

基于不确定推理的军用软件故障诊断方法研究

王维锋, 孙敬德, 肖庆

(装甲兵工程学院 信息工程系, 北京 100072)

摘要: 针对基于构件的军用软件使用过程中故障频发, 传统规则推理的方法进行故障诊断存在不确定性问题, 提出了基于不确定规则知识库的故障诊断方法; 首先根据不确定推理逻辑, 分析传统规则推理的不确定因素, 构建不确定规则知识库; 然后基于该知识库和正向链推理策略, 计算确定性因子; 合理设置激活规则的阈值, 并对中间结论的确定性因子进行排序以消解冲突; 实验结果表明, 基于不确定推理的诊断方法与传统精确推理方法相比, 具有更高的诊断准确率。

关键词: 规则推理; 军用软件故障; 构件; 确定性因子; 专家系统; CLIPS

Study on the Method of Military Software Fault Diagnosis Based on Uncertainty Reasoning

Wang Weifeng, Sun Jingde, Xiao Qing

(Department of Information Engineering, Academy of Armored Force Engineering, Beijing 100072, China)

Abstract: This paper focuses on the frequent faults of Component-based military software during the process of the software using, the uncertainty problems of traditional methods of rule-based reasoning for software fault diagnosis, and proposes the fault diagnosis method based on uncertain rule knowledge base. First, analyze the uncertain factors of traditional rule-based reasoning according to the logic of uncertain reasoning, and build the knowledge base of uncertain rules. Then, based on this knowledge base and the forward chain of reasoning strategies, calculate the uncertainty factor, legitimately set the threshold to activate rules, and sort the certainty factors of intermediate conclusions to eliminate conflicts. Experiment results show that, the military software fault diagnosis method based on uncertain reasoning has a higher accuracy rate compared with traditional accurate reasoning method.

Keywords: rule-based reasoning; military software fault; component; expert system; certainty factor; CLIPS

0 引言

当前, 随着信息化战争的发展要求, 软件密集型装备大量装配部队, 信息化装备的可靠性问题越来越受到多方的重视^[1]。我军军用软件的开发广泛采用基于构件的体系结构模式^[2-3]。但是, 一套军用软件包含的构件数量庞大, 构件与构件间的依赖关系复杂、耦合性强等因素造成军用软件部署以及使用过程中安装配置复杂, 故障率较高。此类故障诊断和排除工作难度大, 严重影响了信息化装备的战斗力。以人工智能领域的专家系统原理为基础的软件故障诊断技术应用广泛, 可以有效地在大量故障事件中快速定位故障源^[4-6]。基于精确推理的专家系统在诊断过程中存在不确定性问题, 导致故障误诊率较高。针对不确定因素从故障的特征事实、规则知识的可信度以及推理过程三方面出发^[8], 结合不确定推理逻辑^[9-10], 提出基于不确定推理的故障诊断方法^[11]。

1 基于构件的军用软件故障诊断分析

1.1 军用软件故障分析

根据引发军用软件故障的原因所处时间的不同, 可以将军

用软件故障分为两类: 一类是软件开发过程中由设计缺陷引发的软件故障; 另一类是软件使用过程中由操作人员引入的软件故障^[13]。笔者讨论的基于构件的军用软件故障属于后者, 故障源包括构件版本安装错误、构件重复安装、服务构件未启动、配置文件错误以及运行环境配置错误等等。常见的故障征兆包括客户端程序启动失败、数据库读取异常、构件功能异常等等。

基于构件的军用软件故障的根本原因在于系统复杂性。由于基于构件的军用软件系统包含的构件数量庞大, 构件之间的依赖关系复杂, 部队的软件操作人员由于缺乏相关专业知识的, 在发现故障征兆后很难定位故障源。因此, 有必要开发能够模拟人类专家进行故障分析与诊断的工具进行故障现场排除工作。

1.2 专家系统原理

基于专家系统原理的故障诊断方法, 在医疗、汽车维修、网络管理等领域应用广泛。专家系统工作原理图如图 1 所示:

专家系统工作过程中, 在推理机的控制下, 将知识库里的规则前件与事实列表中的事实进行匹配, 匹配成功则激活该规则进行推理。推理过程中, 知识库和推理机是相互独立的, 这样利于知识库扩展和优化, 利于开发人员将工作重点放在规则知识的获取和组织上。

收稿日期:2014-05-07; 修回日期:2014-05-30。

基金项目:军队科研计划项目。

作者简介:王维锋(1969-),男,博士,教授,主要从事装备软件保障方向的研究。

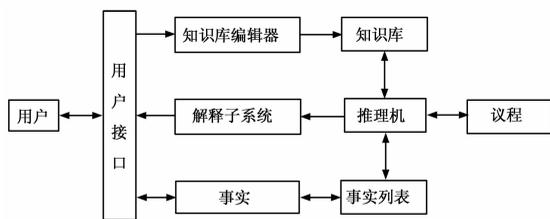


图1 专家系统原理图

1.3 诊断过程中的不确定性分析

传统的专家系统在工作过程中并不理解每条规则知识隐含的真正因果逻辑,而是默认每条产生式规则都是正确无误的,激发每条规则的事实是确定的,基于形式化推理得出结论也认为是精确的。但现实情况下,由于专家系统的用户本身掌握的军用软件知识有限,掌握的软件故障特征事实并不是完整的,即存在不确定性,采用传统的专家系统故障诊断方法并不一定得到满意的诊断结果。此外,构建专家系统知识库的知识工程师也会由于掌握知识的局限性、考虑不完全而导致精确规则的兼容性问题。

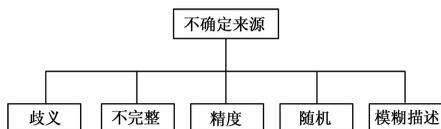


图2 规则的不确定性来源

基于精确推理的专家系统不确定性主要表现在如下3个方面^[14]:

(1) 事实的不确定性:事实或证据的不确定性直接导致规则激发的不确定性。主要反映于事实信息不完整,表述模糊。如在军用软件故障诊断时,提供的事实“发送失败”未指明什么场景下由哪个客户端构件的发送功能异常。

(2) 规则的不确定性:专家系统的知识库一般由大量的启发式的规则知识构成,这些知识来源于软件故障诊断领域专家处理问题而长期积累的经验。领域专家的经验必然存在主观概率影响、误判等因素,所以故障诊断知识库中必然存在不确定、不精确的规则。专家系统中通常采用产生式规则进行推理,其不确定性表现为规则前件的不确定性、表述模糊;规则本身的可信程度;以及规则与规则之间隐含的优先关系、兼容性问题。

(3) 推理过程的不确定性:由于事实的不确定性、规则的不确定性在推理过程中不断传播和积累,导致推理方向存在偏差,甚至得出错误的推理结论^[15]。由于推理过程是形式化的,不存在定量分析、语义分析,推理质量难以得到保障。因此在基于构件的军用软件故障诊断过程中,存在故障修复决策困难,冗余推理等问题。

2 知识库的构建

通过对基于构件的军用软件故障诊断过程中不确定性问题的分析,为提高推理的可靠程度和效率,结合不确定推理的逻辑理论,进行知识库的构建。

2.1 确定性因子

当领域专家构建知识库时对一条规则并不是非常肯定,

在规则表示过程中存在主观影响因素,这时规则可信程度可以由确定性因子CF(certainty factor)表示。部队基层的软件操作人员在使用专家系统过程中,由于自身知识和环境所限,对提供的故障特征不能完全肯定。描述故障特征事实的不确定性也可以由确定性因子表示。不确定性的程度通常由从0到1的数值表示。其中的确定性因子 $CF = 1$ 时表示该专家系统认为该条事实或者规则是非常肯定的、真实的, $CF = 1$ 的特征事实以及规则可以选自基于精确推理的知识库;而 $CF = 0$ 表示该专家系统对该事实或规则为真是非常不确定的^[16]。

例如规则R:

IF<XX故障特征> THEN<XX构件未安装>(CF=0.8),表示由事实“XX故障特征”推导出“XX构件未安装”的结论规则R的可信程度为0.8。

对于事实F:

“XX端接收文件失败”(CF=0.5)

表示用户认为“XX端接收文件失败”这一事实有一半的可能性为真。

2.2 不确定规则的表示

通过收集工业部门提供的软件使用手册等软件资源以及到数字化部队调研获得的故障数据,建立基于构件的军用软件故障案例库。在分析基于构件的军用软件故障机理和故障案例的基础上,建立故障特征集以便于特征事实的表示以及规则知识库的建立。故障特征集用E表示, E_i 代表E中的故障特征。

一般精确推理中的规则产生式的表示方法为:

IF < ANTECEDENT > THEN < CONSEQUENT > 其中,“ANTECEDENT”代表规则前件。规则前件一般来源于E,可以是某一个故障特征也可以是多个故障特征的组合。“CONSEQUENT”代表规则后件。可以是故障特征、构件描述、诊断结论或者是故障修复方法。而不确定规则的表示方法为:

IF < ANTECEDENT > THEN < CONSEQUENT > (CF),其中CF代表该条规则的确定性因子, $CF \in (0, 1]$, $CF = 1$ 时表示该条规则可选自精确规则知识库。

根据在诊断推理过程中不同的推理目的,规则知识分为问题构件定位规则,故障分析规则,故障修复规则。3种规则的产生式表达:

① IF<特征/特征组合> THEN<特征>(CF)、IF<特征/特征组合> THEN<候选问题构件>(CF);

② IF<构件信息/构件信息组合> THEN<目标问题构件>(CF)、IF<构件信息/构件信息组合> THEN<故障结论>(CF)、IF<特征/特征组合> THEN<故障结论>(CF);

③ IF<构件故障信息> THEN<修复操作>(CF)。

2.3 不确定规则的组织

不确定规则知识库的构建过程不仅要考虑传统精确推理中规则表达的一致性、规范性,还要考虑不同类型规则的确定性因子的设定。规则知识库的构建过程是一个反复迭代的过程,经历了从规则知识获取、表达再到规则知识的检查、更新。构建故障诊断知识库应该遵循如下原则^[17]:

1) 保持规则知识的一致性。规则知识的前提条件和结论

选取要注意保持一致的规则分辨率,避免出现规则包含规则、前提条件表述或结论表述出现互相包含的现象;避免出现二义性规则。

2) 规则知识的规范性。规范的表述约束,有利于保持规则的一致性,利于规则解释以及知识库的可扩充性。

3) 综合考虑经验知识,合理设置规则确定性因子的大小,保证推理过程中被激活规则的数量、质量。

3 基于不确定推理的军用软件故障诊断系统设计

3.1 系统总体结构设计

基于不确定推理的军用软件故障诊断系统采用 VC++ 与 CLIPS 相结合的混合编程方式实现。CLIPS (C Language Integrated Production System) 专家系统工具是基于 C 语言开发,由美国国家航天局约翰逊空间中心人工智能部在 1985 年推出的。CLIPS 良好兼容 VC++ 开发环境,系统采用直接嵌入的方式与 VC++ 结合。它的基本结构是产生式系统,采用正向推理机制^[18]。利用 CLIPS 进行规则知识的规范化表示,建立不确定规则知识库。CLIPS 专家系统工具提供了推理机外壳,可以控制规则的激发和推理过程。利用 VC++ 进行用户界面的设计与开发,弥补了 CLIPS 用户接口单一的不足。系统的总体结构图如图 3 所示。

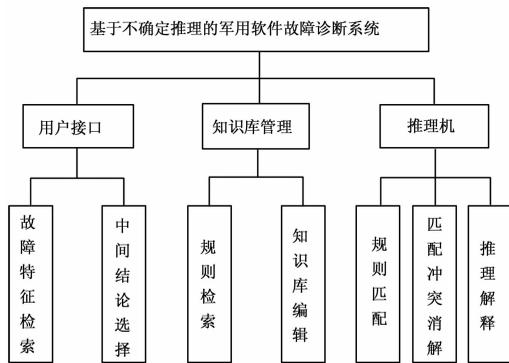


图 3 故障诊断系统结构

3.2 基于不确定规则的推理控制策略

3.2.1 正向链推理

控制策略是指在不确定推理过程中,不确定规则的选择与推理方向的控制,通过规则前件与事实的匹配决定规则的触发以及匹配过程中冲突的解决^[20]。常用的控制策略有两种:一种是基于正向链的控制策略,即从已知的软件故障特征事实出发,让其不断与规则前件匹配直到推理出故障结论,这种方法又称为数据驱动控制。另一种是目标驱动控制,又称为反向链。基于不确定推理的故障诊断采用正向链推理策略,推理过程中通过计算每次推理除的结论确定性因子和阈值的设定,对盲目推理以及匹配冲突进行控制。

3.2.2 结论的确定性因子

依据不确定规则前件的数量和逻辑关系,可以将知识库中的规则分为简单规则和复合规则。

简单规则表达方式为: IF A THEN C (CF_r), 其中 A 表示规则只有一个前件, CF_r 代表规则的确定性因子。若事实列表中存在 A' 与 A 匹配, A' 的确定性因子为 CF_f, 那么由该规则

和事实 A' 推导出来的结论 C' 的确定性因子计算方法如公式 (1):

$$CF_c = CF_f * S * CF_r \quad (1)$$

其中: S 代表规则前件 A 与事实 A' 匹配时的相似性度量值。S 的值根据 A 与 A' 的共同论域上的隶属函数的交集确定^[21]。综合考虑专家系统的用户专业知识水平以及诊断的复杂性,一般 S 取 1。

对于“与”运算的多前件的复合规则:

IF A₁ AND A₂ AND ... AND A_n THEN C (CF_r), 与规则前件匹配的事实 A₁' A₂' ... A_n', 对应的事实确定性因子为 CF_{f1}、CF_{f2} ... CF_{fn}, 结论的确定性因子计算方法如公式 (2):

$$CF_c = \min (CF_{f1} * S_1, CF_{f2} * S_2, \dots, CF_{fn} * S_n) * CF_r \quad (2)$$

其中 S_i (i=1...n) 代表 A_i' 与 A_i 的相似程度度量值。

对于“或”运算的多前件的复合规则:

IF A₁ OR A₂ OR ... OR A_n THEN C (CF_r), 作如下等价处理:

IF A₁ THEN C (CF_r), IF A₂ THEN C (CF_r) ... IF A_n THEN C (CF_r)。

在推理过程中,由一个规则 R1 推导出来的结论可能作为与另一个规则 R2 前件相匹配的事实。则该事实的确定性因子 CF_{f'} = CF_c (3), 其中 CF_c 为规则 R1 推导出来的结论的确定性因子。

3.2.3 激活不确定规则的阈值

在不确定推理过程中,由于事实以及待激活规则本身具有不确定性,为了提高推理效率,尽量避免盲目推理,有必要设置一个激活不确定规则的阈值 T。通过阈值的设置可以限制议程中激活规则的数量,保证被激活的规则对正确合理的推理过程起到促进作用。当结论的确定性因子 CF_c ≥ T 时,激活该规则,否则不激活。

根据经验,阈值 T 的值一般设置为 0.2。领域专家可以根据规则知识库的复杂程度、推理目的以及推理效率等因素进行调整。在基于构件的军用软件故障诊断推理过程中,依据规则的类型及相同类别的规则数量规模设定合理的规则激发阈值 T。

3.2.4 匹配冲突消解

推理过程中,由于事实和规则的传递性,可能出现两条不同规则推理出相同的结论,这时该规则结论的确定性因子采用公式 4 的方法计算:

$$CF_c = CF_{c1} + CF_{c2} - CF_{c1} * CF_{c2} \quad (4)$$

其中: CF_{c1}、CF_{c2} 代表两条不同规则推导出的相同结论的确定性因子, CF_{c1}、CF_{c2} 的计算方法参照公式 (1) 和公式 (2)。若有多条规则推导出相同结论, CF_c 可以参照迭代公式 (5) 运算得到:

$$CF_c = CF_c + CF_{ci} - CF_c * CF_{ci} \quad i = 1 \dots n - 1 \quad (5)$$

其中: CF_c 的初始值为 CF_{c1}, i = 1 ... n - 1, n 代表规则数量。

针对具有相同前件的规则,先将其推导出来结论的确定性因子进行排序,然后优先选择规则结论 CF (CF ≥ T) 最大的规则进行下一步推理。

3.3 不确定推理流程

根据控制策略设计的不确定推理方法、匹配冲突解决方法, 该系统的不确定推理流程如图 4 所示。

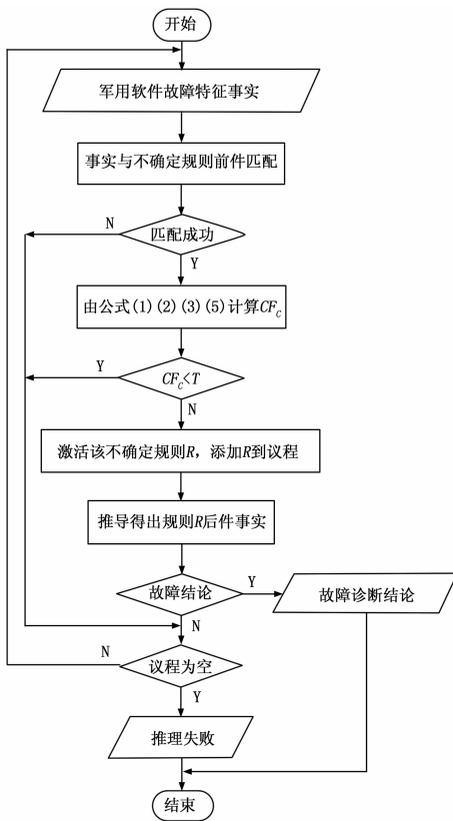


图 4 不确定推理流程图

3.4 诊断结果分析

为检验该故障诊断系统的有效性、可靠性, 选择大量发生频率高、涉及构件多、具有一定代表性的软件故障进行诊断。诊断结果显示, 诊断结论与导致军用软件故障的实际原因相符, 给出的故障修复方法可以指导用户进行故障排除工作, 基本解决了软件故障问题。经统计, 该系统对软件故障诊断的准确率在 90% 以上, 依赖的构件数量越多, 该构件发生故障的概率越大。基于不确定推理的军用软件故障诊断方法能够快速有效地完成故障诊断工作并给出排除方法, 诊断的故障类型较全面, 未出现余冗推理、循环推理, 一定程度上解决了现场条件下军用软件保障工作效率低下问题。

4 结束语

笔者依据不确定推理逻辑以及专家系统原理提出了解决基于构件的军用软件故障的方法, 并对传统推理方法进行了处理不确定问题的改进。然后结合 VC++ 开发环境和 CLIPS 专家系统工具设计实现了基于不确定推理的军用软件故障诊断系统。该系统实现了对基于构件军用软件系统故障的推理分析过程, 并给出故障诊断结论和故障修复方法。通过诊断大量故障实例, 验证了基于不确定推理方法可行性、有效性, 与传统精确推理方法相比, 该方法更加灵活、高效。基于不确定推理的军用软件故障诊断系统具有重要的军事意义

和经济价值。规则知识的自动提取以及故障预测是下一步研究的重点。

参考文献:

- [1] 李 鹏, 卢兴华, 乔 洋. 软件列装后保障活动研究 [J]. 微计算机信息, 2007, 23 (23): 32-34.
- [2] 史浩辉, 何 炜. 基于构件的指控软件复用 [J]. 计算机技术与发展, 2011, 21 (2): 159-162.
- [3] 宾晓华. 利用领域专用软件体系结构集成大型指挥系统的方法 [J]. 海军航空工程学院学报, 2008, 23 (2): 194-198.
- [4] 贺怀清, 李建伏. 一种集成式不确定推理方法研究 [J]. 计算机科学, 2012, 38 (6): 246-250.
- [5] White J, Benavides D, Thrinidad, et al. Automated diagnosis of feature model configurations [J]. The Journal of Systems and Software, 2010; 1094-1107.
- [6] 吕 琛, 栾家辉, 王立梅, 等. 故障诊断与预测—原理技术与应用 [M]. 北京: 北京航空航天大学出版社, 2012.
- [7] 卞玉涛, 李志华. 基于专家系统的故障诊断方法的研究与改进 [J]. 电子设计工程, 2013, 21 (16): 83-86.
- [8] 赖于树. 基于 FMAGDM 理论的复杂设备故障诊断不确定推理方法研究 [D]. 成都: 电子科技大学, 2011.
- [9] 郭芳芳. 不确定推理规则确定度的研究 [D]. 大连: 辽宁师范大学, 2000.
- [10] 周 青, 彭 为. 经典逻辑中不确定性及其支持度 [J]. 计算机学报, 2006, 29 (10): 1882-1888.
- [11] 邓 超, 郭茂祖, 王亚东. 一种基于产生式规则的不确定推理模板模型的研究 [J]. 计算机工程与应用, 2003, 30: 57-61.
- [12] Miao Jiang, Mohammad A. Munawar, Thomas Reidemeister. Efficient Fault Detection and Diagnosis in Complex Software Systems with Information-Theoretic Monitoring [J]. IEEE Transactions on Dependable and Secure Computing, 2011, 8 (4): 510-522.
- [13] 谢瑞生. 软件失效原因分析 [J]. 软件可靠性与评测技术, 2009, 27 (3): 13-19.
- [14] 贾维弟. 不精确推理方法研究 [D]. 哈尔滨: 哈尔滨工业大学, 2007.
- [15] 黄元亮, 李 冰. 不确定性推理中确定性的传播 [J]. 计算机仿真, 2008, 25 (7): 133-136.
- [16] 王道平, 冯振生, 何海燕. 故障诊断专家系统中不确定性问题 [J]. 计算机工程与应用, 1998, (12): 33-35.
- [17] 刘海明. 一个基于 CLIPS 的后向不确定推理系统 [J]. 电脑知识与技术, 2012, 8 (8): 1830-1835.
- [18] Joseph Giarratano, Gary Riley. 专家系统原理与编程 [M]. 北京: 机械工业出版社, 2005.
- [19] 邓海平, 何玉林, 杜 静, 等. CLIPS 嵌入 VC++ 技术的实现与应用 [J]. 计算机工程与应用, 2005, 15: 88-91.
- [20] George F. Luger. 人工智能复杂问题求解的结构和策略 [M]. 北京: 机械工业出版社, 2010.
- [21] 裴道武. 模糊推理的基本理论 [J]. 高校应用数学学报, 2012, 27 (3): 340-350.
- [22] 侯俊杰. 深入浅出 MFC [M]. 武汉: 华中科技大学出版社, 2002.
- [23] 孙 鑫, 余安萍. VC++ 深入详解 [M]. 北京: 电子工业出版社, 2006.