International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 21 (2017) pp. 10530-10540

Performance Enhancement in Job Shop Scheduling with the Aid of Hybrid Social Spider Optimization and Gray Wolf Optimization

K. B. Gavali[1,*], Anand Bewoor[2], Debabrata Barik[3] and Prabagaran S[4]

[1]Research Scholar, Department of Mechanical Engineering, Karpagam Academy of Higher Education Karpagam University, Coimbatore-641021, Tamil Nadu, India

[2] Professor Department of Mechanical Engineering, Cummins College of Engineering for Women, Savitribai Phule Pune University, Pune-411052, Maharashtra, India.

[3,4] Professor, Department of Mechanical Engineering, Karpagam Academy of Higher Education Karpagam University, Coimbatore-641021, Tamil Nadu, India.

**Corresponding Author*

[1]Orcid Id: 0000-0002-4085-9624

Abstract

This paper incorporates the fusion-updating process in an optimization technique by solving NP hard problems. Here, the process of scheduling jobs and machines in a sequential order is made to attain in a minimized makespan time. The proposed methodology manipulates couple of conventional optimization technique Social Spider Optimization (SSO) and Grey Wolf Optimization (GWO) by fusing method. This fusion takes part in updating process in the optimization. Solving job shop scheduling, problem manually takes enormous time to reveal settling result; instead, solving with met heuristics technique will certainly conserve the time consumption in manipulating. The proposed hybrid (SSO-GWO) reveals superior result (minimized makespan time) compare to SSO, GWO and Particle Swarm Optimization (PSO) techniques. Even for manipulating with huge problem size in job shop scheduling the proposed technique would achieve the average of 99.90% closer to benchmark minimized makespan time.

Keyword: Job Shop scheduling, makespan time, Social Spider Optimization (SSO), Grey Wolf Optimization (GWO) and Particle Swarm Optimization (PSO).

INTRODUCTION

The automobile is one of the slightest maintainable human frameworks. However, it has likewise turned out firmly join with present day social orders and economies, making it difficult to lure towards sustainability that is more prominent. [1] Considering the restricted adaptability of automobile production frameworks as far as creation and obtainment abilities, the synchronized change of limit with the unpredictability of the market condition is not viable. [2] In the market, global sourcing and lean operations are the fundamental drivers of supply chain disruptions. [3]

Supply chain disruptions have affected the execution of companies. [4] Modern ideas of supply chains are clearly in light of these benefits, yet more frequently, they signify any development of items, administrations, data and money related exchanges from all levels of providers, unique makers to extreme end-users or consumers. [5] The supply chain is the network of associations that are included, through upstream and downstream linkages, in the distinctive procedures and exercises that deliver an incentive as items and administrations in the hands of a definitive consumer. [6]

Planning and scheduling are the two consecutive activities in a typical manufacturing framework. In spite of the fact that there is a solid connection between process planning and scheduling, they were done consecutively in most manufacturing frameworks. In the previous years the scheduling was directed independently after the procedure plans had been generated. [7] The process planning exercises incorporate choice of machining procedures, determination of machine apparatuses, operation grouping, choice of cutting devices, calculation of process times etc. [8] In the iterative approach, the change for the schedule is performed by constantly swapping the respective plan keeping in mind the end goal to fulfill the schedule delivery performance. [9] The point of scheduling is to locate an attainable schedule which can get multi-objective optimization, for example, minimizing machines' makespan and total jobs' tardiness. [10] Traditionally, multi-criteria scheduling problems have been considered with the goal of minimizing criteria that apply to each of the jobs being scheduled. [11] The fundamental issue of concern is the exchange off between job completion times and the costs of compression. [12]

In traditional job shop scheduling, in supposition, the framework's capacity is to complete jobs on time is obliged just by the number of machines that are accessible to process jobs.[13] Job shop scheduling problem (JSSP) is an NP-hard problem, and a standout amongst the most unmanageable combinatorial optimization problems considered to date. [14] The job shop scheduling problem (JSSP) is the most difficult one which has been the subject of many research contemplates amid the current decades. The problem portray essentially as takes after: given n jobs to be processed on m machines. Each comprises of a foreordained grouping of undertaking operations, each of which requires preparing without intrusion for a given period on a given machine. [15] The goal of this problem is to discover optimal schedule by minimizing the makespan and total completion time. [16] However, right now the scientists' consideration is centered on a few hybrid algorithms that appear to be more appropriate for real practical manufacturing environments suggesting bigger arrangements of limitations, which are hard to optimize by simply utilizing a simple meta-heuristic. [17] Hybrid algorithms are formulated by coordinating more than one evolutionary computing technique. The unwavering quality of the proposed algorithm is exhibited in the parts of quality of solution and computational efficiency. [18] The planning frame work develops a lean score board that establishes the gap between current performance and desired performance target. [19]

LITERATURE REVIEW

Hamed Piroozfard *et al.* [19] 2014, had arranged job shop scheduling problems as massively entangled issues in machine scheduling area, and they were named NP-hard problems. Finding optimal solutions for the job shop scheduling problems with correct techniques cause high cost, thusly, searching for approximate solutions with meta-heuristics are supported. In this paper, a hybrid framework which depends on a mix of genetic algorithm and simulated annealing is proposed by keeping in mind the end goal to minimize maximum completion time i.e. makespan. The proposed hybrid genetic algorithm was tried with an arrangement of benchmarking issues, and simulation results uncovered effectiveness of the proposed hybrid genetic algorithm contrasted with conventional genetic based algorithm.

R. J. Kuo *et al.* [20] 2013, had recommended to solve the job shop scheduling problem with both due information time window and discharge time. The goal was to minimize the sum of earliness time and delayed time in order to lessen the capacity cost and improve the consumer loyalty. A novel hybrid meta-heuristic, which consolidates Ant Colony Optimization (ACO) and Particle Swarm Optimization (PSO), called Ant Colony – Particle Swarm Optimization (ACPSO), was proposed to take care of this issue. Computational outcomes demonstrate that ACPSO performs superior to ACO and PSO.

Xueni Qiu *et al.* [21] 2012, had envisaged a static job shop scheduling problem (JSSP) in a class of JSSP which was a combinational optimization problem with the suspicion of no interruptions and beforehand known learning about the jobs and machines. Another hybrid algorithm in view of artificial immune systems (AIS) and particle swarm optimization (PSO) hypothesis was proposed for these issues with the goal of makespan minimization. By contrasting and other prevalent methodologies announced in existing written works, this algorithm demonstrates incredible intensity and potential, particularly for small size problems as far as calculation time.

Rui Zhang *et al.* [22] 2011, had suggested that the job shop scheduling problem (JSSP) was a famously troublesome problem in combinational optimization. Regarding the goal work, most existing exploration has been centered on the makespan criterion. Nevertheless, in contemporary manufacturing frameworks, due-date-related exhibitions were more vital in light of the fact that they are fundamental for keeping up high administration notoriety. Considering the high complexity, a hybrid differential evolution (DE) algorithm is proposed for the problem. As indicated by the broad computational tests, the proposed approach is productive in tackling the job shop scheduling problem with aggregate weighted lateness objective.

PROPOSED METHODOLOGY

This proposed study aims at condensing the time in Automotive parts supply industries and there are two distinct features included specifically job and machine. In this process, we need to schedule the machine for every job. Each job in each machine has its own specific process time, control of all process time said to be makespan time that should minimize. The standard intention of the job shop scheduling is to minimize the makespan time and discover the optimal schedule of the operations. If this method occurs in manual will then it would set aside enormous time to finish the extraction of optimal solution with a particular true objective to minimize the event of huge time consuming and to integrate optimization technique. This paper employed distinctive optimization technique for the improvement of NP complex hard benchmark problems with various sizes (la30, la31, la32, la33, la34, la35, la36, la37, la38, la39 and la40). Distinctive optimizations techniques include in this procedure are Hybrid Algorithm (SSO-GWO), Social Spider Optimization (SSO), Gray Wolf Optimization (GWO) and Particle Swarm Optimization (PSO) algorithm. In this analysis, it is quite evident that the proposed Hybrid Algorithm (SSO-GWO) exposes minimized makespan time in all kind of examination over other comparative techniques gave a way to examine the principle of Hybrid Algorithm (SSO-GWO) and its impact in solving the makespan time in Job Shop Scheduling Problems (JSSP). The entire work has demonstrated obviously in figure 1 and implemented in the platform of MATLAB 2016a.

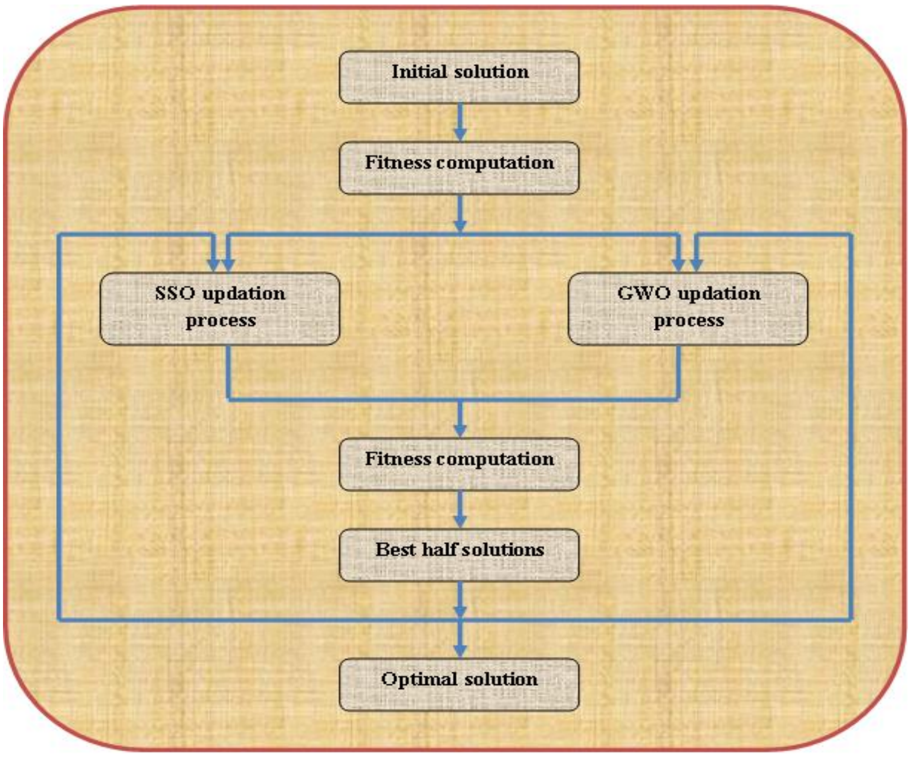

Figure 1: Flow of hybrid algorithm

Initialization

The initial solution consist of randomly hobbled machine assign to a job consider as one solution. This random shuffled solution fed to fitness and then further process of updating in couple of updating process SSO and GWO. For example, $M_1(J_1, J_3, J_2)$, $M_2(J_2, J_3, J_1)$, $M_3(J_2, J_1, J_3)$ likewise desired solution will develop in this section.

Fitness computation

The randomly generated solutions provided to evaluate its fitness (F_i) (i.e. makespan time). The fitness supports forthcoming updating process to incite generating solution to achieve minimized makespan time.

```
Make span time=Null matrix (size of problem)
      for     i=1 : Number of jobs
         for  j=1 : Number of machine
            if i=j=1
               make span time (i,j) = hybrid (i,j)
            else if i≠1 and j=1
               make span time (i,j) = sum (hybrid (1: i,j))
            else if i=1 and j≠1
               make span time (i,j) = sum (hybrid (i,1: j))
            else
               Neighbour 1 = make span time (i-1, j)
               Neighbour 2 = make span time (i, j-1)
            if  Neighbour 1 ≤ Neighbour 2
               make span time (i,j) = Neighbour 2 + hybrid (i,j)
            else
               make span time (i,j) = Neighbour 1 + hybrid (i,j)
            end
         end
      end
               Fitness (Fi) = make span time (end).
```

From the aforementioned computation, the result accomplish for makespan time.

Case 1) Social Spider Optimization (SSO) updation process

Preliminary process

Appealing features for social-spiders are the exceedingly female-biased populations. The algorithm starts by describing the number of female and male spiders that will be depicted as people in the search space. The number of females SE_{Nf} arbitrarily picked inside the extent of 65–90% of the whole population N_S. Thus, SE_{Nf} figured by the associated condition:

$$SE_{Nf} = floor[(0.9 - rand.025).N_{SE}] \quad (1)$$

In spite of the fact that floor () maps a real number to an integer number where rand is an uneven number between [0, 1]. The number of male spiders SE_{Nm} handled as the enhancement between N_{SE} and SE_{Nf}... It is outline beneath,

$$SE_{Nm} = N_{SE} - SE_{Nf} \quad (2)$$

Therefore, the complete population SE, collected by N_{SE} elements separated in two sub-groups Female (F) and Male (M).

The algorithm begins by presenting the set SE of N_{SE} spider positions. Such values are arbitrarily and reliably appropriated between the pre-demonstrated upper initial parameter bound pt_j^{high} and the lower initial parameter bound pt_j^{low}, likewise as it depicted by the supplementary expressions.

$$f_{i,j}^0 = pt_j^{low} + rand(0,1).(pt_j^{high} - pt_j^{low}) \quad (i = 1,2...SE_{Nf}, j = 1,2,..n) \quad (3)$$

$$m_{k,j}^0 = pt_j^{low} + rand(0,1).(pt_j^{high} - pt_j^{low}) \quad (k = 1,2...SE_{Nm}, j = 1,2,..n) \quad (4)$$

While, zero signals the initial population; j, i and k are the parameter and individual record in a specific order and the function rand create irregular values between (0,1). Hereafter $f_{i,j}$ is the j^{th} parameter of the i^{th} female spider position. By evaluating the radius of mating by then,

$$radius = \frac{\sum_{j=1}^{n}(pt_j^{high} - pt_j^{low})}{2.n} \quad (5)$$

The weight (wt_i) value is apply to upgrade the probability of expanding opportunity to create fine tune solution for next iterations to achieve optimal makespan time in small interim.

$$wt_i = \frac{J(SE_i) - worst_{SE}}{best_{SE} - worst_{SE}} \quad (6)$$

While $J(SE_i)$ is the probability investigation obtained by the appraisal of the spider position S_{pi} relating to the objective function $J(.)$. The qualities $worst_{SE}$ and $best_{SE}$ described as takes after

$$best_{SE} = \min_{k\in\{1,2,...N\}}(J(SE_k))\, and \quad worst_{SE} = \max_{k\in\{1,2,...N\}}(J(SE_k)) \quad (7)$$

Modeling the vibrations via communal web

To convey information with the colony members, the common web was used as a segment. The vibrations depend on the weight and distance of the spider. With a particular ultimate objective to copy this technique, the vibrations saw by the individual i because of the information transmitted by the member j shown by the associated condition.

$$Vibtn_{i,j} = wt_j.e^{-ds_{i,j}^2} \quad (8)$$

While the $ds_{i,j}$ is the Euclidian distance among the spiders i and j, where $ds_{i,j} = ||SE_i - SE_j||$.

Cooperative operators

There are two sorts of cooperative operators, to be specific, male cooperative operator and female cooperative operator where examined beneath.

Male cooperative operator

Male individuals, with a weight of incentive over the center value inside the male population, were viewed as the prevalent individuals B. On the other hand, those under the center value are naming as non-winning NB males. To actualize such figuring, the male population M $(M= \{m_1, m_2,...m_{Nm}\}$ is sort out by their weight an incentive in decreasing request. The Vibration $Vibm_i$ saw by that individual $i(B_i)$ as a result of the information passed on by the part $m(B_m)$ with m reliably the closest female unmistakable to i.

$$Vibm_i = wt_m.e^{-ds_{i,m}^2} \quad (9)$$

As the number of female individuals, SE_{Nf} extends arrangements of the male population M as for the whole population SE, the middleweight is requested by SE_{Nf+m}. As shown by this, change of positions for the male spider can exhibit as take after

$$m_i^{k+1} = \begin{cases} m_i^k + \alpha.Vibm_i.(SE_m - m_i^k) + \delta.(rand - \tfrac{1}{2}) & if \quad wt_{SE_{Nf+i}} > wt_{SE_{Nf+m}} \\ m_i^k + \alpha.\left(\frac{\sum_{h=1}^{SE_{Nm}} m_h^k.wt_{SE_{Nf+h}}}{\sum_{h=1}^{SE_{Nm}} wt_{SE_{Nf+h}}} - m_i^k\right) & if \quad wt_{SE_{Nf+i}} \le wt_{SE_{Nf+m}} \end{cases} \quad (10)$$

Where the individual SE_m represents the closest female individual to the male part *i,* while, $(\Sigma_{h-1}^{SE}{}_{Nm}\ m_h^k.Wt_{SENf+h}/\ \Sigma_{h-1}^{SE}{}_{Nm}\ Wt_{SENf+h})$ contrast to the weighted mean of the male population *M*. Two unique behaviors were made by using this operator. At first, set *B* of particles is polarizing with others keeping in mind the end goal to animate mating and such an execution licenses joining differing qualities into the population. On other hand, the set *NB* of particles polarize by methods for male population *M*. Change his the truth is using to some degree control the inquiry technique as demonstrated by the typical execution of a subgroup of the population. Such framework goes into a channel, which keeps up a vital separation from that extraordinary individual or greatly bad individual and influences the search system.

Female Cooperative Operator

An innovative operator is portraying with a specific end goal to copy cooperative behavior of the female spider. At each iteration, the operator considers the location vary of the female spider *i*. Such location vary may pull in or shock the procedure as a mix of three single segments. The main incorporates the change concerning the nearest part to *i* that holds a higher weight and makes the vibration $Vibf_i$. The second one considers the adjustment with respect to the best individual of the whole population *SE* who conveys the vibration $Vibl_i$. At long last, the third one consolidates an irregular development.

$$Vibf_i = wt_f.e^{-ds_{ii,f}^2}\ ,\ Vibl_i = wt_l.e^{-ds^2{}_{i,l}} \tag{11}$$

A uniform random number r_{ma} made inside the range [0, 1]. If, r_{ma} is lesser than an edge PF, an interest improvement made; for the most part, a repulsion development is conveying. Thus, such operator can be show as represent underneath,

$$f_i^{k+1} = \begin{cases} f_i^k + \alpha.Vibf_i.(SE_f - f_i^k) + \beta.Vibl_i.(SE_l - f_i^k) \\ + \delta.(rand - \tfrac{1}{2})\ with \quad proability\ \ PF \\ f_i^k - \alpha.Vibf_i.(SE_f - f_i^k) - \beta.Vibl_i.(SE_l - f_i^k) \\ + \delta.(rand - \tfrac{1}{2})\ with \quad proability\ \ 1-PF \end{cases} \tag{12}$$

In spite of the fact that *k* represents the iteration number where *α, β, δ* and *rand* are arbitrary numbers between [0, 1]. The individual SE_f and SE_l address the nearest part to *i* that holds a higher weight and the best individual of the whole population *SE,* independently.

Mating process

In this mating methodology, each included spiders weight (elements of *Tg*) describes the probability of effect for each individual into the new brood. The spiders holding a heavier weight will likely influence the new item, while parts with lighter weight have a lower probability. The effect probability P_{Spi} of every part is assign by roulette methodology, which is described as take after.

$$pt_{SE_i} = \frac{wt_i}{\sum_{j \in T^k} wt_j} \qquad where \quad i \in T^g \tag{13}$$

Once the new spider is made, it is diverged from the new spider competitor SE_{new} holding the most exceedingly awful spider SE_{wo} of the colony, as demonstrated by their weight values (where $wt_{wo} = min_{l\in\{1,2,\ldots,N\}}\ (w_l)$). If the new spider is better than the most exceedingly worst spider, the new one replaces the most recognizably worst spider. Thus the new spider is discarded, the population does not change. In the event that ought to emerge an event of substitution, the new spider expects the gender and index from the supplanted spider. Such reality ensures that the whole population *SE* keeps up the top notch among female and male people. Under this operation, recently made particles locally manhandle the search space inside the mating range with a specific end goal to find better individuals.

Case 2) Grey Wolf Optimization (GWO) updation process

It is expected that the alpha (best competitor arrangement), beta and delta have the improved data about the potential region of the prey with a particular true objective to rehash numerically the pursuing behavior of the dim wolves. In this manner, the initial three best arrangements accomplished up until this point were stored and require the other interest masters (checking the omegas) to reconsider their positions according to the position of the best chase operator. For excess, the new arrangement *p(t+1)* is surveyed by using the formulae said underneath.

$$J^\alpha = \left| P_1\, p_\alpha - p \right|, \quad J^\beta = \left| P_2\, p_\beta - p \right|, \quad J^\delta = \left| P_3\, p_\delta - p \right| \tag{14}$$

$$p_1 = p_\alpha - Gr_1.(J_\alpha),\ p_2 = p_\beta - Gr_2.(J_\beta),\ p_3 = p_\delta - Gr_3.(J_\delta) \tag{15}$$

To have hyper-spheres with various irregular radii, the discretionary parameters *Gr* and *p* help the applicant solutions. The examination and the utilization are guaranteed by the versatile estimations of *Gr* and *a* and it permit the GWO to travel them effectively among the examination and the use. With reducing *Gr*, half of the iterations are focused on the examination *(/Gr/<1)* and the other half are committed to the use. Encasing the lead, the resulting conditions are used remembering the true objective to give numerical model.

$$J = \left| P\,.\,p_k(t) - p(t) \right| \tag{16}$$

The coefficient vectors are found by the equation (17)

$$Gr = 2a\,.\,r_1 - a, \qquad\qquad p = 2\,.\,r_2 \tag{17}$$

Where *t* demonstrates the present iteration, *Gr* and *P* are coefficient vectors; P_k is the position vector of the prey *C* and shows the position vector of a grey wolf. The segments of *a* are directly diminished from 2 to 0 through the span of iterations and r_1, r_2 are arbitrary vectors in [0, 1].

The GWO has just two principle parameters (*Gr* and *P*) to be balance. Nevertheless GWO algorithm was kept as straightforward as conceivable with the least operators to be balance.

Once the fusion updating process has taken place, these new solutions would feed in to fitness process there and the solution will evaluate and assigned with fitness score. Then half of the best solution in that updating solution will utilize for next round of iteration. This loop will repeat until optimal solution is attained.

With the aforementioned both updating process and the proposed system enhances its updating solution in couple of aspects. First, fashioning couple of conventional updating solution incorporation definitely enhances the solutions and tries to catch up the optimal elements casually. Second, the time consumption to manipulate the entire working process will certainly reduce with this proposed fusion technique. This sort of protocol will be a quite evident approach in the perspective of real time environmental approach in auto industries.

RESULTS AND DISCUSSION

This section incorporates various analyses to show the performance evaluation of proposed hybrid (SSO-GWO) with SSO, GWO, PSO, HGA, Param.active, GRASP, ACA and LSGA techniques. Initial analyses were carried out techniques wise and then comparison was made for all implemented problems. Further leads to analyses the actual behaving of entire algorithm throughout the process and it was evaluated by the convergence graph. This entire implementation executed in the working platform MATLAB 2016 had system configuration of Intel (R) core (TM) i3-2328M CPU @ 2.20GHz with 6GB RAM. Let us discuss one by one in detail, From the aforementioned table it is quite evident that the proposed hybrid (SSO-GWO) technique achieves minimized makespan time with respect to Benchmark (BM) problem size. In all eleven BM problems, the proposed technique reveals better result than other techniques. Even though, with same problem size having different scheduling time the proposed technique never derailed in its stability and try to produce minimum makespan time for entire problems.

Table 1: Minimum makespan time from/for different techniques/problems

Technique	LA30	LA31	LA32	LA33	LA34	LA35	LA36	LA37	LA38	LA39	LA40
BM [23]	1355	1784	1850	1719	1721	1888	1268	1397	1196	1233	1222
Hybrid (SSO-GWO)	1357	1784	1852	1719	1725	1890	1271	1400	1196	1233	1222
HGA [23]	1355	1784	1850	1719	1721	1888	1287	1407	1196	1233	1229
SSO	1357	1786	1854	1720	1726	1891	1272	1407	1200	1235	1224
GWO	1358	1786	1854	1721	1727	1892	1275	1407	1201	1235	1228
PSO	1361	1788	1855	1722	1728	1895	1279	1411	1203	1239	1224
Param.active [23]	1355	1784	1850	1719	1721	1888	1279	1408	1219	1246	1241
GRASP [23]	1368	1784	1850	1719	1753	1888	1334	1457	1267	1290	1259
ACA [23]	1355	1784	1850	1719	1725	1888	1275	1412	1196	1240	1222
LSGA [23]	1451	1784	1850	1745	1784	1958	1358	1517	1362	1391	1323

Table 2: Average value for each problem using different techniques compared with BM

Problems	Hybrid (SSO-GWO)	HGA [23]	SSO	GWO	PSO	Param.active [23]	GRASP [23]	ACA [23]	LSGA [23]
LA30	99.85	100	99.85	99.77	99.55	100	99.04	100	93.38
LA31	100	100	99.88	99.88	99.77	100	100	100	100
LA32	99.89	100	99.78	99.78	99.73	100	100	100	100
LA33	100	100	99.94	99.88	99.82	100	100	100	98.51
LA34	99.76	100	99.71	99.65	99.59	100	98.17	99.76	96.46
LA35	99.89	100	99.84	99.78	99.63	100	100	100	96.42
LA36	99.76	98.52	99.68	99.45	99.13	99.139	95.05	99.45	93.37
LA37	99.78	99.28	99.28	99.28	99	99.21	95.88	98.93	92.08
LA38	100	100	99.66	99.58	99.41	98.113	94.39	100	87.81
LA39	100	100	99.83	99.83	99.51	98.95	95.58	99.43	88.64
LA40	100	99.43	99.83	99.51	99.83	98.46	97.06	100	92.36

In table 2, assorted algorithms has used for each problem and estimate the values. The algorithms such as Hybrid (SSO-GWO), HGA, SSO, GWO, PSO, Param.active, GRASP, ACA and LSGA are compared with benchmark solutions. Different problems were taken in to account for validating the performance of proposed technique with respect to benchmark makespan time solution. In overall, it is quite evident that the performance analysis reveals the individual capability of proposed Hybrid (SSO-GWO) over other comparative techniques. Even though the comparative technique HGA achieve eight optimal solutions, which is three units greater than the proposed technique, the average outcome, ruin the superiority of HGA over proposed technique and placed the proposed technique on top.

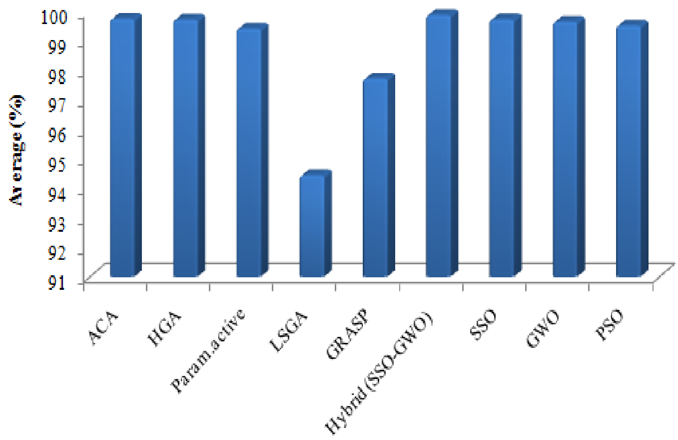

Figure 2: Average make-span time for all abovementioned problems with different techniques

From this figure 2, the average make-span time for different techniques evaluated obviously. There are diverse categories of algorithms utilized such as Hybrid (SSO-GWO), HGA, SSO, GWO, PSO, Param.active, GRASP, ACA and LSGA. The overall average make-span time for Hybrid (SSO-GWO) algorithm reveal 99.90%, HGA algorithm reveals 99.74%, SSO algorithm reveals 99.75%, GWO algorithm reveals 99.67%, PSO algorithm reveals 99.54%, Param.active algorithm reveals 99.44%. In GRASP algorithm the average value is 97.74%, ACA algorithm exposes 99.77% and finally the LSGA algorithm exposes 94.45%. Based on this overall performance the proposed Hybrid (SSO-GWO) algorithm achieves higher average value.

Convergence graph

This convergence graph is plotting to illustrate the performance of individual technique throughout the entire process. Here, four different techniques had been taken for comparison and evaluation. Proposed hybrid (SSO-GWO), SSO, GWO and PSO amid proposed technique reveals minimized makespan time of 1357 in almost 900th iteration for La30 (20 X 10).

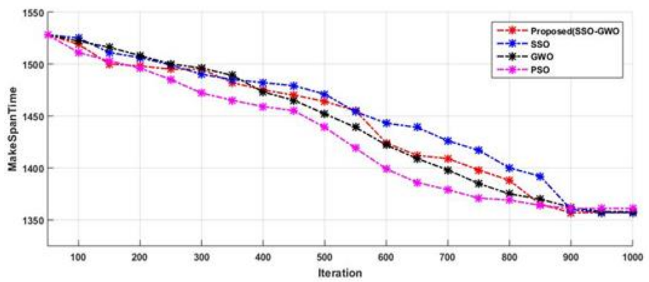

Figure 3: La30 Convergence graph

In this contest, entire comparative techniques collide in 900th iteration and the proposed technique start to fluctuate right from the beginning and settle almost in 900th iteration. Whereas the PSO lags 0.30% than proposed technique, but it performs really well right from the beginning and produce, consistently stable result throughout.

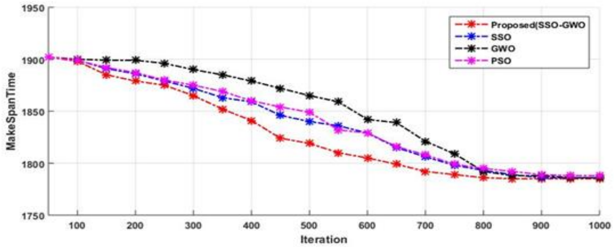

Figure 4: La31 Convergence graph

This convergence graph is plotting to illustrate the performance of individual technique throughout the entire process. Here, four different techniques had been taken for comparison and evaluation. Proposed hybrid (SSO-GWO), SSO, GWO and PSO amid proposed technique reveals minimized makespan time of 1784, in this La31 (30 X 10) the proposed technique was performed well right from the beginning and it finally converge in 800th iteration then saturate throughout. Here, SSO and PSO fluctuate until 800th iteration then it saturate in remaining portions. Whereas the PSO lags, 0.23% than proposed technique and other two techniques SSO and GWO both swarm intelligent technique lags 0.12% than the proposed technique.

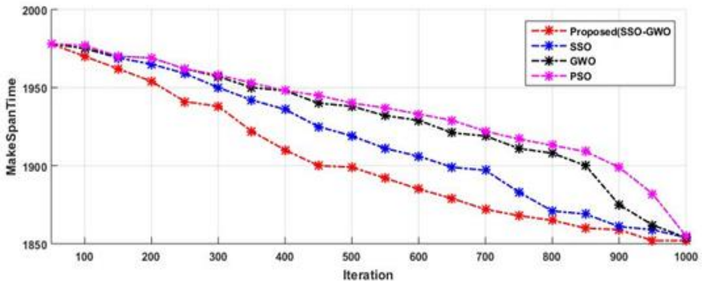

Figure 5: La32 Convergence graph

This convergence graph is plotting to illustrate the performance of individual technique throughout the entire process. Here, four different techniques had been taken for comparison and evaluation. Proposed hybrid (SSO-GWO), SSO, GWO and PSO amid proposed technique reveals minimized makespan time of 1852, in this La32 (30 X 10) the proposed technique perform well right from the beginning and it finally converge in 950th iteration then saturate throughout. Here, SSO, GWO and PSO start from the position and try to converge in 1000th iteration. Whereas the PSO lags, 0.16% than proposed technique and other two techniques SSO and GWO both swarm intelligent technique lags 0.11% than the proposed technique.

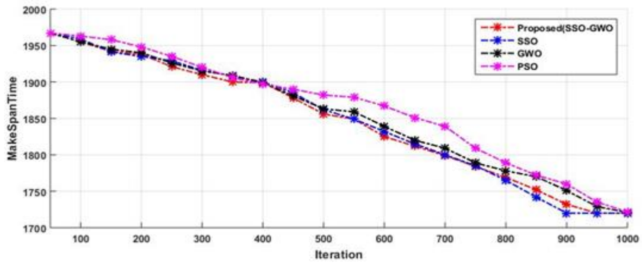

Figure 6: La33 Convergence graph

This convergence graph is plotting to illustrate the performance of individual technique throughout the entire process. Here, four different techniques had been taken for comparison and evaluation. Proposed hybrid (SSO-GWO), SSO, GWO and PSO amid proposed technique reveals minimized makespan time of 1719, in this La33 (30 X 10) the proposed technique starts with a delay then try to converge in 1000th iteration and lags 0.06% than actual minimized makespan time. Other comparative technique lags 0.06%, 0.12% and 0.18% respectively than the proposed technique. In this particular contest, all technique reveals almost colloidal makespan time for consecutive iterations.

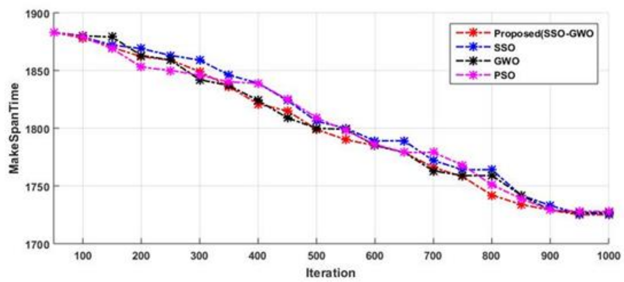

Figure 7: La34 Convergence graph

This convergence graph is plotting to illustrate the performance of individual technique throughout the entire process. Here, four different techniques had been taken for comparison and evaluation. Proposed hybrid (SSO-GWO), SSO, GWO and PSO amid proposed technique reveals minimized makespan time of 1725, in this La34 (30 X 10) the proposed technique starts with a fluctuation then try to converge in 900th iteration and lags 0.05% than actual minimized makespan time. Other comparative technique lags 0.05%, 0.11% and 0.17% respectively than the proposed technique. In this particular contest, all technique reveals almost colloidal makespan time for consecutive iterations.

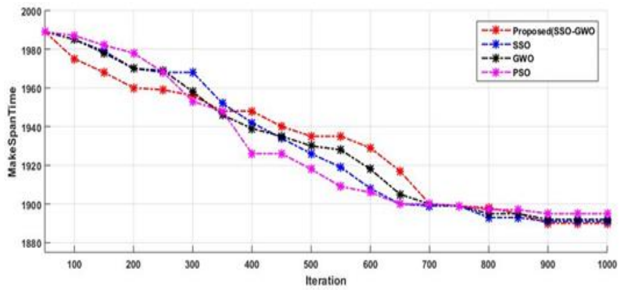

Figure 8: La35 Convergence graph

This convergence graph is plotting to exemplify the concert of individual technique during the entire process. Here, four different techniques had been taken for comparison and evaluation. Proposed hybrid (SSO-GWO), SSO, GWO and PSO amid proposed technique reveals minimized makespan time of 1890, in this La35 (30 X 10) the proposed technique starts with a fluctuation then try to converge in 900th iteration and lags 0.05% than actual minimized makespan time. Other comparative technique lags 0.05%, 0.11% and 0.26% respectively than the proposed technique. In this particular contest, all technique reveals almost colloidal makespan time for consecutive iterations.

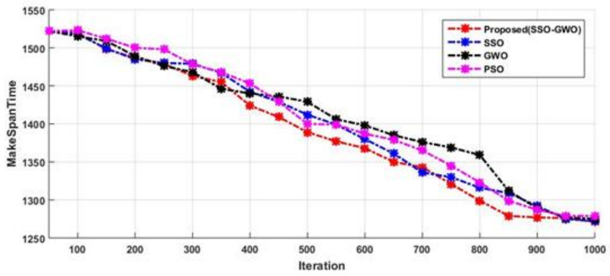

Figure 9: La36 Convergence graph

This convergence graph is plotting to illustrate the performance of individual technique throughout the entire process. Here, four different techniques had been taken for comparison and evaluation. Proposed hybrid (SSO-GWO), SSO, GWO and PSO amid proposed technique reveals minimized makespan time of 1271 in almost 900th iteration for La36 (15 X 15). In this contest, entire comparative techniques collide in 900th iteration and the proposed technique start to fluctuate right from the beginning and settle almost in 900th iteration. Whereas the PSO lags 0.63% than proposed technique, but it performs really well right from the beginning and produce, consistently stable result throughout.

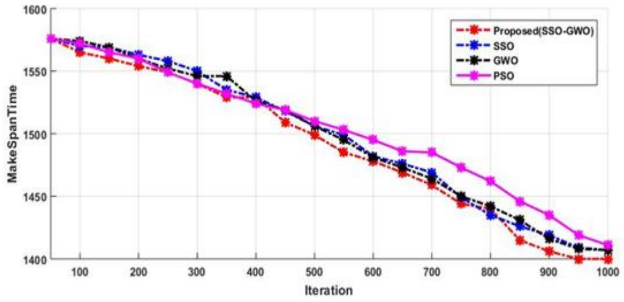

Figure 10: La37 Convergence graph

This convergence graph is plotting to illustrate the performance of individual technique throughout the entire process. Here, four different techniques had been taken for comparison and evaluation. Proposed hybrid (SSO-GWO), SSO, GWO and PSO amid proposed technique reveals minimized makespan time of 1400, in this La37 (15 X 15) the proposed technique perform well right from the beginning and it finally converge in 1000th iteration then saturate throughout. Here, SSO and PSO fluctuate until 1000th iteration then it saturate in remaining portions. The proposed technique lags just 0.5% in actual makespan time which is almost close enough and other two technique GWO and PSO both swarm intelligent technique lags 0.5% and 0.78% than the proposed technique.

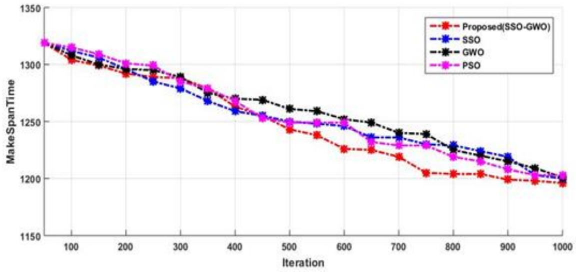

Figure 11: La38 Convergence graph

This convergence graph is plotting to illustrate the performance of individual technique throughout the entire process. Here, four different techniques had been taken for comparison and evaluation. Proposed hybrid (SSO-GWO), SSO, GWO and PSO amid proposed technique reveals minimized makespan time of 1196, in this La38 (15 X 15) the proposed technique perform well right from the beginning and it finally converge in 1000th iteration then saturate throughout. Here, SSO, GWO and PSO start from the position and try to converge in 1000th iteration. Other comparative technique lags 0.34%, 0.42% and 0.59% respectively than the proposed technique. In this particular contest, all technique reveals almost colloidal makespan time for consecutive iterations.

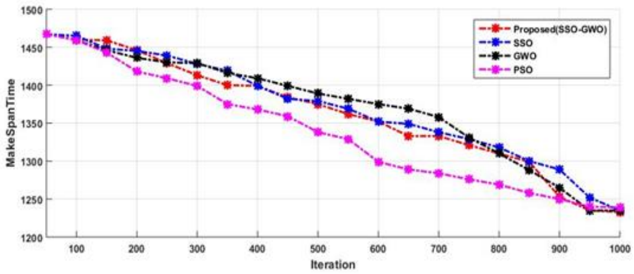

Figure 12: La39 Convergence graph

This convergence graph is plotting to illustrate the performance of individual technique throughout the entire process. Here, four different techniques had been taken for comparison and evaluation. Proposed hybrid (SSO-GWO), SSO, GWO and PSO amid proposed technique reveals minimized makespan time of 1233, in this La39 (15 X 15) the proposed technique starts with a fluctuation then try to converge in 1000th iteration. Other comparative technique lags 0.17%, 0.17% and 0.49% respectively than the proposed technique. In this particular contest, all technique reveals almost colloidal makespan time for consecutive iterations.

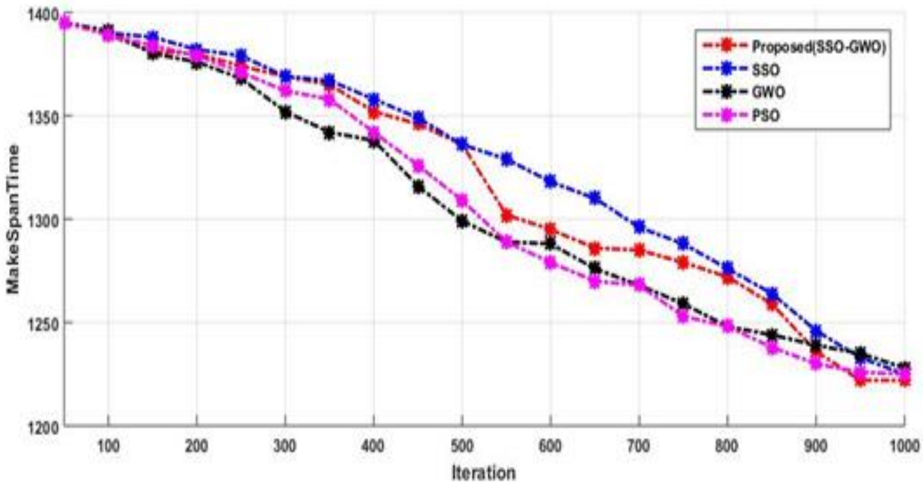

Figure 13: La40 Convergence graph

This convergence graph is plotting to exemplify the concert of individual technique during the entire process. Here, four different techniques put in to comparison for evaluation proposed hybrid (SSO-GWO), SSO, GWO and PSO amid proposed technique reveals minimized makespan time of 1222, in this La40 (15 X 15) the proposed technique starts with a fluctuation then try to converge in 950th iteration. Other comparative technique lags 0.17%, 0.49% and 0.17% respectively than the proposed technique. In this particular contest, all technique reveals almost colloidal makespan time for consecutive iterations.

CONCLUSION

The method of incorporating hybrid meta heuristic technique shows the evident result over conventional technique in the platform for solving job shop scheduling problems. From eleven different validations (La30, La31, La32, La33, La34, La35, La36, La37, La38, La39 and La40), the proposed hybrid (SSO-GWO) stands the contest throughout by achieving 99.90%. By proving the benchmark problems, it is quite convinced that this sort of methodology is well suited for real time problem of solving job shop scheduling in auto industries. Hence successful implementation effects vital revolution in automotive industries to make their final product in early time without foregoing its quality. This aforementioned process will make serious impact in the outcome of the supply chain management. In future, upcoming researcher put forth their research path in trim down the time consumption in manipulating process even in the case of hybrid/fusing methodology.

REFERENCES

[1] Paul Nieuwenhuis and Eleni Katsifou,"More sustainable automotive production through understanding decoupling points in leagile manufacturing",Journal of Cleaner Production, Vol.95, pp.232-241, 2015.

[2] Thomas Volling and Thomas S.Spengler,"Modeling and simulation of order-driven planning policies in build-to-order automobile production",Int. J.Production Economics, Vol.131, pp.183-193, 2011.

[3] Abroon Qazi, John Quigley and Alex Dickson,"Supply Chain Risk Management",Proceedings of International Conference on Industrial Engineering and Operations Management, pp.1-13, 2015.

[4] William Ho, Tian Zheng, Hakan Yildiz and Srinivas Talluri,"Supply chain risk management: a literature review",International Journal of Production Research, Vol.53, No.16, pp.5031-5069, 2015.

[5] Danuta Kisperska-Moron and Artur Swierczek,"The selected determinants of manufacturing postponement within supply chain context: An international study",Int. J.Production Economics, Vol.133, pp.192-200, 2011.

[6] Maria Caridi, Margherita Pero and Andrea Sianesi,"Linking product modularity and innovativeness to supply chain management in theItalian furniture industry",Int. J.Production Economics, Vol.136, pp.207-217, 2012.

[7] Mingrang Yu, Yingjie Zhang, Kun Chen and Ding Zhang,"Integration of process planning and scheduling using a hybrid GA/PSO algorithm",The International Journal of Advanced Manufacturing Technology, Vol.78, No.1, pp.588-592, 2015.

[8] P. S. Srinivas, V. Ramachandra Raju and C.S.P Rao,"Optimization of Process Planning and Scheduling using ACO and PSO Algorithms",International Journal

of Emerging Technology and Advanced Engineering, Vol.2, No.10, pp.343-354, 2012.

[9] Y.F. Wang, Y.F. Zhang and J.Y.H. Fuh,"A PSO-based Multi-objective Optimization Approach to the Integration of Process Planning and Scheduling",Proceedings of International Conference on Control and Automation (ICCA), pp.614-619, 2010.

[10] Li-Lan Liu, Rong-Song Hu, Xiang-Ping Hu, Gai-Ping Zhao and Sen Wang,"A hybrid PSO-GA algorithm for job shop scheduling in machine tool production",International Journal of Production Research, Vol.53, No.19, pp.5755-5781, 2015.

[11] Hari Balasubramanian, John Fowler, Ahmet Keha and Michele Pfund,"Scheduling interfering job sets on parallel machines",European Journal of Operational Research, Vol.199, pp.55-67, 2009.

[12] Guohua Wan, Sudheer R. Vakati, Joseph Y.-T. Leung and Michael Pinedo,"Scheduling two agents with controllable processing times",European Journal of Operational Research, Vol.205, pp.528-539, 2010.

[13] Benjamin J. Lobo, James R. Wilson, Kristin A. Thoney, Thom J. Hodgson and Russell E. King,"A practical method for evaluating worker allocations in large-scale dual resource constrained job shops",Journal of IIE Transactions, Vol.46, No.11, pp.1209-1226, 2014.

[14] Mohamed Kurdi,"An effective new island model genetic algorithm for job shop scheduling problem",Computers and Operation Research, Vol.67, pp.132-142, 2016.

[15] Amin Jamili, Mohammad Ali Shafia and Reza Tavakkoli-Moghaddam,"A hybrid algorithm based on particle swarm optimization and simulated annealing for a periodic job shop scheduling problem",The International Journal of Advanced Manufacturing Technology, Vol.54, No.1, pp.309-322, 2011.

[16] Chinyao Low, Min Ji, Chou-Jung Hsu and Chwen-Tzeng Su,"Minimizing the makespan in a single machine scheduling problems with flexible and periodic maintenance",Applied Mathematical Modelling, Vol.34, No.2, pp.334-342, 2010.

[17] Florentina Alina Toader,"A Hybrid Algorithm for Job Shop Scheduling Problem",Studies in Informatics and Control, Vol.24, No.2, pp.171-180, 2015.

[18] J. Sanjeevikumar and M. Sudhakaran,"Application of Integrated GA-PSO-TS Algorithm for Solving Hydro Thermal Scheduling Problem with Prohibited Operating Zones",International Journal of Applied Engineering Research, Vol.11, No.6, pp.3842-3847, 2016.

[19] Hamed Piroozfard, Adnan Hassan, Ali Mokhtari Moghadam and Ali Derakhshan asl,"A Hybrid Genetic Algorithm for Solving Job Shop Scheduling Problems",Advanced Materials Research, Vol.845, pp.559-563, 2014.

[20] R. J. Kuo and W. C. Cheng,"Hybrid meta-heuristic algorithm for job shop scheduling with due date time window and release time",Int J Adv Manuf Technol, Vol.67, pp.59-71, 2013.

[21] Xueni Qiu and Henry Y. K. Lau,"An AIS-based hybrid algorithm for static job shop scheduling problem",Journal of Intelligent Manufacturing, Vol.25, No.3, pp.489-503, 2014.

[22] Rui Zhang and Cheng Wu,"A Hybrid Differential Evolution and Tree Search Algorithm for the Job Shop Scheduling Problem",Mathematical Problems in Engineering, pp.1-21, 2011.

[23] Habibeh Nazif,"Solving Job Shop Scheduling Problem using an Ant Colony Algorithm",Journal of Asian Scientific Research, Vol.5, No.5, pp.261-268, 2015.