Find Root Cause by Using Google Page Rank Algorithm

Young Ho Chae [a*], Poong Hyun Seong [a]
[a]*Department of Nuclear and Quantum Engineering, KAIST, 291 Daehak-ro, Yuseong-gu, Daejeon 34141, Korea*
[*]Corresponding author: cyhproto@kaist.ac.kr

1. Introduction

Nowadays, we live with tremendous number of information. According to the article in 2015, in Earth, 2.5 Quintillion bytes of data is generated every day. Therefore, the ability to judge the quality of information and gather only the necessary information is an essential, not an option. In 1998, Sergey Brin and Larry Page thought that with conventional search environment, it is hard to find the information what they really want. Therefore, they tried to rank the usefulness of web pages and as a result of effort, they finally made "Google Page Rank Algorithm" [1]. Conventional search engines used keywords to find the page what user searched. However, Google Page Rank Algorithm evaluates web page's value by using random suffer. When multiple web pages are linked together, Google Page Rank Algorithm Calculates the probability that random suffer will stay on each web page. The more web pages that are linked, the higher the probability that random suffer will stay. By ordering web pages with random suffer probability, user can get the order of web pages with impact order.

When an accident happens at a nuclear power plant, operator will hear various alarms simultaneously. Alarm system has quite complex structure. Alarms are induced by many components, and also some alarms trigger other alarms. Such simultaneous ringing alarms make operator hard to recognize and prioritize their work. Therefore, in this paper, the method to figuring out the which alarm can be dealt as root cause by using Google Page Ranking Algorithm. The limitation of this method is that the method is hard to consider totally independent alarms. Nevertheless, providing information about which alarms can solve the most problems can be a good indicator for operator. For the further work we will try to consider independent alarms.

2. Google Page Rank Algorithm

In this section, the principle of Google Page Rank Algorithm and the convergence criteria of Google Matrix will be explained with example network (Figure. 1). The solution for the example network (Figure.1) will be also provided.

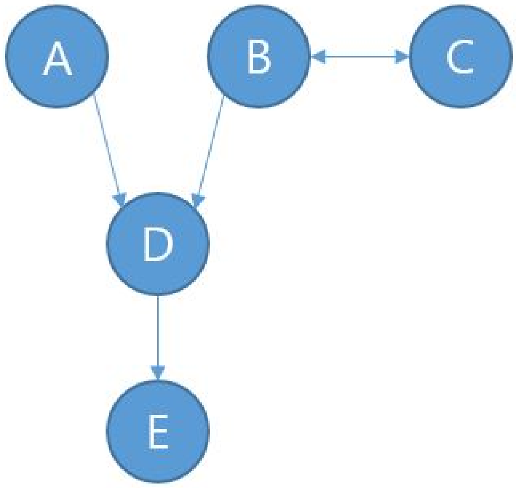

Figure 1 Example Network Architecture

2.1 Adjacent Matrix & Normalized Adjacent Matrix

At first stage, adjacent matrix which describes physical network architecture is needed. The adjacent matrix A can be constructed by the rule below.

1. If there is an arc from node i to node j, then the element of adjacent matrix will be $A_{i,j} = 1$
2. If there is any arc from node i to node j, then the element of adjacent matrix will be $A_{i,j} = 0$

With this rule, the adjacent matrix for example network will be:

$$A = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Adjacent matrix describes the relation between nodes. However, we need a normalize because the sum of a node's impact $A_{i,:}$ (effect starts from node i) should be 1 (or equal). Therefore, we have to normalize the adjacent matrix A. Adjacent matrix can be made by the rule below.

$$N_j = \frac{A_j}{\sum_{i=1}^{n} A_{i,j}}$$

`With this rule, the normalized adjacent matrix for example network will be:

$$N = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0.5 & 0 & 0 & 0 \\ 1 & 0.5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

2.2 Stochastic Matrix for Convergence Criteria

If there are any dangling nodes (The node only receives arc), then with recursive calculation, the matrix will converge to some value. However, if there are dangling nodes, then the matrix will not converge or it converges to wrong value (especially 0). The recursive calculation result with N matrix is shown as figure (2).

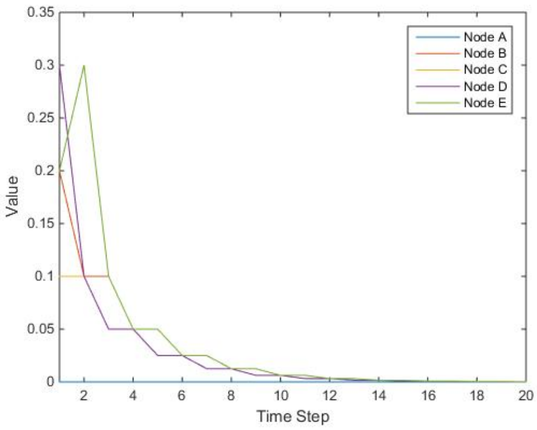

Figure 2 Calculation Result of non-stochastic matrix

Therefore, to assure matrix's convergence, column stochastization is needed. The column stochastic matrix should satisfy the condition that square matrix of nonnegative real numbers with each row and column summing to 1. With the rule in below, normalized matrix N can be converted to column stochastic matrix.

$$S = N + \frac{e * a^T}{n}$$

Where, e is the column vector that all elements are 1 and n is the size of the matrix and a is the column vector which is defined as follows:

1. If, $\sum_{i=1}^{n} N_{i,j} = 0$ then $a_j = 1$
2. Else, $a_j = 0$

Finally, stochanized matrix can be written as follows:

$$S = \begin{bmatrix} 0 & 0 & 0 & 0 & 0.2 \\ 0 & 0 & 1 & 0 & 0.2 \\ 0 & 0.5 & 0 & 0 & 0.2 \\ 1 & 0.5 & 0 & 0 & 0.2 \\ 0 & 0 & 0 & 1 & 0.2 \end{bmatrix}$$

2.3 Google Matrix

With only stochastic condition, the matrix may not have a specific convergence point. To have specific convergence point, the matrix should satisfy not only stochastic condition but irreducible condition. If the matrix A does not have p, q values and matrix P, within the condition $P^T AP = \begin{bmatrix} p & q \\ 0 & r \end{bmatrix}$. Then the matrix A is called as irreducible matrix. According to Horn & Johnson, primitive matrix always satisfies irreducible condition and also from Perron-Frobenius theorem, the necessary and sufficient condition for primitive matrix is that the matrix A^k should be positive with positive integer k. To satisfy both irreducible and stochastic condition Sergey Brin and Larry Page modified the matrix as follows:

$$G = m * S + (1 - m)E$$

Where, m is the number in between 0 to 1, and $E = \frac{e * e^T}{n}$. Usually Google Search engine uses 0.85 as m value. The reason why 0.85 is suitable value for google matrix is described in [2].

Finally, the Google Matrix for our Example network can be written as follow (m=0.85):

$$G = \begin{bmatrix} 0.0300 & 0.0300 & 0.0300 & 0.0300 & 0.2000 \\ 0.0300 & 0.0300 & 0.8800 & 0.0300 & 0.2000 \\ 0.0300 & 0.4550 & 0.0300 & 0.0300 & 0.2000 \\ 0.8800 & 0.4550 & 0.0300 & 0.0300 & 0.2000 \\ 0.0300 & 0.0300 & 0.0300 & 0.8800 & 0.2000 \end{bmatrix}$$

2.4 Result

From the Google Matrix, the impact of nodes A, B, C, D, and E are 0.0780, 0.2258, 0.1739, 0.2402 and 0.2821. Convergence speed and impact reflected network diagram are shown in figure 3 and4.

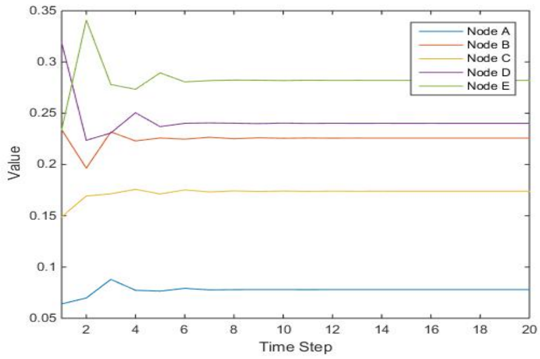

Figure 3 Calculation Result of Google Matrix

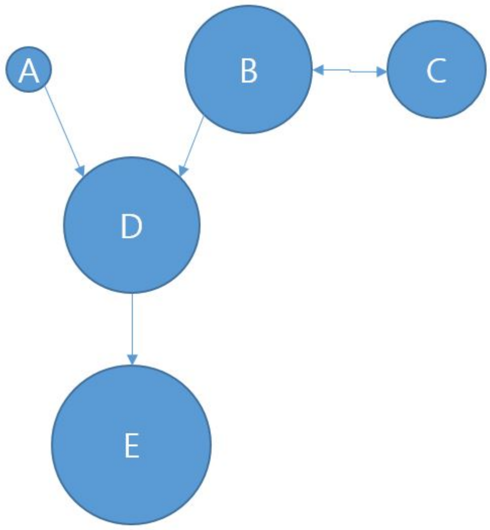

Figure 4 Impact Reflected Network Diagram

From the calculation result, the most significant node is node E. Because Node E contains information from D and node D contains information from A, B and C. The least significant node is node A. Node A has only relation with node D. Therefore, it's impact in the network is small.

3. Application of Google Page Rank Algorithm to Nuclear Power Plant's Alarm System

Nuclear power plant (OPR-1000) has 56 alarms, 10 reactor protection system signals, 11 control related signals, 5 safety system actuation signals and 18 reactor shut down signals. In this section, we provide an example how the Google Page Rank Algorithm can prioritize alarms at LOCA (Loss of Coolant Accident) situation. LOCA simulation is done by using Compact Nuclear Simulator. The size of LOCA is 100cm^2 break.

When the LOCA take place, 15 alarms are occurred within 7 seconds after the LOCA happened. (The list of list of alarm is provided in table 1.) We made the network with alarms and instrumentation components which induce alarms. For example, in case of PZR Press Low alert, we draw an arc start from PZR Pressure instrument to PZR Press Low alert. The result is shown in Figure 5.

Table 1 Occured Alarms

Alarm#	Description	Time
01	PZR Press Low Back Up Heater On	0:0:1
02	PZR Press Low Alert	0:0:1
03	OTdeltaT Rx Trip	0:0:2
04	PWR Range High Flux Rate Rx Trip	0:0:3
05	TBN Trip P-4	0:0:3
06	TBN Trip & P-7 Rx Trip	0:0:3
07	Tref/Auct Tavg Deviation Hi	0:0:3
08	SG 1, 2, 3 STM/FW flow Deviation	0:0:3
09	Two or More Rod at Bottom	0:0:4
10	PZR Low Press & P-7 Rx Trip	0:0:6
11	RCS 123 Tavg/Auct Tavg Deviation Hi	0:0:6
12	Rad Hi	0:0:6
13	PZR Press Lo SI	0:0:7
14	FWP Trip	0:0:7
15	AFW Actuated	0:0:7

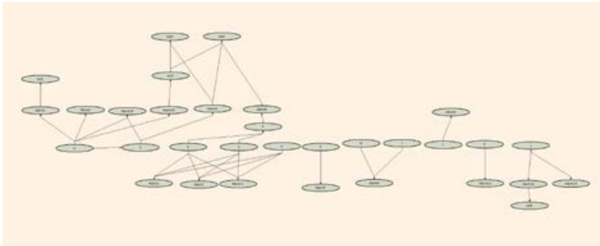

Figure 5 Alarm Network Structure

By applying Google Page Rank method to example, the result is shown in Figure 6.

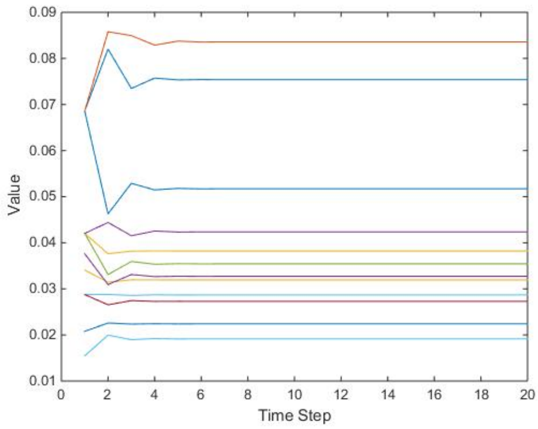

Figure 6 Network Analysis Result

As a result, alarms which have low hierarchy are PZR Press Low Back Up Heater On, PZR Press Low Alert and PZR Press Lo SI. The reason why we should focus on low valued alarm is they can be dealt as root causes. And also by looking the signals which have low value we can find the root signals. The signals which can be dealt as root causes in this experiment are PZR pressure, Hot Leg Temperature, Cold Leg Temperature and Main Steam Line Pressure. Therefore, operator should focus on why PZR pressure goes low and why the Hot Leg Temperature and Cold Leg Temperature are changing.

4. Further Work

By using Google Page Rank Algorithm to alarm system, we can figure out which alarm can be dealt as root causes. However, this method has limitation. This method is hard to consider totally independent alarms. Therefore, for the further work, we will try to make algorithm which can consider this kind of problems.

5. Conclusion

In this paper, the principle of Google Page Rank Algorithm is explained with example network. And also application of Google Page Rank Algorithm to the nuclear power plant’s root causes classification is proposed. Suggested approach can provide information about which alarms can be dealt as root causes. We expect this method will help operator’s to prioritize their works.

REFERENCES

[1] Larry Page, The PageRank Citation Ranking: Bringing Order to the Web, 1998
[2] Amy N. Langville and Carl D. Meyer, Deeper Inside PageRank, 2004