



DOI:10.13364/j.issn.1672-6510.20180105

基于图论与灰色关联理论的挤出机故障诊断方法

王玉鹏, 赵继民, 杨世凤, 仇明辉
(天津科技大学电子信息与自动化学院, 天津 300222)

摘要: 针对现有故障诊断方法在面对复杂结构设备时的故障定位困难问题,以挤出机为例,将图论算法与灰色关联理论相结合,建立了故障传播网络模型.该模型通过对故障传播有向图进行矩阵化、层次化处理,结合故障定位算法确定故障的位置.最后,以虚拟仪器为开发平台,结合工业以太网等技术开发挤出机故障诊断系统,实现了对挤出机运行状态的实时监控和对故障信息及相关知识库的管理.运行结果表明:系统能够快速有效的判断挤出机的故障位置与起因,缩短维修时间,提高生产效率.

关键词: 挤出机; 图论算法; 故障诊断系统; 灰色关联理论; LabVIEW

中图分类号: TP273 **文献标志码:** A **文章编号:** 1672-6510(2019)01-0070-06

Extruder Fault Diagnosis Based on Graph Theory and Grey Relational Theory

WANG Yupeng, ZHAO Jimin, YANG Shifeng, QIU Minghui

(College of Electronic Information and Automation, Tianjin University of Science & Technology, Tianjin 300222, China)

Abstract: With the current fault diagnosis method, if the structure of equipment is complex, the location of the fault is difficult to determine. Taking extruder as an example, combining graph theory algorithm with gray relational theory, a fault propagation network model was established. Through using matrix and hierarchical methods to deal with the fault propagation directed graph, combining with fault location algorithm, the location of the fault can be determined. Using a virtual instrument as a development platform, together with industrial Ethernet technology, a new extruder fault diagnosis system was developed. The system can realize the real-time monitoring of the running of the extruder and the management of information about faults and the relevant knowledge base. The results show that the system can quickly and effectively determine the location and cause of the extruder fault, shorten maintenance time and improve production efficiency.

Key words: extruder; graph theory; fault diagnosis system; grey relation theory; LabVIEW

挤出机作为双复合轮胎胎面挤出联动生产线的重要设备,其故障诊断系统是不可缺少的.图形诊断法在故障诊断中使用较多,例如故障树诊断法,虽然被广泛应用于机械行业中^[1],但某些复杂设备的结构很难用故障树表达.为此,研究人员提出利用诸如马尔可夫状态图、主逻辑图、事件树等图形实现对部件之间故障传播机理的解释^[2-3],然而这些算法都还仅仅局限于树形结构,在处理复杂结构时鲁棒性较差.

Nakano等^[4]于1974年首次将图论诊断法应用到

故障诊断中,但仅针对特定的问题. Kokawa等^[5]于1983年将故障传播概率和传播时间的权重添加到每对节点之间的边上,并且去除了图中的符号关系.这种方法虽然提高了故障诊断的成功率,但也仅局限于单一故障,对于多个故障的诊断能力比较薄弱.

本文采用基于灰色关联理论^[6]的故障模式及影响分析(FMEA)方法对传统故障定位算法进行改进,结合挤出机系统的故障特点,引入权重因素,将“全局风险影响度”作为挤出机系统故障定位的主要依

收稿日期: 2018-04-04; 修回日期: 2018-07-09

基金项目: 天津市应用基础与前沿技术研究计划资助项目(14JCZDJC39000)

作者简介: 王玉鹏(1991—),男,河北人,硕士研究生;通信作者: 杨世凤,教授, yangsf@tust.edu.cn

据,提高了故障定位的准确率. 采用 LabVIEW 平台进行软件开发,结合工业以太网技术,可以实时监控设备运行状态,显示故障信息.

1 故障诊断原理

1.1 故障传播有向图

将图论算法用于故障诊断,需要根据具体系统的组成结构及工作原理绘制故障传播有向图^[7],以表达故障之间存在的传播关系.

根据挤出机系统的集合性,系统是由两个或两个以上有相同属性且相互区别的要素组成的,可表示为

$$V = \{v_i | v_i \in V\} \quad (i=1,2,\dots,n;n \geq 2) \quad (1)$$

其中,故障传播有向图中的元素 v_i 代表挤出机系统中的一个故障. 根据挤出机系统的相关性, $v_i R v_j$ 表示故障之间的传播关系. 则这个故障在故障传播有向图上可以表示为

$$S = \{V | R\} \quad (2)$$

由于故障传播有向图中的每一个节点都代表挤出机系统的一个部件或子设备,并且系统的故障传播特性是固定不变的,因此挤出机系统的故障传播有向图也是唯一的.

1.2 矩阵化处理

将故障模型转化为故障传播有向图后,可通过矩阵化处理的方式将有向图转化为矩阵形式. 若有向图中存在 n 个节点,那么故障传播有向图可以用 $n \times n$ 阶矩阵 A 表示:

$$A = \begin{cases} 1 & \text{从节点 } i \text{ 指向节点 } j \text{ 的边} \\ 0 & \text{点 } i, j \text{ 之间没有边相连} \end{cases} \quad (3)$$

对于不存在自回环的节点,它的对角线元素值为0.

图 1 所示故障传播有向图对应的矩阵为

$$A = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \dots & a_{nn} \end{pmatrix}$$

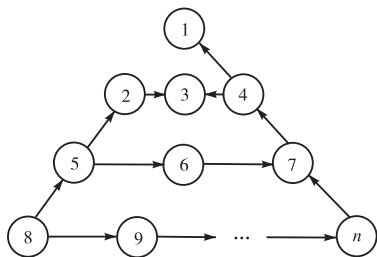


图 1 故障传播有向图示例

Fig. 1 An example of a failure propagation diagram

在实际情况中,一个故障往往会引发其他故障. 通常把故障 v_i 可以直接或者间接导致故障 v_j 发生的情况称之为 v_i 可达 v_j . 用于描述故障传播有向图中故障之间可达关系的矩阵,就可以称之为可达矩阵,用 M 表示,且其中元素 M_{ij} 需满足

$$M = \begin{cases} 1 & v_i \text{ 可达 } v_j \\ 0 & v_i \text{ 不可达 } v_j \end{cases} \quad (4)$$

在这里假设每一个故障都能到达其本身,因此连接矩阵与可达矩阵之间存在如下关系:

$$M = I \cup A \cup A^2 \dots \cup A^{n-1} \quad (5)$$

式(5)中 I 为 n 阶单位矩阵,则图 1 的可达矩阵 M 可以表示为

$$M = \begin{pmatrix} b_{11} & \dots & b_{1n} \\ \vdots & \ddots & \vdots \\ b_{n1} & \dots & b_{nn} \end{pmatrix} \quad (6)$$

通过创建连接矩阵和可达矩阵,实现了对于任意故障传播关系的描述,从不同的角度解释了传播有向图之间复杂的关系.

1.3 层次化处理

分层重构(level structuring)对于故障源的辨识有着十分重要的意义,它在一定程度上减少了原始故障传播有向图的复杂程度,降低了故障诊断求解过程的难度. 本文使用基于可达性的分层重构方法,故障集被分成若干个层次,每一个层次内的故障都只能由本层向下一层传播,这种层次关系表明了系统的故障传播结构. 对图 1 中的故障传播有向图进行重构,重构之后的有向图见图 2.

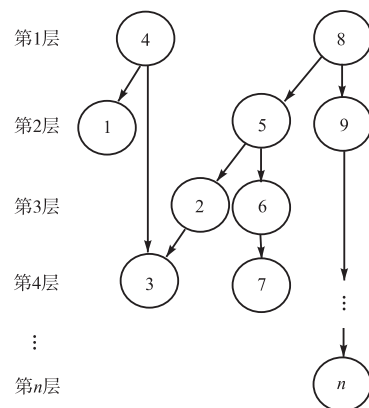


图 2 具有层次结构的故障传播有向图

Fig. 2 Fault propagation directed graph with hierarchical structure

1.4 故障定位算法

为便于计算,将故障传播有向图的分层用矩阵的

形式表述,进行故障定位.区别于传统的故障定位算法,故障模式及影响分析(FMEA)是一种将故障或隐患在系统建立之初就加以排查的方法,通过发生率(O)、严酷度(S)、检测度(D)三个元素确定故障位置.但这种方法过度依赖专家经验知识,针对这种不足,将其与善于处理不确定性问题的灰色关联理论相结合,以提高故障定位精度.其故障定位计算过程见图3.

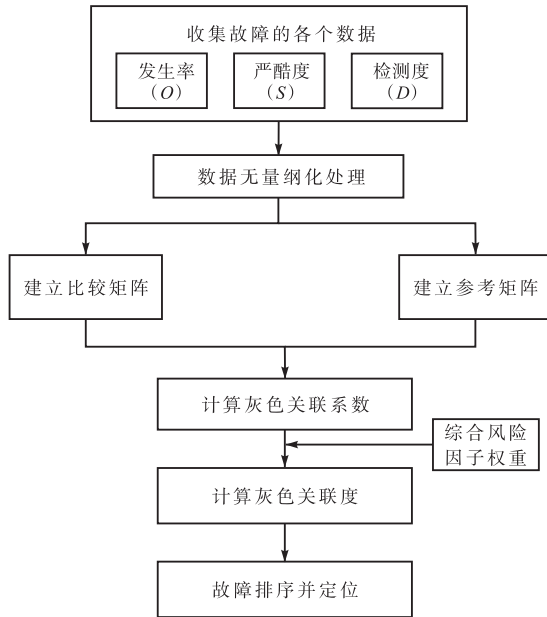


图3 基于灰色关联理论的故障定位计算过程

Fig.3 Flowchart of fault location calculation based on grey correlation theory

首先,依据图论算法计算发生各故障的概率(发生率),并依据扩展 FMEA 表格中的故障的严酷度与检测度,计算出初始的故障数据矩阵,对于初始故障数据矩阵进行无量纲化处理.本文使用向量归一化的方法实现矩阵的无量纲化处理,公式为

$$X'_i(k) = \frac{X_i(k)}{\sqrt{\sum_{i=0}^m X_i^2(k)}} \quad (0 \leq i \leq m, 1 \leq k \leq n) \quad (7)$$

式中: $X_i(k)$ 表示原始矩阵中有量纲的数据,且满足 $0 \leq X_i(k) \leq 1$,方向保持不变; $X'_i(k)$ 表示处理后矩阵中无量纲的数据.

然后,建立比较矩阵和参考矩阵.假设挤出机系统存在 m 个故障,记为 $X_1, X_2, \dots, X_i, \dots, X_m$, 其中 X_i 表示第 i 个故障,并且每个故障都存在发生率(O)、严酷度(S)、检测度(D)三个变量,那么可以将第 i 个故障的数据表示为 $X_i = \{X_i(1), X_i(2), X_i(3)\}$, 其中 $X_i(k) (k=1,2,3)$ 表示 FMEA 的三个数据 O、S、D. 因

此,根据上述建立比较矩阵的方法,就能够得到反映 m 个故障的比较矩阵

$$P = \{X_i(k)\} = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_m \end{bmatrix} = \begin{bmatrix} X_1(1) & X_1(2) & X_1(3) \\ X_2(1) & X_2(2) & X_2(3) \\ \vdots & \vdots & \vdots \\ X_m(1) & X_m(2) & X_m(3) \end{bmatrix} \quad (8)$$

对于故障风险的排序是参照某一个参考基准进行的.根据实际情况,将最优值组成的矩阵作为参考基准,并建立对应的参考矩阵

$$P_0 = \{X_0(k)\} = \begin{bmatrix} O_1 & S_1 & D_1 \\ O_2 & S_2 & D_2 \\ \vdots & \vdots & \vdots \\ O_m & S_m & D_m \end{bmatrix} \quad (9)$$

选取零矩阵作为参考矩阵,即

$$P_0 = \{X_0(k)\} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \vdots & \vdots & \vdots \\ 0 & 0 & 0 \end{bmatrix} \quad (10)$$

随后,计算比较矩阵与参考矩阵在某一指标点 k 处的灰色关联系数,计算公式为

$$\xi(X_0(k), X_i(k)) = \frac{j_{\min} i_{\min} |X_0(k) - X_i(k)| + \gamma j_{\max} i_{\max} |X_0(k) - X_i(k)|}{|X_0(k) - X_i(k)| + \gamma j_{\max} i_{\max} |X_0(k) - X_i(k)|} \quad (11)$$

式中; γ 表示分辨系数,一般取 0.5; $X_0(k)$ 和 $X_i(k)$ 表示参考矩阵和比较矩阵在指标点 k 处的值.

最后,计算灰色关联度,并进行故障定位,计算公式为

$$\delta(X_0(k), X_i(k)) = \sum_k^3 L_k \{\xi(X_0(k), X_i(k))\} \quad (12)$$

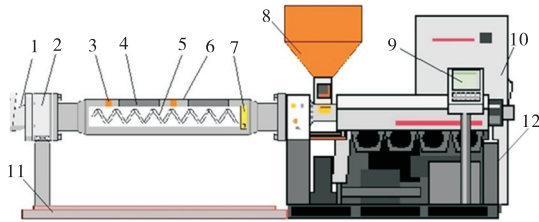
式中: $\sum_{k=1}^3 L_k = 1$, L_k 是专家根据实际情况提前确定的“综合风险因子权重”.灰色关联度表示的是故障与正常状态之间的关联程度,灰色关联度与故障发生的概率成反比,关联度越高,发生故障的可能性越低.因此,根据灰色关联度对故障升序排列,进行故障定位.

2 挤出机故障诊断系统

2.1 硬件结构

挤出机由机头、机架、机筒、螺杆、加热冷却系统、PLC 控制器等组成^[8-9].以挤出机作为故障诊断对象搭建故障诊断系统,硬件主要由挤出机、检测设

备和现场工控机组成, 见图 4.



- 1. 机头; 2. 温度传感器; 3. 排气孔; 4. 加热冷却系统; 5. 螺杆;
- 6. 机筒; 7. 超声波位移传感器; 8. 加料口; 9. 工控机; 10. PLC;
- 11. 机架; 12. 减速箱

图 4 硬件结构示意图

Fig. 4 Hardware structure diagram

检测设备包括温度传感器与超声波位移传感器等. 由传感器测得挤出机胶料温度、电机转速、机头压力与螺杆位移量等原始数据, 处理后传给 PLC^[10], 再通过工业以太网传送到工控机.

工控机采用研华 IPPC-6152A, 实时监控与记录设备运行状态. 软件系统对 PLC 传来的数据进行处理, 从而判断故障原因, 并显示故障诊断结果.

2.2 软件开发

使用 LabVIEW 平台进行软件开发^[11], 将改进后的图论算法编写入程序中, 采用 Access 数据库作为诊断规则、故障数据及诊断策略的存储介质, 通过对数据检索链路的分析, 进而获得故障原因和相对应的诊断策略. 程序框图见图 5.

针对不同的故障, 程序通过从数据库中选择不同的故障模式和故障规则, 实现对故障原因的定位. 如果数据库中不存在需要使用的故障模式, 还可以通过手动添加故障模式的方式更新数据库.

故障诊断人机界面如图 6 所示, 可实时显示故障

现象、潜在故障原因及对应的灰色关联度. 此外, 可给出针对特定故障常用的解决办法, 为现场检修人员提供参考, 方便用户快速排除故障. 操作人员还可通过人机界面手动向数据库中实时添加故障诊断规则, 而不必直接改写数据库, 简化了操作流程, 从而不断更新故障诊断规则和提高故障诊断准确率.

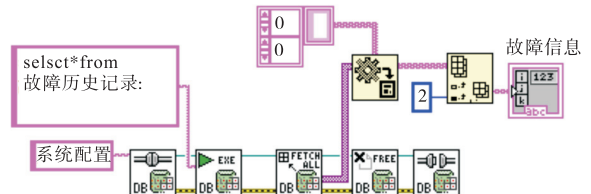


图 5 LabVIEW 程序框图

Fig. 5 LabVIEW block diagram



图 6 故障诊断人机界面

Fig. 6 Fault diagnosis interface

3 应用实例

以挤出机不下料这一故障现象为例, 进行故障原因的定位. 表 1 列出了与此问题相关的故障原因及其对应的扩展 FMEA 信息^[12].

表 1 挤出机系统扩展 FMEA 表

Tab. 1 Extending FMEA form of the extruder system

代号	故障名称	严酷度	检测度	原因项	原因项发生率
1	挤出机不下料	10	4	[2, 3, 4]	[0.4, 0.2, 0.4]
2	加料口阻塞	10	4	[5, 6, 7]	[0.4, 0.3, 0.3]
3	加料口没有料	4	1	[3]	[1]
4	倒流	10	3	[8, 9]	[0.6, 0.4]
5	加料口溶胶粘结	10	1	[10]	[1]
6	物料颗粒较大	3	1		
7	挡板未拉开	4	1		
8	螺杆有缝隙	7	4	[11, 12]	[0.7, 0.3]
9	螺杆过滑	7	4	[13]	[1]
10	机筒温度过高	8	2		
11	过胶垫磨损	9	4		
12	螺杆和机筒过磨损	9	4		
13	胶料油品成分过多	8	2		

参照表 1 中的相关数据, 绘制挤出机的初始故障传播有向图, 见图 7. 故障 1 是由故障 2、故障 3、故障 4 引起的, 故障 2 是由故障 5、故障 6、故障 7 引起的, 故障 5 是由故障 10 引起的. 同理, 故障 4 是由故障 8、故障 9 引起的, 故障 8 是由故障 11、故障 12 引起的, 故障 9 是由故障 13 引起的.

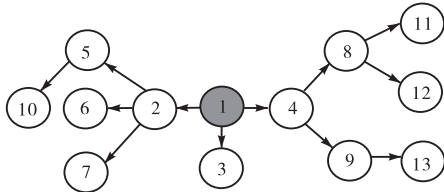


图 7 初始故障传播有向图

Fig. 7 Initial fault propagation digraph

根据式 (3) 求得初始故障传播有向图的连接矩阵

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

根据式 (4) 求得初始故障传播有向图的可达矩阵

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

根据求得的可达矩阵, 对初始故障传播有向图进行层次化处理, 重构该有向图的层次, 见图 8. 经层次化处理后, 可以很清楚地看出故障之间的传播关

系: 已发生的故障 1 是第一层, 故障 2、故障 3、故障 4 是第二层, 故障 5、故障 6、故障 7、故障 8、故障 9 是第三层, 故障 10、故障 11、故障 12、故障 13 是第四层.

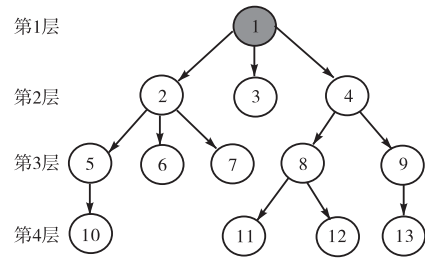


图 8 故障层次传播有向图

Fig. 8 Fault propagation digraph

将层次化处理后的故障传播有向图转化为矩阵形式. 根据灰色关联理论, 对故障的发生率、严酷度、检测度进行无量纲化处理, 根据式 (7) 得到无量纲的 FMEA 表, 见表 2.

表 2 无量纲 FMEA 表

Tab. 2 Dimensionless FMEA table

代号	故障名称	发生率	严酷度	检测度
1	挤出机不下料	0.156	0.347	0.370
2	加料口阻塞	0.052	0.347	0.370
3	加料口没有料	0.052	0.139	0.092
4	倒流	0.052	0.347	0.277
5	加料口溶胶粘结	0.052	0.347	0.092
6	物料颗粒较大	0.104	0.104	0.092
7	挡板未拉开	0.052	0.139	0.092
8	螺杆有缝隙	0.313	0.243	0.370
9	螺杆过滑	0.261	0.243	0.370
10	机筒温度过高	0.521	0.278	0.185
11	过胶垫磨损	0.417	0.312	0.370
12	螺杆与机筒过磨损	0.261	0.312	0.370
13	胶料油品成分过多	0.521	0.278	0.185

将表 2 中的数据代入式 (8) 得到比较矩阵, 再将得到的数据代入式 (9) 得到参考矩阵, 随后根据式 (11) 计算出所有故障关于每个风险因子的灰色关联系数 $\xi(X_0(k), X_i(k))$. 将灰色关联系数以及综合风险因子权重代入式 (12) 中, 得到所有故障的灰色关联度 (表 3), 并根据从小到大的顺序进行排列. 其中综合风险因子权重为 $L_1 = 0.36, = 0.45, = 0.19$.

灰色关联度表示的是故障与正常状态之间的关联程度, 灰色关联度数值与故障发生的概率成反比, 关联度越高, 发生故障的可能性越低. 从表 3 可以看出, 故障原因按出现概率由大到小进行排序依次是: 故障 11 > 故障 10 > 故障 13 > 故障 12 > 故障 8 > 故障 9 > 故障 1 > 故障 2 > 故障 4 > 故障 5 > 故障 6 > 故

障 3 > 故障 7, 可见故障 11 为故障源的概率最大, 应优先排查。

表 3 灰色关联度表
Tab. 3 Grey relational scale

代号	故障名称	灰色关联度
11	过胶垫磨损	0.506
10	机筒温度过高	0.538
13	胶料油品成分过多	0.539
12	螺杆与机筒过磨损	0.556
8	螺杆有缝隙	0.570
9	螺杆过滑	0.589
1	挤出机不下料	0.596
2	加料口阻塞	0.686
4	倒流	0.702
5	加料口溶胶粘结	0.760
6	物料颗粒较大	0.863
3	加料口没有料	0.880
7	挡板未拉开	0.914

注: 表中数据按灰色关联度数值从小到大排序。

4 结 语

本文以挤出机作为研究对象, 将图论算法和灰色关联理论相结合, 实现了对挤出机设备控制系统的故障分析与研究。根据研究结果, 采用 LabVIEW 进行编程, 开发了轮胎挤出机故障诊断系统。故障诊断系统可以为现场技术人员解决故障提供帮助, 提高对故障原因定位的准确性, 节省故障检测时间。

参考文献:

[1] Reay K A, Andrews J D. A fault tree analysis strategy

using binary decision diagrams[J]. Reliability Engineering and System Safety, 2002, 78(1): 45-56.

[2] 曾士勇, 龚时雨. 基于主逻辑图的安全风险建模研究[J]. 中国安全科学学报, 2003, 13(10): 66-68.

[3] 谭钦文, 苗东涛, 刘建平, 等. 基于可靠性理论的事件树分析方法研究[J]. 中国安全生产科学技术, 2015, 11(6): 75-81.

[4] Nakano H, Nakanishi Y. Graph representation and diagnosis for multiunit faults[J]. IEEE Transactions on Reliability, 1974, R23(5): 320-325.

[5] Kokawa M, Miyazaki S, Shingai S. Fault location using digraph and inverse direction search with application[J]. Automatica, 1983, 19(6): 729-735.

[6] 邓聚龙. 灰色系统基本方法[M]. 武汉: 华中理工大学出版社, 1987.

[7] 陈侃, 李昌禧. 故障传播有向图的故障定位研究[J]. 自动化仪表, 2011, 32(4): 14-17.

[8] Ishibuchi H, Nii M. Numerical analysis of the learning of fuzzified neural networks from fuzzy If-then rules[J]. Fuzzy Sets and Systems, 2001, 120(3): 281-307.

[9] 张红梅. 基于 SVM 的胎面生产过程的专家故障诊断系统[D]. 武汉: 武汉理工大学, 2008.

[10] 刘振全, 王汉芝. 基于触摸屏和 PLC 的多路温度监控系统[J]. 天津科技大学学报, 2007, 22(1): 44-47.

[11] 刘乐平, 王亮. 基于 LabVIEW 的旋臂水池远程测控系统开发[J]. 测控技术, 2007, 26(7): 41-44.

[12] Wang Y H, Deng C, Wu J, et al. A corrective maintenance scheme for engineering equipment[J]. Engineering Failure Analysis, 2014, 36: 269-283.

责任编辑: 常涛