Distribution Automation Monitoring Using Petri Nets

Mehdi Saki, Alireza Fereidunian, Hamid Lesani, and Mohamad Amin Sharifi Kolarijani

***Abstract*-- The growth of the electricity distribution networks and increasing application of distributed generators, has led to complexity of monitoring of automation process and possibility of making wrong decisions in operation of these networks. Hence, a system that can be housed in energy control centers to assist distribution network operators in diagnosing cases of failure diagnosis sounds necessary. Regarding this, our study employs Petri net modeling to provide a model of automation system on a distribution network that can be implemented in the monitoring system of distribution networks control centers, and using the information received at these centers, to monitor various stages of automation process, detect faulted sections, and also, to inform the operator about the incorrect states in a real time manner. Therefore, the proposed model reduces incorrect outages and increases reliability of distribution networks by informing the operator and assisting him to decide correctly.**

***Index Terms*--distribution automation; failure diagnosis; monitoring; Petri net; feeder reconfiguration; restoration; Smart Grid.**

I. Introduction

TODAY, expansion of electrical energy distribution networks together with increasing DG penetration has caused the monitoring of distribution automation to be very difficult and complicated. Abundance of distribution feeders and using several DG show that the traditional ways of monitoring by operator cannot handle this complexity and may lead to making wrong decisions during executing of automation sequences. Sending wrong control commands due to incorrect diagnosis by the operator of control centers results in false switching and consequently reducing the reliability of electrical energy distribution systems. Hence, a need for systems which can be housed in control centers and help the operators in making proper decisions sounds necessary [1].

For this reason, this research will present a system based on Petri net modeling which assists the operator to decide correctly during executing of automation sequences. Method of Petri net modeling has been employed in several researches in the field of power system monitoring. In papers [1-3], Petri net method has been used for modeling of primary and secondary protection systems in a power network and monitoring of correct function of these systems through occurring faults. Paper [4] has used this method for modeling of over current protection relays and diagnosis of malfunction of them in distribution feeders. But in none of the researches in this area, Petri nets have not been used for modeling and monitoring of automation process in distribution networks.

Petri net method is applicable for modeling of hierarchical and sequential processes; and therefore, it is an appropriate method for modeling of automation process. The most important property of Petri net modeling is utilizing of simple calculations which leads to decrease in the response time in spite of dealing with too much information in distribution networks.

In this research, at first, a model, based on Petri net modeling methods, has been designed and presented which can follow all sequences of distribution automation in real time from occurring fault in a section of a feeder till restoring of it to its initial normal state. Then using a proper diagnostic algorithm, will try to diagnose possible incorrect states during automation process and inform the operator for making correct decisions.

II. Principles of a Petri net

In General, a Petri net is a graphical network which consists of three main components: places, transitions and arcs. Usually, Circles are used for showing places and bars or rectangles are symbols of transitions. Amount of available resources of each place are showed by some dots which are called tokens (Fig. 1).

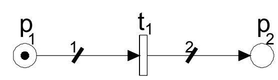

Fig.1. A simple Petri net view

Distribution of tokens in places follows two rules:

1. Enabled transition: A transition t is said to be enabled if each input place p of t contains at least the number of tokens equal to the weight of the directed arc connecting p to t.
2. Transition firing: only an enabled transition has possibility to fire or occur and a firing of an enabled transition t removes from each input place p the number of tokens equal to the weight of the directed arc connecting p to t. It also deposits in each output place p the number of tokens equal to the weight of the directed arc connecting t to p [5].

As an example in Fig. 1,t1 is an enabled transition and is possible to occur because one token exist in p1 and firing t1 removes one token from p1 and deposits two tokens to p2 according to weighting factor of their directed arcs (Fig. 2).

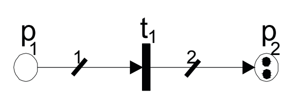

Fig. 2. Firing of t_1 in Fig. 1

Mathematical Description of a Petri Net

Although a Petri net is known as a graphical model but it is based on some matrix algebra rules. For a Petri net S containing n places and m transitions, the state equation can be written as below:

$$q_s[k+1] = q_s[k] + Bx[k] \quad (1)$$

Where $q_s[k+1]$ is a $m \times 1$ matrix as the state matrix at time instant $k \times 1$ and its i^{th} row entry shows the number of tokens exist in i^{th} place at time instant $k \times 1$. $q_s[k]$ is the state matrix of model in previous sequence.   is called incidence matrix and is defined as

$$B = [b_{ij}^+ - b_{ij}^-] \quad (2)$$

Where   is the weight of the arc from p_i to t_j and   is the weight of the arc from t_j to p_j.

$x[k]$ is the control vector and for Petri net consists of m transitions will be a $m \times 1$ matrix in which the entry related to each sequence is just one and other entries are zero. In other words, this matrix defines priority of transition firing [6].

III. Failure Diagnostic Algorithm

In order to diagnose a failure in a Petri net, the method presented in reference [1] will be used. In accordance with the mentioned method, some additional places named as monitoring places are added to the main Petri net forming a new Petri net like H which has a relation with main Petri net S for all K, as bellow:

$$q_h[k] = \begin{bmatrix} I_n \\ G \end{bmatrix} q_s[k] \quad (3)$$

Where I_n is the $n \times n$ identity matrix, G is a $d \times n$ matrix and is called generator matrix, and   is the number of added places.

Considering (3), the state equation of Petri net H can be written as

$$q_h[k+1] = q_h[k] + \begin{bmatrix} B \\ GB \end{bmatrix} x[k] \quad (4)$$

If a failure occurs at time instant k-1, the related state matrix cab be expressed as

$$q_f[k] = q_h[k] + e[k] \quad (5)$$

Where $q_h[k]$ is the state matrix of Petri net without failures and $e[k]$ is the effect of the failure.

In order to detect failures in a Petri net, a parity check matrix is defined as

$$S \equiv [-G \;\; I_d] \quad (6)$$

and an error syndrome matrix $s[k]$ at time instant   can be expressed as

$$s[k] \equiv [-G \;\; I_d] q_f[k] \quad (7)$$

A failure can be detected if $s[k]$ is nonzero and also it can be identified if the mentioned syndrome can uniquely be expressed by columns of parity check matrix. In this case the number of each column shows the failure place numbers.

IV. Distribution Automation

When a fault occurs in an electrical distribution network, corrective activities should be performed by energy control centers (ECC's) in order to restore the power and return the network to its initial normal state. These activities which may be automatically or manually done are a part of distribution automation (DA) system [7-9]. Generally, all sequences of automation in a distribution network with distributed generators can be categorized as below [10], [11]:

1- A distribution feeder is supposed to work normally.

2- A fault occurs in a section of the mentioned feeder and activates the feeder protection relay, DG protection relay, and fault indicators).

3- Feeder's circuit breaker will be opened and DG will be isolated from network due to tripping of the related protection relays.

4- Faulty section of feeder will be isolated from other sections using received information from fault indicators.

5- Healthy sections of feeder will be reenergized till clearing the fault by closing the proper circuit breakers.

6- De-energizing the mentioned healthy sections in previous sequence after clearing the fault.

7- Connecting the fault cleared section to the other sections of the feeder.

8- Energizing the whole feeder and returning to its initial normal state.

V. A Petri net-based Model for DA

In order to monitor automation process in a distribution network by using a Petri-net method, a model based on this method, which was described in part 2, should be designed and provided firstly. In this research, the distribution network presented in [11] has been borrowed (Fig. 3).

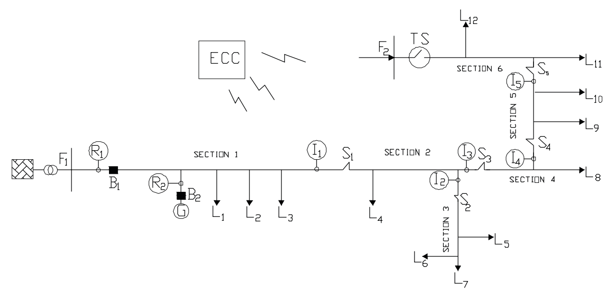

Fig. 3. A distribution network for modeling [11]

In order to provide a Petri net model for a network, the sequences described in part 4 will be used. This model has been illustrated in Fig. 4. Weighting of all arcs equals one; and, so it has not been shown.

As it can be seen, this model consists of two main blocks shown in dash lines. The upper block is used for detecting the faulty section of the feeder according to the received information from fault indicators. Note that some arcs have a small circle on their end – inhibitor arcs – and change the enabling rule in a way that existence of tokens in the corresponding input places prevents the transitions to be enabled [5].

TABLE I

Descriptions of the Transitions in Fig. 4

Transition	Description
t_x	Detecting faulty section
t_1	Occurring a fault
t_2	Tripping of the feeder's circuit breaker
t_3	Tripping of the DG's circuit breaker
t_{x4}	Isolating of the faulty section
t_{x5}	Reenergizing of the healthy sections
t_{x6}	De-energizing of the healthy sections after clearing the fault
t_{x7}	Connecting of the fault cleared section to other sections
t_{x8}	Energizing the feeder and connecting the DG
t_{x9}	Return to initial normal state of the feeder

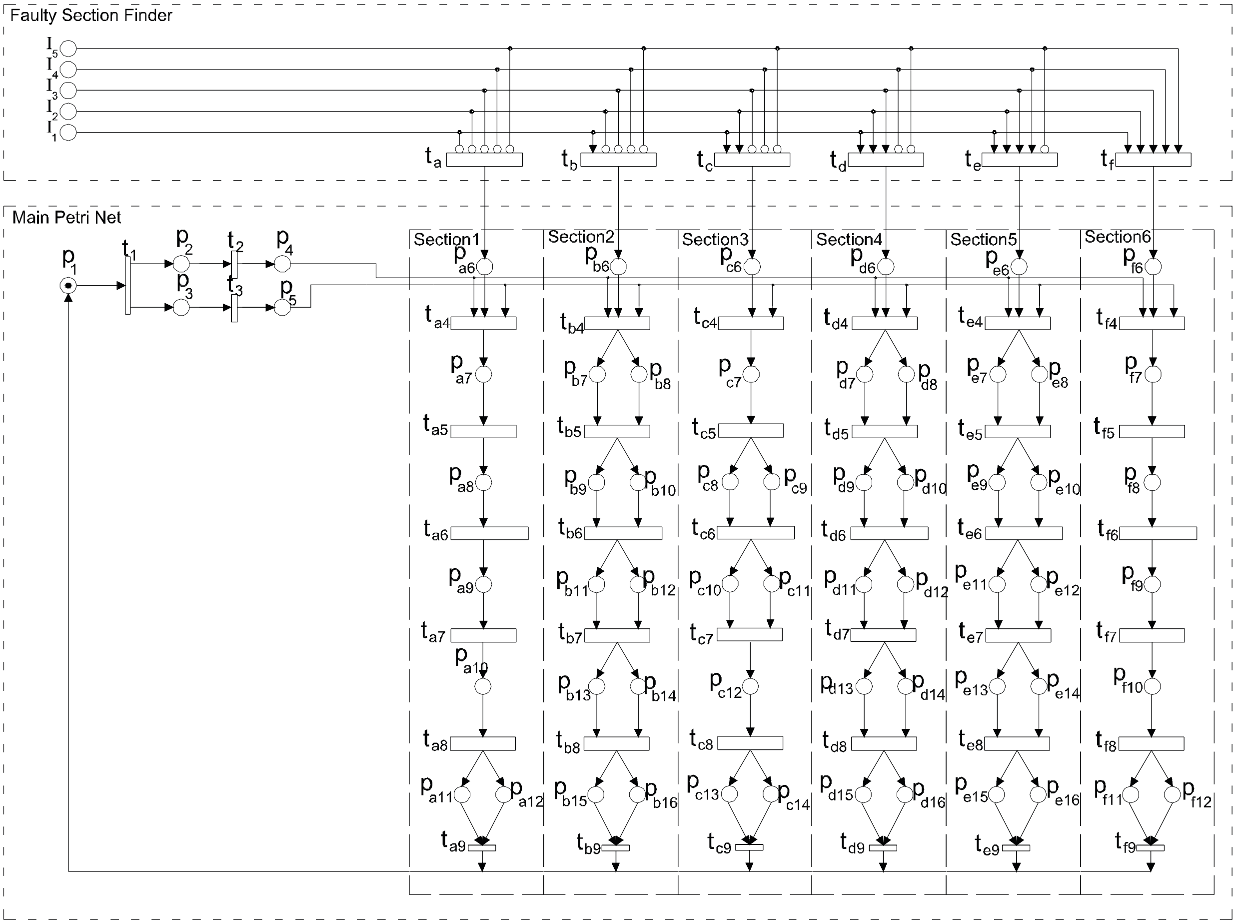

Fig. 4. A Petri net model for the network in Fig. 3

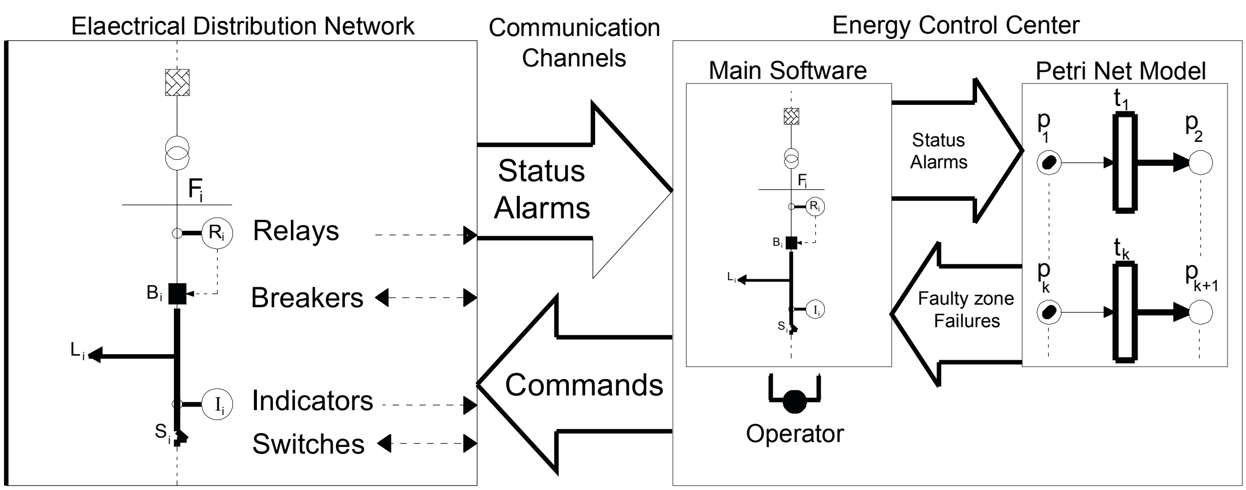

Fig. 5. The role of the monitoring program in an electrical distribution system

After detecting the faulty section, next sequences of DA continue according to part 4 which are modeled in lower block of the model of Fig. 4. As can be seen, the lower block consists of a common area and six separated small blocks related to six sections of the feeder of the network. When a fault occurs in a section, only one of the mentioned small blocks corresponding to the faulty section contributes in automation and other blocks have no effect. All transitions of the Petri net model have been described in Table I.

VI. ROLE OF PRESENTED MODEL IN MONITORING

In distribution automation process, decisions are made by operator based on information sent from distribution network to control centers, and necessary commands are sent to the network.

In this research, a program has been provided in Matlab based on failure diagnostic algorithm described in part 3 which should be installed onto ECC's computer and using its available information to monitor sequences of automation, diagnose failures, and inform the operator in a real time manner. The presented program prevents from making wrong decisions and assists operator to take correct action in response to various conditions. Fig. 5 shows the role of the mentioned program in an electrical distribution system.

In general, different information is communicated between a distribution network and ECC. The parts that are used among them, in our Petri net model can be categorized as below:

- Status of circuit breakers
- Status of sectionalizing switches
- Alarms of fault indicators
- Alarms of protection relays
- A signal which informs us that the fault was cleared

Automation software which is based on Petri net receives this information and activates places that are corresponding to each of them in the Petri net model. If the received information conforms to expected pattern of the Petri net model, the approval signal will be issued from the program and next action will be recommended to the operator. Vice versa, if the received information does not conform to the pattern, the error messages will be issued which can be listed as below:

- Informing the fault place in the related feeder
- Informing the sequence of occurring failure
- Informing the failed signals

VII. CASE STUDY

In this part our model will be practically tested through two scenarios which examine the model in diagnosing of the fault place, and incorrect cases, for distribution network. These scenarios are based on a fault occurred in section 2 of the network (Fig. 6).

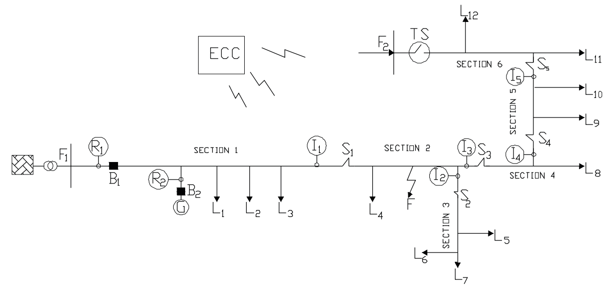

Fig. 6. A fault has occurred in section 2

Before presenting the scenarios, the necessary parameters of Petri net should be calculated firstly. When a fault occurs in a section, just one of the blocks corresponding to that section will take part in modeling the automation process; therefore, the matrices of the Petri net model are calculated for each section separately. This method makes the matrices very smaller; and, consequently, the calculations will be reduced considerably. Thus, there are six sets of matrices in accordance with six sections in our sample network. In this case which the fault has been occurred in section 2, the related matrices can be calculated as below:

a. Incidence matrix: according to small block of Petri net presented in Fig. 4 which is related to section 2, the matrix can be written as in (8):

$$
B_2 = \begin{array}{c} \\ p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \\ p_{b6} \\ p_{b7} \\ p_{b8} \\ p_{b9} \\ p_{b10} \\ p_{b11} \\ p_{b12} \\ p_{b13} \\ p_{b14} \\ p_{b15} \\ p_{b16} \end{array}
\begin{array}{c}
\begin{array}{ccccccccc} t_1 & t_2 & t_3 & t_{b4} & t_{b5} & t_{b6} & t_{b7} & t_{b8} & t_{b9} \end{array} \\
\begin{bmatrix}
-1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1
\end{bmatrix}
\end{array}
\qquad (8)
$$

b. Initial state matrix: for a fault occurred in section 2, corresponding places have been described in Table II. In this case, the number of places are $n=16$ according to corresponding block of model; and, thus its initial state matrix can be written as in (9).

TABLE II
DESCRIPTION OF PLACES WHEN A FAULT OCCURRED IN SECTION 2

Place	Description
p_1	The feeder is in normal state
p_2	Feeder protection R_1 has been activated
p_3	DG protection R_2 has been activated
p_4	Feeder circuit breaker B_1 has been opened
p_5	DG circuit breaker B_2 has been opened
p_{b6}	The fault has been occurred in section 2
p_{b7}	Switch S_1has been opened by ECC according to section of fault
p_{b8}	Switch S_2has been opened by ECC according to section of fault
p_{b9}	Feeder circuit breaker B_1has been closed by ECC
p_{b10}	Tie switch *TS* has been closed by ECC
p_{b11}	Feeder circuit breaker B_1has been opened by ECC
p_{b12}	Tie switch *TS* has been opened by ECC
p_{b13}	Switch S_1has been closed by ECC
p_{b14}	Switch S_2has been closed by ECC
p_{b15}	Feeder circuit breaker B_1has been closed by ECC
p_{b16}	DG circuit breaker B_2 has been closed by ECC

$$q_s[0]=\begin{array}{c}\begin{array}{cccccccccccccccc} p_1 & p_2 & p_3 & p_4 & p_5 & p_{b6} & p_{b7} & p_{b8} & p_{b9} & p_{b10} & p_{b11} & p_{b12} & p_{b13} & p_{b14} & p_{b15} & p_{b16}\end{array}\\ \left[\begin{array}{cccccccccccccccc} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{array}\right]^T\end{array} \qquad (9)$$

c. Control vector matrix: the number of transitions is $m=9$ and the control vector matrix for first sequence can be written as (10).

$$x_i\Big|_{i=1}=\begin{array}{c}\begin{array}{ccccccccc} t_1 & t_2 & t_3 & t_{b4} & t_{b5} & t_{b6} & t_{b7} & t_{b8} & t_{b9}\end{array}\\ \left[\begin{array}{ccccccccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{array}\right]^T\end{array} \qquad (10)$$

where i is the sequence number. For each sequence, the i^{th} entry of x_i will be one and other entries will be zero depicting the priority of the sequences.

d. Generator matrix: in order to calculate the generator matrix G, we have used the method presented in [2]. So it can be calculated as (11).

$$G=\begin{bmatrix} 1 & 1 & 1 & \cdots & 1\\ 1 & 2 & 3 & \cdots & n\\ 1 & 2^2 & 3^2 & \cdots & n^2\\ \vdots & \vdots & \vdots & \ddots & \vdots\\ 1 & 2^{d-1} & 3^{d-1} & \cdots & n^{d-1}\end{bmatrix} \qquad (11)$$

where d is the number of monitoring places which will be added to the main Petri net.

The number of monitoring places depends on the number of the failures which are expected to diagnose concurrently every sequence. If we expect from the model to diagnose h failures every sequence, then the number of the monitoring places which should be added to main Petri net is d=2h [4]. Now that we want to diagnose h=4 failures, we need d=2h=8 additional places. Thus, G2 can be written as

$$G_2=\begin{bmatrix} 1 & 1 & 1 & \cdots & 1\\ 1 & 2 & 3 & \cdots & 16\\ 1 & 2^2 & 3^2 & \cdots & 16^2\\ \vdots & \vdots & \vdots & \ddots & \vdots\\ 1 & 2^7 & 3^7 & \cdots & 16^7\end{bmatrix} \qquad (12)$$

Note that if a transition in the model cannot be monitored, none of the monitoring places have connections to it; and, accordingly, the corresponding column of matrix $G \times B$ will be zero [1]. Since t_1 is not able to be monitored, the first column of $G{\times}B$ should be zero:

$$[G_2 B_2](:,1)=0 \qquad (13)$$

Where $[G_2B_2](:,1)$ represents the first column of G_2B_2. According to (13), the first column of G_2 can be rewritten as

$G_2(:,1)=[\underline{2513\ 35972757932315}]^T$ (14)

Now the parity check matrix can be simply calculated as

$$S_2=[-G_2 I_8] \qquad (15)$$

Scenario 1: suppose that automation sequences have been correctly executed to sequence forth. q_f can be calculated as

$$q_f[1]=\begin{array}{c}\begin{array}{cccccccccccccccc|cccc} p_1 & p_2 & p_3 & p_4 & p_5 & p_{b6} & p_{b7} & p_{b8} & p_{b9} & p_{b10} & p_{b11} & p_{b12} & p_{b13} & p_{b14} & p_{b15} & p_{b16} & \multicolumn{4}{c}{\text{monitoring places}}\\ & & & & & & & & & & & & & & & & 1 & 2 & \cdots & 8\end{array}\\ \left[\begin{array}{cccccccccccccccc|cccc} 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 11 & \ldots & 282251\end{array}\right]^T\end{array} \qquad (16)$$

$$q_f[2]=\left[\begin{array}{cccccccccccccccc|cccc} 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 13 & \ldots & 298507\end{array}\right]^T \qquad (17)$$

$$q_f[3]=\left[\begin{array}{cccccccccccccccc|cccc} 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 15 & \ldots & 374445\end{array}\right]^T \qquad (18)$$

$$q_f[4]=\left[\begin{array}{cccccccccccccccc|cccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 15 & \ldots & 2920695\end{array}\right]^T \qquad (19)$$

where q_f [4] has been updated by received information of ECC as below:

Open status of $S_1=0$

Open status of $S_2=0$

According to (7), error syndrome matrix can be written as

$$s_f[4]=[2\ 15\ 113\ 855\ 6497\ 49575\ 379793\ 2920695]^T \qquad (20)$$

where it can be rewritten as

$$s_f[4]=S_2(:,7)+S_2(:,8) \qquad (21)$$

Equation (21) shows that failures have been occurred in places 7th and 8th which means that open status of S1 and S2 has not been received. The output window of the program has been shown in Fig. 7.

In Fig. 7, first line represents the section of occurring fault. All preconditions of each sequence will be checked by the program before execution and a message like second line in Fig. 7 will be issued. Next proper decision will be recommended to the operator and his approval will be requested continuation (3rd and 4th lines in this figure). Finally failed signals will be informed in last lines.

```
7/22/11 4:00 PM                    MATLAB Command Window

fault occurred in section 2
all preconditions are ok
please open S1&S2 to isolate section 2, do you agree?[y/n]y
"isolating sequence" failed (seq.4 fail)
fail on following signals:
open status of S1 not received
open status of S2 not received
>>
```

Fig. 7. Output window of the program for scenario 1

Scenario 2: suppose that following signals have been received in ECC:

close status of $S_1=1$

close status of $S_2=1$

Matrix $q_f[6]$ can be written as below after updating by these signals

$$q_f[6]=\begin{array}{c}\begin{array}{cccccccccccccccc|cccc} p_1 & p_2 & p_3 & p_4 & p_5 & p_{b6} & p_{b7} & p_{b8} & p_{b9} & p_{b10} & p_{b11} & p_{b12} & p_{b13} & p_{b14} & p_{b15} & p_{b16} & \multicolumn{4}{c}{\text{monitoring places}}\\ & & & & & & & & & & & & & & & & 1 & 2 & \cdots & 8\end{array}\\ \left[\begin{array}{cccccccccccccccc|cccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 2 & 23 & \ldots & 55318979\end{array}\right]^T\end{array} \qquad (22)$$

And error syndrome matrix can be written as below consequently

$$s_f[6]=[0\ {-4}\ {-100}\ {-1882}\ {-31600}\ {-499234}\ {-7598800}\ {-112843042}]^T \qquad (23)$$

where it can be rewritten as

$$s_f[6]=S_2(:,11)+S_2(:,12)+S_2(:,13)+S_2(:,14) \qquad (24)$$

This means that there are failures on places 11th to 14th that are related to open status of B_1 and TS and close status of S_1 and S_2. Fig. 8 shows the output window of this scenario.

```
7/22/11 4:31 PM                    MATLAB Command Window

fault occurred in section 2
all preconditions are ok
please open S1&S2 to isolate section 2, do you agree?[y/n]y
all preconditions are ok
please close B1&TS to switch healthy sections on, do you
agree?[y/n]y
"switch healthy section off" failed (seq.6 failed)
fail on following signals:
open status of B1 not received
open status of TS not received
close status of S1 received unexpectedly
close status of S2 received unexpectedly
>>
```

Fig. 8. Output window of the program for scenario 2

VIII. Conclusions

A Petri net based model for monitoring of distribution automation was presented in this research. The model can be installed on computer of ECC's using available information to monitor DA and issue proper messages to assist operator in the decision making phase.

Since the presented model uses the available information in ECC's, its ability to monitor the automation is based on the amount of this information gathered from distribution network. Hence, this model can be extended by increasing the received information especially in communication fields and also internal failures of some equipment like Circuit breakers. Therefore, gathering information from the network plays an important role in such monitoring models.

References

[1] Hadjicostis, C.N. and Verghese, G.C.: 'Power system monitoring using Petri net embeddings', *IEE Proc. –Gener. Transm. Distrib*., Vol. 147, No. 5, 2000, pp. 299-303

[2] Ren, H., Mi, Z., Zhao, H. and Yang, Q.: 'Fault diagnosis for substation automation based on Petri nets and coding theory', *IEEE Power Engineering Society General Meeting*, Vol. 1, 2004, pp. 1038-1042

[3] Lo, K.L., Ng, H.S., Grant, D.M., and Trecat, J., 'Extended Petri net models for fault diagnosis for substation automation', *IEE Proc. –Gener. Transm. Distrib*., Vol. 146. No. 3, May 1999, pp. 229-234

[4] Calderaro, V., Galdi, V., Piccolo, A. and Siano, P.: 'A Petri net based protection monitoring system for distribution networks with distributed generation', *Elsevier Electric Power Systems Research*, No. 79, 2009, pp. 1300-1307

[5] Zurawski, R. and Zhou, M.: 'Petri Nets and Industrial Applications: A Tutorial', *IEEE. Transactions on Industrial Electronics,* Vol. 41, No. 6, Dec. 1994, pp. 567-583

[6] Murata, T.: 'Petri Nets: Properties, Analysis and Applications', *Proc. of the IEEE*, Vol. 77, No. 4, April 1989, pp. 541-580

[7] Fereidunian, A., Lucas, C., Lesani, H., Rahmani, R., Wymore, A.W., "A Policy-Driven Method for IT Infrastructure Selection in Power Distribution Automation System", *International Review of Electrical Engineering*, Vol. 5, No. 2, Part B, March-April 2010, pp. 671-682.

[8] Fereidunian, A., Nordman, M., Lesani, H., Lucas, C., Lehtonen, M., "A Systems Approach to Information Technology (IT) Infrastructure Design for Utility Management Automation Systems", *Iranian Journal of Electrical and Electronic Engineering*, Vol. 2, No. 3&4, July 2006, pp. 91-105.

[9] Fereidunian, A.R., Lesani, H., Lucas,C., "Distribution System Reconfiguration Using Pattern Recognizer Neural Networks" *International Journal of Engineering (IJE), Transactions B: Applications*, Vol.15, No.2, July 2002, pp.135-144.

[10] He, Y., Andersson, G. and Allan, R.N.: 'Modeling the Impact of Automation and Control on the Reliability of Distribution Systems', *IEEE Power Engineering Society Summer Meeting*, Vol. 1, 2000, pp. 79-84

[11] He, Y., Soder, L.; Allan, R.N: 'Distribution automation: impact of communication system on reliability of automatic control', *IEEE Power Tech Proceedings*, Vol. 3, 2001, pp. 6-11

Biographies

Mehdi Saki is a M.Sc. student at Islamic Azad University, Tehran South Branch. He received the B.Sc. degree in electrical power engineering from the Ferdowsi University of Mashhad, Iran, in 2004. He serves as an electrical engineer with Mapna Company, Tehran, Iran. His research interests include power systems automation, and electrical protection systems for gas and steam power plants.

Alireza Fereidunian is an Assistant Professor at the K. N. Toosi University of Technology and serves as an independent consultant to GTEDC. He received his PhD and MSc from University of Tehran (SMRL, CIPCE), in 2009 and 1997, where he is a Post-Doctoral Research Associate now. His research interests include Smart Grid, power systems automation, and application of intelligent systems, human-automation interaction, data mining, decision-support, decision-making, IT and signal processing in power systems. He is a member of IEEE (a member of IEEE-SMC-HCI TC) and INCOSE (as INCOSE Iran point of contact).

Hamid Lesani is a Professor at the Center of Excellence for Control and Intelligent Processing (CIPCE), School of Electrical and Computer Engineering, University of Tehran. He received the M.S. degree in electrical power engineering from the University of Tehran, Iran, in 1975, and the Ph.D. degree in electrical engineering from the University of Dundee, U.K., in 1987. Early in his career, he served as a Faculty Member with Mazandaran University. After obtaining the Ph.D. degree, he joined the Department of Electrical and Computer Engineering, Faulty of Engineering, University of Tehran. His teaching and research interests are design and modeling of electrical machines and power systems. Professor Lesani is a member of IEEE (PES) and IEEE Iran Section.

Mohamad Amin Sharifi Kolarijani is a B.Sc. Student at the Center of Excellence for Control and Intelligent Processing (CIPCE), School of Electrical and Computer Engineering, University of Tehran. His research interests include systems theory, nonlinear control, discrete-event systems, and intelligent control.