A New Heuristic Optimization Approach to the Single Hoist Cyclic Scheduling Problem

Aymen El Amraoui
Research Laboratory in Automatic Control, National Engineering School of Tunis, Tunis El Manar University, Tunis, Tunisia
aymen.elamraoui@bizerte.r-iset.tn (corresponding author)

Mohamed Benrejeb
Research Laboratory in Automatic Control, National Engineering School of Tunis, Tunis El Manar University, Tunis, Tunisia
ohamed.benrejeb@enit.utm.tn

Received: 18 August 2024 | Revised: 25 September 2024 | Accepted: 4 November 2024

 DOI: https://doi.org/10.48084/etasr.8767

ABSTRACT

This paper introduces an innovative heuristic optimization approach, referred to as Optimization Approach-Single Hoist Cyclic Scheduling Problem (OA-SHCSP), which aims to minimize the cycle time of the Single Hoist Scheduling Problem (SHCSP). The effectiveness of this proposed heuristic is compared with a previously established heuristic, the Earliest Starting Time (EST). The comparison results reveal that the proposed OA-SHCSP heuristic consistently outperforms the EST heuristic in minimizing cycle time, particularly when more than two products are produced simultaneously. Moreover, as the number of part tasks soaked during a cycle increases, OA-SHCSP demonstrates significantly improved computational efficiency over the EST heuristic. The reduction in average cycle time achieved by OA-SHCSP ranges from 28.73% to 60.29%, underscoring its effectiveness and potential for application in high-volume production environments.

Keywords-Single Hoist Scheduling Problem (SHCSP); cyclic; heuristic; optimization

I. INTRODUCTION

The Hoist Scheduling Problem (HSP) addresses the scheduling of tasks in surface treatment lines, where treatment and handling (robot movement) operations must be managed together. These tasks involve two main resources: Treatment stations, such as tanks, and robots for transporting items between stations. Due to strict sequencing and timing constraints, handling and treatment tasks must be carefully synchronized to avoid failures. There are three main types of HSP:

- Cyclic Hoist Scheduling Problem (CHSP): Common in mass production, where the line handles limited product types. Robots follow a fixed sequence of movements (cycle), repeated continuously. The objective is to minimize the cycle time, optimizing productivity by scheduling robot movements and task execution in a predictable pattern [1-2].
- Predictive Hoist Scheduling Problem (PHSP): This non-cyclic approach applies to situations where different products or batches are processed. It focuses on optimizing transitions between different production phases and managing incoming product batches, with the goal of ensuring efficient scheduling without repetitive cycles [3].
- Dynamic Hoist Scheduling Problem (DHSP): In environments with frequent changes or urgent orders, the system needs flexibility. Robots must adapt quickly to handle new priorities, such as urgent batches, while avoiding operational conflicts, like robot collisions. This dynamic mode requires reactive scheduling to manage sudden changes in production [4-5].

Each HSP variant handles different production scenarios, focusing on balancing productivity, quality, and flexibility while adhering to complex scheduling and resource constraints.

This study focuses on the CHSP variant of the HSP, because of its relevance in mass production environments, as it offers a clear framework for analyzing and improving scheduling efficiency, making it particularly suitable for industries aiming to maximize throughput in high-volume production settings.

II. PROBLEM STATEMENT

The HSP has been extensively studied in the literature, garnering considerable attention. This study presents an innovative optimizing approach for an SHCSP. The problem can be described as follows.

Initially, a set of n products is available to be processed on m_c tanks M_1, …, M_{mc}. These tanks contain chemical baths and are arranged, in general, in a row. Each product j (where: j=1,…,n) has to be processed, or soaked n_j times and each soaking operation $O_{i,j}$ (where: i = 1,…, n_j) lasts $p_{i,j}$ time units (where: $p_{i,j} \geq 0$). The time durations are confined by minimum ($a_{i,j}$) and maximum ($b_{i,j}$) durations, as outlined in the processing specifications, and any delay can make the product defective. A single hoist, or automated guided vehicle, has to transport each product from one tank to another according to the processing sequence. The $m_{u,v,j}$ denotes the hoist transport of product j, from tank M_u to tank M_v. Tanks are considered of single capacity, which means that they cannot receive more than one product at the same time. Moreover, waiting, interruption, and storage are not allowed during the soaking process. Therefore, efficiently planning the soaking operations aligns directly with optimizing the schedule for hoist movements (Figure 1).

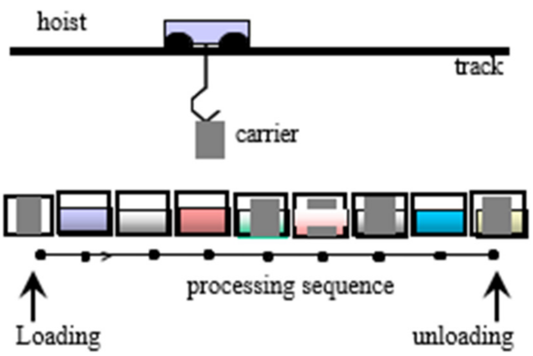

Fig. 1. Line configuration example.

For several years, this problem has been a challenge because it could not be classified into P or NP classes. So, to solve the problem optimally, researchers tried to make it simpler by considering a cyclic schedule. With such schedules, the problem complexity is reduced and the objective is limited to finding a sequence of feasible hoist moves, which can be performed an undefined number of times and until the treatment of all products. The time duration of this sequence is called cycle time (Tmin). If during this period, r products enter the line and r products leave it, the schedule is called r-cyclic (or an r-degree schedule). Authors in [1] have shown that this problem is NP-hard, even with a single-part product.

Figure 2 illustrates two cyclic sequences performed by the handling hoist. It represents the cyclic operation of an electroplating line with a loading and unloading tank, tank 1, and four processing tanks, tanks 2-5. An index is associated with each loaded movement of the robot, and a color code is used to differentiate the transported part-product. Horizontal lines correspond to soaking durations, oblique continuous lines present hoist movements, oblique discontinuous lines present unload hoist movements (without product), and the time required to accomplish a sequence of movements is the cycle.

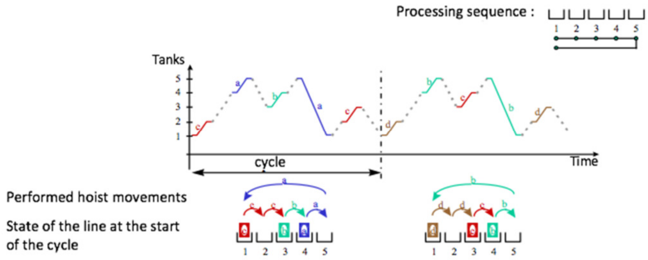

Fig. 2. Cycle time diagram example.

III. CYCLIC SHSP REVIEW

Since the first model was presented [2], the SHCSP has been widely studied [3], and a large number of mathematical models and approaches have been developed [6-12].

In the realm of hoist scheduling, the challenge lies in determining the optimal sequence of hoist operations. However, due to constraints imposed by upper bounds on processing times, not all sequences are viable. Consequently, numerous efforts have been dedicated to devising methods for checking feasibility and calculating the initiation times for each move, with the objective of minimizing the cycle time for a given hoist operation sequence.

Authors in [13] pioneered a search procedure aimed at identifying optimal integer starting times for hoist moves within the basic scheduling problem framework. In [14], the basic scheduling problem was addressed as a parametric critical path problem and was resolved through a modified Bellman-Ford algorithm. In parallel, authors in [15] tackled the same problem, transforming it into cycle time evaluation challenges in bi-valued graphs. They proposed a polynomial algorithm with a time complexity of $O(n^4m^2)$, where n and m denote the number of vertices and arcs in the graph, respectively. Despite its higher worst-case complexity compared to [14], the approach in [15] exhibited superior computational effectiveness. Authors in [16] displayed the equivalence of the multi-degree basic scheduling problem with setup times to a parameter critical path problem, introducing a strongly polynomial algorithm. Focusing on a 2-degree scheduling problem, authors in [17] presented a polynomial algorithm with a complexity of $O(m^8\log(m))$, where m represents the number of tanks. Subsequently, they refined their algorithm, achieving a reduced complexity of $O(m^8)$ for the same problem [18].

For identical part-products, 1-cyclic schedule and line configuration without duplicated and/or associated tanks, the most developed approaches used to solve this problem are MILP models and branch-and-bound algorithms, with both of them sharng the common goal of obtaining an optimal solution. However, the NP-hard nature of the problem poses a challenge, as the computational time for larger instances may become unacceptably long. To address this issue, researchers have explored approximation algorithms as an alternative approach. In contrast to exact algorithms, the primary objective of approximation algorithms is to produce satisfactory or near-optimal solutions within a reasonable time frame. Authors in [19] pioneered a Constraint Logic Programming (CLP) approach, while authors in [20] proposed a max-min algebra model for the problem and introduced heuristics to handle conflicts in tank and hoist usage. The model was further extended to cover scenarios involving multiple hoists and degrees. Authors in [21] designed a Genetic Algorithm (GA) to address the problem, representing sequences of hoist moves as chromosomes. In [22], a Tabu Search (TS) procedure was developed to partition the solution space based on the number of work-in-process parts. Additionally, a repairing procedure was introduced to obtain feasible solutions from infeasible ones.

It has been noted in various studies, [10, 23-25], that schedules involving multiple degrees or cycles can lead to significantly increased productivity, compared to their simpler single-cyclic counterparts in numerous scenarios. Nevertheless, addressing these related issues becomes more intricate, and both modeling and optimization processes become considerably more challenging. In [26], an alternative branch-and-bound approach was explored for addressing the problem. In [27], a Mixed-Integer Linear Programming (MILP) model was formulated, which was extended to scenarios where the hoist is permitted to wait during loaded moves, a consideration also explored in [11, 28]. This model was later broadened in [25], to accommodate configurations, such as load-unloaded buffers, characterized by time-windows at input stations, multi-function tanks, and multi-capacity tanks.

Authors in [29] proposed the first heuristic approach for the one-degree SHCSP problem, which was an adaptation of the heuristic approach to the cyclic problem presented in [30]. The proposed iterative procedure is known as the EST heuristic. The principle of this approach is to schedule products consecutively at their required tanks. It attempts to schedule the hoist loading operations for similar products one after the other. If a constraint is violated, the entire hoist moves are re-scheduled and the introduction date of the product, into the line, is delayed. The proposed solution corresponds to a common period. However, processing times are fixed to the minimum times required. So, a single part-product problem was considered [29].

Besides, the problem complexity is closely dependent on the number of products and, in other words, in the cycle degree. Thus, to bypass problem complexity and get a very good quality multi-degree cyclic schedule, an innovative heuristic approach is proposed in the current study. The particularity of this heuristic is the number of products to be considered on a cycle, which is not defined in advance but which allows to achieve a reasonable and high-quality cycle degree.

IV. PROPOSED METHOD

The algorithm principle of EST heuristic, which is the only heuristic approach dealing with the SHCSP problem, is briefly described in this section before the presentation of the OA-SHCSP algorithm.

In the EST heuristic, hoist operations are scheduled for the first product and then the subsequent ones are performed one at a time. But, if it turns out that a no-wait condition has been violated, the procedure reschedules the entire operation. The EST procedure guarantees the generation of a feasible schedule, without taking flexible processing times into account. To simplify the model, the processing times were fixed to the minimum required duration [29].

The OA-SHCSP aims to take advantage of the flexibility provided by the processing time windows, to reduce the gap between the EST heuristic and optimal schedules. Besides, the proposed heuristic approach schedules the hoist move-in load, which remains the same thing as scheduling the soaking operations. The OA-SHCSP algorithm is:

1. Input parameters: The number of soaking tanks and products m_c and n, respectively. The values of the minimum and maximum soaking durations $a_{i,j}$, and $b_{i,j}$, the list of Minimum Part Sets configurations MPS={MPS_1; MPS_2; …}, and the number (N) of the hoist moves in load to be scheduled for an MPS

2. While (MPS≠{}) do

3. Initialize: Set an MPS configuration MPS_x, the schedule list solution $S=S^0=\{m_{0,1}\}$, the first list of possible following transport operations $L^0=\{m_{1,1}; m_{0,2}\}$, and a counter: $counter_1 \leftarrow 0$

4. Do $counter_1 \leftarrow counter_1+1$
5. Update $L^{counter}{}_1$
6. Sort $L^{counter}{}_1$
7. Select the first hoist move operation
8. Check the feasibility of the selected move operation
9. If (the tank constraints are satisfied)
10. If (time windows are satisfied)
11. Update S
12. Else Compute the Delay and Optimize using a Back Propagation
13. Else backtrack to 6 and select the following move operation
14. Update de schedule list solution
15. While (Card(S)<N)
16. MPS ← $MPS_{\backslash\{MPSx\}}$
17. Back to Step 2
18. Select the best MPS Solution
19. End while

- Input parameter values (Line 1): the number of soaking tanks m_c and products n is very essential to define the hoist moves to be carried out. Indeed, for each product, the hoist has to ensure m_c+1 transport operation: It starts from the load, then passes through the m_c soaking phase, and is finally uploaded. Thus, a hoist move matrix can be produced, where the n lines are defined by products and the (m_c+1) columns by tanks. As a result, the number of hoist moves N/n is implicitly defined by the matrix dimension and the soaking time windows, $a_{i,j}$; $b_{i,j}$.

It is important to note that in loading and unloading stations the minimum required soaking time is zero and the maximum required soaking time is a very large number (+∞).

Considering that the sequence of how products are inserted into the production line is critical and can have a large impact on the objective function, a list of Minimum Part Sets (MPS) configurations MPS= {MPS_1; MPS_2; …} is defined. One of the innovations of the OA-SHCSP heuristic lies in the selection of this list. The number of MPSs that can be generated to produce r types of products is r!, which leads to an increased problem complexity. To address this issue, equivalent MPS are used. For instance, when dealing with three different products A, B, and C, six MPSs can be generated, but only two unique configurations are selected, {A-B-C} and {A-C-B}, thanks to circular permutation. Circular permutation considers certain configurations as equivalent. Thus, {A-B-C} is treated the same as {B-C-A} and {C-A-B}, while {A-C-B} is equivalent to {C-B-A} and {B-A-C}.

Another innovation of the OA-SHCSP heuristic involves repeatedly selecting MPS configurations in a periodic way to ensure a cycle (for example A-B-C-A-B-C-…-A-B-C). Following a transition period, a stable cyclic schedule can be established, taking advantage of resource constraints.

- Initialize (Line 3): After an MPS configuration is selected, a first hoist move operation is added to the solution list. This operation corresponds to the transportation of product 1 from the loading station, tank 0, to the first soaking tank $m_{0,1}$, denoted as (M_0, j_1, $t_{0,1}$) = (0, 1, 0). Here, the first parameter represents the departure tank, the second the product type, and the third the starting time of this hoist move.
- A second loop (Line 4): Establishes a loop to iterate through the algorithm. The heuristic procedure is designed to come to an end when all the transportation movements are selected and incorporated into the solution list S.
- Update procedures (Line 5): Two procedures are defined in this step. The first one is related to the new movement operations to be considered. If the last selected solution is $m_{0,k}$, the new operations to be added to the list are $m_{1,k}$ and $m_{0,k+1}$. However, if $m_{j,k}$ is the selected solution, only $m_{j,k+1}$ will be added. Moreover, if the selected operation is the last operation of a product, no movement operation is added.

  The second procedure is an update procedure. It is related to the hoist move starting time. When a hoist move $m_{k,j}$ is selected, the starting time of the hoist moves of the candidate list L^i, has to be reconsidered. Specifically, for each move $m_{q,p}$ of L^i, its $t_{q,p}$ and can be computed by:

$$t_{q,p} = t_{k,j} + d_{k,k+1} + e_{k+1,q} \qquad (1)$$

  where $t_{k,j}$ is the starting time of the hoist move $m_{k,j}$, and $d_{k,k+1}$ and $e_{k+1,q}$ are the duration of the $m_{k,k+1}$ and $m_{k+1,q}$ move, respectively.

  Additionally, the newly added hoist move $m_{k+1,j}$ to L^i must have a starting time calculated by:

$$t_{k+1,j} = t_{k,j} + d_{k,k+1} + a_{k+1,j} \qquad (2)$$

  where $a_{k+1,j}$ is the minimum required soaking duration.
- Sort procedure (Line 6): Aims to sort operations according to their starting time. This procedure allows the early insertion of products into the line, enabling the simultaneous processing of multiple operations reducing the cycle time.
- Selection Step (Line 7): Reflects the Earliest Starting Time procedure which defines the way to select the hoist movements.
- Feasibility checking (Line 8-Line 9): Two conditions have to be satisfied to assert that the partial schedule is feasible. The first one is related to tank availability. To perform a hoist move, the destination tank has to be available. The second one concerns the soaking time windows and more precisely the maximum soaking duration, where for each selected hoist move $m_{k,j}$ from the partial solution sequence, the inequality (3) has to be satisfied:

$$t_{k,j} - t_{k-1,j} - d_{k-1,k} \le b_{k,j} \qquad (3)$$

  where $b_{k,j}$ is the maximum required soaking duration.
- Optimizing procedure (Lines 12-13): If the tank availability constraint is not respected for the first hoist move, then, the following move candidate of the list is considered. However, if the move candidates of the list L^i are considered and no solution is possible, backtrack is allowed to consider another solution of the L^{i-1}.

  If the soaking maximum duration is exceeded, a delay parameter $\delta_{k,j}$ is calculated by:

$$\delta_{k,j} = \left(t_{k,j} - t_{k-1,j} - d_{k-1,k}\right) - b_{k,j} \qquad (4)$$

  Then, backtracking is allowed as previously. But, if its previous hoist movement $m_{k-1,j}$ is completed, the starting time is delayed by $\delta_{k,j}$, to benefit from the bounded soaking times. After that, a new sort of list is considered and the resolution procedure steps are considered again.

  The procedure steps come to an end when all considered hoist moves (N)/n of the matrix are scheduled, while the OA-SHCSP heuristic stops when all the MPSs configurations are considered.

V. COMPUTATIONAL RESULTS AND ANALYSIS

To evaluate the performance of the OA-SHCSP heuristic, tests were conducted, which was/were implemented and solved using the C++ software on an HP computer with a 2.13 GHz processor and 8 GB of memory.

It is important to notice that there are few problem instances available in the literature. Therefore, inspired by previous works, an algorithm was developed to create a dataset for different line sizes. The minimum time durations for each soaking operation are generated using a uniform distribution.

The maximum time duration for each soaking operation is then produced according to three types of Time Windows (TW): Close Windows (C), Half-opened Windows (H), and Open Windows (O). The hoist movements without load between consecutive baths are generated using a uniform distribution. Based on the load value of the hoist move between consecutive baths, the instances are classified into three Hoist Speed (HS) categories: Fast Hoist (FH), Half-fast Hoist (HH), and Slow Hoist (SH).

The results, as reported in Table I, pertain to each pair of time windows and hoist speed. For each test series, the results include the cycle degree, meaning the number of different products to be produced during a cycle, T_{min}, constituting the cycle time for the considered r-degree cycle, and $T_{average}$, which is the average cycle time calculated as T_{min}/r.

Also, a cycle time reduction percentage $R_1(\%)$ is presented. It defines the percentage of absolute deviation between the average cycle time of the OA-SHCSP heuristic and the average cycle time of the EST heuristic, and is calculated by:

$$R_1(\%) = \frac{T^1_{average} - T^2_{average}}{T^1_{average}} \quad (5)$$

where $T^1_{average}$ and $T^2_{average}$ represnt the average cycle time for EST and OA-SHCSP, respectively.

A comparison between the EST heuristic and the OA-SHCSP reveals that OA-SHCSP consistently outperforms EST in terms of average cycle time across all generated datasets. The reduction in average cycle time ranges from 28.73% to 60.29%, highlighting the significant improvement and high performance of the proposed algorithm. Additionally, the CPU time remains nearly identical for both heuristics, staying under 1 second for all considered instances.

It is also noteworthy that the study approach enables to find high-degree cyclic schedules (up to 60 cycles) within a few seconds, which is a remarkable improvement over the exact approach proposed in [23]. The last simulations had to be stopped after a four hour duration, as the system was unable to find an optimal solution for cycles with more than 10 degrees, demonstrating the superior scalability and efficiency of OA-SHCSP in handling larger and more complex scheduling problems.

The influence of the time window width on the cycle time reduction percentages between OA-SHCSP heuristic and the EST heuristic was examined and the results reveal that OA-SHCSP heuristic yields better outcomes, particularly in the presence of wide time windows. This observation is attributed to the enhanced flexibility offered by such time windows in processing lines when multiple products are simultaneously handled during a cycle. In this scenario, the hoist benefits from more time to introduce new products or perform multiple transport operations while processing the existing ones. This contradicts with the EST heuristic, which lacks a strategy to capitalize on this flexibility (Figure 3).

Moreover, the current study investigates how the cycle time reduction percentages, when comparing OA-SHCSP heuristic with the EST, are affected by variations in hoist speed. In all conducted simulations, superior performance was achieved compared to the EST heuristic. Specifically, an average cycle time reduction percentage of 46.89% was observed for an SH, 44.43% for an HH, and 42.35% for an FH (Figure 4). These findings highlight the critical role of selecting the appropriate hoist speed to achieve significant cyclic degrees. This choice must be well-defined to enhance line production while ensuring optimal line functionality and high-quality products.

TABLE I. SIMULATION RESULTS

#	TW	HS	#T	r	EST heuristic [29]		OA-SHCSP Our heuristic		R1 (%)
					T^1_{min}	$T^1_{average}$	T^2_{min}	$T^2_{average}$	
1	C/H	SH	5	60	22200	370.00	14670	244.50	33.92
2	C/O	HH	5	60	20160	336.50	11100	185.00	45.02
3	C/H	FH	5	60	22980	383.50	11310	188.50	50.84
4	C/O	SH	5	60	23820	397.00	11010	184.00	53.65
5	C/H	HH	5	60	26940	449.00	17280	288.00	35.85
6	C/O	FH	5	60	24780	413.00	11490	191.50	53.63
7	C/H	SH	6	60	22800	380.00	10980	183.00	51.84
8	C/O	HH	6	60	22380	373.00	10500	175.00	53.08
9	C/H	FH	6	60	30990	516.50	19860	331.00	35.91
10	C/O	SH	6	60	28380	473.00	17490	291.50	38.37
11	C/H	HH	6	60	25110	418.50	15300	255.00	39.06
12	C/O	FH	6	60	30540	509.00	18960	316.00	37.92
13	C/H	SH	7	60	33450	557.50	18680	311.33	44.16
14	C/O	HH	7	60	32190	536.50	12780	213.00	60.29
15	C/H	FH	7	60	31200	520.00	19050	317.50	38.94
16	C/O	SH	7	60	30540	509.00	13560	226.00	55.64
17	C/H	HH	7	60	39420	657.00	23520	392.00	40.33
18	C/O	FH	7	60	39060	651.00	27840	464.00	28.73
19	C/H	SH	8	60	32310	538.50	16940	282.33	47.57
20	C/O	HH	8	60	35220	587.00	16848	280.80	52.16
21	C/H	FH	8	60	33210	553.50	16542	275.70	50.19
22	C/O	SH	8	60	34440	574.00	14325	238.75	58.41
23	C/H	HH	8	60	43110	718.50	27600	461.00	35.84
24	C/O	FH	8	60	44250	737.50	29160	486.00	34.10
25	C/H	SH	9	60	35820	597.00	20760	346.00	42.04
26	C/O	HH	9	60	40320	672.00	23190	386.50	42.48
27	C/H	FH	9	60	39240	654.00	18510	308.50	52.83
28	C/O	SH	9	60	42960	716.00	27870	464.50	35.13
29	C/H	HH	9	60	50070	834.50	31680	528.00	36.73
30	C/O	FH	9	60	48360	806.00	29130	485.50	39.76
31	C/H	SH	10	60	39720	662.00	18840	314.00	52.57
32	C/O	HH	10	60	39000	650.00	17595	293.25	54.88
33	C/H	FH	10	60	47070	784.50	25850	430.83	45.08
34	C/O	SH	10	60	42300	705.00	21280	354.67	49.69
35	C/H	HH	10	60	48210	803.50	30150	502.50	37.46
36	C/O	FH	10	60	45990	766.50	32790	546.50	40.24

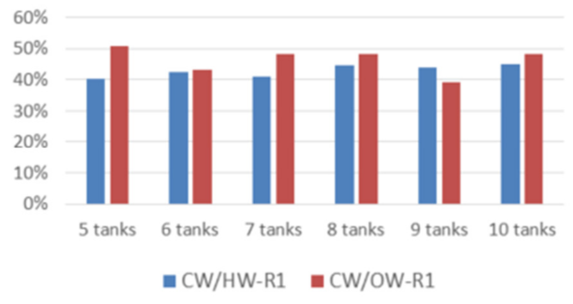

Fig. 3. Impact of time windows on cycle time reductions percentages.

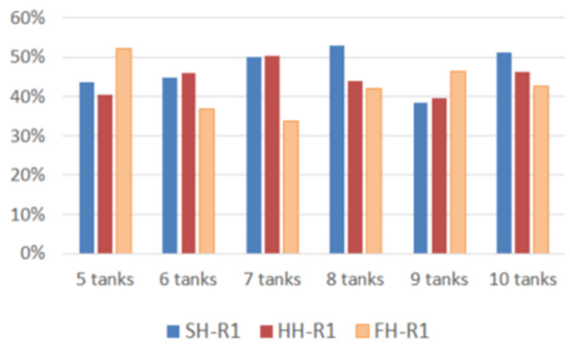

Fig. 4. Impact of hoist speed on cycle time reductions percentages.

VI. CONCLUSION

In this research, an innovative heuristic optimization approach was introduced, called Optimization Approach-Single Hoist Cyclic Scheduling Problem (OA-SHCSP), specifically designed to address the Single Hoist Scheduling Problem (SHSP) with the goal of minimizing cycle time. The experimental analysis demonstrated that the OA-SHCSP heuristic consistently outperforms the established Earliest Starting Time (EST) heuristic across all generated instances. Additionally, this approach efficiently identifies r-degree cyclic schedules, crucial for optimizing operations in mass production environments, within a reasonable simulation time. The results clearly stress the efficiency of the method in solving cyclic scheduling problems, offering a powerful tool for industries where maximizing throughput is critical.

The significance of this study lies in its direct applicability to industries engaged in high-volume production settings, such as surface treatment lines, where scheduling efficiency directly impacts productivity. Focusing on the Cyclic Hoist Scheduling Problem (CHSP), provides a structured approach to optimizing repetitive processes, which is essential for companies aiming to enhance operational efficiency.

The contribution to the community is twofold. Initially, a new heuristic that improves upon existing methods in terms of performance and cycle time reduction, is been offered. Secondly, insights into handling more complex scheduling issues, such as higher cycle degrees and heterogeneous product management, are provided. New perspectives emerge for further exploration, particularly in extending the OA-SHCSP approach to multi-hoist environments and incorporating more dynamic, real-time scheduling challenges faced by modern industries. The novelty of the current approach lies in its practical optimization potential and its ability to be adapted for a wide range of production scenarios.

REFERENCES

[1] L. Lei and T. J. Wang, "A proof: the cyclic hoist scheduling problem is NP-complete," Working Paper, Graduate School of Management, Rutgers University, 1989.

[2] L. W. Phillips and P. S. Unger, "Mathematical Programming Solution of a Hoist Scheduling Program," *A I I E Transactions*, vol. 8, no. 2, pp. 219–225, Jun. 1976, https://doi.org/10.1080/05695557608975070.

[3] M. A. Manier and C. Bloch, "A Classification for Hoist Scheduling Problems," *International Journal of Flexible Manufacturing Systems*, vol. 15, no. 1, pp. 37–55, Jan. 2003, https://doi.org/10.1023/A:1023952906934.

[4] A. E. Amraoui and M. Elhafsi, "An efficient new heuristic for the hoist scheduling problem," *Computers & Operations Research*, vol. 67, pp. 184–192, Mar. 2016, https://doi.org/10.1016/j.cor.2015.10.006.

[5] D. Ramin, D. Fraizzoli, A. Ballarino, and A. Brusaferri, "Dynamic hoist scheduling for multi-recipe and multi-stage production lines: A logical framework," *Computers & Industrial Engineering*, vol. 182, Aug. 2023, Art. no. 109360, https://doi.org/10.1016/j.cie.2023.109360.

[6] J. Feng, A. Che, C. Chu, E. Levner, and V. Kats, "Scheduling robotic cells with fixed processing times or time windows: Classification, solution approaches, polynomial algorithms and complexity," *European Journal of Operational Research*, vol. 319, no. 2, pp. 468–483, Dec. 2024, https://doi.org/10.1016/j.ejor.2024.01.041.

[7] A. Hosseini, A. Otto, and E. Pesch, "Scheduling in manufacturing with transportation: Classification and solution techniques," *European Journal of Operational Research*, vol. 315, no. 3, pp. 821–843, Jun. 2024, https://doi.org/10.1016/j.ejor.2023.10.013.

[8] A. El Amraoui and M. Benrejeb, "SHCSP Heuristic for a Single Hoist Scheduling Problem," in *2023 IEEE International Conference on Advanced Systems and Emergent Technologies (IC_ASET)*, Apr. 2023, pp. 01–05, https://doi.org/10.1109/IC_ASET58101.2023.10150572.

[9] H. J. Kim and J. H. Lee, "Look-ahead based reinforcement learning for robotic flow shop scheduling," *Journal of Manufacturing Systems*, vol. 68, pp. 160–175, Jun. 2023, https://doi.org/10.1016/j.jmsy.2023.02.002.

[10] A. El Amraoui, "Combinatorial optimization approaches for multi-part cyclic hoist scheduling problem," *4OR*, vol. 13, no. 1, pp. 111–112, Mar. 2015, https://doi.org/10.1007/s10288-014-0263-6.

[11] A. Che and C. Chu, "Cyclic hoist scheduling in large real-life electroplating lines," *OR Spectrum*, vol. 29, no. 3, pp. 445–470, Jul. 2007, https://doi.org/10.1007/s00291-006-0040-9.

[12] H. Alghamdi, T. Alsubait, H. Alhakami, and A. Baz, "A Review of Optimization Algorithms for University Timetable Scheduling," *Engineering, Technology & Applied Science Research*, vol. 10, no. 6, pp. 6410–6417, Dec. 2020, https://doi.org/10.48084/etasr.3832.

[13] L. Lei, "Determining the optimal starting times in a cyclic schedule with a given route," *Computers & Operations Research*, vol. 20, no. 8, pp. 807–816, Oct. 1993, https://doi.org/10.1016/0305-0548(93)90102-O.

[14] E. Levner and V. Kats, "A parametric critical path problem and an application for cyclic scheduling," *Discrete Applied Mathematics*, vol. 87, no. 1, pp. 149–158, Oct. 1998, https://doi.org/10.1016/S0166-218X(98)00054-7.

[15] H. Chen, C. Chu, and J. M. Proth, "Cyclic scheduling of a hoist with time window constraints," *IEEE Transactions on Robotics and Automation*, vol. 14, no. 1, pp. 144–152, Oct. 1998, https://doi.org/10.1109/70.660860.

[16] V. Kats, L. Lei, and E. Levner, "Minimizing the cycle time of multiple-product processing networks with a fixed operation sequence, setups, and time-window constraints," *European Journal of Operational Research*, vol. 187, no. 3, pp. 1196–1211, Jun. 2008, https://doi.org/10.1016/j.ejor.2006.07.030.

[17] V. Kats and E. Levner, "Parametric algorithms for 2-cyclic robot scheduling with interval processing times," *Journal of Scheduling*, vol. 14, no. 3, pp. 267–279, Jun. 2011, https://doi.org/10.1007/s10951-010-0166-0.

[18] V. Kats and E. Levner, "A faster algorithm for 2-cyclic robotic scheduling with a fixed robot route and interval processing times," *European Journal of Operational Research*, vol. 209, no. 1, pp. 51–56, Feb. 2011, https://doi.org/10.1016/j.ejor.2010.10.002.

[19] P. Baptiste, B. Legeard, M. A. Manier, and C. Varnier, "Optimization with constraint logic programming: the hoist scheduling problem solved with various solvers," *WIT Transactions on Information and Communication Technologies*, vol. 1, pp. 599–614, 1993.

[20] P. Spacek, M. A. Manier, and A. E. Moudni, "Control of an electroplating line in the max and min algebras," *International Journal of Systems Science*, vol. 30, no. 7, pp. 759–778, Jan. 1999, https://doi.org/10.1080/002077299292065.

[21] J. M. Lim, "A genetic algorithm for a single hoist scheduling in the printed-circuit-board electroplating line," *Computers & Industrial Engineering*, vol. 33, no. 3, pp. 789–792, Dec. 1997, https://doi.org/10.1016/S0360-8352(97)00254-4.

[22] P. Yan, A. Che, N. Yang, and C. Chu, "A tabu search algorithm with solution space partition and repairing procedure for cyclic robotic cell scheduling problem," *International Journal of Production Research*, vol. 50, no. 22, pp. 6403–6418, Nov. 2012, https://doi.org/10.1080/00207543.2011.645953.

[23] A. El Amraoui, M. A. Manier, A. El Moudni, and M. Benrejeb, "A linear optimization approach to the heterogeneous *r*-cyclic hoist scheduling problem," *Computers & Industrial Engineering*, vol. 65, no. 3, pp. 360–369, Jul. 2013, https://doi.org/10.1016/j.cie.2013.03.007.

[24] A. El Amraoui, M. A. Manier, A. E. Moudni, and M. Benrejeb, "A genetic algorithm approach for a single hoist scheduling problem with time windows constraints," *Engineering Applications of Artificial Intelligence*, vol. 26, no. 7, pp. 1761–1771, Aug. 2013, https://doi.org/10.1016/j.engappai.2013.02.004.

[25] A. Amraoui, M. A. Manier, E. Abdellah, and M. Benrejeb, "Resolution of the Two-Part Cyclic Hoist Scheduling Problem with bounded Processing Times in Complex Lines' Configuration," *European J of Industrial Engineering*, vol. 6, pp. 454–473, Jul. 2012, https://doi.org/10.1504/EJIE.2012.047661.

[26] M. Mateo and R. Companys, "Hoist Scheduling in a chemical line to produce batches with identical sizes of different products," in *6th Francophone Conference on Modeling and SIMulation*, Apr. 2006.

[27] A. Amraoui, M. A. Manier, A. Moudni, and M. Benrejeb, "A Mixed Linear Program for a Multi-Part Cyclic Hoist Scheduling Problem," *IJ-STA*, vol. 2, pp. 612–623, Dec. 2008.

[28] W. C. Ng, "A branch and bound algorithm for hoist scheduling of a circuit board production line," *International Journal of Flexible Manufacturing Systems*, vol. 8, no. 1, pp. 45–65, Jan. 1996, https://doi.org/10.1007/BF00167800.

[29] W. Song, Z. B. Zabinsky, and R. L. Storch, "An algorithm for scheduling a chemical processing tank line," *Production Planning & Control*, vol. 4, no. 4, pp. 323–332, Jan. 1993, https://doi.org/10.1080/09537289308919454.

[30] Y. Yih, "An algorithm for hoist scheduling problems," *International Journal of Production Research*, vol. 32, no. 3, pp. 501–516, Mar. 1994, https://doi.org/10.1080/00207549408956949.