engineering proceedings

MDPI

Proceeding Paper

Order Prioritization and Selection for Batch Sequencing Recovery in Multi-Model Assembly Lines: A Use Study with a Greedy Order-Selection Algorithm †

Meryem Boutbagha * and Laila El Abbadi

Laboratory of Advanced Systems Engineering, National School of Applied Sciences, Ibn Tofail University, Kenitra 14000, Morocco; laila.elabbadi@uit.ac.ma

* Correspondence: meryem.boutbagha@uit.ac.ma

† Presented at the 8th edition of the International Conference on Advanced Technologies for Humanity (ICATH 2026), Rabat, Morocco, 15–17 July 2026.

Abstract

This paper studies order prioritization and selection for Multi-Model Assembly Line recovery under minor disruption. When one line breaks down, reduced production capacity prevents all pending orders from being processed during the recovery period. The Order Prioritization Coefficient, a composite metric based on Customer Value, Profit Margin, Due Date, and Order Size, is used to rank orders. A greedy selection algorithm then retains the highest priority orders within the available production capacity. The approach is evaluated on 15 instances across three size categories, proving its effectiveness and producing sequencing input parameters for a subsequent recovery sequencing phase.

Keywords: order prioritization; order selection; Multi-Model Assembly Lines; minor disruption; recovery period; production line breakdown; production capacity reduction; batch sequencing; greedy algorithm

1. Introduction

Flexible Production Systems aim to produce different products in the same production system [1]. Their widespread use stems from their vital role in the survival of industries in an ever-changing, market-diverse environment [1]. A well-known category of Flexible Production Systems is Multi-Model Assembly Lines (MuMALs), which are widely used by advanced lean companies to achieve high productivity along with product variety [2]. However, such systems may still face unexpected disruptions, making the adoption of resilience in supply chain processes a prerequisite to effectively managing risks [3].

Resilience refers to an organization's capacity to anticipate potential threats, adapt to disruptions, and restore its performance to pre-disruption levels while maintaining required operational control levels [3]. Disruptions can be categorized into major disruptions, long-lasting events whose impacts could spread across different supply chain echelons, such as natural and man-made disasters, and minor disruptions, which originate from internal supply chain operations, such as demand and manufacturing [4]. Among the most common sources of minor disruptions is a production line breakdown, which temporarily reduces available production capacity, making it impossible for production managers to satisfy all pending orders within the recovery period (the time interval before the production line is repaired and put back into service) and forcing them to make quick decisions about which pending orders to keep for production and which to postpone. This situation is similar to

Check for updates

Academic Editors: Brahim El Bhiri, Ayman S. Mosallam and Younes Karfa Bekali

Published: 17 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.

the Order Acceptance Problem, where companies face limited production capacity relative to incoming orders and must sometimes refuse or delay them to maintain operational performance and profitability [5]. These kinds of decisions are critical for the company because they directly impact customer satisfaction, profits, and the company's reputation as a consequence.

Although Multi-Model Assembly Lines are growing in importance, the literature addressing them is scarce [2]. To the best of our knowledge, no study has yet examined the problem of order selection under disruption, including production capacity reduction, in a Multi-Model Assembly Line context.

In our previous work [6], we defined the Order Prioritization Coefficient (OPC) that integrates economic, strategic, and time-related decision criteria into a single prioritization score to support decision-makers in prioritizing their orders to address disruptions [6].

Based on the results of this work, the present study proposes a greedy order-selection algorithm that uses the OPC to identify the orders to keep for production during the recovery period following an assembly line breakdown. The set of retained orders will later serve as input to a sequencing phase.

The remainder of this paper is organized as follows: After the Introduction, Section 2 presents a brief review of the relevant literature. In Section 3, the problem is described, and the proposed OPC-based greedy order-selection procedure is explained. Section 4 presents the computational experiments and discusses the results. The last Section contains the conclusion and the directions for future research.

2. Literature Review

Assembly lines are "flow-oriented production systems" [7], used in the mass production of standard and personalized products [8]. The Multi-Model Assembly Line is a product-characteristic classification of assembly lines, along with the Single-Model and Mixed-Model Assembly Lines (MiMAL) [8]. The Single-Model Assembly Line (SMAL) is dedicated to a single product type [8], whereas in MiMAL, several models of the same parent, with negligible differences, are assembled [8]. The same applies to MuMAL, where models are assembled one by one, but in batches [2]. As a result, we can consider MiMAL a special case of MuMAL with a batch size of one. In contrast to MiMAL, where the setup is negligible, operations in MuMAL require setups between different models (see Figure 1). The setup should be reduced to its minimum, so that the time spent is no longer than the assembly work itself [2].

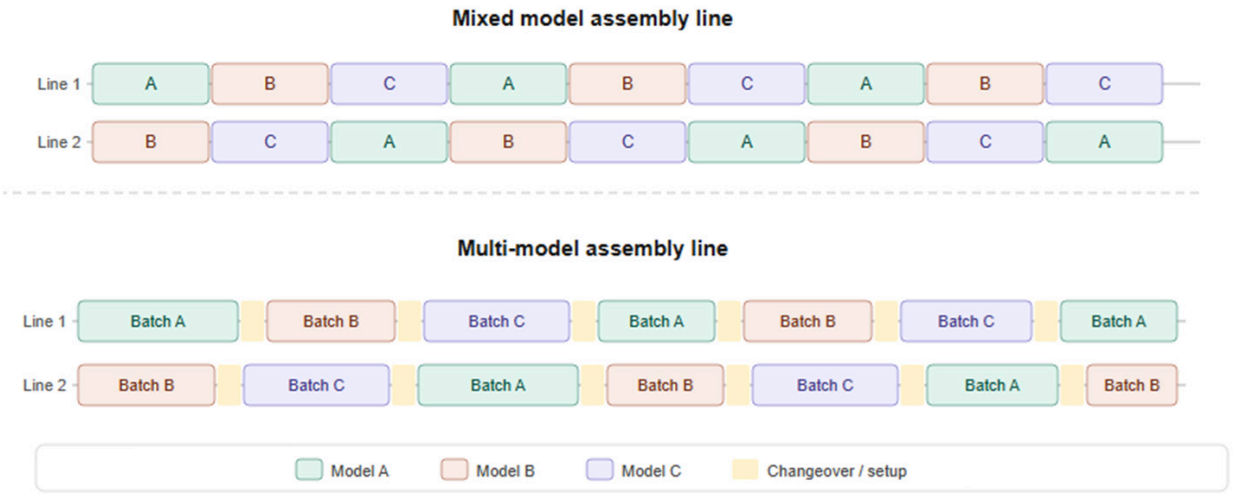

Figure 1. The difference between Mixed-Model Assembly Lines and Multi-Model Assembly Lines.

https://doi.org/10.3390/engproc2026168012

However, the degree of similarity of the assembled products in MuMAL does not suggest that the sequencing could be optional [9]. On the contrary, the assembly must be arranged in batches to optimize setup time and cost [9]. The production processes in MuMAL are also in some way similar. In most cases, the same machines and operators are used to manufacture the different models [9]. Certainly, MuMALs are of utmost importance in industry, along with the MiMAL, but they have not been given the same importance in the literature as is the case for the SMAL and MiMAL [2,10]. Moreover, it is the line-balancing problems in MuMAL that are studied, not the sequencing problems.

Disruption management in production systems, however, has attracted growing research interest [4]. The authors in [4] highlight that while the minor disruptions have a low impact individually, their frequency makes their cumulative effect significant. Although minor disruptions are common, they are vital factors in production systems [11].

When a disruption occurs, managers are obliged to adjust their production plan to fit the new conditions. Ideally, a recovery plan should deviate as little as possible from the original one, as addressed by the authors in [12], who studied the problem of restoring operations following disruptions while preserving proximity to the pre-disruption plan. However, the literature reveals a complete absence of econometric approaches to disruption analysis due to insufficient data. Most companies fear sharing data with researchers to avoid damage to their reputation [4].

Order prioritization and selection have primarily been addressed in the scientific literature in the context of order acceptance decisions made before production begins, focusing on identifying which orders to accept, given production capacity, to maximize profitability [6]. Indeed, selecting the most appropriate customer orders from potential candidates with respect to the remaining capacity is a paramount challenge [13] that has received considerable attention in the scientific literature [13–15]. Several factors influence the order acceptance decisions, namely: the decision period, order size, due date obligations, capacity constraints, and order preference [14].

While these studies share the same philosophy as our present work, namely the need to select a subset of orders under constrained production capacity, they remain fundamentally different in their context: Order acceptance decisions are made prior to production, while the problem addressed in the current study arises during a recovery period, when production capacity is temporarily reduced due to the occurrence of a disruption, and urgent decisions must be made on already-confirmed orders.

To address this gap, we introduced the OPC in our previous work [6], a single company-specific prioritization score that ranks orders based on four criteria: Customer Value, Profit Margin, Due Date Commitment, and Order Size. The different weights associated with each criterion are calculated using the Analytical Hierarchy Process (AHP), one of the most commonly used Multi-Criteria Decision Analysis techniques for establishing the relative priority of each element of a decision problem [6].

The present work takes this methodology a step further by proposing a greedy OPC-based order selection algorithm during the recovery period of a disrupted Multi-Model Assembly Line. The details of the OPC calculation and the proposed procedure are presented in the following section.

3. Problem Description and Proposed Solution Approach

3.1. Problem Description

The problem addressed in this study occurs within a production system containing identical Multi-Model Assembly Lines; each line is capable of producing different product models. Under regular conditions, all the L lines work simultaneously. We suppose that

a minor disruption caused the breakdown of one of the L lines. The other lines remain operational while the disrupted one is being repaired.

The recovery period T_r is defined as the time interval between the occurrence of the line breakdown and the complete restoration of the broken line to full operation. Unlike approaches that attempt to adjust existing production plans, the proposed approach redefines a new production plan for the recovery period. T_r serves as the new planning horizon for the new plan, which will subsequently be distributed over the functional lines. During T_r, the available production capacity is reduced, making it impossible to accommodate all pending orders within the recovery period. Consequently, a selection must be made among the pending orders to determine which ones will be retained for production during the recovery period, while ensuring that the total required production time does not exceed T_r.

Let R denote the set of pending orders at the time of the disruption, where each order is associated with a specific product model i, an Order Size O_r (in units), and a set of attributes including a Customer Value C_r, a Profit Margin P_r, and a Due Date DD_r (number of days to the delivery deadline). The processing time of order r, which is the time needed to fulfill it, is expressed by $ST_i + O_r.p_i$, where Where p_i is the time required to process model i, and ST_i is the setup time related to model i.

The batch size b_i of each product model i is assumed to be given and known, as it is typically determined by the production system configuration and remains fixed during the recovery period.

The objective of the proposed approach is twofold:

- First, to select a subset $S \subseteq R$ of orders to be kept for production during the recovery period, such that the cumulative processing time of the retained orders does not exceed the available recovery capacity T_r. This capacity constraint is expressed as follows:

$$\sum_{r \in S} ST_i + O_r \,.\, p_i \;\leq\; T_r \tag{1}$$

  The order selection should reflect company-specific priorities, linked to customer satisfaction, profitability, delivery urgency, and production capacity. The order selection is carried out using a greedy algorithm guided by the OPC and detailed in the following two sections (Sections 3.2 and 3.3).
- Second, to derive from S the following parameters: number of batches per retained order n_r, total number of batches to be sequenced Q^{tot}, the number of product models N, and the number of batches per model q_i^{tot}, through a conversion from retained orders to batches, which will serve as inputs to the sequencing phase of the recovery plan.

3.2. OPC-Based Order Prioritization

As mentioned above, and in order to guide the order selection decision, we will use the OPC, a composite metric that assigns a single prioritization score to each pending order r, representing its relative importance with respect to the four aforementioned decision criteria. The methodology of OPC calculation is summarized as follows:

- Step 1. AHP weighting: The different criteria are weighted using AHP pairwise comparison (w_j is the weight related to criterion j) and subject to a consistency check to validate the judgments.
- Step 2. Normalization: The data is normalized depending on whether the criterion is a benefit (higher is better) or a cost criterion (lower is better).
- Step 3. OPC Score: A weighted sum aggregates the four normalized values into a single priority score per order. The higher the OPC value, the higher the order's priority. The details of the full procedure are provided in [6].

https://doi.org/10.3390/engproc2026168012

3.3. Greedy Order-Selection Algorithm

To address the order selection problem during the recovery period, we will rely on the OPC score computed using the previously described methodology and propose a greedy order-selection algorithm. The algorithm will sort all pending orders in descending order of their OPC score and add them one by one to the retained set S, as long as τ, the cumulative processing time of each order, does not exceed the available recovery time T_r. The algorithm stops as soon as the next order in the ranked list would violate the capacity constraint. The pseudocode for the procedure is presented in Algorithm 1.

Algorithm 1. Pseudocode for OPC-based greedy order selection.

Input: R, T_r, p_i, ST_i, C_r, P_r, DD_r, O_r, w_1, w_2, w_3, w_4
Output: Retained order set S

1. Compute OPC_r for all $r \in R$
2. Sort orders in R in descending order of $OPC_r \rightarrow$ ranked list R^*
3. Initialize $S \leftarrow \emptyset$ and cumulative time $\tau \leftarrow 0$
4. For each order r in R^* do:
    - If $\tau + ST_i + O_r \cdot p_i \leq T_r$ then:
        - Add r to S
        - Update $\tau \leftarrow \tau + ST_i + O_r \cdot p_i$
    - Else: Stop
5. Return S

The algorithm guarantees that orders with the highest OPC score are always considered first, ensuring that the most important orders are kept whenever capacity allows. The computational complexity of the greedy order-selection algorithm is dominated by the sorting step and is of order $O(|R|\log|R|)$, where $|R|$ denotes the cardinality of the set R. This makes the algorithm highly efficient and suitable for real-time decision-making in disruption contexts, where the production managers need to make immediate decisions.

3.4. Conversion of Retained Orders and Definition of Sequencing Inputs

The retained order set S produced by the greedy algorithm serves not only as the result of the order selection procedure but also as the basis for defining the sequence input parameters, essential for the recovery period. The point behind the conversion is to translate the retained orders, which are expressed in units, into a batch representation, serving as input to the sequencing phase of the recovery plan.

Once S is determined, the following four steps are applied to obtain the sequencing input parameters n_r, N, q_i^{tot}, and Q^{tot}.

- Step 1. Identification of the different product models: This first step aims to identify the number of distinct product models included in S, which we denote N. $i \in \{1, 2, \ldots, N\}$ is the product model index.
- Step 2. Calculation of n_r, the number of batches per retained order: For each retained order $r \in S$, n_r is calculated as follows, with b_i the batch size of the product model associated with order r:

$$n_r = \left\lceil \frac{O_r}{b_i} \right\rceil \tag{2}$$

The ceiling function is used to guarantee that the number of batches is always an integer, since a partial batch must still be produced as a full batch.
It is noteworthy that determining the batch size is a batching problem in itself and falls outside the scope of the present study. The batch size is assumed to be predefined and fixed.

https://doi.org/10.3390/engproc2026168012

- Step 3: Computation of the total number of batches of the model: For each model $i \in \{1, 2, \ldots, N\}$, the total number of batches of model i across all retained orders is calculated as, where $i(r)$ denotes the product model associated with order r:

$$q_i^{tot} = \sum_{\substack{r \in S \\ i(r)=i}} n_r \quad (3)$$

- Step 4: Computation of the total number of batches, all models included: The total number of batches to be sequenced in the Multi-Model Assembly Line disrupted production system, during the recovery period, across all models, is

$$Q^{tot} = \sum_{i=1}^{N} q_i^{tot} \quad (4)$$

In summary, the following parameters N, q_i^{tot}, and Q^{tot} along with other MuMAL parameters, constitute the sequencing input parameters that will serve to solve the sequencing problem, which will be the scope of a future study.

4. Computational Experiments

4.1. Data Generation

Since no publicly available dataset or established benchmark exists for the specific problem considered in this study, and given that real industrial data could not be acquired due to confidentiality requirements, the computational study will be based on a set of generated data instances designed to reflect realistic MuMAL conditions. Existing assembly line datasets address sequencing under normal conditions and do not include the order selection dimension or the recovery period constraints that characterize the problem studied in the present work. In addition, the studies that consider assembly line disruptions focus on different problem settings and use parameters that are not directly applicable to our formulation. As a result, existing datasets are inappropriate for direct use, which motivates the generation of instances tailored to the specific characteristics of our problem.

Building on the instance generation methodology of [16], which addresses a multi-objective lot sizing and scheduling problem in flexible parallel production lines and classifies demand scenarios into three categories, namely: small, medium, and large, depending on the number of orders and product models, we adopt a similar structure to generate our instances. Three categories of instances are defined to evaluate the proposed solution approach at different levels of complexity, each containing five instances, for a total of 15 test instances. The categories have different values for the number of pending orders $|R|$, the number of product models N, and the Order Size O_r. The other parameters, p_i, ST_i and b_i are drawn from fixed ranges across all categories, while the recovery period T_r is generated per instance to reflect realistic minor disruption durations on an assembly line. The instance parameter ranges are presented in Table 1.

Table 1. The generated instance parameter ranges.

	$\|R\|$	N	O_r (units)	p_i (min/unit)	ST_i (min)	b_i (units)	T_r (min)
Small	DU[5,10]	DU[2,5]	DU[15,30]	U[2,5]	U[10,30]	DU[2,8]	DU[360,1440]
Medium	DU[10,20]	DU[6,9]	DU[30,60]	U[2,5]	U[10,30]	DU[2,8]	DU[360,1440]
Large	DU[20,30]	DU[10,12]	DU[60,90]	U[2,5]	U[10,30]	DU[2,8]	DU[360,1440]

4.2. Computational Results and Analysis

The proposed algorithm was implemented in Python (version 3.14.3) and tested on the 15 generated instances. The implementation is based directly on the pseudocode presented above, using a fixed seed of 42 to ensure reproducible results.

Table 2 presents the results obtained by applying the greedy algorithm to the 15 instances across the three categories. The first observation is that the algorithm has successfully determined sets of retained orders across all instances, proving its applicability irrespective of problem size. Second, τ is close to T_r in 73.33% of the instances (the proportion of τ relative to T_r exceeds 70%), indicating that the algorithm makes effective use of the recovery period with minimal idle time.

Table 2. Results of the greedy order-selection algorithm on small, medium, and large instances, and comparison with the modified greedy algorithm.

Category	Instance	$\|R\|$	N	T_r (min)	Basic Greedy Algorithm				Modified Greedy Algorithm			
					$\|S\|$	% of Retained Order	τ (min)	Q^{tot}	$\|S\|$	% of Retained Order	τ (min)	Q^{tot}
SMALL	1	8	2	373	4	50	272	42	**5**	**62.5**	**353**	**56**
	2	6	3	1123	6	100	643	51	6	100	643	51
	3	8	5	590	5	62.5	529	19	5	62.5	529	19
	4	9	4	472	4	44.4	395	35	**5**	**55.6**	**466**	**45**
	5	6	3	910	6	100	624	61	6	100	624	61
MEDIUM	1	11	9	1432	6	54.5	1321	62	**7**	**63.6**	**1400**	**78**
	2	10	9	1133	5	50	1089	62	5	50	1089	62
	3	13	7	1316	8	61.5	1276	82	8	61.5	1276	82
	4	12	8	1422	8	66.7	1380	90	8	66.7	1380	90
	5	14	7	1395	7	50	1337	55	7	50	1337	55
LARGE	1	29	11	662	2	6.9	467	65	**3**	**10.3**	**635**	**80**
	2	27	11	862	2	7.4	495	75	**3**	**11.1**	**832**	**83**
	3	21	11	920	3	14.3	794	112	3	14.3	794	112
	4	26	10	546	1	3.8	268	21	**2**	**7.7**	**422**	**36**
	5	26	12	1244	4	15.4	1025	68	**5**	**19.2**	**1179**	**77**

The values in red indicate instances where the modified greedy algorithm retains more orders than the basic greedy algorithm.

Another observation concerns the impact of order size on order retention. The results show that the larger the order (the instance), the smaller the percentage of retention, which goes from 44.4% to 100% in small, 50% to 66.7% in medium, and 3.8% to 15.4% in large instances. This is explained by the fact that the larger orders need more time individually, making it difficult to accommodate multiple orders within T_r. In particular, two small instances retain all the pending orders, corresponding to the situation where the recovery period is sufficiently long compared to the total workload, which constitutes a real situation when the disruption is minor and the number of pending orders is limited.

However, it is worth noting that the proposed algorithm stops as soon as the first order in the OPC-sorted list does not fit within the recovery time, without searching further down the list for an order that could still be suitable for insertion within the remaining recovery time. This behavior may result in a low number of retained orders and, consequently, longer idle time due to underutilization of the remaining capacity.

To address this limitation, a modified version of the basic greedy algorithm is proposed, which continues scanning the OPC-ranked list and inserts any subsequent order that fits within the remaining recovery capacity. As a result, instead of stopping at the first infeasible order (Else: Stop), it continues scanning the ranked list (Else: Continue) to insert any subsequent order that fits within the remaining recovery period.

It is worth noting that the modified algorithm introduces a trade-off: it improves capacity utilization, but may violate the OPC-based prioritization by retaining lower-priority orders over higher-priority ones that could not fit within the remaining recovery time.

In order to evaluate the benefit of the modified greedy algorithm, both algorithms are compared on the same 15 instances (see Table 2). The modified greedy algorithm gave better results in 7 out of 15 instances (marked in red), making better use of the remaining recovery period. In the remaining instances, both algorithms produce identical results, as no additional order could fit within the remaining time after the first infeasible order was met. The average retention rate increases from 45.8% to 49% across all instances and from 9.6% to 12.5% in large instances specifically, where the improvement is most significant, as the basic algorithm stops early due to large individual order processing times, leaving significant idle time.

5. Conclusions

The present work addresses the problem of order prioritization and selection for Multi-Model Assembly Line recovery under minor disruption. In a production context where a production line breaks down, the proposed approach aims to determine which pending orders should be retained for production during the recovery period, in accordance with the remaining production capacity.

The proposed approach comprises two steps. First, the Order Prioritization Coefficient (OPC), derived from the Analytical Hierarchy Process (AHP), is computed for each pending order. Second, a greedy order-selection algorithm uses the OPC ranking to iteratively retain orders as long as the capacity constraint is satisfied. The set of retained orders is then used to derive the input parameters for the sequencing phase.

Evaluated in 15 instances across three size categories, the greedy algorithm gave feasible order sets in all cases, although retention rates decreased with instance size due to the difficulty of accommodating large orders within the recovery period. A modified version addressing this limitation by continuing to scan for smaller orders demonstrated improved performance and reduced idle time.

Several directions for future work have been identified. First, the assumption of a known recovery period T_r may not always hold in practice; that is why future work could incorporate uncertainty on T_r through a stochastic programming framework. Second, extending our proposed approach to incorporate the batching decision problem rather than assuming fixed batch sizes also constitutes a relevant research perspective. The sequencing of the retained orders during the recovery period, using the parameters derived in the present study, will be addressed in a forthcoming work.

Author Contributions: M.B. and L.E.A. contributed equally to this work. Conceptualization, M.B. and L.E.A.; methodology, M.B. and L.E.A.; formal analysis, M.B.; investigation, M.B.; resources, M.B. and L.E.A.; writing—original draft preparation, M.B.; writing—review and editing, L.E.A.; supervision, L.E.A. Both authors contributed equally to the development of the manuscript. L.E.A. carried out the final review and corrections. 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: Data are available in the manuscript.

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

Abbreviations

MuMAL	Multi-Model Assembly Line
OPC	Order Prioritization Coefficient
MiMAL	Mixed Model Assembly Line
SMAL	Single Model Assembly Line
AHP	Analytical Hierarchy Process

References

1. Alavian, P.; Denno, P.; Meerkov, S.M. Multi-Job Production Systems: Definition, Problems, and Product-Mix Performance Portrait of Serial Lines. *Int. J. Prod. Res.* **2017**, *55*, 7276–7301. [CrossRef] [PubMed]
2. Fortuny-Santos, J.; Ruiz-de-Arbulo-López, P.; Cuatrecasas-Arbós, L.; Fortuny-Profitós, J. Balancing Workload and Workforce Capacity in Lean Management: Application to Multi-Model Assembly Lines. *Appl. Sci.* **2020**, *10*, 8829. [CrossRef]
3. Fogliatto, F.S.; Tortorella, G.L.; Saurin, T.A.; Furlan, A.; Galeazzo, A. Contribution of Lean Production to the Resilience of the Brazilian Automotive Supply Chain. *Prod. Plan. Control* **2025**, *36*, 1863–1878. [CrossRef]
4. Wu, M.; Holguin Veras, J. The Impacts of Minor Disruptions on a Production System: Empirical Analysis of Assembly Line Data. *Prod. Plan. Control* **2024**, *35*, 655–669. [CrossRef]
5. Slotnick, S.A. Order Acceptance and Scheduling: A Taxonomy and Review. *Eur. J. Oper. Res.* **2011**, *212*, 1–11. [CrossRef]
6. Boutbagha, M.; El Abbadi, L. Applying the Analytical Hierarchy Process (AHP) to Define the Order Prioritization Coefficient (OPC) for Production Planning under Disruptions. In Proceedings of the 2025 11th International Conference on Optimization and Applications (ICOA), Kenitra, Morocco, 16–17 October 2025; pp. 1–6.
7. El Ahmadi, S.E.A.; El Abbadi, L. Reducing Flow Time in an Automotive Asynchronous Assembly Line–An Application from an Automotive Factory. *Manag. Prod. Eng. Rev.* **2022**, *13*, 99–106. [CrossRef]
8. Saif, U.; Guan, Z.; Wang, B.; Mirza, J.; Huang, S. A Survey on Assembly Lines and Its Types. *Front. Mech. Eng.* **2014**, *9*, 95–105. [CrossRef]
9. Boysen, N.; Fliedner, M.; Scholl, A. Assembly Line Balancing: Which Model to Use When? *Int. J. Prod. Econ.* **2008**, *111*, 509–528. [CrossRef]
10. Jafari Asl, A.; Solimanpur, M.; Shankar, R. Multi-Objective Multi-Model Assembly Line Balancing Problem: A Quantitative Study in Engine Manufacturing Industry. *Opsearch* **2019**, *56*, 603–627. [CrossRef]
11. Pal, B.; Sana, S.S.; Chaudhuri, K. A Multi-Echelon Production–Inventory System with Supply Disruption. *J. Manuf. Syst.* **2014**, *33*, 262–276. [CrossRef]
12. Yang, J.; Qi, X.; Yu, G. Disruption Management in Production Planning. *Nav. Res. Logist. (NRL)* **2005**, *52*, 420–442. [CrossRef]
13. Akyildiz, B.; Kadaifci, C.; Topcu, Y.I.; Ulengin, B. Prioritization of Customer Order Selection Factors by Utilizing Conjoint Analysis: A Case Study for a Structural Steel Firm. *Int. J. Civ. Environ. Eng.* **2014**, *8*, 498–502.
14. Wang, J.; Yang, J.-Q.; Lee, H. Multicriteria Order Acceptance Decision Support in Over-Demanded Job Shops: A Neural Network Approach. *Math. Comput. Model.* **1994**, *19*, 1–19. [CrossRef]
15. Kalantari, M.; Rabbani, M.; Ebadian, M. A Decision Support System for Order Acceptance/Rejection in Hybrid MTS/MTO Production Systems. *Appl. Math. Model.* **2011**, *35*, 1363–1377. [CrossRef]
16. Yue, L.; Guan, Z.; Zhang, L.; Ullah, S.; Cui, Y. Multi Objective Lotsizing and Scheduling with Material Constraints in Flexible Parallel Lines Using a Pareto Based Guided Artificial Bee Colony Algorithm. *Comput. Ind. Eng.* **2019**, *128*, 659–680. [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.