Review Article

PROBLEM SOLVING IN CROSS OVER LINE BALANCING USING HHO

R. Saravanakumar[1], Dr.D. Chandra Mohan[2]

[1]Department of Mechanical Engineering, Research Scholar, St. Peter's Institute of Higher Education and Research, Chennai, Tamilnadu. saravanan shakthi@yahoo.com
[2]Department of Mechanical Engineering, Professor & HOD, St. Peter's Institute of Higher Education and Research, Chennai, Tamilnadu.

Received: 10.12.2019 *Revised: 16.01.2020* *Accepted: 26.02.2020*

Abstract
The assembly line is a series of workflow manufacturing structures that are still common in large scale manufacturing of sustainable items. In the unbalanced tax system, more idle time is required to complete the task; A similar hanging problem arises when the tasks are not properly planned. This article proposes a technique to the line balance problem with crossover workstations. Different techniques were developed to resolve assembly line balance troubles. Existing methods for solving the assembly line balance problem can't be implemented to included type issues. Using the Harris Hawks optimization, improve the performance of a network-based line balancing problem. The assembly line optimization is performed using the Harris Hawks optimization cycle multipoint crossover method to acquire actual posterity and the high-quality chromosome is chosen to determine the performance of the sequential construction system.

Keywords: Assembly Line Balancing, Harris Hawks Optimization, Workstation.

 DOI: http://dx.doi.org/10.31838/jcr.07.04.51

INTRODUCTION

Manufacturing is the process of planning and deploying the optimal way to turn people into products by integrating people, processes, systems and organizations to deliver valuable products to society. Manufacturing technology completely limits creativity due to abundant automation and lack of staff within the production facility. A planning process is a set of rules that determine the task to be performed at a given moment. Although there are many planning methods proposed in the literature, the design of that algorithm is challenged by the need to support a variety of services, integrity and implementation complexity.

Jason Crabtree (2010) et al. had proposed the execution of a preventive maintenance development software tool (PMOST) is proposed dependent on the techniques for planning of preventive maintenance (PM) responsibilities in semiconductor production activities. We use the effects of four complicated simulation case research based on real enterprise statistics and use full fab models to illustrate the use, data requirements and effects produced by PMOST [1]. KeYi Xing (2012) et al. had suggested that deadlock-free control and scheduling is critical to improving the efficiency of automatic manufacturing systems (AMSs) with shared resources and way adaptability. Based on the Petri web models of AMSs, this paper inserts the most advantageous gridlock shirking principle in the genetic set of rules and builds up a brand new non-stop gene planning mechanism for AMSs [2].

Fei Qiao (2013) et al. had proposed that reentrant stream is a marvel where an item visits multiple machines on various instances over its manufacturing course. A creation framework with redundant stream is nicely known as a completely complex system. Its revisiting factor demanding situations making plans researchers. To overcome this trouble, a sort of drum-buffer-rope (DBR) technique has tested to be a probably vigorous system for complicated production machine planning. This paper looks at DBR-based totally planning for re-fabricating with a new method [3]. Kei Jing (2018) et al. have presented that Petri Net (PN) methods of flexible manufacturing systems (FMS) are proposed, and this article focuses on solving the planning problem of decreasing the total energy intake of FMS. In view of the exceptional energy consumption values of assets below exclusive operating situations, energy intake functions are taken into consideration. Dynamic Programming (DP) models of PNs-based planning issues had been set up, in which the PN version is considered the achievable reference, the preparing time vector, the path vector, and the alternate collection leading to the workable reference from the begin. As a nation, and based totally on the Bellman equation transformation firing [4].

Kei Jin (2010) et al. had proposed that Dirty Paper Coding (DBC) is the ability to reach the transmission mechanism in multi-user MIMO downlink channels. As a associate solution to DPC, the subsequent zero-compulsory TBC (SECF-TBC) has lately been proposed. Zero-interrupt manage supports the wide variety of aid users in designing pre-coding metrics. In this paper, we suggest three less complicated sub-top-quality user scheduling algorithms to use the multicore range advantage in SZF-DPC as the wide variety of customers will increases [5]. Xiao yang Dong (2011) et al. had suggested that security-sensitive package, along with digital transaction processing structure, stock quote update structures, require terrific security to guarantee authentication, integrity and confidentiality of data, and adopt the Multiple Distribution System (HDS) as their platform. This is normally because of the fact that single parallel-architecture primarily based systems are not enough to apply the parallelism available with walking applications. Most security-conscious package emerges as handling with dependency tasks on these HDSs, additionally called as Direct Acyclic Graph (DAG). Unfortunately, most techniques of making plans such DAGs in HDS fail to completely address the safety requirements. In this paper, we systematically design a security-driven scheduling framework that can dynamically measure the confidence level of every node in the system by using the usage of exceptional equations [6].

Yang Wang (2013) et al. have presented that task-level scheduling algorithms for finances and timeline constraints for a block diagram, minimizing workload on a fixed of multivariate (digital) machines added on cloud structures. Diversity manifests in the famous "pay for" charging model, where distinct efficient service machines have one of a kind service values. We prepare the set of tasks into a cash-value and examine two associated optimization issues, whether or not it's miles within the coin prices range or in making plans the period of the workflow [7]. Jing De Wang (2019) et al. had proposed that the wavelength tuning time of the optical network unit (ONU) is an important issue that can't be unnoticed within the ONU planning calculation for passive optical network (PON) for multiple wavelengths. In this paper, propose an adaptive scheduling set of rules for the concurrence of ONUs with exceptional tuning instances in a digital PON, known as the Multi-Tuning-Time ONU Planning (MOS) set of rules. In the simulation, the MOS set of rules can correctly avoid the extra array put off as a result of the wavelength tuning of the ONUs and reduce the waste of bandwidth resources [8].

Abdulai Tall (2014) et al. had suggested that Self-organizing networks (SON) technology is aimed at automating, upgrading, and troubleshooting radio access networks (RANs). SON algorithms normally use key performance indicators (KPIs) from the RAN. In some cases, it has been indicated that it is necessary to remember the effect of the backhaul state on the layout of the SON set of rules. We combine the proposed load estimator with the self-optimal load balancing (LP) set of rules [9]. Xuesong Qiu (2018) et al. have proposed that network virtualization is one of the most promising methods of solving the fiber-wireless (Fi-Wi) access networks resource mobilization issue. A virtual resource embedding system including three guide mechanisms is provided in this paper to make complete use of the substrate properties. The first is the selection method for the Priority Based Virtual Network Request (VNR), which prioritizes VNRs for deployment and embedding of VNRs. To increase the recognition value of VNRs, a dynamic embedding mechanism is given, based entirely on load balance. Second, while embedding is complete, its suggested that an embedding update mechanism tackle the advent of recent VNRs. This promotes the recognition charge of excessive priority VNRs and the go back of infrastructure provider (INP). Finally, a backup resource sharing mechanism is designed to improve reliability, which enables sharing of backup assets and responds to screw ops with restricted resources [10].

Gang Lee (2012) et al. had proposed that a multi-purpose sequencing optimization method (MSOM) is provided to conquer the realistic design issues of commercial electromagnetic devices. The MSOM multi-objective optimization version functions a non-stop optimization method and a modified important hybrid design (CCD) version. To enhance the optimization overall performance, its used to develop approximate multi-purpose optimization version. A updated CCD sampling approach is proposed to optimize the use of the optimum points and regression models of the obtained Pareto sample set. Subsequently, with the assistance of investigating a test function and a three-dimensional permanent magnetic pass flux device, the proposed solution can be seen to be successful and the computational expense of finite information evaluation can be dramatically saved [11]. M. Sarshanas (2015) et al. had suggested a new hybrid technique to detect complex two-dimensional representation of the green functions of micro-strip systems is being provided. This methodology completely blends the Taguchi algorithm with the process based on the BFGS gradient. Analytic Green functions are valid for a huge variety of frequencies and areas. The superiority of the method is achieved using the systematic numerical analysis of the Somerfield integrals to test its results [12].

Sung-Hue (2015) et al. have presented that the alternative directional approach of amplifiers (ADMM), we propose a new distributed optimization approach, known as proximal dual consensus ADMM (PDC-ADMM). The BTC-ATMM polyhedron constraint converts quadratic fines into sub-issues, making the sub-issues greater effectively solved, hence reducing the general computational overhead of the sellers. In addition, a random BTC-ATMM for time-varying networks with random ON / OFF sellers and communication errors, and a valid (random) BTC-ATM for less complicated calculations [13]. Donkey Wang (2018) et al. had proposed that U-type assembly lines have come to be an essential mode of production because of their more flexibility and productivity as compared to immediately strains. Since of a large-scale U-type assembly line the equilibrium is known to be NP-difficult, a powerful mathematical version and evolutionary mechanism are needed to address these issues. This article opinions the work popularity of relevant literature in recent years and introduces a hybrid evolutionary approach, specially, changed ant colony optimization inspired by simulated annual activity, to reduce the threat of being stuck at the neighborhood most appropriate for the Random large-scale U-type assembly line balancing problem [14]. Patong Chen (2018) et al. had suggested that the energy-aware load balancing and scheduling (ELBS) technique is proposed based primarily on fog estimation. First, the workload-associated power consumption model is established at the fog end, and an optimization characteristic is designed to cope with the load balance of the manufacturing cluster. Subsequently, the improved particle swarm optimization (BSO) algorithm is used to achieve the most advantageous solution, and the concern for attaining responsibilities is configured towards the output cluster. Finally, a multi-agent framework for achieving distributed scheduling of the production cluster [15].

PROPOSED METHOD

In this way, the assembly line is circuitously generated by using the concern-based totally encoding technique that is included with guidance information including priority constraints. The primary aim is to increase the wide variety of non-dominant solutions for the preliminary search, which is twice the population length. Calculate the assembly line, dealer admission and assembly allocation, and assembly line cycle time and idle time inside the population of the estimation version to calculate objective features. Sort the population into non-dominant solution sets based on the multi-useful cost of each particle. To update the location vector. All assembly operations are conducted within a unique cycle time. The cycle time is decreased through a particular unit in each evolution until the current short cycle time is reached, then a technology is obtained, and then the subsequent phase of evolution evolves. If a certain wide variety isn't reached in a head, it is repeated to calculate the fitness value of the hawks.

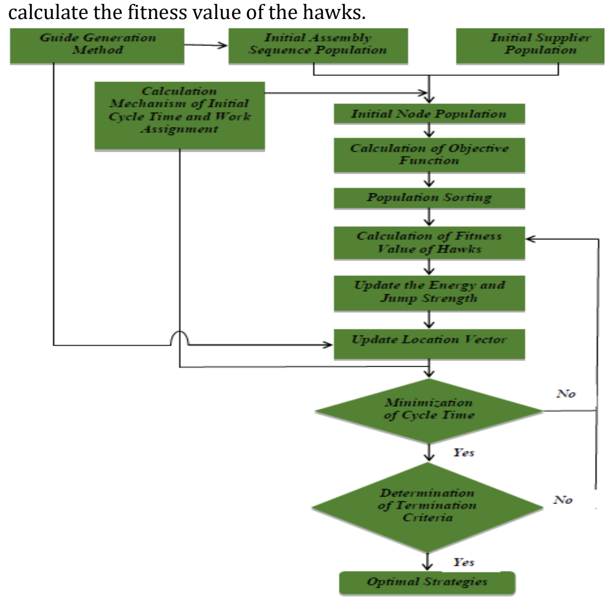

Figure 1: Block Diagram of Proposed Method

Problem Description

The issue considered in this thesis is the assembly line balancing issue, which offers with the introduction of equal workstations. The priority map is developed to assist imagine the forwarding tasks. The issue taken into consideration is the combined-model assembly line balancing issue with the goal of decreasing cycle time. M is a fixed of equal models (m = 1 .., M) that line up simultaneously. q_m is the ratio of the variety of units of every patterns to the general interest.

$$q_m = 0 \leq q_m \leq 1 \text{ and } \Sigma_m q_m = 1$$

Every version has its set of priority relationships, but all relationships can be mixed right into a single priority map with N tasks. The union of preceding maps can be made simplest if there are not anomalies of choice among the models. The processing time for each assignment (i = 1…., N) can vary between different samples. t_{im} is the cumulative assignment i for a sample m (time = 0 indicates that the assignment does not need to be assigned to pattern m).

Most strategies used to balance the composite vision assembly line balancing problem are to assign every task to an operator. Consequently, the longest processing time determines the rate of manufacture. To increase the production rate in addition, it is feasible to replicate the range of operators in a workstation. The replication process creates workstations that run parallel. The method of replication gives greater flexibility in designing or redesigning the assembly line, which can minimize the cycle time by means of a whole lot longer than the processing time. With the growth within the variety of parallel workstations, each operator can carry out a distinct wide variety of tasks. If the copying process of the workstations is not managed, it loses the main benefits of using assembly lines.

The problem taken into consideration in this paper consists of a method for controlling the workstation copy. This method may be described with a parameter defined by the user, such as the minimum processing time (MRT, minimum replication time) that triggers the workstation copying process. This method allows to copy a computing workstation, only if the processing time reaches the required duplication time.

Cross Over Line Balancing

The assembly line balance is to assign special undertaking to different stations where priority connections are maintained and a few measure of overall execution is upgraded. The most important motive of line balancing is to disseminate the desired tasks at the workstation by decreasing the time for the machines and operators. Rotation time is one of the more essential to the information for line balancing on any creation line.

The current composite version significantly complicates the layout in phrases of the processing time of the assembly line undertaking and the process duration of the entire assembly line. There is a cycle time (DC) within the single-sample assembly line balance problem, while within the case of a mixed-version assembly line balance problem, each version has its own cycle time, that is calculated primarily based on the quantity of output required for a shift in that version. The issue with both models is that if the production volume of assembly line 1 is 24 assemblies per shift, the corresponding cycle time is 18 minutes. If the production volume of assembly line 2 is 48 assemblies for every move, the corresponding cycle time is 24 minutes. In the case of a single version, the time of every cycle can be taken into consideration without any change, but in the mixed-vision assembly line equilibrium problem, the assembly may be modeled for a non-unusual cycle time, that is calculated based on the cycle times of individual samples. I the event that the cycle instances (Tc) of the samples are one and the equivalent, at that point the regular process duration (Tc) is same to the single value; in any case, the normal line of rotation times can be taken into consideration the standard cycle time for the equilibrium problem.

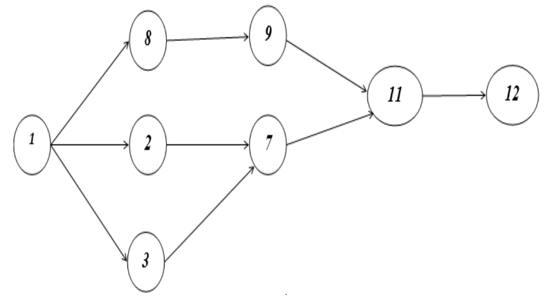

Figure 2: Precedence Network of Assembly Line 1

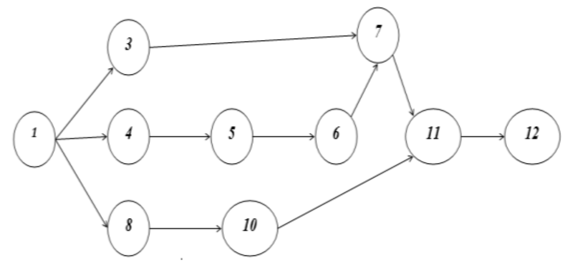

Figure 3: Precedence Network of Assembly Line 2

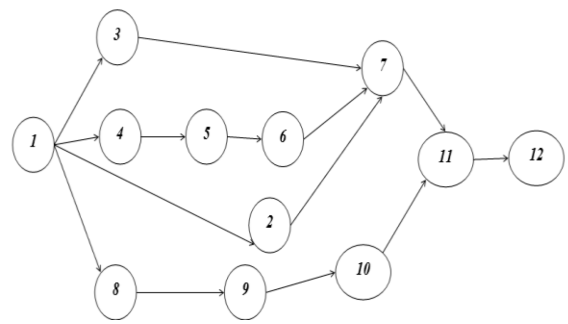

Figure 4: Combined Model of Assembly Line 1 and 2

Assume assembly line 1 and assembly line 2, whose priority systems are given in Figure 2 and Figure 3, respectively. The numbers on each side of the node constitute their respective working hours. The priority network of the mixed assembly line vision is given in Figure 4. In different works at the assembly line, various creators have considered the normal time of every vision in all models. Now the hybrid-model assembly lines for the genetic algorithm make use of the cycle shortcut approach to achieve the chromosome and its corresponding collection vector. As given in Figure 4, the ten-undertaking assembly line 1 and assembly line 2 are taken into consideration to be the unified priority network for integration. Each chromosome inside unique populace is produced through arbitrarily allotting undertaking to various genetic situations on the chromosome.

HARRIS HAWK'S OPTIMIZATION

The exploration and exploitation levels of the proposed HHO are stimulated with the aid of the exploration of a prey, the distinct offensive strategies of surprise hopping and Harris hawks. HHO is a populace principally based, slope-unfastened optimization method; therefore, it is able to be used for any enhancement issue subject to the correct components. Figure 5 indicates all levels of HHO.

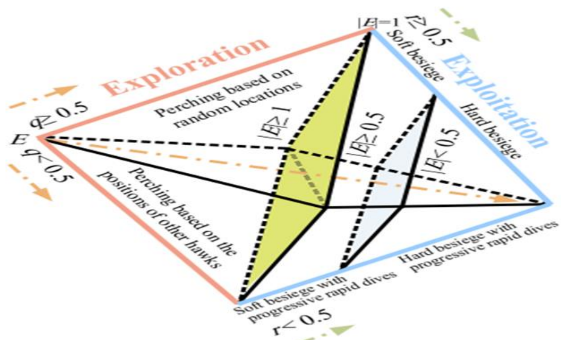

Figure 5: Different Phases of HHO

Algorithm: Pseudo-code of HHO set of rules
Inputs: The population size N and most wide variety of iterations T
Outputs: The vicnity of rabbit and its fitness value
Initialize the random Population X_i (I = 1, 2… N)
While (Stopping situation isn't met) do
Calculate the fitness values of hawks
Set X_{rabbit} as the vicinity of rabbit (fine region)
for (every hawk (X_i)) do
Update the initial energy E_0 and jump power J
E_0 = 2rand () – 1, J = 2 (1 – rand ())
Update the E use of Eq. (3)
if (|E| ≥ 1) then ➡ Exploration Phase
Update location vector use of Eq. (1)
if (|E| < 1) then ➡ Exploitation phase
if (r ≥ 0.5 and |E| ≥ 0.5) then ➡ Soft besiege
Update the location vector use of Eq. (4)
else if (r ≥ 0.5 and |E| < 0.5) then ➡ Hard besiege
Update the location vector use of Eq. (6)
else if (r < 0.5 and |E| ≥ 0.5) then ➡ Soft besiege with progressive fast dives
Update the location vector use of Eq. (10)
else if (r < 0.5 and |E| < 0.5) then ➡ Hard besiege with progressive fast dives
Update the location vector use of Eq. (11)
Return X_{rabbit}

Exploration phase
HHO's investigate calculation is proposed. In the event that we keep mind the character of Harris' hawks, they can be detected with the aid of their incredible eyes, however no longer by way of the occasional prey. So, after numerous hours the hawks wait, look at and reveal the desolate tract site to find a prey. At HHO, Harris' hawks are considered preferential or nearly greatest with candidate solutions and the nice candidate answer goal at every step. At HHO, Harris' hawks randomly arrive later and hold back to find a prey dependent on procedures. In the event that we except q to be an same opportunity for each perch method, they are based totally at the positions of the other circle of relatives contributors (which must be near them while attacked) and the rabbit that is the equation version. (1) q <0.5, or roost of arbitrary tall hedges (irregular areas in the gathering's domestic variety), which is demonstrated in Eq. (1) For q situation 0.5 condition.

$$X(t+1) = \begin{cases} X(t)_{rand} - r_1 |X(t)_{rand} - 2r_2 X(t)| & q \geq 0.5 \\ (X(t)_{rabbit} - X(t)_m) - r_3 (LB + r_4 (UB - LB)) & q < 0.5 \end{cases} \quad (1)$$

X (t + 1) is the placement vector of hawks inside the subsequent cycle, X_{rabbit} (t) is the rabbit function, X (t) is the modern function vector of hawks r_1, r_2, r_3, r_4, and q are inside (0 , 1) irregular numbers, which can be updated with every new release, LB. Furthermore UB show the higher and lower limits of the factors, the halo selected randomly from the X_{rand} (t) cutting-edge population, and the X_m imply position for the present populace of hawks.

We proposed a straightforward adaptation to create arbitrary areas inside the gathering's home range (LB, UB). The first rule is to create answers primarily dependent on an irregular space and different hawks. In the second rule of Eq. (1), we have got the satisfactory spatial differentiation to the point and the approximate scaled component based at the suggest function of the group and the range of variables, while r_3 takes r_4 close qualities to 1 and comparative circulation examples can happen. Right now, include about the measured working length to the LP. We then have taken into consideration the irregular scaling coefficient for the element to offer further broadening patterns and to investigate various areas of the function space. It is feasible to create specific refreshing standards; however we used a simple principle that reflects the conduct of prey. The common position of hawks is finished using Eq. (2)

$$X_m(t) = \frac{1}{N} \sum_{i=1}^{N} X(t)_i \quad (2)$$

Where $x_i(t)$ iteration denotes the location of every hawk at t and N indicates the overall range of hawks. It is viable to get the common region in distinct approaches; however we used the simple standard.

Transition from exploration to exploitation
The HHO calculation can change from exploration to exploitation, trade among extraordinary exploitative practices based absolutely at the getting away from the quality of the prey. The strength of a prey reduces impressively at some stage in the escaping behavior. To display this reality, the strength of a prey is demonstrated as:

$$E = 2E_0 \left(1 - \frac{t}{T}\right) \quad (3)$$

Where E represents the getaway strength of prey, T is the most extreme amount of cycles, and E_0 is the preliminary condition of its energy. In the HHO, E_0 is approximated by the interval (−1, 1) in every generation. At the point when the estimation of E_0 diminishes from 0 to −1, the rabbit is bodily flagged, while the value of E_0 increases from 0 to 1, this implies the rabbit is strengthening.

Dynamic escape vitality has a lowering tendency in the course of electric repetition. Energy to break out | E | ≥1, hawks seek unique areas to explore rabbit vicinity, therefore, HHO plays look at segment, while | E | <1, the calculation attempts to take advantage of the ambience of the solution during abuse steps. To put | E | When the take a look at takes region ≥1, | E | When exploitation happens in subsequent steps <1.

Exploitation phase
At this point, Harris' hawks do a surprising banging on the prey they found earlier. However, preys regularly try to get away risky conditions. Therefore, specific chase styles arise in real situations. Four possible techniques have been proposed in HHO to model the attack phase, according to the escape behavior of Harris' hawk prey and chase strategies.

Preys are commonly attempting to break out from compromising conditions. Assume that r is the chance of a prey to get away ($r < 0.5$) or no longer to break out effectively ($r \geq 0.5$) earlier than the surprise. Whatever the prey does, the hawks will make a tough ot soft blockade to capture prey. Depending on the retention power of the prey, this means that they prey on or around the prey from different directions. In real situations, hawks are approaching prey aimed at increasing their chances of killing the rabbit cooperatively by surprising hopping. After numerous minutes, the break out prey loses more and more energy; later, hawks accentuate the siege system to catch prey from the exhausted prey. To vision this method and to enable HHO to replace clean smooth and difficult siege strategies, the power parameter is used. In this regard, while $|E| \geq 0.5$, gentle siege happening, while $|E| < 0.5$, causing a hard blockade.

Soft besiege

At the point when $r \geq 0.5$ and $|E|$ 0.5, the rabbit nevertheless has sufficient strength, and attempt and break out through some random stray jumps, but in the end it won't. During these attempts, Harris's hawks tenderly surround it to make the rabbit increasingly exhausted, at that point make a unexpected hop. This conduct is modeled by means of the subsequent regulations:

$$X(t+1) = \Delta X(t) - E\,|JX_{rabbit}(t) - X(t)| \quad (4)$$

$$\Delta X(t) = X_{rabbit}(t) - X(t) \quad (5)$$

Where $\Delta X(t)$ is the contrast between the rabbit position vector and the current location of the emphasis t, r_5 is a arbitrary number (0,1), and $J = 2(1 - r_5)$ denoted the arbitrary jump strength of the rabbit throughout the escape process. To simulate the character of the rabbit movements, the J value modifications randomly at each generation.

Hard besiege

At the point when $r \geq 0.5$ and $|E| < 0.5$, the prey is extremely depleted and it has low break out limit. However, Harris's hawks were not surrounded by prey that wanted to make a surprise hoop. In this case, the current conditions are updated using Eq. (6):

$$X(t+1) = X_{rabbit}(t) - E\,|\Delta X(t)| \quad (6)$$

PARAMETER ESTIMATION

Cycle Time

It is one of the greatest significant information for line balancing on any production line. The time required finishing a item, or the time it takes to depart the item computing device and move to the subsequent computing device, is known as cycle time.

Cycle time, C = (effective time / manufacturing volume according to duration)

Idle Time

Idle time is the time needed subtracted from the time allocated.

Idle Time, I = Computational Time – Cycle Time

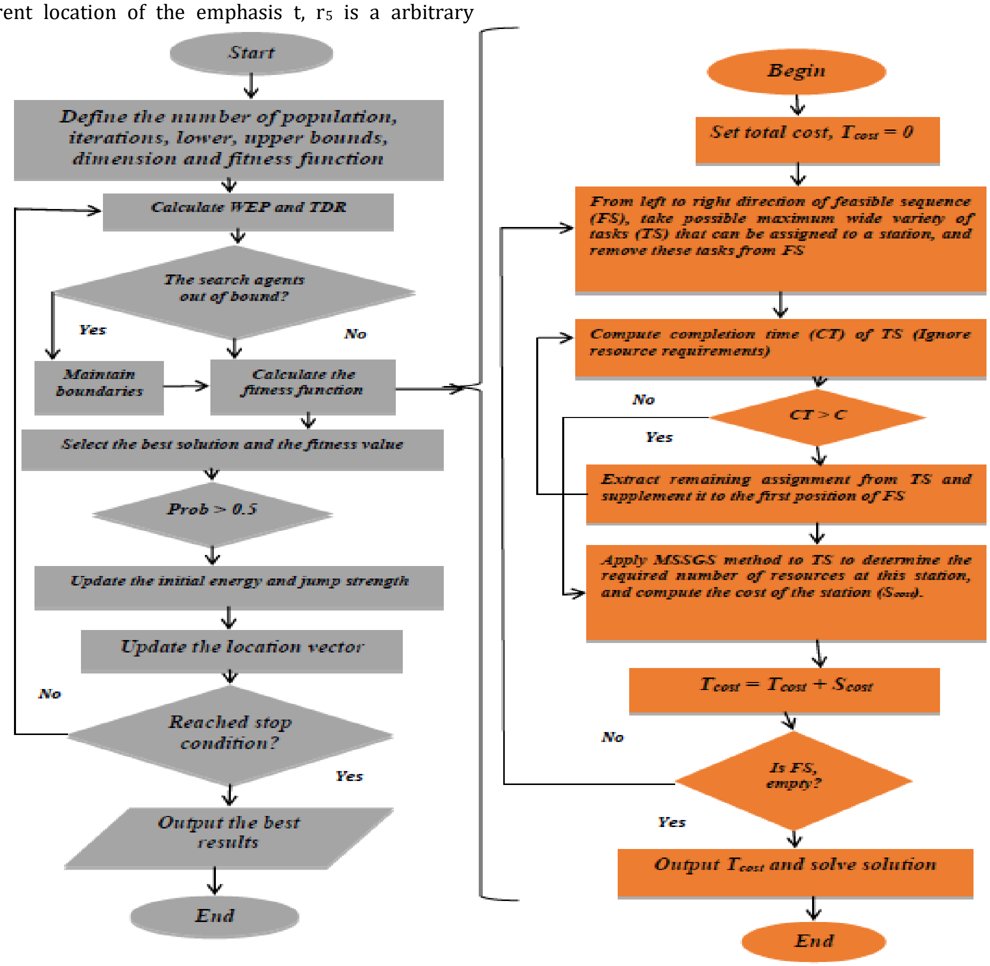

Figure 6: Algorithm for HHO optimization

RESULT

Step: 1 Number of work station connected in unbalanced condition for 14 workstation

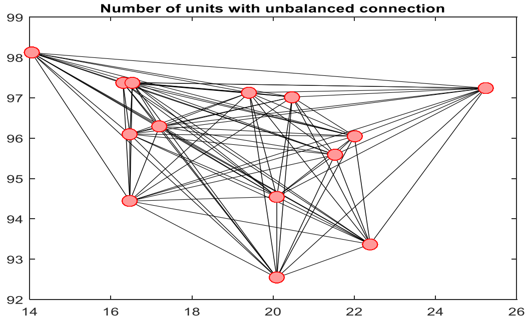

Fig. 7: Interconnected Workstation with Unbalanced Condition

Fig 7 shows that Number of the workstation are interconnected with the unbalanced condition. It has 14 workstation are assumed as the 14 nodes these are all interconnected each other without the proper arrangement in the network. In the unbalanced workstation the huge number of task cannot be completed on time. Because idle time is increase and some of the high efficient workstation are waiting for next task until low efficient workstation to complete present task. To overcome this problem optimization technique is used to schedule the task based upon the efficiency of the workstation.

Step: 2 Evaluate sample task 1

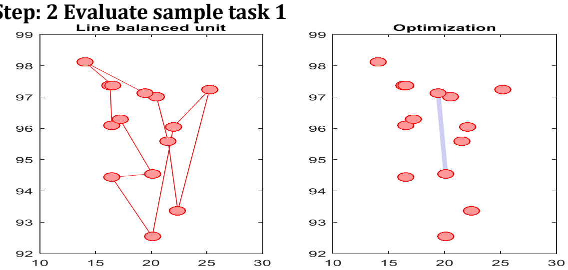

Figure 8: Line Balanced task for 1

Fig 8 shows that line balanced optimization of task 1. Here optimization technique HHO is used to schedule the task with proper line balanced. In this figure first two tasks are assign to do at the same time other workstation are waiting for getting the task. Based upon the scheduling technique large amount of task can be processed in networking. HHO reduces the cycle time of individual task. This optimization technique schedule the individual task based upon the workstation, efficiencies and its performance. It also reduces the idle time of individual workstation and increase overall execution time of networking based manufacturing companies.

Step: 3 Evaluate tasks

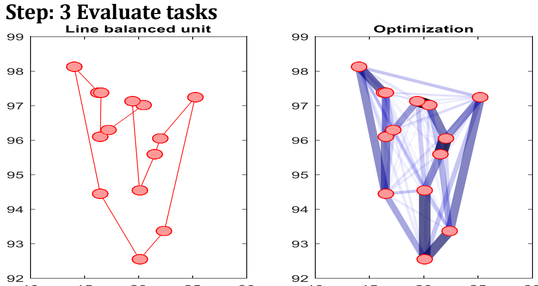

Figure 9: Line Balanced task for 50

Fig 9 show that the line balanced optimization for 50 tasks. The Harris Hawks optimization is used to schedule the task. The 50 tasks are assigned at the same time. Here there are no tasks which are scheduled for waiting. The dark line reveals the high efficiency of the balanced optimization that it can complete 2 tasks at a minute. Other nodes can complete a task in a time. The scheduling time depends upon the workstation, efficiencies and its performances. The HHO reduces the cycling time and the idle time. Also it improves the overall execution time of manufacturing.

Step: 4 Convergence curve

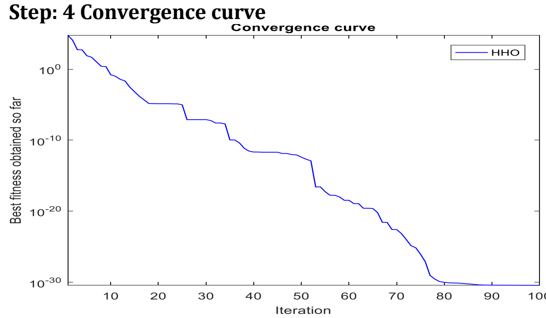

Figure 10: Variation of Convergence Curve with iteration

Fig 10 shows that the variation of convergence curve with iteration. To more visually observe the convergence properties of HHO, a convergence curve from 90-fold independent flows. Initially when the HHO reaches its best value within a faster merge rate and less than 3 iterations.

Step: 5 Cycle-time curve

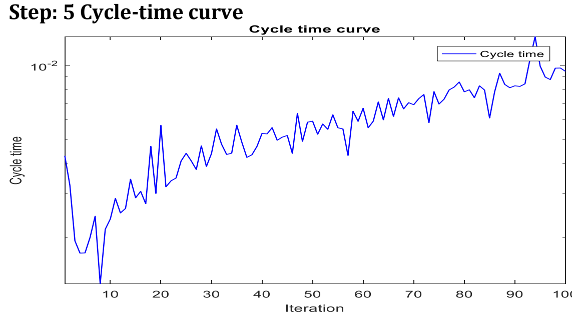

Figure 11: Variation of Cycle Time with Iteration

Fig 11 shows that variation of cycle time with iteration. The cycle time processes the task completely and it will not take waiting time for processing the task. It completes the task by allotting the task to the workstation which is idle. Therefore the overall work can be completed fastly with the use of HHO optimization technique. So that overall execution time increases and also the efficiency increases.

Step: 6 Idle time

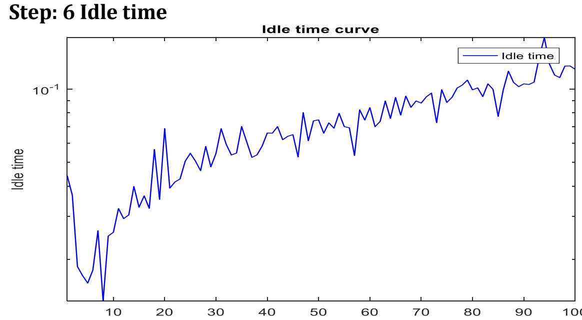

Figure 12: Variation of Idle Time with Iteration

Fig 12 shows that variation of idle time with iteration. Idle time is generally defined as the idleness of the workstation i.e., waiting the arrival of next work. By using HHO optimization technique idle time is reduced. In working unit it cause the higher efficiency for some units and cause the lower efficiency for other units.

High efficiency workstations take less time to complete huge works. While lower efficiency workstations take more time to complete the works. Based on these parameters HHO optimization schedules the task properly with best solution. So this will increase the yield of the network based companies. In this figure the idle time of the workstation is decreasing although large amount of work is given to workstations.

Table 1: For 500 tasks with 14 nodes

S. No	No. of Task	Average time for Execution time (m Sec)
1.	10	0.012589
2.	50	0.036048
3.	100	0.077210
4.	150	0.149094
5.	200	0.249926
6.	500	1.883148

CONCLUSION

This paper outlines an operational technique for fixing line balancing using the Harris Hawks optimization with crossover workstations. The primary aim is to increase the production value of the assembly line to a pre-described wide variety of operators. The cause of this paper is to reduce the idle time of the workstation within the line balance problem. Using the Harris Hawks optimization, improve the performance of a network-based line balancing problem.

REFERENCES

1. Marcus, Gaucherand, Fu, M.C. and S.I., 2004. Optimal preventive maintenance scheduling in semiconductor manufacturing. IEEE Transactions on Semiconductor Manufacturing, 17(3), pp.345-356.
2. Wang, F. Zhou, M. and 2011. Deadlock-free genetic scheduling algorithm for automated manufacturing systems based on deadlock control policy. IEEE Transactions on Systems, Man, and Cybernetics, 42(3), pp.603-615.
3. Wu, F. and Qiao, Q., 2013. Layered Drum-Buffer-Rope-Based scheduling of reentrant manufacturing systems. IEEE transactions on semiconductor manufacturing, 26(2), pp.178-187.
4. Wu, Y., Zhou, M., Wang, X., 2018. Modified dynamic programming algorithm for optimization of total energy consumption in flexible manufacturing systems. IEEE Transactions on Automation Science and Engineering, 16(2), pp.691-705.
5. Kaur H, Saini S, Peer S, Singh J. "Current Therapies and Novel Targets in Treatment of Breast Cancer." *Systematic Reviews in Pharmacy* 1.1 (2010), 40-49. Print. doi:10.4103/0975-8453.59511
6. Een Keen., Le Nam., 2010. Multiuser diversity for successive zero-forcing dirty paper coding: Greedy scheduling algorithms and asymptotic performance analysis. IEEE Transactions on Signal Processing, 58(6), pp.3411-3416.
7. Zeng. Kenli Li., and Bharadwaj, 2010. A novel security-driven scheduling algorithm for precedence-constrained tasks in heterogeneous distributed systems. IEEE Transactions on computers, 60(7), pp.1017-1029.
8. Yang. Wei.., 2014. Budget-driven scheduling algorithms for batches of Map Reduce jobs in heterogeneous clouds. IEEE Transactions on Cloud Computing, 2(3), pp.306-319.
9. Tong, Chaoqin and, W. Xingdi., 2019. Adaptive Scheduling Algorithm for the Coexistence of ONUs with Different Tuning Time in Virtual Passive Optical Network. IEEE Photonics Journal, 11(5), pp.1-8.
10. Tall, A., Altman, Z. and Altman, E., 2015. Self-optimizing load balancing with backhaul-constrained radio access networks. IEEE Wireless Communications Letters, 4(6), pp.645-648.3.
11. Xu, S., Li, P., Guo, S.Y. and Qiu, X., 2018. Fiber-wireless network virtual resource embedding method based on load balancing and priority. IEEE Access, 6, pp.33201-33215.
12. Lei, G., Chen, X.M., Zhu, J.G., Guo, Y.G., Xu, W. and Shao, K.R., 2012. Multiobjective sequential optimization method for the design of industrial electromagnetic devices. IEEE transactions on magnetics, 48(11), pp.4538-4541.
13. Sarshenas, M. and Firouzeh, Z.H., 2015. A Robust Hybrid Taguchi-Gradient Optimization Method for the Calculation of Analytical Green's Functions of Micro strip Structures. IEEE Antennas and Wireless Propagation Letters, 14, pp.1366-1368.
14. Chang, T.H., 2016. A proximal dual consensus ADMM method for multi-agent constrained optimization. IEEE Transactions on Signal Processing, 64(14), pp.3719-3734.
15. Zhang, H., Zhang, C., Peng, Y., Wang, D., Tian, G., Liu, X. and Peng, Y., 2018. Balancing problem of stochastic large-scale U-type assembly lines using a modified evolutionary algorithm. IEEE Access, 6, pp.78414-78424.
16. Wan, J., Chen, B., Wang, S., Xia, M., Li, D. and Liu, C., 2018. Fog computing for energy-aware load balancing and scheduling in smart factory. IEEE Transactions on Industrial Informatics, 14(10), pp.4548-4556.
17. Sahu, A. and Pradhan, S.K., 2018. Comparative analysis and optimization of Mixed Model assembly line using Genetic Algorithm. Materials Today: Proceedings, 5(11), pp.25075-25084.
18. Raj, A.V., Mathew, J., Jose, P. and Sivan, G., 2016. Optimization of cycle time in an assembly line balancing problem. Procedia Technology, 25, pp.1146-1153.
19. Samouei, P. and Ashayeri, J., 2019. Developing optimization & robust models for a mixed-model assembly line balancing problem with semi-automated operations. Applied Mathematical Modelling, 72, pp.259-275.
20. Şahin, M. and Kellegöz, T., 2019. A new mixed-integer linear programming formulation and particle swarm optimization based hybrid heuristic for the problem of resource investment and balancing of the assembly line with multi-manned workstations. Computers & Industrial Engineering, 133, pp.107-120.
21. Heidari, A.A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M. and Chen, H., 2019. Harris hawks optimization: Algorithm and applications. Future generation computer systems, 97, pp.849-872.
22. Ewees, A.A. and Elaziz, M.A., 2020. Performance analysis of Chaotic Multi-Verse Harris Hawks Optimization: A case study on solving engineering problems. Engineering Applications of Artificial Intelligence, 88, p.103370.
23. Mourtzis, D., Doukas, M. and Psarommatis, F., 2013. Manufacturing network design for mass customisation using a genetic algorithm and an intelligent search method. *Procedia Cirp*, *7*, pp.37-42.
24. Kumar, T., Patra, S., Ramalingam, R., Agrawal, N., Agarwal, A., Manjunath, C.N.Pulmonary hypertension due to presence of isolated partial anomalous pulmonary venous connection: A case report(2013) Journal of Cardiovascular Disease Research, 4 (4), pp. 239-241.
DOI: 10.1016/j.jcdr.2014.01.002