



PHYSICS-BASED SIMULATION FOR CONSTRUCTION ACTIVITY SEQUENCE PLANNING

M.R. Karim^{1*}, B.M. Gue² and Y. Mohamed¹

¹ Department of Civil and Environmental Engineering, University of Alberta, Edmonton, AB, Canada

² PCL Industrial Management Inc., Edmonton, AB, Canada

ABSTRACT: Construction activity sequence planning ensures the structured execution of construction processes by defining task order, dependencies, and constraints. While traditional planning methods rely on predefined templates, fragnets, or automated approaches using large language models, machine learning, and 4D building information modeling (BIM), these methods often struggle to address site-specific and real-time constructability challenges. To address these challenges, this study introduces a physics-based simulation approach that automates sequencing by evaluating structural dependencies and spatial constraints. In this study, a 3D BIM model is transformed into a physics-simulated environment using universal robot description format files, where a brute-force search iteratively refines sequences based on stability and path clearance constraints. To confirm the method's ability to dynamically generate feasible sequences that can subsequently be structured into construction schedules, testing is conducted on an industrial module comprising steel frames, pipe spools, and cable trays. Notably, as structural complexity increases, computational demands grow exponentially, demonstrating the limitations of brute-force search. The results indicate that physics-based simulation effectively validates constructability but requires optimization for scalability. Therefore, future advancements in this area should focus on AI-driven sequence optimization, improved BIM data integration, and distributed computing to enhance efficiency. By bridging the gap between digital models and real-world constraints, this study offers a novel method for advancing automated construction sequencing, making it more practical for industry-wide adoption.

1. INTRODUCTION

Construction activity sequence planning is a structured approach that organizes the execution of construction activities by defining their order, dependencies, and constraints, ensuring a systematic workflow for project implementation (Diego et al., 1991). The process begins with the development of a Work Breakdown Structure (WBS), which is a hierarchical framework that divides the project into deliverables—or phases—and work packages (Wang and Rezazadeh Azar, 2019). Activities are further derived from the work packages and specify the actions necessary for each work package, with a clearly defined job to be done with durations, as well as resource allocations (Baker and Trietsch, 2009).

To derive these activities, a planner depends on predefined fragnets, standardized work templates, or automated methods that use natural language processing (NLP), large language models (LLMs), and machine learning (ML); utilizing historical data, pattern recognition, and predictive modeling, complemented and verified by experience-based judgment (Amer et al., 2021). However, activity identification does not follow a fixed rule—or universal standard—and varies across a wide range of project types, industry standards, and planning methodologies (Fischer and Aalami, 1996). The level of detail and granularity required for defining construction activities is also influenced by various project-specific factors such as

project complexity, construction methods, stakeholder requirements, and scheduling levels. Since the primary goal of construction planning is to transform designs into a physical structure, every scheduled activity is inherently linked, either directly or indirectly, to the realization of specific structural components. While not a widely adopted approach, construction activities can be systematically generated at the component level using a bottom-up method.

When sequencing generated construction activities to establish the precedence network for scheduling, four key factors are to be considered: (1) the physical relationships between structural components, (2) interactions among construction trades, (3) the need for interference-free object movement onsite, and (4) code and safety regulations that dictate execution order. However, as construction planning is inherently focused on translating design into a physical structure, activity sequencing is predominantly influenced by the physical interdependence among components, making it the most dominant out of these four factors and forms the foundation of the sequencing framework. (Baker and Trietsch, 2009; Diego et al., 1991).

The established precedence network—or project network diagram—is a fundamental tool in the Critical Path Method (CPM); CPM is used to identify and optimize the most efficient sequence of construction activities by utilizing critical path analysis (Bagshaw, 2021; Russell et al., 2009). However, CPM relies on predefined logical relationships, requiring planners to manually assign dependencies rather than having them automatically determined. To address this limitation, researchers have explored various approaches for automating dependency development. For instance, NLP and LLMs analyze unstructured textual data, such as project specifications and descriptions, to extract activities and infer logical sequences based on linguistic patterns and predefined rules (Singh et al., 2023). ML models also further enhance automation by using historical project data to predict task dependencies and durations. Moreover, when integrated with building information modeling (BIM), ML-based methods facilitate data extraction, activity generation, and automated sequencing, creating a more systematic approach to schedule development (Al-Sinan et al., 2024; Amer et al., 2021; Weldemihret Weldu, 2016).

Despite these advancements, significant limitations persist. Indeed, most current methods rely on large ML-based historical datasets of similar projects or predefined sequencing rules based on BIM, which often fail to capture project and site-specific constructability challenges, interference, and path obstructions. In this context, researchers have utilized 4D BIM simulations, aiming to identify schedule conflicts generated from CPM-based methods. These simulations primarily function as visualization tools, though, rather than dynamically generating or adjusting sequences that consider actual structural and environmental constraints in real-time (de Vries and Harink, 2007; Kim et al., 2013; Koo and Fischer, 2000; Staub-French et al., 2008).

With the rapid increase in computational capabilities, a physics-based simulation provides a novel approach in the autonomous generation of construction sequences within a virtual physical environment while ensuring constructability through real-world physical constraints (Chakrabarty et al., 2024; Xu et al., 2024). This methodology continuously refines the execution order of activities by assessing structural dependencies, potential collisions, force interactions, stability, support dependencies, and spatial clearance, resulting in feasible and interference-free construction sequence. By modeling activities as the actions of physical bodies, the system can iteratively derive an executable sequence from any arbitrary input order (Liu and Negrut, 2024; Tian et al., 2023; Wang et al., 2022), fully automating the basic framework of the sequence generation process. The iterative process is executed using brute-force search. Brute-force search is a straightforward problem-solving approach that systematically examines all possible solutions through exhaustive trial-and-error until it finds a valid one, without utilizing heuristics or efficient search strategies (Bernstein, 2005). While effective, this method becomes highly time-consuming as structural complexity increases, requiring significantly more time with each additional iteration. However, successful execution of this model could enable intelligent agents to learn from these simulations and apply this knowledge to plan activity sequencing for more complex projects. This paper presents the methodological framework for developing a physics-based simulation approach to automate construction activity sequencing, detailing the transformation of a 3D BIM model into a physics-simulated environment. It then discusses the implementation process, which employs a brute-force search method for sequence generation. A case study on an industrial module is conducted to evaluate the effectiveness of the proposed

approach, followed by an analysis of computational challenges, potential optimizations, and future research directions.

2. OBJECTIVE AND METHODOLOGICAL FRAMEWORK

The primary objective of this study is to develop a feasible construction activity sequencing methodology using a physics-based simulation approach, wherein a brute-force search iterates through activities, retaining valid ones progressively until a set of executable sequences is generated.

To achieve this, the study identifies construction activities in direct correlation with their respective structural components, mapping each activity to a specific structural section or component. The 3D structural model—augmented with BIM data—is converted into its corresponding rigid body virtual representation with original geometric and material properties essential for simulation within a physics-based simulation environment. To develop the physics-based simulation environment, a physics engine is employed and configured to account for real-world physical constraints, gravity, collisions, spatial limitations, rigidity, base platform, and the connectivity behavior of structural components. This environment serves as a testing platform for evaluating the feasibility of executing any activity to be implementable within practical constructability requirements.

Finally, an arbitrary or rudimentary rule-based organized sequence of activities passes through the physics engine-generated environment. A brute-force search algorithm is employed, whereby the system iteratively evaluates and compiles feasible activity sequences. This process continues until a complete and viable sequence is identified, ensuring that the structure is constructed in full accordance with its physical specifications displayed in its 3D model.

3. IMPLEMENTATION OF THE MODEL

3.1 Construction Algorithm

A construction algorithm encapsulates construction knowledge, which can be categorized into structural behaviour knowledge and construction method knowledge (de Vries and Harink, 2007). Structural behaviour is dictated by physical laws such as gravity, load distribution, force interactions, joint configuration, and material properties, while construction methods are influenced by material type, assembly techniques, construction equipment, regulatory codes, and safety considerations.

In this research, two fundamental considerations are adopted to address structural behaviour and construction methods. For structural behaviour, when considering gravitation, it is essential that any construction activity depicting the installation or construction of a physical component of the structure must be supported by a base or another component representing a construction activity within the sequencing process. Support may be provided in various forms including simple supports, which allow for the structure to rest on a surface, fixed joints, which require a single support point, and hinge supports, which necessitate two support points (Diego et al., 1991). In the case of construction methods, it is assumed that components are prefabricated according to their specifications and must be placed by equipment in their designated positions, following a predefined path. Afterward, connections are made by tradesmen. Therefore, each construction activity must have a clear installation path to be executed and, when tradesmen from two different disciplines are required to work in the same area, a time gap must be provided to avoid conflicts.

In practical applications, construction methods can vary widely, leading to multiple valid construction sequences. However, all these sequences must adhere to structural stability, physical dependencies, and spatial constraints.

3.2 Methodology

In this study, a simplified methodology is developed, as shown in Figure 1, to generate an automated construction sequence and schedule from a 3D BIM model using the PyBullet physics simulation engine

(Yang et al., 2021). The simulation engine requires that component entities be in universal robot description format (URDF). Therefore, module elements such as steel structures, columns, beams, and pipe spools must be converted into URDF before integration into the simulation environment. URDF files, which are XML-based, define the physical and visual properties of structures including components such as joints, links, and their relationships. These files are widely used in robotics simulations such as Robot Operating System (ROS) and physics engines like PyBullet (Yang et al., 2021). In this model, both the geometric and metadata files of the structural components are imported from a 3D BIM model. The 3D geometric representation is obtained through .obj files, while additional construction-related metadata—such as installation time, material cost, and fabrication cost—that aren't readily available in the BIM data are hypothetically formulated and extracted from separate .csv files. If available, these datasets can also be obtained from an Industry Foundation Classes (IFC) file of the structure and integrated into a physics engine. The extracted data for each component is then converted into a modified URDF file. In the physics simulation engine, the primary input is an arbitrary sequence of structural components: a list of URDF files. The first stage of the simulation involves a stability check, where the physics engine evaluates whether each component, as it appears in the sequence remains structurally stable upon placement.

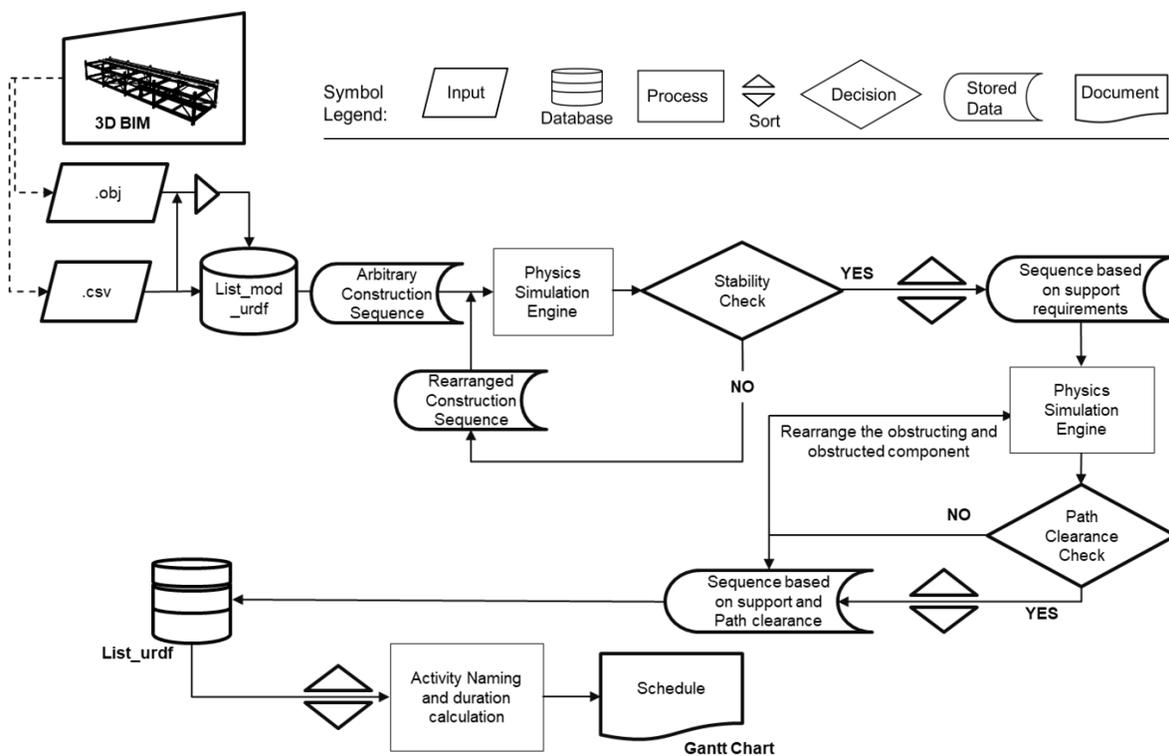


Figure 1: Process flow of activity sequencing and schedule generation using physics-based model

If a component meets stability criteria, the sequence moves forward; if not, the component file is dynamically repositioned to the next available spot in the queue. The iteration continues until all components are ordered according to structural support requirements. A support-based sequencing process ensures that the installation order adheres to load-bearing relationships, preventing components from being placed before receiving sufficient structural support. Once stability is achieved, a path clearance check ensures that each component can move unobstructed from its loading position to its designated placement. If an obstruction is detected, the system dynamically adjusts the sequence by repositioning the obstructing component to the end of the queue and re-evaluating the installation order until a clear path is established.

Once stability and path clearance are validated, a physically viable sequence of URDF files is then generated. The subsequent process of this model focuses on transforming the generated sequence into a format suitable for traditional scheduling methods. Here, a process is developed to assign each construction activity a unique label, derived from the corresponding physical component. This label includes details such

as material type, global position, and installation level. The system groups listed components based on their trade application similarity or other predefined criteria, assigns a general activity name, and calculates the duration of each activity using metadata embedded in the modified URDF files. Finally, the validated construction sequence is organized into a Gantt chart to visualize the schedule. During the implementation of this process, key performance metrics such as execution time and iteration count are monitored to assess computational requirements of the simulation. These metrics provide an overview of the system's performance, offering an understanding of the efficiency of the simulation in generating construction sequences based on the model, as well as how the model performs as complexity increases.

3.3 Mathematical Modeling for Brute-Force Search Performance in Sequence Generation

A structure with N activities, each represented by a component or group of components, requiring a defined construction sequence. The sequencing process is governed by two key constraints: support requirement and installation path clearance. Support requirement warrants that a component can only be placed after its necessary supporting components are constructed to maintain structural stability. Here, $S(z_i) \geq S_{min}$, where $S(z_i)$ represents the total support provided by completed components beneath z_i , and S_{min} is the minimum required support threshold. If this condition is not met, component z_i cannot be placed.

The installation path clearance constraint ensures that there is an unobstructed space for the placement of a component. This condition is represented as $C(z_i) = 1$, where $C(z_i)$ is a binary function indicating whether a valid installation path exists (1 for clear, 0 for obstructed). If $C(z_i) = 0$, the sequence must be iterated again, placing the obstructed component before the obstructing component, and requires reordering and re-evaluation of all remaining components to establish a feasible sequence. A valid construction sequence $S = (S_1, S_2, \dots, S_N)$ must satisfy both constraints for all components, ensuring that each component is placed in a physically feasible and interference-free arrangement (Bernstein, 2005). In an unconstrained scenario, the number of possible sequences follows factorial growth represented by (Eq. 1):

$$[1] \Omega_{unconstrained} = N!$$

This expression accounts for all possible permutations of the N components. However, in practical construction scenarios, each component z_i is subject to structural stability requirements and installation path clearance conditions. These constraints significantly reduce the number of feasible sequences, which can be expressed as (Eq. 2):

$$[2] \Omega_{constrained} = \frac{N!}{\prod_{i=1}^N k_i! + \prod_{i=1}^N c_i!}$$

The parameters k_i and c_i represent the number of immediate predecessor components required for structural stability before placing z_i and the number of components obstructing its path clearance, respectively.

Since both support and clearance constraints impose order dependencies on component placements, the number of feasible sequencing options decreases as the values of k_i and c_i increase. Accordingly, the reduction in valid sequences follows an inverse proportionality (Eq. 3):

$$[3] \Omega_{constrained} \propto \frac{1}{\prod_{i=1}^N k_i! + \prod_{i=1}^N c_i!}$$

The reduction in valid sequences significantly impacts computational search methods, particularly for brute-force. In a constrained scenario, the worst-case computational effort required to obtain a valid sequence through brute-force search is (Eq. 4):

$$[4] T_{brute-force} = N! - \Omega_{constrained}$$

In extreme cases where constraints are highly restrictive, such as when every component has a predetermined order of placement dictated by both structural and clearance constraints, the number of valid sequences may be reduced to a single feasible sequence, i.e., $\Omega_{constrained} = 1$.

From Eq. 4, it is evident that the computational effort increases factorially with N . Since brute-force search exhaustively tries all possible options, the constrained search space can lead to substantial computational inefficiencies. When constraints can be grouped into subgroups based on support dependencies and clearance requirements, the number of valid sequences is not merely adjusted by a factorial term but is further reduced by redefining the number of independent elements requiring sequencing. If p represents the number of constraint pairings or groups of interdependent components, then the effective number of independent elements to be sequenced is reduced from N to N' as shown in Eq. 5:

$$[5] N' = N - \sum_{j=1}^p (g_j - 1)$$

Here, g_j denotes the size of the j^{th} grouped set of interdependent components, and the subtraction reflects that, within each group, components follow a fixed internal sequence and do not contribute to independent permutations. Finally, the worst-case effort required to find a valid sequence is reduced to (Eq. 6):

$$[6] T_{grouped} = N'! - \Omega_{constrained}$$

Replacing $N!$ with $N'!$ (in Eq. 4 to obtain Eq. 6) in the brute-force search space reduces factorial growth, lowers unsuccessful search attempts, and decreases computational effort. Since factorial growth is super-exponential, even a small reduction in N saves significant computation.

4. CASE STUDIES AND RESULTS

The test case for applying the model involves generating a construction activity sequence and schedules for an industrial module as illustrated in Figure 2, from .nwd format (Navisworks) file—to be fabricated at a module fabrication yard. This module consists of three key structural components, steel frames, pipe spools, and electrical cable trays, each requiring installation by different trades.

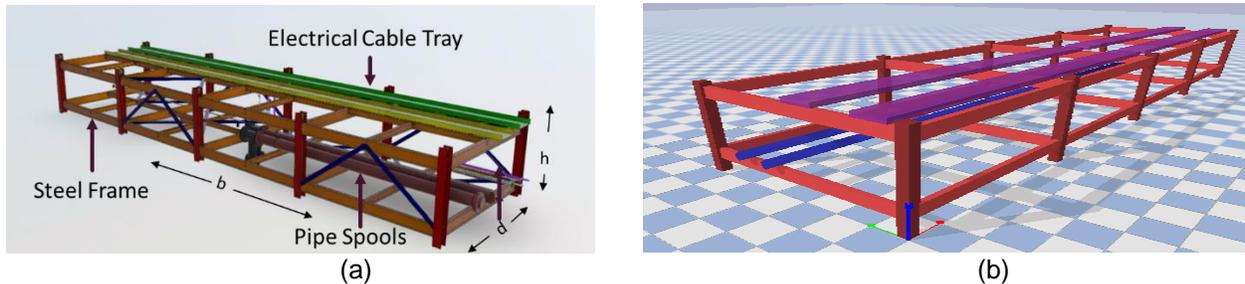


Figure 2: (a) A 3D BIM of industrial module, and (b) 3D URDF representation of it in physics engine

In practice, the construction process follows a hierarchical assembly sequence to ensure structural stability and feasible installation. Assembling steel structures, pipe spools, and cable trays adheres to a well-defined sequence to maintain constructability and efficiency. The process begins with erecting the steel frame, which serves as the primary support for subsequent installations. Columns are first placed on prepared foundations or fixed support points, ensuring vertical alignment and securing them in place. These columns act as the primary load-bearing elements and must be fully stabilized before proceeding. Once the columns are set, horizontal beams are installed, connecting the columns to form a rigid frame. The connections between columns and beams are secured using either bolted or welded joints, depending on the design. The steel frame assembly follows a bottom-up approach, completing each level before advancing to the

next. Bracing elements may also be added to enhance stability, particularly in taller or more complex structures.

After constructing the steel frame, pipe spools are installed. Their placement depends on the steel frame, as pipes require support from structural beams or dedicated pipe racks. Hangers and brackets are attached to the steel frame to secure the pipes. Following pipe installation, cable trays are mounted to create structured pathways for electrical wiring and instrumentation cables. These trays are typically supported by the steel frame, either attached directly to beams or secured using secondary support brackets. Since pipe spools and cable trays are assumed to be placed from above using crane support, their installation paths must remain unobstructed. Therefore, any upper-level steel beam should be installed only after the lower-level pipes or cable trays are positioned.

In a physics-based simulation model, the initial construction sequence is arbitrary. From this starting point, a viable sequence is iteratively generated based on stability constraints, support dependencies, and path clearance requirements. The model dynamically evaluates various placement possibilities and determines a feasible sequence where all components remain stable under gravity and structural forces while ensuring unobstructed installation paths. Each component, for example, column, beam, pipe spool, etc., is represented as a discrete entity using URDF files. These files define the component's geometry, mass connection behavior etc., enabling the simulation to accurately model physical interactions and constraints.

The simulation begins by placing components in their designated positions. If a component lacks the necessary structural support after placement, such as a beam being placed before its supporting columns, it is deemed unstable and moved to the end of the queue for repositioning in the sequence. Similarly, if the installation of a component is obstructed by an already placed element, the obstructed component is placed ahead of the obstructing component; the process continues until a clear installation path for all components is available. This iterative process refines the sequence until all components are positioned in a structurally sound manner without interference. In the first iteration of the simulation, columns are accepted first as the physical environment provides the necessary base support for their erection. Once stable columns are in place, beams are accepted next, forming a structural framework that supports additional elements. After the steel frame is sufficiently developed, the simulation evaluates when pipe spools can be installed, ensuring they are properly supported by beams or designated pipe racks. Likewise, cable trays are placed only when appropriate structural supports are available. Since pipe spools and cable trays are installed from above, the simulation verifies path clearance to prevent obstructions during placement. When higher-level steel beams block the positioning of pipe spools and cable trays, the installation of these beams is deferred until the pipe spools and cable trays are securely in place. Throughout the process, the model autonomously adjusts the sequence based on real-time stability validation and path clearance, preventing unsupported, unstable, or obstructed placements.

Table 1: Computational time for sequence generation across module sizes

Module Name	Size (d x b x h)	Total number of components	Time to generate structurally stable sequence (sec)	Time to generate installation path cleared sequence (sec)	Total time (sec)
Module_1	4x1x1	41	0.36	0.69	1.05
Module_2	4x1x2	82	2.27	2.81	5.08
Module_3	4x2x2	138	6.66	10	16.66
Module_4	4x2x3	207	21.63	23.71	45.34
Module_5	4x3x3	291	44.09	49.33	93.42
Module_6	4x3x4	388	90.14	92.31	182.45
Module_7	4x4x4	500	152.17	160.42	312.59

Seven module sizes varying in complexity based on number of horizontal bays (*d*), longitudinal bays (*b*), and vertical levels (*h*), are simulated, as shown in Table 1. As the number of components of the modules increases, sequencing becomes more complex, revealing the limitations of brute-force search. The results

in Table 1 illustrate how growing complexity affects sequencing efficiency and computational feasibility. A higher component count leads to a significant rise in computational time, highlighting the challenges of applying brute-force searches to large-scale construction scheduling.

Furthermore, Figure 3 illustrates the timing variation in the module structure as the number of components increases. As the number of components increases, the computational time per component grows significantly, demonstrating the exponential nature of brute-force search used for sequencing. When the component count in the structure is low (41 components), the average time required per component is 0.0256 seconds. However, as the structure becomes more complex with 500 components, the time per component increases drastically to 0.6252 seconds. This indicates that, as complexity rises, the sequencing process becomes more computationally demanding. In fact, the time per component in the high-complexity scenario is approximately 24 times higher than in the low-complexity case, highlighting the limitations of brute-force search methods. This sharp increase in computational demand confirms the necessity for more efficient sequencing algorithms, such as artificial Intelligent (AI)-driven optimization techniques to improve scalability and performance in large-scale construction sequencing.

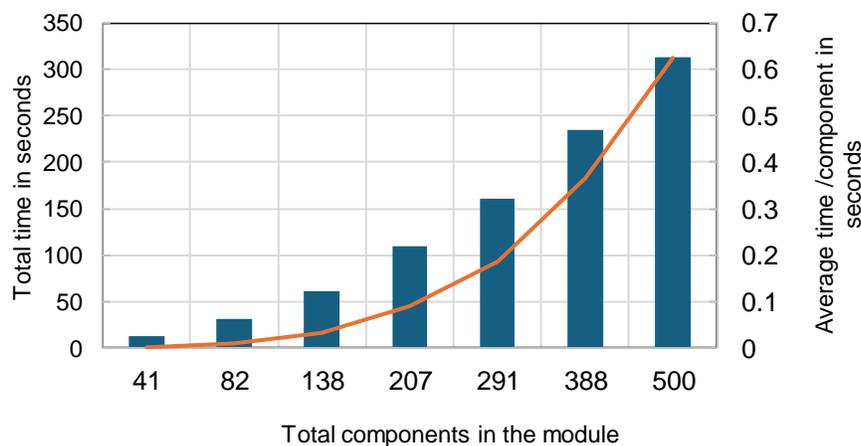


Figure 3: Computation time vs. increased complexity (number of components)

5. CONCLUSIONS

Physics-based simulation offers a physics-informed framework for construction sequencing, integrating stability analysis, support dependencies, and spatial constraints to ensure constructability. By dynamically adjusting sequences based on support requirements and force interactions among structural components, this method can automatically generate constructible sequences, minimizing dependence on heuristic-based planning.

When compared to 4D simulation, which integrates time as the fourth dimension within BIM for visualization, physics-based simulation offers distinct advantages. While 4D simulation excels in project visualization, trade coordination, and stakeholder communication, it does not inherently validate constructability. Physics-based simulation, on the other hand, ensures that components are placed in a structurally viable manner, making it particularly useful for complex load-bearing structures, temporary stability conditions, and dynamic site constraints. This ability to integrate real-world physics into sequencing decisions makes it a more robust alternative for ensuring stability and safety during construction.

Although physics-based simulation offers numerous advantages, there are also several challenges that must be addressed to improve its practical applicability in construction sequence. For instance, modern BIM frameworks primarily function as geometric representations of structures and often lack explicit definitions for many structural parameters such as load-bearing relationships, joint configurations, and force interactions. This limitation complicates their direct integration with physics-based validation methods, as

the absence of these essential attributes demands extensive pre-processing and data augmentation. Converting a standard 3D BIM model into a simulation-ready format that accurately represents physical interactions and constraints remains a significant challenge. Overcoming these barriers requires interdisciplinary advancements in computational modeling, data integration, and automation to bridge the gap between BIM and physics-based simulations, ultimately enhancing the accuracy and efficiency of construction sequencing methodologies. Moreover, as project scale increases, computational complexity grows exponentially. Larger and more complex structures demand significantly greater computational resources. Mathematical modeling indicates that generating a near-valid sequence significantly reduces the computational load of physics iterations, thereby decreasing overall simulation time. This observation suggests that introducing intelligent agents can optimize sequence generation by reducing redundant computations and improving efficiency.

Therefore, future research should focus on integrating AI techniques to enhance sequence optimization and scalability. Additionally, improving BIM data representation by incorporating force interaction models, dynamic stability parameters, and construction constraints would facilitate unified integration with physics-based simulations, bridging the gap between digital design and real-world construction processes. Finally, utilizing federated simulation and distributed processing can mitigate computational limitations, enabling real-time feedback and rapid scenario analysis for complex construction projects. These advancements would collectively enhance the applicability of physics-based simulations, making them more practical and scalable for industry-wide adoption.

ACKNOWLEDGMENTS

Sincere appreciation to PCL Industrial Management Inc., the Construction Innovation Center (CIC) of the University of Alberta, and NSERC for their contributions in data provision and research funding for conducting this study.

REFERENCES

- Al-Sinan, M.A., Bubshait, A.A., Aljaroudi, Z. 2024. Generation of Construction Scheduling through Machine Learning and BIM: A Blueprint. *Buildings*, 14, 934.
- Amer, F., Koh, H.Y., and Golparvar-Fard, M. 2021. Automated Methods and Systems for Construction Planning and Scheduling: Critical Review of Three Decades of Research. *Journal of Construction Engineering and Management*, 147(7).
- Bagshaw, K.B. 2021. New PERT and CPM in Project Management with Practical Examples. *American Journal of Operations Research*, 11(4): 215-226.
- Baker, K.R., and Trietsch, D. 2009. *Principles of Sequencing and Scheduling*. Wiley, Hoboken, NJ, USA.
- Bernstein, D.J. 2005. Understanding Brute Force. *Ecrypt Stvl Workshop on Symmetric Key Encryption*, 10-19.
- Chakrabarty, A., Vanfretti, L., Tang, W.T., Paulson, J.A., Zhan, S., Bortoff, S.A., Deshpande, V., Wang, Y., and Laughman, C.R. 2024. Assessing Building Control Performance Using Physics-Based Simulation Models and Deep Generative Networks. *2024 IEEE Conference on Control Technology and Applications (CCTA)*, 547-554.
- Diego, E., Ibbs, C.W., and Kim, S. 1991. Sequencing Knowledge for Construction Scheduling. *Journal of Construction Engineering and Management*, 117(1): 118-130.
- Fischer, M.A., and Aalami, F. 1996. Scheduling with Computer-Interpretable Construction Method Models. *Journal of Construction Engineering and Engineering and Management* 122 (4): 337-47.
- Kim, H., Anderson, K., Lee, S., and Hildreth, J. 2013. Generating Construction Schedules Through Automatic Data Extraction Using Open BIM (Building Information Modeling) Technology. *Automation in Construction*, 35: 285–295.
- Koo, B., and Fischer, M. 2000. Feasibility Study of 4D CAD in Commercial Construction. *Journal of Construction Engineering and Management*, 126(4): 251-260.
- Liu, C.K., and Negrut, D. 2024. The Role of Physics-Based Simulators in Robotics. *Robotics and Autonomous Systems Annu. Rev. Control Robot. Auton. Syst.*, 51: 28.

- Russell, A., Staub-French, S., Tran, N., and Wong, W. 2009. Visualizing High-Rise Building Construction Strategies Using Linear Scheduling and 4D CAD. *Automation in Construction*, 18(2): 219–236.
- Singh, A.K., Pal, A., Kumar, P., Lin, J.J. and Hsieh, S.-H. 2023. Prospects of integrating BIM and NLP for automatic construction schedule management. *Proceedings of the 40th International Symposium on Automation and Robotics in Construction*.
- Staub-French, S., Russell, A., and Tran, N. 2008. Linear Scheduling and 4d Visualization. *Journal of Computing in Civil Engineering*, 22(3): 192–205.
- Tian, Y., Willis, K.D.D., Al Omari, B., Luo, J., Ma, P., Li, Y., Javid, F., et al. 2023. ASAP: Automated Sequence Planning for Complex Robotic Assembly with Physical Feasibility. *IEEE International Conference on Robotics and Automation (ICRA)*.
- Vries, B. de, and Harink, J.M.J. 2007. Generation of Construction Planning from a 3D CAD Model. *Automation in Construction*, 16(1): 13–18.
- Wang, J., Li, Y., Gao, R.X., and Zhang, F. 2022. Hybrid Physics-Based and Data-Driven Models for Smart Manufacturing: Modelling, Simulation, and Explainability. *Journal of Manufacturing Systems*, 63(April): 381–391.
- Wang, Z., and Rezazadeh Azar, E. 2019. BIM-based Draft Schedule Generation in Reinforced Concrete-Framed Buildings. *Construction Innovation*, 19(2): 280–294.
- Weldu, W.Y. 2016. Automated Generation and Visualization of Initial Construction Schedules from Building Information Models. *Graduate School Dissertations*, Louisiana State University.
- Xu, L., Veeramani, D., and Zhu, Z. 2024. Automated Physics-Based Modeling of Construction Equipment Through Data Fusion. *Automation in Construction*, 168(PB): 105880.
- Yang, X., Ji, Z., Wu, J., Lai, YK. (2021). An Open-Source Multi-Goal Reinforcement Learning Environment for Robotic Manipulation with Pybullet. *Towards Autonomous Robotic Systems*. Lecture Notes in Computer Science, vol 13054. Springer, Cham.