Proceedings of the IISE Annual Conference & Expo 2025
E. Gentry, F. Ju, X. Liu, eds.

An Integrated Optimization Model for Production and Inventory Management in Distributed Manufacturing

Jun Kyu Ahn[1] and Faisal Aqlan[2]
[1]H. Milton Stewart School of Industrial & Systems Engineering, Georgia Institute of Technology
[2]Department of Industrial & Systems Engineering, University of Louisville

Abstract

Optimizing distributed manufacturing (DM) systems for efficient resource utilization and cost-effective production poses significant challenges due to the complex interdependencies among DM entities and dynamic operational conditions. These conditions can be classified into internal and external factors; for instance, internal factors include raw material shortages and capacity limitations, while order cancellation by customers may be considered as an external factor. Manual decisions in offloading orders and transshipping raw parts across DM sites often lead to inefficient operation costs and bottlenecks in production scheduling. In response to the need for automated and optimized decision-making, this research proposes a discrete, time-dependent integer programming approach that optimizes offloading and transshipment decisions while accounting for inventory levels and capacity constraints. The model is evaluated using a real case study from a distributed manufacturing system of an international electronics manufacturing company. The objective is to minimize operational costs and identify optimal timeframes for system operation. Although our model demonstrates notable performance improvements, the inherent NP-hard nature of the problem introduces runtime gaps for large-scale instances which presents opportunities for future research to enhance conventional integer programming methods.

Keywords
Distributed manufacturing, integer programming, order offload, inventory transshipment, electronics industry.

1. Introduction
The expansion of market size and production scale has driven companies to target a broader customer base, positioning distributed manufacturing (DM) as an effective management approach for large-scale production networks. DM systems primarily seek to expand the supply chain and enhance its cost-efficiency, resiliency, and sustainability through decentralizing warehouses and logistic hubs not only to meet the augmented demands worldwide but also to mitigate the operational risks from contingencies. With its merits, DM has emerged as a potential approach for developing advanced manufacturing framework. For developing high-quality DM systems, the Engineering Research Visioning Alliance (ERVA) pointed out data analytics and quality assurance as one of the research priorities for DM systems to improve the efficacy of production process and optimize inventory management [1]. However, as DM systems continue to expand and integrate advanced manufacturing approaches, their increasing complexity poses significant challenges in optimizing operations and maintaining coordination across distributed sites.

One of the key challenges limiting the optimization of operations is the technical complexity introduced by the diversification of manufacturing sites. When viewed as a connected network, the increase of nodes, denoting manufacturing sites and customer regions, not only amplifies the system's complexity but also increases interdependencies among manufacturing sites and product orders. This heightened interconnectivity introduces additional manufacturing and logistics constraints, making coordination more challenging. In most advanced manufacturing industries, researchers and engineers have developed various optimization strategies for complex networks and queues. For instance, Parvez et al. [2] proposed a lookahead strategy using an integer linear programming model to optimize computer server manufacturing systems with compatibility constraints. Their approach aimed to minimize backlogs caused by production scheduling limitations while maximizing the throughput. In DM systems, on the other hand, the internal operations such as order offloading and inventory transshipment are often managed manually based on expert decisions that raise the potential of having suboptimal and subjective operational decisions.

As supply chains and manufacturing systems grow worldwide, the need for greater diversification of facilities and resources becomes more pronounced, and the reliance on manual strategies may impede optimal decision-making and lead to unnecessary financial losses. Therefore, this study aims to investigate the effectiveness of optimization in operations of DM system including order assignment, inventory replenishment, order offloading, and parts transshipment utilizing an integrated optimization model. This paper presents a discrete integer programming (IP) model and examines the effectiveness of adopting optimization methods for enhancing the optimality of decisions made in complex DM systems. The objective is to examine the fundamental logic necessary for optimizing the four operational problems in DM systems mentioned above. With a premise that the optimization model is NP-hard due to the discrete nature of the problem, this research also tries to understand the challenges of utilizing conventional optimization models for large-scale problems and provide ideas of potential alternatives for future extensions.

2. Problem Description

Given is a set of distributed sites (A, B, and C) of our industry partner, which is a manufacturing company serving different geographical locations (see Figure 1). Customer orders are allocated to the three sites based on cost and other planning factors. Each site serves customers in its respective geographical areas. Manufacturing site A serves the North and South American markets; site B serves the European, Middle Eastern, and African (EMEA) markets, and site C serves the Asian and Australian markets. Raw materials are sourced from suppliers across the globe, including USA, Mexico, China, Taiwan, and Japan. The manufacturing environment is based on build-to-plan, make-to-order production strategies. Parts are received from suppliers and tested according to a predefined fabrication plan which is based on customer order forecasts. The tested parts are either used in customer orders or shipped to sister sites. Shipping the parts to sister sites can be based on a predefined shipping plan or an immediate shipping request by the sister plant. The immediate shipping requests arise from inherent uncertainties in demand forecasts, quality issues, and machine failures. Orders can also be offloaded from one site to another to avoid supply and demand shortages.

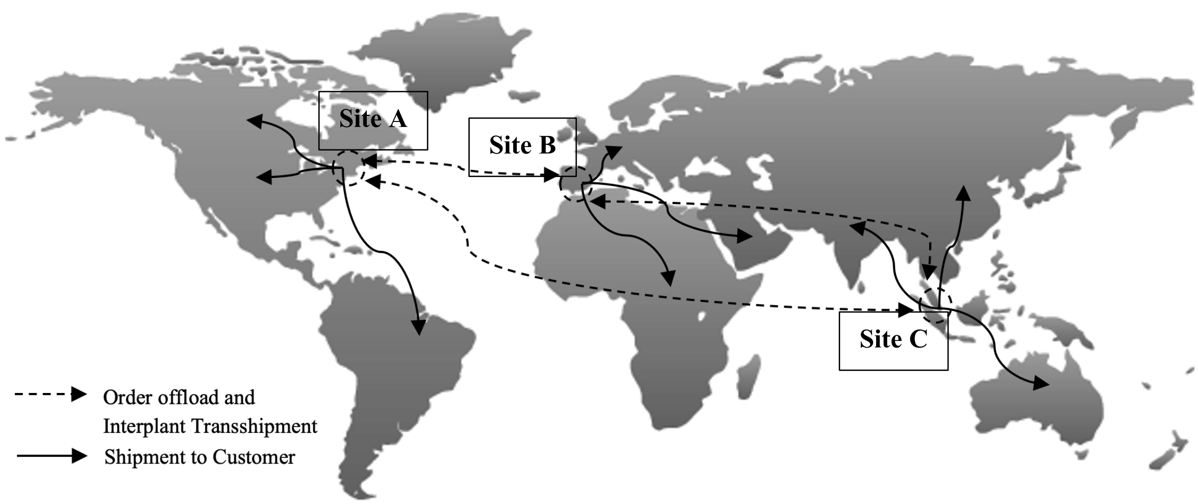

Figure 1. Distributed manufacturing system with order offload and inventory transshipment

Customer orders are characterized by order quantity, type, configuration, destination, and ship date. A manufacturing site may not be able fulfill a customer order due shortage of raw materials, capacity limitations, and/or policy factors. For raw materials shortage, the site can obtain parts from a sister site or an external supplier. For the other factors (i.e., capacity and policy), the site can offload the order to a sister site that can fulfill the order. The orders are received by a centralized order management system and then get assigned to the different sites. The order assignment is updated on a weekly basis and the company follows a quarterly planning horizon. The manufacturing environment operates on a build-to-plan, make-to-order production strategy. All manufacturing sites adhere to an (s, S) policy for parts replenishment, meaning that when the inventory of parts falls below the threshold s, the operations department will order parts to bring the inventory level up to a fixed level S.

3. Relevant Literature

This study focuses on optimizing order scheduling and resource allocation in DM systems. Notable studies in this field include Taylor and Whicker's optimization framework and heuristics for material delivery planning [3]. Additionally, researchers such as Shen et al. have explored agent-based approaches for optimizing order scheduling problems [4]-[6]. However, research on DM remains relatively nascent, particularly concerning internal operations and inventory replenishment. Existing optimization frameworks primarily address high-level scheduling and logistics rather than internal system efficiencies. To address this gap, this study proposes an optimization framework for two key internal DM system operations: order offloading and inventory transshipment. We provide a mathematical justification for optimizing DM system operations through integer programming logic constraints. Further, we introduce a novel optimization model designed to improve offload scheduling and inventory management by integrating real-world constraints and operational complexities.

4. Methodology

Figure 2 illustrates the overall manufacturing and shipping process, including offloading and transshipment, modeled as a discrete-time process. The simulation was primarily tested on a case study from an international electronics manufacturing company, incorporating 100 scheduled orders with initial assignments, along with capacity constraints, production costs, and shipping times from all manufacturing sites. Additionally, two randomly sampled datasets with 300 and 500 orders each was tested to compare the runtime with those of the smaller dataset from the case study. The objective function of the optimization model minimizes operational costs associated with inventory, manufacturing, and shipping. The model was implemented in Python using the Gurobi Optimizer, with order data stored in .csv files. Missing data were imputed using a uniform distribution via the Monte Carlo method, implemented through a Visual Basic Application (VBA) macro in Microsoft Excel. Performance statistics were generated with a system time of $T = 50$, comparing optimized scheduling with offloading against initial assignments.

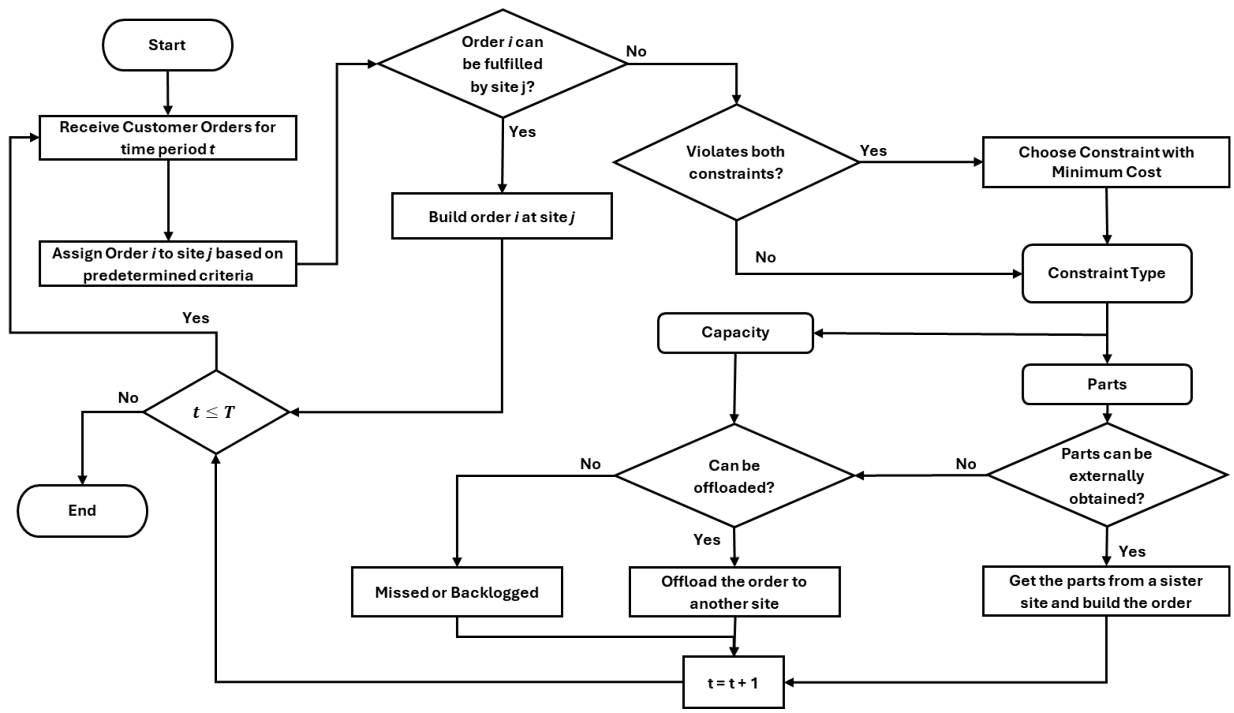

Figure 2. Flowchart of distributed manufacturing with order offload and inventory transshipment

The integer programming model below uses a constraint programming approach to satisfy the following key assumptions: each order can only be assigned to one site at most (1), order must be scheduled within the feasible time range (2), transshipment must be made between distinct sites (3), the order is shipped only after the production is confirmed (4), order is shipped to the customer right after the production and shipping is completed (11+12), product under production occupies the server (11+13), and operations must be completed before the deadline (17+18). Decisions on offloading and transshipment were primarily influenced by capacity constraints, including product type capacities (7), parts inventory levels (8), aggregate product capacities (9), and server capacities (14). Subsequently, inventory levels were governed by the discrete-time inventory balance equation constraints for parts (5) and products (6), the (s, S) inventory replenishment policy for parts (15+16), and transshipment quantity constraints (19+20). Helper constraints were included in computing the total processing time (10) and handling conditional logic using binary helper variables (21). Additionally, nonnegativity was ensured for all variables (22-24). Logic constraints including (11-13) and (15-20) used the big-M method for the conditional if-else logic that M is a large real number.

For the optimization to yield feasible solutions, all processing time combinations had to fit within the order deadlines, and the total system time, T_e, needs to be sufficiently large to accommodate all orders while adhering to capacity constraints. Since production costs were identical across all three sites in the case study, the statistical analysis excluded production costs to allow a clearer comparison of operational costs between initial assignments and optimized decisions. The following assumptions were made for simplification: (1) a fixed cost is charged for the transshipment between the sites regardless of the quantity (in the real manufacturing system, this cost depends on shipping speed and quantity), (2) parts replenishment and offloading decisions are made in the beginning of the day, (3) deadlines are given as integers that are compatible with the model's discrete timeframe.

Integer Programming Model

Sets

$I = \{i \mid i \in \mathbb{Z}_0^+\}$: set of orders

$J = \{j \mid j \in \mathbb{Z} \wedge [1,3]\}$: set of manufacturing sites

$N = \{n \mid n \in \mathbb{Z} \wedge [1,5]\}$: set of part types

$V = \{v \mid v \in \mathbb{Z} \wedge [1,4]\}$: set of product types

$T = \{t \mid t \in \mathbb{Z}^+ \wedge [0, T_e]\}$: set of time periods

Parameters

$b_{jnt} \in \mathbb{Z}_0^+$: quantity of part n that should be available at site j at the end of time t, s.t. $b_{jn0} = c_{jn}\ \forall j \in J, n \in N$

$\beta_{ijt} = \begin{cases} 1 & \text{order } i \text{ is shipped from site } j \text{ at time } t \\ 0 & \text{otherwise} \end{cases}$, s.t. $\beta_{ij0} = 0\ \forall i \in I, j \in J$

$\rho_{jvt} \in \mathbb{Z}_0^+$: quantity of product type v occupied in site j at time t, s.t. $\rho_{jv0} = 0\ \forall j \in J, v \in V$

$\varphi_{jj't} = \begin{cases} 1 & \text{transshipment occurred from site } j' \text{ to site } j \text{ at time } t \\ 0 & \text{otherwise} \end{cases}$, s.t. $\varphi_{jj'0} = 0\ \forall j, j' \in J$

$\tau_i \in \mathbb{Z}_+$: total processing time of order i

$R_{ijt} = \begin{cases} 1 & \text{order } i \text{ occupies a server at site } j \text{ at time } t \\ 0 & \text{otherwise} \end{cases}$, s.t. $R_{ij0} = 0\ \forall i \in I, j \in J$

Decision Variables

$X_{ijt} = \begin{cases} 1 & \text{order } i \text{ is assigned to site } j \text{ at time } t \text{ (assignment confirmation)} \\ 0 & \text{otherwise} \end{cases}$, s.t. $X_{ij0} = 0\ \forall i \in I, j \in J$

$Y_{jnt} \in \mathbb{Z}_0^+$: quantity of part n purchased at site j at t, s.t. $Y_{jn0} = 0\ \forall j \in J, n \in N$

$S_{jj'nt} \in \mathbb{Z}_0^+$: quantity of parts n are shipped from site j to site j' at time t, s.t. $S_{jj'n0} = 0\ \forall j, j' \in J, n \in N$

Data

$a_{ij} = \begin{cases} 1 & \text{order } i \text{ is initially assigned to site } j \\ 0 & \text{otherwise} \end{cases}$

PC_{ij}: processing cost of order i at site j

PT_{ij}: processing time of order i at site j

SC_{ij}: Shipping cost of order i from site j

ST_{ij}: Shipping time of order i from site j

MC_n unit purchasing cost of part n

h_n: unit inventory holding cost of part n

$TC_{jj'}$: Transshipment cost from site j' to site j

$TT_{jj'}$: Transshipment time from site j' to site j

P_{in}: quantity of part n required for order i

$p_{iv} = \begin{cases} 1 & \text{order } i \text{ is product type } v \\ 0 & \text{otherwise} \end{cases}$

c_{jn}: capacity allocated for part n at site j

C_{jv}: product capacity for type v at site j

$PSSD_i$: due date of order i

L_j: number of manufacturing stations at site j

s_{jn}: inventory replenishment level for part n at site j

Objective Function

$$\min Z = \sum_{t \in T} \left[\sum_{i \in I} \sum_{j \in J} X_{ijt}\left(PC_{ij} + SC_{ij}\right) + \sum_{j \in J} \sum_{n \in N} \left(Y_{jnt} MC_n + b_{jnt} h_n\right) + \sum_{j \in J} \sum_{j' \in J} \varphi_{jj't} TC_{jj'} \right]$$

Constraints

$$\sum_{t\in T}\sum_{j\in J} X_{ijt} = 1 \qquad \forall i\in I \ (1)$$

$$\sum_{t>PSSD_i}\sum_{j\in J} X_{ijt} = 0 \qquad \forall i\in I \ (2)$$

$$\sum_{t\in T} S_{jjnt} = 0 \qquad \forall j\in J, n\in N \ (3)$$

$$\sum_{t\in T}\sum_{j\in J}\left(X_{ijt}-\beta_{ijt}\right)\ge 0 \qquad \forall i\in I \ (4)$$

$$b_{jn(t+1)} = b_{jnt} + Y_{jn(t+1)} + \sum_{j'\in J\setminus j} S_{jj'n(t+1)} - \left(\sum_{i\in I} P_{in}X_{ij(t+1)} + \sum_{j'\in J\setminus j} S_{j'jn(t+1)}\right) \qquad \forall j\in J, n\in N, t\in T \ (5)$$

$$\rho_{jv(t+1)} = \rho_{jvt} + \sum_{i\in I} p_{iv}\left(X_{ij(t+1)} - \beta_{ij(t+1)}\right) \qquad \forall j\in J, v\in V, t\in T \ (6)$$

$$\rho_{jvt}\le C_{jv} \qquad \forall j\in J, v\in V, t\in T \ (7)$$

$$b_{jnt}\le c_{jn} \qquad \forall j\in J, n\in N, t\in T \ (8)$$

$$\sum_{v\in V}\rho_{jvt}\le \sum_{v\in V} C_{jv} \qquad \forall j\in J, t\in T \ (9)$$

$$\tau_i \ge \sum_{j\in J}\left\{\left(PT_{ij}+ST_{ij}\right)\cdot\left(\sum_{t\in T} X_{ijt}\right)\right\} \qquad \forall i\in I \ (10)$$

$$X_{ijt}\le M\left(1-u_{ijt}\right) \qquad \forall i\in I, j\in J, t\in T \ (11)$$

$$-\beta_{ij(t+\tau_i)}\le M\cdot u_{ijt} \qquad \forall i\in I, j\in J, t\in T \ (12)$$

$$1-\frac{\sum_{k=1}^{PR_{ij}-1} R_{ij(t+k)}}{PR_{ij}-1}\le M\cdot u_{ijt} \qquad \forall i\in I, j\in J, t\in T \ (13)$$

$$\sum_{i\in I}\left(X_{ijt}+R_{ijt}\right)\le L_j \qquad \forall j\in J, t\in T \ (14)$$

$$s_{jn}-b_{jnt}\le M\left(1-u_{jnt}\right) \qquad \forall j\in J, n\in N, t\in T \ (15)$$

$$c_{jn}-b_{jnt}-Y_{jn(t+1)}\le M\cdot u_{jnt} \qquad \forall j\in J, n\in N, t\in T \ (16)$$

$$\sum_{t\in T} X_{ijt}\le M\left(1-u_{ij}\right) \qquad \forall i\in I, j\in J \ (17)$$

$$\left(PR_{ij}+ST_{ij}\right)\cdot\left(\sum_{t\in T} X_{ijt}\right) - PSSD_i \le M\cdot u_{ij} \qquad \forall i\in I, j\in J \ (18)$$

$$\sum_{n\in N} S_{jj'n(t+TT_{jj'})}\le M\left(1-u_{jt}\right) \qquad \forall j\ne j'\in J, t\in T \ (19)$$

$$-\varphi_{jj't} < M\cdot u_{jt} \qquad \forall j\ne j'\in J, t\in T \ (20)$$

$$u_\omega\in\{0,1\} \qquad \forall \omega \ (21)$$

$$X_{ijt}\in\{0,1\} \qquad \forall i\in I, j\in J, t\in T \ (22)$$

$$0\le Y_{jnt}\le c_{jn} \qquad \forall j\in J, n\in N, t\in T \ (23)$$

$$S_{jj'n}(t)\ge 0 \qquad \forall j,j'\in J, n\in N, t\in T \ (24)$$

5. Results and Analysis

In Figure 3, the optimization of a schedule for 100 orders involved 67,596 integer variables. The final solution assigned 39 orders to the first site, 30 to the second, and 31 to the last, with 63 orders offloaded to another site from the initial schedule, as shown in Figure 4. This optimization yielded an objective value of $z = 5{,}019{,}759.00$, reflecting an 8.523% reduction from the initial schedule's final cost. As shown in Figure 4, the optimization process began with the Barrier algorithm, which completed 13 iterations of LP relaxation in 8.03 seconds, followed by the Dual Simplex algorithm, which took 3.75 seconds for root relaxation. The process then proceeded to the Branch-and-Bound algorithm, which required 83.38 seconds to yield a globally optimal solution with a gap of 0.0098%. However, for larger order schedules, the model encountered a memory leak at the Branch-and-Bound step and terminated before reaching global optimality. For instance, in the simulation with 500 orders, the model ran for 58,410.67 seconds before terminating at the Branch-and-Bound step.

Prior to termination, LP relaxation and root refinement were completed sequentially, with the Barrier algorithm taking 261.31 seconds and the Dual Simplex algorithm taking 329.99 seconds. We observe a sharp increase in computational time within the Branch-and-Bound algorithm for large-scale instances, highlighting factorial complexity that makes solving the problem infeasible within polynomial runtime as the order size grows [4].

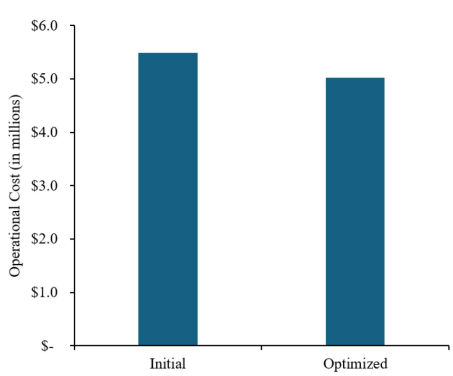

Figure 3: Operational cost with 100 orders

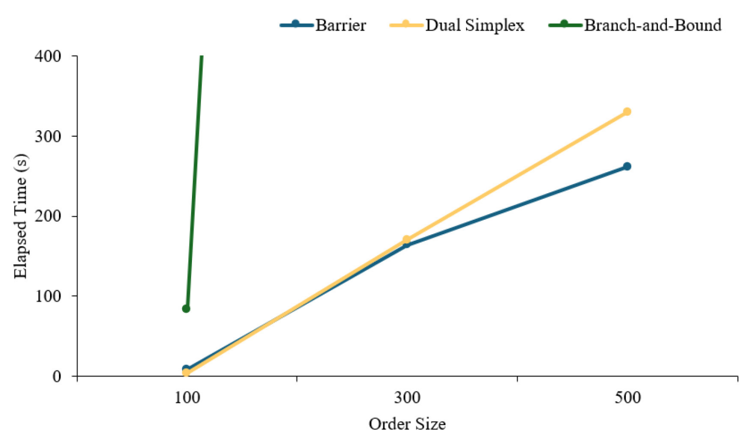

Figure 4: Runtime analysis of optimization algorithms

6. Conclusions and Future Work

In this study, we proposed an IP model for managing DM systems, focusing on order offload and inventory transshipment under real-world policies. The 8% reduction in operational costs demonstrates the advantages of mathematical optimization over manual decision-making, improving efficiency, reducing delays, and optimizing resource allocation. However, due to the NP-hard nature of the problem, IP remains computationally feasible only for small-scale instances. The primary challenge in large-scale DM optimization stems from the exponential growth of the solution space and the added complexity of auxiliary variables from the big-M method. A classical job shop scheduling problem with n jobs and m machines has a runtime complexity of $(n!)^m$ [4], illustrating the computational burden of IP models in industrial applications where timely decision-making is crucial. Additionally, the factorial growth of the Branch-and-Bound algorithm runtime manifests the difficulty of yielding a feasible integer solution with global optimality in large-scale simulations.

Given these limitations, heuristics provide a promising alternative for scalable optimization. For instance, Nezamoddini et al. [7] proposed the use of genetic algorithms with artificial neural networks to optimize inventory levels in stochastic supply chain networks, and Bertacco et al. [8] introduced the feasibility pump heuristic for general mixed-integer programming models that enable near-optimal solutions with significantly reduced runtime. To accelerate the Branch-and-Bound algorithm, Deza and Khalil [9] provided a comprehensive survey on machine learning-driven approaches to enhance cut selection process, yet challenges remain in model interpretability and generalizability.

Future research may explore heuristic-based and machine-learning approaches to enhance computational efficiency and assess the feasibility of IP for large-scale DM system management. By leveraging the mathematical rigor and logical structure of the IP model, we can incorporate machine learning techniques to efficiently generate high-quality solutions within a computationally feasible timeframe, making this approach particularly suitable for large-scale simulations and complex decision-making environments.

Acknowledgments

This study is based upon work supported by the National Science Foundation under grants CMMI-2038325 and EEC-2244119. Any opinions, findings, or recommendations expressed in this study are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

[1] Engineering Research Visioning Alliance, *Engineering the Future of Distributed Manufacturing: A Visioning Report*. Columbia, SC: SSRN, Sep. 2023, doi: 10.2139/ssrn.4512398

[2] M. Parvez, P. J. Parikh, F. Aqlan, and Md. Noor-E-Alam, "An online dynamic dual bin packing with lookahead approach for server-to-cell assignment in computer server industry," *Computers & Industrial Engineering*, vol. 196, pp. 110404, Jul. 2024, doi: 10.1016/j.cie.2024.110404.

[3] G. D. Taylor, and G. L. Whicker, "Optimization and Heuristic Methods Supporting Distributed Manufacturing." *Production Planning & Control*, vol. 13, no. 6, pp. 517–528, 2002, doi: 10.1080/09537280210162932.

[4] W. Shen, L. Wang, and Q. Hao, "Agent-based distributed manufacturing process planning and scheduling: a state-of-the-art survey," in *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 36, no. 4, pp. 563-577, July 2006, doi: 10.1109/TSMCC.2006.874022.

[5] P. Leitão, "Agent-based distributed manufacturing control: A state-of-the-art survey," *Engineering Applications of Artificial Intelligence*, vol. 22, no. 7, pp. 979–991, Oct. 2009, doi: 10.1016/j.engappai.2008.09.005.

[6] G. Jules and M. Saadat, "Agent Cooperation Mechanism for Decentralized Manufacturing Scheduling," in *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 12, pp. 3351-3362, Dec. 2017, doi: 10.1109/TSMC.2016.2578879.

[7] N. Nezamoddini, A. Gholami, and F. Aqlan, "A risk-based optimization framework for integrated supply chains using genetic algorithm and artificial neural networks," *International Journal of Production Economics*, p. 107-569, Dec. 2019, doi: 10.1016/j.ijpe.2019.107569.

[8] L. Bertacco, M. Fischetti, and A. Lodi, "A feasibility pump heuristic for general mixed-integer problems," *Discrete Optimization*, vol. 4, no. 1, pp. 63–76, 2007, doi: 10.1016/j.disopt.2006.10.001.

[9] A. Deza and E. Khalil, "Machine Learning for Cutting Planes in Integer Programming: A Survey," *arXiv preprint*, arXiv:2302.09166, Oct. 2023, [Online]. Available: https://arxiv.org/pdf/2302.09166