Research on Topological Sorting Algorithm Suitable for Project Scheduling

Huilin Yuan[1], Wei Hong[2,*], Jinbo Chao[3]

1 Northeastern University at Qinhuangdao
Qinhuangdao 066004, China
2 China petroleum & chemical corp., Hebei oil products co.
Shijiazhuang, China
3 Yanshan University
Qinhuangdao 066004, China

***Abstract* — Project schedule is very important in a whole project, it is the basement of other parts of a project. Because it involves lots of activities, it is always complex and error prone. So checking project schedule becomes absolutely necessary before put into effect, one of which is to detect rings between activities in a project schedule. The paper proposes a new topology algorithm to detect rings, introduces the concept of design structure matrix, it is suitable for project schedule. Compared to other topology methods, it is more effectively.**

Keywords - component; project schedule; topology; design structure matrix

I. Introduction

Project schedule is central to the construction organization design, under the terms of the contract, on the basis of the provisions of the construction program to make specific schedule for start and end times of each part of the project. Generally available Gantt schedule or network schedule indicates it (as showed in Figure 1).

Project schedule is to ensure delivery of a construction according to the term of the contract. Other project work must be around and meet the requirements of the project schedule arrangements.

Preparation of construction schedule is based on the principles of the assembly-line method of network planning. Assembly line is a scientific organization of production methods, based on the division of labor and mass production. It features embodied in continuity, rhythm and balance on the production. Due to technical and economic characteristics of project products and the production, when using assembly-line methods in building construction, the project should be divided into several segments (activities), when the first group of professional construction team completed the first step of the first segment of construction and move away the working surface and transferred to the second construction section, the second professional construction team enter the working surface of the first segment to complete further process, and then transfer to the second construction segment continuous operation. This will ensure the work continuity of every construction team group, and so the further process can be done ahead of time, ensure full use of the space and time, shorten the project period, so that the project can be rapidly and stably.

The network schedule can be used as project schedule planning method throughout the construction process, reflect technology and organic links between the activities (or process engineering) and can provide a variety of useful manage information for managers. And the project process will be linked together as an organ.

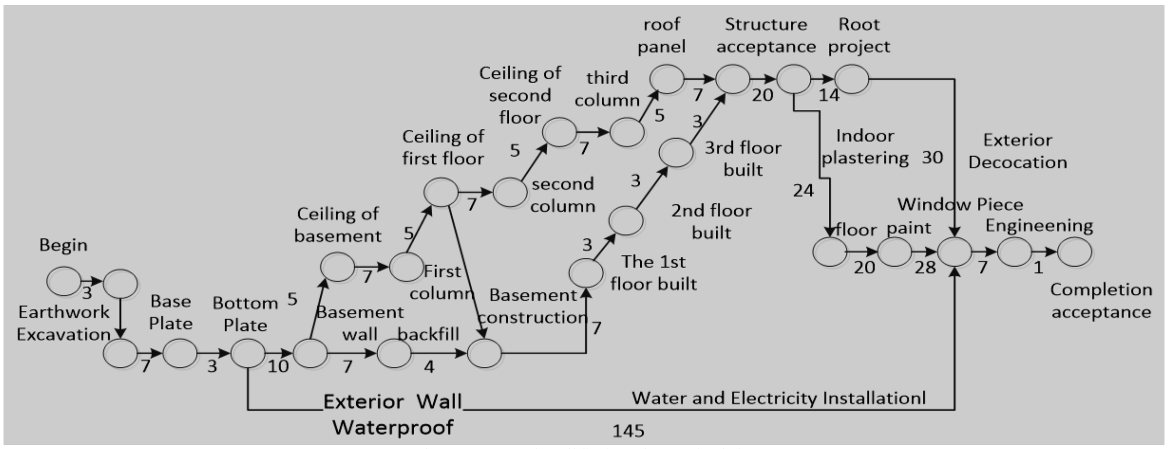


Figure 1. A simplified project schedule.

As can be seen above, the project schedule is complex, close convergence between activities, the successful completion of an activity is the basis of another activity began. If the activity present in a ring, it means that the end of an activity is entrusted to the previous activity, it is possible that in these activities the project cycle many times, delay the construction period.

Construction delay means that project can not be completed on schedule according to the contract, which is caused by non-contractor, and the construction unit or supervision units agrees to postpone the completion date, and extend the duration of the project. [1]

Common way to check rings is topological sorting. Traditional topological sorting algorithm first construct the network according to Gantt chart, determine the number of vertices(n), and then select a vertex that has no direct predecessor vertex, delete the vertex and edge issued by it in the network, and then select a vertex has no direct predecessor vertex again, n times the process is repeated until all vertices are treated ...

For simplified project as shown in Figure 1, the process will be repeated 25 times inefficiently. Because of involving a number of activities, Practical project is more time-consuming. [2-3]Common method is not fit for project process, in the paper a new topological method based on DSM is proposed as follows:

II. Topological Sorting Method Based on DSM

A. DSM

There has already been an excellent review by Browning in general areas of DSM application. They categorized DSM models into four types according to their characteristics and applications:

1) Component-Based DSM: which documents interactions among product components, can be used to facilitate appropriate modulation of a complex system;

2) Team-Based DSM, useful for depicting the interactions among design teams;

3) Activity-Based DSM, effective in modeling the information dependencies among design activities;

4) Parameter-Based DSM, for documenting physical design parameter relationships.

Iteration, the repetition of design tasks, is a fundamental characteristic of design process. Iteration occurred in product development for two main reasons. First, tasks may need to be reworked when review or testing activities detect some faults in the original design. Second, when modified information is passed along from upstream, some of the downstream work may be obsolete and need be reworked.

The iterative nature of product development can be addressed by using DSM. As shown in Figure 2, the basic DSM is a binary matrix representation of a project with elements denoting individual design tasks and off diagonal marks representing the information dependencies among these tasks. Along each row, the off diagonal marks indicate all of the tasks whose output information is required to perform the task corresponding to that row. Reading down each column reveals that which other tasks receive its output. When activities are executed in the order listed from top to bottom, sub diagonal marks represent an input from upstream activities/stages to downstream activities/stages. Super diagonal marks denote a feedback from downstream activities to upstream activities. As such, the DSM approach provides a concise way in describing and investigating information transfer and iteration.

The principal concepts of DSM can be summarized as follow:

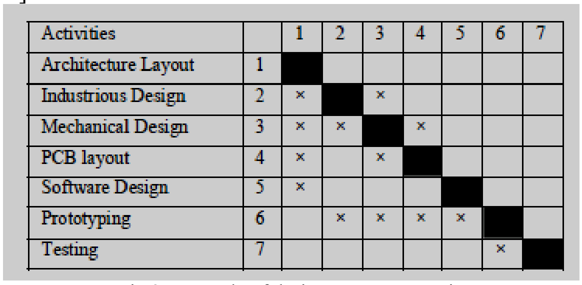
1) DPs of a system are intuitively elicited, often based on experience

2) Dependencies between the DPs are represented in matrix form (Fig. 2)

3) An 'X' in the matrix indicates a dependency between two DPs – the row-DP depends on the column-DP

4) DP2 is concluded to be dependent on DP1, when DP1 is an input to DP2 – modifying DP1 affects DP2

Modular design (modularity) is achieved by clustering inter-dependent DPs, and by information hiding, to form modules.Modules are units in a larger system that are structurally independent of one another, but work together [5-6] .



Activities		1	2	3	4	5	6	7
Architecture Layout	1	■						
Industrious Design	2	×	■	×				
Mechanical Design	3	×	×	■	×			
PCB layout	4	×		×	■			
Software Design	5	×				■		
Prototyping	6		×	×	×	×	■	
Testing	7						×	■

Fig.2. Example of design structure matrix

B. DSM topological sort

1) DSM sorting

DSM sorting try to make information flow fluently and ensure every task acquiring requirement before it begin.

In many sorting methods, Lawler develop a simple and effective topological sort, which use matrix and fit for processes without loops. Because we care for whether the predecessor tasks completed each ahead of the task, reverse sorting become more meaningful, so as to find the predecessor tasks for each task. Lawler's algorithm are applied on the columns of the matrix, through the discovery of task i (Ii) dependence level (which is equal to the sum of the task list, and then the task is classified). If there is whole zero column, it is DSM's final mission, a line of the same tasks with this column and other dependencies are deleted, this procedure is repeated until the discovery of another whole zero column. If no such columns, and DSM is not null matrix, the design process must contain the circulation of information, process is terminated [7-8]

The procedure of the lawler's:

Step 1. Begin: sum each column of the task.

Let: $I_i = \sum_{j=1}^{n} a_{ij}, \quad i = 1, 2, ... n$

$N=\{1,2,...,n\}$, $m=1$

Step 2. look for the whole zero column.

Seeking $k \in N$ to meet $I_k=0$, if there isn't such k, then there must be loops.

Let: Rank（k）$=m$

$m=m+1$

$N=N-k$

If N=$\varnothing$, then it finish.

Step 3. Calculate the remaining tasks

Let: $I_i = I_i - a_{ik}$ $i \in N$

Return to Step 2.

C. *topological sorting method based on DSM*

Most cases in topological sort (such as AOV network) is not allowed to ring there. so have to consider the ring, and modify the above algorithm to based on DSM topological sort method:

1) According to flow chart to establish DSM matrix *A*,

$$A = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 7 \\ 0 & 2 & 0 & 0 & 5 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

2) Adjust the order of activities. In *A*, not "0" mark in the line display the flow of information required to carry out the activity, and in the column indicates that the output flow of the activity. If all the elements of a row on *A* are zero, the activity responding to this row should be executed as early as possible, because it is the source point of the whole activity process and does not need any information of other activities. We adjust it on the first row and column on *A*, and mark the activities (which need information flow of this point) as candidate sources.

If all the elements of a column are zero, the activity responding to this column should be executed behind other activities, because it is the terminal point and provides no information to the others. We put it on the last column and row on *A* and mark the activities (which flow information in this point) as candidate meeting point.

In the matrix, all the elements of the first row are zero, it is the source point of the whole activity process and adjust on the first row and column, and mark Activity '2' as candidate source; all the elements of the sixth column are zero, it is the terminal point and put on the last column and row, mark Activity '4' as candidate meeting point. And then the matrix is:

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 7 \\ 2 & 0 & 0 & 5 & 0 \\ 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 \end{pmatrix}$$

3) In a similar way, we can do the process in the candidate points and adjust other activities on ***A*** in turn.

Now there are no optional lines and columns, so there are loops. Then have to select '2' in the candidate sources and and mark Activity '3' and '7' as candidate ends:

$$\begin{pmatrix} 0 & 0 & 5 & 0 \\ 3 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

All elements in the line of Activity '7' are zero, that mains '7' is in the same ring with '2', and the matrix is:

$$\begin{pmatrix} 0 & 0 & 5 \\ 3 & 0 & 0 \\ 0 & 4 & 0 \end{pmatrix}$$

4) Let the number of activities is n. If the activity does not exist with the ring, then the process goes through (n +1) / 2 times, all sort of activities will be sort sequence. If the sort sequence is gained because get rid of a activity in the candidate source, the sequence is in the same ring with the candidate source; If there are still rings, select a Meeting Point in the candidate terminates as secondary Meeting Point, the algorithm returns to step 2.

5) Otherwise, when there are rings, the source point from the candidate is selected as a secondary source, the algorithm returns to step 2.

$$A^1=\begin{pmatrix}
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&2&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&3&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&4&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&5&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&6&0&8&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&5&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&7&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&9&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&10&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&11&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&12&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&13&0&0&0&17&0&0&0&0&0&0&0\\
0&0&0&0&0&0&7&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&15&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&16&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&14&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&18&0&0&0&0&0&0\\
0&0&0&4&0&0&0&0&0&0&0&0&0&0&0&0&0&0&19&0&0&22&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&18&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&21&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&20&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&23&0
\end{pmatrix}$$

III. AN EXAMPLE

For example, a simplified process is expressed in Figure 1- A simplified project schedule.

According to the method, matrix A^1 is established:

In the matrix, all the elements of the activity ‘0’ are zero, it is the source point of the whole activity process and adjust on the first row and column, and mark Activity ‘1’ as candidate source; all the elements of the last column are zero, it is the terminal point and put on the last column and row, mark Activity ‘23’ as candidate meeting point. And ‘1’ 、‘23’ is selected，then ‘2’ and ‘20’, after 3 times the matrix is showed as A^2. Now Activity ‘3’ as candidate source, Activity ‘22’, ‘19’ and ‘4’ as candidate end point.

And the next source ‘3’ and end points ‘22’, ‘19’ are deleted. And the process is repeated.

Now Activity ‘4’ as candidate source, Activity ‘18’ and ‘21’ as candidate end point.

Then ‘5’, ‘7’ and ‘18’...

‘6’, ‘8’ and ‘14’...

‘9’ and ‘16’...

‘10’ and ‘15’...

‘12’ and ‘17’ showed as A^8...

Through 10 times, the topology end.

$$A^2 = \begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 6 & 0 & 8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 9 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 11 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 12 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 13 & 0 & 0 & 0 & 17 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 15 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 16 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 14 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 18 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 18 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 21 & 0
\end{pmatrix}$$

$$A^8 = \begin{pmatrix}
0 & 0 & 0 \\
12 & 0 & 0 \\
0 & 13 & 17
\end{pmatrix}$$

Compare to the traditional method's 25 times, the new topology is more effective. For the general case, by using DSM topological sort, n-first search can find source and sink points of the entire sequence, respectively, on the matrix of its first row, first column, Row *n-1* and Column *n-1*; then search sub-level source and sink points for (n-2) times in the remainder (*n-2*)*(*n-2*) matrix; so, if there is no ring, when *n* is even, the algorithm is executed about *n/2* times.

IV. Conclusion

Because of involving a number of activities, Practical project is more time-consuming. traditional method is not fit for project process. The new Topological sort approach based on DSM is to identify the starting point of a sequence of activities and sink points, it look for its source and sink points from two directions at the same time, and its adjacency matrix method achieve a simple design ideas; and the time complexity of it is much lower. Because sweeping from its two (source and sink points) direction at the same time, so they use less time than the traditional topological sorting, and is suitable for project process.

ACKNOWLEDGMENT

The authors are grateful to the valuable and constructive comments from the editor and anonymous referees. This work was supported by a grant from the National Natural Science Foundation of China (No. 121212405), the National Natural Science Fund #60821063 and Science and Technology Plan Project of Hebei Province #13214716.

REFERENCES

[1] http://baike.so.com/doc/1522798-1609908.html.

[2] http://baike.baidu.com/link?url=dMqWIKxoYn9KOhPJyPERxG1tW wm5ivuVR5gGOwF9k G3fRnG4OY-SDQp0g__rfaUhgRUi-8MrycAE7VytiLmCsq

[3] Renkun Yin, "Data Structure", Tsinghua university press: Beijing, 2012.

[4] Steward D V, He Dengfa, Sun Yanpeng, et al., "The design structure system: A method for managing the design of complex systems edited by IEEE Trans.", Engineering Management, vol. 78, No. 03, pp. 71-74, 1981.

[5] Yuan Huilin, Sun Fuquan, Wang Dingwei, "Process optimization algorithm based on the double value design structure matrix", Systems engineering theory & practice, vol. 34, No. 4, pp. 854-860, 2014.

[6] Steven D. Eppinger, Tyson R. Browning, "Design structure matrix methods and applications," The MIT press: Massachusetts London, 2012.

[7] Lee Yaching, Chu Pinyu, Tseng Hsienlee, "Corporate performance of ICT-enabled business process re-engineering", Industrial Management Data Systems, vol. 111, No. 05, pp. 735-754, 2011.

[8] Tyson R. Browning, "Design structure matrix extensions and innovations: a survey and new opportunities", IEEE transaction on engineering management, vol. 63, No. 01, pp. 27-52, 2016.