GRAPHICAL EVALUATION AND REVIEW TECHNIQUE (GERT)

P.R Sreemahadevan Pillai “A comprehensive study of project scheduling techniques and the introduction of new techniques” Thesis. Department of Civil Engineering, Regional Engineering College Calicut, University of Calicut, 2002
CHAPTER 9

GRAPHICAL EVALUATION AND REVIEW TECHNIQUE (GERT).
9.1 Introduction

GERT is mostly concerned with the generalizations of the basic CPM/PERT type of network to permit probabilistic activities and nodes. The components of GERT networks are directed branches (arcs, edges, transmittances) and logical nodes (vertices). Two parameters are associated with the branch:

1. The probability that a branch is taken, $p$, given that the node from which it emanated is realized.
2. A time, $t$, required, if the branch is taken, to accomplish the activity, which the branch represents.

The time, $t$, can be a random variable. If the branch is not part of the realization of the network then the time for the activity represented by the branch is zero.

A node in a GERT network consists of an input (emitting, distributive) function and an output (emitting, distributing) function. Three logical relations on the input side and two types of relations on the output side will be considered. This yields six types of nodes, which are described in Table 9.1.

First, consider the AND node as depicted by Node.3 in Fig.9.1. This is the standard node type in PERT networks. By definition Node.3 will be realized only if both $a$ and $b$ are realized. The probability and time associated with the realization of Node.3 are $p_a p_b$ and $\max(t_a, t_b)$ assuming activities $a$ and $b$ are independent.
**TABLE 9.1**

**Node Characteristics and Symbols**

<table>
<thead>
<tr>
<th>Output</th>
<th>Input</th>
<th>Exclusive-or</th>
<th>Inclusive-or</th>
<th>and</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deterministic</td>
<td>□</td>
<td>△</td>
<td>△</td>
<td>△</td>
</tr>
<tr>
<td>Probabilistic</td>
<td>△</td>
<td>△</td>
<td>△</td>
<td>△</td>
</tr>
</tbody>
</table>

**Exclusive-or:** The realization of any branch leading into the node causes the node to be realized; however, one and only one of the forward branches leading into this node can be realized at a given time.

**Inclusive-or:** the realization of any branch leading into the node causes the node to be realized. The time of the realization is the smallest of the completion times of the activities leading into the *inclusive-or* node.

**And:** The node will be realized only if all the branches leading into the node are realized. The time of the realization is the largest of the completion times of the activities leading into the *and* node.
**Deterministic:** All branches emanating from the node are taken if the node is realized, that is, all branches emanating from this node have a $p$-parameter equal to one.

![Diagram of a simple network with an AND node]

![Diagram of a simple network with an inclusive-or node]

**Probabilistic:** At most one branch emanating from the node is taken if the node is realized. For the inclusive-or relation, the analysis proceeds as in the AND case. The inclusive-or node is shown in Fig. 9.2.

The reduction process involves the enumeration of all mutually exclusive alternative methods of realizing node 3 from node S. These are listed in Table 9.2. For the Exclusive-or relation, the probability and time parameter can be combined into a single parameter as shown in Equation 9.1.
\[ w(s) = pM_f(s) \quad \text{(9.1)} \]

**TABLE 9.2**

Possible Events for inclusive-or Nodes

<table>
<thead>
<tr>
<th>Branch occurrence</th>
<th>Probability</th>
<th>Equivalent time</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a) but not (b)</td>
<td>(p_a - p_a \cap b)</td>
<td>(t_a)</td>
</tr>
<tr>
<td>(b) but not (a)</td>
<td>(p_b - p_a \cap b)</td>
<td>(t_b)</td>
</tr>
<tr>
<td>(a \text{ and } b)</td>
<td>(p_a \cap b)</td>
<td>(\min(t_a; t_b))</td>
</tr>
</tbody>
</table>

where \(M_f(s)\) is the moment generating function (or other exponential transform) of the time parameter. For network consisting of exclusive-or nodes, \(w(s)\) can be calculated using (9.2), the topology equation or Mason's rule of signal flow graph theory

\[
w_g(s) = \frac{\sum_i (\text{path } i) \left[ 1 + \sum_{m=1}^{\infty} (-1)^m (\text{loops of order } m \text{ not touching path } i) \right]}{\left[ 1 + \sum_{m=1}^{\infty} (-1)^m (\text{loops of order } m) \right]} \quad \text{(9.2)}
\]

Equation (9.2) defines the equivalent \(w\)-function, \(w_g(s)\) for any network consisting only of exclusive-or node:

In Equation (9.2), a loop is defined as a sequence of branches such that every node is common to two and only two branches of the loop, one terminating at the node and the other emanating from that node. In a first-order loop every node can be reached from
every other node. A loop of order \( n \) is set of \( n \) disjoint first-order loops. Disjoint loops are loops, which have no nodes in common. The parameter of the loop is the product of the parameters of the branches of the loop. A forward path is a sequence of branches from one node to the other such that every node except the two specified is common to two and only two branches of the forward path. The difference between a forward path and a loop is that the start node of a path has no input branch and the terminal node of a path has no output branch.

<table>
<thead>
<tr>
<th>Network type</th>
<th>Graphical representation</th>
<th>Paths</th>
<th>Loops</th>
<th>Equivalent function ( w_E )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Series</td>
<td><img src="image" alt="Series Network Diagram" /></td>
<td>( w ) ( w ) ( w )</td>
<td>--</td>
<td>( w ) ( w )</td>
</tr>
<tr>
<td>Parallel</td>
<td><img src="image" alt="Parallel Network Diagram" /></td>
<td>( w ) ( w )</td>
<td>--</td>
<td>( w ) + ( w )</td>
</tr>
<tr>
<td>Self-loop</td>
<td><img src="image" alt="Self-loop Network Diagram" /></td>
<td>( w ) ( w )</td>
<td>( w ) ( w )</td>
<td>( \frac{w}{1 - w_b} )</td>
</tr>
</tbody>
</table>

has no output branch.

**Fig. 9.3 Calculation of the equivalent \( w \) - function**

In Fig. 9.3, the equivalent \( w \) - function is calculated for three basic network types.

From the definition of \( w(s) \) we have

\[
p = w(0) \quad \text{(9.3)}
\]

and

\[
M(s) = \frac{w(s)}{p} \quad \text{(9.4)}
\]

since

\[
M(0) = 1.
\]
Thus, the probability of realizing a node or the moment generating function of the time to realize a node can be obtained from knowledge of \( w(s) \). An example of how GERT is used for analyzing the research and development expenditures is given in the following example (Alan B. Pritsker, 1968).

9.2 Analysis of research and development expenditures

An example of the analysis of research and development expenditure using the network is shown in Fig. 9.4.

Fig. 9.4. GERT network of a research and development process
The definitions of events and outcomes (nodes) and the activities (branches) are given in Table 9.3.

### TABLE 9.3

**Definitions for research and development Network**

**Events**

1. *Feasibility study indicates electrical control of high temperature is/is not feasible.*
2. *AC control found suitable /unsuitable.*
3. *DC control found suitable /unsuitable*
4. *Optimum integration of AC/DC circuits achieved.*
5. *Unit found to be within/outside potential market price.*
6. *Pneumatic control found to be feasible/ unfeasible.*
7. *Unit found to be within/ outside potential market price.*

**Outcomes**

i. *Project dropped.*
ii. *Project dropped.*
iii. *Project dropped.*
iv. *Project put into production and marketed.*
v. *Project dropped.*
vi. *Project dropped.*

vii. *Project put into production and marketed.*

**Activities**

A. *Pneumatic feasibility study.*
B. *AC control investigation.*
C. *DC control investigation.*
D. *Report writing.*
E. *Investigation of optimum AC/DC integration.*
F. *Report writing.*
G. *Investigation of optimum AC/DC integration.*
H. *Economic analysis of system.*
I. *Report writing.*
J. *Report writing.*
K. *Report writing.*
L. *Economic analysis of system.*
M. *Report writing.*
N. *Report writing.*
Fig 9.5 Decision box network

Fig 9.6 Detailed segment of the network
Fig 9.7. A two-port description of an activity

For each branch of the network, it gives the probability that the branch is realized given that the preceding node is realized, and the time and cost associated with the activity represented by the branch if the activity is performed. These values are inserted on the GERT network given in Fig.9.4 by an ordered triple of probability, time (weeks) and cost in Rs.1,000 units, namely, \((p,t,c)\). Time in this example is not a duration but the amount of effort required to perform the activities measured in weeks.

While constructing the GERT network first the AC and DC control investigations (Activities B and C) are performed simultaneously and this should be indicated on the network without the aid of the bracket. Second nodes I and II do not result in the project being dropped as implied in Fig.9.5. Also, the decision nodes
represent specific events, not either-or types of events. For ease of reference between Figs. 9.4 and 9.5, nodes have been labeled with two numbers (2 and 3) and the complements of these numbers (\(\overline{2}\) and \(\overline{3}\)). Thus, node \(23\) represents the event AC control has been found to be suitable and DC control has been found to be unsuitable. The detailed segments of the network between node 1 and combination of nodes 2, 2, 3 and 3 is given in Fig. 9.6.

Third, three terminal nodes, U, S and T, have been added in Fig 9.4. Node U represents the event “project dropped, S represents “project successful”, and node T represents the event “project terminated” whether it was successful or not.

The GERT analysis for the network presented in Fig.9.4 requires the extension of the \(w\)-function to handle two additive parameters. If \(t\) and \(c\) are independent or information is only desired about them separately, the \(w\)-function for a branch becomes

\[
w(s_1, s_2) = p e^{s_1 t + s_2 c}
\]

(9.5)

For an event of interest, say Event IV, one has

\[
w_{I-IV}(s1, s2) = (17e^{8s_1} + 40s_2)[0.24e^{2s_1} + 11.5s_2 + 0.24e^{2s_1} + 11.5s_2]
\]

\[+ 0.36e^{2s_1 + 20s_2}/(e^{s_1 + 20s_2})(0.7e^{s_1 + 1.5s_2})
\]

(9.6)

The performance measure associated with event IV are computed by
\[ p_{1-N} = w_{1-N}(0,0) - (.7)(.84)(.7) = 0.4116, \]

\[
E\{t_{1-N}\} = \left( \frac{\partial}{\partial s_1} \right) \left[ \frac{1}{p_{1-N}} w_{1-N} (s_1, 0) \right]_{s_1=0} = 14.90 \text{ weeks},
\]

\[
E\{C_{1-N}\} = \left( \frac{\partial}{\partial s_1} \right) \left[ \frac{1}{p_{1-N}} w_{QIV} (0, s_2) \right]_{s_2=0} = \text{Rs. 76,643 (thousands)}
\]

### 9.3 Queuing analysis using GERT

An example of a machine-spares-repairman problem is considered here for analysis. A plant consists of one machine and a spare machine. The time between failures for the machine is exponential with mean rate \( \lambda \). A single repairman is employed and his service time is exponential with mean rate \( \mu \). The GERT network describing this queuing system is given in Fig.9.8 where the node represents the number of failed machines.

![GERT network](image)

Fig 9.8. GERT representation of machine - spare - repairman problem

The branch from node 1 to node 2 represents the activity that the spare fails before the machine is fixed which has a probability equal to \( \lambda / (\lambda + \mu) \) and the time to fail, given failure occurs before service, is exponential with mean rate \( (\lambda + \mu) \). A similar analysis is required for the activity between node 1 and node 0.
The information obtainable from the GERT analysis is given in Table 9.4.

In addition to the information presented in Table 9.4, the steady-state probabilities can be obtained from the following equation:

\[ P_i = \frac{\text{Expected time in state } i \text{ during a cycle}}{\text{Expected cycle time}} \]

**TABLE 9.4**

<table>
<thead>
<tr>
<th>Description</th>
<th>Network Modification</th>
<th>Moment Generating Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time to plant failure</td>
<td>Delete Branch 2-1</td>
<td>[ M_{0.2}(s) = \frac{\frac{\lambda}{(\lambda+\mu)}[1-s/(\lambda+\mu)]^{-1}}{1-\frac{\mu}{(\lambda+\mu)}[1-s/(\lambda+\mu)]^{-1}}(1-s/\lambda)^{-1} ]</td>
</tr>
<tr>
<td>Duration of plant failure</td>
<td></td>
<td>[ M_{2,1}(s) = (1-s/\mu)^{-1} ]</td>
</tr>
<tr>
<td>Busy period</td>
<td>Delete Branch 0-1</td>
<td>[ M_{1.0}(s) = \frac{[-\frac{\mu}{(\mu+\lambda)}][1-s/(\lambda+\mu)]^{-1}}{1-[-\frac{\lambda}{(\mu+\lambda)}][1-s/(\lambda+\mu)]^{-1}}(1-s/\mu)^{-1} ]</td>
</tr>
<tr>
<td>Idle time</td>
<td></td>
<td>[ M_{0.1}(s) = (1-s/\lambda)^{-1} ]</td>
</tr>
<tr>
<td>State 0 renewal time</td>
<td>Split node 0 into node 0 and node 0'</td>
<td>[ M_{0.0}(s) = M_{0.1}(s) M_{1.0}(s) ]</td>
</tr>
</tbody>
</table>

It can be said that GERT is a general procedure for the formulation and evaluation of systems using a graphical approach. The GERT approach to problem solving utilizes the following steps:

1. Convert a qualitative description of a system or problem to a model in stochastic network form.
2. Collect the necessary data to describe the branches of the network.
3. Determine the equivalent function or functions of the network.

4. Convert the equivalent function into the following two performance measures of the network:
   The probability that a specific node is realized.
   The moment generating function of the time associated with an equivalent network

5. Make inferences concerning the system under study from the information obtained in 4 above