engineering proceedings

MDPI

Proceeding Paper

A MILP Model and Heuristic Approach for Optimizing Cutting Sequences to Minimize Work-in-Progress Inventory in Multi-Product Manufacturing †

Miklós Szabó [1,*], Zoltán András Nagy [1], Tamás Hajba [2] and Tamás Hartványi [1]

[1] Department of Logistics and Forwarding, Faculty of Architecture, Civil Engineering and Transport Sciences, Széchenyi István University, 9026 Gyor, Hungary; nagyz@sze.hu (Z.A.N.); hartvanyi@sze.hu (T.H.)
[2] Department of Mathematics and Physics, Faculty of Informatics and Electrical Engineering, Széchenyi István University, 9026 Gyor, Hungary; hajbat@sze.hu
* Correspondence: szabo.miklos@sze.hu; Tel.: +36-30-346-7710
† Presented at the Sustainable Mobility and Transportation Symposium 2026, Győr, Hungary, 8 October 2026.

Abstract

This study addresses an optimization problem arising in multi-product manufacturing, where components are produced by cutting raw material bars. Combining components of several products on the same bar can reduce material waste, but it also creates a sequencing problem: components of different products must be stored separately until all parts of the corresponding product are completed. These unfinished product groups form work-in-progress (WIP) inventory, whose level depends on the cutting sequence of the bars. The objective is therefore to determine a bar-processing sequence that minimizes WIP inventory. Although sequencing problems are widely studied, this specific industrial variant has received limited attention. Since the number of feasible sequences grows rapidly even for small instances, exhaustive enumeration is impractical. To address the problem, an integer linear programming model and several heuristic methods, including a graph-theoretical approach, are developed, and their effectiveness is demonstrated through test examples.

Keywords: space utilization; production scheduling; pathfinding; heuristics

1. Introduction

One of the fundamental challenges in manufacturing processes within the machinery industry is the efficient utilization of available resources, particularly in terms of raw materials and production space. In cases where the initial raw material is long bar stock, which is cut into shorter pieces by CNC machines, lathes or cutting equipment, the optimization of production sequencing and space utilization plays a particularly critical role. The goal of production in the currently examined manufacturing environment is twofold. The primary objective is to achieve the best possible alignment between the manufactured units and the raw materials, thereby minimizing production waste. In this context, it means maximizing the utilization of the bar stock—in other words, grouping parts in such a way that minimizes the leftover material at the end of the operation. However, this article considers this optimization step as given and treats it as an input parameter for further investigations.

The focus of this study is a secondary, yet practically significant issue: how to feed the bar stock used as raw material into the production machine in a sequence that minimizes the spatial footprint of the resulting semi-finished products before the cutting process. The primary optimization concerns planning the cutting of parts—which are components of

Check for updates

Academic Editors: Gábor Dogossy, Szabolcs Kocsis-Szürke, Dániel Csikor, Dániel Serfőző, Miklós Kuczmann, András Lajos Nagy, László Lendvai and Ambrus Zelei

Published: 12 November 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

various final products—from a single bar stock. To avoid post-process sorting, all components belonging to each final product are stored separately next to the production line until the entire set is completed. By altering the production sequence of the bar stock, the amount of storage space required during the workflow can, in most cases, be influenced. The aim of this article is to determine this sequence, present the heuristic procedures related to this determination, and validate the described models using a MILP (Mixed-Integer Linear Programming) model. The main novelty of the study is that the sequencing of already optimized cutting patterns is treated as a separate WIP-space minimization problem. While classical cutting stock approaches primarily focus on material utilization, the proposed approach evaluates the production sequence according to the temporary storage space required by simultaneously open batches. In this way, the study extends the cutting-stock-based planning logic with a practically relevant WIP-space criterion.

2. Problem Description Through an Industrial Case Study

As mentioned in the introduction, the task begins after the production planning phase aimed at minimizing material waste. As a result of this preceding procedure, a set of rules is available that specifies which semi-finished parts are to be cut from each raw material item. Furthermore, it is also predetermined which group of semi-finished parts constitutes a final product. To refer to this group, the term Batch is introduced. For identifying the raw materials (bar stocks), the references T01, T02, ... Tn are introduced, and for identifying the batches, K01, K02, ... Km are used. Based on this, a mutual assignment rule can be established, indicating which batches are found on each bar stock and from which bar stock each batch is manufactured. Table 1 presents the assignment rule for an arbitrary task, which is also illustrated in parts of Figure 1.

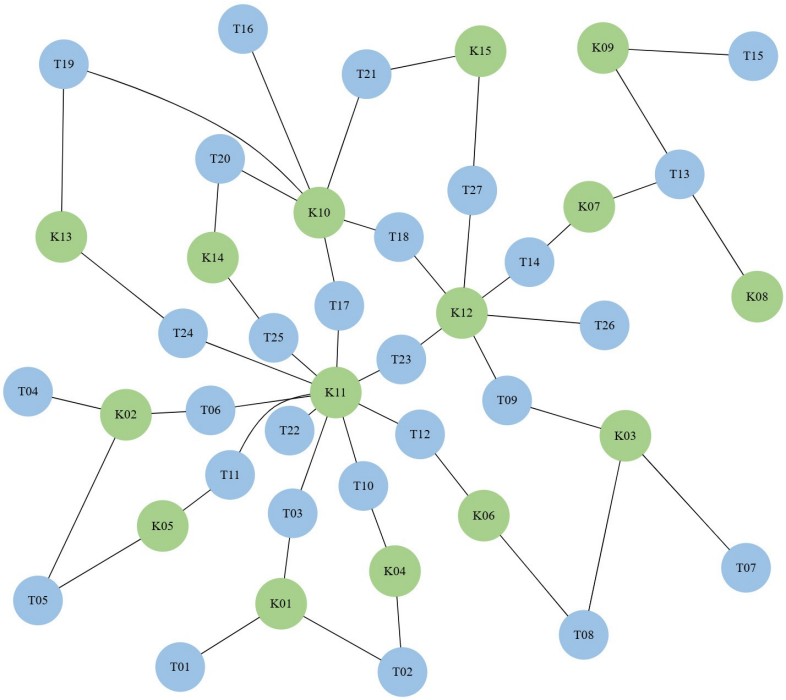


Figure 1. Connection Network of Tn Bar Stocks and Km Batches.

The task is influenced to some extent by the fact that the cut parts occupy different amounts of space on the production floor due to their varying sizes. However, since the parts within a given batch are very similar in size, a predefined space requirement can be assigned to each batch, which is considered during the optimization process.

https://doi.org/10.3390/engproc2026157065

Table 1. Connection Matrix of Tn Bar Stocks and Km Batches.

	K01	K02	K03	K04	K05	K06	K07	K08	K09	K10	K11	K12	K13	K14	K15
T01	1														
T02	1			1											
T03	1										1				
T04		1													
T05		1			1										
T06		1									1				
T07			1												
T08			1			1									
T09			1									1			
T10				1							1				
T11					1						1				
T12						1					1				
T13							1	1	1						
T14							1					1			
T15									1						
T16										1					
T17										1	1				
T18										1		1			
T19										1			1		
T20										1				1	
T21										1					1
T22											1				
T23											1	1			
T24											1		1		
T25											1			1	
T26												1			
T27												1			1

Note: The table shows which product batches are represented in each bar-stock type.

The goal is to determine a sequence for processing the bar stocks T01, T02, ... Tn such that the total space occupied by the batches currently in progress is minimized. A batch K is considered to be in progress if at least one of its parts has already been cut, but not all of them have been completed yet.

For calculating the WIP state, a distinction is made between the start and the end of processing a bar stock. New batches appearing on a bar are opened at the start of processing, while batches completed by that bar are closed only after the bar has been processed. For example, if T01 opens K01, T02 opens K04, and T03 both opens K11 and completes K01, then K11 is first opened at the start of T03, and K01 is closed only at the end of T03 as presented in Table 2. This convention ensures that the temporary maximum WIP requirement during processing is not underestimated.

A sequence of bar stocks to be processed is given, from which semi-finished products are generated through cutting. These semi-finished products are assigned to final products, meaning that for every semi-finished part, it is known which final product it will be built into. During the cutting process, the semi-finished parts are temporarily stored after being manufactured. If at least one component of a final product is completed, that product

occupies a dedicated storage position, which is maintained until all its components have been manufactured.

Table 2. Illustration of maximum open batches during bar-stock processing.

Event	Bar Stock	Batch Event	Maximum Open Batches During Processing
1.	T01	Opens K01	K01
2.	T02	Opens K04	K01, K04
3.	T03	Opens K11, then closes K01	K01, K04, K11

The objective is to determine a processing order for the bar stocks that minimizes the number of storage positions occupied simultaneously throughout the entire production process.

3. Literature Review

The problem is closely related to several areas of research. The Cutting Stock Problem (CSP) has been studied for decades, beginning with the pioneering work of Gilmore and Gomory, who applied linear programming methods to minimize material waste [1,2]. This field was later expanded with heuristic and metaheuristic approaches [3,4], as well as with models incorporating various uncertainty factors [5–7].

Although the present study does not focus on material waste, but rather on minimizing production sequencing and temporary storage requirements, the CSP literature provides a strong mathematical and algorithmic foundation. The problem is also related to production sequencing, but it is not treated here as a classical flow-shop or job-shop scheduling problem. In the examined industrial setting, the cutting patterns are assumed to be already determined, and the main decision concerns the order in which the bar-stock types are processed on the cutting equipment. Thus, the production process considered in the model consists of one dominant processing step, namely the cutting of the raw material bars. The objective is not the minimization of makespan, machine idle time, or job completion time, but the minimization of the maximum WIP-related storage space generated by simultaneously open batches.

Consequently, the investigated problem is closer to a single-machine pattern-sequencing problem with a weighted open-stack objective than to a conventional flow-shop or job-shop formulation. Flow-shop and job-shop scheduling models [8–11] are therefore considered only as broader methodological references for production sequencing, while the specific contribution of this study lies in the WIP-space-based sequencing of already generated cutting patterns.

The literature, therefore, offers a wide range of approaches that can serve as methodological inspiration for the sequencing optimization investigated in this study. Comparing different problems allows for the development of existing models toward a specific objective—namely, minimizing spatial occupation determined by production order.

4. Materials and Methods

Five solutions were compared for the presented problem in six case studies.

4.1. MILP Model

The following notations will be used in the mathematical model of the problem:

N: number of the bars;

M: number of the batches;

$S(k)$: set of the bars that contain an element of the k-th batch.

Parameters:

$l(k)$: the length of the k-th batch.

Binary variables:

$Z(i,j)$: equals 1 if the j-th bar is placed to i-th place of the sequence;

$Y(i,k)$: equals 1 if the k-th batch is opened after finishing the i-th bar of the sequence.

Variable:

T: the maximum storage space needed for the open batches during the production sequence.

For a given processing sequence, let $\pi = (\pi 1, \pi 2, ..., \pi N)$ denote the order of the bars, where πi is the bar processed in the i-th position. The objective value T is calculated as the maximum total storage-space requirement of the batches that are open at any position of the sequence:

$$T(\pi) = \max_i \left(\sum_{k=1}^{K} Y(i,k) \times l(k) \right)$$

where

$Y(i,k) = 1$ if batch k is open at position i; otherwise, $Y(i,k) = 0$.

Thus, the objective is to find a feasible processing sequence π for which $T(\pi)$ is minimal.

Our mixed integer linear programming model contains the following constraints.

$$\sum_{i=1}^{N} Z(i,j) = 1 \quad 1 \le j \le N \tag{1}$$

$$\sum_{j=1}^{N} Z(i,j) = 1 \quad 1 \le i \le N \tag{2}$$

$$\sum_{j \in S(k)} Z(i,j) \le Y(i,k) \quad 1 \le i \le N \quad 1 \le k \le K \tag{3}$$

$$\sum_{l=1}^{i} Z(l,j) + \sum_{l=i}^{N} Z(l,m) \le 1 + Y(i,k) \quad 1 \le i \le N \quad 1 \le k \le K \quad j,m \in S(k) \tag{4}$$

$$\sum_{k=1}^{K} Y(i,k) \times l(k) \le T \quad 1 \le i \le N \quad 1 \le k \le K \tag{5}$$

$$T \to min \tag{6}$$

Equation (1) ensures that each bar is placed at exactly one position of the sequence. Constraint (2) implies that for each position of the sequence, exactly one bar is placed at that position. Constraints (3) and (4) state that the k-th batch is opened after finishing the i-th bar of the sequence if either the i-th bar of the sequence contains an element of the k-th batch, or there are bars j and m, both containing an element of the k-th batch such that the order of j in the sequence is strictly less than i and the order of m in the sequence is strictly greater than i. Equality (5) ensures that for each index i, the minimum space needed for the batches is at least the space needed for the batches after finishing the i-th bar of the order. Finally, (6) states that the space needed for the batches has to be minimized.

The formulations of the MILP model were written in GAMS 24.1.3 modelling language and solved using CPLEX 12.3 on an Intel Xeon E31225 3.1 GHz personal computer equipped with 4 GB. The CPLEX options applied were mixed integer programming, a time limit of 1 h, and deterministic parallel mode with 2 threads.

4.2. Heuristics

4.2.1. Starting Sequence

The production planning process assigns the parts to be manufactured to specific bar stocks. The list resulting from this assignment process was categorized into bar stock types, where each bar stock within a given type contains parts from the same batches. Based on the batch numbers associated with each type, the bar stock types were ordered and

labelled (T01, T02, ...). This sequence is preserved and passed on to the production floor, where we analyse how much space is required next to the manufacturing equipment under this arrangement. In the examined case studies, the starting sequence corresponds to the production order generated by the CutStop V2.0 software, which is owned by Globalsteel Kft. and available through the company's website, globalsteel.hu. This software-generated order represents the baseline sequence used in practice. Therefore, it provides a suitable reference point for evaluating how much the proposed sequencing methods can reduce the required WIP-related storage area compared with the original production plan.

4.2.2. Graph-Based Approach

There is a clear assignment relationship between the bar stock types—or simply the bar stocks—and the batches, as shown in the bipartite graph in Figure 1. As an initial heuristic, we define the distance between any two elements as the length of the shortest path between their corresponding vertices in the graph. This distance is computed as the minimum number of edges that must be traversed to move from one vertex to another, and can be efficiently obtained using Dijkstra's algorithm because edge weights are uniform and non-negative. Based on this distance metric, a natural strategy is to anchor the production sequence at the two endpoints of the graph's longest simple path, as these vertices represent the most structurally distant elements.

The remaining elements are then ordered according to their computed distances from these endpoints. Each vertex is assigned a distance value derived from the shortest-path computation, and the final sequence is obtained by sorting the vertices in either ascending or descending order with respect to this measure. This procedure constitutes our second heuristic. A detailed evaluation of its performance is provided in the Results section.

4.2.3. A New Greedy Algorithm

The fourth heuristic represents the main algorithmic contribution of the study. In contrast to the starting sequence and the graph-based approach, this heuristic evaluates each candidate raw material type according to its direct WIP-space effect, including newly opened batches, completed batches, and the amount of storage space released after completion. Therefore, the selection rule is based not only on structural relationships between bar stocks and batches, but also on the expected change in temporary storage demand. The steps of this heuristic are as follows:

1. Identify the types of raw materials (within one raw material type, all items contain components for the same set of batches).
2. Establish the connection matrix between batches and raw material types.
3. Know the size category of each batch (this determines how much physical space it occupies next to the machine).
4. During scheduling, consider the following:
    - Which batches are completed;
    - Which batches are currently open;
    - How many additional raw material types are still required to complete an open batch;
    - How much new space must be allocated when a given raw material type is introduced;
    - How much space can be freed by completing batches.

    The raw material type to be selected next should be the one with the lowest space demand per unit of production progress, treated as a cost metric.
    The progress value is defined by taking the length category of the batches associated with a given raw material type, dividing it by the number of raw material types linked to that batch, and summing these values.

https://doi.org/10.3390/engproc2026157065

The space requirement is calculated as the difference between newly needed space and the space to be freed. If this result is not positive (i.e., at least as much space is freed as is newly needed), then the raw material type should be prioritized. However, among other non-positive values, there is no need to differentiate further.

5. The selection of raw material types should continue until all are scheduled.

During the execution of this heuristic, we often encounter decision points where the algorithm sees no difference between the options—it evaluates multiple choices as equally favourable. In the cases we examined, these branch points had no impact on the final result.

In terms of performance, this heuristic closely approximated the MILP results for low element counts (often producing optimal solutions). For higher element counts, it ran significantly faster and outperformed previous methods by an average of 40%. Unfortunately, the optimal solution is not known for large-scale problems, so we cannot determine how close this heuristic gets to the theoretical optimum or how much further optimization is possible.

The results of the solutions for the six case studies are presented in Table 3.

Table 3. Results of examinations in required space size [m].

Case Study	Number of Bars (Tn)	Number of Batches (Km)	MILP	Starting Sequence	Graph-Based Approach	New Greedy Algorithm
GY0451	12	12	24	48	30	24
GY1557	15	11	24	60	48	24
GY3859	122	51	-	146	131	62
GY3863	164	51	-	138	111	93
GY6171	34	38	25	55	38	30
GY8045	27	15	30	54	72	30

5. Conclusions

In this paper, a problem is examined that builds upon the output of classical cutting optimization tasks, but focuses instead on the impact of production sequencing on minimizing the space occupied by temporarily stored semi-finished products. The underlying assumption is that the sequence of already optimized cutting patterns significantly affects how much space is occupied by completed, yet undelivered parts during production.

The problem is formally defined, and various methods are proposed for determining an optimal or near-optimal sequence of cutting patterns. The proposed model takes into account the removal conditions of parts and the storage constraints that arise during the production process. Experimental results show that appropriate sequencing can significantly reduce temporary storage needs, thereby improving the overall efficiency of the manufacturing process.

Based on the research findings, it can be concluded that optimizing the execution order of cutting patterns is an important yet previously underappreciated aspect of production planning. In the future, different variations of the problem will be further investigated. It would be valuable to determine whether certain properties of the cut parts can help predict the expected storage space in advance. Another objective is to integrate primary optimization goals, such as cutting waste reduction, and secondary optimization goals, such as semi-finished product storage. This could be achieved by formulating a combined objective function in which, by adjusting certain parameters and even allowing a slight increase in cutting waste, a global optimum can be targeted.

https://doi.org/10.3390/engproc2026157065

Author Contributions: Conceptualization, M.S., Z.A.N., T.H. (Tamás Hajba) and T.H. (Tamás Hartványi); methodology, M.S., Z.A.N. and T.H. (Tamás Hajba); MILP model development, T.H. (Tamás Hajba); heuristic development, M.S. and Z.A.N.; case study preparation and industrial problem definition, M.S. and T.H. (Tamás Hartványi); literature review, Z.A.N.; validation, M.S., Z.A.N., T.H. (Tamás Hajba) and T.H. (Tamás Hartványi); formal analysis, M.S., Z.A.N. and T.H. (Tamás Hajba); investigation, M.S. and T.H. (Tamás Hartványi); writing—original draft preparation, M.S. and Z.A.N.; writing—review and editing, Z.A.N., M.S., T.H. (Tamás Hajba) and T.H. (Tamás Hartványi); visualization, M.S. and Z.A.N.; supervision, T.H. (Tamás Hartványi); project administration, Z.A.N. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Gilmore, P.C.; Gomory, R.E. A Linear Programming Approach to the Cutting-Stock Problem. *Oper. Res.* **1961**, *9*, 849–859. [CrossRef]
2. Gilmore, P.C.; Gomory, R.E. A Linear Programming Approach to the Cutting Stock Problem-Part II. *Oper. Res.* **1963**, *11*, 863–1025. [CrossRef]
3. Golden, L.B. Approaches to the Cutting Stock Problem. *A I I E Trans.* **1973**, *8*, 265–274. [CrossRef]
4. Cui, Y.; Zhao, X.; Yang, Y.; Yu, P. A heuristic for the one-dimensional cutting stock problem with pattern reduction. *Proc. Inst. Mech. Eng. Part B J. Eng. Manuf.* **2008**, *222*, 677–685. [CrossRef]
5. Delorme, M.; Iori, M.; Martello, S. Bin packing and cutting stock problems: Mathematical models and exact algorithms. *Eur. J. Oper. Res.* **2016**, *255*, 1–20. [CrossRef]
6. Köksal, Ö.; Eroğlu, E. Solution Approach to Cutting Stock Problems Using Iterative Trim Loss Algorithm and Monte-Carlo Simulation. *J. Oper. Res. Stat. Econom. Manag. Inf. Syst.* **2023**, *11*, 125–136. [CrossRef]
7. Hadj Salem, K.; Silva, E.; Oliveira, J.F. Cutting and packing problems under uncertainty: literature review and classification framework. *Int. Trans. Oper. Res.* **2023**, *30*, 3329–3360. [CrossRef]
8. Fernandez-Viagas, V.; Prata, B.A.; Framinan, J.M. A critical-path based iterated local search for the green permutation flowshop problem. *Comput. Ind. Eng.* **2022**, *169*, 1–15. [CrossRef]
9. Utama, D.M.; Primayesti, M.D.; Umamy, S.Z.; Kholifa, B.M.N. Energy-efficient permutation flow shop scheduling problem: A systematic literature review. *J. Tek. Ind.* **2024**, *25*, 13–54. [CrossRef]
10. Singh, H.; Oberoi, J.S.; Singh, D. Multi-objective permutation and non-permutation flow shop scheduling problems with no-wait: A systematic literature review. *RAIRO Oper. Res.* **2021**, *55*, 27–50. [CrossRef]
11. Park, J.; Chun, J.; Kim, S.H.; Kim, Y.; Park, J. Learning to schedule job-shop problems: representation and policy learning using graph neural network and reinforcement learning. *Int. J. Prod. Res.* **2021**, *59*, 1–18. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.