Authors' version of: Li, X. and Wu, C. and Wu, P. and Xiang, L. and Shen, G.Q. and Vick, S. and Li, C.Z. 2019. SWP-enabled constraints modeling for on-site assembly process of prefabrication housing production. Journal of Cleaner Production. 239: 117991.
DOI: 10.1016/j.jclepro.2019.117991

SWP-enabled Constraints Modeling for On-site Assembly Process of Prefabrication Housing Production

Abstract

Prefabrication housing production (PHP) processes are fragmented and full of variability. Their schedule reliability is particularly disturbed by the constraints deriving from task executions in the on-site assembly process. Proactive constraints modeling, including identifying constraints and understanding their interrelationships, is crucial to ensure successful task executions and enhance sociability in collaborative working. However, current methods for constraints modeling are often sluggish and heavily rely on human's commitments because there is no real-time and value-added information for decision-making. To address this issue, this study proposes an approach of smart work packaging (SWP)-enabled constraints modeling service, which consists of three dynamic sub-services: social network analysis (SNA) service, hybrid system dynamics (SD)-discrete event simulation (DES) model service, and constraints scenario analysis service. It can equip the workers with the ability to (1) automatically identify the critical constraints, (2) dynamically explore interactional and interdependent relationships of these constraints, and (3) simulate and analyze the impact on schedule performance under different constraints scenarios. Five critical constraints are identified, including adverse weather conditions, lack of collision-free path planning, lack of visible and audible communication mechanism, lack of optimal buffer layout, and lack of optimal installation sequence. Most interrelationships are depicted in the four modules of the hybrid SD-DES model, including the assembly process, resource availability, operation efficiency, and schedule performance. Finally, the most influential constraint "lack of collision-free path planning" to schedule performance is identified in the constraints scenario analysis process.

Keyword

Constraints Modeling, Smart Work Packaging, System Dynamics, Discrete Event Simulation, Prefabrication Housing Production

1. Introduction

As reported by McKinsey (2017), construction-related spending accounts for 13% of the world's GDP, but productivity growth of the construction industry has only increased by 1% over the past 20 years. The productivity of the manufacturing industry is nearly 1.7 times higher than that of the construction industry. The underlying reason could be the relatively slow adoption and integration of advanced information technologies and industrialization principles such as mechanization, automation, robotics, standardization, modularization, and information-driven construction (Li et al., 2019). Prefabrication Housing Production (PHP) is an innovative solution in the construction industry. It uses the principles of industrialization in the lifecycle of construction projects, including design, manufacturing, transportation, on-site assembly, maintenance, and deconstruction stages. The benefits of PHP have been investigated in many studies. PHP can provide a safer and more sustainable construction environment by testing products in controlled factories using consistent standards (Wu et al., 2016; Wu et al., 2017). It can also help reduce construction waste (Mao et al., 2016; Wu et al., 2018). Moreover, widespread adoption of PHP in densely populated regions, such as Hong Kong, can be used to mitigate the impact of labor shortage and unbalanced housing supply and demand (Li et al. 2018a). Although the public high-rise residential buildings in Hong Kong have benefited significantly from PHP, the supply of public housing is still plagued by the pathological schedule delay of PHP. For example, the government planned to construct 13,300 flat units of public housing in the financial year of 2016-2017. However, the actual amount of PHP is only 11,276 units; a 15.22% delay (Housing Authority, 2018). The uncertainties and constraints in the fragmented PHP process have proven to be the dominant drivers (Li et al., 2016). Uncertainty refers to something that may occur, whereas constraint (e.g., limited space and buffers) is something that will happen (Li et al., 2017a). Constraints are the apparent bottlenecks and thus are more predictable than the uncertainties to be removed in task executions. As such, reliable constraint-free schedules are vital for achieving an industrialized PHP environment across the fragmented stages including design, manufacturing, logistics, and on-site assembly so as to avoid schedule delays and cost overruns (Wang et al., 2016a).

The reliability of PHP schedules can be enhanced via proactive constraints management, which is the process of identifying, optimizing and monitoring of bottlenecks (e.g., unavailable drawings and specifications, shortage of workforce and materials, limited workspace, uncompleted preceding works, lack of work permits, quality, and safety issues) to ensure that work package-level tasks assigned to workers can be successfully executed (Blackmon et al., 2011). Managing constraints in PHP processes means preparing more (e.g., on detailed and dynamic planning with lean solutions) and acting fast (e.g., on decision-making and collaborative working) using available information and knowledge. As such, the principal objective of constraints management is to continually improve the reliability of workflow by guaranteeing that precise information is always available at the right time in the right format to the right person. There have been a significant number of studies focusing on how to support decision makers and collaborative workers with precise, timely, and well-formatted information for task execution (Zhong et al., 2017; Li et al., 2018b). For example, an internet of things (IoT)-enabled Building Information Modeling (BIM) platform is developed with the support of smart construction objects (SCOs) by equipping objects with information and communication technologies such as radio frequency identification (RFID), augmented reality (AR), and other sensing and tracking technologies (Li et al., 2018c; Niu et al., 2016). Other studies, such as Blackmon et al. (2011) and Wang et al. (2016a), have made efforts to develop frameworks by considering the use of information technologies for constraints management in the oil and gas industry. However, there is so far no widely accepted approach for constraints management in PHP.

The development of smart work packaging (SWP) in recent years seems to be adequate to address the challenge. Work packaging is the approach to break down PHP processes into manageable pieces to facilitate execution of activities or tasks. However, it is limited in offering practical constraints management solutions such as automatic identification and analysis of constraints and their interrelationships (Hamdi, 2013; Isaac et al. 2017), real-time sensing and tracking constraints status (Liu et al. 2015), and optimal constraints improvement planning (Abuwarda and Hegazy, 2016). Smart Construction Objects (SCOs) (Niu et al., 2016) are the smart resources with characteristics of awareness, communicativeness, and autonomy, which can improve the capacity of resources-related constraints modeling, optimization, and monitoring. However, SCOs are defined on single construction objects, without encapsulating the construction project operations like work packaging. Thus, SWP, as the integration and extension of work packaging and SCO aims to develop smart tasks execution procedure to improve constraints management for achieving mass production in PHP. Smarter constraints management involves sophisticated autonomy, adaptivity, and sociability, based on the intensive interaction among people, technologies, environment, and resources. If this process fails, severe schedule delay/cost overrun may happen. In PHP, there are a few studies which investigate the smart transformation of a group of tasks (i.e. the lowest level in the work breakdown structure) based on the building systems of product breakdown structure (PBS) by embedding the capabilities of visualizing, tracking, sensing, processing, computing, networking, and reacting. The smart transformation centers upon autonomy, adaptivity, and sociability, which can facilitate better tasks execution from workers. For example, the machinery (i.e., vehicles, crane towers) can be augmented with the autonomy to transport or hoist the prefabricated products independently and without direct intervention from surroundings (Chi et al., 2012). In addition, the PHP planning approaches can be enhanced with adaptivity to be capable of reacting flexibly and resiliently through re-planning in a dynamic manner when constraints are not removed (Abuwarda and Hegazy, 2016). Work packages can also be strengthened with sociability to interact in a peer-to-peer manner with other work packages or resources in the work packages to collectively improve constraints management (Taghaddos et al., 2012).

Despite the merits of deploying SWP for constraints management in PHP to achieve a constraint-free environment, numerous implementation barriers have also been raised (Wang et al., 2016a). The first barrier for SWP that prevents the success of constraints management in PHP is to identify the critical constraints and understand the interrelationships of the constraints. To better explore the SWP-enabled constraints management system of PHP from a holistic view, constraints modeling, including identifying constraints and mapping their interrelationships, should be investigated before optimizing and monitoring. This study concentrates on the on-site assembly process of PHP due to the fact that it is the central piece for delivering the final product. This study also proposes a two-phase solution to model the constraints, which includes: (1) encapsulating social network analysis (SNA) module into SWP to automatically identify the trades associated constraints in the on-site assembly process, and (2) developing a hybrid dynamic model which integrates system dynamics (SD) and discrete event simulation (DES) to map the interactions and interrelationships of the constraints. The specific objectives of this study are to: (1) automatically identify the trades associated critical constraints; (2) dynamically explore interactional and interdependent relationships of these constraints; (3) simulate and analyze the impact of these constraints on schedule performance under various constraints scenarios.

2. Literature Review

2.1 Prefabrication Housing Production (PHP)

Schedule delay continually impedes the success of PHP due to the lack of required coordination to prevent work starvations between prefabrication factories, logistics, and on-site construction (Li et al., 2018a). The issue of fragmentation is amplified when the manufacturing work of PHP in Hong Kong has been completely shifted offshore, e.g., to the Great Bay Area (GBA) of Mainland China, which results in all uncertainties and constraints prior to tasks execution could not be timely satisfied to enhance and improve the reliability of PHP processes (Li et al., 2016). Previous studies investigated the stakeholder-associated risks in the whole PHP processes, such as low interoperability between different enterprise resource planning systems (ERPs), logistics information inconsistency, delivery delay of prefabricated products to the site (Li et al., 2016). To help reduce these risks, the internet of things (IoT)-enabled BIM platform, including the services of production, logistics, and on-site assembly, was developed to improve the visibility and traceability of prefabricated products for achieving just-in-time (JIT) coordination (Zhong et al., 2017; Li et al., 2018b). Meanwhile, data analytics methods, e.g., the hybrid simulation model, are also developed to facilitate risk identification and interrelationships mapping in the PHP processes (Li et al., 2018a). If the level of detail (LOD) in schedule can be classified into LOD 100 (master schedule), LOD 200 (phase schedule), LOD 300 (weekly schedule), and LOD 400 (daily work plan), previous outcome truly works in mitigating the risks to improve the phase schedule which is a LOD 200 covering each PHP phase. However, risks and constraints are different and must be identified and treated differently. Constraints can usually be identified, improved, and removed in a more detailed schedule (e.g., LOD 300 and LOD 400) (Wang et al., 2016a). For example, the detailed task or activity still beset some missing or incomplete prerequisites including design (drawings and BIM models), prefabricated products, space, buffer, labor, equipment, permits, specifications, prerequisite work, which prevent the reliability of PHP workflow, particularly in the on-site assembly process (Li et al., 2018b). This research concentrates on the development of constraints modeling service for the four-day assembly cycle (FDAC) process, which means the typical floor can be assembled and finished by the four-day plan, as shown in Fig.1 (a) (b).

<Insert Figure 1 here>

2.2 Constraints Management

Constraints management is one of the critical strategies for production control and planning. The concept of constraint was firstly introduced in 1984 as the theory of constraints, which is an overall management philosophy (Goldratt and Cox, 2016). Constraints management systems have proven to be more effective when compared to the reorder-point (ROP) systems and material requirements planning (MRP) systems in the aspects of capacity management, inventory management and process improvement in the manufacturing industry. It is also argued that constraints management can outperform the Just-in-time (JIT) system due to the more targeted nature of improvement efforts in constraints (Boyd and Gupta, 2004). The construction industry has widely recognized the significance of performing control and planning with constraint management to issue executable work plans. For example, work packaging is a planned, executable process to strategically break down the construction scope into distinct and manageable packages with proper sizing and criteria. Each work package should be assigned to a single organizational unit that is capable of handling all its constraints. The dependencies between tasks/activities contained in different work packages should also be considered. One of the practical examples is advanced work packaging (AWP) (Hamdi, 2013). AWP uses a hierarchy of engineering work packages (EWPs), construction work packages (CWPs), and installation work packages (IWPs) to allow engineering and procurement planning driven by construction sequencing. It breaks down the project processes into CWPs aligned with WBS. CWPs, in turn, contain one or more IWPs. Wang et al. (2016a) developed a conceptual framework for using AWP to improve total constraints management in the oil and gas industry. However, the direct implementation of AWP in PHP may be limited. AWP works well in handling complex mega project (e.g., oil and gas project). AWP has a hierarchical structure with CWP, EWP, and IWP. The structure is not flattening enough for PHP to improve the efficiency of decision making and collaborative working. In addition, there are still significant limitations in the work packaging method for efficiently managing constraints in PHP, particularly in the area of constraints modeling. For example, the process for identifying and analyzing constraints and their interrelationships is sluggish because the constraints are only discussed in a static manner (e.g., through look-ahead meeting) rather than in a real-time manner (Hamdi, 2013; Isaac et al. 2017). Some studies have also conducted static constraints identification by social network analysis (Gong et al., 2019). However, automatic constraints identification and dynamic constraints interrelationship mapping have not been investigated. Enlightened by the smartness of smart construction object (SCO) (Niu et al., 2016), a more collaborative, autonomous, and adaptive approach for dynamic constraints modeling may be possible.

2.3 Smart Work Packaging

Much effort has also been made in using cutting-edge information technologies to make work packages smart (Ibrahim et al., 2009; Abuwarda and Hegazy, 2016). For example, Isaac et al. (2017) developed algorithms for BIM which can be integrated with design structure matrix and domain mapping matrix to automatically label relationships between prefabricated products and their following sequence in which the prefabricated products should be assembled. The development of smart work packaging (SWP) originated from the manufacturing industry to improve the smartness of the workflow. Some studies, although not directly using the term "smart work packaging" or SWP, address the interaction between humans, resources (e.g., machines and products) and environment with smartness using emerging technologies such as IoT, wireless sensor networks, big data, cloud computing, or other enabling technology to facilitate tasks execution. Compared with traditional task execution process, SWP has many unique characteristics, including traceability, value-added, and awareness. However, information communication, adaptive to changes, and autonomous actions during task executions have been identified as necessary requirements of SWP in previous studies (Lu et al.2017; Wang et al. 2016b; Ren et al. 2017; Lee et al., 2009). For example, based on simulated or historical data, SWP could achieve autonomy by executing particular tasks when specific requirements are met (Lu et al., 2017). In addition, each smart work package can gain sociability by communicating with its internal elements, as well as other SWPs to work as a distributed multi-agent system for collaborative working (Ren et al., 2017). Most importantly, SWP must be adaptive and can react flexibly to changes by learning from its own experiences, environment, and interaction with others (Wang et al. 2016b; Lee et al., 2009). Thus, it is believed that the three critical characteristics of SWP are autonomy, adaptivity, and sociability. The potential functions of SWP have also been introduced and assessed in different scenarios including modeling (i.e. the understanding of the interconnections among tasks), monitoring (i.e. the tracking and updating of real-time status), and optimization (i.e. the planning and scheduling of tasks) (Luo et al. 2018; Wan et al. 2018; Zhang et al. 2018). Although the SWP is expected to improve the constraint management, modeling the constraints (i.e., identification and relationship mapping) through an automatic and dynamic approach is the very first step toward a "zero-constraint" environment. Such a step requires identifying critical constraints and understanding the interrelationships of them in a smart manner.

3. Methodology

To achieve the objectives of this study, a constraints modeling service is proposed for the on-site assembly process of PHP (See Fig. 2). This service should work as a function in the overall smart work packaging (SWP). The authors have proposed a service-oriented architecture (SOA) to encapsulate SWP into the Infrastructure as a Service (IaaS) layer in the previous study (Li et al., 2019). Based on this conceptual study, in order to provide practical and useful tools for workers to automatically identify critical constraints, dynamically explore interactional and interdependent relationships of these constraints, and understand the impact of these constraints on schedule performance, we further embed a few practical techniques and analytics methods into the SWP. The identification of critical constraints and their interrelationships, as the first step in constraints modeling service, is supported by social network analysis (SNA) technique, which applies social network theory to help explore the complex system that contains miscellaneous relationships. The on-site assembly process can be considered as an intricate network involving different workers. The integration of SNA can, therefore, help facilitate the identification of critical trades associated constraints and their cause-and-effect relationships in the on-site assembly process of PHP. The use of SNA can be found in various research fields, such as schedule risk (Li et al.,2016), urban renewal (Yu et al., 2017), and social responsibility (Lin et al., 2018). The authors have also investigated the use of SNA for constraints identification in a static manner (Gong et al., 2019). However, the adoption of automatic and dynamic SNA has not been investigated. Therefore, in this study, the SNA sub-service is proposed (see Fig. 2). It has three major steps: (1) The workers of different trades register or log-in the SNA service of their own SWP and get the constraints template; (2) they score and evaluate the constraints interrelationships; (3) they visualize the network and identify the critical constraints and interactions in an automatic manner.

<Insert Figure 2 here>

Secondly, assessing and simulating the potential effect of the identified critical constraints on the schedule performance of PHP should be considered in SWP to facilitate the decision making of the workers. Computer simulation has been widely adopted in diverse decision-making in construction processes by enabling ‘what-if’ scenarios (Lee, 2017). Discrete Event Simulation (DES) has been a primary means for such simulation, representing sequential operation details (Alvanchi et al., 2011). As DES models can offer detailed information for execution, they have been primarily used to solve operational issues (e.g., physical constraints) such as shop-floor fabrication and on-site assembly which can replicate the PHP processes for helping different trades to analyze their constraints. However, DES is deficient in the dynamic analysis of system interaction. For example, DES models can analyze on-site assembly process with an event-oriented view but cannot organize feedback structures between process performance (e.g., schedule performance) and its project contexts (Hwang et al., 2016). Instead, the control theory-based system dynamics (SD) models can be applied to analyze the interactions (e.g., casual loop) and structures (e.g., stock and flow) of the project environments due to their perfect demonstration of feedback effects. Also, SD models are efficient to integrate management actions. Unlike the DES models which target operational details, SD models focus on handling strategic issues (e.g., informational constraints) (Li et al. 2018a). Thus, by considering the advantages of DES and SD, a hybrid SD-DES dynamic model can be embedded into SWP to help workers of different trades conduct a more comprehensive constraints evaluation in both operational and strategic levels. In this research, a customized SD-DES hybrid dynamic model sub-service is developed to encapsulate the SD models into each event in the DES model. DES model primarily facilitates to measure the operation level of the on-site assembly system including the capacity and number of project resources, the duration of on-site assembly tasks, and the lifting distance of the crane tower. SD models are primarily linked to strategic level context, such as the satisfaction level of the tasks, level of worker fatigue, level of worker skill. The development of SD-DES hybrid dynamic model sub-service has three significant steps: (1) Define the system boundaries of the SD-DES hybrid dynamic model service; (2) Encapsulate the SD models and their associated attributes into the DES model for simulating the variations in the schedule performance of PHP; (3) Validate the developed model through conducting structure and behavior tests. This validation process can build up the confidence of the simulation results. Thirdly, constraints scenario analysis is conducted for both project managers and workers to understand different simulation results so that the influence of different critical constraints on schedule performance can be understood.

4. SWP-enabled Constraints Modeling Service

4.1 Constraints Identification

The SNA sub-service of constraints modeling in the SWP can automatically identify the critical constraints and their interrelationships. The functions of SNA sub-service can be divided into three parts (see the interface in Figure 3):

(1) The workers of different trades register or log-in SNA sub-service in their mobile device and get the constraints template. The initial list of constraints is generated from the look-ahead meeting of a real PHP project owned by the Hong Kong Housing Society (HKHS) (see Table 1). The templated constraints are pre-programmed with an open-data integration approach for constraints instantiation.

(2) The interrelationships among identified constraints are determined by links representing the influence of constraints over another constraint. There are two steps in this process. The workers of different trades (The trade list is collected from on-site assembly process of the same PHP project which can represent a typical four-day assembly cycle (FDAC) (see Table 2) were required to clearly set the direction of potential influence according to their empirical knowledge in the service interface, and the direction of relationships can be mutual. For example, the influence generated by T_1C_2 on T_3C_4 was distinct from the influence of T_3C_4 on T_1C_2, and they are considered as two different links. After tabulating the identified links, they can be quantified by two metrics including the *intensity of influence* (adopting a five-point scale where "0" and "5" signify the lowest and highest levels) and likeliness of the influence occurrence (adopting a ten-point scale where "0" and "1" represent the lowest and highest levels, i.e., 0.1, 0.2, etc.). The multiplication of the *intensity of influence* and *likeliness* offers a basis for evaluating the influence level between two trades associated constraints. When no influence occurs between two nodes, the influence level is zero.

(3) The SNA sub-service calls the NetMiner tool (an SNA application analytics) to visualize and analyze the adjacency matrix lists of link and node. There are three steps in this process. The on-site superintendent can visually exam the primary constraints and their relationship distribution in the network. The metrics value and description of network density and cohesion can be displayed to reflect the overall connectedness and complexity of the network. In addition, the pre-selected node-level metrics (e.g., out-degree/out-status centrality, node betweenness centrality, and out-closeness /eigenvector centrality) can be computed to investigate the characteristics and roles of individual nodes for determining

the critical constraints. Besides node-level metrics, link betweenness centrality was also calculated to assess the critical interrelationships among constraints. It can help disclose the cause-and-effect relationships of these constraints. As shown in Fig. 3, the output of SNA sub-service is a list of critical constraints and critical interrelationships among these constraints,which is used in the subsequent SD-DES hybrid model, and more details can be found in the authors' previous study (Gong et al., 2019). The trades can re-evaluate the constraints, and the SNA service can also re-generate the output in a real-time manner.

<Insert Figure 3 here>

<Insert Table 1 here>

<Insert Table 2 here>

4.2 Development of Hybrid SD-DES Model

This hybrid SD-DES model sub-service is developed to help workers and site managers to investigate the influence of the critical constraints and interrelationships (identified in the above SNA sub-service) on the schedule performance of the FDAC in the on-site assembly process. To achieve this objective, the development of the system boundary, the SD model and DES model are explained below.

4.2.1 System Boundary

The definite system boundary can facilitate to generate specific system structures and behaviors. In this study, the SD-DES model includes three subsystems: the FDAC process, constraints, and schedule performance. The connection between the three subsystems can be presented in Figure 4. The first system, i.e., the FDAC subsystem (See Fig.1), includes activities related to prefabricated products installation and in-situ tasks. The schedule performance subsystem mainly consists of the planned schedule and the actual schedule to measure their differences. The PHP assembly process will delay if the actual schedule lags behind the planned schedule.

<Insert Figure 4 here>

According to the literature review and on-site surveys conducted in Gong et al. (2019), constraints can mainly impede on-site assembly in three ways: resource availability, operation efficiency, and work quality. First, resources can include labor, prefabricated products, machinery (e.g., crane) and workspace. Resource availability has positive interactions with schedule performance because the PHP workflow can be interrupted or suspended if resources become unavailable due to certain constraints, especially in the compact assembly site of PHP projects in Hong Kong. In contrast, if the schedule is delayed, the project team can increase the number of resources (e.g., labor, crane) to recover the delay. Second, operation efficiency indicates the proficiency and accuracy of machinery and labor, and the constraints in operation efficiency still depress the productivity in PHP project even though the information and communication technologies have been widely adopted in a construction site (Li et al., 2017a). It is also the reason to develop the SWP approach in the authors' series of studies. Efficient operations can speed up the installation rate, whereas inefficient operations can increase the installation error rate, thereby leading to the schedule delay. Besides, when schedule delay occurs, workers and machinery may be pushed to conduct unsafe and fatigued operations. The operation efficiency may again decrease due to low installation rate and high error rate. Finally, quality concerns, such as defects of prefabricated products, are also significant. Quality control is conducted when products arrive on-site or when they are assembled. Defective products should be reworked or reproduced and extra time is consumed in terms of re-installation and transportation of new products, thereby resulting in schedule delay. Besides, pushing the progress to recover existing delay may also increase the possibility of quality problems. Thus, quality concerns are interrelated with other subsystems of the model.

4.2.2 System Dynamics (SD) Model

To perform a detailed quantitative analysis of system's structure and behavior, the previously defined and described casual loop relationships in Section 4.1 and Section 4.2.1 are transformed to an SD diagram (See Fig.5) to address the subsystems of constraints and the schedule performance for both prefabricated products installation and in-situ tasks.

The SD model is based on the SD scheme adopted in several studies (Nasirzadeh and Nojedehi, 2013; Li et al., 2018a; Wu et al., 2019). In SD, stocks, dynamic variables, and flows are the basic building blocks. Stocks monitor cumulative quantities (e.g., task completion rate); dynamic variables monitor non-cumulative quantities (e.g., labor and crane efficiency); and in- and out-flows are used to connect stocks to indicate the increasing and decreasing rate of the stock value. SD also has parameters whose values are fixed during the simulation and are used to depict the static attributes of a system (e.g., basic inspection rate and production rate of prefabricated products). All the SD elements are linked together to form feedback loops that reflect the underlying mechanism of a system (Wu et al., 2019). This SD model works as a standardized element to depict the specific FDAC process with surrounded constraints. The rationale of this SD model is supported by four modules, namely, assembly process module, resource availability module, operation efficiency module, and schedule performance module. The details of these modules are discussed in the following sections. It should be noted that the SD structures of installation and in-situ tasks are similar. Thus, in the following sections, (1) - (4) introduces the modules for installation tasks, whereas the modules of in-situ tasks are introduced in (5) by highlighting the differences.

(1) Assembly Process Module (APM)

This module is the main skeleton of the SD model for installation tasks, which simulates different statuses of prefabricated products by SD stocks (see Fig.5), such as “Products To Be Assembled,” “Assembled Products” and “Inspected Products.” The “Products To Be Assembled” stock refers to the total amount of prefabricated products (e.g., prefabricated facades) that have been delivered to an on-site buffer and should be assembled. This is linked to another “Assembled Products” stock by a flow named “Installation Rate,” which is determined by some dynamic variables, such as “Crane Efficiency,” “Labor Efficiency” and “Resource Availability.” At the quality checking stage, the installed façades are translated into the “Inspected Products” stock at the “Inspection Rate,” which is determined by the parameter “basic Inspection Rate” and several constraints identified using the method introduced in Section 4.1. The mechanism of other stocks, dynamic variables, and flows, such as “Products To Be Delivered,” “Products To Be Re-assembled,” “Delivery Rate” and “Re-installation Rate,” follow the same principles.

<Insert Figure 5 here>

(2) Resource Availability Module (RAM)

The resources in this study include labor, material (e.g., prefabricated products), machinery (e.g., crane), and workspace (e.g., buffer, workface). An optimal resource availability level can keep the installation rate at a reasonable range to align with the planned schedule. In RAM, the critical feedback loop is determined by two SD variables. One loop starts from the critical constraint C22: lack of optimal buffer layout, identified by the SNA sub-service in Section 4.1. This constraint can affect inadequate buffer space (i.e., C21), and constraints related to availability and capacity of labor and cranes (i.e., C4, C13, and C23). Moreover, C22 also affects the stock “Products To Be Assembled” indirectly by the flow “Delivery Rate” in the APM. The other dominant SD variable is "PSD," standing for the predicated schedule delay. "PSD" can directly push to increase the number of labor and cranes, and indirectly affect the number of prefabricated products by "Delivery Rate." However, the labor and cranes could not exceed the expected maximum quantity limited by the buffer space or workspace. This module is integrated with the APM bidirectionally. For example, "Resource Availability" is embedded in the APM as one major affecting factor of the flow "Installation Rate". At the same time, the stock "Products To Be Assembled" in the APM is embedded in the RAM which affects the congestion level of workspace.

(3) Operation Efficiency Module (OEM)

The operation efficiency includes the workers' efficiency and cranes' efficiency. The workers' efficiency is largely determined by constraints related to information, quality and safety, such as C20: lack of visible and audible communication mechanism, C9: unavailable quality control hold points, and C29: Inadequate safety training and hazards identification. The relationships among the work pressure, fatigue, and other constraints that can hinder safety and quality operations have also been investigated in previous studies (Lee, 2017). Thus, the work pressure and fatigue degree can also affect the workers' efficiency. The crane-related constraints are in the critical path of the assembly schedule, which has also been identified as the critical constraints in SNA sub-service. If these constraints are not timely removed, crane efficiency in terms of transporting prefabricated products (from lift point to the place point) in a Just-in-time (JIT) manner cannot be achieved. For example, the lack of optimal installation sequence and the lack of collision-free path planning can lead to numerous rework in the horizontal and vertical transportation of prefabricated products. Additionally, bad weather conditions (e.g., heat-stress) that always happen in the summer of Hong Kong can impede the progress of the PHP project or reduce worker efficiency, therefore, affecting the installation rate. The OEM module is integrated into the APM unidirectionally, i.e., the OEM only compute the worker and crane efficiency data and transfer it to the APM.

(4) Schedule Performance Module (SPM)

This module is used to calculate schedule delay when constraints are not timely removed. For this purpose, the planned percentage of completion (PPC) and the actual percentage of completion (APC) are computed by extracting data from APM. The two indicators are then used to evaluate "PSD" which is sent back to APM, RAM, and OEM. Therefore, actions such as employing extra workers and renting additional cranes, can be taken to remove constraints based on the degree of delay. Some details of calculation in this module can be seen in Table 4.

(5) Modules for In-situ Tasks

An FDAC process, as shown in Fig 1, also includes in-situ tasks, such as wall reinforcement and conduit installation, slab and beam rebar and inspection, and wall, slab and beam concreting. All the tasks can be modeled by a similar SD structure, which also includes four parts similar to the APM, RAM, OEM, and SPM. However, there are several differences. A Work Progress Module is set up to replace the APM, including four stocks, i.e., "Work To Be Completed," "Completed Work," "Inspected Work" and "Work To Be Redone," respectively. No prefabricated products are needed for in-situ tasks, therefore stocks, dynamic variables, and flows relating to prefabricated products delivery and production are omitted. Second, in the RAM and OEM for in-situ tasks, the workspace congestion caused by crane and crane efficiency is no longer considered because the material transported by crane for the in-situ tasks is not on the critical schedule path according to the project documents. Furthermore, in the SPM, the mechanism to compute schedule delay is the same, but dynamic variables used to compute "Total Quantity To Be Completed" are different. The structures of modules for in-situ tasks are shown in Fig 6.

<Insert Figure 6 here>

4.2.3 Encapsulating SD model into the DES model

An FDAC cycle requires to arrange multiple specific tasks with proper preceding and succeeding dependencies. To mimic this process, as shown in Fig. 7, a DES model is built, which addresses the FDAC subsystem of the system boundary. Building blocks in the DES model are “delay” and “hold.” The “delay” block refers to an ongoing installation or in-situ task; the “hold” block controls the pace of construction according to the project plan and completion rate of preceding tasks. There are two types of “hold” block. One type, e.g., the “hold” between “Wall_Rebar_A” and “Slab_Beam_Rebar_A” in Fig. 7, prevents succeeding tasks from starting too early to stick to the original plan in Fig. 1(b), which is necessary to avoid workers being idle due to early completion of preceding tasks (Kenley and Seppänen, 2006). The other type, however, forces succeeding tasks to wait until all preceding tasks are completed, such as the “hold” before “Concrete_A.”

<Insert Figure 7 here>

The conceptual structures of installation and in-situ tasks defined in Section 4.2.2 are used to generate and assign tasks into the DES model. For this purpose, a technique in object-oriented programming, i.e., encapsulation, is applied, where a class is defined as a blueprint of all objects belonging to that class by grouping (or encapsulating) common information of the objects into a logical unit. As illustrated in the SD structures, installation and in-situ tasks have distinct characteristics. Thus, they are defined as two classes, with all relevant information encapsulated in their SD models. The installation task class generates tasks such as “Pre_facade_A” while the in-situ task class generates tasks such as “Concrete_A.” The encapsulation contributes three merits to the SD-DES model: (1) It keeps the properties integrality of each task module; (2) It facilitates the scalability of the DES model; and (3) It enhances the reusability of SD models.

The integration mechanism between DES and SD is bidirectional and is shown in Fig. 8. On the one hand, each task is generated and assigned into a "delay" block at the time *a*, and is released at time *b*, when two conditions are satisfied: (1) the earliest start time defined in the project plan is reached; (2) the variable "APC" in the SPM becomes 100%. On the other hand, in the DES model, a timer is activated in "delay" blocks to record the time spent for each task by subtracting *a* from *b*. Thus, the sum of all timers is the total working time (TWT) of all tasks whereas the total cycle time (TCT) of one FDAC cycle is recorded at time *c* (measured by the model engine's timer directly) when the "End" block in Fig. 7 is reached. TWT is greater than TCT since some tasks are performed in parallel. TWT and TCT are important indicators for model validation and results comparison (see Section 5.2 and 5.3). Meanwhile, "Total Work Hours" is derived from TWT and is sent back to SD models to evaluate values of dynamic variables (see Section 5.1).

<Insert Figure 8 here>

All variables in the SD models are linked to the database in the SD-DES model service. When the model starts, data can be extracted from the database. Meanwhile, the results of the simulation can be saved to the database for further analysis. In addition, a set of interfaces and data input/output plug-ins are developed in the SD-DES model service to capture, store, and visualize the real-time modeling and simulation process.

4.3 Constraints Analysis

The constraints analysis sub-service can be activated when the hybrid SD-DES model sub-service has been successfully developed. This kind of scenario analysis can work as a sub-service of SWP to quantitatively measure the influence of these critical constraints on schedule performance under different constraints scenarios. The simulation results can not only be visualized by considering various constraints scenarios at the different time points of the FDAC but also provide decision support by predicting the assembly duration variation when different constraints are not timely removed at different time points. In this constraints analysis sub-service, a set of constraints scenarios are proposed based on real project experience, and a comparative analysis is conducted between these scenarios.

5. Case Simulation

Based on the three sub-services, a case study is conducted to verify the proposed SWP-enabled constraints modeling service. The case is a Subsidized Sale Flats project owned by the Hong Kong Housing Society (HKHS) and is located at 48 Chui Ling Road, Tseung Kwan O Area 73A. It adopts the typical FDAC to construct one residential tower of 33 stories, 330 flats with 1020 units, including one basement (car park, plant room), and 4-level podium. In this project, only the prefabricated façade is considered, which incorporates nine different kinds of modules to form 26 different types of façades.

The first step is to identify critical constraints. After the trial of SNA sub-service in Section 4.1, five constraints are identified as critical, i.e., C5 bad weather conditions, C14 lack of collision-free path planning, C20 lack of visible and audible communication mechanism, C22 lack of optimal buffer layout, and C23 unavailable optimal installation sequence.

5.1 Data Collection and Quantification of the SD-DES Model

Prior to launching the simulation, the SD-DES model must have accurate data inputs. According to the attributes of these data inputs, they are categorized into three categories, i.e., parameters, dynamic variables, and constraints. Data sources of each group are summarized in Table 3 and are explained below.

<Insert Table 3 here>

Parameters, such as “Basic Inspection Rate” and “Basic Work Efficiency,” have fixed values during the simulation and usually serve as the baseline to evaluate values of dynamic variables. The values of parameters are collected by reviewing project documents, such as planned schedules, construction plans and bill of quantities. Dynamic variables are SD elements (introduced in Section 4.2.2) whose value are determined by other elements (e.g., constraints, stocks, parameters, and other dynamic variables) linked to them. Thus, the value of a dynamic variable is not collected but computed, by embedding equation in the variable, considering all elements linked to it. Finally, constraints are divided into two groups according to their interrelationships identified in Section 4.1. One group consists of dependent constraints where their effect is affected by other constraints. For instance, C14 Lack of collision-free path planning is affected by C20 Lack of visible and audible communication mechanism, and the effect of C14 on “Crane Efficiency” will increase if C20 is not removed. The other group consists of independent constraints, which only affect others but are not affected during the project. These constraints are related to the environment, supply chain and project planning problems, such as C5 Bad weather conditions, C8 Bad conditions of transportation vehicle and rout and C29 Inadequate safety training and hazards identification.

Then, the quantification of dynamic variables and constraints is completed in three ways. First, equations in similar SD models from qualified journals are searched, which are mainly used to calculate values of dynamic variables. Using such equations in is a common practice in SD model building to reduce development time and increase model reliability (Wu et al., 2019). Besides, some equations can be built directly based on the structures of SD model and common knowledge of project management. Finally, a project-level approach is adopted to quantify the effect of constraints on dynamic variables and the mutual effect between constraints, because such information is highly project-dependent and off the shelf equations cannot be found. For this purpose, engineers of the case project are asked to give an estimation, and the average value is taken. For instance, the negative effect of C14 Lack of collision-free path planning on "Crane Efficiency" will be further increased by if C20 is not removed.

Given the limited space, Table 4 gives some examples of establishing equations in the SD modules of installation tasks, which includes all the three ways to quantify the SD model. It should be noted the FDAC procedure is very mature and standard in Hong Kong (Jailon and Poon, 2009). Thus data collected from a typical FDAC project, and the estimation provided by experienced engineers can be considered as stable and representative.

<Insert Table 4 here>

5.2 SD-DES Model Validation

This section verifies the validity of the SD-DES model with two tests, i.e., direct structure test (DST) and Structure-oriented Behavior Test (SBT).

5.2.1 Direct Structure Test (DST)

Direct structure test (DST) directly performs qualitative comparison between the model structure and the real system, which includes three sub-tests: (1) structure and parameter confirmation tests, which examines if all the causality, feedbacks, and parameters of this model can be reflected from the real system; (2) dimensional consistency test, which examines the dimensional consistency of equations and ensures that there is no illogical parameter; (3) boundary adequacy test, which ensures all crucial variables keep in line with the research objectives (Barlas, 1996). As mentioned in Section 4.2.2 and Section 5.1, the SD structures and equations are established based on verified works while the DES model is built by referring to the mature FDAC cycle. Furthermore, the model structures and the selection of parameters and variables have been explained to the project managers of the case project to gain their agreements. Thus, the model meets the requirements of DST and can reflect the real project.

5.2.2 Structure-oriented Behavior Test (SBT)

Structure-oriented behavior test (SBT) is a quantitative test, which investigates model-generated behavior patterns to uncover potential structural flaws (Wakeland et al., 2005). It can be achieved by extreme-condition test, behavior sensitivity test, and integral error test.

(1) Extreme Condition Test

The extreme-condition test exams, whether a model is reasonable under extreme conditions. Given the aim of the study is to investigate the influence of constraints, the status of constraints are used to setup extreme conditions. For example, the most optimistic and pessimistic cycle time (TCT) of constructing 33 typical floors adopting FDAC, according to the project plan, is 132 (i.e., 4 days per floor) and 231 days (i.e., 7 days per floor), respectively. The test results are shown in Table 5. After 200 simulation runs, when no constraint exists, the average TCT and deviation rate is 132.50 days and 0.71%, respectively, whereas when all constraints are not removed, the average TCT and deviation rate is 230.06 days and 4.75%, respectively. Both results are acceptable (i.e., the deviation rate is less than 5%) and comply with the plan.

<Insert Table 5 here>

(2) Sensitivity Test

This test detects parameters to which FDAC tasks are sensitive and asks if the real system exhibits similar high sensitivity to these parameters. To interpret in details, ten parameters are selected, i.e., “Basic Worker Efficiency,” “Error Rate” and “Basic Inspection Rate” for both in-situ and installation tasks, as well as “Basic Delivery Rate,” “Defect Rate,” “Defect Rate After Assembly” and “Basic Crane Efficiency” for installation tasks, because these parameters are essential in the SD model by affecting many other dynamic variables. To assess their potential influence on project duration, each parameter is assigned with a maximum, minimum, and most likely value. For instance, the minimum “Defect Rate” of prefabricated products is 0 whereas the most pessimistic (i.e., maximum) “Defect Rate” in the PHP projects is 10% in Hong Kong (Li et al., 2018a). However, due to the paralleled tasks planning, the variation of TCT is not significant. Thus, the TWT which aggregates the time spent for each task is adopted to eliminate the paralleling effect, and the TWT baseline is 140.86 days in the optimal case. The variety range of each parameter, i.e., the sensitivity is computed using the following equation.

$$\left|\frac{\textit{Minimum duration} - \textit{Most likely Duration}}{\textit{Most likely duration}}\right| + \left|\frac{\textit{Maximum duration} - \textit{Most likely Duration}}{\textit{Most likely duration}}\right|$$

Table 6 summarizes the simulation results. This study uses 20% as the threshold of variation. Thus a variable is treated as sensitive if its variety range exceeds 20% (Li et al., 2018a). As a result, “Basic Delivery Rate,” “Basic Worker Efficiency^{I}” and “Error Rate^{I}” are sensitive variables, which complies with the reality. For one thing, installation is only a small part of the FDAC cycle in terms of total duration. Thus the sensitivity of parameters related to installation is less than those related to in-situ parameters. Among these installation parameters, “Basic Delivery Rate” is most sensitive as it starts the installation and affects subsequent processes; “Basic Worker Efficiency” and “Basic Crane Efficiency” are less sensitive as they depend on the delivery rate of facades meanwhile they are constrained by each other and none of them determines the installation rate alone (see Fig. 5); “Error Rate^{P}” is not sensitive because the total amount of prefabricated facades is small, and even the pessimistic estimation of the installation error rate is still low given the mature FDAC cycle. For another, “Basic Worker Efficiency^{I}” and “Error Rate^{I}” are sensitive for in-situ tasks as they are the dominant factors behind the construction pace when work is less constrained delivery and crane (see Fig. 6)

<Insert Table 6 here>

(3) Integral Error Test

This test investigates whether the model behavior varies with the different integration method or time step. This study uses 4th order Runge-Kutta with the different time step: 0.5, 0.25, 0.125 and 0.0625 day/time, the model behaviors with durations are 132.35, 132.74, 133.18, and 133.52 days, indicating that this model can meet the requirement of this test.

5.3 Constraints Analysis Results

In this section, the five identified critical constraints are fused into the SD-DES model with different scenarios through three simulation tasks. In the initial stage, all constraints are assumed to be satisfied, whereas some critical constraints can re-appear at certain time points. Besides, one FDAC has 4 days, and the planned duration of that cycle is 5760 minutes (4*24*60=5760).

The first task assesses the impact of different constraints on schedule performance when they appear at the same time point. For example, 500th minute after simulation lauching is selected as the investigated point, at which all the five critical constraints are scheduled to appear. After summarizing results of 200 simulation runs, a histogram with a density curve of the simulated duration is drawn in Fig.9. The simulated duration ranges between 5800 and 8940 mins and has a 95% probability of falling in between 6006.5 and 8758.75 mins, with a median duration of 7060.32 mins. Through the statistical analysis of simulated duration, workers and project managers can adjust and re-plan the task executions at the specific time point.

The second task is to assess the impact of different constraints on schedule performance at various time points. For instance, the scenario, "C5 cannot be satisfied at the 100^{th} minute," which takes the form of $C5^{100th}$, has different impact level on schedule performance compared to the scenario $C5^{800th}$. As shown in Fig. 10, the horizontal axis indicates the incidence time of constraints, and the vertical axis denotes the results of simulated duration. The larger the length of the box signifies, the more significant impact it will have on the schedule performance. Measured by the median (the band inside the box) value, $C5^{100th} > C5^{800th}$ can be observed at the different time points which indicates that some constraints have more influence on the whole schedule performance at the early stage of the cycle. As another example, the median of $C14^{100th}$ is just slightly higher than $C14^{800th}$, signifying that the inefficient crane operations cannot be significantly reflected in a short time period, which is usually accumulated to a considerable difference in the later stage.

<Insert Figure 9 here>

<Insert Figure 10 here>

The final task is to find out the constraint that has most significant influence on the schedule performance of FDAC cycle. For this purpose, each critical constraint is individually scheduled to appear at 100^{th} minutes. After 200 simulations, density curves of all the five critical constraints are generated and shown in Fig.11. Table 7 presents the constraints ranking in terms of their impact on schedule and includes relevant statistical information of TCT. Based on the mean value of FDAC duration, the constraints can be divided into three levels in terms of their effect on schedule. The first level includes C14 and C5 that can lead to a delay up to 249.85 minutes (i.e., 4.16 hours) in one FDAC cycle. In other words, if C14 is not satisfied, the total delay of the 33 floors can be 137.61 hours, nearly 6 days. The second level contains C22 and C20 that result in delay up to

137.75 minutes (i.e., 2.30 hours) in one FDAC cycle. Finally, the third level includes C23 which causes minimal delay (i.e., 16.63 minutes) in one FDAC cycle.

<Insert Figure 11 here>

<Insert Table 7 here>

In summary, constraints' impact on schedule delay varies along the timeline of the FDAC. If constraints that cannot be satisfied happen at an earlier stage, they have a more significant impact on schedule performance. To this end, this study provides an in-depth understanding of how the impact of constraints can be systematically analyzed, thus offers valuable insights to the project team to adopt constraints improvement approaches to achieve a reliable workflow.

6. Conclusion

In order to automatically identify the critical constraints of on-site FDAC of PHP process and dynamically understanding the interrelationships of them, a constraints modeling service of SWP has been developed within the three sub-services, namely, social network analysis (SNA), hybrid SD-DES model, and the constraints analysis. The SNA helps identify the trades associated critical constraints including bad weather conditions, lack of collision-free path planning, lack of visible and audible communication mechanism, lack of optimal buffer layout , and lack of optimal installation sequence. The hybrid SD-DES model helps dynamically explore the interactional and interdependent relationships of the constraints in the modules of the assembly process, resource availability, operation efficiency, and schedule performance, and encapsulate these dynamic relationships into the DES model. The hybrid SD-DES model has also been validated by the model structure and behavior tests to guarantee the confidence and validity of this sub-service. The constraints analysis then helps analyze the impact of critical constraints on schedule performance over various simulated scenarios.

The main contributions of this study to the body of knowledge are twofold. It enhances the role of constraints management within dynamic modeling methods (e.g., SNA, SD, DES) and extending its contribution to achieving the sociability of the SWP at the task execution level. Compared with previous studies, such as Li et al. (2018a) which investigated the issue at the phase level, this study provides approaches to identify critical constraints and evaluate the impact of these constraints at the work package level. Secondly, the dynamic modeling methods in this study extend the process of constraints modeling of the trades associated work packages in a more structural and convenient way. The system dynamic models are established as reusable "task modules" to be encapsulated into the DES model. Thus, this hybrid model can be utilized for other PHP projects due to the sufficiently generic nature of this model. This constraints modeling service also provides a comprehensive view of constraints relationships and interconnections, which is beneficial for identifying critical constraints and evaluating the influence of each constraint. Such influence can be evaluated at an early stage of the project, which leaves enough time for project teams to establish relevant constraints management plan.

However, there are a few limitations of the developed constraints modeling service and enhancement, especially in the area of automatic constraints tracking and sensing, are needed to eventually create living digital simulation models to represent near real-time modeling. In addition, given the number of interrelationships among the constraints and the variables of the model that influence the schedule performance, there is a challenge to collect data from multiple data sources and establish all dynamic interactions into the model. Thus, future studies can focus on using sensor data, which conveys various aspects of the operating condition of SWP to generate the data-driven constraints modeling service.

Acknowledgments

This research was supported by the Research Institute for Sustainable Urban Development of the Hong Kong Polytechnic University. This research was also funded by the Australian Research Council Discovery Project (No. DP180104026), National Natural Science Foundation of China (No. 71801159), and Natural Science Foundation of Guangdong Province (No. 2018A030310534).

Declarations of interest

None

Reference

Abuwarda, Z., & Hegazy, T. (2016). Work-package planning and schedule optimization for projects with evolving constraints. *Journal of Computing in Civil Engineering, 30*(6), 04016022.

Alvanchi, A., Lee, S., & AbouRizk, S. (2011). Modeling framework and architecture of hybrid system dynamics and discrete event simulation for construction. *Computer‐ Aided Civil and Infrastructure Engineering*, *26*(2), 77-91.

Barlas, Y. (1996). Formal aspects of model validity and validation in system dynamics. System Dynamics Review: *The Journal of the System Dynamics Society, 12*(3), 183-210.

Blackmon, T., Saxena, R., & Song, L. (2011). A conceptual framework for total constraint management in construction. *Proceedings of the 28th ISARC,* 419-424.

Boyd, L., & Gupta, M. (2004). Constraints management: what is the theory?. *International Journal of Operations & Production Management*, *24*(4), 350-371.

Chi, H. L., Chen, Y. C., Kang, S. C., & Hsieh, S. H. (2012). Development of user interface for teleoperated cranes. *Advanced Engineering Informatics, 26*(3), 641-652.

Emuze, F., Smallwood, J. and Han, S. (2014). Factors contributing to non-value adding activities in South African construction. *Journal of Engineering, Design and Technology*, 12(2) 223-243.

Goldratt, E. M., & Cox, J. (2016). The goal: a process of ongoing improvement. Routledge.

Gong, P., Teng, Y., Li, X., & Luo, L. (2019). Modeling Constraints for the On-Site Assembly Process of Prefabrication Housing Production: A Social Network Analysis. *Sustainability, 11*(5), 1387.

Hamdi, O. (2013). Advanced Work Packaging from project definition through site execution: driving successful implementation of WorkFace Planning.

Hanna, A.S., Taylor, C.S. and Sullivan, K.T. (2005). Impact of extended overtime on construction labor productivity. *Journal of Construction Engineering and Management*, 131(6), 734-739.

Housing Authority (2018), Number of Applications and Average Waiting Time for Public Rental Housing, https://www.housingauthority.gov.hk/en/about-us/publications-and-statistics/prh-applications-average-waiting-time/

Hwang, S., Park, M., Lee, H. S., & Lee, S. (2016). Hybrid simulation framework for immediate facility restoration planning after a catastrophic disaster. *Journal of Construction Engineering and Management, 142*(8), 04016026.

Ibrahim, Y. M., Lukins, T. C., Zhang, X., Trucco, E., & Kaka, A. P. (2009). Towards automated progress assessment of work package components in construction projects using computer vision. *Advanced Engineering Informatics, 23*(1), 93-103.

Isaac, S., Curreli, M., & Stoliar, Y. (2017). Work packaging with BIM. *Automation in Construction, 83,* 121-133.

Jaillon, L., & Poon, C. S. (2009). The evolution of prefabricated residential building systems in Hong Kong: A review of the public and the private sector. *Automation in construction, 18*(3), 239-248.

Kenley, R., & Seppänen, O. (2006). Location-based management for construction: Planning, scheduling and control. Routledge.

Lee, K., Paton, N. W., Sakellariou, R., Deelman, E., Fernandes, A. A., & Mehta, G. (2009). Adaptive workflow processing and execution in Pegasus. *Concurrency and Computation: Practice and Experience, 21*(16), 1965-1981.

Lee, S. (2017). Applying system dynamics to strategic decision making in construction. *Frontiers of Engineering Management, 4*(1), 35-40.

Li, C. Z., Hong, J., Xue, F., Shen, G. Q., Xu, X., & Mok, M. K. (2016). Schedule risks in prefabrication housing production in Hong Kong: a social network analysis. *Journal of Cleaner Production, 134*, 482-494.

Li, C. Z., Xu, X., Shen, G. Q., Fan, C.,Li, X., Hong, J. (2018a). A model for simulating schedule risks in prefabrication housing production: a case study of six-day cycle assembly activities in Hong Kong. *Journal of Cleaner Production,* 165, 10

Li, C. Z., Xue, F., Li, X., Hong, J., & Shen, G. Q. (2018b). An Internet of Things-enabled BIM platform for on-site assembly services in prefabricated construction. *Automation in Construction, 89,* 146-161.48-1062.

Li, X., Chow, K. H., Zhu, Y., & Lin, Y. (2016). Evaluating the impacts of high-temperature outdoor working environments on construction labor productivity in China: A case study of rebar workers. *Building and Environment,* 95, 42-52.

Li, X., Shen, G. Q., Wu, P., Fan, H., Wu, H., & Teng, Y. (2017a). RBL-PHP: Simulation of Lean Construction and Information Technologies for Prefabrication Housing Production. *Journal of Management in Engineering, 34*(2), 04017053.

Li, X., Yi, W., Chi, H. L., Wang, X., & Chan, A. P. (2018c). A critical review of virtual and augmented reality (VR/AR) applications in construction safety. *Automation in Construction, 86,* 150-162.

Li, X., Shen, G. Q., Wu, P., & Yue, T. (2019). Integrating Building Information Modeling and Prefabrication Housing Production. Automation in Construction, 100, 46-60.

Lin, X., Ho, C. M. F., & Shen, G. Q. (2018). For the balance of stakeholders' power and responsibility: A collaborative framework for implementing social responsibility issues in construction projects. *Management Decision, 56*(3), 550-569.

Liu, H., Al-Hussein, M., & Lu, M. (2015). BIM-based integrated approach for detailed construction scheduling under resource constraints. *Automation in Construction, 53,* 29-43.

Lu, S., Xu, C., Zhong, R. Y., & Wang, L. (2017). An RFID-enabled positioning system in an automated guided vehicle for smart factories. *Journal of Manufacturing Systems, 44,* 179-190.

Luo, Y., Duan, Y., Li, W., Pace, P., & Fortino, G. (2018). Workshop Networks Integration Using Mobile Intelligence in Smart Factories. *IEEE Communications Magazine*, *56*(2), 68-75.

Mao, C., Xie, F., Hou, L., Wu, P., Wang, J., & Wang, X. (2016). Cost analysis for sustainable off-site construction based on a multiple-case study in China. *Habitat International*, 57, 215-222.

McKinsey. (2017). Reinventing Construction: A Route to Higher Productivity. *McKinsey Global Institute,* https://www.mckinsey.com/~/media/McKinsey/Industries/Capital%20Projects%20and%20Infrastructure/Our%20Insights/Reinventing%20construction%20through%20a%20productivity%20revolution/MGI-Reinventing-Construction-Executive-summary.ashx

Nasirzadeh, F. and Nojedehi, P. (2013). Dynamic modeling of labor productivity in construction projects. *International Journal of Project Management*, 31(6), 903-911.

Niu, Y., Lu, W., Chen, K., Huang, G. G., & Anumba, C. (2016). Smart construction objects. *Journal of Computing in Civil Engineering, 30*(4), 04015070.

Ren, G., Hua, Q., Deng, P., Yang, C., & Zhang, J. (2017). A Multi-Perspective Method for Analysis of Cooperative Behaviors Among Industrial Devices of Smart Factory. *IEEE Access, 5,* 10882-10891.

Taghaddos, H., Hermann, U., AbouRizk, S., & Mohamed, Y. (2012). Simulation-based multiagent approach for scheduling modular construction. *Journal of Computing in Civil Engineering, 28*(2), 263-274.

Wakeland, Wayne and Hoarfrost, Megan, "The Case For Thoroughly Testing Complex System Dynamic Models" (2005). Systems Science Faculty Publications and Presentations. 78.

Wan, J., Chen, B., Imran, M., Tao, F., Li, D., Liu, C., & Ahmad, S. (2018). Toward Dynamic Resources Management for IoT-Based Manufacturing. *IEEE Communications Magazine, 56*(2), 52-59.

Wang, J., Shou, W., Wang, X., & Wu, P. (2016a). Developing and evaluating a framework of total constraint management for improving workflow in liquefied natural gas construction. *Construction Management and Economics, 34*(12), 859-874.

Wang, J., Sun, Y., Zhang, W., Thomas, I., Duan, S., & Shi, Y. (2016b). Large-Scale Online Multitask Learning and Decision Making for Flexible Manufacturing. *IEEE Transactions on Industrial Informatics*, *12*(6), 2139-2147.

Wang, T., Wang, J., Wu, P., Wang, J., He, Q., & Wang, X. (2018). Estimating the environmental costs and benefits of demolition waste using life cycle assessment and willingness-to-pay: A case study in Shenzhen. *Journal of cleaner production*, 172, 14-26.

Wu, C., Chen, C., Jiang, R., Wu, P., Xu, B., Wang, J. (2019). Understanding laborers' behavioral diversities in multinational construction projects using integrated simulation approach. *Engineering, Construction and Architectural Management,* https://doi.org/10.1108/ECAM-07-2018-0281

Wu, P., Wang, J., & Wang, X. (2016). A critical review of the use of 3-D printing in the construction industry. *Automation in Construction*, 68, 21-31.

Wu, P., Song, Y., Shou, W., Chi, H., Chong, H. Y., & Sutrisna, M. (2017). A comprehensive analysis of the credits obtained by LEED 2009 certified green buildings. *Renewable and Sustainable Energy Reviews*, 68, 370-379.

Yi, W., & Chan, A. P. (2017). Effects of heat stress on construction labor productivity in Hong Kong: a case study of rebar workers. *International Journal of Environmental Research and Public health*, *14*(9), 1055.

Yu, T., Shen, G. Q., Shi, Q., Lai, X., Li, C. Z., & Xu, K. (2017). Managing social risks at the housing demolition stage of urban redevelopment projects: A stakeholder-oriented study using social network analysis. *International Journal of Project Management, 35*(6), 925-941.

Zhong, R. Y., Peng, Y., Xue, F., Fang, J., Zou, W., Luo, H., ... & Huang, G. Q. (2017). Prefabricated construction enabled by the Internet-of-Things. *Automation in Construction, 76*, 59-70.