

摘 要

如今软件在各个领域应用非常广泛，但因为软件问题而引发的事故也频频出现。对于轨道交通领域来说，一旦发生事故，可能会引起交通瘫痪，严重甚至会威胁生命安全。通过对已完成的轨道交通联锁软件进行全生命周期的可信量化评估，可以更加全面地反映软件整体质量以及开发过程的完善性，以实现对未来软件开发流程的指导。实验室团队之前在航空航天领域和核装备领域开展了可信评估工作并应用于实际项目中，本文针对另一个安全攸关领域轨道交通进行研究。因此，本文工作围绕以下三个方面展开：

首先，建立了轨道交通联锁软件的可信量化评估模型。本文立足于四个可信属性，构建了面向全生命周期的可信量化评估指标体系和结构方程模型，据此计算度量元、子属性和属性的权重；结合可信度量模型的公式，自底向上计算子属性、属性、阶段以及软件整体的可信值；参考可信量化分级模型表，判断软件可信值满足的条件，划分软件的可信等级。

其次，基于可信量化分级模型表，设计了两种可信性分配算法。第一种算法按照度量元优先指数从高到低进行分配；第二种是基于贪心选择策略的分配算法，计算出使软件达到某一可信等级所需要的最低改进成本。

最后，开发了轨道交通联锁软件的可信量化评估工具。工具共包含三大模块：基本信息管理模块、全生命周期可信评估模块和评估结果输出模块。其中全生命周期可信评估模块分为需求、设计、编码和测试四个阶段可信评估子模块，每个子模块有可信证据输入、权重分配和可信值计算三个功能。该工具已应用于富欣智控公司研制的苏州有轨二号线联锁软件的可信评估。

关键词：可信软件，轨道交通，可信评估模型，可信性分配，量化评估工具

ABSTRACT

Nowadays, software is widely used in various fields, but accidents caused by software problems also occur frequently. For rail transit, once an accident occurs, it may cause traffic paralysis and even threaten life and safety. Through the trustworthy quantitative evaluation of the whole life cycle of the completed rail transit interlocking software, the overall quality of the software and the perfection of the development process can be reflected more comprehensively, so as to realize the guidance of the future software development process. The laboratory team has carried out trustworthiness evaluation in the field of aerospace and nuclear equipment and applied it to practical projects previously. This paper studies another rail transit area in the field of safety. Therefore, the work of this paper revolves around the following three aspects:

Firstly, a trustworthiness quantitative evaluation model of rail transit interlock software is established. Based on the four trusted attributes, this paper constructs trustworthy quantitative evaluation index system and structural equation model for the whole life cycle, and calculates the weight of the metric elements, sub-attributes and attributes accordingly; combines the formula of the trustworthiness measurement model to calculate the trustworthiness value of the sub-attributes, attributes, stages and the whole software; refers to the trustworthiness quantitative classification model table to determine the conditions satisfied by the trustworthiness value of the software, and to divide the trustworthiness level of the software.

Secondly, based on the trustworthiness quantitative classification model table, two trustworthiness allocation algorithms are designed. The first algorithm allocates

according to the priority index of metric element from high to low; the second is an allocation algorithm based on greedy selection strategy to calculate the minimum improvement cost needed to make the software reach a certain trustworthiness level.

Finally, this paper develops a trustworthiness quantitative evaluation tool for rail transit interlocking software. The tool consists of three modules: basic information management module, whole life cycle trustworthiness evaluation module and evaluation result output module. The whole life cycle trustworthiness evaluation module is divided into four stages: requirement, design, coding and testing. Each sub-module has three functions: trusted evidence input, weight allocation and trustworthiness value calculation. The tool has been applied to the trustworthiness evaluation of Suzhou Rail Line 2 interlocking software developed by Fuxin Intelligent Control Company.

Keywords: Trustworthy Software; Rail Transit; Trustworthiness Evaluation Model; Trustworthiness Allocation; Quantitative Evaluation Tool

目录

第一章 绪 论	1
1.1 研究背景与意义	1
1.2 国内外研究现状	2
1.3 研究内容	3
1.4 文章结构	4
第二章 预备知识	6
2.1 软件可信相关概念	6
2.1.1 软件可信性定义	6
2.1.2 软件属性、子属性和度量元	6
2.1.3 软件可信度量模型与分级方法	8
2.2 结构方程模型的原理	9
2.2.1 结构方程模型的概念	9
2.2.2 结构方程模型的结构	10
2.3 系统开发框架与技术	11
2.4 本章小结	12
第三章 轨道交通联锁软件可信量化评估模型	13
3.1 轨道交通联锁软件可信评估指标体系	13
3.1.1 评估指标体系建立的原则	13
3.1.2 可信评估指标体系的建立	13
3.2 轨道交通联锁软件可信评估结构方程模型	16
3.2.1 模型构建	16

3.2.2	模型设定	17
3.2.3	模型检验与拟合	19
3.3	示例说明	20
3.4	本章小结	24
第四章	软件可信性分配	25
4.1	改进优先指数	25
4.1.1	改进优先指数概念	25
4.1.2	改进优先级计算示例	27
4.2	可信性分配	28
4.2.1	可信性分配模型	29
4.2.2	可信性分配算法 I	30
4.2.3	可信性分配示例 I	33
4.2.4	可信性分配算法 II	33
4.2.5	可信性分配示例 II	37
第五章	轨道交通联锁软件可信评估工具研究	38
5.1	系统分析与设计	38
5.1.1	系统分析	38
5.1.2	系统设计	40
5.2	轨道交通联锁软件评估工具实现	41
5.2.1	基本信息管理实现	43
5.2.2	全生命周期可信评估实现	44
5.2.3	评估结果输出实现	47
5.3	轨道交通联锁软件评估工具使用	49
5.4	本章小结	53
第六章	总结与展望	57
6.1	论文总结	57
6.2	下一步工作	58
参考文献		58
附录 A	设计、编码和测试阶段的可信评估指标体系	64

插图

2.1	轨道交通联锁软件可信属性分层模型	7
3.1	轨道交通联锁软件需求阶段的结构方程模型	16
3.2	可信评估结构方程模型路径图	17
3.3	轨道交通联锁软件需求阶段的结构方程模型标准化结果	21
4.1	算法 I 流程图	31
4.2	算法 II 流程图	35
5.1	联锁软件可信评估工具用例图	40
5.2	联锁软件可信评估工具系统结构图	41
5.3	联锁软件可信评估工具活动图	42
5.4	MVC 模式体系结构图	42
5.5	工具主页	43
5.6	软件产品信息维护界面	43
5.7	软件人员信息维护界面	44
5.8	需求阶段可信证据上传界面	45
5.9	阶段可信评估层次结构模型	46
5.10	需求阶段属性正互反矩阵输入界面	47
5.11	四个阶段属性可信值图谱与堆叠条形图	48
5.12	四个阶段可信值表格界面	48
5.13	评估报告界面	49
5.14	登录界面	50

5.15 增加待评估软件	50
5.16 产品信息维护界面	51
5.17 人员信息维护界面	51
5.18 上传可信证据 Excel 文件	51
5.19 可信证据上传成功	52
5.20 属性正互反矩阵	52
5.21 子属性正互反矩阵	53
5.22 度量元正互反矩阵	54
5.23 权重显示	54
5.24 可信值计算结果	55
5.25 四个阶段子属性可信值图谱	55
5.26 可信值表格分析	56
5.27 评估报告	56
5.28 软件产品评估结束对应的产品信息	56

表格

2.1	可信量化分级模型表	10
3.1	需求阶段可信评估指标体系	15
3.2	模型适配度指标检验	20
3.3	信度检验表	20
3.4	需求阶段可信评估效度检验	22
3.5	示例适配度指标检验结果	22
3.6	需求阶段可信评估	23
4.1	属性改进优先级计算示例	28
4.2	子属性改进优先级计算示例	28
4.3	度量元改进优先级计算示例	28
4.4	可信量化分级模型表（子属性）	29
4.5	可信性分配示例	33
4.6	基于单位贡献成本的可信性分配示例	37
5.1	随机一致性指标 RI 数值	46
A.1	设计阶段可信评估指标体系	64
A.2	编码阶段可信评估指标体系	65
A.3	测试阶段可信评估指标体系	66

第一章 绪 论

本章叙述了软件可信的研究背景与意义、软件可信国内外研究现状，并介绍了本文的研究内容以及文章的目录结构 [1]。

1.1 研究背景与意义

处于信息时代的软件行业飞速发展，软件应用环境也日益复杂，对软件质量的要求也越来越高，给用户提供一个可信的软件很有必要。软件内部业务处理逻辑更加复杂使其整体规模相较之前日益庞大，相关软件人员必须投入更多精力确保软件的可信性。各种类型的软件已经渗透在人们的生活之中，网上购物软件、地图导航软件和医院挂号软件等。比如飞机地铁，一半以上的功能都是由软件控制的。还有其他关键应用领域（银行，证券交易，军事等），这些软件一旦运行发生错误，可能会造成灾难性的后果，损失巨额财产或失去生命 [2]。

软件可信的要求之一就是软件能按照用户预期正常工作，即使在输入错误或者其他异常情况发生的条件下，软件也不会崩溃 [3]。因此，需要评估软件的可信性，当软件满足可信要求之后再投入使用，可以适当地降低软件出现问题的概率。用户对于刚推出软件的可信性不满意，是因为这些软件产品的构造与保障技术有待完善，使其存在许多已知或未知的缺陷。构造和开发软件，如果想保证其是可信的，必然要选择对其可信性进行计算并加以评估，以保障运行的安全可靠，这是当下软件技术发展的重要趋势 [4]。于是，国内外相关政府、科研机构和企业制定计划，把目光聚焦于可信软件研究。例如，国家科技部“863”计划；2019 年，华为与华东师范大学签约，双方深入开展合作，支持高校高可信创新实验室的建设。

城市轨道交通的信号系统逐渐采用基于通信的列车运行控制系统 CBTC (Communication-

Based Train Control System), 联锁系统是其子系统, 为了保证列车在线路上的安全运行, 信号机、道岔和进路之间必须满足一定的制约关系, 联锁系统的作用就是保证该关系的实现 [5]。计算机联锁是地铁信号系统的安全核心, 联锁软件是实现车站信号系统功能安全的核心软件模块, 可以间接通过提高地铁的自动化程度和管理水平, 实现提高地铁整体的运营效率的目标, 并且行车指挥调度人员的工作强度明显降低 [6]。对联锁软件进行可信评估可为现有的软件开发过程提供参考和指导, 改进之前做的不好的地方, 提高软件整体的可信度, 满足城市轨道交通安全、高效运行的要求。

高可信的软件不仅降低企业的维护成本还能提升使用人员的好感度。文献 [7] 提到随着许多城市地铁建设步伐的不断加快和工作人员日益增加, 大多数人上下班首选的交通工具从公交转变成了地铁, 地铁一旦运行发生问题, 短短几分钟就可能导致交通拥堵或者瘫痪。为了保障其投入使用之后能正常稳定地运行, 有必要对这个安全有关领域的软件系统进行可信评估, 发现软件存在的问题并改进, 尽量避免运行时发生故障, 带给乘客良好的用户体验, 保证乘客生命安全。

1.2 国内外研究现状

可信软件的构造要以理论为基础, 为此不同专家对于软件可信分别选择不同的切入点展开探讨。文献 [8] 通过证据收集来动态构造软件可信指标体系并应用于北京大学软件资源库。文献 [9] 以工业领域中的金属液态检测软件为例, 运行指标树生成算法得到软件可信指标树来动态构造软件可信指标系统。文献 [10] 提到 [11] 在分析了传统可信证据收集与分类方法的不足之后, 提出了一个基于验证的可信证据模型。文献 [12] 对某个航空航天软件的可信性采用了传统的模糊综合评估模型进行评估。文献 [13] 将可信评估应用在航天领域上, 提出了基于出厂报告的可信度量模型与评估体系。文献 [14, 15] 聚焦于软件开发过程提出了一种称为可信软件方法学 TSM (Trusted Software Methodology) 的理论, 其制定了软件开发的安全原则和软件工程原则; 然后, 对软件进行可信性度量的原则是以可信规则为准,

对比软件开发方法是否与其保持一致。[16] 阿布瑞尔教授创立了可信软件开发“B方法”，并应用于航空航天、轨道交通和汽车电子等领域。文献 [17] 针对核装备领域，提出了基于可信证据的软件可信性计算模型。文献 [18] 基于软件源代码提出了一种静态可信度量体系。文献 [19] 建立了软件过程与软件产品可信结构方程模型，更关注过程实体，过程行为，过程产品。

目前在轨道交通领域，关于可信评估的研究比较匮乏。对于可信评估来说，权重的获取是至关重要的一环。许多研究确定权重主要采用专家经验获取、主成分分析法、层次分析法、信息熵、粗糙集理论等方法 [20]。在本文提出的轨道交通联锁软件可信量化评估模型中，首先建立软件面向全生命周期的可信评估指标体系，然后分别构建可信量化评估结构方程模型以确定指标权重；接着提出了指标改进优先指数和改进优先级概念并设计了基于指标改进优先指数和单位贡献成本的可信性分配算法；最后，开发了基于软件全生命周期的轨道交通联锁软件可信量化评估工具。旨在依据当前联锁软件评估结果，结合联锁的自身特点，提高联锁软件的可信性，为联锁软件质量的提高和项目的管理提供参考意见。

1.3 研究内容

轨道交通作为安全攸关领域之一，针对其遵循的国际标准技术规范以及联锁软件的特点，本文本文选取了四个属性，分别为功能性、可靠性、安全性和可维护性，建立软件全生命周期可信量化评估模型。由于联锁软件对指标的重视程度不同，提出改进优先指数的概念 [21]，并在此基础上设计两种软件可信性分配算法。最后，开发联锁软件的可信量化评估工具。

- 建立轨道交通联锁软件可信量化评估模型。首先构造面向全生命周期的阶段可信评估指标体系，每个阶段评估目标是阶段的可信度量值，一级指标为每个阶段的属性，二级指标为每个阶段的度量元。然后建立可信评估结构方程模型，对其标准化路径图中的因素负荷量进行归一化得到各级指标的权重，自下而上计算软件的可信度量值并进行可信等级的划分。

- 设计两种软件可信性分配算法。可信性分配是为了提升软件的可信等级，将子属性待提高可信值分配给其下一级的度量元。其一法根据指标改进优先指数进行可信性分配，改进优先指数是综合考虑了该指标的权重和当前的可信值。另一种分配算法基于贪心选择策略，按照度量元的单位贡献成本，从小到大依次选取，得到最低改进成本。
- 开发轨道交通联锁软件可信量化评估工具。主要功能为三大模块：一是基本信息管理模块，包含产品信息维护和人员信息维护；二是全生命周期可信评估模块，由需求分析、系统设计、编码实现和软件测试阶段可信评估四个子模块构成，每个子模块功能有可信证据输入、权重分配和可信值计算；三是评估结果模块含有数据可视化子模块与评估报告子模块，数据可视化子模块对每个阶段属性和子属性的可信值分布以雷达图和堆叠条形图的形式进行可视化展示，评估报告子模块给出软件等级等信息和具体改进意见。

1.4 文章结构

第二章中首先介绍了软件可信的相关理论；然后介绍了结构方程模型基本概念；最后介绍工具实现使用的语言和框架。

第三章中介绍了轨道交通联锁软件的可信量化评估模型。首先建立每个阶段的可信评估指标体系和可信评估的结构方程模型。然后对数据进行效度和信度检验，将路径图和数据输入 AMOS 软件，得到标准化的路径系数。最后，对路径系数归一化确定每个指标的权重，自下而上计算软件的可信值，根据可信量化分级模型表，划分软件的可信等级。

第四章主要介绍了两种可信性分配算法。首先在同时考虑指标权重和可信值的前提下，提出指标改进优先指数的概念。轨道交通联锁软件对每个指标的侧重程度不同，提升软件可信等级时，可选择按照度量元的改进优先指数对子属性可信值进行分配。由于在实际的项目中，可能会受到成本约束，因此，在此基础上又提出度量元单位贡献成本的概念，基于贪心选择策略的分配算法，得到软件可信等级

提高到某一级别时的最低改进成本。该算法得出的具体分配方案可供后续的开发过程参考，具有指导意义。

第五章主要介绍了轨道交通联锁软件可信评估系统。该系统第一个模块是基本信息管理，包括产品信息维护和人员信息维护；第二个模块是软件全生命周期四个阶段的可信评估，每个阶段包括可信证据输入，权重分配和可信值计算；最后的评估结果输出模块通过可信值图谱、堆叠区域图和表格对评估完的数据进行可视化展示，评估报告可供决策人员查看下载，安排后续的相关改进计划。

第六章对文章内容进行总结，并提出未来的研究设想。

第二章 预备知识

本章将要介绍的预备知识包括软件可信的相关理论，涵盖软件可信性的定义，软件属性、子属性和度量元，软件可信度量模型与分级方法。接着介绍结构方程模型的原理，以及系统开发所用到的框架与技术。

2.1 软件可信相关概念

2.1.1 软件可信性定义

国内外学者和科研机构从不同的方面来定义软件可信性，目前学术界对软件可信性没有统一意见。

文献 [22] 提及 [23] 给出如果一个系统是可信的，即使在运行环境发生崩溃，操作人员出现操作错误、系统遭到恶意攻击、系统存在设计和实现错误的情况下，其也能够按照预期的方式运行。文献 [24] 认为可信性是客观对象诸多属性在人们心目中的综合反映。文献 [25] 认为可信软件具有客观和主观两种性质，其中，客观上的性质即表示软件可信性，主观上用户对软件质量的认可指代可信软件。

综合国内外对于软件可信性的定义，虽然没有确定一个标准，但普遍认为可信性是主客观的综合反映，受软件基本属性的影响 [26]。软件在非正常情况下仍能正常运行的能力越强，说明软件的可信性就越高。

2.1.2 软件属性、子属性和度量元

文献 [27] 将可信性定位在正确性、私密性、可靠性、服务质量和安全性。文献 [28] 认为高可信的性质应含有可靠性、防危性、安全性、可生存性、容错性和实时性。

文献 [22] 建立的可信属性模型将可信属性划分为关键属性和非关键属性，关键属性来源于专家对可信定义，使用者根据特定需求选择非关键属性。EN50126[29] 是轨道交通软件开发遵循的欧洲标准之一，可以由铁路部门和铁路工业在在铁路应用全过程的任何阶段系统地进行应用，以制定铁路具体的 RAMS 要求并做到符合这些要求。这个标准中定义的方法与 ISO9000 系列国际标准中定义的质量管理要求的应用从某种程度上来说是一致的。

综合考虑了不同专家选取的属性和轨道交通遵循的行业标准，可信属性由多维属性组成，最终本文选取联锁软件可信性由功能性、可靠性、安全性和可维护性四个属性共同反映。

属性对于实际问题研究过于抽象，需要继续向下分解为子属性，子属性继续分解为直接获得可信证据度量的度量元 [30]。比较经典的质量模型之一 ISO/IEC9126[31, 32] 中定义的质量模型就是经过层层分解的。

在结合轨道交通联锁软件特定需求，本文对属性向下分解到子属性之后的属性分层模型如图2.1。

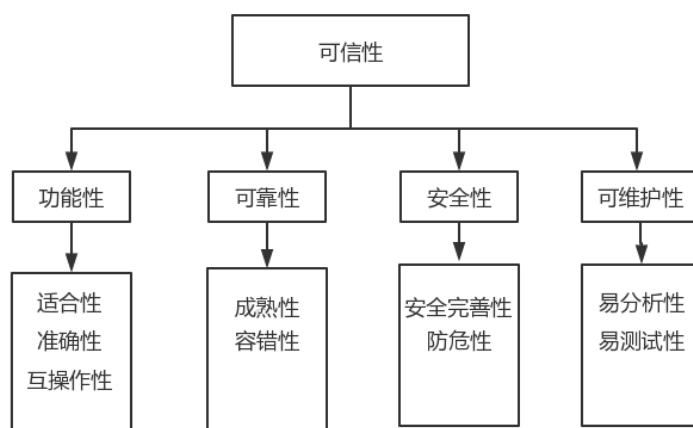


图 2.1: 轨道交通联锁软件可信属性分层模型

子属性向下进一步分解为更低层次的度量元，用来计算度量元可信值的数据称为可信证据。这些证据可以从软件源码，项目文档如需求说明书、设计文档和测试文档等，还有一些其他记录软件开发过程的文件中获取 [33]。

2.1.3 软件可信度量模型与分级方法

文献 [13] 提出了航天型号软件可信性度量与评估体系，将度量元分为两种，将子属性的可信度划分为四个等级，从 A 到 D 依次递减。子属性的可信级别由专家给出，属性的可信度由下列公式计算得出：

$$y_i = Z_{i1}^{\beta_{i1}} \cdot Z_i^{\beta_{i2}} \cdots Z_{iw_i}^{\beta_{iw_i}} \quad (2.1)$$

其中， y_i 为第 i 个属性的可信度值， Z_{ij} 表示第 i 个属性的第 j 个子属性可信度值， β_{ij} 为第 i 个属性中第 j 个子属性所占权重。可信性度量计算模型为：

$$\begin{cases} \alpha_1 + \alpha_2 + \cdots + \alpha_n = \sum_{i=1}^n \alpha_i = 1 & 0 \leq \alpha_i \leq 1 \\ T = y_1^{\alpha_1} y_2^{\alpha_2} \cdots y_n^{\alpha_n} = \prod_{i=1}^n y_i^{\alpha_i} & 0 \leq y_i \leq 1 \end{cases} \quad (2.2)$$

其中， T 为软件可信度， y_i 为第 i 个属性的可信度， α_i 为第 i 个属性的权重值。该模型满足“木桶原理”，必须保证每个属性的可信度值满足一定的条件，总体可信度才能达到要求。这就类似于物理学科中的串联电路，属性可看作电路中的单一的元器件，任意一个单元都能决定该电路是否能连通。

本文的软件可信度量模型与分级方法在 [13, 17] 的基础上，从四个阶段自下而上地计算度量元、子属性、属性、阶段到软件产品的可信值，进行可信等级划分。第 i 个度量元可信值 x_i 的取值范围为 $[0,10]$ ，分两种类型的度量元，定性的度量元可信值根据打分得来，定量度量元可信值计算如下：

$$x_i = (1 - \frac{n}{m}) * 10 \quad (2.3)$$

其中， m 为相关文档中规定该项度量元需完成的所有的任务数量， n 为实际未完成的任务数量。

$$y_i = \sum_{j=1}^n x_{ij} * \alpha_{ij} \quad (2.4)$$

x_{ij} 为第 i 个子属性中第 j 个度量元, α_{ij} 为第 i 个子属性中第 j 个度量元所占权重, y_i 为第 i 个子属性的可信值。

$$z_i = \prod_{j=1}^n y_{ij}^{\beta_{ij}} \quad (2.5)$$

y_{ij} 为第 i 个属性中第 j 子属性, β_{ij} 为第 i 个属性中第 j 个子属性所占权重, z_i 为第 i 个属性的可信值。

$$p_i = \prod_{j=1}^n z_{ij}^{\gamma_{ij}} \quad (2.6)$$

z_{ij} 为第 i 阶段中第 j 个属性, γ_{ij} 为第 i 阶段中第 j 个属性所占权重, p_i 为第 i 阶段的可信值。共有四个阶段。

$$T = \prod_{i=1}^4 p_i^{w_i} \quad (2.7)$$

软件的可信值为 T , w_i 为第 i 阶段所占的权重。

本文对 [17] 进行修改, 制定了适用于联锁软件的可信量化分级模型表, 如2.1:

2.2 结构方程模型的原理

2.2.1 结构方程模型的概念

结构方程模型 (SEM) 又称为潜在变量模型 (LVM), 它是一种验证性的分析方法, 能够同时处理测量与分析的问题并非常重视多统计指标的运用, 可以分析多个变量之间的影响关系以及变量和观测指标的一致性程度 [34]。软件可信性评估包括功能性、可靠性、安全性和可维护性, 四个可信属性之间是有内在联系的, 所以运用结构方程模型是比较合理的选择, 并且该方法允许测量误差的存在 [35]。该模型可以解决可信属性作为抽象的概念, 无法直接观测或测量, 也无法以数据量化来呈现的问题, 在实际软件可信性评估中, 是非常可取的。

表 2.1: 可信量化分级模型表

约束目标 可信等级	软件最低 可信值	阶段可信值要求	属性可信值要求
V	9.5	最低 8.5 且可信值低于 9.5 的阶段不超过 1 个	最低 8.5 且可信值低于 9.5 的属性不超过 1 个
IV	8.5	最低 7.0 且可信值低于 8.5 的阶段不超过 1 个	最低 7.0 且可信值低于 8.5 的属性不超过 1 个
III	7.0	最低 4.5 且可信值低于 7.0 的阶段不超过 1 个	最低 4.5 且可信值低于 7.0 的属性不超过 1 个
II	4.5	可信值低于 4.5 的阶段不超过 1 个	可信值低于 4.5 的属性不超过 1 个
I	1	无要求	无要求

结构方程模型由结构模型和测量模型两部分构成，前者是描述潜变量间的因果关系，后者是反映潜变量和观测变量之间的关系。潜变量是构念因素，不可直接测量或无法直接观察得到，像学习动机、智力、学业成就等 [36]。潜变量分为外因潜变量和内因潜变量，外因潜变量指模型中未受任何其他变量的影响，而它却直接影响别的变量的变量，受到任一变量影响的变量称为内因潜变量。观测变量又称显性变量、指标变量，可直接观察或测量获得，获得数据可被量化，比如语文、数学、外语三科成绩可作为学业成绩（潜变量）的测量变量 [36]。

通常用 LISREL 和 AMOS 分析 SEM，本文选择使用 AMOS，原因如下：AMOS 隶属 SPSS 系列，数据可以互用；AMOS 界面简单，用户容易上手；AMOS 得到的报表对用户来说相对容易解读 [37]。

2.2.2 结构方程模型的结构

(1) 测量模型

如下方程表示潜变量与观测指标之间的关系 [36]:

$$x = \Lambda_x \xi + \delta \quad (2.8)$$

$$y = \Lambda_y \eta + \varepsilon \quad (2.9)$$

x 和 y 表示指标变量组成向量; Λ_x 和 Λ_y 分别表示潜变量与指标变量之间的关系, 称为指标变量在潜变量上的因子负荷矩阵; ξ 和 η 分别为外因潜变量和内因潜变量; δ 和 ε 为指标变量的测量误差 [37]。

(2) 结构模型

潜变量之间的关系用如下方程刻画 [38]:

$$\eta = B\eta + \Gamma\xi + \zeta \quad (2.10)$$

其中, B 为内因潜变量之间的关系; Γ 为外因潜变量对内因潜变量的影响; ζ 为结构方程的残差项, 反映了 η 在方程中未能被解释的部分 [39]。

2.3 系统开发框架与技术

本文开发的轨道交通联锁软件可信测评系统基于 web 端, 使用后台开发语言为 Java, 框架是 Springboot, 它可以帮助开发者快速搭建 Spring 框架, 简化配置文件的操作, 开发人员主要将精力集中在业务逻辑实现上即可 [40], 视图层使用 Spring 推荐的 Thymeleaf 取代 JSP 作为模板渲染引擎。其模板的 HTML 文件实现了前后端分离, 在和后端无连接的情况下也可直接在浏览器中打开, 显示文本内容, 成功获取后端数据会取代默认文本渲染在页面上 [41]。数据库使用了 Mybatis 框架, 配置文件写好之后不需要在 java 代码中写 SQL 查询语句, 简单易学, 因为不会对应用程序或者现有的数据库设计产生任何影响, 可以降低代码和持久层的耦合。实现相关图表展示功能时用到了第三方插件 Echarts 库, 其中包含丰富的图表类型可供选择, 用户还可以根据自己的业务需求, 自定义图表显示内容。

2.4 本章小结

本章内容涉及三个方面。第一方面是软件可信性的相关概念，如软件可信性定义，软件可信属性、子属性、度量元，可信度量模型。第二方面介绍了结构方程模型的原理。最后一方面阐释了相关技术点在工具开发中的应用。

第三章 轨道交通联锁软件可信量化评估模型

文献 [42] 认为不同属性之间可能存在本质上或外在的联系，将属性分离出来研究不足以体现实际软件系统设计问题的复杂性。目前涉及属性之间的关系模型研究大多是基于经验从定性的角度出发，对于可信属性之间关系的量化没有较为系统的研究 [43]。根据轨道交通联锁软件的项目需求，本章首先建立轨道交通联锁软件的可信评估指标体系；然后构建轨道交通联锁软件的结构方程模型，得出度量元，子属性，属性的权重；最后结合软件可信度量模型与量化分级方法计算轨道交通联锁软件的可信值并进行可信等级的划分。

3.1 轨道交通联锁软件可信评估指标体系

3.1.1 评估指标体系建立的原则

软件可信评估所涉及的可信属性是多维的，动态的，评估指标比较复杂，涉及的范围比较广泛。正是由于软件可信评估体系内部变量选取的复杂与困难，根据相关理论研究，列出了可信评估指标体系建立依据的六个原则，分别是目标性、可比性、科学性、独立性、系统性和实用性 [44]。

3.1.2 可信评估指标体系的建立

本文是基于轨道交通联锁软件全生命周期进行可信评估，最后综合每个阶段的可信值计算软件整体的可信值。在实验室团队研究成果的基础上，对需求、设计、编码和测试四个阶段建立可信评估指标体系。为了保持前后阶段的一致性，属性在每个阶段一样，同时由于每个阶段针对属性具体做的事情的不同，各个阶段设计的度量元有所不同。开发一个软件首先要做的事情就是进行需求分析，确定这个

软件到底是要做什么，是后续工作的基础 [45]。本文主要介绍需求阶段可信评估指标体系的建立与结构方程模型的构建。评估目标是每个阶段可信性，一级评估指标对应属性，二级评估指标对应度量元。

在需求阶段，对功能性来说，功能定义充分程度、功能定义适配程度、功能定义正确程度、接口关系、协议、数据定义完整程度、接口可扩展程度是其二级评估指标，分别用 X1 至 X5 表示。可靠性通过成熟性要求定义充分程度、需求稳定性程度、错误处理规则识别程度、失效种类和失效后处理措施明确程度来反映，分别用 X6 至 X9 表示。[46] 安全性的二级评估指标包括通用和特定安全需求定义充分程度、软件安全性功能需求充分且需求标识定义完整程度、防危性要求定义充分程度、危险日志明确程度，分别用 X10 至 X13 表示。可维护性二级评估指标为双向追踪关系明确程度、问题分析定位能力、合格性审查要求明确程度、修改可再确认能力，分别用 X14 至 X17 表示。为了让用户深入了解软件产品的可信性，从用户角度出发建立一套评估指标也是必要的。对于软件可信性这个评估目标而言，将用户对该软件的使用反馈作为其评估指标，包含软件功能满意程度、软件运行稳定程度，软件用户交互友好程度，分别用 Y1、Y2 和 Y3 表示。

在建立评估指标体系时，首先用四个基本可信属性对抽象的软件可信性进行分解，再对四个属性进一步分解成可以直接获得观测数据的具体指标，相关从事人员可以依据具体指标的表现追根溯源，提出改进方案。需求阶段可信评估指标体系如表3.1所示：

本文用于需求阶段可信性评估的一级评估指标属性有功能性、可靠性、安全性、可维护性，假设 17 个二级评估指标之间没有重叠，保持相互独立。在实际开发的软件系统中，功能性、可靠性、安全性、可维护性是相互作用的。例如，如果对于安全性要求和功能性要求之间的冲突没有进行妥善处理，会对系统的可信性产生影响，可见安全性与功能性是存在内在关联的关系的；要想在地铁运行期间保证其安全性和功能性的目标，必须要实现可维护性和可靠性的要求，并且对于正在进行或者是周期很长的维护和运行活动以及运行环境都要严格控制。

表 3.1: 需求阶段可信评估指标体系

评估目标	一级评估指标	二级评估指标	符号
需求阶段可信性	功能性	功能定义充分程度	X1
		功能定义适配程度	X2
		功能定义正确程度	X3
		接口关系、协议、数据定义完整程度	X4
		接口可扩展程度	X5
	可靠性	成熟性要求定义充分程度	X6
		需求稳定性程度	X7
		错误处理规则识别程度	X8
		失效种类和失效后处理措施明确程度	X9
	安全性	通用和特定安全需求定义充分程度	X10
		软件安全性功能需求充分且需求标识定义完整程度	X11
		防危性要求定义充分程度	X12
		危险日志明确程度	X13
	可维护性	双向追踪关系明确程度	X14
		问题分析定位能力	X15
		合格性审查要求明确程度	X16
		修改可再确认能力	X17

3.2 轨道交通联锁软件可信评估结构方程模型

3.2.1 模型构建

根据上文建立的联锁软件可信评估指标体系，构建测量模型。由于受到四个属性的影响，软件可信性不可直接观测，作为内生潜变量；四个属性概念也比较抽象，还需进一步分解，则可作为外生潜变量；从用户角度考虑的三个指标可作为内生观测变量；因此，测量模型包括功能性、可靠性、安全性和可维护性四个外生潜变量，还有潜变量的 17 个观测指标；内部观测指标则以软件功能满意程度、软件运行稳定程度，软件用户交互友好程度表示；e1 至 e21 为误差项。使用 AMOS25.0 绘制路径关系图，建立的轨道交通联锁软件需求阶段的结构方程模型如图3.1所示：

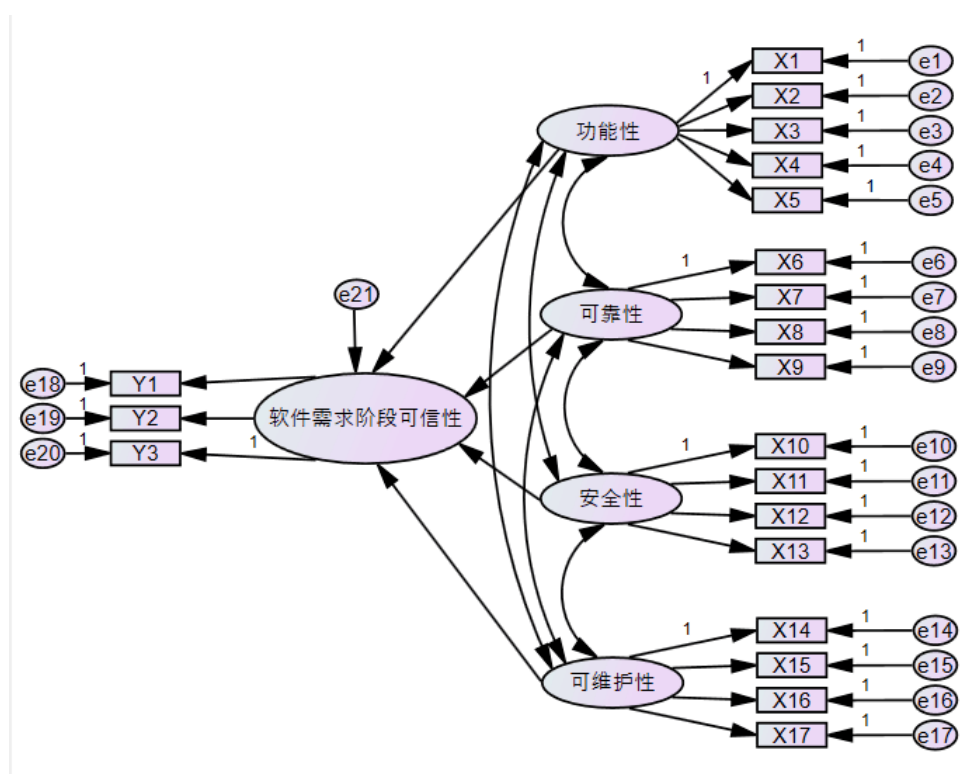


图 3.1: 轨道交通联锁软件需求阶段的结构方程模型

3.2.2 模型设定

初步假设不同潜变量的观测变量之间没有相关关系，根据假设使用 Amos 绘制的路径图如图所示。

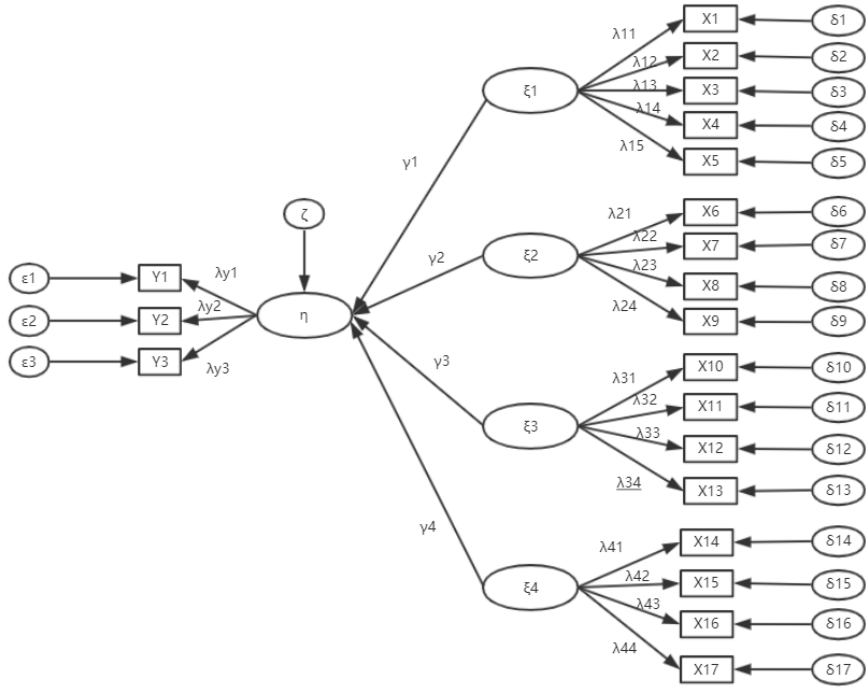


图 3.2: 可信评估结构方程模型路径图

根据上图，对于测量模型来说，外生潜变量有 4 个，因此 ξ 为 4×1 阶矩阵；内生潜变量只有 1 个，所以 η 为 1 阶矩阵。对于结构模型来说， Γ 为外因潜变量对内因潜变量的影响，是一个 4×1 阶矩阵。 ζ 为残差项，是 1 阶矩阵 [47]。

$\gamma_1 \gamma_2 \gamma_3 \gamma_4$ 表示外生潜变量对内生潜变量路径系数，分别是 $\xi_1 \xi_2 \xi_3 \xi_4$ 对内生潜变量 η 分别表示的影响程度。

$X_1 - X_{17}$ 是 ξ 的观测变量， $Y_1 - Y_3$ 是 η 的观测变量。 λ_{ij} 是因素负荷量，反映的是观测变量对外生潜变量的反映程度。因此，根据测量方程式 2.8 和 2.9，得出

的轨道交通联锁软件可信评估的测量模型如下：

$$x = \Lambda_x \xi + \delta \quad (3.1)$$

$$y = \Lambda_y \eta + \varepsilon \quad (3.2)$$

$$\eta = \Gamma \xi + \zeta \quad (3.3)$$

也可采用矩阵来表示上述模型之间的关系

$$\begin{pmatrix} X_1 \\ X_2 \\ \dots \\ X_{16} \\ X_{17} \end{pmatrix} = \begin{pmatrix} \lambda_{11} & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots \\ \lambda_{15} & 0 & 0 & 0 \\ 0 & \lambda_{21} & 0 & 0 \\ \dots & \dots & \dots & \dots \\ 0 & \lambda_{24} & 0 & 0 \\ 0 & 0 & \lambda_{31} & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \lambda_{34} & 0 \\ 0 & 0 & 0 & \lambda_{41} \\ \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \lambda_{44} \end{pmatrix} \begin{pmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \\ \xi_4 \end{pmatrix} + \begin{pmatrix} \delta_1 \\ \delta_2 \\ \dots \\ \delta_{16} \\ \delta_{17} \end{pmatrix} \quad (3.4)$$

$$\begin{pmatrix} Y_1 \\ Y_2 \\ Y_3 \end{pmatrix} = \begin{pmatrix} \lambda_{y11} \\ \lambda_{y21} \\ \lambda_{y31} \end{pmatrix} (\eta) + \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \end{pmatrix} \quad (3.5)$$

$$(\eta) = \begin{pmatrix} \gamma_{11} & \gamma_{12} & \gamma_{13} & \gamma_{14} \end{pmatrix} \begin{pmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \\ \xi_4 \end{pmatrix} + (\zeta) \quad (3.6)$$

3.2.3 模型检验与拟合

(1) 信度检验

信度是指在对同一对象使用相同的手段反复测量时所得结果的一致性程度 [48]。本文对信度检验使用在线 SPSS 工具, 检验标准通常参考 Cronbach' s Alpha 系数。一般 α 系数的值在 0 和 1 之间, 0.6 为下界, 小于等于 0.6 则不足以相信内部的一致性, 如果在 0.7 与 0.8 之间那么可以接受, 0.8-0.9 说明量表信度非常好 [49]。

(2) 效度检验

效度指测量对象能否被测量手段或者工具准确测出的程度 [50]。本文采用 AMOS25.0 进行效度检验。效度一般分为三类, 包括内容效度、准则效度和结构效度。内容效度过多依赖于主观判断, 对于准则效度来说, 很难确定一个良好的标准。因此本文针对结构效度进行检验, 求出潜变量的平均方差抽取量 (AVE), AVE 值的大小若是在 0.5 以上, 表示观测变量可以有效反映其潜变量。

(3) 模型拟合

模型拟合是判断所构造的联锁软件可信评估结构方程模型的适配度指标是否达到适配标准。根据文献 [51] 所提出的适配统计量, 本文选取以下几个常用的适配度指标:

- 卡方值与自由度比: 若卡方自由度比小于 1.0, 表示模型过度适配; 若其值大于 2.0 或 3.0 (较宽松的规定值是 5.0) 表示假设的模型无法反应真实数据, 即需要对模型进行改进, 提高模型契合度。
- 渐进残差均方和平方根 (RMSEA): 这个值越小, 越能说明模型的适配度好。通常情况, 值在 0.08-0.10 之间模型尚可, 具有普通适配度; 0.05-0.08 之间表示模型适配合理; 若其小于 0.05 表示模型的适配度非常好。
- 非规范化拟合指数 (TLI): 数值大于 0.9, 则认为模型契合度良好。
- 比较适配指数 (CFI): 数值大于 0.9, 则认为模型契合度良好。

- 增值适配指数 (IFI): 数值大于 0.09, 则认为模型契合度良好。

最终本文确定模型适配度的指标检标准验如表3.2所示:

表 3.2: 模型适配度指标检验

拟合标准	CMIN/DF	RFI	IFI	TLI	CFI	RMSEA
参照指标结果	大于 0.9	大于 0.9	大于 0.9	大于 0.9	大于 0.9	小于 0.08

3.3 示例说明

由于数据收集是一个漫长的过程, 本节针对需求阶段所用到的数据来自第三方的问卷调查, 发放给参与过软件开发过程相关的工作人员填写, 回收了有效问卷 170 份。为了更具有区分度, 问卷打分参考 Likert 量表采用 10 分制, ”非常同意”、”同意”、”不一定”、”不同意”、”非常不同意” 五种分别记为 10、8、6、4、2, 奇数表示的认可程度评价介于其前后两个数字之间。

对收集的数据进行信度分析的结果如表 2 所示, 每个因子的信度都大于 0.8, 结果表明量表信度非常好。本文样本数据的信度检验表如3.3:

表 3.3: 信度检验表

因子	Cronbach' s Alpha 系数
功能性	0.865
可靠性	0.896
安全性	0.882
可维护性	0.906
软件可信性	0.864

然后将数据文件输入到 Amos 25.0, 轨道交通连锁软件可信评估结构方程模型的标准化之后的结果如下图。

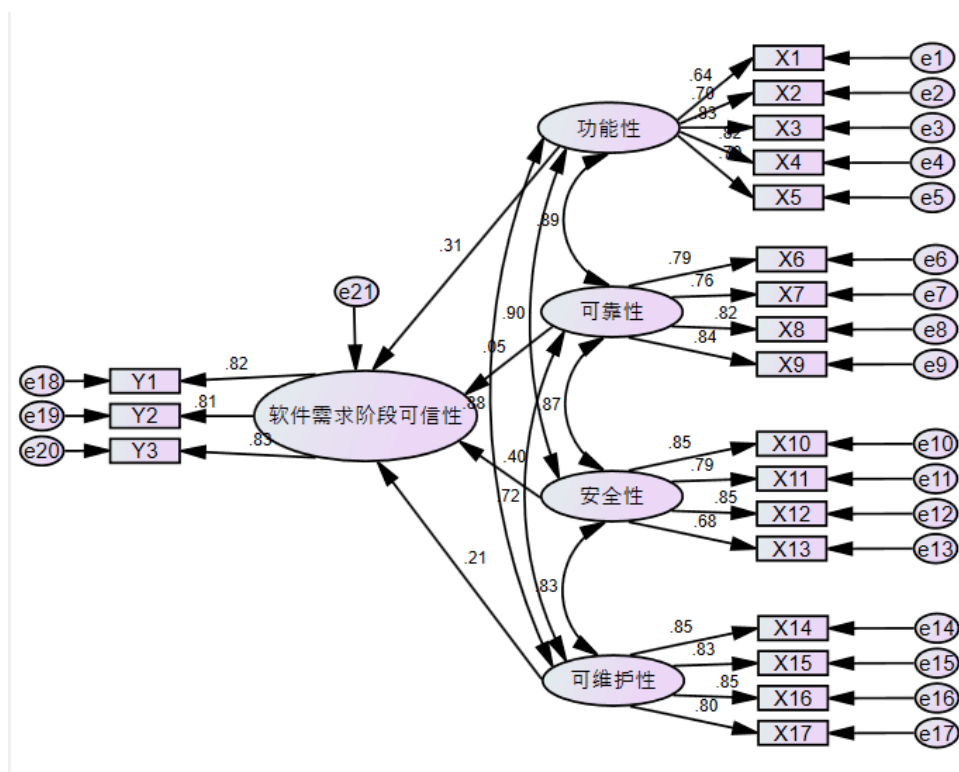


图 3.3: 轨道交通联锁软件需求阶段的结构方程模型标准化结果

样本数据效度分析结果如表3.4所示,所有指标因素负荷量的值均在 0.5 到 0.95 之间。功能性、可靠性、安全性和可维护性的平均方差抽取量 AVE 的值均大于 0.5,表明量表具有很高的结构效度。

表 3.4: 需求阶段可信评估效度检验

一级评估指标	二级评估指标	因素负荷量	AVE (平均方差抽取量)
功能性	功能定义充分程度	0.64	0.56
	功能定义适配程度	0.70	
	功能定义正确程度	0.83	
	接口关系、协议、数据定义完整程度	0.82	
	接口可扩展程度	0.76	
可靠性	成熟性要求定义充分程度	0.79	0.64
	需求稳定性程度	0.76	
	错误处理规则识别程度	0.82	
	失效种类和失效后处理措施明确程度	0.84	
安全性	通用和特定安全需求定义充分程度	0.85	0.63
	软件安全性功能需求充分且需求标识定义完整程度	0.79	
	防危性要求定义充分程度	0.85	
	危险日志明确程度	0.68	
可维护性	双向追踪关系明确程度	0.85	0.69
	问题分析定位能力	0.83	
	合格性审查要求明确程度	0.85	
	修改可再确认能力	0.80	

表3.5为模型拟合的适配度指标结果，从表中可以得出模型拟合效果从整体而言，是可以接受的。

表 3.5: 示例适配度指标检验结果

统计检验量	CMIN/DF	RFI	IFI	TLI	CFI	RMSEA
适配标准	<3	>0.9	>0.9	>0.9	>0.9	<0.08
适配结果	2.074	0.842	0.934	0.912	0.933	0.079

接下来对轨道交通联锁软件需求阶段可信性进行评估，具体步骤：首先得到每一个观测变量的分值，然后进行观测变量的因素负荷量归一化分别计算对应度量元的权重和子属性的权重，最后根据分值和权重结合公式2.4、2.5和2.6自下而上计算需求阶段可信值，如表3.6所示：

表 3.6: 需求阶段可信评估

总目标	外生潜变量（属性）	权重	子属性	权重	观测变量（度量元）	权重	分值
阶段可信值 7.12	功能性 7.10	0.32	适合性 7.48	0.36	X1	0.17	7.32
					X2	0.19	7.63
			准确性 7.83	0.22	X3	0.22	7.83
					X4	0.22	6.73
	可靠性 6.84	0.05	互操作性 6.44	0.42	X5	0.20	6.12
					X6	0.24	6.81
			成熟性 6.79	0.48	X7	0.24	6.76
					X8	0.26	7
	安全性 7.34	0.41	容错性 6.88	0.52	X9	0.26	6.75
					X10	0.27	7.12
			安全完善性 7.25	0.52	X11	0.25	7.39
					X12	0.27	7.26
	可维护性 6.81	0.22	防危性 7.44	0.48	X13	0.21	7.68
					X14	0.26	7.25
			易分析性 6.98	0.5	X15	0.24	6.69
					X16	0.26	6.54
			易测试性 6.64	0.5	X17	0.24	6.74

设计、编码和测试三个阶段的可信评估过程类似于需求阶段。假设其他三个阶段的数据和需求阶段是一样的，每个阶段同等重要所以占的权重相等，均为 0.25，根据公式2.7，得出软件整体的可信值为 $7.12^{0.25} * 7.12^{0.25} * 7.12^{0.25} * 7.12^{0.25} = 7.12$ ，对应软件可信量化分级模型表，第二列软件的可信值大于 7.0，没有可信值低于 7.0 的阶段，但每个阶段都有两个属性的可信值介于 4.5 和 7 之间，不能满足 III 级条

件，所以软件的可信等级为 IV 级。

3.4 本章小结

本章介绍了轨道交通联锁软件的可信量化评估模型。该模型首先包括可信评估体系的建立，然后在评估体系的基础上构建结构方程模型，最后结合度量模型与可信量化分级模型表得出软件可信值与等级。在章末阐述了一个示例进行具体说明。

第四章 软件可信性分配

对轨道交通联锁软件全生命周期进行可信评估之后，根据评估结果，发现问题，对于每个阶段做的不完善的地方提供改进意见。属性，子属性，度量元是要改进的指标对象，度量元是不可分解的最小单元，因此，通过改进度量元来实现改进子属性和属性的目标，最终提高软件的可信值。所以，软件可信性分配实质是将子属性待提高的可信值分配到其最低层次的度量元。为了向客户交付高可信的软件产品，那么企业要当前可信等级较低的软件进行改进。本章首先提出了改进优先指数的概念，指数的计算同时考虑了指标的权重和可信值。然后，参考软件可信分级模型表，要想使软件达到某一等级，属性，子属性的可信值必须满足一定的条件，将子属性对应需要提高的可信值分配给度量元。分配算法有两种，一种是只按照改进优先指数或改进优先级分配；另一种是基于贪心选择策略，按照度量元的单位贡献成本从小到大进行分配。

4.1 改进优先指数

4.1.1 改进优先指数概念

企业对于做的不好有待完善的工作，在时间、金钱等资源有限的前提下，不可能投入无限的人力物力成本去改进。首先选择改进的是能给企业带来最大经济效益或者是其他方面对企业最有利的。指标的改进优先指数是关于其权重和可信值的函数。改进优先级指资源受限的情况下，决策者根据改进优先指数制定的指标改进顺序。改进优先指数越低，改进优先级越高；如果指标的改进优先指数相同，权重越大优先级越高；如果改进优先指数和权重都相同，可信值越小优先级越高；

如果改进优先级、权重和可信值均相同，优先级顺序随意指定。用 P 表示指标改进优先级。

轨道交通联锁软件可信评估是一个多维、多级别的过程，在这样一个复杂的过程中，采用算术平均法根据子属性可信值计算属性可信值，根据属性可信值计算阶段可信值，根据阶段可信值计算整体软件产品的可信值，这是不合理的。

优先级的制定原则有很多种。假设指标没有权重，即将每个指标的优先级都看做是一样的；如果给指标赋予了权重，可以考虑权重高的指标优先级也高；还可以仅根据指标的完成情况制定，完成情况最差的优先级最高，以此类推。本文的改进优先指数的计算公式如下：

$$P = W * \frac{T_{max} - T}{T_{max} - T_{min}} \quad (4.1)$$

其中， P 表示指标改进优先指数， W 为当前指标权重， T_{max} 为极大可信值， T_{min} 为极小可信值， T 为当前指标可信值。

属性的改进优先指数：

$$P_{z_{ij}} = \gamma_{ij} * \frac{z_{jmax} - z_j}{z_{jmax} - z_{jmin}} \quad (4.2)$$

其中， $P_{z_{ij}}$ 表示第 i 个阶段中第 j 个属性的改进优先级， γ_{ij} 表示第 i 个阶段中第 j 个属性所占权重， z_{jmax} 表示第 i 个阶段的所有属性的最大可信值， z_{jmin} 表示第 i 个阶段的所有属性的最小可信值， z_j 表示第 i 个阶段的第 j 个属性的可信值。

子属性的改进优先指数：

$$P_{y_{ij}} = \beta_{ij} * \frac{y_{jmax} - y_j}{y_{jmax} - y_{jmin}} \quad (4.3)$$

其中， $P_{y_{ij}}$ 表示第 i 个属性中第 j 个子属性的改进优先级， β_{ij} 表示第 i 个属性中第 j 个子属性所占权重， y_{jmax} 表示第 i 个属性的所有子属性的最大可信值， y_{jmin} 表示第 i 个属性的所有子属性的最小可信值， y_j 表示第 i 个属性的第 j 个子属性的可信值。

度量元的改进优先指数公式如下：

$$P_{x_{ij}} = \alpha_{ij} * \frac{x_{j_{max}} - x_j}{x_{j_{max}} - x_{j_{min}}} \quad (4.4)$$

其中， $P_{x_{ij}}$ 表示第 i 个子属性中第 j 个度量元的改进优先级， α_{ij} 表示第 i 个子属性中第 j 个度量元所占权重， $x_{j_{max}}$ 表示第 i 个子属性的所有度量元的最大可信值， $x_{j_{min}}$ 表示第 i 个子属性的所有度量元的最小可信值， x_j 表示第 i 个子属性的第 j 个度量元的可信值。

本文计算优先指数时，考虑到指标的权重和可信值提升的难易程度两个因素，并不仅仅单一考虑权重或者可信值大小，与文献 [22] 提出的软件可信度量的凝聚性和灵敏性这两条性质相对应。凝聚性即表示随着下一级指标可信值的增加，其对上一级指标的贡献效率减少；灵敏性用来描述下一级指标的百分比变化所导致的上一级指标的百分比变化情况 [22]。灵敏性与指标的可信值和其权重有关。比如，某个指标权重最大，可信值也很大，它的优先级并不一定是最高的，因为根据凝聚性，当可信值本身已经较高时，再提高的难度也越大。

4.1.2 改进优先级计算示例

本节基于轨道交通联锁软件需求阶段制定的可信评估指标体系，选取某个属性、子属性到度量元由上至下举例如何计算指标改进优先级。

需求阶段有四个属性，安全性的权重最高，但根据改进优先级计算公式得出安全性的优先级并不是最高的。功能性和安全性的优先指数相同，安全性的权重高于功能性，所以安全性的优先级高于功能性。

功能性的子属性有三个，分别是适合性、准确性和互操作性。准确性的可信值最低，但根据改进优先指数计算公式得出的优先级别并不是最高的，它的权重拉低了整体优先指数的值。

下面是安全性属性下的安全完善完善性这个子属性的五个度量元。

表 4.1: 属性改进优先级计算示例

属性名称	权重	可信值	优先指数	优先级别
功能性	0.24	0.83	0	d 级
可靠性	0.09	0.73	0.04	b 级
安全性	0.53	0.83	0	c 级
可维护性	0.14	0.61	0.14	a 级

表 4.2: 子属性改进优先级计算示例

子属性名称	权重	可信值	优先指数	优先级别
适合性	0.22	1	0	c 级
准确性	0.16	0.63	0.16	b 级
互操作性	0.63	0.84	0.27	a 级

表 4.3: 度量元改进优先级计算示例

度量元名称	权重	可信值	优先指数	优先级别
通用安全需求	0.13	0.8	0.04	c 级
特定安全需求	0.05	0.3	0.05	b 级
软件安全性工作定义完整	0.49	0.7	0.21	a 级
安全功能需求	0.24	1	0	d 级
安全性需求标识	0.08	1	0	e 级

4.2 可信性分配

软件可信评估的过程是根据可信证据自底向上计算度量元、子属性、属性和阶段的可信值，最后计算软件整体的可信值和划分可信等级。如果软件的可信等级大于 III 级，是可以接受的；如果小于等于 III 级，理论上需要采取措施提高可信等级。可信量化分级模型表中详细列出了每个等级子属性可信值的下限，即如果子属性可信值大于等于表中的下限，属性、阶段和软件的可信值一定能满足该等级的要

求。所以，软件可信性分配实质是将子属性当前可信值与某一等级规定的子属性可信值下限进行比较，如果小于该下限，那么两者之间的差值为子属性待提高的可信值，再将这个值分配给度量元，是提高软件可信性的手段。在此之前，先计算属性的优先指数，按照优先指数从大到小计算其子属性的优先指数，再按照从大到小的顺序计算对应度量元的改进优先指数，将子属性的要提高的可信值分配给度量元。

4.2.1 可信性分配模型

在实际的软件开发项目中，提高度量元、子属性和属性的可信值是需要花费成本的。属性和子属性概念层次比较抽象，其改进成本难以直接给出。度量元处于最低层次相对具体，不再继续向下划分，其可信值是由可信证据直接计算得出的，成本较容易衡量。假设度量元的可信值每提高一个单位所需要的成本（人时/每单位）称为单位改进成本，折算成人民币用 C 表示。

在第二章的可信量化分级模型表2.1的基础上，列出不同可信等级对应的子属性最低可信值，表明若所有子属性可信值满足了某个等级要求的最低值，软件一定可以达到某个等级，即前者是后者的充分条件。

表 4.4: 可信量化分级模型表（子属性）

可信等级	V	IV	III	II	II
子属性最低可信值	9.5	8.5	7.0	4.5	1

下面是可信性分配模型

目标：

$$\min \sum_{j=1}^n \Delta x_{ij} * C_{ij} \quad (4.5)$$

满足：

$$\begin{cases} y_i = \sum_{j=1}^n x_{ij} * \alpha_{ij} \geq y_{min} \\ 0 \leq x_{ij} \leq 10 \end{cases} \quad (4.6)$$

其中, Δx_{ij} 表示第 i 个子属性的第 j 个度量元需要提高的可信值; C_{ij} 表示第 i 个子属性的第 j 个度量元的单位改进成本; y_i 为第 i 个属性的可信值; x_{ij} 表示第 i 个子属性的第 j 个度量元的可信值; α_{ij} 表示第 i 个子属性的第 j 个度量元的权重; y_{min} 表示第 i 个子属性需达到的最小可信值。

4.2.2 可信性分配算法 I

在不考虑成本约束的情况下, 仅根据度量元的改进优先指数进行改进, 指数大优先级高的先改进, 每次将度量元的可信值增加 0.1, 直到将子属性待分配的可信值全部分到度量元为止。算法流程图如图4.1。

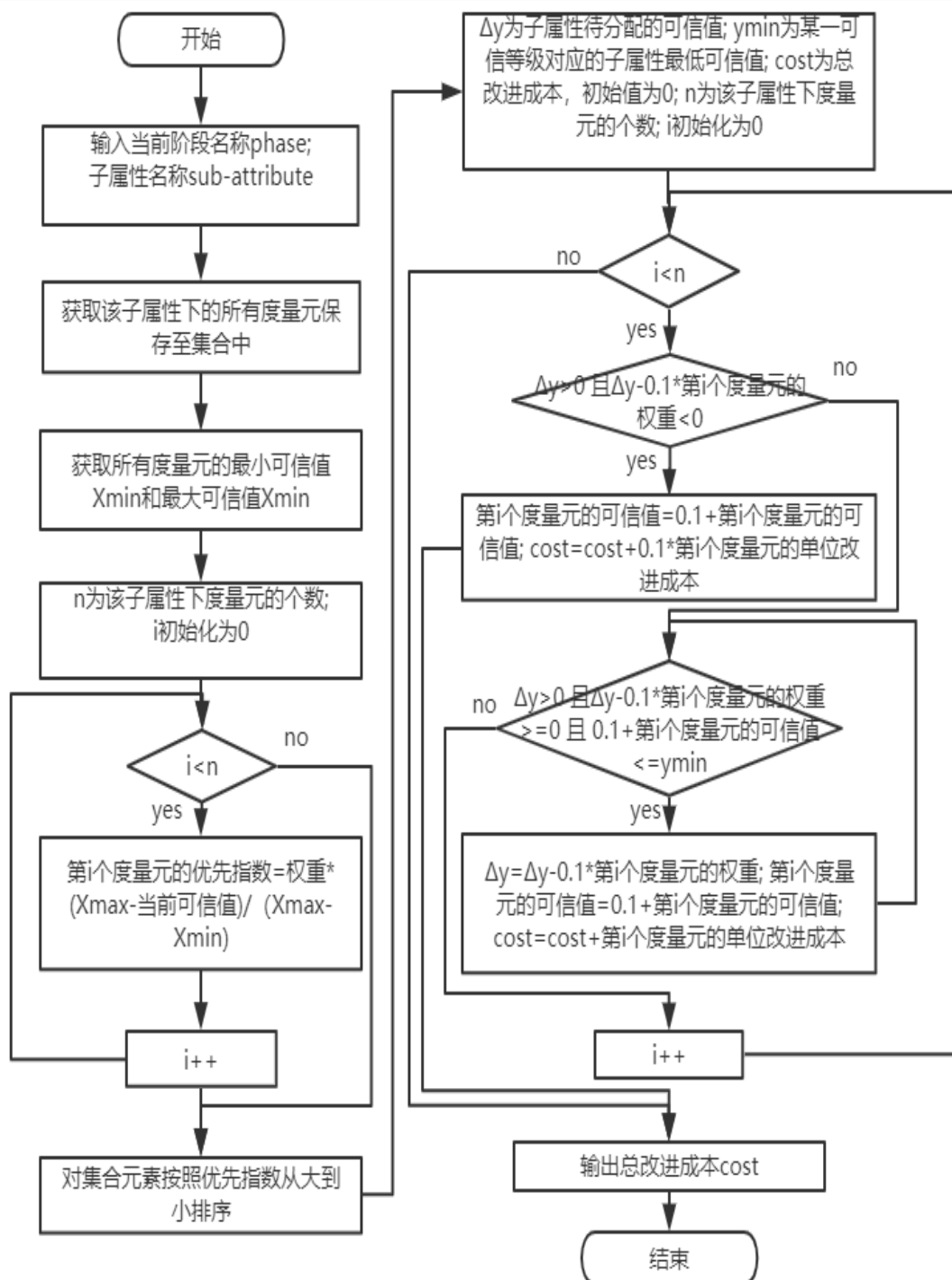


图 4.1: 算法 I 流程图

如果有 N 个度量元, 算法的时间复杂度为 $O(n)$, 空间复杂度为 $O(n)$ 。

Algorithm 1 基于改进优先指数的可信性分配算法.

Input: current phase, $phase$; current sub-attribute, $subAttribute$;

Output: total cost, $cost$;

```

1: get all measure data list  $list$  of current sub-attribute;
2: get the max trustworthy value  $x_{max}$  and the min trustworthy value  $x_{min}$  of
   metricIndex;
3:  $n$  is the length of list;
4: for  $i = 1; i < n; i++$  do
5:    $m \rightarrow priority = m \rightarrow weight * \frac{x_{max} - m \rightarrow tValue}{x_{max} - x_{min}}$ 
6: end for
7: Collections->sort(list)
8: for  $i = 1; i < n; i++$  do
9:   if  $\Delta y > 0$  and  $\Delta y - 0.1 * m \rightarrow weight < 0$  then
10:     $m \rightarrow tValue = 0.1 + m \rightarrow tValue$ 
11:     $cost = cost + 0.1 * m \rightarrow cost$ 
12:    break
13:   end if
14:   while  $\Delta y > 0$  and  $\Delta y - 0.1 * m \rightarrow weight \geq 0$  and  $0.1 + m \rightarrow tValue \leq$ 
      $y_{min}$  do
15:     $\Delta y = \Delta y - 0.1 * m \rightarrow weight$ 
16:     $m \rightarrow tValue = 0.1 + m \rightarrow tValue$ 
17:     $cost = cost + 0.1 * m \rightarrow cost$ 
18:   end while
19: end for
20: return  $cost$ 
21: end function

```

4.2.3 可信性分配示例 I

以需求阶段的安全完善性这个子属性为例,当前软件的可信等级为 *III* 级,安全完善性的可信值为 7.82。现在要将软件的可信等级提高到 *IV* 级,根据可信量化分级模型表,子属性可信值最低要达到 8.5,则待提升可信值为 0.68。假设度量元的单位改进成本为表格中数值。接下来将按照算法的流程,将子属性可信值分配给度量元,并计算提高到 *IV* 级需要投入的成本。

表 4.5: 可信性分配示例

度量元名称	权重	可信值	优先指数	单位改进成本
通用安全需求	0.14	8	0.04	2
特定安全需求	0.05	3	0.05	6
软件安全性工作定义完整	0.49	7	0.21	4
安全功能需求	0.24	10	0	8
安全性需求标识	0.08	10	0	5

首先计算度量元的优先指数分别为 $[0.04, 0.05, 0.21, 0, 0]$ 。按照优先级高先分配的原则,先将可信值分配给优先指数大的度量元。第三个度量元的优先指数最大,度量元可信值提高 1.3 个单位之后变成 8.3,子属性待分配可信值为 0.043,此时研发成本为 5.2;接下来分配第二个度量元,度量元可信值提高 0.8 个单位增加到 3.8,子属性待分配可信值剩余 0.003,研发成本为 10;子属性剩余可信值分配给下一个度量元,不足以让度量元的可信值提升 0.1 个单位,此时,按照 0.1 单位计算,可信值提高到 8.1。总改进成本为 10.2。

4.2.4 可信性分配算法 II

在实际的项目中,某些度量元的优先指数高,但改进成本可能小于优先指数低的度量元的成本。在基于优先指数的可信性分配算法中,是按照度量元的优先指数从大到小进行分配。本节中,称度量元使子属性提高 1 单位可信值需要花费的

成本为单位贡献成本。比如，安全完善性这个子属性下特定安全需求度量元的单位改进成本为 6，权重为 0.05，则该度量元的单位贡献成本为 $6 * \frac{1}{0.05} = 120$ ，意义是如果安全完善性这个子属性下其他度量元保持不变，只通过改变特定安全需求这个度量元使安全完善性的可信值提高 1 个单位，需要消耗的成本为 120。每个度量元的单位贡献成本越低，总成本越低，局部最优满足全局最优，可使用贪心法解决。具体方法：将度量元的单位贡献成本按照从小到大的顺序排列，先选择单位贡献成本小的，循环下去，直至不能再选，所得到的成本为改进子属性可信值需要的最低成本。度量元可信值每次仍然以 0.1 单位进行迭代增加。图4.2为算法流程图。

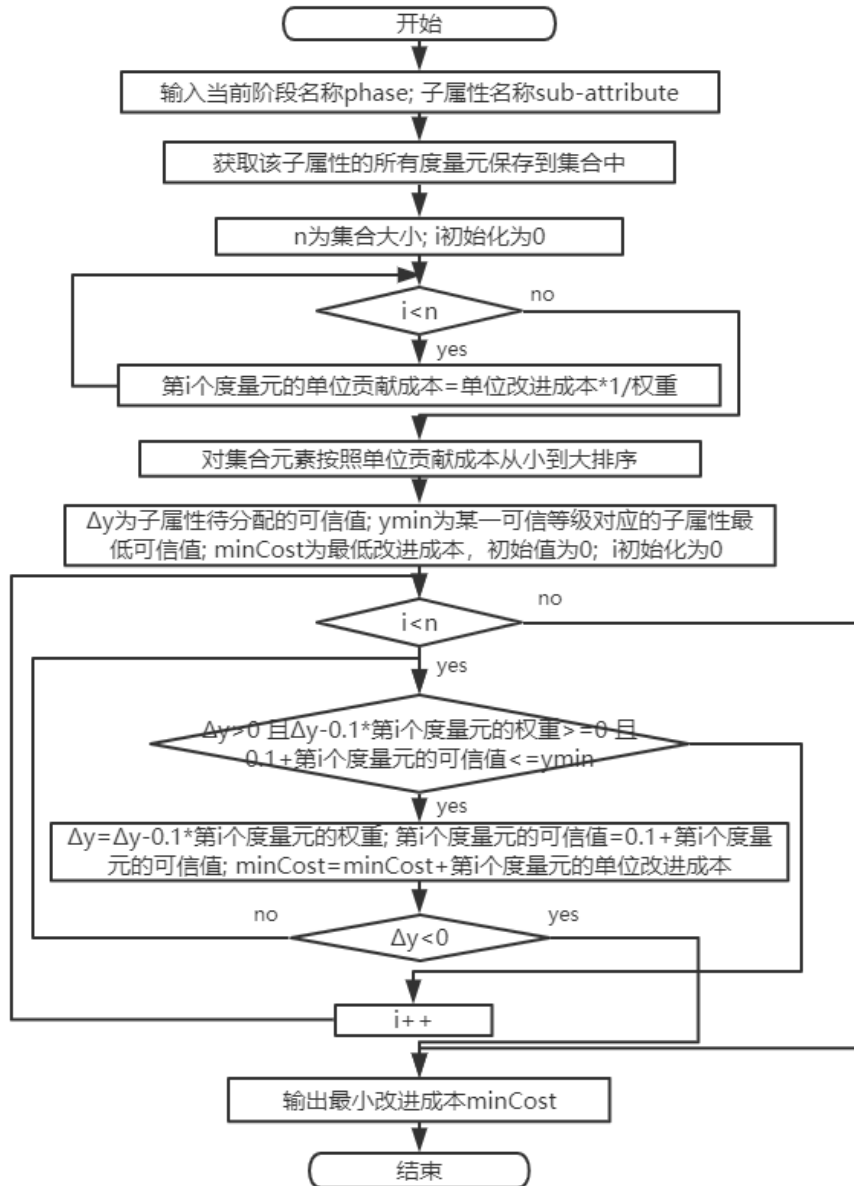


图 4.2: 算法 II 流程图

假设度量元的个数为 N , 获得子属性的所有度量元的时间复杂度为 $O(n)$, 求每个度量元的单位贡献成本时间复杂度为 $O(n)$, 对每个度量元按照单位贡献成本从小到大排序时间复杂度为 $O(n \lg n)$, 接下来的双重循环时间复杂度为 $O(n)$ * 常数, 所以算法的时间复杂度为 $O(n \lg n)$, 空间复杂度为 $O(n)$ 。

Algorithm 2 基于单位贡献成本可信性分配算法.

Input: current phase, $phase$; current sub-attribute, $subAttribute$;

Output: total cost, $minCost$;

```

1: get all measure data list  $list$  of current sub-attribute;
2:  $n$  is the length of list;
3: for  $i = 1; i < n; i++$  do
4:    $m \rightarrow conCost = m \rightarrow cost * \frac{1}{m \rightarrow weight}$ 
5: end for
6: Collections->sort(list) sort by the conCost value from small to large
7: for  $i = 1; i < n; i++$  do
8:   while  $\Delta y > 0$  and  $0.1 + m \rightarrow tValue \leq y_{min}$  do
9:      $\Delta y = \Delta y - 0.1 * m \rightarrow weight$ 
10:     $m \rightarrow tValue = 0.1 + m \rightarrow tValue$ 
11:     $minCost = minCost + 0.1 * m \rightarrow cost$ 
12:    if  $\Delta < 0$  then
13:      break
14:    end if
15:  end while
16: end for
17: return  $minCost$ 
18: end function

```

4.2.5 可信性分配示例 II

表4.6是考虑度量元单位改进成本的可信值分配示例。

表 4.6: 基于单位贡献成本的可信性分配示例

度量元名称	权重	可信值	单位改进成本	单位贡献成本
通用安全需求	0.14	8	2	14.29
特定安全需求	0.05	3	6	120
软件安全性工作 定义完整	0.49	7	4	8.16
安全功能需求	0.24	10	8	33.33
安全性需求标识	0.08	10	5	62.5

软件当前的可信等级为 *III* 级, 为使软件可信等级达到 *IV* 级, 子属性可信值需要提高 0.68 个单位。按照度量元的单位贡献成本从小到大进行分配, 其中第三个度量元单位贡献成本最低, 其可信值每次增加 0.1, 增加了 14 次之后, 可信值变为 8.4, 子属性可信值已经被分配完毕, 此时改进的最低成本为 $14 * 0.1 * 4 = 5.6$ 。

第五章 轨道交通联锁软件可信评估工具研究

本章主要介绍轨道交通联锁软件可信评估工具的分析、设计和实现过程。章末以苏州轨道交通二号线的联锁软件产品为例，介绍工具的具体使用流程。

5.1 系统分析与设计

本节首先对系统进行分析，用例图可以明确系统的用户及其对应的功能。接着系统结构图列出了系统的主要模块，最后流程图详细描述了业务流程。

5.1.1 系统分析

由于现在各界人士对软件可信性的关注度越来越高，在可信领域也有很多可信评估工具的出现，通过调研，发现一些工具在以下几个方面仍存在不足：

- 评估工具只针对某个特定的软件；
- 评估工具与用户交互的界面设计复杂，实际操作繁琐，使用者难以上手；
- 评估工具仅关注当前评估的软件可信性，而没有给出相应的改进建议或者指导。

开发工具的目的是让程序代替人类执行某些操作，减轻人们工作负担的同时高效地提供准确的结果。本文在之前的评估工具的基础上，主要在以下几个方面做了优化：

- 评估工具对轨道交通领域的所有联锁软件均适用，仅需要自定义评估可信评估数据表即可；

- 评估过程最大程度实现自动化，尽可能减少用户的操作；
- 评估结果展示当前评估软件产品的可信值与可信等级，同时对软件可信等级的提高，具体应如何改进，提供建议。

UML (Unified Modeling Language) 是针对采用面向对象开发系统的产品进行说明、可视化展示和撰写文档的一种标准建模语言，是需求分析常用工具之一[52]。其中用例图主要回答了两个问题：1、谁使用软件工具；2、工具的功能。从使用者的角度描述了系统工具的功能模块，并指出了各个功能的执行者。

本文设计的软件可信评估工具主要分为软件产品信息管理、全生命周期可信评估、评估结果三大模块。工具的使用人员包括项目负责人（管理员）和普通用户。软件产品信息管理模块包括软件产品信息维护和软件产品人员信息维护两个子模块；全生命周期可信评估即需求阶段可信评估、设计阶段可信评估、编码阶段可信评估和测试阶段可信评估，每个阶段可信评估有可信证据输入、权重分配和可信值计算三个功能点；评估结果包含了数据可视化和生成评估报告子模块，主要用于数据展示和提供改进意见。系统的用例图见图5.1：

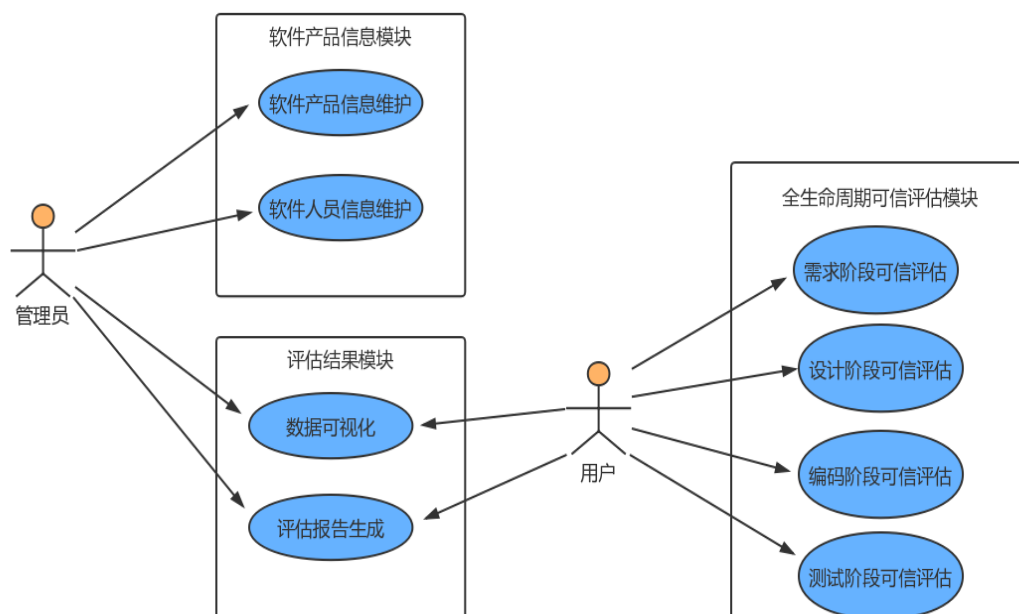


图 5.1: 联锁软件可信评估工具用例图

5.1.2 系统设计

工具共包含三大模块：基本信息管理模块、全生命周期可信评估模块和评估结果输出模块。基本信息模块由软件产品信息维护和相关人员信息维护组成。其中全生命周期可信评估模块分为需求、设计、编码和测试四个阶段可信评估子模块，每个子模块有可信证据输入、权重分配和可信值计算三个功能；评估结果输出模块包括数据可视化和评估报告生成，数据可视化子模块主要是图表分析，评估报告生成子模块给出软件的可信等级以及改进意见。图5.2是系统结构图。

整个系统的业务流程是：首先录入待评测的轨道交通联锁软件产品相关信息后，从需求阶段开始进行全生命周期的可信评估，四个阶段评估完成之后对数据进行可视化展示，显示每个阶段属性和子属性可信值图谱和双柱形图，并生成评估报告，给出当前评估软件的可信等级并对之后的开发过程提出建议。流程图如5.3。

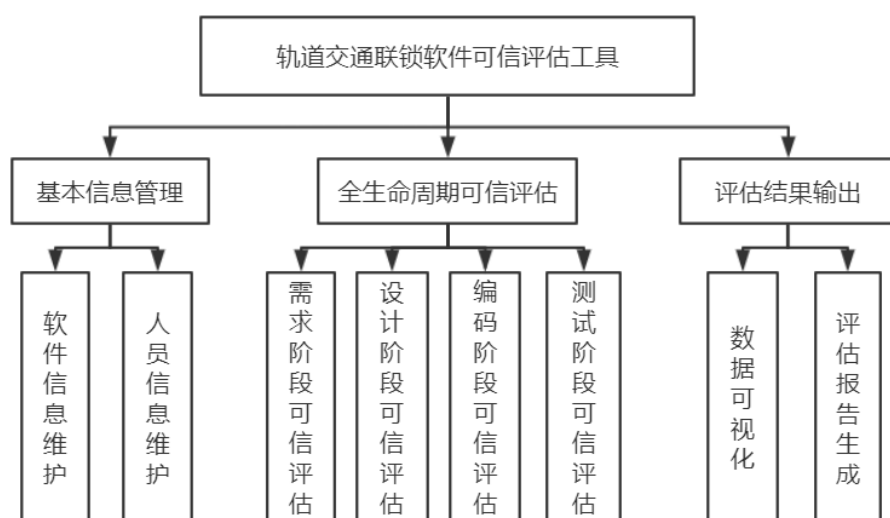


图 5.2: 联锁软件可信评估工具系统结构图

5.2 轨道交通联锁软件评估工具实现

轨道交通联锁软件评估工具实现整体采用模型-视图-控制器模式，简称 MVC (Model View Controller) 模式。用一种将业务逻辑、数据模型和 UI 界面显示分离开来的方式编写代码，将软件系统分为了三个部分：模型负责存储系统的中心数据，即实体类；视图负责用户交互；控制器处理用户输入的信息，负责从视图读取数据，并向模型发送数据。[53] 下图表示了这三者之间的关系5.4。工具后台采用当下流行的 springboot 技术，可以简化开发过程，省去很多繁冗的 XML 文件配置。工具模板引擎选择和 springboot 配套的 thymeleaf，渲染数据到前端页面。数据库同时使用了 MySQL 和 MongoDB。MySQL 是一种关系型数据库，用来存储软件产品信息和软件产品人员信息，对 MySQL 的操作通过 ibatis 框架配置的映射文件。MongoDB 是当下用的比较多最接近关系型数据库的非关系数据库，用来存储可信证据表，文档内部采用键值形式保存数据。工具主页如图5.5所示。

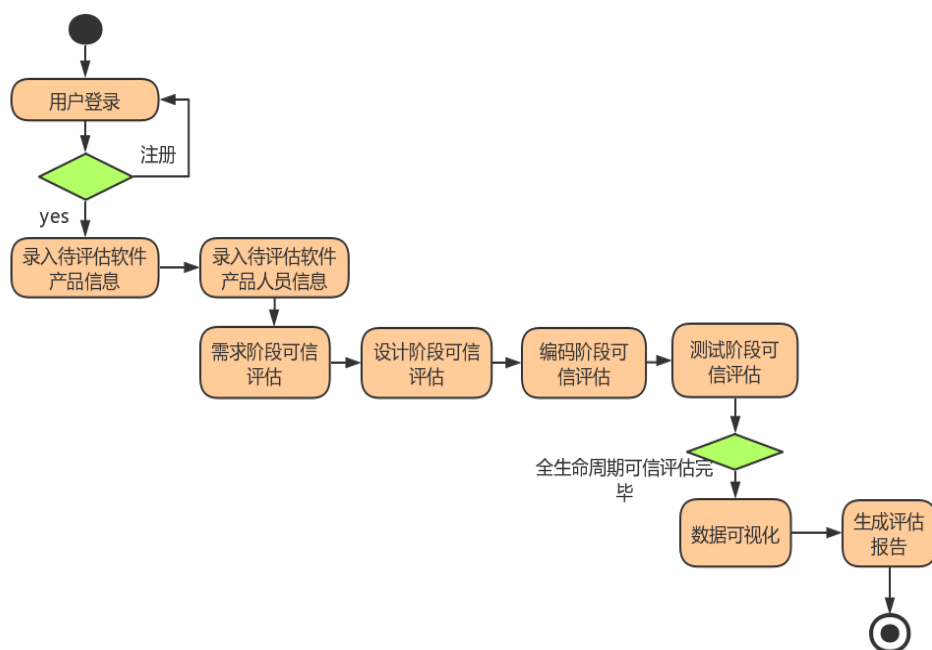


图 5.3: 联锁软件可信评估工具活动图

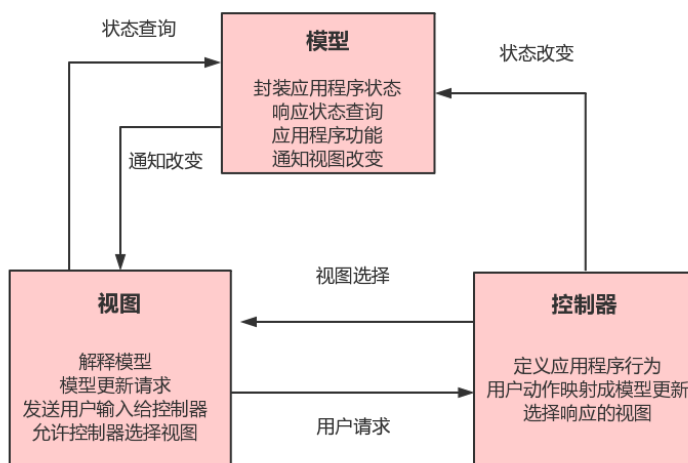


图 5.4: MVC 模式体系结构图

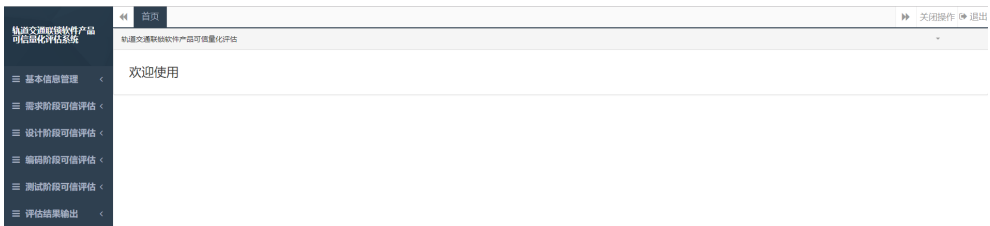


图 5.5: 工具主页

5.2.1 基本信息管理实现

(1) 软件产品信息维护子模块

本文研究的工具并不是针对某一个单独的软件产品，在某个轨道交通联锁软件评估结束之后，可以添加其他软件产品进行评估。在对软件全生命周期进行可信评估之前，首先录入待评估软件的基本信息如软件产品编号，软件产品名称，软件产品状态（待评测、评测结束），项目组长，开发负责人等。可以有选择地对当前选中的软件产品信息进行删除或者修改。录入产品信息时候默认状态均为待评估，查看评估报告按钮不可用，评估过程完成之后，状态自动显示为评估结束，可点击查看按钮查看评估报告。界面如图所5.6示。

✚添加

🗑批量删除






☐	编号	名称	状态	软件可信值	项目组长	开发负责人	测试负责人	保密等级	描述	操作	
☐	szyg-2	苏州轨道交通二号线	评测结束	0.843	aa	bb	cc	一级	这个项目由	<div><div>🔗</div><div>✕</div><div>🔍</div></div>	进入测评

显示第 1 到第 1 条记录，总共 1 条记录

图 5.6: 软件产品信息维护界面

(2) 软件人员信息维护子模块

对软件产品评估的同时也要录入负责整个软件产品开发过程的每个工作人员，若评估结果不是很理想，这样有利于决策人员之后制定改进计划，将分配分配到具体的负责人员。项目组的成员可能会变动，管理员可对人员进行维护，做增删改的操作，及时更新。

编号	姓名	部门	分组	邮箱	手机号	状态	操作
20	hewenxuan	开发	管理员	51174500080@stu.ecnu.edu.cn	19921314170	正常	 
19	AA	质量评估	QD	2250541251@qq.com	15621314170	正常	 
16	peifen	质量评估	QA	51174500010@stu.ecnu.edu.cn	18895613909	正常	 
13	yinfel	开发	DEV	51174500010@stu.ecnu.edu.cn	18895613910	正常	 

显示第 1 到第 4 条记录, 总共 4 条记录

图 5.7: 软件人员信息维护界面

5.2.2 全生命周期可信评估实现

本文软件全生命周期可信评估包括需求、设计、编码和测试四个阶段的评估。四个阶段评估过程所涉及的方法类似, 只是上传的可信证据表每个阶段不一样, 所以本节仍围绕需求阶段可信评估展开叙述。

(1) 可信证据输入

阶段可信评估首先上传可信证据(评估数据)表, 表格部分内容如下。表格中每一行表示一条可信证据, 从属性到度量元, 每一个度量元对应一个度量指标, n 和 m 的值均来源于软件产品开发过程依据的文档、程序等文件, 保证真实有效。对于定量的度量元, m 表示需要完成的指标数量, n 表示未完成的指标数量; 对于定量的度量元, 直接由经验人士打分。

在界面最上方有个下载模板文件的按钮, 上传的文件格式按照模板文件填写。填写 Excel 文件对用户来说操作比较简单, 用户可以根据当前待评估的软件产品特点, 自定义属性、子属性和度量元。Excel 文件的数据填好保存之后, 进入系统点击上传文件按钮, 选择该阶段的可信证据表。若是上传失败, 页面会弹出文件中数据不符合要求, 返回修改数据格式即可。上传成功会将 Excel 中的数据显示在页面的表格中, 点击左下角调整每页显示数据条数, 点击分页条可进行翻页操作。用户也可在页面上对可信证据进行修改或批量删除。如果用户想添加表中没有的属性, 必须相对应的添加子属性和度量元。UI 界面如图5.8。

(2) 权重分配

阶段可信评估的第二步就是依次对属性、子属性和度量元进行权重分配。文献[54]提到常见的赋权法主要有主观赋权法、客观赋权法和主客观结合赋权法, 主观



图 5.8: 需求阶段可信证据上传界面

赋权法也称专家赋权，是一种比较成熟的方法，原始数据综合专家或者决策者根据经验主观判断给出的信息，将各指标元素依据主观上的重要程度进行比较、分配权重或计算得出其权重，这种方法考虑到客观存在的实际情况，从而使指标的权重更有现实意义。层次分析法就属于其中一种，是一种灵活而又实用的层次权重决策分析方法，适用于具有分层交错评价指标的目标系统，而目标值又难以定量描述的决策问题，同时对于样本数据量要求不高。

由于实际应用中，联锁软件结构方程模型所需要的数据收集是一个漫长的过程，需要长时间的积累，因此，本节以确定属性权重为例，叙述层次分析法确定元素权重的过程。

1) 建立阶段可信评估的层次结构模型

由上至下通常分为目标层、准则层、方案层。评估阶段可信值的层次结构模型如图5.9。

2) 构造属性的正互反判断矩阵

准则层有四个属性，需要比较它们对目标层的影响程度，进而确定每个属性在该层所占的比重。属性之间两两进行比较，尺度选择 1—9。构造正互反判断矩阵 A ，在矩阵中用 a_{ij} 表示第 i 个属性相对于第 j 个属性的重要程度，数值越大，表示越重要。

$$A = (a_{ij})_{n \times n} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}$$

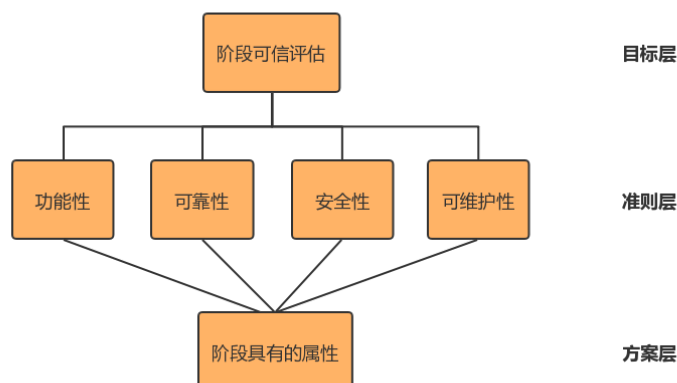


图 5.9: 阶段可信评估层次结构模型

正互反判断矩阵满足一下性质：

$$* a_{ij} > 0;$$

$$* a_{ij} = \frac{1}{a_{ji}}$$

$$* a_{ii} = 1$$

3) 计算单排序向量并进行一致性检验

层次单排序就是确定下层因素对上层某一目标影响程度的过程, 常用权值表示影响程度 [55]。一致性指标 $CI = \frac{\lambda - n}{n - 1}$, 其中 n 为 A 的对角线元素之和, 也为 A 的特征根之和; λ 为 A 的最大特征根。随机一致性指标 RI 数值如下图, 一般认为随机一致性比率 $CR = \frac{CI}{RI} < 0.1$ 时, 则认为 A 通过一致性检验。对 A 计算最大特征值及其对应的特征向量, 若此时 A 满足一致性, 则归一化之后的特征向量即为权向量。

表 5.1: 随机一致性指标 RI 数值

n	1	2	3	4	5	6	7	8	9	10	11
RI	0	0	0.58	0.90	1.12	1.32	1.41	1.45	1.45	1.49	1.51

依次输入属性、子属性和度量元的正互反判断矩阵，计算属性相对阶段的重要程度、子属性占属性的权重、度量元占子属性的比重。在正互反矩阵输入界面底部，点击保存数据，可在下次加载页面的时候显示用户上次输入的数据回显，如果用户想要修改数据，只需修改变动的数据，不需要重新输入每个值，给用户带来方便。初始化时页面默认每个值都是 1，如果用户在填写过程中想要清空所有值，点击重置数据即可。属性正互反矩阵输入页面点击下一步跳转到子属性正互反矩阵输入页面，再点击下一步是度量元正互反矩阵输入页面，三个正互反矩阵输入完成之后点击提交，如果一致性检验通过，会显示计算的属性，子属性，度量元的权重；否则页面会弹出某个正互反矩阵一致性检验没有通过的提示，返回修改数据。

属性正互反判断矩阵

需求阶段的属性正互反判断矩阵

属性	功能性	可靠性	安全性	可维护性
功能性	<input type="text" value="1"/>	<input type="text" value="1"/>	<input type="text" value="1"/>	<input type="text" value="1"/>
可靠性	<input type="text" value="1"/>	<input type="text" value="1"/>	<input type="text" value="1"/>	<input type="text" value="1"/>
安全性	<input type="text" value="1"/>	<input type="text" value="1"/>	<input type="text" value="1"/>	<input type="text" value="1"/>
可维护性	<input type="text" value="1"/>	<input type="text" value="1"/>	<input type="text" value="1"/>	<input type="text" value="1"/>

图 5.10: 需求阶段属性正互反矩阵输入界面

(3) 可信值计算

点击左侧工具栏的可信值计算按钮，系统会将后台计算的可信值结果展示在页面上，可以很直观地表示属性和子属性的对应关系。定量度量元的可信值根据可信证据（评估数据）表中的 n 和 m，通过公式 $x = (1 - \frac{n}{m}) * 10$ 计算; 定性度量元则由专业人士打分，分值范围 [0,10]。子属性的可信值根据下一层度量元的可信值和权重计算。属性的可信值根据其分解的子属性可信值和权重计算。阶段可信值由其包含的属性的可信值和权重得来。

5.2.3 评估结果输出实现

这个模块是为了将轨道交通联锁软件评估的情况以可视化的形式向用户展示，用户可以可通过图表和报告看出当前软件产品存在的问题以及后续软件开发过程还需要在哪些方面进行改进。

(1) 数据可视化

页面左侧是四个阶段属性和子属性的可信值图谱，右侧是四个阶段属性和子属性的堆叠条形图。通过可信值图谱，可以看出某个属性可信值在四个阶段比较，哪个阶段最小一目了然；还可以看出某个阶段所有属性可信值分布是否均匀。堆叠条形图具有更强的二维表现力，可以通过直条的长短清楚地看出具体可信值的大小，也可以清晰地比较某一维度数据的差异。



图 5.11: 四个阶段属性可信值图谱与堆叠条形图

图5.12具体地展示了每个阶段的所有信息，包括阶段可信值，属性可信值和权重，子属性可信值和权重，度量元可信值和权重。

需求阶段	设计阶段	编码阶段	测试阶段	
名称	所属阶段	类型	权重	可信度
需求阶段	需求阶段	阶段	0.25	0.785
功能性	需求阶段	属性	0.2385	0.833
适合性	需求阶段	子属性	0.2245	1.0
功能定义充分性	需求阶段	度量元	0.7509	1.0
功能定义适配性	需求阶段	度量元	0.2491	1.0
准确性	需求阶段	子属性	0.1555	0.63
功能定义正确性	需求阶段	度量元	0.6667	1.0
数据元素定义	需求阶段	度量元	0.3333	0.25
互操作性	需求阶段	子属性	0.6201	0.836
接口关系定义	需求阶段	度量元	0.1123	1.0
接口协议定义	需求阶段	度量元	0.1972	1.0
接口数据明确	需求阶段	度量元	0.5748	1.0
接口方式定义	需求阶段	度量元	0.0664	0.167
接口可扩展	需求阶段	度量元	0.0493	0.3

图 5.12: 四个阶段可信值表格界面

(2) 评估报告生成

评估报告页面展示了待评估软件产品的名称、编号、可信值以及可信等级。若当前软件可信等级小于等于三级，会相应地给出提高可信等级的改进建议。

软件产品编号: szyg-2	软件产品名称: 苏州轨道交通二号线	软件可信度: 0.843	可信等级: 3级
需求阶段			
功能性		0.833	
可靠性		0.726	
安全性		0.831	
可维护性		0.609	
设计阶段			
功能性		0.933	
可靠性		0.773	
安全性		0.907	
可维护性		0.473	
编码阶段			
功能性		0.783	
可靠性		0.976	
安全性		0.814	
可维护性		0.723	
测试阶段			
功能性		0.989	
可靠性		0.948	
安全性		0.989	
可维护性		0.836	

图 5.13: 评估报告界面

5.3 轨道交通联锁软件评估工具使用

本节以评估苏州有轨二号线联锁软件可信性为例，详细介绍工具使用流程。

浏览器中访问 localhost:8081，轨道交通联锁软件可信测评工具的登录页面。已经注册过的用户可在表单中直接输入用户名和密码，校验通过即可使用工具，校验不通过如姓名或密码不对应，会提示错误信息。没有注册的用户点击注册链接，弹出用户注册页面，输入相应的信息，点击确定，则注册成功，跳转到登录界面，可用注册的用户名和密码进行登录。登录成功即进入首页，如图??

软件产品管理包括产品基本信息维护和人员（用户）信息维护。

(1) 软件产品基本信息维护

产品基本信息即产品编号，产品名称，产品状态（待评测、评测结束），项目组长，开发负责人等。点击添加按钮，增加测评软件信息。默认新增的测评软件状态为待评测，可信值为 0。操作栏的图标从左至右分别表示编辑、删除、查看报



图 5.14: 登录界面

告。点击编辑图标按钮即可对软件信息进行编辑。点击删除图标按钮即可删除软件，此操作会将与该软件有关的所有信息如度量数据、评测数据全部删除。待软件的状态为评测结束之后，点击查看报告图标按钮可查看评测报告，状态为待测评测该按钮是不可用的。

图 5.15: 增加待评估软件

(2) 人员信息维护

人员（用户）信息包括姓名，部门，分组，邮箱等。主要是跟待评估软件相关的工作人员。状态分为正常和离职，管理员可删除离职的员工。点击添加按钮，增

+添加		批量删除								
<input type="checkbox"/>	编号	名称	状态	软件可信值	项目组长	开发负责人	测试负责人	保密等级	描述	操作
<input type="checkbox"/>	szyg-2	苏州轨道交通二号线	待评测	0	张三	李四	王五	三级	开发周期为两年	   
显示第 1 到第 1 条记录，总共 1 条记录										

图 5.16: 产品信息维护界面

加工作人员，填写人员基本信息，填写完成点击提交，格式不符合要求的信息会有提示。

+添加		批量删除						
<input type="checkbox"/>	编号	姓名	部门	分组	邮箱	手机号	状态	操作
<input type="checkbox"/>	20	hewenxuan	开发	管理员	51174500080@stu.ecnu.edu.cn	19921314170	正常	 
<input type="checkbox"/>	19	AA	质量评估	QD	2250541251@qq.com	15621314170	正常	 
<input type="checkbox"/>	16	peifen	质量评估	QA	51174500010@stu.ecnu.edu.cn	18895613909	正常	 
<input type="checkbox"/>	13	yinfei	开发	DEV	51174500010@stu.ecnu.edu.cn	18895613910	正常	 
显示第 1 到第 4 条记录，总共 4 条记录								

图 5.17: 人员信息维护界面

接下来就是对软件产品开发的四个阶段进行可信评估。需求阶段首先左侧导航栏的可信证据上传，然后点击下载模板的链接，参考模板文件中的格式，本地计算机中填写需求阶段可信证据 Excel 文件。接着点击选择，弹出文件对话框，选择要上传的文件，点击上传按钮，上传成功或者文件格式不符合要求会弹出提示。打开本地文件之后，点击移除按钮可移除文件，重新选择文件。文件上传成功之后，会自动刷新请求数据库，将当前测评软件的需求阶段可信证据用表格的形式显示。分页功能会显示总共多少条记录，可选择每页显示多少条记录，可选择性查看具体哪一页的记录，并且可以返回首页，跳到尾页。

← 首页

人员信息维护

可信证据输入

关闭操作 退出

下载模板

requirePhase.xlsx

上传

+添加

批量删除

<input type="checkbox"/>	软件编号	阶段	属性	子属性	度量元	目标	指标名称	类型	定义	n和m具体含义	n的值	m的值	操作
没有找到匹配的记录													

图 5.18: 上传可信证据 Excel 文件

点击添加按钮，增加一条可信证据，填完信息点击提交，或者可点击重置重新填写。选中想要删除的度量数据，点击批量删除按钮，可一次删除多条记录，避免

了逐条删除的麻烦。



图 5.19: 可信证据上传成功

权重分配首先录入属性的正互反判断矩阵，每页都点击下一步直到度量元正互反矩阵录入完毕，点击提交，三个正互反矩阵都通过一致性检验之后，跳转到权重显示页面。



图 5.20: 属性正互反矩阵

点击可信值计算按钮，后台根据可信度量模型的公式进行可信值计算，并将计算结果显示在页面上，如图中的需求阶段可信度，每个属性的可信度，每个子属性的可信度。

设计、编码和测试阶段操作同需求阶段。

点击左侧导航栏的数据可视化，查看每个阶段属性子属性可信值图谱和堆叠条形图。如果想查看每个阶段评估的具体信息，点击表格分析。

最后点击生成评估，查看苏州有轨 2 号线联锁软件的可信评估报告结果。该软

子属性正互反矩阵

功能性的子属性正互反判断矩阵

子属性	适合性	准确性	互操作性
适合性	1	2	1/4
准确性	1/2	1	1/3
互操作性	4	3	1

可靠性的子属性正互反判断矩阵

子属性	成熟性	容错性
成熟性	1	1/3
容错性	3	1

安全性的子属性正互反判断矩阵

子属性	安全完善性	防范性
安全完善性	1	1/2
防范性	2	1

可维护性的子属性正互反判断矩阵

子属性	易分析性	易测试性
易分析性	1	1
易测试性	1/2	1

保存数据 下一步 重置数据

图 5.21: 子属性正互反矩阵

件的可信值为 8.43，可信等级是 III，报告中给出了提高可信等级的具体建议，相关决策人员可参考。

此时，刷新页面，点击产品基本信息维护，当前测评软件的状态已从待评测变为评测结束，可信值也给出，点击操作栏的查看报告图表标也可以查看评测报告。

5.4 本章小结

本章主要介绍了轨道交通联锁软件可信评估工具的分析、设计与实现，并以富欣公司所做的苏州有轨二号线联锁软件为例，阐述了工具使用流程。该工具实现了不同软件基本信息的管理。用户针对不同软件可自定义每个阶段的可信证据表，上传到工具，实现软件全生命周期的可信评估。工具以图表这种非常直观的形式，让用户了解软件每个阶段的可信评估情况。当前软件评估结束之后，根据可信等级，对当前软件以及软件开发过程提出指导建议，并在评估报告中展示。

度量元正互反矩阵					
适合性的度量元正互反判断矩阵					
度量元	功能定义充分性		功能定义适配性		
功能定义充分性	1		1/3		
功能定义适配性	3		1		
准确性的度量元正互反判断矩阵					
度量元	功能定义正确性		数据元素定义		
功能定义正确性	1		2		
数据元素定义	1/2		1		
互操作性的度量元正互反判断矩阵					
度量元	接口关系定义	接口协议定义	接口数据明确	接口方式定义	接口可扩展
接口关系定义	1	1/2	1/7	2	3
接口协议定义	2	1	1/3	4	3
接口数据明确	7	3	1	9	9
接口方式定义	1/2	1/4	1/9	1	2
接口可扩展	1/3	1/3	1/9	1/2	1
成熟性的度量元正互反判断矩阵					
度量元	成熟性要求定义		需求稳定性		
成熟性要求定义	1		1/3		
需求稳定性	3		1		

图 5.22: 度量元正互反矩阵

属性名称	权重
功能性	0.24
可靠性	0.09
安全性	0.53
可维护性	0.14

属性名称	子属性名称	权重
功能性	适合性	0.22
	准确性	0.16
	互操作性	0.62
可靠性	成熟性	0.25
	容错性	0.75
安全性	安全完善性	0.33
	防范性	0.67
可维护性	易分析性	0.58
	易测试性	0.42

子属性名称	度量元名称	权重
适合性	功能定义充分性	0.25
	功能定义适配性	0.75
准确性	功能定义正确性	0.67
	数据元素定义	0.33
	接口关系定义	0.11
	接口协议定义	0.2

图 5.23: 权重显示

阶段名称	阶段可信度
需求阶段	8.19

属性名称	属性可信度
功能性	8.91
可靠性	7.28
安全性	8.37
可维护性	6.78

属性名称	子属性名称	子属性可信度
功能性	适合性	10.0
	准确性	7.53
	互操作性	8.95
可靠性	成熟性	8.88
	容错性	6.8
安全性	安全完善性	7.82
	防危性	8.67
可维护性	易分析性	7.69
	易测试性	5.25

图 5.24: 可信值计算结果

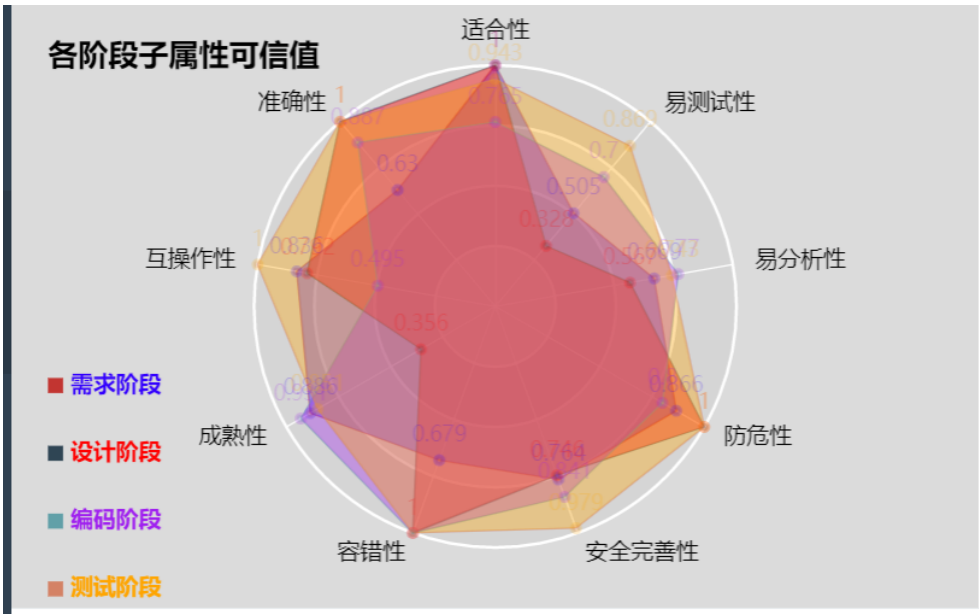


图 5.25: 四个阶段子属性可信值图谱

需求阶段	设计阶段	编码阶段	测试阶段	
名称	所属阶段	类型	权重	可信度
需求阶段	需求阶段	阶段	0.25	8.19
功能性	需求阶段	属性	0.24	8.91
适合性	需求阶段	子属性	0.22	10.0
功能定义充分性	需求阶段	度量元	0.75	10.0
功能定义适配性	需求阶段	度量元	0.25	10.0
准确性	需求阶段	子属性	0.16	7.53
功能定义正确性	需求阶段	度量元	0.67	10.0
数据元素定义	需求阶段	度量元	0.33	2.5
互操作性	需求阶段	子属性	0.62	8.95
接口关系定义	需求阶段	度量元	0.11	10.0
接口协议定义	需求阶段	度量元	0.2	10.0
接口数据明确	需求阶段	度量元	0.57	10.0
接口方式定义	需求阶段	度量元	0.07	0.0
接口可扩展	需求阶段	度量元	0.05	3.0
可靠性	需求阶段	属性	0.09	7.28
成熟性	需求阶段	子属性	0.25	8.88
成熟性要求定义	需求阶段	度量元	0.25	10.0
需求稳定性	需求阶段	度量元	0.75	8.51

图 5.26: 可信值表格分析

⏪	首页	生成评估报告	⏩	关闭操作	退出
软件产品编号: szyg-2 软件产品名称: 苏州轨道交通二号线 软件可信度: 0.843 可信等级: 3级					
需求阶段					
功能性		0.833			
可靠性		0.726			
安全性		0.831			
可维护性		0.609			
设计阶段					
功能性		0.933			
可靠性		0.773			
安全性		0.907			
可维护性		0.473			
编码阶段					
功能性		0.783			
可靠性		0.976			
安全性		0.814			
可维护性		0.773			

图 5.27: 评估报告

⏪

首页

产品信息维护

⏩

关闭操作

退出

+添加

批量删除

<input type="checkbox"/>	编号	名称	状态	软件可信值	项目组长	开发负责人	测试负责人	保密等级	描述	操作	
<input type="checkbox"/>	szyg-2	苏州轨道交通二号线	评估结束	0.843	aa	bb	cc	一级	这个项目自由	<div><div>🔍</div><div>✖</div><div>🔍</div></div>	进入测评

显示第 1 到第 1 条记录, 总共 1 条记录

图 5.28: 软件产品评估结束对应的产品信息

第六章 总结与展望

6.1 论文总结

软件应用领域广泛，软件可信综合考虑了软件的多个属性，是衡量软件质量的标准之一。全生命周期的软件可信评估对于认识当前软件的不足以及指导之后的软件开发过程有重要的参考意义。因此，本文面向轨道交通领域联锁软件的可信评估提出了相关模型与方法，具体包括下面三点：

首先，本文建立了轨道交通联锁软件可信量化评估模型。立足于四个可信属性，建立了可信评估指标体系；构建了轨道交通联锁软件可信评估的结构方程模型；根据结构方程模型的标准化结果，对因素负荷量进行归一化，计算度量元、子属性和属性的权重；结合可信度量模型的公式，自底向上计算子属性、属性、阶段以及软件整体的可信值；参考可信量化分级方法，判断当前软件可信值满足的条件，划分软件所属的可信等级。

其次，设计了两种可信性分配算法。对软件进行可信等级划分之后，如果软件的可信等级较低，需要提高软件的可信等级，即参考可信量化分级模型表，通过将子属性待提高的可信值分配给其下一级的度量元，实现软件整体可信值增加的目标。第一种分配算法是没有成本限制的条件下，按照度量元的改进优先指数从高到低进行分配，得出总改进成本；第二种分配算法是在受到成本约束条件下，基于优先选择单位贡献成本低的度量元进行分配的贪心选择策略，得出改进的最低成本。

最后，开发了轨道交通联锁软件可信评估工具。工具共包含三大模块，基本信息管理模块分两个子模块，分别为软件产品信息维护和软件人员信息维护；全生命周期可信评估模块分为需求阶段可信评估、设计阶段可信评估、编码阶段可信

评估和测试阶段可信评估四个子模块，每个子模块有可信证据输入、权重分配和可信值计算三个功能；评估结果输出模块包括数据可视化和评估报告生成子模块，数据可视化主要是图表分析，用户可以直观各个阶段看到软件可信值分布，评估报告给出软件的可信等级以及改进意见。

6.2 下一步工作

本文的工作还有很多待完善之处，为了使软件评估结果更具有说服力，以下几点需要深入研究。

首先，软件全生命周期可以考虑增加使用阶段，研究使用阶段应该关注的属性、子属性和度量元具体涉及哪些方面，与前面四个阶段进行对照。可信评估指标体系需要进一步完善，本文选取了四个可信属性，适当地增加属性以及对应的二级评估指标。

其次，对可信值分配算法进行改进。将子属性可信值分配到度量元时，根据度量元所属定性和定量两种类型，分别处理。考虑实现成本一定的情况下，如何分配使软件的可信值达到最高。

最后，完善评估工具。从用户友好性出发，界面设计更加简洁，尽量减少用户手动操作；加强功能模块的封装以及接口的标准化等，以便提高工具的可扩展程度，并且降低二次开发的难度。

参考文献

- [1] ANON. 基于“环境-行为”本体模型的软件可信演化研究 [D]. [S.l.]: 重庆大学, 2011.
- [2] ANON. 贝叶斯网络在软件可信性评估指标体系中的应用 [D]. [S.l.]: 山东轻工业学院, 2009.
- [3] WANG H, TANG Y, GANG Y, et al. Trustworthiness of Internet-based software[J]. Science China Information Sciences, 2006, 49(6): 759–773.
- [4] 熊伟, 王娟丽, 蔡铭, et al. 基于 QFD 技术的软件可信性评估研究 [J]. 计算机应用研究, 2010, 27(8): 2991–2994.
- [5] 林瑜筠. 城市轨道交通信号设备 [M]. 2006.
- [6] 彭涛. 基于 SCADE 的信息物理融合系统的分析和设计方法 [D]. [S.l.]: 广东工业大学, 2014.
- [7] 杨学伟. 论城市地铁文化建设的策略 [J]. 城市建设理论研究: 电子版, 2012.
- [8] 蔡斯博, 邹艳珍 and 邵凌霜. 一种支持软件资源可信评估的框架 [J]. 软件学报, 2010, 21(2): 359–372.
- [9] 杨善林, 丁帅, 褚伟. 一种基于效用和证据理论的可信软件评估方法 [J]. 计算机研究与发展, 2009(7).
- [10] 赵倩, 王慧强, 冯光升, et al. 基于 Pi 演算的软件可信性度量方法 [J]. 吉林大学学报 (工), 2011, 41(6): 1684–1689.

- [11] ANON. [J], .
- [12] ZHANG Y, ZHANG Y, MO H. An Evaluation Model Of Software Trustworthiness Based On Fuzzy Comprehensive Evaluation Method[C] // International Conference on Industrial Control & Electronics Engineering. 2012.
- [13] 王婧, 陈仪香, 顾斌. 航天嵌入式软件可信性度量方法及应用研究 [J]. 中国科学: 技术科学, 2015(2): 221–228.
- [14] E A, T N, J W. Toward an Approach to Measuring Software Trust[J]. IEEE Computer Society Symposium on Research in Security & Privacy, 1991.
- [15] E A, C T, J W. A process-oriented methodology for assessing and improving software trustworthiness[J]. Conference on Ccs, 1994..
- [16] ABRIAL J R. Modeling in Event-B - System and Software Engineering[M]. 2013.
- [17] 伍志强. 基于可信证据的软件可信性计算模型设计与工具实现 [D]. [S.l.]: 华东师范大学, 2019.
- [18] 李岩. 软件可信性静态度量模型设计与工具实现 [D]. [S.l.]: 华东师范大学, 2017.
- [19] 于本海. 可信软件测度理论与方法 [M]. [S.l.]: 科学出版社 s, 2014.
- [20] 张俊, 周勇. 一种基于软件属性相互影响和重要性的属性权重分配方法 [J]. 计算机应用研究, 2016, 33(5): 1390–1394.
- [21] ANON. 改进的最大优先指标及在计算机化自适应诊断测验中的应用 [D]. [S.l.]: 江西师范大学, 2011.
- [22] 陶红伟. 基于属性的软件可信性度量模型研究 [D]. [S.l.]: 华东师范大学, 2011.

- [23] SCHNEIDER F B. Trust in Cyberspace[J], 1999.
- [24] 刘克, 单志广, 王戟. “可信软件基础研究”重大研究计划综述 [J]. 中国科学基金, 2008, 22(3): 145–151.
- [25] 王怀民, 尹刚. 网络时代的软件可信演化 [J]. 中国计算机学会通讯, 2010, 6(2): 28–36.
- [26] 李俊霖, 高涛, 郁湧. 一种可信属性之间的相关性分析方法 [J]. 软件工程, 2016, 19(1): 32–34.
- [27] BECKER S, HASSELBRING W, PAUL A, et al. Trustworthy software systems: a discussion of basic concepts and terminology[J]. Acm Sigsoft Software Engineering Notes, 2006, 31(6): 1–18.
- [28] 陈火旺, 王戟, 董威. 高可信软件工程技术 [J]. 电子学报, 2003, 31(S1): 1933–1938.
- [29] ANON. 铁路应用-可靠性, 可用性, 可维护性和安全性 (RAMS) 的规范和示例 [R]. .
- [30] 张俊, 周勇. 一种基于软件属性相互影响和重要性的属性权重分配方法 [J]. 计算机应用研究, 2016, 33(5): 1390–1394.
- [31] ISO9126. Software engineering - Product quality, Part 2:. External metrics[R]. [S.l.]: ISO, 2001.
- [32] ISO9126. Software engineering - Product quality, Part 3 . Internal metrics[R]. [S.l.]: ISO, 2001.
- [33] 王德鑫, 王青. 支持软件过程可信评估的可信证据 [J]. 软件学报, , 29.
- [34] ANON. 结构方程模型及其应用 [M]. 2004.

- [35] 王玖河, 刘琳. 顾客参与价值共创机理研究——基于结构方程模型的量化分析 [J]. 企业经济, 2017(02): 75–83.
- [36] 周涛, 鲁耀斌. 结构方程模型及其在实证分析中的应用 [J]. 工业工程与管理, 2006(5).
- [37] 吴明隆. 结构方程模型——AMOS 的操作与应用 [M]. [S.l.]: 重庆: 重庆大学出版社, 2010.
- [38] 程开明. 结构方程模型的特点及应用 [J]. 统计与决策, 2006(10).
- [39] 邹德玲. 网络嵌入视角下 KIBS 企业服务创新绩效影响机制研究 [D]. [S.l.]: 东华大学, 2016.
- [40] GUTIERREZ F. Spring Boot[M]. 2017.
- [41] ANON. Spring 技术内幕: 深入解析 Spring 架构与设计原理 [M]. 2012.
- [42] HASSELBRING W, REUSSNER R. Toward Trustworthy Software Systems[M]. 2006: 91–92.
- [43] 张卫祥, 刘文红, 吴欣. 软件可信性定量评估: 模型、方法与实施 [M]. [S.l.]: 清华大学出版社, 2015.
- [44] 戴君, 贾琪, 谢琨, et al. 基于结构方程模型的可持续供应链绩效评价研究 [J]. 生态经济, 2015, 31(4): 86–89.
- [45] 汪莹, 孙玉涛. 浅谈软件需求分析 [J]. 电子世界, 2012(17): 105–106.
- [46] 郭婉琴, 王新刚. 一种机载软件安全性需求获取方法 [J]. 航空科学技术, 2014(9): 48–51.
- [47] 赵怡晴, 胡晓运, 李仲学, et al. 产品质量风险监控绩效评估的结构方程模型 [J]. 数理统计与管理, 2016, 35(5): 856–867.

- [48] 张运新, 王泉武. 某校学生对校园文体活动主观感受的调查问卷信度分析 [J]. 卫生职业教育, 2017, 35(5): 107–107.
- [49] BAIDUBAIKE. 克隆巴哈系数 [J/OL], 2019.
<https://baike.baidu.com/item/%E5%85%8B%E6%9C%97%E5%B7%B4%E5%93%88%E7%B3%BB%E6%95%B0/4350690>.
- [50] 赵学金, 吴育华. 基于结构方程的知识型服务质量的评价方法 [J]. 电子科技大学学报 (社会科学版), 2009, 11(3): 32–36.
- [51] 乔红丽. 移动图书馆用户体验的结构方程模型分析 [J]. 情报科学, 2017(2): 56–62.
- [52] LI H, ZHAO A, ZHANG D, et al. Research on building software usage model based on UML model[J]. International Journal of System Assurance Engineering & Management, 2017, 9(10): 1–9.
- [53] BAIDUBAIKE. mvc 体系结构 [J/OL], 2019.
<https://www.cnblogs.com/xiaxiaoshu/p/9073209.html>.
- [54] 刘秋艳, 吴新年. 多要素评价中指标权重的确定方法评述 [J]. 知识管理论坛, 2017.
- [55] 刘畅. 我国火电企业投资决策影响因素评价方法及模型研究 [J]. 重庆文理学院学报: 自然科学版, 2006, 5(10): 22–24.

附录 A 设计、编码和测试阶段的可信评估指标体系

表 A.1: 设计阶段可信评估指标体系

评估目标	一级评估指标	二级评估指标	符号
设计阶段可信性	功能性	设计覆盖性	X1
		功能设计正确性	X2
		数据元素设计	X3
		接口设计完整	X4
		接口通用、可扩展性设计	X5
	可靠性	成熟性设计	X6
		缺陷预防设计	X7
		模块耦合	X8
		容错处理设计	X9
		失效后处理措施设计	X10
	安全性	安全关键模块设计	X11
		人因安全设计	X12
		高风险设计	X13
		防危性设计	X14
	可维护性	可测试性设计	X15
		合格性审查设计	X16
		与需求的双向追踪关系设计	X17
		问题的分析定位设计	X18

表 A.2: 编码阶段可信评估指标体系

评估目标	一级评估指标	二级评估指标	符号
编码阶段可信性	功能性	实现覆盖程度	X1
		功能适配	X2
		数据处理精度与数据一致性实现	X3
		软件代码、单元圈和单元规模符合	X4
		接口完整、可扩展性实现	X5
	可靠性	成熟性实现	X6
		失效后处理措施实现	X7
		容错处理实现	X8
	安全性	安全相关编码规范完整	X11
		异常处理	X9
		错误源防护	X10
	可维护性	问题的分析定位实现	X15
		复杂逻辑避免实现	X16

表 A.3: 测试阶段可信评估指标体系

评估目标	一级评估指标	二级评估指标	符号
测试阶段可信性	功能性	功能测试完整	X1
		功能测试用例有效 (正常值、异常值与边界值)	X2
		功能测试用例通过	X3
		接口测试完整	X4
		接口测试用例有效	X5
	可靠性	成熟性测试完整	X6
		成熟性测试有效	X7
		故障覆盖程度	X8
		错误处理规则测试 完整程度	X9
		错误处理规则 测试有效	X10
	安全性	安全功能响应时间测试	X11
		状态转换符合程度	X12
		SIL 审查符合程度	X13
		防危性测试完整	X14
		防危性测试有效与 符合程度	X15
	可维护性	测试阶段双向追踪关系	X16
		问题的分析定位测试	X17
		自动化测试程度	X18
		可测试性的测试完整、有效 与符合程度	X19