24th International Symposium on Automation & Robotics in Construction (ISARC 2007)
Construction Automation Group, I.I.T. Madras

MODELING DESIGN ITERATION USING DSM AND SIMULATION

J. Uma Maheswari
Young Scientist
Department of Civil Engineering
Indian Institute of Technology Madras
Chennai – 600 036, India
uma@smail.iitm.ac.in

Koshy Varghese
Professor
Department of Civil Engineering
Indian Institute of Technology Madras
Chennai – 600 036, India
koshy@iitm.ac.in

ABSTRACT

DSM (Design/Dependency Structure Matrix) is a potential tool for planning iterations and also for evaluating alternate sequences. The standard DSM operations can determine the basic sequence and can identify the existence of cycles/loops. These cycles/loops are grouped as blocks. There can be many strategies for executing the activities within the block. The strategy proposed in the present study is 'speed' of execution/iteration. It happens when an activity waits for information from all its predecessors or when it proceeds as soon as new information arrives from any one of its predecessor. The choice on the 'speed' is decided based on the (a) estimation of duration and (b) number of repetition of the activities. Stroboscope, a Monte Carlo simulation engine is attempted to model the proposed strategy. The scope of the present study is limited to time aspects alone and the concepts are elaborated with an example.

KEYWORDS

DSM, Simulation, Basic Sequence, Execution Sequence, Blocks

1. INTRODUCTION

The execution phase of a construction project accounts for a major portion of the project cost and hence most of the planning and management efforts were focused on this phase [1]. However, the decisions made during the design phase have a greater impact on the schedule of the project. This design phase is a process of information exchange among various design domains and is generally iterative in nature [2], [3], [4]. Hence, planning for the design phase is critical.

The basic requirements for design planning include representation of the activities and dependency relationships among the activities. The information dependency between the activities is of three types – independent, dependent, and interdependent or loops [2]. The presence of cycles/loops requires that an assumption has to be made to start/proceed. When the activities start to execute with these assumptions, it is likely that these assumed information might change. Any changes made in the initial assumptions will prompt the succeeding activities to repeat until the desired criterion is met. This periodic repetition of the cyclic activities is termed as iteration [2], [4].

Conventional tools like CPM (Critical Path Method) or PERT (Program Evaluation and Review Technique) are not suitable for sequence analysis because they cannot model iteration [5]. Although advanced graph-based techniques such as GERT (Graphical Evaluation and Review Technique), IDEF (Integration Definition or ICAM Definition), Petri Nets, etc. can represent feedback loops, they are practically limited by the size of the project [6], [7]. Moreover, they cannot explicitly/ distinctly model iterations [6], [7].

Researchers have investigated DSM (Design/Dependency Structure Matrix) as a potential tool to model iterations and to evaluate the resulting sequence [2], [3], [8]. The basic DSM operations such as partitioning and tearing enable the planners to identify the overall sequence along with determination of cycles/loops. This sequence is referred as basic sequence and these cycles/loops if grouped together are denoted as blocks [9].

For a given block, there can be multiple strategies for execution. The available iteration models do not allow any variation in the strategy for executing the basic sequence [2], [4], [9]. In the present study, a strategy of execution based on "Speed" of iteration is defined. "Speed" is based on whether an activity waits for new information from all predecessors before repeating or repeats as soon as new information arrives from any predecessor. For a given block it is not apparent which strategy will result in a shorter duration. The duration depends on the iterative relationships and its dependency attributes. As there are multiple cyclic relationships and probabilistic attributes in a block, models for estimating the duration is not simple.

The objective of this work is to develop a simulation-based model to estimate the duration for a specified execution strategy. To address this objective, Monte Carlo Simulation is investigated. The results of the Stroboscope model are analyzed and discussed. This paper also illustrates the application of the above concept using an example.

2. PAST WORK

The existing iteration models based on DSM are elaborated in this section. Gaps identified from the literature review follows the model types.

The iteration models applied to construction industry treat iteration as a black-box and follow CPM/PERT type scheduling approaches [10], [11], [12]. In Taiwan, researchers have used Stroboscope for modelling iterations in construction projects [13]. They estimate the project duration for no iteration and one-time iteration. Still, 'true' iterations are not modelled for construction applications.

The analytical models cited in the literature such as parallel iteration model and sequential iteration model were applied initially in manufacturing applications [14], [15]. As the real world system is more complex, these models required more assumptions. Browning & Eppinger [16] developed the first DSM-based simulation model that analyzes design iterations with more generalizations than the previous analytical models. The work policy which governs the successive execution assumes that all activities have to wait for all the information to arrive before starting the next activity. The model fails to account for the number of times iteration occurs.

In general, the applications cited above do not allow any variation in the strategy for executing the basic sequence. As stated earlier, there can be multiple strategies for executing the activities within the block. Variation in the strategy allows for more options. Hence, the focus of the present study is to strategically execute the basic sequence for the DSM blocks.

3. PROPOSED APPROACH - 'SPEED' OF ITERATION

The proposed execution strategy depends on the information or assumptions required to start/execute any activity [9], [17]. Assumptions made through tearing process have to be checked when updated information becomes available. During the process of checking, if the assumption made was not appropriate, the succeeding activities have to be repeated. If an activity is repeated whenever new information from any one of its predecessors is available, then it is 'Fast' speed of iteration. When the activities repeat as soon as new information arrives, more resources are expected for this case. Instead, if an activity waits for the information from all its predecessors and then repeats, it is referred as 'Slow' speed of iteration. 'Speed' of iteration arises, when; at least one of the activities in the block requires multiple information inputs. When all the activities in the block require single information input, 'fast' and 'slow' process will be the same.

The concept of 'speed' is illustrated in Figure 1. In this case, C is the start activity and when C is completed; activities A and E will be executed. Then, F will be executed. For 'slow' strategy, if the assumption for C was inappropriate, C will repeat after the completion of A & F as seen in Figure 1(b). When the repetition of activities within the block is completed, information is passed on to activities outside the block. On the other hand, for 'fast' strategy, as soon as A is completed, C can repeat in parallel along with the execution of F, as in Figure 1(c). Here, information will be passed on simultaneously to activities outside the block also. Hence, more repetitions can occur within and outside the block for 'fast' if initial assumptions were made inappropriate [9], [17].

Here, C-AE-FB-D forms the basic sequence and can be determined with the help of standard DSM operations (partitioning & tearing). The 'execution sequence' is C-AE-F-C-AE-F-C-AE-F…BD for slow and C-AE-CFB-AECD-… for fast respectively.

The performance of 'fast' or 'slow' strategy can be evaluated by estimating its duration. As project duration is a function of the dependency relationships, number of assumptions, location of assumptions, etc. it is difficult to judge which 'speed' is better. Hence, there is a need for software tool to estimate the project duration for both the strategy and is presented next. The subsequent section elaborates on the Stroboscope model for estimating the duration.

4. MODEL IMPLEMENTATION

For simplicity, an illustration as shown in Figure 2 is explained throughout this section.

4.1. Parameters Involved – Inputs

Apart from the basic inputs to any DSM such as list of activities, and dependency relationship among the activities, the present model uses two other inputs – Rework probability values and probabilistic duration to estimate the project duration.

4.1.1. Rework probability values

These are numbers between '0'-'1' and can be included along the off-diagonal cells instead of 'X' marks [16], [17]. These values decide the choice

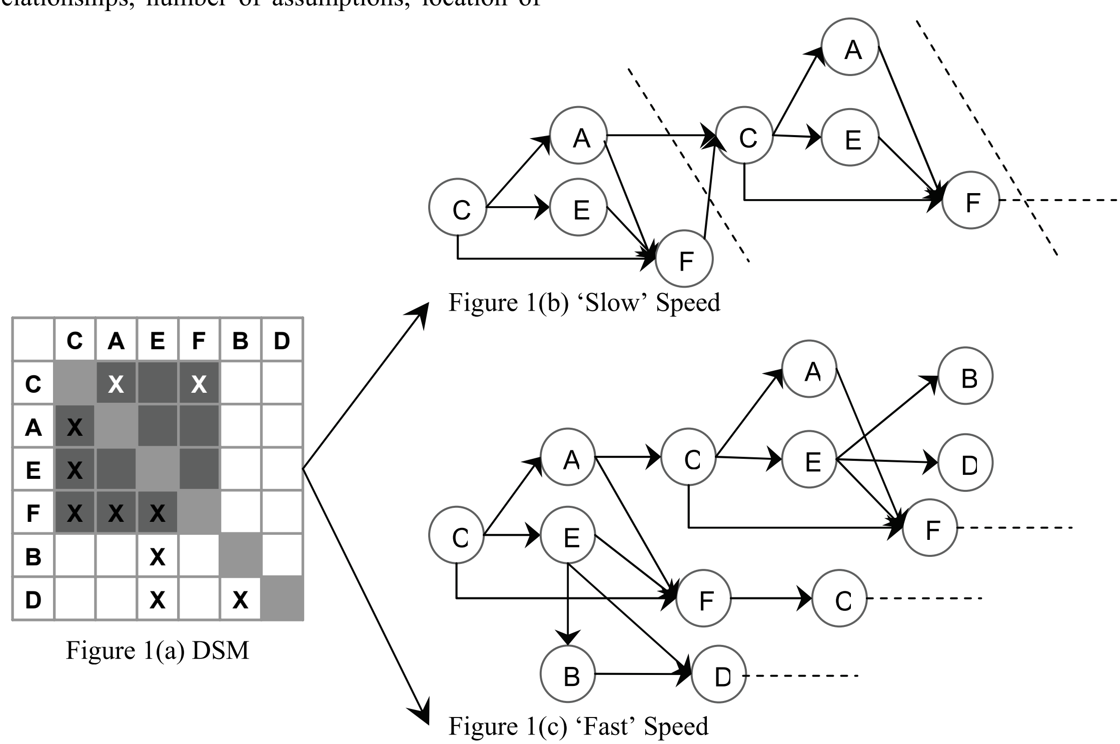

Figure 1 Iteration Strategy on 'Speed'

of repetition of the successor activity. In figure 2, consider the value of 0.1. This implies that if information from activity 'A' changes, then there is a 10% chance that 'B' will repeat.

	A	B	C
A	Pert[10,16,20]	0.5	0.5
B	0.1	Pert[17,20,25]	
C		0.9	Pert[26,29,32]

Figure 2 Partitioned DSM Sequence (Case-1)

4.1.2. Probabilistic duration values

The proposed model allows three point estimates for activity duration (PERT type) and is included along the diagonal cells of DSM. In figure 2, the value (Pert[10,16,20]) along the diagonal cell for A represents that the duration for A follows a three-point estimate of 10, 16, and 20 for BCV (best case value), MLV (most likely value) and WCV (worst case value) respectively.

4.2 Monte Carlo Simulation

Monte-Carlo simulation is attempted for estimating the project duration. Stroboscope, an acronym for STate and ResOurce Based Simulation of COnstruction ProcEsses has been selected to perform the Monte-Carlo simulation [18]. Among the several simulation languages, Stroboscope (ver 1,7,0) was chosen because of its ability to model complex operations and widespread usage.

For the stroboscope representation, information is assumed to be the resource flowing between the activities. Assumptions/uncertain information made for any activity is modelled as initial resource for the queues. The same stroboscope network was used to model 'fast' and 'slow' strategy based on flagging the input. Extra queues along with several 'controls' are introduced to control the flow of information during the 'slow' strategy.

The assumption made in this model is that the attributes (i.e. rework probability values and duration) do not vary with time. Further, rework duration is assumed to be the same as the normal duration for any activity. The above assumptions are made based on the following fact – Assumptions made for any activity can prompt the dependent activities to repeat or not to repeat. Further, based on the assumptions the activities may consume more duration or less duration than the normal duration. The PERT type duration distribution is assumed to take care of this uncertainty.

Once the stroboscope model is constructed, it is verified to determine whether the simulation model performs as intended. For verification, many trial (trace) runs were conducted.

4.3 Results

Table 1 presents the comparison of the results of the Stroboscope model for both the 'speed' (fast and slow).

Table 1 Comparison of the Results of Stroboscope Model for Fast and Slow (Case-1)

Strategy			Ave	Min	Max
Duration (days)		Fast	82.12	56.94	255.86
		Slow	89.20	58.09	345.16
No. of Repetitions for	A	Fast	2.20	0	9
		Slow	2.21	0	10
	B	Fast	0.22	0	5
		Slow	0.22	0	5
	C	Fast	0.20	0	4
		Slow	0.20	0	5

Observation of the results presented in Table 1 implies that 'fast' speed takes 82.12 days and the 'slow' speed takes 89.20 days. Further, the number of repetitions for the activities A, B, and C were almost the same for both the speed. Hence, for this particular case, the planner can select the 'fast' strategy for execution.

5. ANALYSIS & DISCUSSIONS

The above situation may not happen always. For the same example, consider an alternate sequence – CBA as shown in Figure 3. A comparison of the results for 'fast' and 'slow' for this DSM (Case-2) is displayed in Table 2.

	C	B	A
C	Pert[26,29,32]	0.9	
B		Pert[17,20,25]	0.1
A	0.5	0.5	Pert[10,16,20]

Figure 3 Partitioned DSM Sequence (Case-2)

Table 2 Comparison of the Results of Stroboscope Model for Fast and Slow (Case-2)

Strategy			Ave	Min	Max
Duration (days)		Fast	91.87	29.42	504.26
		Slow	61.24	30.14	156.28
No. of Repetitions for	A	Fast	1.09	0	18
		Slow	0.16	0	2
	B	Fast	1.88	0	18
		Slow	1.04	0	3
	C	Fast	0.29	0	5
		Slow	0.20	0	3

In this particular case, 'fast' speed consumes 91.87 days and the 'slow' speed was 61.24 days. The difference in the project duration for the 'speed' strategy is large in this case than the former one. Comparing the duration for both the cases shows that it is difficult to decide on the 'speed' strategy based on the basic DSM. Thus, there is a need for the Stroboscope model to decide on the 'speed'.

Further in table 2, all the activities for the 'fast' speed, repeated more number of times than for 'slow'. More number of activity repetitions implies requirement of abundant resources and project cost. Hence, for this particular sequence, the planners would select the 'slow' speed for execution.

From the above illustration it is clear that the choice of the strategy is directly dependent on (a) project duration, and (b) number of repetition of the individual activities. Analyzing the behaviour of the Stroboscope model, there are many attributes which influence the project duration These attributes include the basic sequence, the activity duration, rework probability values, etc. Research work in investigating the critical factors which govern the execution sequence and the resulting project duration is a potential area of future work.

Also, in the above two illustration, the authors have decided on the 'speed' strategy based on the output values (project duration and the number of repetition of activities). But the variation of the output values was very small in both the cases. This is because of the simplicity in the illustration.

It was stated earlier that it is difficult to decide on the execution sequence considering the basic sequence alone. So, stroboscope model has to be run each time for deciding the strategy. This process is tedious and not practical. Research investigation to determine the pattern recognition (may be by using Artificial Neural Network) is a critical area of future work.

Moreover, analysis of the schedule scenarios is theoretical and assumes no practical constraints. In reality, additional inputs such as resource availability, cost aspects, etc. is required for a practical analysis of the schedule and is an area of future work.

6. SUMMARY & CONCLUSIONS

The present study focuses on modelling iterations using DSM and simulation. Here, DSM determines the basic sequence and Simulation determines the execution sequence. The selection of 'speed' strategy is decided based on the project duration and the number of repetition of the individual activities. It was concluded that Stroboscope can be used for deciding the 'speed' strategy.

Illustration from the above example reveals that the proposed concepts can be applied to any design domain because of its simplicity. Even though the current example considers only one block, the same concept can be extended to multiple blocks. Further, the present approach is very basic and there is a need for further detailed research work to be done in this area as discussed earlier.

7. ACKNOWLEDGMENTS

The first author is grateful for the support received from the DST (Department of Science and Technology), New Delhi for the project under 'Fast Track Scheme for Young Scientists' to execute future work in this area.

8. REFERENCES

[1] Austin, S.A., A.N. Baldwin, B. Li & P. Waskett (1999) Analytical Design Planning Technique for Programming Building Design, Structures and Building, Vol. 134, No. 2, 111-118.

[2] Eppinger, S.D. (2001) Innovation at the Speed of Information, Harvard Business Review, Vol. 79, No. 1, 149-158.

[3] Yassine, A.A. (2004) An Introduction to Modeling and Analyzing Complex Product Development Processes Using the Design Structure Matrix (DSM) Method, Quaderni di Management (Italian Management Review), No. 9.

[4] Unger, D.W. & Eppinger, S.D. (2002) Planning Design Iterations, SMA Working Paper.

[5] Oloufa, A.A., Y.A. Hosni, M. Fayez & P. Axelsson (2004) Using DSM for Modeling Information Flow in Construction Design Projects, Civil Engineering and Environmental Systems, 21 (2), 105-125.

[6] Austin, S.A., A.N. Baldwin, B. Li & P. Waskett (2000) Analytical Design Planning Technique (ADePT): A Dependency Structure Matrix Tool to Schedule the Building Design Process, Construction Management and Economics, 18, 173-182.

[7] Yassine, A., D. Falkenburg, & K. Chelst (1999) Engineering Design Management: An Information Structure Approach, International Journal of Production Research, Vol. 37, No. 13, 2957 – 2975.

[8] Steward, D.V. (1981) The Design Structure System: A Method for Managing the Design of Complex Systems, IEEE Transactions on Engineering Management, EM-28, 71-74.

[9] Maheswari, J.U. (2006) Modeling Activity Sequencing for Construction Projects using Dependency Structure Matrix, Ph.D. Dissertation, IIT Madras, Tamil Nadu, India.

[10] Austin, S.A., A.N. Baldwin, B. Li & P. Waskett (1999) Analytical Design Planning Technique: A Model of the Detailed Building Design Process, Design Studies, Vol. 20, 279-295.

[11] Huovila, P., L. Koskela, M. Lautanala, K. Pietilainen, & V. Tanhuanpaa (1995) Use of the Design Structure Matrix in Construction, Proceedings of the 3rd International Workshop on Lean Construction, Albuquerque, Oct 16-18.

[12] Fayez, M., P. Axelsson, A. A. Oloufa & Y. Hosni (2003) DSM Versus CPM: Issues for Planning Design & Construction Activities, ASCE Construction Congress, Honolulu, HI, Mar. 10-14.

[13] Wang, W., T. Liau, & J. Liu (2004) Applying Simulation Technique to Model Design Iterations, Proceedings of 21st International Symposium on Automation and Robotics in Construction - ISARC 2004, Sep. 21-25, Jeju, Korea, 413-418.

[14] Smith, R.P., & S.D. Eppinger (1997) Identifying Controlling Features of Engineering Design Iteration, Management Science, 43 (3), 276-293.

[15] Smith, R.P., & S.D. Eppinger (1997) A Predictive Sequential Engineering Design Iteration, Management Science, 43 (3), 276-293.

[16] Browning, T.R. & S.D. Eppinger (2002) Modeling Impacts of Process Architecture on Cost and Schedule Risk in Product Development, IEEE Transactions on Engineering Management, 49 (4), 428-442.

[17] Maheswari, J.U. & K. Varghese (2006) Estimating Iterative Design Duration using Design Structure Matrix (DSM) & Simulation, Proceedings of INCITE/ITCSED 2006, Vol.1, 113-123.

[18] Martinez, J.C. (1996) STROBOSCOPE: State and Resource Based Simulation of Construction Processes, Ph.D. Dissertation, University of Michigan, Ann Arbor, Michigan.