CHEMICAL ENGINEERING TRANSACTIONS

VOL. 121, 2025

Guest Editors: Petar S. Varbanov
Copyright © 2025, AIDIC Servizi S.r.l.
ISBN 979-12-81206-22-9; **ISSN** 2283-9216

A publication of
AIDIC
The Italian Association
of Chemical Engineering
Online at www.cetjournal.it

DOI: 10.3303/CET25121026

Comparative Analysis of Discrete-Time and Precedence-Based MILP Formulations for Sustainable Scheduling in Furniture Manufacturing

Botond Bertók*,[a], Károly Kalauz[a], Márton Frits[b]

[a]Széchenyi István University, Egyetem tér 1, 9026 Győr, Hungary
[b]University of Pannonia, Veszprém, Egyetem utca 10, Hungary
bertok.akos.botond@sze.hu

Efficient production scheduling plays a pivotal role in enhancing productivity and reducing energy consumption in mass manufacturing environments. This study presents a comparative evaluation of two mixed-integer linear programming (MILP) formulations - Discrete-Time Process Network Synthesis (PNS) and Precedence-Based Time-Constrained Process Network Synthesis (TCPNS) - for optimizing production scheduling in furniture manufacturing. Both approaches are grounded in the P-graph framework, which excels at representing complex, flexible process recipes commonly found in large-scale production systems. The TCPNS model, with its precedence-based structure, offers high-resolution scheduling capabilities and accurately manages complex changeover constraints. It enables the computation of exact start times and resource allocations, leading to highly optimized schedules. However, this precision comes with increased computational demand, which can become impractical for large-scale instances. Conversely, the PNS approach discretizes the planning horizon into time slots, significantly reducing model size and complexity. While this may result in less granular schedules, the formulation allows for faster solution times and easier integration of combinatorial simplifications, making it a practical alternative for real-time applications. The research also explores automated model generation techniques for both formulations, highlighting multi-resolution capabilities in the discrete-time approach that allow flexible trade-offs between accuracy and computational effort. A real-life case study from the furniture manufacturing sector is used to benchmark the two optimization strategies. The results demonstrate the practical implications of each method in terms of schedule precision, computational performance, and energy-aware utilization, i.e., if minute-to-minute scheduling is sufficient instead of milliseconds, then traditional PNS algorithms can offer the same sustainable solution with 10,000 times faster computation.

1. Introduction

Manufacturing plants are challenged to reduce their environmental impact and operational costs while maintaining production flexibility. A significant contributor to inefficiency in such systems is the constant energy draw from infrastructure components—heating, lighting, air handling, and extraction systems—that remain active regardless of the actual production rate. Consequently, lower plant utilization leads to increased energy consumption per manufactured unit.

In the context of furniture manufacturing, this inefficiency is especially critical due to batch-based, multi-step processes involving sawing, drilling, edge banding, assembly, and finishing. Integrating smaller subcontractor jobs into idle periods of the main production line presents a viable opportunity for energy and cost optimization. To guarantee the optimal schedule, mathematical programming models apply either time interval (Lim et al., 2023) or precedence-based (Kasapidis et al., 2023) formulations. The time horizon can be divided into fixed (Lim et al., 2022) or variable length (Markowski et al., 2022) time intervals. Besides binary variables, precedence can be expressed by process graphs as well (Kalauz et al., 2012). Process graphs or P-graphs are capable of managing complex changeover times (Frits and Bertok, 2020) or flexible recipes (Kalauz et al., 2024) in both fixed and variable time representations.

Paper Received: 23 July 2025; Revised: 13 September 2025; Accepted: 4 November 2025
Please cite this article as: Bertók B., Kalauz K., Frits M., 2025, Comparative Analysis of Discrete-Time and Precedence-Based MILP Formulations for Sustainable Scheduling in Furniture Manufacturing, Chemical Engineering Transactions, 121, 151-156 DOI:10.3303/CET25121026

This study compares the latest two optimization strategies aimed at better utilizing production resources by modeling how smaller operations can fill idle periods. Both methods leverage the P-graph framework for Process Network Synthesis, but differ fundamentally in their treatment of time constraints. The first approach uses discrete time slots and is computationally light, while the second uses precise time variables assigned to each operation ordered by their precedence.

2. Methodology

The P-graph framework (Friedler et al., 2022) is a combinatorial method designed to generate and evaluate feasible process structures starting from a superstructure that contains all possible operations. Its foundation relies on three core algorithms. The first, Maximal Structure Generation (MSG), constructs the complete superstructure from which feasible networks can be derived. The second, Solution Structure Generation (SSG), enumerates all structurally feasible sub-networks. Finally, the Accelerated Branch-and-Bound (ABB) algorithm identifies the n-best solutions within the space of feasible structures.

2.1 Discrete Time Process Flow Model (PNS)

The PNS model divides the production horizon into discrete time slots, with each activity mapped to a set of eligible slots according to its duration and required resources. Capacity constraints are defined for each slot, while transitions between tasks—such as equipment reconfiguration—are explicitly represented. The model relies on key assumptions, including fixed-duration operations, predefined time windows, and limited representation of dynamic interactions or overlapping processes. Its main advantages lie in the simplicity of its mathematical formulation, the fast computation enabled by Mixed Integer Programming (MIP), and its effectiveness in rough-cut capacity planning. Nevertheless, certain limitations arise from the rigidity of slot boundaries, as well as the potential loss of precision when modeling real-time production interactions.

2.2 Time Constrained Process Network Synthesis (TCPNS)

TCPNS extends the PNS framework by directly incorporating temporal dynamics through the association of timing variables with the start and end of each operation. The model accounts for fixed and proportional time charges, such as setup and processing times, as well as precedence constraints between operations, equipment limitations, and changeover requirements. It can be formulated using continuous or mixed-integer variables, capturing the temporal flow of materials in an explicit manner. TCPNS further enables precise scheduling, supports varying batch sizes, facilitates detailed changeover modeling, and allows real-time resource allocation.

3. Case Study: Laminated Furniture Production

Balaton Bútor, founded in 1896, is a Hungarian furniture manufacturer operating seven production facilities across the country. The company’s manufacturing processes include operations such as sawing, drilling, edge banding, assembly, painting, and packaging. This case study focuses on the production of a standard laminated cabinet, which involves a sequence of operations with the following standard processing times: 30 min for sawing, 5 min for drilling, and 15 min for edge banding.

The raw material used is laminated chipboard, and the target product is a set of pre-assembled cupboard parts. The main feature of this process is its flexible recipe, which allows the drilling and edge banding operations to be performed in either order.

An important goal in the allocation of available manufacturing resources is maximizing utilization. Due to varying batch sizes, this can only be achieved if the optimization process accounts for the fact that the order of machining operations is interchangeable. The modeling of the manufacturing process based on PNS and TCPNS is presented in the next section.

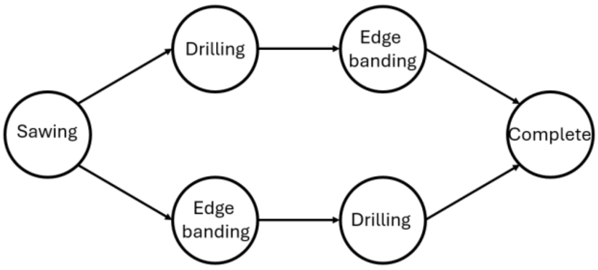
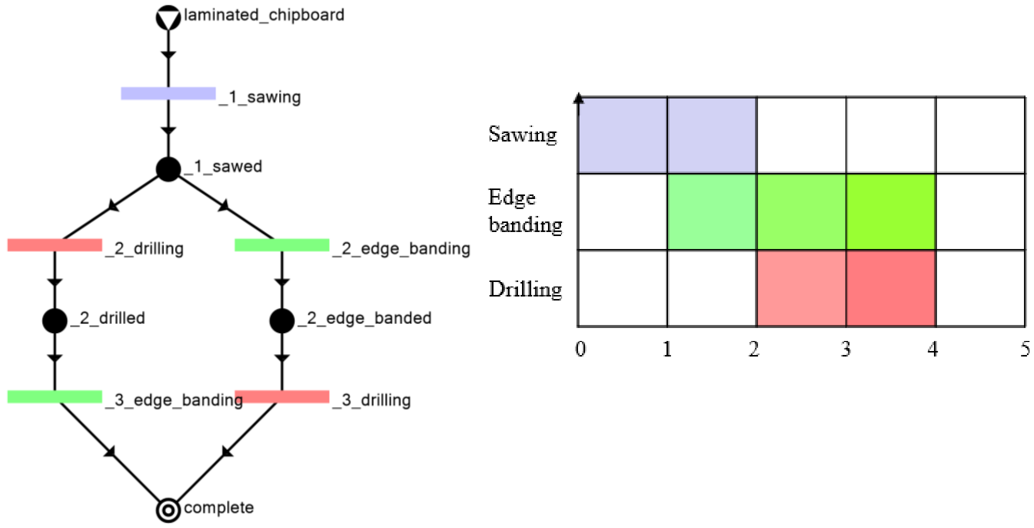


Figure 1: Flowchart of manufacturing steps



a) b)

Figure 2: a) P-graph modeling the flexible recipe for the case study b) Time slots available for the operations

3.1 Application of Models

The two approaches are presented through the furniture manufacturing process with a flexible recipe. Their runtime performance is evaluated, along with an analysis of the advantages and disadvantages of each model. The modeling steps are illustrated through a simplified example related to the presented manufacturing process. The initial step focuses on formulating the presented manufacturing process within the P-graph framework. The input of the process is a laminated chipboard, on which the sawing operation is performed first. As a result of the cutting, a size-adjusted laminated chipboard is produced.

The process does not specify the order of the drilling and edge-banding operations, meaning that the graph must allow both possible sequences. Depending on which operation is performed first, the other must be executed next to complete the process. The P-graph structure representing the process is shown in Figure 2a, which was created with the P-graph Studio software (Bertok and Heckl, 2016).

The main goal of the original problem in this paper is to utilize unused capacity during production for subcontracting work. However, the illustrative example only demonstrates how to manage the idle time of individual resources. For easier interpretation, every operation is assumed to take 1 time unit, and the available idle time slots are shown in Figure 2.b.

In the illustrative example, the saw machine is available only during the 1st and 2nd time slots, while edge banding can be performed in the 2nd, 3rd, and 4th time slots, and drilling is possible in the 3rd and 4th time slots. In the example, the production of two pieces of furniture is scheduled during the machines' available free time slots according to the previously defined manufacturing process. The appropriate P-graph structure is defined, and algorithms for the automatic generation of this structure are formulated for both standard PNS and TCPNS applications.

3.2 Modeling by PNS

The PNS was not originally designed to handle time-dependent parameters. Therefore, the processing of individual time slots must be achieved by appropriately structuring the P-graph. For each time slot, it is necessary to determine how many ways the given activity can be executed, and separate operations must be created for each variation. In the illustrative example, the sawing operation requires one time unit, whereas the saw is available for two time units. Consequently, two distinct sawing operations are defined in the P-graph: one corresponding to sawing in the 0th time slot, and another in the 1st time slot. As part of this step in model construction, all possible operations are generated; however, the structure must also ensure that these operations are arranged in accordance with the process recipe. The flexible recipe allows drilling and edge banding to be performed in either order for a given batch, which requires recording these operations twice. The naming convention for the generated operations is as follows: in the case of _2_drilling_3, the drilling operation is the second step and starts in the third time slot.

In the next phase of model generation, the inputs and outputs of each operation are defined. The input of an operation represents the material on which the processing step is to be performed, and which must be available

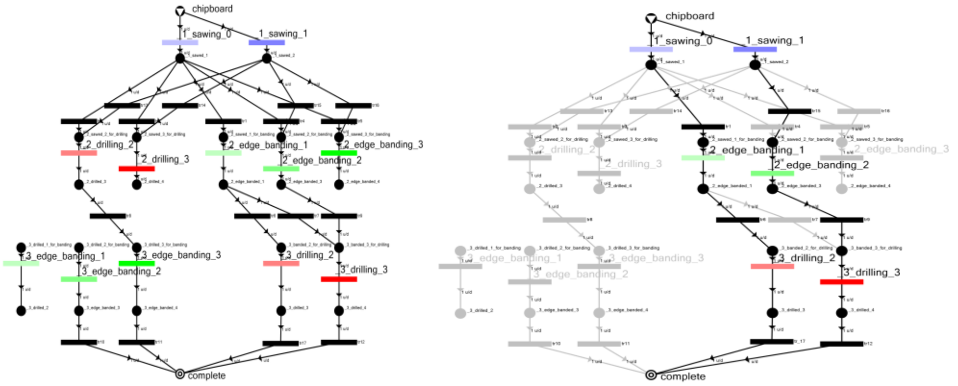


Figure 3: a) The P-graph model after the last step of model generation, b) and the solution structure associated with the optimal solution

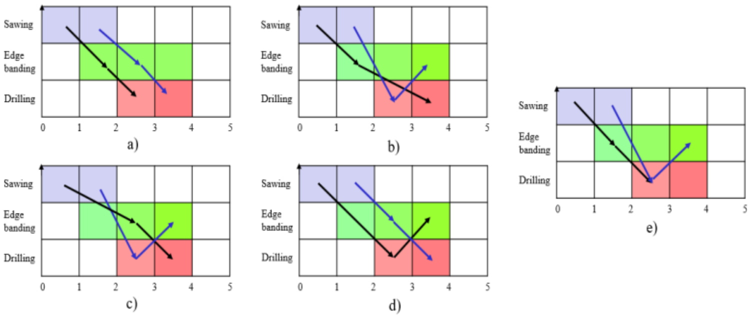


Figure 4: a) The alternative solutions of the case study represented in a Gantt diagram

at the appropriate time. For instance, for the operation _2_drilling_3, the input is a pre-cut laminated chipboard that must be available in the third time slot (_2_sawed_3_for_drilling).

In the final step of generation, the transitions between process steps are created to ensure that only valid transitions are included in the graph. These transitional operations connect the inputs and outputs generated in the previous step in such a way that the production time of an output has to precede its utilization as an input. For example, as a result of sawing performed in time slot 2, a cut-to-size laminated chipboard becomes available in the second time slot, which can then serve as input for edge banding starting in time slots 2 or 3. The resulting maximal structure obtained through these steps is shown in Figure 3a.

The resulting structure already contains all feasible solutions to the problem. However, some nodes do not belong to any valid solution according to the five axioms of PNS. These redundant nodes can be removed using the Maximal Structure Generator (MSG) algorithm. By solving the generated maximal structure, five feasible solution structures are identified, and their representations are shown in a Gantt chart in Figure 4.

By extending the P-graph model with mutually exclusive sets of operations, only solutions (a), (b), (c), and (d) remain feasible. However, an additional challenge arises: how can we incorporate into the objective function the preference for faster execution? A possible approach is to assign a cost to each operation that is negligible compared to the value of the final product. This cost allocation ensures that earlier activities incur lower costs. With these settings, the solver identifies structure (a) as the optimal solution, and its corresponding solution structure is illustrated in Figure 3b

3.3 Modeling by TCPNS

In the case of TCPNS, the structural elements of the P-graph model remain unchanged; however, a different terminology is recommended compared to the original framework. Specifically, raw materials are referred to as

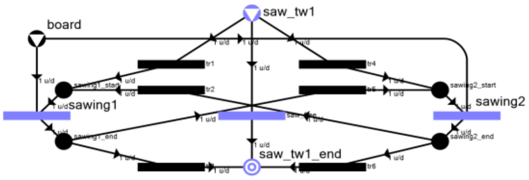


Figure 5: Modeling saw utilization within a time window for two batches

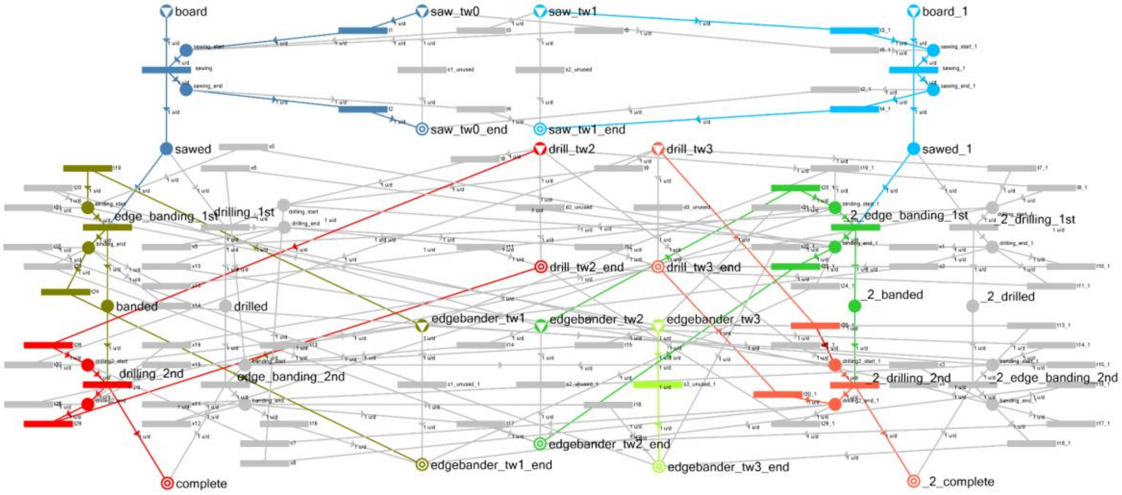


Figure 6: The optimal solution of the TCPNS model generated for the case study

resources, operational units as activities, and products as goals. During the modeling process, resources are consumed by the system to execute individual activities. These resources may represent equipment units (e.g., saws or drills) or other limited inputs required for a specific activity (e.g., laminated chipboard).

Since each activity can be assigned both an earliest start time and a latest completion time, it is sufficient to represent each time window with a single activity, rather than subdividing it into multiple time slots. For instance, in the case of the edge banding operation, a single activity is adequate to represent the three-time-unit window, with an earliest start time of 1 and a latest completion time of 4. This approach significantly reduces the number of generated activities.

However, due to the flexible nature of the recipe, the drilling and edge banding operations must still be duplicated to account for the two possible execution sequences. To ensure that the equipment can process both batches, transitions are designed to accommodate both sequences. Precedence relationships between batches are defined to limit the number of possible combinations, but these constraints would otherwise exclude several feasible solution structures. Therefore, all possible combinations are retained within the model. The scenario in which the equipment remains idle during a time window is represented by a separate operation. This operation does not involve actual processing but enables the equipment to revert to its original (non-subcontracted) state by the end of the time window, marking a transition back to its primary function.

For each processing operation, an input is defined to indicate the availability of the required equipment for execution (e.g., for sawing1, the input is sawing1_start). Similarly, an output is specified to represent when the equipment is released and becomes available for subsequent activities (e.g., sawing1_end). Consequently, the sawing operation for two batches can be modeled using the substructure illustrated in Figure 5.

The saw is represented as a shared resource available for both batches (sawing1 and sawing2). The second sawing operation can either be performed after the first one, or the saw can be transitioned to its target state.

This modeling approach ensures that all previously described combinations are accounted for. By applying the modeling approach presented in Figure 5, the maximal structure is generated for the sawing, drilling, and edge banding operations across all their respective time windows.

Following the solution of the model, the set of solutions depicted in Figure 4 is reproduced, while Figure 6 presents the solution structure corresponding to the previously identified optimal solution. The resulting structure, shown in Figure 6, employs color coding consistent with that used in Figure 2.b.

4. Results and Discussion

The examination of two recently published approaches of PNS (Discrete Time) and TCPNS (Precedence-Based) approaches highlights fundamental differences in their computational characteristics and modeling capabilities; see Table 1. The PNS method offers very fast computation, making it highly suitable for quick planning tasks; however, its scheduling precision is only low to moderate, and its model complexity, i.e., the number of decision variables, increases drastically by improving the precision, see cases 1 and 2. In contrast, the TCPNS approach provides high scheduling precision, though its computational speed ranges from moderate to slow due to the increased complexity of precedence-based modeling.

When it comes to changeover modeling, PNS applies a simplified representation, whereas TCPNS incorporates explicit and detailed modeling, enabling more accurate handling of transitions between operations. Similarly, batch scheduling is only limited in PNS, while TCPNS fully supports this functionality. From the perspective of real-time flexibility, PNS is characterized by low adaptability, while TCPNS is highly flexible, making it better suited for dynamic and real-time scheduling scenarios. However, for quick planning tasks, PNS is generally considered excellent, whereas TCPNS is less suitable due to its computational overhead.

Table 1: Comparison of different solution approaches for the case study

Id	*Problem formulation*	*Model class*	*Number of decision variables*	*Time precision*	*Computation time*	*Optimal value: Number of products completed*
1	PNS	Discrete time resource flow	72	300 s	0.0004 s	2.00
2	PNS	Discrete time resource flow	377	60 s	0.022 s	2.00
3	TCPNS	Precedence-based scheduling	95	0.00001 s	1,422.321 s	2.00

5. Conclusions

This study demonstrated the applicability of two P-graph-based optimization methods for reducing per-unit energy demand in furniture manufacturing by increasing the utilization of production time. The comparison of recently published model formulations highlighted that the Discrete Time Process Flow Model (PNS) offers rapid solutions with manageable complexity, making it suitable for high-level scheduling and systems with low variability. In contrast, the Time-Constrained Process Network Synthesis (TCPNS) enables detailed, real-time optimization, accommodating complex equipment constraints and changeovers. While TCPNS provides a more realistic representation of the factory floor, it requires significantly greater computational resources. Depending on the operational context—whether early-stage planning or fine-tuned scheduling—either approach may be more appropriate. Future research will explore hybrid strategies that combine the strengths of both models.

References

Bertók B., Heckl I., 2016, P-graph Studio: Software for Optimal Process Synthesis. P-graph.org. <http://p-graph.org/downloads/>, accessed 09.10.2025.

Friedler F., Orosz Á., Pimentel Losada J., 2022. P-graphs for Process Systems Engineering: Mathematical Models and Algorithms. Springer International Publishing, Cham, Switzerland, DOI:10.1007/978-3-030-92216-0.

Frits M., Bertók B., 2020, Scheduling Custom Printed Napkin Manufacturing by P-graphs. Computers & Chemical Engineering, 141, 107017.

Kalauz K., Süle Z., Bertók B., Friedler F., Fan L.T., 2012, Extending Process-Network Synthesis Algorithms with Time Bounds for Supply Network Design. Chemical Engineering Transactions, 29, 1189–1194.

Kalauz K., Bertók B., Frits M., 2024, Scheduling Manufacturing with Flexible Recipes to Maximize the Utilization of Renewable Energy. Chemical Engineering Transactions, 114, 955-960.

Kasapidis G.A., Dauzère Pérès S., Paraskevopoulos D.C., Repoussis P.P., Tarantilis C.D., 2023, On the multiresource flexible job shop scheduling problem with arbitrary precedence graphs. Production and Operations Management, 32(7), 2322-2330.

Lim L.K., Ab Muis Z., Hashim H., Ho W.S., Chee W.C., 2022, Electric bus charging schedule for multiple route via mathematical modelling. Chemical Engineering Transactions, 97, 151-156.

Lim L.K., Ab Muis Z., Hashim H., Ho W.S., Chee W.C., 2023, Impact of Battery Price on Electric Bus Charging Scheduling and System. Chemical Engineering Transactions, 106, 757-762.

Markowski M., Trzcinski P., Markowska D., 2022, The Non-Random Method of Cleaning Schedule Optimization for Heat Exchangers in a HEN. Chemical Engineering Transactions, 94, 1243-1248.