The SSD Graph: A Tool for Project Scheduling and Visualization

HYUNG LEE-KWANG, MEMBER, IEEE, AND JOEL FAVREL

Abstract—A graphic tool, the SSD graph, is proposed for three important phases of project management: planning and scheduling, control, and evaluation. The SSD graph represents the structure, states (scheduled and actual states), and deviation of a system. A project management system using the SSD graph is developed, and it visualizes the actual state of a project and keeps the historical record of the project performance. This system allows a project manager to see overall status and to review the project performance. Several applications have shown its usefulness for scheduling and control of project systems.

I. INTRODUCTION

THERE ARE TWO classes of graphic models which are used for project and production management. The first class of graphs including Gantt chart, PERT, GAN [8], GERT [10], LOGOS [9], and GRAI [7] visualize the structure of a system. The second class including Petri net [1], [4], [13], [14] GRAFCET [5], [6], and UCLA graph [11] visualize the structure and states of a system. The aim of this paper is to propose a graphic model which can visualize the structure, states and deviation (SSD) of the project system and evaluate its performance.

The Gantt chart represents the execution time of an activity by length of a bar. The conception is utilized in some graphic models [2], [3]. Petri nets represent an actual state of a system by tokens. We synthesize and extend these conceptions to construct the SSD graph which can be used for scheduling, control, and evaluation of project systems.

II. DEFINITION

A. Structure

The structure of a project system represents the scheduling of the system. The structure of SSD graph shows the scheduling of a project, and is defined by \( (A, R, T, S) \), where \( A = \{a_1, a_2, \cdots, a_n\} \) is a set of activities, \( R = \{(a_i, a_j), \cdots\} \) is a set of precedence relationships between activities, \( T: A \rightarrow \mathbb{R}^+ \) is a mapping of \( A \) into the set of positive real numbers. \( T \) is given by a vector \((t_1, \cdots, t_l, \cdots, t_n)\) where \( t_i \) indicates the execution time of activity \( a_i \). \( S: A \rightarrow \mathbb{R}^+ \cup \{0\} \) is a mapping of \( A \) into the set of nonnegative real numbers. \( S \) is given by a vector \((s_1, \cdots, s_l, \cdots, s_n)\) where \( s_i \) indicates the float of activity \( a_i \).

On a graph, an activity is represented by a bar, the relationship between activities by a directed arc, the execution time by length of a bar, and the float of an activity by the discontinuous line. We omit arcs between bars which form a continuous sequence. An SSD graph is visualized on the time scale.

To illustrate, consider the SSD graph in Fig. 1. This graph represents the structure of SSD graph and visualizes the scheduling of a system as follows:

\[
A = \{a_1, a_2, a_3, a_4\}
\]

\[
R = \{(a_1, a_2), (a_1, a_3), (a_2, a_4), (a_3, a_4)\}
\]

\[
T = (t_1, t_2, t_3, t_4) = (2, 3, 1, 2)
\]

\[
S = (s_1, s_2, s_3, s_4) = (0, 0, 2, 0).
\]

B. States

To represent the dynamic states of a system, we add two types of markings to the structure of SSD graph: scheduled (expected) markings and actual markings.

1. Scheduled Marking: A scheduled marking \( m_s \) is defined to specify the scheduled state on time \( t \). The marking \( m_s \) is given by a column vector \((m_{s_1}, \cdots, m_{s_n})\) where \( n \) is the number of activities, and is a mapping of the activities \( A \) into the set \( \{-1, 0, 1\} \). The notation \( m_{s_i} \) indicates the state of an activity \( a_i \) on time \( t \) such as:

\[
m_{s_i} = -1, \quad \text{if } a_i \text{ is not begun}
\]

\[
m_{s_i} = 0, \quad \text{if } a_i \text{ is in process}
\]

\[
m_{s_i} = 1, \quad \text{if } a_i \text{ is finished}
\]

2. Actual Marking: An actual marking \( m_a \) is a matrix of \( (m_{a_{ij}}) \) for \( 1 \leq i \leq n \) and \( 1 \leq t \leq h \), which shows the scheduled status of a system up to time \( h \). On a graph, the scheduled marking is represented by a token on the time scale.

We call the vertical line containing the time token as the time vertical line. Activities on the time vertical line are

Fig. 1. Structure of the SSD graph.
graph with markings is then defined by
the first day
This historical record of system evolution is given by the
Therefore, an SSD graph with actual marking can also
scheduled marking is also used in the actual marking.
represent the scheduled marking. For example, we can read
both of the scheduled and actual states in Fig. 3. An SSD
deviation in the following section.
The actual marking is useful to calculate the performance
finished on the fourth day
fourth day and in process on the fourth day
activity, we deposit a token on the start line of the bar as
the tokens from its preceeding activities are arrived, and it
represents the status of an activity,
SSD graph in Fig.
2.
Actual Marking: An actual marking \( m_i \) is defined to
specify the actual states on time \( t \). The marking \( m_i \) is a
mapping of \( A \) into the set \( \{-1, 0, 1\} \), and given by a column
vector \( (m_{1i}, \ldots, m_{ni}, \ldots, m_{ni}) \). The notation \( m_{ih} \) indicates
the state of activity \( a_i \) at time \( t \) as in the scheduled marking. The
actual marking matrix \( M_t = (m_{ih}) \) is given by \( (m_{1h}, \ldots, m_{nh}) \)
which shows the actual evolution of a system up to time \( h \).
On a graph, the actual marking is represented by distribution
of tokens as shown in Fig. 3. An activity begins when all
the tokens from its preceeding activities are arrived, and it
gives one token to each following activity when it is finished.
If an activity is not yet begun after termination of its preceding
activity, we deposit a token on the start line of the bar as
shown in Fig. 3. A token represents the status of an activity,
and the distribution of tokens on an SSD graph shows the
overall project status. In Fig. 3, \( a_1 \) is in process, \( a_2 \) is finished,
and \( a_2 \) is not begun. That is, the SSD graph shows the project
status on the fifth day.
To keep the historical record of the project performance, we
define the actual marking matrix as in the scheduled marking.
The actual marking is useful to calculate the performance
deprivation in the following section.
If we assume that \( a_1 \) is begun on the first day and finished on
the first day \( (m_{11} = 0); \) \( a_2 \) is begun on the third day and
finished on the fourth day \( (m_{23} = m_{24} = 0); \) \( a_1 \) is begun on the
fourth day and in process on the fourth day \( (m_{44} = m_{34} = 0). \)
This historical record of system evolution is given by the
actual matrix in Table II.
We note that, on a graph, the token on the time scale for the
scheduled marking is also used in the actual marking.
Therefore, an SSD graph with actual marking can also
represent the scheduled marking. For example, we can read
both of the scheduled and actual states in Fig. 3. An SSD
diagram with markings is then defined by \((A, R, T, S, M)\). The

\[
\begin{array}{c|ccccc}
\text{day} & 1 & 2 & 3 & 4 & 5 \\
\hline
\text{a}_1 & 0 & 0 & 1 & 1 & 1 \\
\text{a}_2 & -1 & -1 & 0 & 0 & 0 \\
\text{a}_3 & -1 & -1 & -1 & 0 & 1 \\
\text{a}_4 & -1 & -1 & -1 & -1 & -1 \\
\end{array}
\]

C. Deviation
A deviation \( d_{ij} \) is defined to indicate difference between the
actual and scheduled states on time \( t \). The deviation \( d_{ij} \) is a
mapping of \( A \) into \( \{-2, -1, 0, 1, 2\} \). \( d_{ij} \) is represented by
\( (d_{1i}, \ldots, d_{ni}, \ldots, d_{ni}) \) and the value of \( d_{ij} \) is given by \( d_{ij} = m_{ij} - m_{ih} \). The notation \( d_{ij} \) indicates the progress deviation of
activity \( a_i \) on time \( t \) such as:
\[
d_{ui} = -2, \text{ if } a_i \text{ is delayed by the second order (a_i \text{ is not yet}
begun but it was scheduled to be finished)},
= -1, \text{ if } a_i \text{ is delayed by the first order (a_i \text{ is not begun}
but it was scheduled to be in process, or it is in process
but was scheduled to be finished}),
= 0, \text{ if } a_i \text{ is not delayed or in advance},
= 1, \text{ if } a_i \text{ is in advance by the first order (a_i \text{ is finished}
but it was scheduled to be in process, or is in process
but was scheduled not to be begun) },
= 2, \text{ if } a_i \text{ is in advance by the second order (a_i \text{ is}
finished but scheduled not to be begun) }.
\]
A deviation matrix \( D_k \) is given by \((d_{1i}, \ldots, d_{ij}, \ldots, d_{ni})\) for
$1 \leq t \leq h$ which shows historical deviations of the system evolution up to time $h$. The elements of the deviation matrix $d_{it}$ correspond to the $ith$ element in the deviation $d_{it}$. In other words, the element $d_{it}$ shows the deviation of the activity $i$ on time $t$. The matrix is obtained by $D_t = M_t - M_0$. This matrix is useful to represent the historical deviations by digits $\{-2, -1, 0, 1, 2\}$.

On a graph, the deviation is represented by an actual marking as shown in Fig. 3. That is, the actual markings on the left of the time vertical line represent delayed activities; the markings on the right represent activities in advance; the marking on the time vertical line represent activities which are not delayed or in advance. Therefore, the SSD graph shows overall project status and deviation.

We see in Fig. 3 that the progress of $a_2$ is in advance by the first order while $a_1$ is delayed by the first order. That is, $d_{1.5} = m_1 - m_{1.5} = (1, 1, 0, -1) - (0, 1, -1, 0) = (0, 1, -1, 0)$.

Table III shows the historical deviation of the system evolution up to the fifth day, which is obtained by $D_5 = M_5 - M_1$ in Tables I and II.

III. CONTRIBUTIONS

A. Properties of the SSD Graph

In this section, we discuss some properties of the SSD graph for project management in comparing with the other graphic tools such as the Gantt chart, PERT/CPM, and Petri net (cf., Table IV).

1. Visual Representation of the Passage of Time and Execution Time: This property is borrowed from the Gantt chart. In the PERT network, the time is represented by annotating figures.

2. Visual Representation of Actual State: This property is from Petri nets. In the PERT, we cannot represent the actual state of project.

3. Visual Representation of Scheduled States and Deviations: This property is a proper contribution of the SSD graph for system control. In the PERT, the scheduled states and deviations of a project system cannot be shown.

4. Evaluation of a System: It is possible with the marking and deviation matrices which show the historical evolution of the system. It is not possible in the other tools.

5. Visual Representation of a Float: We have seen in Fig. 1 that the float of an activity is represented by the discontinuous line. When we construct the SSD graph, a critical path without discontinuous lines appear automatically.

6. Simple Structure: To compare the readability of the SSD graph with other graphs, consider the SSD graph in Fig. 1. The Gantt chart in Fig. 4 models the same system as in Fig. 1. Fig. 4(a) shows the earliest scheduling and Fig. 4(b) the latest scheduling. Two Gantt charts are necessary to represent the same structure given by the SSD graph. Fig. 5 shows PERT representing the same system as in Fig. 1. The PERT annotates figures to indicate execution time, the earliest date, the latest date and float activity. The SSD graph, however, does not annotate figures. We see that the SSD has a simple structure and visualizes more information than the other graphs. Therefore, the SSD graph is more readable than the others.

### TABLE III

<table>
<thead>
<tr>
<th>DeVIATION MATRIX $D_t$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_1$</td>
</tr>
<tr>
<td>$s_1$</td>
</tr>
<tr>
<td>$s_2$</td>
</tr>
<tr>
<td>$s_3$</td>
</tr>
<tr>
<td>$s_4$</td>
</tr>
</tbody>
</table>

B. Application

We have applied the SSD graph to a project management system. The system was implemented on an IBM-PC (MS-DOS) and was programmed in BASIC. This system is utilized for projecting scheduling and visualization in the material science laboratory in the Korea Atomic Energy Research Institute and in the Samjung Construction Company in Daejon, Korea.

The control flow of the SSD system is given in Fig. 6.

The SSD project management system consists of three parts; the project input, database, and visualization. In the project input part, we can call or delete a project memorized in the system and enter a new project. If we input the information about the scheduling of a project such as the number, name, execution time, and precedence relationship of activities, a scheduled marking matrix is created in the database. Then, the structure of SSD graph can be visualized to show the scheduling of the project. When the SSD graph is drawn on the CRT, critical paths without discontinuous lines appear automatically. If we want to update the scheduling, we can modify the scheduled marking matrix at any time.

The responsible person for each activity must input the performance rate of the activity every time unit (day or week). Then the actual marking matrix is updated every time unit. If the manager of project wants to see the overall project status on a given time, he gives the date. Then, the distribution of tokens in the SSD graph visualized on the CRT shows the overall status and performance deviations between scheduled states and actual states.

Table V summarizes the SSD project management system.

In the system, we have visualized the SSD graph vertically because it is easy to move the graph vertically. Fig. 7 shows an example of the SSD system.

The time scale and bars are in parallel with the vertical axis. Activities in process are represented by the % bars, activities finished by the # bars, and activities not begun by the * bars. A token is represented by the symbol "X" on the CRT. In the SSD systems in Fig. 6, there is a "X" symbol on the fifth week in the time scale. Therefore, the system shows system status on the fifth week.

In Fig. 6, the bar numbered 1 represents the activity "approve the project," the bar 2 represents the activity "specification of project," the bar 3 represents "preparation," bar 4 "consulting," bar 5 "outline," bar 6 "requirement verification," bar 7 "design," bar 8 "implementation," and bar 9 "test and documentation."
TABLE IV

COMPARISON OF THE SSD GRAPH WITH THE GANTT CHART, PERT/CPM, AND PETRI NET

<table>
<thead>
<tr>
<th>SSD graph</th>
<th>Gantt chart</th>
<th>PERT/CPM</th>
<th>Petri net</th>
</tr>
</thead>
<tbody>
<tr>
<td>Visual representation of time</td>
<td>possible by length of bar</td>
<td>possible by length of bar</td>
<td>no</td>
</tr>
<tr>
<td>Visual representation of actual state</td>
<td>possible by marking</td>
<td>possible by modification</td>
<td>no</td>
</tr>
<tr>
<td>Visual representation of scheduled state</td>
<td>possible by marking</td>
<td>possible by modification</td>
<td>no</td>
</tr>
<tr>
<td>Visual representation of performance deviation</td>
<td>possible by marking</td>
<td>possible by modification</td>
<td>no</td>
</tr>
<tr>
<td>Review of performance history</td>
<td>possible by marking and deviation matrices</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>Visual representation of float</td>
<td>possible by discontinuous lines</td>
<td>two Gantt charts</td>
<td>no</td>
</tr>
</tbody>
</table>

The activities 1, 2, 3, and 4 are finished on the fifth week. The activities 7, 8, and 9 are not begun. There are "o" symbols on the bar 5 and 6. Therefore, the activities 5 and 6 are in process on the fifth week. We see that system performance is not delayed and the performance of activities 5 and 6 are in a little advance.

We have another application of the SSD graph to a decision support system in a bearing production factory in France. The system was programmed in FORTRAN 77 on Honeywell MINI-6 with a color CRT Tektronix 4025 [12]. It visualizes the scheduling and dynamic states of the job shop in the factory. In [12], it is shown that the SSD is useful for visualization of scheduling and control of systems.

IV. CONCLUSION

We have proposed a new project management tool, the SSD graph, for planning and scheduling, control, and evaluation of projects. The SSD graph has a simple structure, and the representation of planning and scheduling is highly readable. The SSD graph shows scheduled and actual states for the control of a system. The marking and deviation matrices show the history of a system and allow us to evaluate the project. We have applied the SSD graph in several engineering management areas, and the applications have shown its usefulness for project management.

ACKNOWLEDGMENT

The authors wish to thank the editor for helpful comments.
Input or update scheduling
Input the state of each activity on the given time
Give the time to review the project performance

Table V
Description of the SSD Project Management System Implemented on IBM-PC (MS-DOS)

<table>
<thead>
<tr>
<th>Input</th>
<th>Database</th>
<th>Visualization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input or update scheduling</td>
<td>Create the scheduled marking matrix</td>
<td>Visualize the structure of SSD graph</td>
</tr>
<tr>
<td>Input the state of each activity on the given time</td>
<td>Update the actual marking matrix and deviation matrix</td>
<td>Visualize actual marking</td>
</tr>
<tr>
<td>Give the time to review the project performance</td>
<td>Search the status of project in the matrix</td>
<td>Visualize the status on the given time</td>
</tr>
</tbody>
</table>
Fig. 7. An example of the SSD project management system. Activities in process are represented by the % bars, activities finished by the # bars, and activities not begun by the * bars. The activity names are given.

REFERENCES