A heuristic to minimize total flow time in permuation flow shop. Omega, vol. 37, pp. 734-739, 2009.

[20]. Framinan JM, Leisten R., An efficient constructive heuristic for flowtime minimization in permutation flow shop. Omega, vol. 31, pp.311-317, 2003.

[21]. Rubén Ruiz, Ali Allahverdi, New heuristics for no-wait flow shops with a liner combination of makespan and maximum lateness. Int J Produc Res, vol. 47, pp.5717-5738, 2009.

[22]. Allahverdi A, Aldowaisan T, New heuristics to minimize total flow completion time in m-machine flow shop. Int J Produc Econ, vol. 7, pp. 71-83, 2002.

[23]. Framinan J, Leisten M, Ruiz-Usano R, Comparison of heuristics for flow time minimization in permutation flow shop. Compu Oper Res, vol. 32, pp. 1237-1254, 2005.
[24]. Xiaojuan ZHAO, Jun TANG, "No-wait Flow Shop Scheduling Based on Artificial Immune System" 2010 IEEE
[25]. GAO Kaizhou, PAN Quanke, ZUO Fengchao, DUAN Junhua," A Harmony search algorithm for the no-wait flow shop optimization scheduling" July 29-31, 2010, Beijing, China
[26]. Sagar U. Sapkal, Dipak Laha," An improved scheduling heuristic algorithm for no-wait flow shops on total flow time criterion" 2011 IEEE
[27]. Xiaoping Li', Qian Wang', Cheng Wu2," An Efficient Method for No-Wait Flow Shop Scheduling to Minimize Makespan", International Conference on Computer Supported Cooperative Work in Design, 2006 IEEE.
[28]. you-gen liu, xia zhu, xiao-ping li," a new hybrid genetic algorithm for the bi-criteria no-wait flowshop scheduling problem with makespan and total flow time minimization", Seventh International Conference on Machine Learning and Cybernetics,Kunming, July 2008
[29]. Weishi Shao, Dechang Pi, Zhongshi Shao," A hybrid iterated greedy algorithm for the distributed no-wait flow shop scheduling problem", 2017 IEEE