

Generating Probability Tables of Dynamic Reliability Graph with General Gates by a Software Tool

Gyoung Tae Goh^{a*}, Poong Hyun Seong^a

^aDepartment of Nuclear and Quantum Engineering, KAIST
 373-1, Guseong-Dong, Yuseong-Gu, Daejeon, South Korea, 305-701

*Corresponding author: phseong@kaist.ac.kr

1. Introduction

Fault tree method is the most widely used among methods to analyze system reliability, but it is not an intuitive method. So, as a system becomes complex, a corresponding fault tree becomes much more complex.

A reliability graph with general gates (RGGG) is an intuitive method to analyze system reliability; it can make a one-to-one match from the actual structure of a system to the reliability graphs of the system [1]. However, RGGG cannot capture the dynamic behavior of the system associated with time dependent events.

To overcome this shortcoming, dynamic reliability graph with general gates (DRGGG) was proposed [2]. By using discrete-time method, we can add dynamic nodes to RGGG. However, as a discretization number (interval number) n becomes larger, it becomes harder to apply this dynamic method to a real system, because the large number n leads to the complexity in making probability tables.

In this paper, we will introduce the software tool which generates probability tables automatically.

2. Rules of Making the Probability Tables

We use the discrete-time method [3,4] to determine the probability tables of each DRGGG node. In this section, the discrete-time method and rules of making the probability tables for each node are described.

2.1. Discrete-Time Method

We divide the total process time line into n same intervals as shown Figure 1.

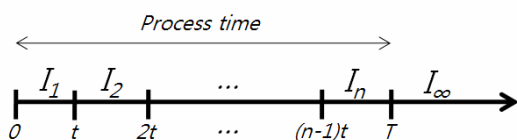


Figure 1. Discretization of process time.

The output of each node is one of $\{I_1, I_2, \dots, I_n, I_\infty\}$. I_k means that the node is failed in k th time interval and I_∞ means that the node is never failed.

If P_{ij}^k denotes the probability that an arc (a_{ij}) from node i (n_i) to node j (n_j) fails in the k th interval, and if $F_{ij}(t)$ denotes the cumulative failure distribution function of a_{ij} , we can derive P_{ij}^k as follows:

$$P_{ij}^k = \int_{(k-1)t}^{kt} \frac{dF(t)}{dt} dt \quad (1)$$

2.2. Calculation Rules

The number of blanks that should be filled is $(n+1)^3$ per each node (with 2 inputs).

Let the outputs of $n_1, n_2, n_3, n_E, n_F,$ and n_G be $I_x, I_y, I_z, I_e, I_f,$ and $I_g,$ respectively ($x, y, z, e, f, g \in 1, 2, \dots, n, \infty$).

2.2.1 PAND node

In the table, each blank that is defined by $x, y,$ and e can be filled in on the basis of the following rules:

- i. If $e > y, 0,$
- ii. If $e = y \leq x,$
 $Pr\{a1E \text{ fails before the } eth \text{ interval}\} \cdot (1 - Pr\{a2E \text{ fails before the } eth \text{ interval}\}),$
- iii. If $e \leq x, e < y,$
 $Pr\{a1E \text{ fails before the } eth \text{ interval}\} \cdot Pr\{a2E \text{ fails at the } eth \text{ interval}\},$
- iv. If $e = y > x, 1 - Pr\{a2E \text{ fails before the } eth \text{ interval}\},$
- v. If $x < e < y, Pr\{a2E \text{ fails at the } eth \text{ interval}\},$
- vi. If $e = \infty,$
 $1 - (\text{sum of the other probabilities in the same row}).$

2.2.2 Spare node

In the table, each blank that is defined by $x, y,$ and f can be filled in on the basis of the following rules:

- i. If $f > x, y, 0,$
- ii. If $f < x, y, Pr\{a1F \text{ fails at the } fth \text{ interval}\} \cdot Pr\{a2F \text{ fails at or before the } fth \text{ interval}\} + Pr\{a1F \text{ fails before the } fth \text{ interval}\} \cdot Pr\{a2F \text{ fails at the } fth \text{ interval}\},$
- iii. If $x < f < y, Pr\{a2F \text{ fails at the } fth \text{ interval}\},$
- iv. If $y \leq f < x, Pr\{a1F \text{ fails at the } fth \text{ interval}\},$
- v. If $f = x < y, Pr\{a1F \text{ doesn't fail before the } fth \text{ interval}\} \cdot Pr\{a2F \text{ fails at or before the } fth \text{ interval}\} + Pr\{a1F \text{ fails before the } fth \text{ interval}\} \cdot Pr\{a2F \text{ fails at the } fth \text{ interval}\},$
- vi. Else,
 $1 - (\text{sum of the other probabilities in the same row}).$

2.2.3 SEQ node

The SEQ node only allows basic events as inputs

because if $n1$, $n2$, and $n3$ have inputs the SEQ node cannot constrain the failure order of the inputs. We therefore explain only one case in which x , y , and z are ∞ . Each blank under that case can be filled in on the basis of the following rules:

i. If $g < 3$, 0.

ii. If $g \geq 3$,

$\sum_{a,b,c} Pr\{a1G \text{ fails at the } a\text{th interval}\} \cdot Pr\{a2G \text{ fails at the } b\text{th interval}\} \cdot Pr\{a3G \text{ fails at the } c\text{th interval}\}$,
 when $a + b + c = g$.

iii. If $g = \infty$,

$1 - (\text{sum of the other probabilities in the same row})$.

3. Example of Using Software

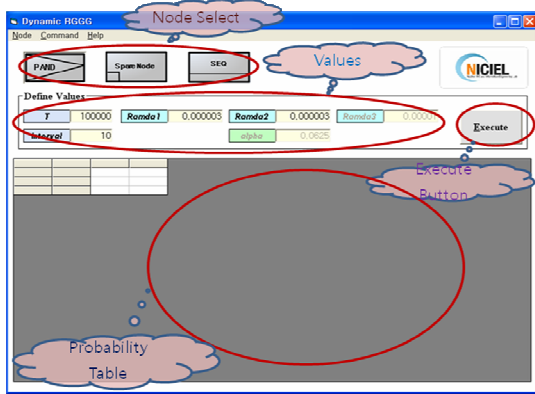


Figure 2. DRGGG Calculation Tool

Figure 2 shows the developing software tool which is generated automatically by using rules of making the probability tables.

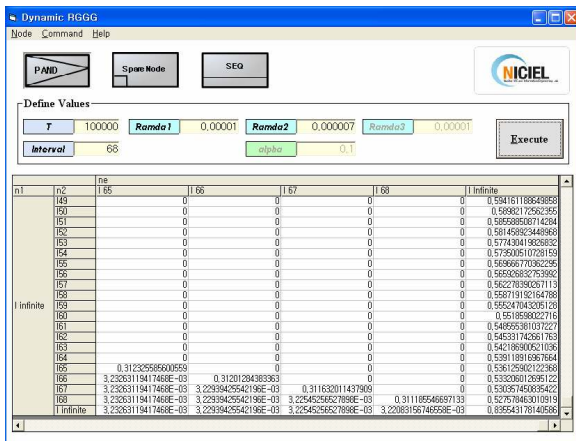


Figure 3. Result of PAND node when $n=68$

Figure 3 shows an example of calculation result about PAND node when $n=68$. To obtain more accurate result, the larger number of n is required. Spare and SEQ node's probability tables can be generated by this software tool.

4. Discussion and Further Study

By using this DRGGG software tool, each DRGGG node's probability table is generated automatically. However, to analyze reliability of real system, DRGGG have to be converted to an equivalent Bayesian network. Therefore we are developing this software tool so that DRGGG tool connect to a Bayesian network software tool such as Microsoft Belief Networks. (Figure 4)

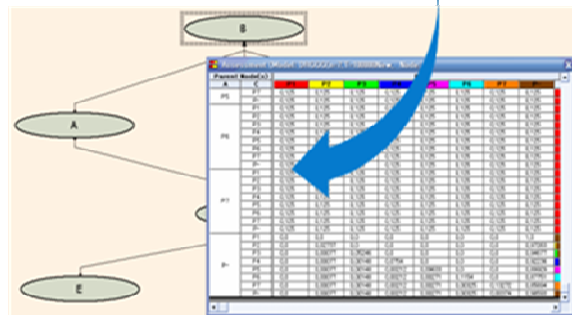
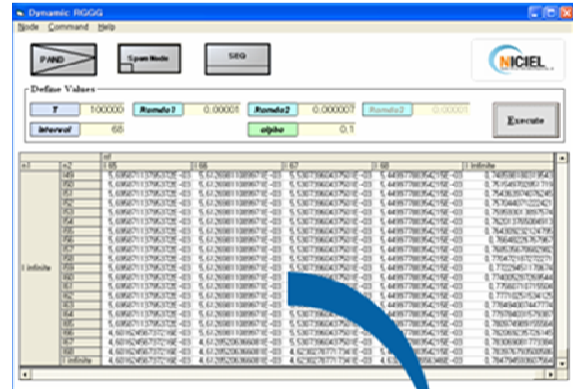


Figure 4. Connecting DRGGG tool to Bayesian Network tool

REFERENCES

[1] M. C. Kim and P. H. Seong, Reliability graph with general gates: an intuitive and practical method for system reliability analysis, Reliability Engineering and System Safety, Vol.78, pp.239-246, 2002.
 [2] S. K. Shin and P. H. Seong, Adding Dynamic Nodes to RGGG and Making Probability Tables, Transactions of the American Nuclear Society, Vol. 97, P. 131-132, 2007
 [3] S. F. Galan and F. J. Diez, Networks of probabilistic events in discrete time, International Journal of Approximate Reasoning, Vol.30, pp.181-202, 2002.
 [4] H. Boudali, J. B. Dugan, A discrete-time Bayesian network reliability modeling and analysis framework, Reliability Engineering and System Safety, Vol.87, pp337-349, 2005.