https://doi.org/10.1016/j.jclepro.2017.09.066

Schedule delay analysis of prefabricated housing production: A hybrid dynamic approach

Clyde Zhengdao Li [a], Jingke Hong [b,*], Cheng Fan [a,**], Xiaoxiao Xu [c], Geoffrey Qiping Shen [d]

a College of Civil Engineering, Shenzhen University, Shenzhen, China

b School of Construction Management and Real Estate, Chongqing University, Chongqing, 400045, China

c Department of Civil and Construction Engineering, Swinburne University of Technology, Melbourne, Australia

d Department of Building and Real Estate, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong

Abstract

Design, manufacturing, storage, transportation, and on-site assembly are fundamental processes in prefabricated housing production (PHP) in Hong Kong. However, these processes are fragmented and entail various risks that adversely affect the schedule performance of PHP, thereby causing frequent delay problems in PHP projects and hindering the government from satisfying the high housing demands. Accordingly, many researchers have examined these schedule risks to resolve these delay problems. However, none of these studies developed an effective tool for managing schedule risks of PHP by envisaging the key characteristics of schedule risks and PHP. Most of previous research regarding to the management of prefabrication construction tends to consider risks from static and isolated perspectives, despite that these risks are coherently interrelated with each other and their influence varies throughout the whole PHP process. To fill the research gaps, a hybrid dynamic model is developed in this research to evaluate and simulate the impact of identified schedule risks on the schedule performance of PHP in view of underlying interrelationships and interactions, employing the hybrid system dynamics (SD) and discrete event simulation (DES) method. The resulting hybrid model is validated through a serial of model structure tests and model behavior tests, with the use of data collected from a PHP project in Hong Kong. This study offers an in-depth understanding of how schedule performance of PHP are dynamically influenced by interrelationships and interactions underlying various schedule risk variables. The developed model not only has the benefits of ease of modifying model structure to reflect real schedule situation of PHP project, performing various risk analyses and communicating with simulation results, but also is of value of providing an experiment

platform for identifying and determining managerial and technical solutions proposed to minimize and mitigate the influence of corresponding schedule risks prior to implementation.

1. Introduction

Prefabrication has been increasingly advocated for housing production in Hong Kong because of its inherent advantages such as environment-friendly, better quality, cleaner and safer working environment (Tam et al., 2015; Tam and Hao, 2014; Hong et al., 2016; Zhang et al., 2015; Wu et al., 2016). Potential benefits may not be fully exploited if its inherent weaknesses of fragmentation, discontinuity, and poor interoperability not being mitigated. These weaknesses also entail various risks that may adversely affect the schedule performance of PHP (Li et al., 2016), which in turn will delay PHP projects and prevent the government from addressing the high housing demands.

Accordingly, researchers all over the world have examined and proposed solutions to such risks. However, the established approaches by previous studies do not manage to gain comprehensive understanding of the schedule management of PHP and prevent frequent schedule delay problems in public prefabrication housing delivery in Hong Kong (Tam et al., 2007a, 2007b; Li et al., 2014; Li et al., 2016). This might be accounted for the fact that previous studies have not comprehensively understand the key characteristics of PHP when conducting research on schedule risk management in PHP practices. These key characteristics of PHP include: (1) largely interdependent activities, that is, conventional PHP studies consider design, production, logistics, and on-site assembly as independent activities although such activities are closely interlinked and influence one another (Tam et al., 2006). Therefore, when addressing the risks that threaten the schedule performance of PHP, one must consider the interdependent nature of these activities; (2) schedule risks management of PHP is dynamic:

conventional research on the management of PHP tends to take PHP management from static perspective while the management process in reality is dynamic, which means that those analytic results will not be able change across with time to reflect actual management effect from a real time manner. The situation of frequent schedule delay in PHP in Hong Kong and lack of research envisaging the key characteristics of PHP lead to the fundamental research problem to be solved: how to effectively manage schedule risks by envisaging key characteristics of PHP to ensure timely project delivery in PHP in Hong Kong.

To better understand a complex risk management system of PHP from a holistic perspective, the schedule risks in the system as well as their dynamic interrelationships must be viewed from a dynamic perspective. This research investigates major processes within PHP, reveals the coordination structural influencing mechanism of interactions underlying various schedule risks, and provides a systematic method for assessing and simulating possible impact of schedule risks on the schedule of PHP, such that potential schedule risks can be identified, analyzed, assessed and handled prior to implementation to ensure timely project delivery in PHP in Hong Kong. The objectives of this research are presented as below: (1) To identify and analyze critical schedule risks that affect the schedule of PHP with consideration of involved stakeholders; (2) To develop a hybrid dynamic model for assessing and simulating potential impacts of the identified major schedule risks on the schedule performance of PHP; (3) To validate the developed model for building up confidence prior to simulation analysis and (4) To conduct simulation analysis to investigate possible impact of various schedule risk on the schedule performance of PHP. The model developed in this research forms an innovative tool for evaluating and simulating the schedule performance of PHP project from a dynamic point of view, with benefits of ease of modifying structure to reflect real situation, performing various sensitive analysis and communicating with simulation results more effectively.

2. Methodology

Introduced by Jay Forrester of the Massachusetts Institute of Technology in the 1960s, system dynamics (SD) utilizes a computer to understand the changes in system behavior across time (Forrester, 1968). This conceptual modeling technique has been widely employed in the literature to understand, examine, simulate, and analyze large-scale complex systems. Discrete event simulation (DES) offers great application prospects and benefits in designing and analyzing complex and interactive construction systems from dynamic point of view (Lu, 2003) as well as in quantitatively analyzing the lifecycle of building facilities (Martinez, 2009). While SD comes equipped with special analytical tools that simplify a complex system into several operable units, DES examines discretely changing state variables and the discreteevent-driven system simulation process (Law et al., 1991). Accordingly, DES has been extensively applied in project management research, such as the use of discrete event system simulation method for the schedule simulation and control in the construction management (Zhang et al., 2015), analyzing the assembly plan of prefabricated construction (Alvanchi et al. (2011) and assessing the pollutant emission during the construction process (Zhang (2015). Although discrete event simulation is demonstrated as a powerful tool for the analysis of discrete events during running processes, there are obvious deficiencies in the analysis of system interaction (Alvanchi et al., 2009). Instead, the system dynamics analysis can effectively analyze the interactions underlying in the various factors in the system. It is widely acknowledged that construction project management mainly contains two levels: strategic project management (macro level) and operational project management (micro level) (Lee et al., 2006; Pena-Mora et al., 2008~). Strategic project management focuses on resource allocation, budgeting and scheduling (Rodrigues and Bowers,1996), containing a multitude of feedbacks. Operational project management is mainly concerned about micro-level issues, such as the predecessor and successor relationship of network activity, detailed information for execution. When it comes to operational project management, SD cannot reflect the physical specifications of construction process (Alvanchi et al., 2009), as it does not generally form a work breakdown structure (WBS) of discrete sub-activities (Pena-Mora et al., 2008~). In contrast, discrete event simulation (DES), which analyzes construction process with an event-oriented view (Martin and Raffo, 2001), could effectively address this issue by utilizing network planning techniques, such as program evaluation technique and critical path method, and make up the deficiencies of network planning technology that cannot consider multiple progress impact factors (Hyari and El-Rayes, 2006; Georgy, 2008).

In this research, a hybrid SD-DES model will be developed with encapsulates SD models into each event in the DES model. SD models mainly address feedbacks, containing resource allocation, rework and any other macro phenomenon while DES mainly focuses on the construction process in terms of the predecessor and successor relationship, resource usage and any other micro variables. Generally, four steps are needed for developing the hybrid dynamic model, as shown in Fig. 1, as follows: (1) To identify major risks that affect the schedule of PHP with consideration of involved stakeholders. This step is to identify and analyze major risks that affect the schedule of PHP with consideration of involved stakeholders. These major schedule risks lay the foundation for subsequent model development; (2) To develop hybrid dynamic model for evaluating and simulating the potential impact of the identified major risks on the schedule of PHP. This part of the study develops hybrid dynamic model for evaluating and simulating the potential impact of the identified major risks on the schedule of PHP. System dynamics model and its associated attributes are encapsulated into the discrete event module to establish a hybrid dynamic model for simulating the variations in the schedule behavior of PHP. The data in this model can be exchanged between SD and DES; (3) the developed model is validated by conducting direct structure and structure-oriented behavior tests. This validation process is essential to enhance our confidence in the results of the model; (4) after validating the model, scenario analysis is conducted to evaluate the influence of devised schedule risk scenarios.

3. Identification of critical schedule risks

The first step of this research is to identify and analyze major risks that affect the schedule of PHP with consideration of involved stakeholders. These major schedule risks lay the foundation for subsequent model development. According to social network theory, PHP is

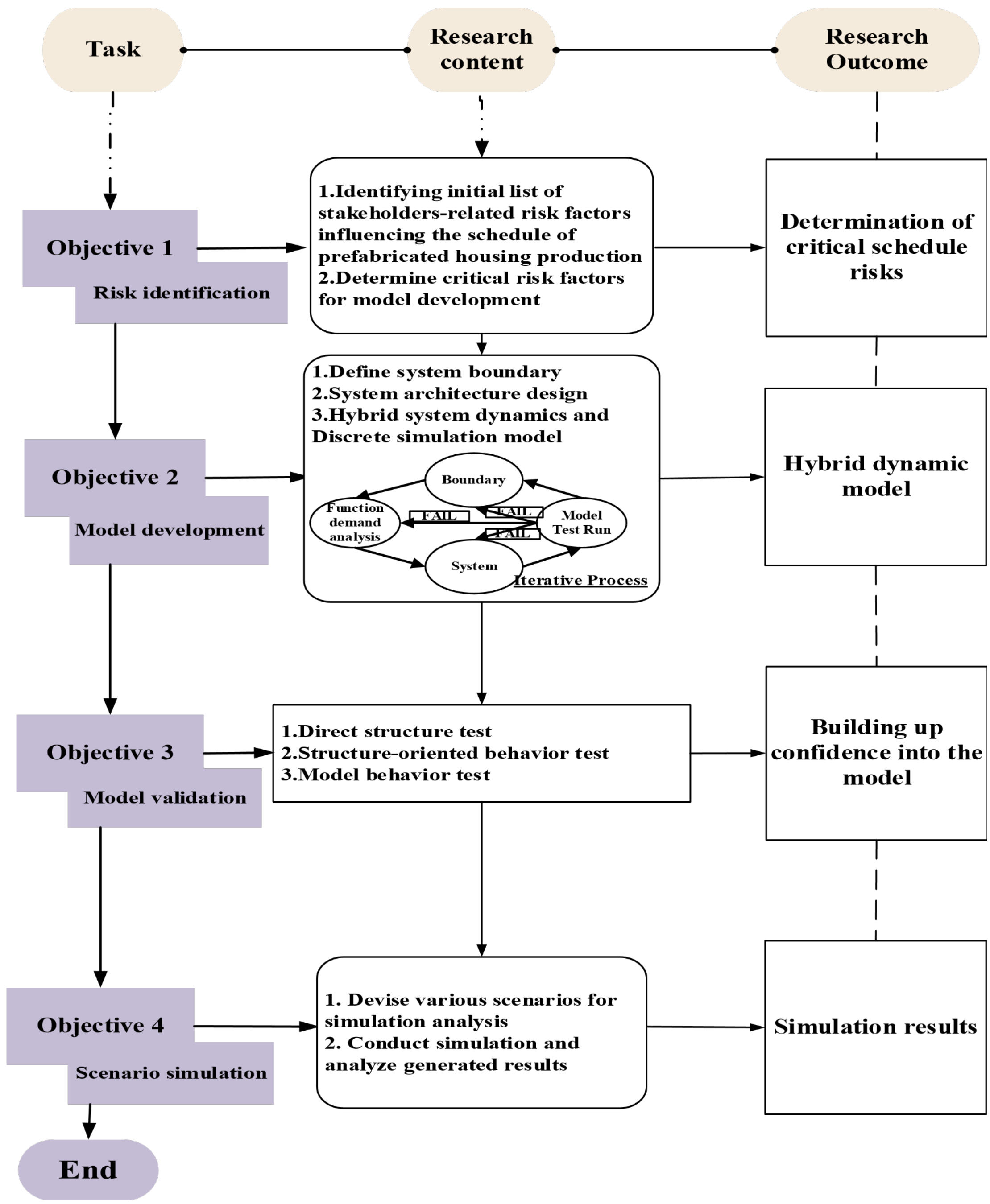


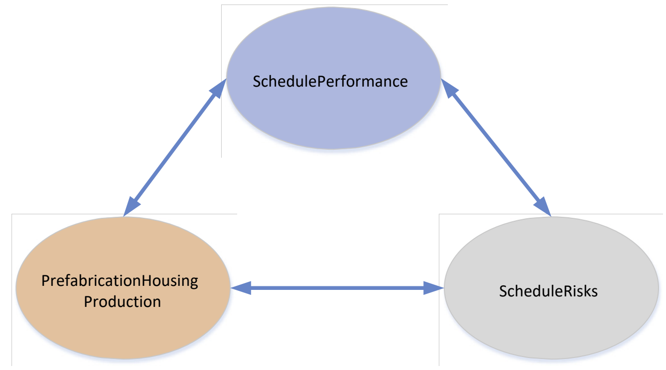
Fig. 1. Research flow.

a complex system that comprises various relationships with various stakeholders involved. Network analysis investigates those schedule risks with consideration of associated stakeholders in PHP along with corresponding cause-and-effect relations. Three steps are needed to identify critical schedule risks with social network theory. First, initial list of schedule risk and associated stakeholder that have direct influence on the schedule performance of PHP are identified. Second, the interrelationships among the identified risk factors are determined. Third, the adjacency matrix of link and node lists are introduced into NetMiner 4, which is a professional software for social network analysis, as key input data to visualize and analyze the network. The network analysis produces a list of critical stakeholder-associated schedule risks based on the node- and link-level metrics of SNA. The identification process of critical schedule risks depends on the outcomes of the SNA indicators, including status centrality, betweenness centrality, brokerage and the degree of nodes. Those risk and interrelationships with high status centrality, betweenness centrality, brokerage and the degree of nodes values are identified as critical risks have significant influence. After the analysis, twelve critical schedule risks, as shown in Table 1, are determined for further examination by the hybrid dynamic model. Please be kindly noted that given limited length, the detailed identification processes and the analysis of critical schedule risks can be referred to the published article by the authors (Li et al., 2016), while this research mainly focuses on illustration of the development of the hybrid dynamic model.

Table 1

Critical schedule risks for further examination.

Fig. 2. The relationships between the three subsystems.



Critical risks	Risk description
R1	Logistics information inconsistency because of human errors (LIIBHE)
R2	Low information interoperability between different enterprise resource planning systems (LIIBDERPS)
R3	Delay of the delivery of precast element to site (DDPES)
R4	Installation error of precast elements (IEPE)
R5	Design change (DC)
R6	Slow quality inspection procedures (SQIP)
R7	Tower crane breakdown and maintenance (TCBM)
R8	Inefficiency of design approval (IDA)
R9	Inefficient design data transition (IDDT)

R10	Design information gap between designer and manufacturer (DIGBDM)
R11	Inefficient verification of precast components because of ambiguous labels (IVPCBAL)
R12	Misplacement on the storage site because of carelessness (MSSBC)

4. Development of hybrid dynamic model

4.1. Purpose and boundary of the model

The hybrid dynamic model is a model that developed with hybrid system dynamic and discrete event simulation method, with an aim to evaluate and simulate impacts of schedule risks in PHP by considering interrelationships underlying various activities and variables from dynamic perspective. The hybrid dynamic model is expected to fulfill three main objectives. Firstly, the hybrid dynamic model allows researchers and managers involved in PHP to comprehend interrelationships underlying activities and variables within the PHP from a dynamic perspective. The model functions as an experimental platform for studying how the changes in the behavior of a system over time are related to its underlying structure and decision rules and evaluating the effects of risk on schedule of PHP. Secondly, the model provides a solid basis to analyze and discuss the possible impact of major risks on the schedule of PHP. The hybrid dynamic model enables the quantification of impact of major risks on the schedule of PHP, such that various risk portfolios that predefined by the researchers and managers can be analyzed and discussed prior to project implementation. Thirdly, the model would serve as a practical tool to explain and validate the benefits and drawbacks of specific strategies proposed to deal with identified major schedule risks prior to implementing those strategies by evaluating the devised scenarios. Once strategies are proposed by the researchers and managers for hedging potential schedule risk, the potential effect of the strategies on the schedule performance can be evaluated and simulated through the model by setting specific scenarios and the findings based on the model can be relayed to others through hands-on training to analyze the benefits and drawbacks of the strategies. Take strategy "mitigating installation error of precast elements for example, through the model, the effect of the strategy can be simulated by devising corresponding scenario to check whether it will result in enhancement on the schedule performance of PHP.

Different system boundaries will generate different system structures and behaviors. System boundaries should be defined clearly to facilitate the system modeling process as well as meet research objectives. This research divides the hybrid dynamic model into three subsystems: prefabrication supply chain subsystem, schedule risks subsystem, and schedule performance subsystem. The relationships between the three subsystems are shown in Fig. 2. The model will be concentrated on examining interrelationships of risk variables affecting schedule performance throughout PHP activities.

4.2. Development of hybrid dynamic model

4.2.1. Basic elements of DES

Based on the Unified Modeling Language provided by AnyLogic®, identified works can be dragged to work space and the relationship among different works can be defined. Consequently, DES becomes intuitive and convenient. The basic five elements of DES are "Source," "Sink," "Split," "Combine," and "Task module." "Split" can convert one entity into two entities when a task has two successors. “Combine” waits for the two entities to arrive (in arbitrary order), produces a new entity, and outputs it for use when a task has two or more predecessors. “Split” and “Combine” ensure that only one entity will flow through each task. The relationships among tasks in the construction project can be divided into “oneto-one,”“one-to-many,” and “many-to-one”, as shown in Fig. 3: (1) “one-to-one”, the relationship between Task 4 and Task 5 is “oneto-one.” When Task 4 is completed, the entity will flow out of Task 4 and flow into Task 5. Simultaneously, the SD model in Task 5 will be operational; (2) “one-to-many”, The relationship among Task 1, Task 2, and Task 3 is “one-to-many.” Task 1 is the predecessor of Task 2 and Task 3. When Task 1 is completed, the entity will flow out of it and be converted into two entities by “Split.” The two entities will flow into Task 2 and Task 3; (3) “many-to-one”, The relationship among Task 2, Task 3, and Task 4 is “one-to-many.” When Task 2 and Task 3 are completed, the two entities will be combined by “Combine,” and a new entity will be generated and flow into Task 4.

4.2.2. Encapsulation for wrapping system dynamics model into discrete event simulation module

The stock-flow diagram, as shown in Fig. 4 is a model with detailed information on system behaviors to be modelled. Relationships underlying variables that have influence on schedule performance of PHP are transformed in the stock-flow diagram for quantitative evaluation. The system dynamics model in this research can be viewed as a standardized model that is designed to depict and model the system behavior of a specific construction activity of PHP. The SD model will then be packaged into the DES module, whereas all DES modules that represent different activities of PHP will eventually be connected to form the final hybrid dynamic model. The SD model hence serves as the fundamental part of the hybrid dynamic model. The logic of SD model is supported by different functional modules, including prefabrication installation module, resource allocation module, project quantity calculation module, and schedule performance module.

Encapsulation is used to combine the behavior and properties of

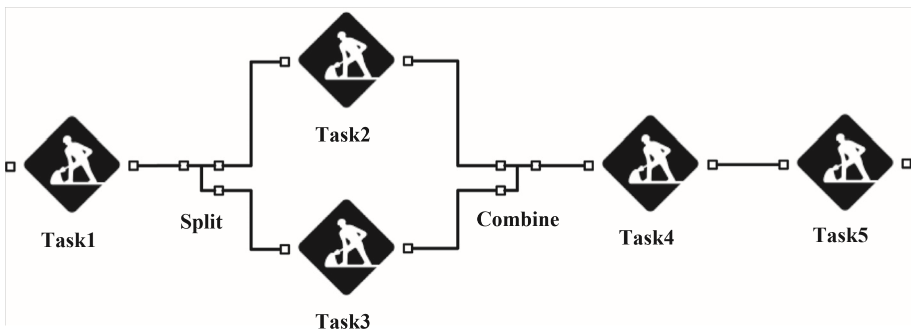


Fig. 3. Relationships among tasks in the construction project.

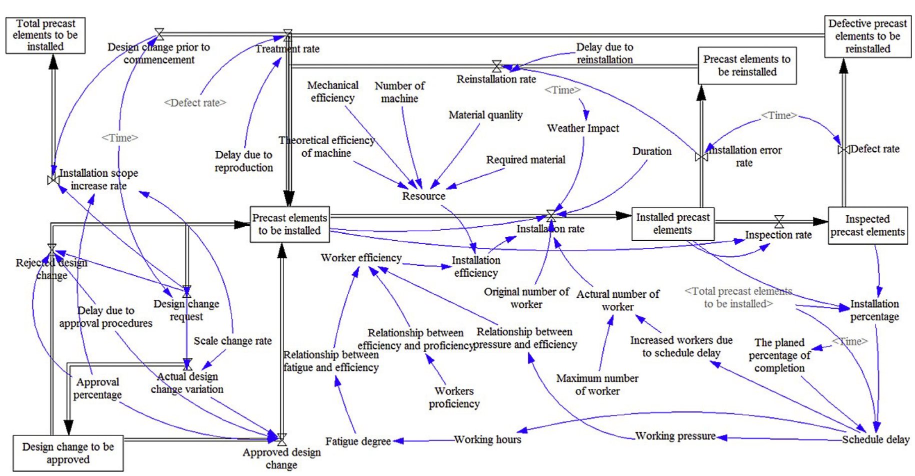


Fig. 4. System dynamics model to be encapsulated.

an object and place them in a logic unit that is responsible for hiding the properties of the object as a private property. Encapsulation does not only realize the protection of the properties of the objects but also improve the applicability and efficiency of the software. Therefore, the SD model in this study is encapsulated into a DES for constructing a task module. The task module is connected with a database that records the different data of each task. In this way, all the tasks can be derived from the task module according to the database, thereby significantly improving the efficiency of the model. The schematic of encapsulation is presented in Fig. 5. "Source" is the starting point of a process that generates entities (Borshchev, 2013). Accordingly, "Sink" is the end point of a process model that disposes entities (Borshchev, 2013). All the entities generated by "Source" will finally flow into "Sink." In this model, "Source" will generate an entity at the beginning. Once the entity flows into the inlet of a task module, the SD model in the task module will be operational. "Queue" is a unique function in DES and is based on the principle of queuing theory. The waiting time of the entity in "Queue" is equal to the running time of SD. "Hold" is used to block the entity flow along a particular connection (Borshchev, 2013). If the SD model does not stop running, then "Hold" will not allow the entity to flow out. Once the SD model stops running (i.e., the work has been completed), the entity will flow out of the task module and flow into the next task module.

As shown in the figure of the encapsulated discrete event module, system dynamic model is wrapped in the center of the figure, while other plug-ins are designed and placed along the

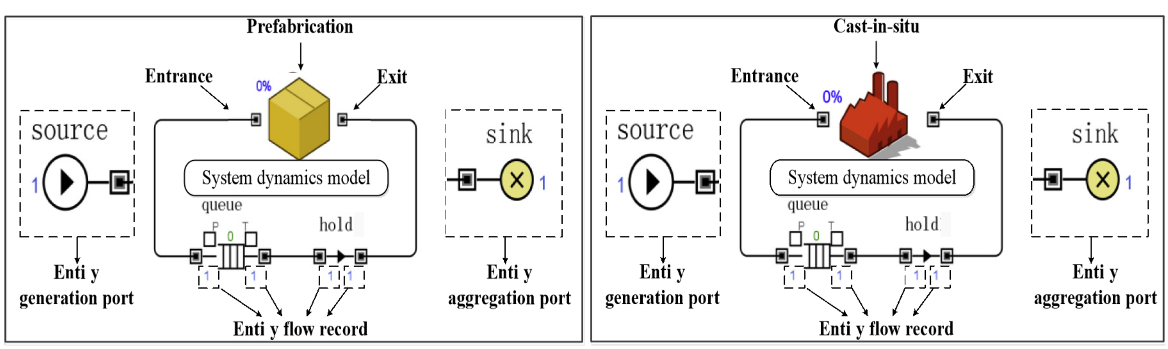


Fig. 5. Schematic of encapsulation.

model for a variety of purpose, such as data exchange among system dynamics model and discrete event module, data reading and collection, creating table function etc. All variables in the system dynamics model are connected to the database through program coding. When the model is trigger to operate, required data will be invoked from database to the encapsulated module for simulation and the generated results will also be stored in the database for further analysis. Moreover, a series of real-time visual interfaces and data output plug-ins are developed through the software to track, record and present the real time modeling process of each module in the simulation.

4.2.3. Connecting discrete event modules

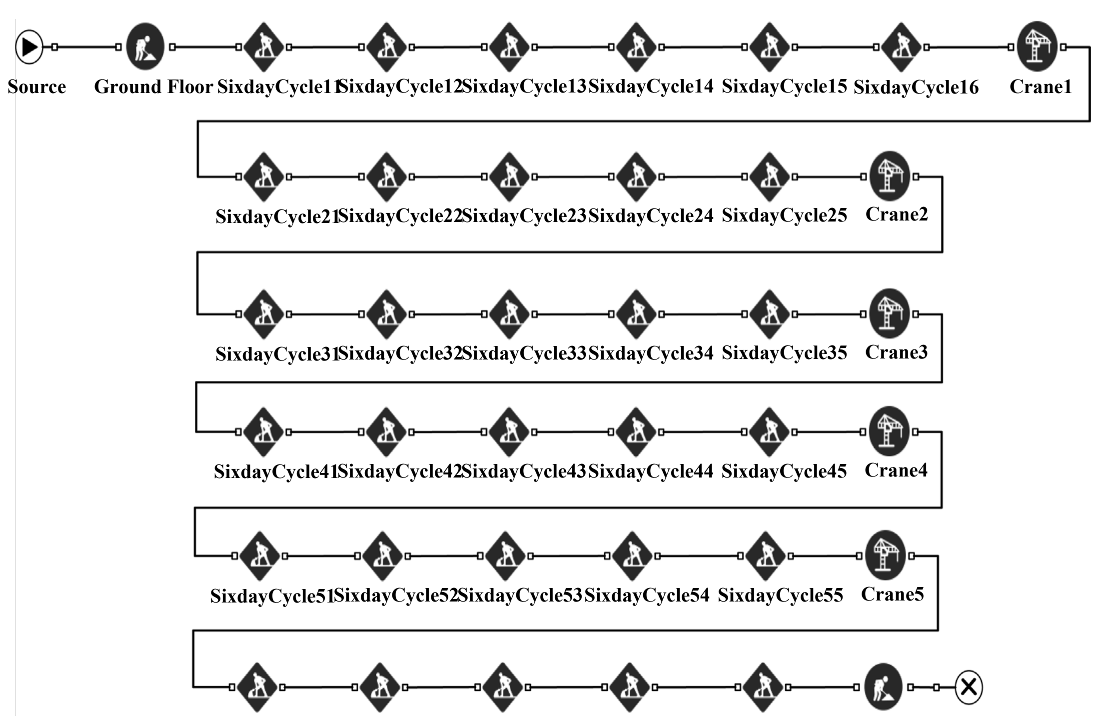
After developing the specific discrete event modules, they should be connected based on predecessor and successor relations according to the actual plan of precast structure works of PHP, which include a serial of six-day cycle installation tasks. The manager draws a plan to instruct corresponding workers to implement those tasks at different times across precast structure works of PHP. By using encapsulation technology and Unified Modeling Language provided by Anylogic software, modules are connected based on predecessor and successor relations of the master diagram of the PHP project. The connecting process can be relatively convenient because the properties of the module are previously defined via the above modeling process, such that corresponding template modules can be easily replicated to present the predecessor and successor relations rather than developing various models from a new start. The final hybrid dynamic model for precast structure works of PHP can be seen in Fig. 6.

4.2.4. Connecting the hybrid dynamic model with database

As a large amount of data are required to be input into the hybrid dynamic model and the generated simulation results are also needed to be stored for further analysis, a database should be developed and connected to the developed hybrid dynamic model. In this research, Access is taken as the database to store initial input data and simulated results. Besides, data reading and exchange plug-ins that assist the hybrid dynamic model communicate with the Access database, are developed through program coding, make it quickly read data from database, exchange required data among various modules and store simulation results generated from the model.

5. Case study

5.1. Validating the hybrid dynamic model



Source
Ground Floor
SixdayCycle11
SixdayCycle12
SixdayCycle13
SixdayCycle14
SixdayCycle15
SixdayCycle16
Crane1
SixdayCycle21
SixdayCycle22
SixdayCycle23
SixdayCycle24
SixdayCycle25
Crane2
SixdayCycle31
SixdayCycle32
SixdayCycle33
SixdayCycle34
SixdayCycle35
Crane3
SixdayCycle41
SixdayCycle42
SixdayCycle43
SixdayCycle44
SixdayCycle45
Crane4
SixdayCycle51
SixdayCycle52
SixdayCycle53
SixdayCycle54
SixdayCycle55
Crane5

SixdayCycle61 SixdayCycle62SixdayCycle63SixdayCycle64SixdayCycle65

Roof Works Sink

Fig. 6. Hybrid dynamic model for precast structure works of PHP.

Prior to further analysis, testing the verification and validation of the hybrid dynamic model, which contains model structure test and model behavior test, is crucial to build confidence into the model. Model structure test, which includes direct structure test and indirect structure test, are conducted to verify whether this research build the model right, while model behavior test is conducted to validate whether this research build the right model. Model structure test includes direct structure test and indirect structure test (Barlas, 1996; Barlas and Kanar, 2000). Direct structure test, including dimensional consistency test, boundary adequacy test, parameter confirmation test, and structure confirmation test, checks model rationality by comparing the structure of developed model with real system structure from a qualitative point of view to help calibrate the model to fit real world situations (Barlas, 1996; Lee et al., 2007). The indirect test takes the advantages of both direct structure test and quantitative test, aiming to validate the model structure indirectly by conducting various behavior tests on model behavior patterns. Indirect test includes extreme-condition test, behavior sensitivity test, and integral error test (Barlas, 1996). The selected case for validating and supporting the model in this research is a public housing project from Hong Kong. The construction practice of Block 5 is taken as a case study for the developed model, which consists of 37-story residential buildings, with the expected project duration period of 509 days.

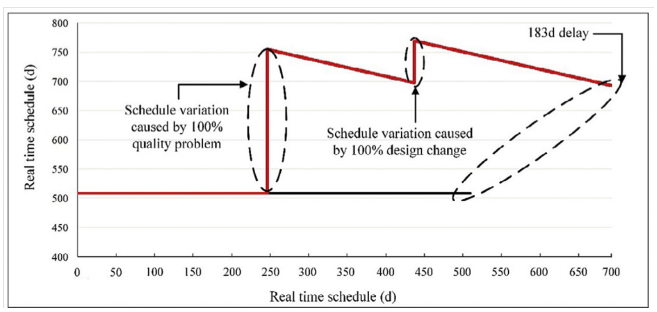
Surveys are mainly conducted toward four major involved parties in the PHP management in the project, namely Wing Hong Shun Ltd., Yingyun Transportation Ltd., and Gammon Construction Ltd., and HKHA.

5.1.1. Direct structure test

Barlas proposed two methods for model structure testing, namely, direct structure and structure-oriented behavior tests (Barlas, 1996), of which the former qualitatively evaluates the validity of the model structure by referring to knowledge regarding the actual system structure, while the latter quantitatively and indirectly evaluates the validity of the structure by subjecting model-generated behavior patterns to certain behavior tests. Four direct structure tests, namely, structure-confirmation, parameterconfirmation, boundary adequacy, and dimensional consistency tests, are often employed in the literature. Each of these tests is discussed in detail as follows: (1) The structure-confirmation test compares the causality and feedbacks of the model with the relationships in the actual system; (2) the parameter-confirmation test assesses the constant parameters via conceptual and numerical reformation based on knowledge regarding the actual system; (3) the boundary adequacy test determines whether the important variables that influence the research objectives have been included in the model; and (4) the dimensional consistency test examines the dimensional consistency between the right- and left-hand sides of each equation (Senge and Forrester, 1980).

5.1.2. Indirect structure test

The indirect structure test can be divided into (1) extremecondition, (2) behavior sensitivity, and (3) integral error tests (Balci, 1994). The extreme-condition test compares the behavior generated by the model with the anticipated behavior of the actual system under extreme conditions (Balci, 1994). Design change and quality problem are acknowledged to have a strong effect on schedule performance of PHP (Li et al., 2016). To set the extreme condition, 100% design change and 100% quality problem are set in the model. In other words, the risk is supposed to occur with the probability of 100% and have the largest influence on the schedule of PHP. Besides, a project with general risk situation is also taken as another extreme condition. Through simulation, the durations of



precast structure works are 750, 780 and 509 days as shown in Fig. 7, complying with practical experience.

The behavior sensitivity test identifies the parameters to which the model is highly sensitive (Barlas,1996). The test is performed by observing the change in model behavior through changing the variables in a reasonable range for identifying the sensitive variable. A sensitive variable is the focus for model correction given that its change would have significant effect on the schedule. In contrast, an insensitive variable does not require high accuracy because of only minor change in schedule caused by its change. Considering that demonstrating all variables is not practical because of limited space, the variables “defect rate,” “installation error rate,”“design change prior to commencement,”“scale change rate,”“delay due to reinstallation,” and “delay due to reproduction” are taken herein as examples for illustration. The five variables are assigned with three possible values, namely, the minimum value, value most likely to occur, and the maximum value, to test their potential effect on the schedule of PHP. The minimum value stands for the most optimistic value, while the maximum indicates the most pessimistic value. Take the defect rate as an example, the most optimistic scenario is that no defect exists; thus, the defect rate has the minimum value of 0. In contrast, the most pessimistic scenario is that the number of defects is the highest, in effect the defect rate has the maximum value of 0.1. After simulation, three kinds of indicators are attained, namely, duration, variety degree, and range of variation, as shown in Table 2. Variety degree involves two values, one is variety range of the minimum duration with respect to the most likely duration (the minimum value minus the most likely value and divide the most likely value); the other is the variety range of the maximum duration with respect to the most likely duration (the maximum value minus the most likely value and divide the most likely value). The sum of their absolute values is the range of variable. Twenty percent is set as the boundary (if the range of variation of a parameter exceeds 20%, it means that the parameter is a sensitive variable; otherwise, the parameter is an insensitive variable) and the scale change rate, is found to be a sensitive variable. Based on the above process, all sensitive variables, including scale change rate and delay due to reproduction, delay due to reinstallation, are found and assigned with relatively accurate values.

Fig. 7. Extreme-condition test of precast structure works.

Table 2

Parameter	Duration (min/most/max)	Variety degree (min/max)	Range of variety
Design change prior to	509.18/517.36/544.00	1.58%/5.15%	6.73%
commencement	509.18/525.09/557.18	3.03%/6.12%	9.15%
Scale change rate	509.18/553.73/702.09		34.83%

Delay due to reproduction	509.18/568.36/737.18	8.05%/26.78%	40.11%
Delay due to reinstallation	509.18/509.18/524.26	10.42%/29.70%	3.00%
Defect rate		0%/3.03%	
Installation error rate	509.18/517.27/529.64	1.56%/2.4%	3.96%

Result of sensitive test of precast structure works.

The integral error test checks whether the model behavior will change significantly with the change in integration step. Through the change in the integration step to 0.5, 0.25, 0.125, and 0.0625 day/time, the corresponding model behaviors with duration of PHP in this research are 509.25, 509.25, 510.61, and 510.93 days, indicating that the model is in line with the requirement of the integral error test (Ford and Sterman, 1998).

5.1.3. Model behavior test

After building sufficient confidence in its structure, the behavior of the model is analyzed to measure how this model can accurately replicate the behavior of an actual system (Barlas, 1996). Comparison analysis of historical data is conducted for model behavior test for precast structure works as shown in Table 3. Planned schedule is

Table 3

Model behavior test of precast structure works.

No.	Task	Planned duration (days)	Model duration (days)	Error rate
1	Ground Floor and Transfer Structure	246	246.00	0.00%
2	Six-day Cycle11	6	6.01	0.09%

3Six-day Cycle12	6	6.01	0.18%
4Six-day Cycle13	6	6.02	0.35%
5Six-day Cycle14	6	6.06	0.95%
6Six-day Cycle15	6	6.07	1.12%
7Six-day Cycle16	6	6.03	0.52%
8Crane Lift 1	1	1.01	1.09%
9Six-day Cycle21	6	6.05	0.83%
10Six-day Cycle22	6	6.02	0.36%
11Six-day Cycle23	6	6.03	0.50%
12Six-day Cycle24	6	6.06	0.98%
13Six-day Cycle25	6	6.06	0.97%
14Crane Lift 2	1	1.07	6.91%
15Six-day Cycle31	6	6.13	2.17%
16Six-day Cycle32	6	6.09	1.48%
17Six-day Cycle33	6	6.10	1.70%
18Six-day Cycle34	6	6.08	1.41%
19Six-day Cycle35	6	6.04	0.67%
20Crane Lift 3	1	1.02	2.09%
21Six-day Cycle41	6	6.19	3.20%
22Six-day Cycle42	6	6.10	1.70%
23Six-day Cycle43	6	6.04	0.68%
24Six-day Cycle44	6	6.09	1.48%

25Six-day Cycle45	6	6.02	0.32%
26Crane Lift 4	1	1.01	1.27%
27Six-day Cycle51	6	6.02	0.36%
28Six-day Cycle52	6	6.04	0.68%
29Six-day Cycle53	6	6.02	0.39%
30Six-day Cycle54	6	6.08	1.29%
31Six-day Cycle55	6	6.04	0.65%
32Crane Lift 5	1	1.05	4.91%
33Six-day Cycle61	6	6.07	1.17%
34Six-day Cycle62	6	6.01	0.11%
35Six-day Cycle63	6	6.03	0.52%
36Six-day Cycle64	6	6.07	1.14%
37Six-day Cycle65	6	6.07	1.12%
38Roof Works	72	72.26	0.36%

chosen as bench mark indicators and the model simulated variable value is compared to its historical data of planned schedule. The planned schedule is the general situation of schedule performance of a PHP project. The model behavior should comply with general situation before adding critical schedule risks for further analysis. To verify the credibility of the established model, the planned schedule is compared with the simulation results via tolerance analysis. The model shows a favorable matching effect if the variable with less than 5% relative error accounts for no less than 70% of all tested variables and if each variable has an average relative error of less than 10% (Maddala, 1986). Table 3 shows that all tested variables of the precast structure works have relative errors of lower than 10%, with an average error of 2.63% and

1.2% respectively. Therefore, the proposed model has a satisfactory matching effect and can accurately reflect actual situations. Additional simulations may be conducted to examine the influence of related scenarios on the schedule performance of PHP.

5.2. Schedule risks analysis

5.2.1. Analysis on single-risk scenarios

As presented above 12 critical schedule risks that have significant influence the schedule of PHP are determined, include IDA, DC, IDDT, LIIBHE, LIIBDERPS, DIGBDM, DDPES, MSSBC, TCBM, IVPCBAL, and SQIP. The twelve critical schedule risks are incorporated into the hybrid dynamic model for further in-depth evaluation and analysis. After 200 simulations, a bundle of curves of density function of all schedule risks can be attained. The curves of density function of PHP duration for the twelve schedule risks occurring at different stages are created and shown in Figs. 8e19. All the curves are similar to a normal distribution but are not statistically normal distributions. Based on the discussion above, using a median as benchmark duration to generally measure these risks is reasonable. Most of curves of density functions of duration have two or more

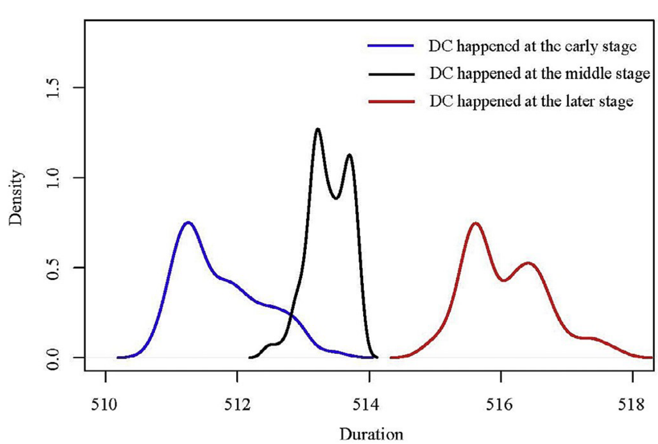


Fig. 8. The curves of density function of DC at different stages.

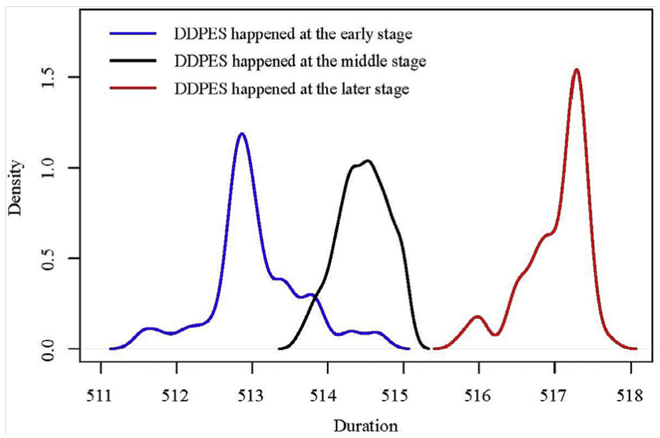


Fig. 9. The curves of density function of DDPES at different stages.

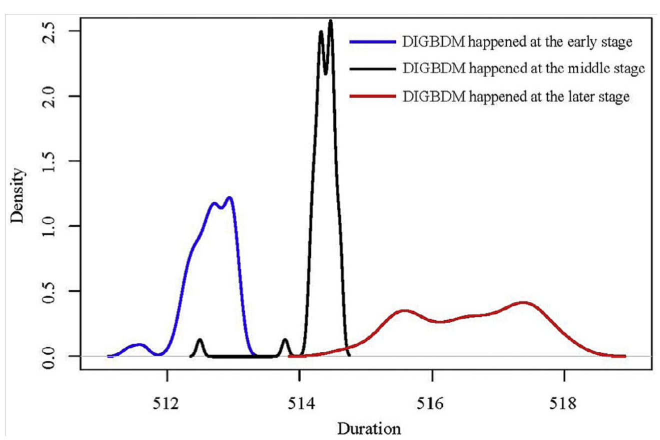


Fig. 10. The curves of density function of DIGBDM at different stages.

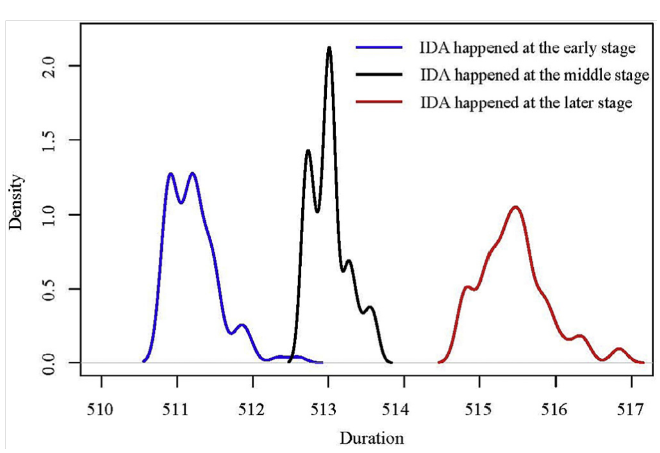


Fig. 11. The curves of density function of IDA at different stages.

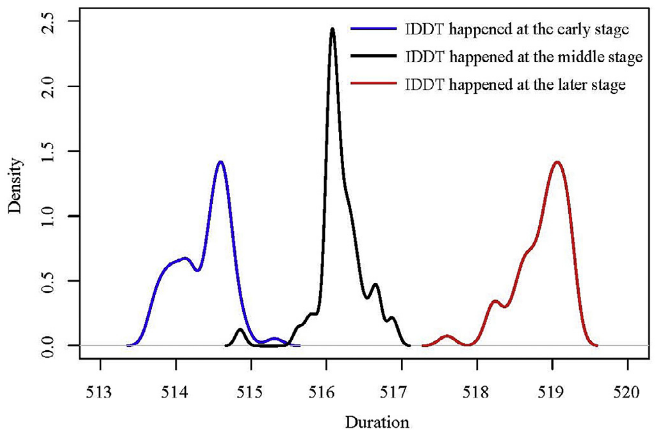


Fig. 12. The curves of density function of IDDT at different stages.

curve peaks. Generally, when the probability distribution of a risk complies with triangular distribution and the "most possible delay" is closer to the "maximum delay", the additional small peaks are

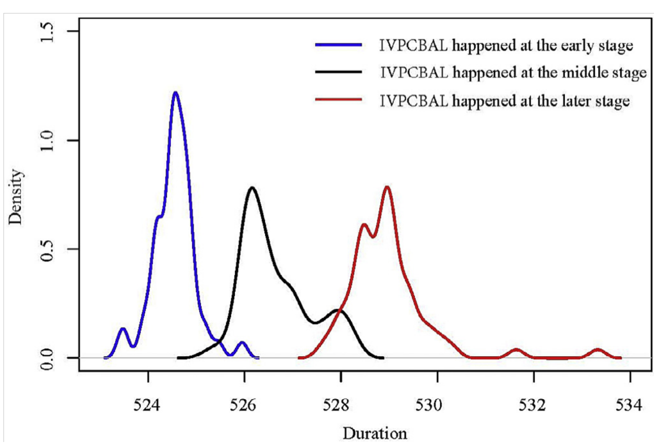


Fig. 14. The curves of density function of IVPCBAL at different stages.

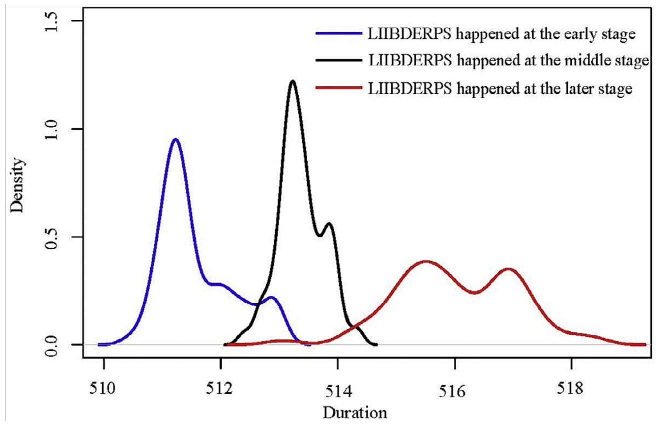


Fig. 15. The curves of density function of LIIBDERPS at different stages.

more likely to appear.

For DC in the simulation of precast structure works as shown in Fig. 8, if DC occurs at the early stage, the rang of schedule ends at

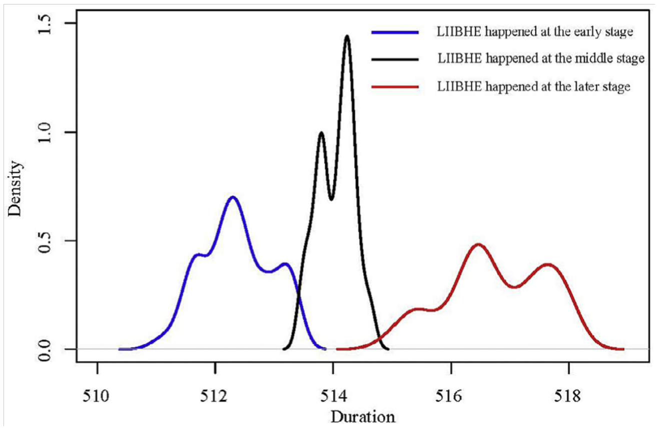


Fig. 16. The curves of density function of LIIBHE at different stages.

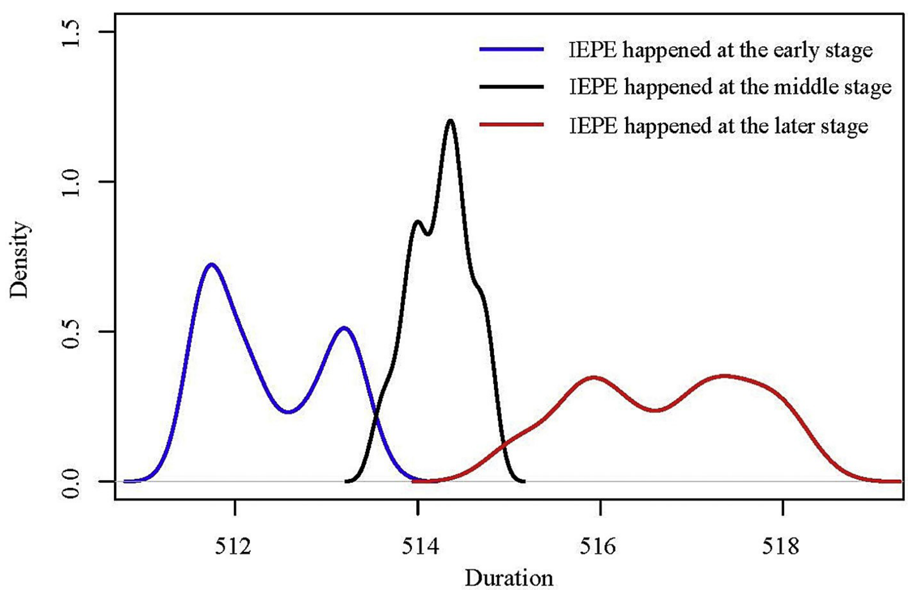


Fig. 13. The curves of density function of IEPE at different stages.

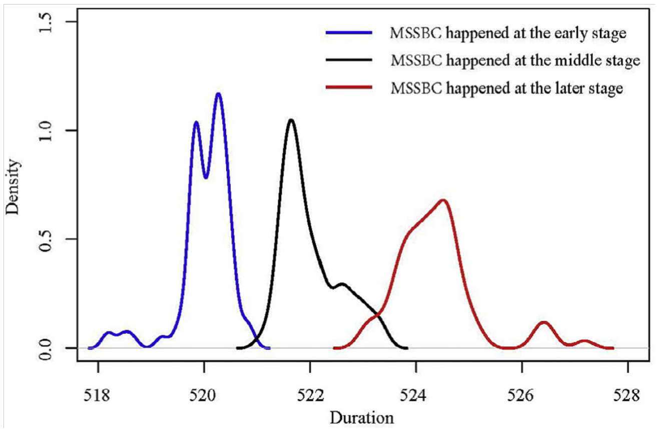


Fig. 17. The curves of density function of MSSBC at different stages.

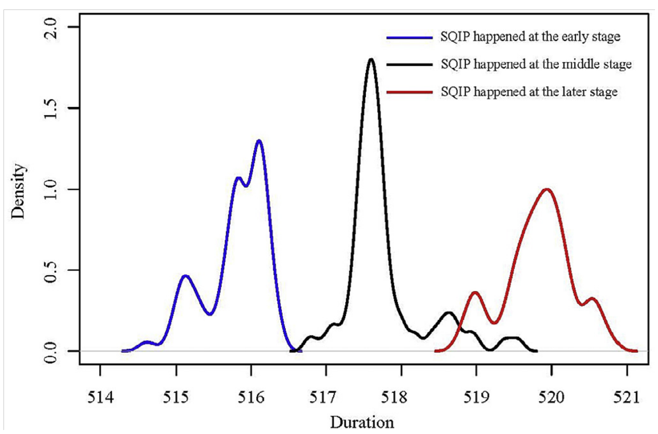


Fig. 18. The curves of density function of SQIP at different stages.

(510.78,513.45), while at the middle and later stage, the schedule respectively ends at the range (512.47, 513.84) and (514.91, 517.71). The median schedule as a result of DC for early stage, middle stage and later stage are 511.58, 513.4 and 516, while the maximum schedule for the three periods are 513.45, 513.84 and 517.71.

For DDPES in the simulation of precast structure works as shown in Fig. 9, if DDPES occurs at the early stage, the rang of schedule ends at (511.55, 514.65), while at the middle and later stage, the schedule respectively ends at the range (513.67, 515.02) and (515.76, 517.71). The median schedule as a result of DC for early stage, middle stage and later stage are 512.91, 514.49 and 517.22, while the maximum schedule for the three periods are 514.65, 515.02 and 517.71.

For DIGBDM in the simulation of precast structure works as shown in Fig. 10, if DIGBDM occurs at the early stage, the rang of schedule ends at (511.42, 513.04), while at the middle and later stage, the schedule respectively ends at the range (512.49, 514.62) and (514.69, 518.07). The median schedule as a result of DIGBDM for early stage, middle stage and later stage are 512.73, 514.38 and 516.61, while the maximum schedule for the three periods are 513.04, 514.62 and 518.07.

For IDA in the simulation of precast structure works as shown in Fig. 11, if IDA occurs at the early stage, the rang of schedule ends at (510.87, 512.60), while at the middle and later stage, the schedule respectively ends at the range (512.69, 513.62) and (514.78, 516.84). The median schedule as a result of IDA for early stage, middle stage and later stage are 511.20, 513.02, and 515.43, while the maximum schedule for the three periods are 511.23, 513.01 and 515.45.

For IDDT in the simulation of precast structure works as shown in Fig.12, if IDDT occurs at the early stage, the rang of schedule ends at (513.69, 515.31), while at the middle and later stage, the schedule respectively ends at the range (514.85, 516.91) and (517.60, 519.27). The median schedule as a result of IDDT for early stage, middle stage and later stage are 514.45, 516.15 and 518.95, while the maximum schedule for the three periods are 515.31, 516.91 and 519.27.

For IEPE in the simulation of precast structure works as shown in Fig.13, if IEPE occurs at the early stage, the rang of schedule ends at (511.42, 513.58), while at the middle and later stage, the schedule respectively ends at the range (513.53, 514.85) and (514.85, 518.38). The median schedule as a result of IEPE for early stage, middle stage and later stage are 512.16, 514.27 and 516.89, while the maximum schedule for the three periods are 513.58, 514.85 and 518.38.

For IVPCBAL in the simulation of precast structure works as shown in Fig. 14, if IVPCBAL occurs at the early stage, the rang of schedule ends at (523.44, 525.95), while at the middle and later stage, the schedule respectively ends at the range (525.29, 528.20) and (527.60, 533.33). The median schedule as a result of IVPCBAL for early stage, middle stage and later stage are 524.56, 526.40 and 528.93, while the maximum schedule for the three periods are 525.95, 528.20 and 533.33.

For LIIBDERPS in the simulation of precast structure works as shown in Fig. 15, if LIIBDERPS occurs at the early stage, the rang of schedule ends at (510.93, 513.33), while at the middle and later stage, the schedule respectively ends at the range (513.45, 514.65) and (514.87, 518.13). The median schedule as a result of LIIBDERPS for early stage, middle stage and later stage are 511.33, 513.37 and 516.04, while the maximum schedule for the three periods are 512.96, 514.33 and 518.31.

For LIIBHE in the simulation of precast structure works as shown in Fig. 16, if LIIBHE occurs at the early stage, the rang of schedule ends at (510.93, 513.33), while at the middle and later stage, the schedule respectively ends at the range (513.53, 514.85) and (514.87, 518.13). The median schedule as a result of LIIBHE for early stage, middle stage and later stage are 512.36, 514.05 and 516.75, while the maximum schedule for the three periods are 513.33, 514.65 and 518.13.

For MSSBC in the simulation of precast structure works as shown in Fig. 17, if MSSBC occurs at the early stage, the rang of schedule ends at (518.18, 520.87), while at the middle

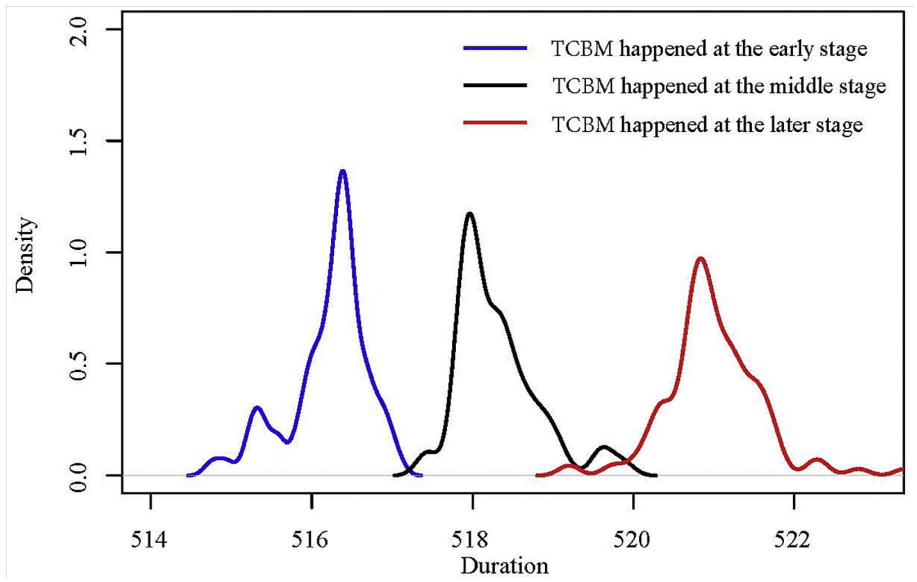


Fig. 19. The curves of density function of TCBM at different stages.

and later stage, the schedule respectively ends at the range (521.13, 523.33) and (522.98, 527.18). The median schedule as a result of MSSBC for early stage, middle stage and later stage are 520.16, 521.80 and 524.29, while the maximum schedule for the three periods are 520.87, 523.33 and 527.18.

For SQIP in the simulation of precast structure works as shown in Fig.18, if SQIP occurs at the early stage, the rang of schedule ends at (514.62, 516.35), while at the middle and later stage, the schedule respectively ends at the range (516.75, 519.58) and (518.82, 520.76). The median schedule as a result of SQIP for early stage, middle stage and later stage are 515.85, 517.62 and 519.86, while the maximum schedule for the three periods are 516.35, 519.58 and 520.76.

For TCBM in the simulation of precast structure works as shown in Fig. 19, if TCBM occurs at the early stage, the rang of schedule ends at (514.76, 517.05), while at the middle and later stage, the schedule respectively ends at the range (517.44, 519.87) and (519.20, 524.96). The median schedule as a result of TCBM for early stage, middle stage and later stage are 516.35, 518.21 and 520.94, while the maximum schedule for the three periods are 517.05, 519.87 and 524.96.

In general, risks IVPCBAL, MSSBC, TCBM and SQIP have more significant influence on schedule performance of PHP project, followed by IDDT, DDPES, DIGBDM, IEPE, LIIBHE, DC, LIIBDERPS and IDA in terms of mean schedule. Besides, risks occurring at the later stage tend to have more significant influence on the schedule performance of PHP than occurring at early and middle stage of precast structure works of PHP. The possible reason accounting for this phenomenon might be that when schedule risks occur at later stage, there might be no enough time for the manager to deal with the risks and implement work expediting activities to make up the time loss caused by the occurred risks and thus leading to more serious schedule delay.

5.2.2. Analysis on multi-risk scenarios

Four risks, namely, IVPCBAL, MSSBC, TCBM, and SQIP, and their combined effects are subjected to a scenario analysis that involves a base case scenario and two modified scenarios (i.e., the risk decreased and increased by 50%) to identify the influence of schedule risks under different situations. Two criteria, namely, influence and likelihood, are used to measure the risk. When the risk decreases by 50%, both its influence and likelihood are reduced by 50%. When the risk with a 0.4 likelihood to affect schedule performance decreases by 50% and complies with triangle distribution, the three values for triangle distribution are all reduced by 50%, while the probability of occurrence is reduced by 50% to 0.2.

5.2.2.1. Scenario A. In scenario A, the initial value (A1) of IVPCBAL decreases by 50% (A2) and then increases by 50% (A3). Table 4 present the simulation results. The width of the range in A2 (1.9) and A3 (2.19) decrease by 24.28% and 87.32%, respectively, thereby indicating that such width is more sensitive to the increase than the decrease of IVPCBAL. By contrast, the average delay in A3 (increased by 71.79%) is more sensitive than that in A2 (decreased by 44.67%).

5.2.2.2. Scenario B. In scenario B, the widths of the range in B2 (1.6) and B3 (3.8) decrease by 40.54% and 41.22%, respectively. Therefore, such width demonstrates equal sensitivity to both the increase and decrease of MSSBC. Moreover, the average delay in B2 (decreased by 42.00%) is less sensitive than that in B3 (increased by 81.27%).

5.2.2.3. Scenario C. In scenario C, the widths of the range in C2 (1.6) and C3 (3.8) decrease by 30.16% and increase by 65.87%, respectively. Table 4 shows that the average delay in C2 (4.4) decreases by 38.95%, while than in C3 (14.5) increases by 101.19%. Both TCBM and IVPCBAL almost demonstrate the same trend, and the width of the range is more sensitive to the increase than the decrease of TCBM. Moreover, the average delay in C2 (decreased by 38.95%) is less sensitive than that in C3 (increased by 101.19%).

5.2.2.4. Scenario D. In scenario D, the widths of the range in D2 (1.9) and D3 (4.5) decrease by 10.00% and increase by 160.52%, respectively. These trends show higher and lower sensitivities than the three aforementioned scenarios. As shown in Table 4, the average delay in D2 decreases by 44.02%, while that in D3 increases by 101.83%. These results underscore the importance of controlling the increase in SQIP, which will significantly extend the duration.

5.2.2.5. Scenario E. In scenario E, the widths of the range in E2 (28.3) and E3 (34.4) decrease by 4.71% and increase by 15.83%, respectively. As shown in Table 4, the average delay in E2 (36.4) decreases by 41.57%, while that in E3 (90.2) increases by 44.78%. In sum, IVPCBAL offers the greatest contribution and effectively reduces the effect of the combination.

Table 4

Statistical information of duration under different risks.

Devised Scenarios	Category	Mean	Median	Range	Standard deviation
Scenario A	IVPCBAL	517.6	517.5	516.7e518.6	0.39
Scenario B	(50%)	535.7	535.3	533.6e538.3	1.08
	IVPCBAL	515.4	515.3	514.7e516.3	0.31
Scenario C	(þ50%)	529.0	528.9	527.7e531.5	0.85
Scenario D	MSSBC	513.4	513.4	512.8e514.4	0.30
Scenario E	(50%)	523.5	523.3	522.0e525.8	0.79
	MSSBC	512.8	512.8	511.9e513.8	0.37
	(þ50%)	522.7	522.5	521.0e525.5	0.86
	TCBM	545.4	544.9	531.7e560.0	7.62
	(50%)	571.3	571.3	557.6e587.3	8.11
	TCBM				
	(þ50%)				

SQIP (50%)				
SQIP (þ50%)				
Combination (50%)				
Combination				
Combination (þ50%)	599.2	599.5	581.2e615.6	8.60

The top four schedule risks can be categorized into weak and strong levels. The weak level includes MSSBC, TCBM, and SQIP. If the risks in this category are decreased by 50%, then the average schedule delays span from 3 days to 8 days, while the schedule ranges between 511 days and 517 days. By contrast, if these risks are increased by 50%, then the average schedule delays span from 13 days to 20 days, while the schedule ranges between 521 days and 532 days. The risk at the strong level, IVPCBAL, greatly contributes to the schedule delay. In other words, even if IVPCBAL decreases by 50%, the average schedule delay (8.6) and ranges of schedule (516.7 and 518.6) remain larger than those at the weak level. By contrast, if IVPCBAL increases by 50%, then the average schedule delay (26.7) and ranges of schedule (533.6 and 538.3) are still much greater than those at the weak level. The average delays of E1, E2, and E3 are 34.88%, 32.26%, and 16.96% larger than the simple sum of A1, B1, C1, and D1. The scenario simulation shows that these schedule risks interrelate and interact with one another. Compared with the simple sum of the single separated risk, multiple schedule risks generate a stronger integrated effect on the schedule performance of PHP, thereby validating showing an amplified effectiveness.

In summary, the above simulation results demonstrate that (1) schedule risks are not isolated, with interrelationships and interactions existing among different schedule risks; (2) the degree of the influence on schedule performance varies across the timeline of the project. Generally, the later the occurrence time of risks, the more significant influence the risks would have on the schedule performance of PHP; (3) compared with the simple sum of the single separated risk, multiple schedule risks generate a stronger integrated effect on the schedule performance of PHP. In this case, the PHP system operates in a highly iterative manner in such a way that a verified variable within the system may improve a blown-up feedback loop and amplify its effectiveness. This "systemic" behavior must be investigated further to provide managers with a valuable perspective. These managers must also consider the combined effect of the possible risk mitigation measures to achieve their anticipated performance. The process of devising scenarios of risk mitigation solution offers specific guides on proposing simulation scenarios for the hybrid dynamic model. The simulation results are informative in facilitating promising solutions for mitigating schedule risks and enhancing schedule performance of the PHP project.

6. Conclusion

The hybrid dynamic model developed by integrating SD and DES can overcome the limitations of previous research on risk evaluation, serving as effective and applicable tool for evaluating the influence of various risks on the schedule performance of a PHP project. Using a hybrid of the SD and DES approaches can effectively depict, model, and simulate the interrelationships among the underlying activities and variables within the PHP. Moreover, hybrid SD and DES approach enables the presence of changes in system behavior over time from a dynamic perspective. DES carefully models a system to reveal the dynamic behavior of each component, while SD uses aggregated data to build a model that can facilitate a system-wide dynamic flow analysis of managerial decisions. Combining these two techniques can effectively evaluate the stability of the system along with the long-term effects of system policies and short-term operational procedures, making the model to simulate how the schedule performance of the system of PHP dynamically change throughout the simulation period in a more effective way. The strict validation process demonstrates that the model is robust and applicable for evaluating the impact of schedule risks in PHP project. Though conducting simulation analyses for a series of devised scenarios, the hybrid dynamic model demonstrates its capacity of serving as an experimental platform to model and simulate the effects of different managerial and technical solutions on the overall schedule performance of PHP throughout the whole supply chain, such that the best solutions can be identified in advance.

The limitations of the hybrid dynamic model developed in the research are outlined for its further development and broader application. First, due to great amount of interrelationships among the activities and identified variables having influence on schedule performance of PHP project, it is not realistic to comprehensively investigate and build all possible dynamic interactions into the model. Moreover, due to limited length and time, it is also not practical to devise, model and simulate all possible scenarios to analyze various schedule risks and evaluate all possible impacts of measures for mitigating risks. Last, due to the limitations of resource, this research only applies the developed model to only one practical project for validation and building up confidence. But the validation case is representative enough as the case is from the public housing project, and in Hong Kong, almost half of houses are public houses that constructed by the public sector such as Hong Kong Housing Authority. Besides, as mentioned above, public housing production often encounter schedule delay problems despite the promise of the government to supply more houses, and for this studied case, the master program has been revised 7 times, it is the typical case that has serious schedule delay problems. For these two points of view, the surveyed case has representativeness. Despite of the above outlined limitations, the research not only pioneers on evaluating the impact of schedule risks on schedule performance of PHP from a new perspective, but also serving as a solid basis for further research.

References

Alvanchi, A., Azimi, R., Lee, S., AbouRizk, S.M., Zubick, P., 2011. Off-site construction planning using discrete event simulation. J. Archit. Eng. 18 (2), 114e122.

Alvanchi, A., Lee, S., AbouRizk, S., 2009. Modeling architecture for hybrid system dynamics and discrete event simulation. In: Paper Read at Proceedings of the Construction Research Conference.

Balci, O., 1994. Validation, verification, and testing techniques throughout the life cycle of a simulation study. Ann. Oper. Res. 53 (1), 121e173.

Barlas, Y., 1996. Formal aspects of model validity and validation in system dynamics. Syst. Dyn. Rev. (Wiley) 12 (3), 183e210.

Barlas, Y., Kanar, K., 2000. Structure-oriented behavior tests in model validation. In: 18th International Conference of the System Dynamics Society, Norway, Bergen.

Borshchev, A., 2013. The Big Book of Simulation Modeling: Multimethod Modeling with AnyLogic 6. AnyLogic North America, Chicago.

Ford, D.N., Sterman, J.D., 1998. Dynamic modeling of product development processes. Syst. Dyn. Rev. 14 (1), 31e68.

Forrester, J.W., 1968. Industrial dynamics-after the first decade. Manag. Sci. 14 (7), 398e415.

Georgy, M.E., 2008. Evolutionary resource scheduler for linear projects. Autom. Constr. 17 (5), 573e583.

Hong, J., Shen, G.Q., Mao, C., Li, Z., Li, K., 2016. Life-cycle energy analysis of prefabricated building components: an inputeoutput-based hybrid model. J. Clean. Prod. 112, 2198e2207.

Hyari, K., El-Rayes, K., 2006. Optimal planning and scheduling for repetitive construction projects. J. Manag. Eng. 22 (1), 11e19.

Law, A.M., Kelton, W.D., Kelton, W.D., 1991. Simulation Modeling and Analysis, vol. 2. McGraw-Hill New York.

Lee, S.H., Pen~a-Mora, F., Park, M., 2006. Dynamic planning and control methodology for strategic and operational construction project management. Autom. Constr. 15 (1), 84e97.

Lee, S.H., Pen~a-Mora, F., Park, M., 2007. Dynamic planning and control methodology for strategic and operational construction project management. Autom. Construct. 15 (1), 84e97.

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. J. Clean. Prod. 134, 482e494. Part B.

Li, Z., Shen, G.Q., Xue, X., 2014. Critical review of the research on the management of prefabricated construction. Habitat Int. 43, 240e249.

Lu, M., 2003. Simplified discrete-event simulation approach for construction simulation. J. Constr. Eng. Manag. 129 (5), 537e546.

Martin, R., Raffo, D., 2001. Application of a hybrid process simulation model to a software development project. J. Syst. Softw. 59 (3), 237e246.

Martinez, J.C., 2009. Methodology for conducting discrete-event simulation studies in construction engineering and management. J. Constr. Eng. Manag. 136 (1), 3e16.

Pen~a-Mora, F., Han, S., Lee, S., Park, M., 2008. Strategic-operational construction management: hybrid system dynamics and discrete event approach. J. Constr. Eng. Manag. 134 (9), 701e710.

Rodrigues, A., Bowers, J., 1996. System dynamics in project management: a comparative analysis with traditional methods. Syst. Dyn. Rev. 12 (2), 121e139.

Senge, P.M., Forrester, J.W., 1980. Tests for building confidence in system dynamics models. TIMS Stud. Manag. Sci. 14, 209e228.

Tam, V.W.Y., Fung, I.W.H., Sing, M.C.P., Ogunlana, S.O., 2015. Best practice of prefabrication implementation in the Hong Kong public and private sectors. J. Clean. Prod. 109, 216e231.

Tam, V.W., Hao, J.J., 2014. Prefabrication as a mean of minimizing construction waste on site. Int. J. Constr. Manag. 14 (2), 113e121.

Tam, V.W., Tam, C., Ng, W., 2007a. An examination on the practice of adopting prefabrication for construction projects. Int. J. Constr. Manag. 7 (2), 53e64.

Tam, V.W., Tam, C.M., Zeng, S., Ng, W.C., 2007b. Towards adoption of prefabrication in construction. Build. Environ. 42 (10), 3642e3654.

Tam, W., Tam, C., Chan, W., Ng, C., 2006. Cutting construction wastes by prefabrication. Int. J. Constr. Manag. 6 (1), 15e25.

Maddala, G.S., 1986. Limited-dependent and Qualitative Variables in Econometrics. Cambridge University Press.

Wu, Z., Shen, L., Yu, A.T.W., Zhang, X., 2016. A comparative analysis of waste management requirements between five green building rating systems for new residential buildings. J. Clean. Prod. 112, 895e902. Part 1.

Zhang, H., 2015. Discrete-event simulation for estimating emissions from construction processes. J. Manag. Eng. 31 (2), 0401e4034.

Zhang, X., Skitmore, M., De Jong, M., Huisingh, D., Gray, M., 2015. Regenerative sustainability for the built environment e from vision to reality: an introductory chapter. J. Clean. Prod. 109, 1e10.