Journal of Southeast University (English Edition) Vol. 22, No. 4, pp. 519 – 522 Dec. 2006 ISSN 1003—7985

Optimization of fault management system in power plants based on DD-RCM and TPN

Chen Lijuan[1] Yang Xin[2] Xia Anbang[1]

([1] School of Electrical Engineering, Southeast University, Nanjing 210096, China)

([2] School of Automation, Southeast University, Nanjing 210096, China)

Abstract: A method of optimizing the fault management system in power plants based on DD-RCM (result-chain based modeling for digital developing) and TPN (temporal Petri net) is proposed. First, the model of the fault management system was set using DD-RCM. Then, it was transformed to the temporal Petri net model by corresponding rules. Secondly, relationships among all the activities, such as choice, conflict, synchronization and concurrency, were confirmed according to the Petri net model and described employing the reengineering algorithm of incidence matrix. Thirdly, the Petri net model was reduced by combining reduction rules and conflict, synchronization relationships to optimize the fault management system in power plants. Finally, the functionality of the reduced net was proved by the temporal logic of the temporal Petri net.

Key words: result-chain based modeling for digital developing (DD-RCM); temporal Petri net; incidence matrix; reduction rules

Fault management is a complex process which needs to be optimized. First, a model must be set up to do some effective simulation. Only in this way, can we raise yield efficiency and guarantee product quality.

In this paper, the model of the fault management system in power plants is discussed based on DD-RCM and TPN. A method for optimizing the fault management system is proposed. First, a model is founded to supply a strategic view. Then, we transform DD-RCM into a temporal Petri net which can confirm choice, conflict, synchronization and concurrency relationships. All the relationships are described using the reengineering algorithm. Finally, temporal logic and the reengineering algorithm are employed to optimize the model of the fault management system.

1 Fault Management System Description Based on DD-RCM

DD-RCM contains four kernel elements[1]: activity (represented by a square box), result (represented by a circle), influence (represented by a beeline arrowhead) and supposing (represented by an equilateral hexagon). The benefit implement approach is fully described using these four elements to form the result-chain model. The basic logic of DD-RCM, concurrency and choice, is detailedly introduced in Ref. [1].

DD-RCM is always employed to set up a model in a complex process. Here, we take fault management in power plants as an example to establish a model.

DD-RCM supplies a clear path for the whole fault management system. It can describe the whole process of benefit implement. In Fig. 1, all the useful information is fed back to establish repair plan and is recorded. We can adjust important parameters dynamically through feedback, according to changes in stratagem, operations and environment etc.

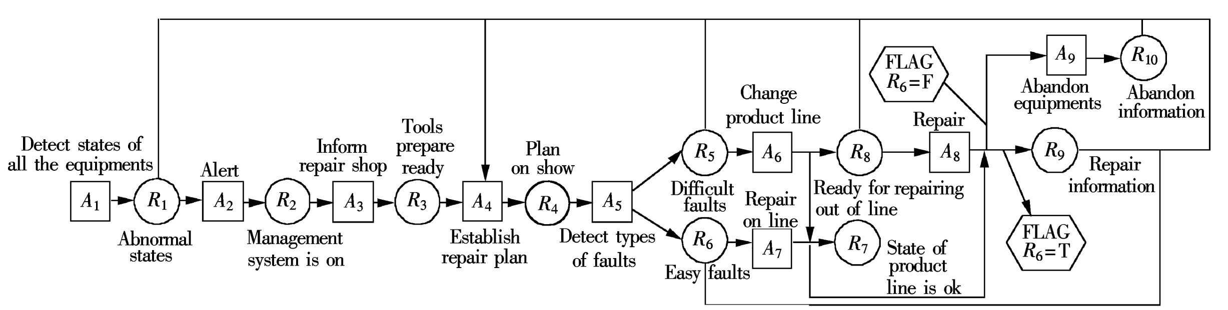

Fig. 1 DD-RCM of fault management system

Received 2006-03-09.

Biographies: Chen Lijuan (1982—), female, graduate; Xia Anbang(corresponding author), male, professor, abxia2004@yahoo.com.cn.

2 Temporal Petri Net and Its Description for DD-RCM

Temporal Petri net is a class of Petri net. Petri net is a pair TPN = (PN, f), where PN is an ordinary Petri net as defined in Ref. [2], and f is a mapping.

There are four elements in the result-chain model. Result and activity can be marked as changing transitions. Influence is the state after activity or transitions which can be represented as place. Supposing can be denoted as one of the restricted conditions for resources. Then, the temporal Petri net model transformed from DD-RCM is obtained, shown in Fig. 2.

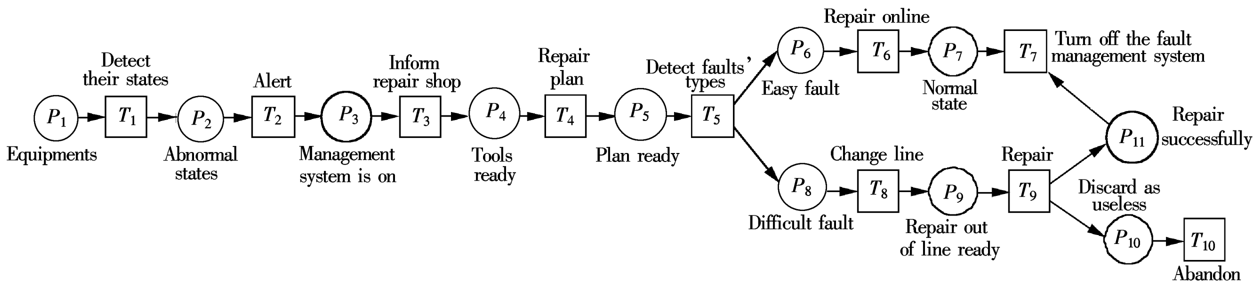

Fig. 2 Temporal Petri net model transformed from DD-RCM

If the description process is complex, the use of the temporal Petri net can reduce the complexity immensely. The temporal Petri net is a class of Petri net which can represent indirect resource flows and control information.

3 Process Reengineering Based on Petri Net

3.1 Reengineering algorithm based on incidence matrix of Petri net

There is a detailed introduction to the basic concepts of the incidence matrix in Ref. [3]. Here, only the reengineering algorithm based on the incidence matrix is proposed in the following:

① Search out all the source transitions, collection transitions, source places, and collection places.

② Search out all the transitions with synchronizations to form a table.

③ Search out all the transitions with conflictions to form a table.

④ The source transitions, collection transitions, source places, and collection places are not considered in the following four sub-steps. If there is not only one element that can be chosen, choose them to form sub-Petri nets separately.

a) Begin with an element $a_{ij}(a_{ij} = -1)$ from the incidence matrix, which is not an element of any subnet.

b) Choose another element $a_{ik}(a_{ik} = 1)$.

c) Choose an element $a_{mk}(a_{mk} = -1)$. If $m = i$, or $k = j$, turn to ⑤. Otherwise, continue.

d) Choose an element $a_{mr}(a_{mr} = 1)$. If $m = i$, or $r = j$, turn to ⑤. Otherwise, back to c).

⑤ If the row and column formed by all the chosen elements have only one "−1" or "1", a subnet is formed. Then go on to the next step. Otherwise, quit and turn back to a).

⑥ List this subnet and the correlation elements. If all the start points are included, continue to the next step. Otherwise, back to a).

⑦ Put the corresponding elements of source and collection transitions into the subnets with some sharing places.

⑧ Put the corresponding elements of source and collection places into the subnets with some sharing transitions.

⑨ All the non-zero elements not included in any subnets must be distributed to the subnet with "−1" having the same transition as this element.

3.2 Algorithm applications

First, the proposed algorithm is employed in the Petri net model in Fig. 2 to obtain the incidence matrix of the whole net and subnets. In Fig. 2, P_1 is a source place; t_1 is a transition between input place P_1 and output place P_2. P_2 is a place having input transition t_1 and output transition t_2. t_7 and t_{10} are collection transitions. The rest may be deduced by analogy. The incidence matrix of Fig. 2 is obtained, shown in Eq. (1).

$$
\begin{array}{c} \\ P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \\ P_7 \\ P_8 \\ P_9 \\ P_{10} \\ P_{11} \end{array}
\begin{array}{c}
\begin{array}{cccccccccc} t_1 & t_2 & t_3 & t_4 & t_5 & t_6 & t_7 & t_8 & t_9 & t_{10} \end{array} \\
\begin{bmatrix}
-1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \\
0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0
\end{bmatrix}
\end{array}
\qquad (1)
$$

According to steps ④ to ⑨ of the reengineering algorithm, this net can be divided into seven subnets.

Their incidence matrices are given as follows:

$$\begin{array}{c} \\ p_4 \\ p_5 \\ p_6 \end{array}\!\!\begin{array}{c} t_4 \quad t_5 \\ \begin{bmatrix} -1 & 0 \\ 1 & -1 \\ 0 & 1 \end{bmatrix} \end{array}, \begin{array}{c} \\ p_4 \\ p_5 \\ p_8 \end{array}\!\!\begin{array}{c} t_4 \quad t_5 \\ \begin{bmatrix} -1 & 0 \\ 1 & -1 \\ 0 & +1 \end{bmatrix} \end{array}, \begin{array}{c} \\ p_8 \\ p_9 \\ p_{10} \end{array}\!\!\begin{array}{c} t_8 \quad t_9 \\ \begin{bmatrix} -1 & 0 \\ +1 & -1 \\ 0 & +1 \end{bmatrix} \end{array}, \begin{array}{c} \\ p_8 \\ p_9 \\ p_{11} \end{array}\!\!\begin{array}{c} t_8 \quad t_9 \\ \begin{bmatrix} -1 & 0 \\ +1 & -1 \\ 0 & +1 \end{bmatrix} \end{array}$$

$$\begin{array}{c} \\ p_6 \\ p_7 \end{array}\!\!\begin{array}{c} t_5 \quad t_6 \quad t_7 \\ \begin{bmatrix} +1 & -1 & 0 \\ 0 & +1 & -1 \end{bmatrix} \end{array}, \begin{array}{c} \\ p_9 \\ p_{11} \end{array}\!\!\begin{array}{c} t_7 \quad t_8 \quad t_9 \\ \begin{bmatrix} 0 & +1 & -1 \\ -1 & 0 & +1 \end{bmatrix} \end{array}, \begin{array}{c} \\ p_9 \\ p_{11} \end{array}\!\!\begin{array}{c} t_5 \quad t_8 \quad t_9 \\ \begin{bmatrix} +1 & -1 & 0 \\ 0 & +1 & -1 \end{bmatrix} \end{array} \tag{2}$$

Eq. (2) indicates that the fault management system is composed of seven independent activities. The conflict and synchronization relationships can be found in subnets.

In Eq. (2), if t_7 fires in the sixth subnet, it also needs P_{11}. However, if t_9 fires in the seventh subnet, it needs P_{11}. So, subnets six and seven have a conflict on resource P_{11}. Subnets three, four and six have a conflict on resource P_9 with subnet seven. If t_7 fires, P_7 is needed in the fifth net and P_{11} is needed in the sixth subnet. So, subnets five and six have a synchronization relationship on resources. The rest are deduced by analogy.

Conflict and synchronization tables are given in Tabs. 1 and 2.

Tab. 1 Conflict relationship

Places	Transitions	The 3rd subnet	The 4th subnet	The 6th subnet	The 7th subnet
P_{11}	t_7			×	
	t_9				×
P_9	t_8				×
	t_9	×	×	×	

Tab. 2 Synchronization relationship

Transitions	Places	The 3rd subnet	The 4th subnet	The 5th subnet	The 6th subnet	The 7th subnet
t_7	P_7			×		
	P_{11}				×	
t_8	P_8	×	×			
	P_9					×
t_9	P_9	×	×		×	
	P_{11}					×

All the activities in the fault management system can be properly planned to optimize resource configuration according to the analysis of relationships among subnets.

4 Reductions

As we know, state explosion is always one of the important problems in Petri-net-based analysis. And when reduced, relationships among all activities must be considered[4-8]. A method of reducing the places and transitions of Petri nets, under the condition of holding functionality, is presented to combat the above problem.

The functionality will be described by temporal logic formulae. These formulae are based on the same variant of a prepositional temporal logic of linear time as adopted in TPN[6-7]. The following formulae specify the functionality of the net shown in Fig. 2.

$$F_1 = \Box(P_1 \Rightarrow \Diamond P_6 \cdot P_8) \tag{3}$$

$$F_2 = \Box((P_6 \cdot P_8) \Rightarrow \Diamond P_7 \cdot P_{10} \cdot P_{11}) \tag{4}$$

The reduced net is shown in Fig. 3 by using reduction rules[8]. And in the reduction, all the activities properly avoid resource competition according to the conflict table.

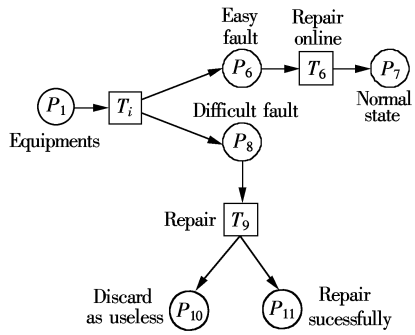

Fig. 3 Reduced Petri net of Fig. 2

The reduced net has only six places corresponding to eleven places in the original Petri net. The number of transitions is reduced to three from ten. In order to meet the definition of reduction rules[9-10], the functionality of Fig. 3 must be proved to be the same as Fig. 2.

Proof Three attributes PR_1, PR_2 and PR_3[11] of temporal Petri nets are introduced first:

$$\langle M, \alpha \rangle \mid = f + \circ f \quad \text{yield} \quad \langle M, \alpha \rangle \mid = \Diamond f$$

$$\langle M, \alpha \rangle \mid = (f_1 \Rightarrow \Diamond f_2) \cdot \Box(f_2 \Rightarrow \Diamond f_3)$$

$$\text{yield} \quad \langle M, \alpha \rangle \mid = \Box(f_1 \Rightarrow \Diamond f_3)$$

$$\langle M, \alpha \rangle \mid = \Box(t_{i\mathrm{fir}} \Rightarrow \Diamond t_i)$$

In order to analyze the net behavior, assuming that there is initially a token in place P_1, we have

$$\langle M, \alpha \rangle \mid = \Box(P_1 \Rightarrow t_{i\mathrm{fir}}) \tag{5}$$

Combining Eq. (5) and PR_3 yields

$$\langle M, \alpha \rangle \mid = \Box(t_i \Rightarrow \circ (P_6 \cdot P_8)) \tag{6}$$

Combining Eqs. (5) and (6) and PR_1 and PR_2, we have

$$\langle M, \alpha \rangle \mid = \Box(P_1 \Rightarrow \Diamond(P_6 \cdot P_8)) \tag{7}$$

$$\langle M, \alpha \rangle \mid = \Box((P_6 \cdot P_8) \Rightarrow (t_6 \cdot t_9)) \tag{8}$$

Eq. (8) and PR_3 yield

$$\langle M, \alpha \rangle \mid = ((t_6 \cdot t_9) \Rightarrow \circ (P_7 \cdot P_{10} \cdot P_{11})) \tag{9}$$

Eqs. (8) and (9) and PR_1 yield

$$\langle M, \alpha \rangle \mid = \Box((P_6 \cdot P_8) \Rightarrow \Diamond P_7 \cdot P_{10} \cdot P_{11}) \tag{10}$$

The net shown in Fig. 3 retains the external functional behavior as indicated by Eqs. (7) and (10) of the original net (shown as Eqs. (3) and (4)). Therefore, the Petri net of Fig. 3 is the reduced net of Fig. 2.

The model analysis will be much simpler and the resources apply maximized after reduction. The fault management is more effective.

5 Conclusions

This paper proposes a new method of optimizing the fault management system in power plants by using DD-RCM, temporal Petri net, the reengineering algorithm of incidence matrix and reduction rules. The contributions are given as follows: ① Give a DD-RCM model to grasp the whole benefit implement paths. ② Transform DD-RCM into the temporal Petri net model to describe the relationships among all the activities. ③ Employ the reengineering algorithm of incidence matrix to form the conflict and synchronization tables. ④ Use reduction rules for the Petri net, and avoid resources completion according to the above tables, then, obtain a reduced net.

References

[1] Mi X Z, Zhao W H, Huang X W. Application of DD-RCM technology in complicated product digital development implementation [A]. In: *Proceedings of the 8th International Conference on Computer Supported Cooperative Work in Design*[C]. Houston, 2004. 62 − 66.

[2] Richard Z. Petri net model functional abstractions and reduction techniques: applications to the design of automated manufacturing systems [J]. *IEEE Transactions on Industrial Electronics*, 2005, **52**(2): 595 − 609.

[3] Marco M, Andreas O. A Petri-net based approach to performance management of collaborative business processes [A]. In: *Proceedings of the 16th International Workshop on Database and Expert Systems Applications* [C]. New York, 2005. 987 − 991.

[4] Kun L J. Process analysis of multi-robot interconnection system using timed Petri nets reduction [J]. *IEEE Transactions on Systems, Man & Cybernetics*, 1997, **1**(1): 251 − 256.

[5] Demongodin I, Koussoulas N T. Differential Petri net models for industrial automation and supervisory control [J]. *IEEE Transactions on Systems, Man & Cybernetics*, 2006, **36**(4): 543 − 553.

[6] Stephen J, Yang H. Fuzzy rule base systems verification using high-level Petri nets [J]. *IEEE Transactions on Knowledge and Data Engineering*, 2003, **15**(2): 457 − 473.

[7] Tsinarakis G J, Tsourveloudis N C, Valavanis K P. Modeling, analysis, synthesis and performance evaluation of multi-operational production systems with hybrid timed Petri nets [J]. *IEEE Transactions on Automation Science and Engineering*, 2006, **3**(1): 29 − 46.

[8] Robert H S. Reduction rules for time Petri nets [J]. *Acta Information*, 1996, **33**(6): 687 − 706.

[9] Llorens M, Oliver J. Structural and dynamic changes in concurrent system: reconfigurable Petri nets [J]. *IEEE Transactions on Computers*, 2004, **53**(9): 1147 − 1158.

[10] Jeng M. Comments on “timed Petri nets in modeling and analysis of cluster tools” [J]. *IEEE Transactions on Automation Sciences and Engineering*, 2005, **2**(1): 92 − 93.

[11] Bouyekhf R, Moudni A E. On the analysis of some structural properties of Petri nets [J]. *IEEE Transactions on Systems, Man & Cybernetics*, 2005, **35**(6): 784 − 794.

基于 DD-RCM 和时序 Petri 网的电厂故障管理系统优化

陈丽娟[1]　杨　欣[2]　夏安邦[1]

([1] 东南大学电气工程学院,南京 210096)
([2]东南大学自动化学院,南京 210096)

摘要:基于 DD-RCM 和时序 Petri 网提出一种优化电厂故障管理系统方法. 用 DD-RCM 方法建立电厂故障管理系统的模型,并根据对应规则将电厂故障管理系统的 DD-RCM 模型转化为相应的时序 Petri 网模型;再根据 Petri 网模型确定出各种活动之间的选择、冲突、同步和并发的关系. 进一步用关联矩阵的重组算法对各种关系加以描述. 然后用简化规则结合冲突表和同步表对 Petri 网模型进行简化,实现对电厂故障管理系统的优化. 最后基于时序 Petri 网的时态逻辑证明了简化 Petri 网的功能性.

关键词:DD-RCM;时序 Petri 网;关联矩阵;简化规则

中图分类号:TP301