

中图分类号：TP391

单位代码：11414
学 号：2016215017



中国石油大学
CHINA UNIVERSITY OF PETROLEUM

硕士学位论文

题 目 智慧油田数据采集与故障诊断研

究与实现

学科专业 信息与通信工程

研究方向 信号检测与处理

硕士生 王森

指导教师 曹旭东副教授

二〇一九年五月

摘要

计算机网络的发展推动了油田智能化的进程。油井多分布于荒凉偏远地区，依靠人工的方式对抽油机运行状态进行监测比较困难且实时性较差。因此，实现采油井自动化监控已是采油井科学管理的必然趋势。

示功图分析法是油井工况分析的常用方法。智慧油田数据采集与故障诊断研究与实现，设计并实现了一整套集数据采集、数据管理、故障诊断和数据展示的智能系统。

本文首先对石油行业的发展现状和抽油机故障诊断研究现状进行了分析，进而表明本文所设计的智能系统对于石油生产的重要意义。

然后详细介绍了系统的核心算法：基于灰度矩阵的示功图特征提取算法。从理论角度剖析了算法应用的可行性。详细阐述了算法的实现步骤以及特征提取过程中应用的算法思想。最终给出了通过算法提取的典型故障示功图各特征值，这些特征值是故障识别应用的依据。

以特征提取算法为核心，设计了一套具有数据管理与故障诊断功能的软件平台。该平台应用 Spring、Spring MVC、MyBatis 架构搭建，具备用户及权限管理、参数汇总、报表查询、曲线分析以及作业区管理等 5 个核心功能模块，能够满足油田生产数据自动化管理的需求。同时能够对故障进行初步的判断，对于及时了解油井工况起着重要作用。

远程终端单元是软件平台的数据来源，采集数据的准确性决定算法诊断的准确性。本文介绍了远程终端单元应用程序设计。

最后，针对系统中算法的执行中遇到的问题进行优化，并对软件平台和远程终端单元分别进行可用性和稳定性测试。测试结果证明，算法识别成功率较高，系统运行流畅稳定。

关键词：故障诊断；灰度矩阵；功图法；SSM；RTU；

Research and Implementation of Data Acquisition and Fault Diagnosis in Intelligent Oilfield

ABSTRACT

The development of computer network has promoted the development of oil field intelligence. The oil well is mostly distributed in the remote and desolate area, so it is difficult and real-time to monitor the running state of the pumping unit by manual means. Therefore, it is an inevitable trend to realize automatic monitoring and control of oil production wells in scientific management of oil production wells.

Dynamometer diagram analysis is a common method in oil well condition analysis. Intelligent Oil Field Data Acquisition and Fault Diagnosis Research and Implementation, designed and implemented a set of data collection, data management, fault diagnosis and data display intelligent system.

Firstly, this paper analyzes the development of petroleum industry and the fault diagnosis of oil pumping unit, then indicates that the intelligent system designed in this paper is of great significance to petroleum production.

Then the core algorithm of the system is introduced in detail: the feature extraction algorithm of indicator diagram based on gray matrix. The feasibility of the application of the algorithm is analyzed theoretically. The implementation steps of the algorithm and the algorithmic ideas in the feature extraction process are described in detail. Finally, the characteristic values of typical fault indicator diagram extracted by the algorithm are given, which are the basis of fault identification application.

Based on the feature extraction algorithm, a software platform with data management and fault diagnosis function is designed. The platform is constructed with Spring, Spring MVC and MyBatis architecture, It has five core functional modules, such as user and authority management, parameter summary, report form query, curve analysis and operation area management. It can meet the need of oil field production data automatic

management. At the same time, the fault can be judged preliminarily, which is very important to know the working conditions of oil well in time.

The terminal is the data source of the software platform, and the accuracy of the acquisition data determines the accuracy of the algorithm diagnosis. This paper introduces the application program of the remote terminal unit.

Finally, the problems in the implementation of the algorithms in the system are optimized, and the software platform and remote terminal unit are tested for availability and stability, respectively. The test results show that the recognition rate of the algorithm is high and the system runs smoothly and stably.

Key Words: Fault Diagnosis; Method of Indicator Diagram; Gray Matrix; SSM; RTU

目 录

声明	I
硕士学位论文版权使用授权书	I
摘要	II
ABSTRACT	III
第 1 章 绪论	7
1.1 石油行业发展现状	7
1.2 故障诊断研究发展现状	7
1.3 建立智慧油田远程诊断系统的意义	8
1.4 本文主要研究内容	8
第 2 章 系统整体设计方案	10
2.1 系统需求分析与设计目标	10
2.2 软件平台设计	11
2.3 远程终端单元设计	12
2.4 本章小结	12
第 3 章 基于灰度矩阵的示功图特征提取算法	13
3.1 相关概念介绍	13
3.1.1 抽油机工作原理	13
3.1.2 示功图	14
3.2 特征提取方法概述	17
3.3 示功图标准化	18
3.4 灰度矩阵的生成	19
3.5 特征向量提取	22
3.6 灰色关联度与示功图诊断	24
3.7 本章小结	26
第 4 章 智慧油田数据管理与故障诊断软件平台	27
4.1 关键技术介绍	27
4.1.1 数据库与 MyBatis 简介	27
4.1.2 Spring MVC 简介	30
4.2 软件平台总体功能设计	33

4.3 软件平台系统架构分析	35
4.4 数据库设计	37
4.5 用户及权限管理模块设计	42
4.5.1 用户及权限管理流程设计	42
4.5.2 用户及权限管理程序设计	43
4.5.3 用户及权限管理时序设计	46
4.6 参数汇总模块设计	48
4.6.1 参数汇总流程设计	48
4.6.2 参数汇总程序设计	49
4.6.3 参数汇总时序设计	51
4.7 曲线分析模块设计	52
4.8 其他模块设计	54
4.9 本章小结	55
第 5 章 远程终端单元设计	56
5.1 远程终端单元总体设计方案	56
5.2 远程终端单元应用程序设计	57
5.2.1 主线程设计	57
5.2.2 Modbus/RTU 通信子线程设计	58
5.2.3 Modbus/TCP 通信子线程设计	59
5.3 本章小结	61
第 6 章 系统调试和效果展示	62
6.1 算法实现优化	62
6.2 远程终端单元测试	65
6.3 软件平台测试和效果展示	68
6.4 本章小结	76
第 7 章 总结与展望	77
7.1 工作总结	77
7.2 工作展望	78
参 考 文 献	79
致 谢	81
攻读硕士学位期间参加的科研项目和学术成果	82

第1章 绪论

1.1 石油行业发展现状

石油是国家发展的重要战略资源，是城市工业化的基石。石油的储备已经成为国家发展安全战略关键之一，不仅关系到国家经济发展，更关系到每个人的生活质量。虽然风能、太阳能等新能源产业近些年有了长足的发展，但传统的石化能源依然占据主导地位。根据《BP 能源统计年鉴》的统计，以石油为参考标准，2016 年全世界的能源消耗总量相当于 133 亿吨石油。现在世界三大能源分别为石油、煤炭、天然气。其中石油消耗总量占全球能源消耗总量的 33%，石油是消耗量最多的能源^[1]。伴随社会经济的发展，对于石油能源的需求仍然与日俱增。

近些年，我国地质储油量增幅不明显甚至略有下降，石油勘察开采难度也随之增加^[2]。与此同时世界政治格局的变化加剧了石油市场和油品价格的波动。2017 年，全球经济延续缓慢复苏，对成品油需求温和增长，成品油价格上行。但存在美联储加息、新兴经济体经济下行等风险因素，同时低油价对需求的编辑拉动效应将持续减弱。作为最重要的能源商品，石油的高度不可再生性、依赖性和分布不平衡性是大国博弈石油价格的深层次原因，也是引发世界大国博弈石油控制权战争的直接原因^[3]。

由此可以看出，石油资源不仅是生活必需品，而且在当今世界经济体系中处于关键地位。但是如今石油产量日趋下降，对比日益增长的需求，我国石油开采行业现状还是很严峻的。提高原油开采的效率是提高原油产量的关键之一。但是在实际生产中，用电能交换石油的情况依然很常见，各个油田抽油机均不同程度地出现“大马拉小车”的现象，造成能源浪费。同时抽油机运行过程中出现的各种故障也会极大地影响原油开采效率。所以及时准确地对抽油机进行工况诊断，发现抽油机故障，对于提高石油开采效率有着极其重要的作用。

1.2 故障诊断研究发展现状

目前针对油井工况的诊断方法主要包括以下几种：

(1) 计算机诊断法由 Gibbs 和 Neely 于 1966 年提出^[4-5]。其基本原理是将抽油泵作为信号源，抽油杆作为传递介质信号以应力波的形式沿抽油杆传递给信号

的接收端，即动力仪。由此建立带阻尼的波动微分方程用来描述应力波在杆中传递的过程。

(2) 深井泵故障诊断专家系统。专家系统是指计算机程序，该程序可替代某一专业领域的专家。程序内部植入了大量的专家知识和经验，用于解决某一专业领域的问题^[6]。20世纪80年代，Svinos推出了一款基于Basic语言的深井泵故障诊断专家系统。该系统建立相关数据库，通过反向推理机得到典型示功图后将计算所得数据纳入专家系统，借助所计算出的数据进行故障诊断^[7]。

(3) 人工神经网络识别示功图。人工神经网络是如今识别示功图最为常见的方法之一。上世纪90年代，J.D.Rogers等人首次将人工神经网络应用于示功图的识别^[8]。此后G.M.Nazi、E.R.Martinez等人优化了原有模型并取得了不错的效果^[9]。

1.3 建立智慧油田远程诊断系统的意义

油田生产中抽油机作为主要的采油设备，其工作状态不仅反映了自身的运行情况，而且往往还能反映出地下采油的工况。因此及时准确的收集、分析能够体现抽油机工作状态的参数，对于确保石油生产工作正常高效运行起到了至关重要的作用。初期通过人工方式对抽油机进行管理，工作人员定时在油田进行巡逻，记录抽油机参数变化。然而油田多开采于偏远的荒凉地带，油田中抽油机分布无规律，且很可能处于地形复杂的区域。这种方式无疑增加了油田生产的运营成本，而且由此获取的数据具有滞后性。无法及时反映油田工况，为安全生产埋下了隐患。

随着计算机网络的普及与物联网的兴起，实现采油井自动化监控已经成为采油井科学管理的必然趋势。基于抽油机远程诊断系统设计，是将物联网概念在油气生产领域的应用，单井部署终端单元、传感器对数据进行采集，借助数据管理与智能诊断分析系统，能够实现及时、准确、直观地了解每口井的工况。在大大降低成本的同时，对于确保生产安全、提高生产效率等都有重要的现实意义。

1.4 本文主要研究内容

本文首先对功图法故障诊断算法进行分析研究，提出了基于灰度矩阵的示功图特征提取算法并详细介绍了算法实现过程。以该算法为核心，设计出一套故障诊断系统。该系统主要由基于Java Web开发环境的数据管理与故障诊断软件平台，和用于采集油井数据的远程终端单元组成。该系统主要应用于大规模井场管理，打

破了依靠人工进行故障诊断的局限性。文章最后对系统进行测试，验证了系统的可靠性。

第2章 系统整体设计方案

2.1 系统需求分析与设计目标

系统整体采用三层架构的设计模式，自底向上分别为感知层、传输层和应用层，系统整体设计方案如图 2.1 所示。感知层主要包括安装在生产系统上的各种传感器和控制设备，用于实现对生产运行状态的全面感知和各种生产设备的自动控制；传输层主要负责网络通信；应用层则是一套集数据管理、数据展示以及故障诊断等功能的智能软件平台。

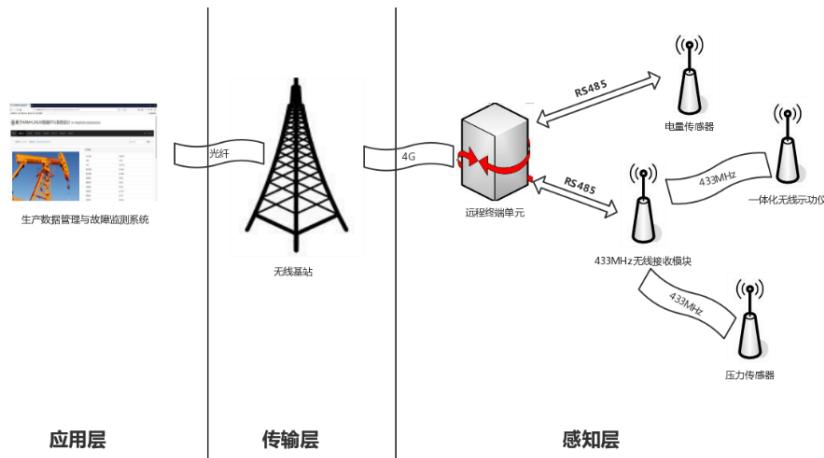


图 2.1 系统整体方案设计图
Fig. 2.1 Overall Scheme Design of The System

应用层需要设计一套软件平台系统，其主要功能包括数据管理以及故障监测，该系统是实现“让数字会说话”的关键。为了及时发现井场的异常信息，确保生产安全，要求系统具有较强的实时性以及较高的稳定性。系统采用主流 Java EE 开发框架，接入 MySQL 数据库，保证了数据管理的可靠性以及系统运行的稳定性。系统前端通过 Bootstrap 技术实现，界面设计风格简洁大方，并且要求具有较高的易用性。

感知层的工作主要是远程终端单元的研发。远程终端单元简称 RTU，是前端设备与后台系统数据传输沟通的中转站。通过 RTU 统一管理同一油井上的所有设备或相邻区域的硬件设备，可有效提高数据传输的效率，减少数据传输通道，降低

通信系统的传输负担，提高数据传输效率。其向上能通过网络与同站的智能软件平台交互，向下支持多种有线和无线的通信方式，与现场传感器进行交互。

由于现场监控的对象不同，所采用的传感器也不同，为适应不同的工作环境，需要 RTU 具备较强的兼容性。因此 RTU 采用模块化设计，用统一的协议将设备的兼容性做到最好。油井终端传感器配置有：一体化示功仪、压力变送器、电流互感器、电量模块等。此外还可根据实际需要进行自由选配，同时方便传感器的维修更换。

2.2 软件平台设计

软件平台应用 Spring、Spring MVC、MyBatis 架构搭建。SSM 架构是目前主流的 Java EE 架构。Java EE 架构具备高内聚低耦合的特点，因其程序模式规范、实现方式灵活、可扩展性强等优势，被广泛应用于企业级应用的开发。

智能软件开发平台采用经典的 Java EE 五层架构设计，这五层架构自底向上分别为领域对象（Domain Object）层、数据访问对象（Domain Access Object）层、业务逻辑（Service）层、控制器（Controller）层和前端层。应用架构如图 2.2 所示。

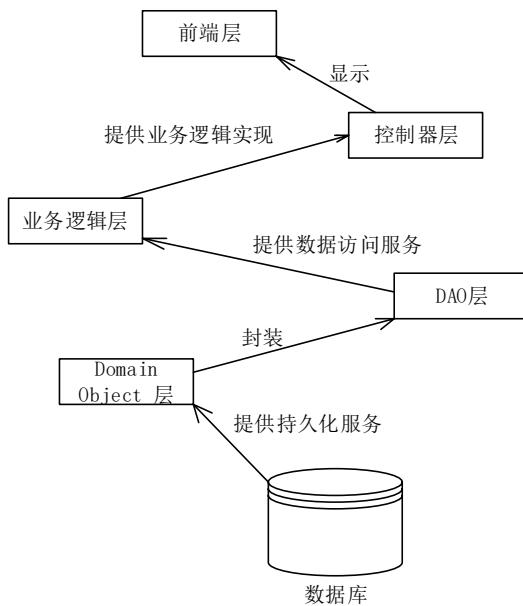


图 2.2 Java EE 应用架构图

Fig. 2.2 Java EE Application Architecture Diagram

领域对象层由一系列的 POJO 类构成。这些 POJO 类仅包含了各自属性和每个属性的 getter、setter 方法，而不包含任何业务逻辑方法。

DAO 层即数据访问层（Data Access Object），它由一系列的 DAO 组件构成，DAO 组件实现了对数据库物理表中的数据增删改查（CRUD）等原子操作。

业务逻辑层负责处理业务逻辑。业务逻辑对象通过调用 DAO 层组件的方法来实现各种业务逻辑。

控制器层用于接收用户通过前端层发送的请求，并调用业务逻辑组件来处理这些请求，最后将处理结果返回前端层。前端层再根据结果进行页面渲染，呈现给用户^[10]。

前端层包括一系列 JSP 页面、以及 Bootstrap 等前端框架，负责收集用户请求并显示处理结果。

2.3 远程终端单元设计

考虑到 RTU 功能复杂，而且对响应速度有较高的要求，RTU 核心处理器选用基于 ARM9 的 AT91SAM9263 微处理器。

RTU 系统运行要求稳定可靠并且实时性强，为了满足其高效的处理能力，采用基于嵌入式 Linux 进行应用程序的开发。嵌入式操作系统对于多任务处理能力较强，并且提供强大的网络支持，因而非常适合硬件系统开发。

RTU 设备运行过程中，为了实现对底层硬件设备的操控，应用程序调用接口向嵌入式 Linux 操作系统发送请求，Linux 系统内核在接到请求后通过调用该硬件的驱动程序来实现对硬件的操作。系统采用 Linux 2.6.3.2 操作系统作为 RTU 应用程序的运行平台，平台搭建完成后根据 RTU 的功能需求来开发应用程序，最终实现设计目标。

2.4 本章小结

本章首先根据系统的应用场景提出了系统的总体设计方案。然后对系统中的软件平台和硬件终端做了初步介绍。硬件方面，从 RTU 的实际需求出发，选定 AT91SAM9263 作为核心处理器，并基于嵌入式 Linux 操作系统开发相关应用程序；软件平台方面，采用主流 Java EE 架构进行开发，性能稳定，开放性好，安全性高。软件平台功能上以基于灰度矩阵的示功图特征提取算法为核心，具有数据管理和展示以及故障诊断等功能。

第3章 基于灰度矩阵的示功图特征提取算法

如今，对于功图法进行故障诊断大多还是通过人工的方法，依靠经验识别无疑提高了用人成本。而利用计算机自动识别示功图的方法也有很多。例如通过建立数学模型对示功图特征进行提取，此类方法需要大量数学公式推导，耗时耗力；或基于BP神经网络，此方法需要大量的功图模板，不断对模型进行训练，识别准确度完全取决于示功图模板的选择。

基于灰度矩阵的示功图特征提取，首先将示功图转换为相应的灰度矩阵，再提取灰度直方图的均值、方差、偏度、峰度、能量以及熵等6个统计特征得到特征向量。特征向量是示功图诊断的依据。

3.1 相关概念介绍

3.1.1 抽油机工作原理

有杆泵采油系统由抽油杆、抽油机和抽油泵组成，如图3.1所示。有杆泵是指利用抽油杆上下往复运动所驱动的柱塞式抽油泵。有杆泵采油具有结构简单、故障率低等特点^[12]。同时发展时间较长，技术成熟，在采油领域占主导地位。

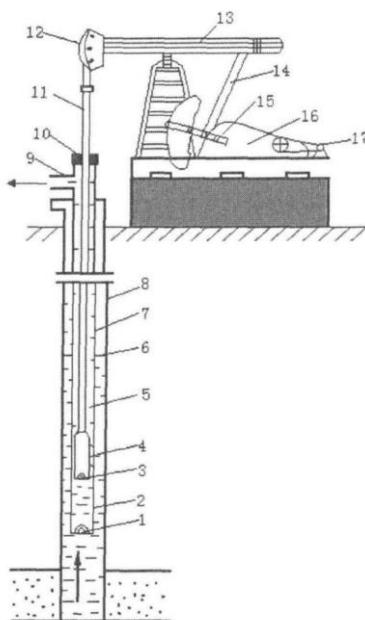


图3.1 有杆泵采油系统示意图

Fig. 3.1 Schematic Diagram of Rod Pump Production System

图中标记部分 1-17 分别为：固定凡尔；泵筒；柱塞；游动凡尔；抽油杆；动液面；油管；套管；三通；盘根盒；光杆；驴头；游梁；连杆；曲柄；减速箱；电动机。

抽油机按照结构可划分为游梁式和无游梁式。其中游梁式抽油机应用最为广泛。游梁式抽油机主要由游梁、四连杆、减速器、电动机和辅助装置等四部分组成。

有杆泵采油工作原理^[13-15]如图 3.2 所示。上冲程过程中，抽油杆与柱塞在抽油机提供的牵引力的作用下向上运动，游动凡尔在其上方液柱的压力作用下关闭；同时在上升过程中，泵筒内压力减小，固定凡尔在两侧压力差的作用下打开，井内液体通过固定凡尔被吸入泵筒。下冲程时，柱塞向下压缩泵筒内液体，由于腔体体积减小，腔体内压力增大，游动凡尔在其两侧的压力差的作用下打开；同时固定凡尔在腔内的压力作用下关闭，腔体内液体通过游动凡尔流出。在此往复的运动中，井内液面不断升高，直至井口。

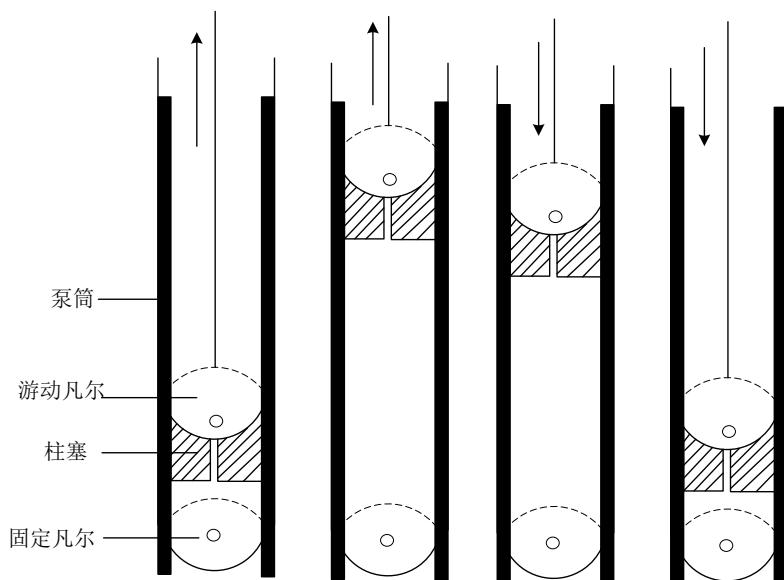


图 3.2 有杆泵采油原理图

Fig. 3.2 Principle Diagram of Sucker Rod Pumping

3.1.2 示功图

功图法是诊断抽油机运行状态的有效方法。抽油机的抽油杆上下往返运动，运动所能达到的最高点被称作上死点，同理运动所能达到的最低点被称作下死点。冲

程是抽油杆由上死点至下死点的一个往复运动的过程。而示功图就是相对下死点的位移与在此位移时刻抽油杆拉力的一一映射所绘制的曲线^[16]。其中位移 S 为横轴，拉力 F 为纵轴。功图法即通过所采集的示功图判断抽油机的运行状态。

理论示功图是在理想条件下绘制的示功图，其图形为一平行四边形，此处的理想条件是指：抽油机正常工作且不考虑抽油杆弹性形变以及抽油杆所受摩擦力。理想状态示功图如图 3.3 所示。ABC 段为上冲程曲线，其中 A 点为下死点；AB 段为加载段，在此段中游动凡尔关闭，固定凡尔关闭；B 点处加载完毕，此点处固定凡尔由关到开；BC 段为吸入段，在此段中游动凡尔关闭，固定凡尔开启；C 点为上死点，此点处固定凡尔由开到关。CDA 段为下冲程曲线，其中 CD 段为卸载段，在此段中游动凡尔关闭，固定凡尔关闭；D 点处卸载完毕，此点处游动凡尔由开到关；DA 段为排出段，在此段中游动凡尔开启，固定凡尔关闭；A 点为下死点。

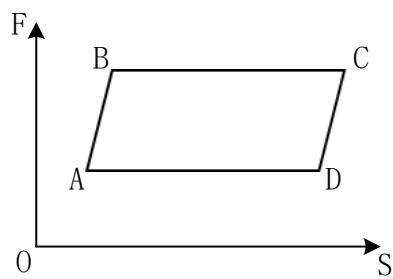


图 3.3 理想状态下示功图
Fig. 3.3 Indicator Diagram in Ideal State

而在实际中粘性原油将会导致抽油杆震动，实际生产中所采集的示功图会在理论上不同程度的变形^[17]。图 3.4 为实际测量示功图。

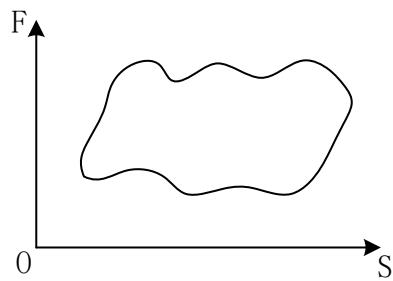


图 3.4 实际测量示功图
Fig. 3.4 Indicator Diagram of Actual Measurement

由于油藏资源的不同和抽油机的机械结构限制等原因，在油田生产过程中可能发生各种各样的故障^[18]，从而影响原油开采效率。常见故障有：泵上碰；泵下碰；冲程损失影响；抽油杆断脱；供液不足；固定凡尔漏失；活塞遇卡；结蜡；连抽带喷；气锁；气体影响；双凡尔同时漏失；液积；油稠；游动凡尔漏失；油井出砂；柱塞脱出工作筒等^[19]。典型故障示功图如图 3.5 所示。

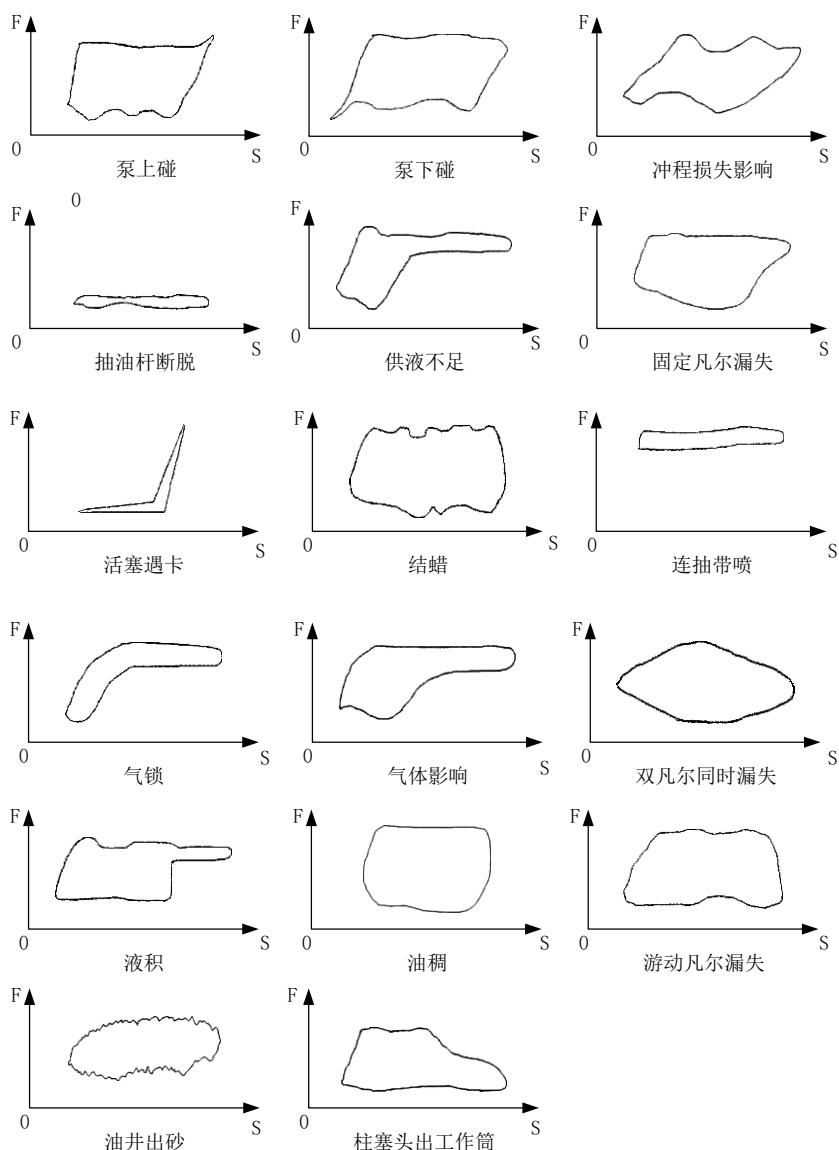


图 3.5 典型故障示功图

Fig. 3.5 Typical Fault Indicator Diagram

以上故障大多会导致原油减产，更严重的，可能会导致设备故障。及时准确地诊断采油系统的故障对保证产量有着重要意义^[20-23]。

3.2 特征提取方法概述

经验丰富的工人也许可以通过观察示功图立即判断出抽油机工作是否正常。而这些经验通过工人“言传身教”给计算机显然是不现实的。这就需要找到示功图之间的差异，即每种示功图的特征；同时需要将特征进行量化，即特征值。有了特征值，计算机就可以通过值的大小来对示功图进行故障诊断了。

目前示功图特征提取方法主要有以下几类：面积法、网格法、矢量法以及差分曲线法^[24-27]。

面积法是将示功图的面积作为识别特征，主要依据示功图面积大小以及示功图面积随时间变化的趋势。此方法只适用于区分出连抽带喷、抽油杆断脱等功图呈窄带形状的示功图。网格法是利用网格将示功图划分为若干的小区域，然后标记示功图曲线所经过的区域并由此提取示功图特征向量，此方法具有一定的局限性^[30]。矢量法以边界与图形重心间的矢量距离作为特征提取依据。虽然该方法提取特征效果较好，但计算量大^[28]，影响故障诊断效率。差分曲线法通过差分曲线来简化示功图识别过程，但该方法只适用于识别普通故障。

本文所采用的基于灰度矩阵的特征提取方法，以网格法的基础，并运用图像处理以及模式识别相关理论进行优化。其特点是计算简单，特征性强，识别效率高。为了更好地说明算法实现的步骤，本章以图 3.6 作为示例示功图。

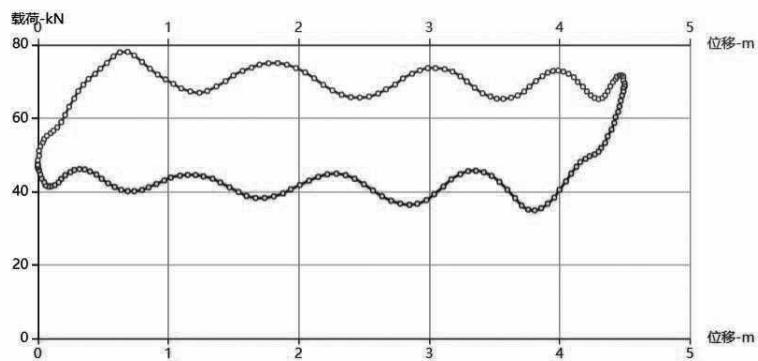


图 3.6 示例示功图
Fig. 3.6 Example of Indicator Diagram

3.3 示功图标准化

为了便于不同示功图的比较，需要对示功图进行标准化处理。利用网格法的思想，将示功图放置在边长比为 2: 1 的矩形内，矩形边界与示功图边界相切，即最小外接矩形。然后以 $M \times N$ 的网格对示功图进行划分，这样做的目的是实现示功图的无量纲化，这是特征提取的基础。参与油井工况分类判别计算的特征向量比为^[29]:

$$R = \frac{6}{M \times N} \quad (3.1)$$

该公式中， R 为特征向量比， $M \times N$ 为网格尺度，其中 M 、 N 满足 $M:N = 2:1$ ，并且 M 、 N 取值越大，对与示功图的细节描述就越细致，识别效果就更好。但与此同时计算量也会随之增加，影响识别效率。通过特征提取的准确率与效率之间进行权衡，最终选定 $M \times N$ 取值为 64×32 ，此时特征向量比 $R = 3/1024$ ，该值在不影响识别效率的同时也保证了识别精度。

在实际应用中，RTU 采集示功图数据在数据库中以 200 点位移和 200 点载荷分别存储。算法程序中，首先初始化一个所有值均为 0 的 64×32 大小的矩阵，对于每对载荷-位移值，通过无量纲化处理，将其分别映射至矩阵并在矩阵该点赋值为 1。

载荷、位移无量纲化公式为

$$x(i) = \frac{l(i)}{\max_k l(k) - \min_k l(k)} \times 32 \quad 1 \leq i \leq 200, 1 \leq k \leq 200 \quad (3.2)$$

$$y(i) = \frac{s(i)}{stroke} \times 64 \quad 1 \leq i \leq 200 \quad (3.3)$$

公式中， $x(i)$ 为第 i 点的载荷值映射到矩阵的行值， $l(i)$ 为第 i 点的载荷值， $\max_k l(k)$ 为最大载荷， $\min_k l(k)$ 为最小载荷。同理， $y(i)$ 为第 i 点的位移值映射到矩阵的行值， $s(i)$ 为第 i 点的位移值， $stroke$ 为冲程。这样就实现了示功图的标准话。示例示功图标准化结果如图 3.7 所示。

图 3.7 标准化示功图

Fig. 3.7 Standardized Indicator Diagram

3.4 灰度矩阵的生成

利用标准化后的示功图矩阵，进一步生成灰度矩阵。将示功图按照等高线划分原则对其灰度值进行赋值。边界内灰度值随相对边界距离增大而增大，边界外灰度值随相对边界距离增大而减小。

这就需要确定矩阵中每个点相对于示功图边界的位置。以示功图边界作为划分依据，将示功图进行分割。

联通区域的划分利用深度优先搜索（Depth First Search）算法。找到矩阵中某一未被访问的点（该点值为 0）执行 DFS 算法，对当前坐标点用当前联通区域计数（计数从 100 起）进行标记，之后判断相对当前点上下左右四个坐标点是否已被访问。若发现其中某一点未被标记，则进入下一层递归，标记该点并将该点作为新的当前点，对该点执行相同的深度优先搜索操作；若上下左右四个方向坐标点均已被标记，则跳出该层递归。很明显，DFS 算法递归的终止条件是数组坐标越界或该坐标点已被标记。若此时矩阵中还有未被访问的坐标点，则将该点作为新的起点对其进行重新执行 DFS，直到矩阵中所有坐标点均被访问。在标记过程中，还需要判断联通块相对于示意图边界的相对位置。最终得到的连通区域划分如图 3.8 所示。

图 3.8 联通块区域划分

Fig. 3.8 Regional Division of Interconnection Blocks

利用 DFS 算法进行联通区域划分的优势是，遍历过程中可判断每个起点的相对位置，该点的相对位置即该连通区域的相对位置，新建 Map 集合来存储每个连通区域标号以及对应的相对位置，避免对每个点逐一进行判断，提高程序运行效率。

在确定每个点的相对位置之后，按照等高线原则，对示功图进行灰度赋值，示功图边界内的点，每远离边界一点，其灰度值加 1；示功图边界外的点，每远离边界一点，其灰度值减 1。

为实现上述需求，可以借助广度优先搜索算（Breadth First Search），对示功图坐标点进行遍历，这是经典迷宫问题的一种应用。BFS 算法实现流程图如图 3.9 所示。

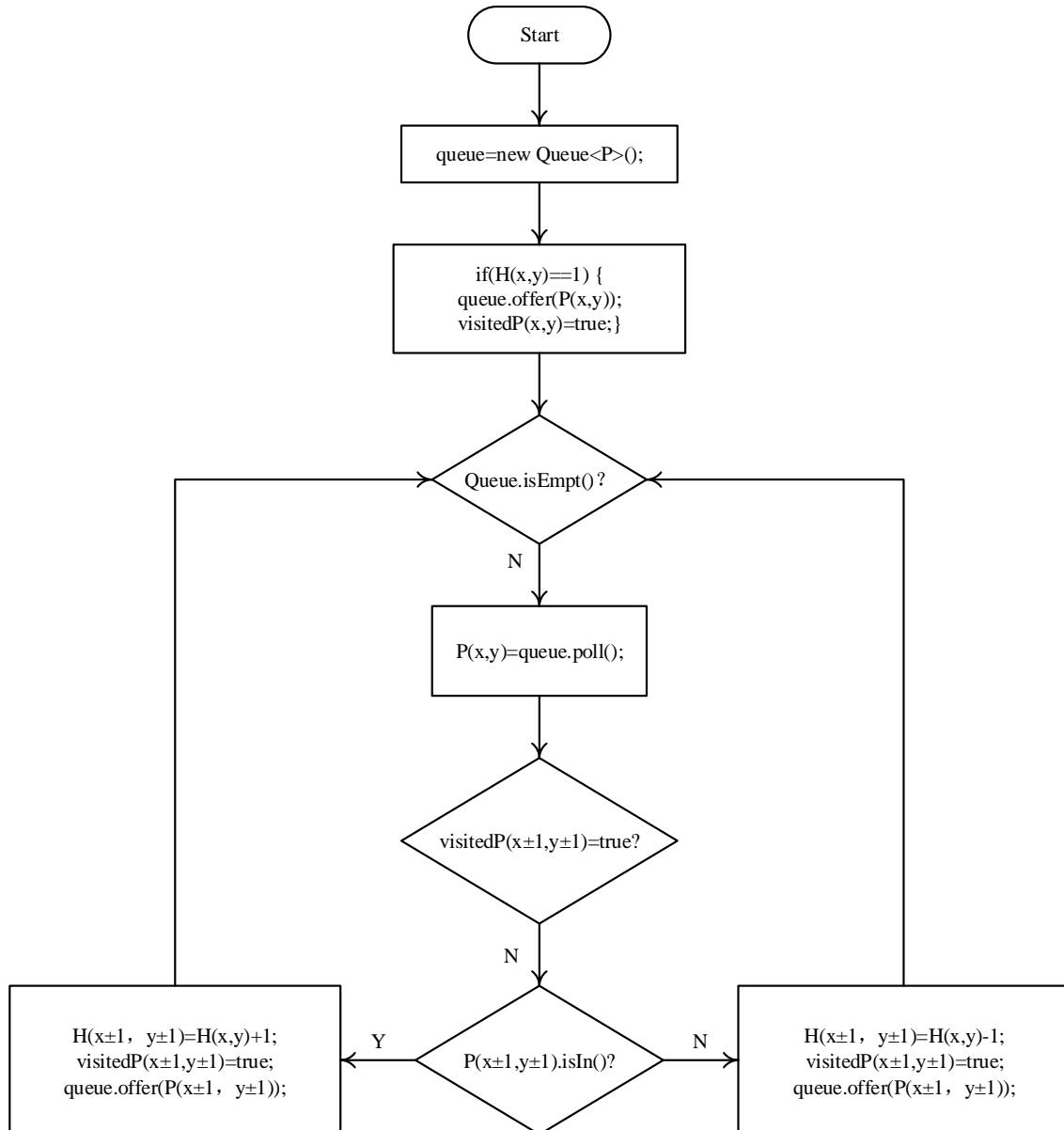


图 3.9 BFS 算法流程图

Fig. 3.9 Flow chart of BFS

新建队列，用于存储坐标，将所有灰度值为 1 的点的坐标进行入队操作，并将其 visited 状态值改为 true。按照先进先出原则，队首元素出队，遍历其四周相邻坐标点 visited 状态值，将所有 visited 值为 false 的点的灰度值赋值为出队元素的灰度值加 1，visited 状态值改为 true，并将其进行入队操作。继续将当前队首元素出队，重复执行上述操作，直至示功图所有坐标点均被遍历，即队列为空，遍历中止。利用此算法生成的灰度矩阵如图 3.10 所示。

图 3.10 灰度矩阵

Fig. 3.10 Gray matrix

3.5 特征向量提取

灰度直方图是灰度级的函数，描述的是图像中具有该灰度级的像素的个数：其中，横坐标是灰度级，纵坐标是该灰度级出现的频率^[30]。灰度直方图可以反映灰度矩阵灰度级的分布情况，若将灰度矩阵的灰度级别看作随机变量，则灰度直方图则是其重要的统计特征。

对于示功图的灰度矩阵，获取其灰度直方图。将灰度直方图的 6 个统计特征作为识别该示功图的特征依据。这 6 个统计特征包括均值、方差、偏度、峰度、能量和熵。

设灰度矩阵为 $G(M, N)$, 灰度矩阵中的元素为 $G(m, n)(1 \leq m \leq M, 1 \leq n \leq N)$, 统计某一灰度级 r 在灰度矩阵中的总个数为 $b(r)$, 那么灰度级 r 在灰度矩阵中出现的概率如公式 3.4 所示。

$$p(r) = \frac{b(r)}{M \times M} \quad (3.4)$$

则 $G(M, N)$ 的 6 个统计特征分别为:

(1) 均值反映了灰度矩阵中所有灰度级的平均值。设灰度矩阵中最大灰度级为 R , 则灰度均值计算公式如公式 3.5 所示。

$$\bar{g} = \sum_{r=1}^R r \cdot p(r) \quad (3.5)$$

(2) 方差是指灰度值于灰度均值的差的平方均值。它反映了灰度矩阵中灰度级的波动大小。方差的计算公式如公式 3.6 所示。

$$\sigma^2 = \sum_{r=1}^R (r - \bar{g})^2 \cdot p(r) \quad (3.6)$$

(3) 偏度是对灰度级所构成的分布的对称性状况的描述。它反映了实际灰度值相对于期望值（均值）的偏离程度。偏度的计算公式如公式 3.7 所示。

$$S = \frac{1}{\sigma^3} \sum_{r=1}^R (r - \bar{g})^3 \cdot p(r) \quad (3.7)$$

(4) 峰度是对灰度级所构成的分布的峰值突兀程度的描述。峰度用四阶中心距来表示，计算公式如公式 3.8 所示。

$$K = \frac{1}{\sigma^4} \sum_{r=1}^R (r - \bar{g})^4 \cdot p(r) \quad (3.8)$$

(5) 能量是灰度分布的均匀情况，用于表征图像纹理特征。灰度能量是灰度均值的平方和，其计算公式如公式 3.9 所示

$$E = \sum_{r=1}^R [p(r)]^2 \quad (3.9)$$

(6) 熵用于表示灰度矩阵的非均匀度，其计算公式如公式 3.10 所示。

$$T = - \sum_{r=1}^R [1 - p(r)] \cdot \lg[1 - p(r)] \quad (3.10)$$

令 $f_1 = \bar{g}$, $f_2 = \sigma^2$, $f_3 = S$, $f_4 = K$, $f_5 = E$, $f_6 = T$, 则将公式 3.11 作为特征向量。通过上述公式进行计算，得到的各典型故障示功图各特征值如表 3.1 所示。

$$F = \{f1, f2, f3, f4, f5\} \quad (3.11)$$

表 3.1 典型故障示功图各特征值
Table 3.1 Eigenvalues of typical fault indicator diagrams

故障类型	均值	方差	偏度	峰度	能量	熵
泵上碰	3.21045	27.92397	-0.38327	3.34437	0.05758	0.42149
泵下碰	2.58691	30.85084	-0.18718	2.60043	0.05156	0.42286
冲程损失影响	1.25488	37.16746	-0.52669	3.25434	0.05028	0.42314
抽油杆断脱	-2.34473	7.65753	-0.87165	3.03634	0.12416	0.4058
供液不足	-2.65771	55.01517	-0.71955	2.7464	0.04724	0.4238
固定凡尔漏失	3.54346	40.47956	-0.55768	3.51882	0.04968	0.42327
活塞遇卡	-9.3374	81.70598	-0.49271	2.05937	0.03823	0.42585
结蜡	5.0498	21.33639	0.15234	2.20084	0.06354	0.42013
连抽带喷	2.75977	20.27627	-0.09968	2.32852	0.06745	0.41919
气锁	-3.98047	54.04161	-0.63477	2.27464	0.04768	0.42371
气体影响	-0.45752	57.09292	-0.66501	2.88985	0.04775	0.42366
双凡尔同时漏失	3.09717	35.12874	-0.1024	2.57234	0.05038	0.4231
液积	1.81689	37.78825	-0.66526	3.75684	0.05286	0.42254
油稠	6.14795	23.84286	0.09955	2.22462	0.06013	0.42092
游动凡尔漏失	4.99512	25.27341	-0.13169	2.81686	0.05861	0.42127
油井出砂	3.4292	25.72546	-0.18157	2.85393	0.06197	0.42044
正常	4.28369	22.69442	-0.01029	2.41381	0.05997	0.42096
柱塞脱出工作筒	2.5752	40.71798	-0.73236	3.77729	0.0549	0.42205

3.6 灰色关联度与示功图诊断

灰色关联思想是通过两序列之间几何形状的相似程度来确定它们之间联系是否密切。关联度为事物间关联大小的度量，是对关联信息的量化。灰色关联度将事物发展趋势的相似度作为衡量事物之间关联程度大小的标准。

利用灰色关联度进行示功图诊断，将前一节提取的特征向量作为参考特征向量，待测示功图样本利用同样的方法提取特征向量。通过量化分析样本特征向量与参考特征向量变化趋势的相似度，得到样本示功图与各个示功图模板之间的关联度。

A型关联度反映了事物变化趋势的相似性。设参考特征向量和待测特征向量分别为 $x_i = \{x_i(1), x_i(2), \dots, x_i(N)\} (i = 1, 2, \dots, L)$, $x_j = \{x_j(1), x_j(2), \dots, x_j(N)\} (j = 1, 2, \dots, M)$, 那么 x_i 与 x_j 之间在第 k 处的灰色关联系数的计算公式如公式 3.12 所示。

$$\varepsilon_{ij}(k) = \frac{\min_i \min_k \Delta_{ij}(k) + \rho \max_j \max_k \Delta_{ij}(k)}{\Delta_{ij}(k) + \rho \max_j \max_k \Delta_{ij}(k)} \quad (3.12)$$

其中 $\Delta_{ij}(k) = |x_i(k) - x_j(k)|$ 为第 k 处特征值差的绝对值; ρ 为分辨系数, 取值在 0 到 1 之间, 一般取 0.5; $\max_j \max_k \Delta_{ij}(k)$ 和 $\min_i \min_k \Delta_{ij}(k)$ 分别为特征向量间所有特征元素差的绝对值的最大值和最小值。L、M、N 分别为参考特征向量个数、待测特征向量个数和特征元素个数。由关联系数定义 A 型关联度计算公式如公式 3.13 所示。其中 $r_{Aij} \in [0, 1]$ 。

$$r_{Aij} = \frac{1}{N} \sum_{k=1}^N \varepsilon_{ij}(k) \quad (3.13)$$

BO 型关联度反映了事物变化的相近性。B 型关联度计算公式如公式所示。

$$r_{Bji} = \frac{1}{1 + \frac{1}{N} d_{ij}^{(0)} + \frac{1}{N-1} d_{ij}^{(1)} + \frac{1}{N-2} d_{ij}^{(2)}} \quad (3.14)$$

其中 $r_{Bij} \in [0, 1]$, $d_{ij}^{(0)}$ 、 $d_{ij}^{(1)}$ 、 $d_{ij}^{(2)}$ 分别为零阶差商、一阶差商和二阶差商, 它们的计算公式分别为公式 3.15、公式 3.16、公式 3.17。

$$d_{ij}^{(0)} = \sum_{k=1}^N d_{ij}^{(0)}(k) = \sum_{k=1}^N |x_i(k) - x_j(k)| \quad (3.15)$$

$$d_{ij}^{(1)} = \sum_{k=1}^{N-1} d_{ij}^{(1)}(k) = \sum_{k=1}^{N-1} |x_i(k+1) - x_j(k+1) - x_i(k) + x_j(k)| \quad (3.16)$$

$$d_{ij}^{(2)} = \sum_{k=1}^{N-2} d_{ij}^{(2)}(k) = \frac{1}{2} \sum_{k=1}^{N-2} |x_i(k+1) - x_j(k+1) - 2[x_i(k) - x_j(k)]| \\ + \frac{1}{2} \sum_{k=1}^{N-2} |x_i(k-1) - x_j(k-1)| \quad (3.17)$$

ABO 型关联度结合 A 型关联度与 BO 型关联度, 兼顾相似性与相近性。由于 BO 型关联度计算比较复杂, 根据示功图诊断的实际需要, 对 BO 型关联度进行简化, 只保留零阶差商部分。将简化的 BO 型关联度与 A 型关联度相加, 得到 ABO

型关联度的计算公式如公式 3.18 所示。其中 a、b 为权系数，有 $a, b \in [0,1]$ ，且 $a + b = 1$ 。

$$r_{ABO} = a \frac{1}{N} \sum_{k=1}^N \varepsilon_{ij}(k) + b / (1 + \frac{1}{N} \sum_{k=1}^N \Delta_{ij}(k)) \quad (3.18)$$

利用此公式计算待检示功图特征序列与参考示功图特征序列之间的关联度。

由于特征向量中每个元素都有其各自的物理意义，在使用上述公式计算关联度前，需要对特征向量各元素进行无量纲化处理，使其具有可比较性。无量纲化处理可用公式 3.19 来实现。

$$y(k) = f(x(k)) = (x(k) - \min_k x(k)) / (\max_k x(k) - \min_k x(k)) \quad (3.19)$$

待测示功图曲线与示功图曲线模板之间的关联程度，由特征向量的关联度大小的排列次序来反映。将待测故障特征向量与标准参考特征向量的关联度值从大到小排序，就可以得到不同的故障对待测示功图的影响程度大小。当待测示功图所计算出的特征向量与某一参考特征向量之间的相关度最大^[31]，即可认为该示功图具有该类故障，最终达到示功图诊断的目的。

3.7 本章小结

本章首先对抽油机原理以及示功图的概念进行了简要的介绍。提出基于灰度矩阵的示功图诊断方法。算法以网格法为基础，应用模式识别理论。将示功图的灰度直方图的 6 个特征作为故障诊断的依据。本章中详述了灰度矩阵的提取过程，包括示功图标准化处理方法；基于 BFS 的联通区域划分；基于 DFS 的等高线划分方法。最后给出了特征向量提取以及关联度的计算方法，该算法是系统的功能核心。

第 4 章 智慧油田数据管理与故障诊断软件平台

智慧油田数据管理与故障诊断软件平台的核心功能是基于前文所述算法的故障诊断，功能模块可分为参数汇总、报表查询、曲线分析、作业区管理以及用户管理等 5 个核心模块。为了保证系统的稳定性，该系统基于 Java EE 架构开发。Java EE 架构由各类开源框架组合而成，Java EE 中自 MVC 层至持久层每一层都有多种开源框架提供完备的解决方案。因此利用 Java EE 所开发的应用往往具有高度的可扩展性和可维护性^[32]。Spring、Struts2、MyBatis、Hibernate 等都是 Java 领域常用的开发框架。本系统采用 Spring、Spring MVC、MyBatis 的框架，该框架已广泛应用于企业轻量级应用开发，技术成熟，性能稳定。利用 Spring 的 IOC 容器管理控制层、业务层以及持久层，可以充分发挥 Spring 依赖注入的优势，使得系统各层进一步解耦和，大大提高了应用系统的可扩展性，同时降低了开发成本。

4.1 关键技术介绍

4.1.1 数据库与 MyBatis 简介

面向对象编程语言是当下的主流，其优势包括多态、继承，简单易用。数据库的发展并未与语言同步，关系型数据库依然具有较大优势，比如大量数据查找、数据库并发访问以及事务等。

ORM（Object/Relation Mapping）即对象/关系型数据库映射，是为了应对面向对象语言与关系型数据库发展不同步而采取的中间解决方案。借助 ORM 框架实现对象关系型映射后，应用程序将以面向对象的方式来持久化对象，而不是直接访问数据库^[33]。这样开发者既享受到面向对象设计语言的简易性，同时也可利用关系型数据库的技术优势。ORM 框架工作示意图如图 4.1 所示。

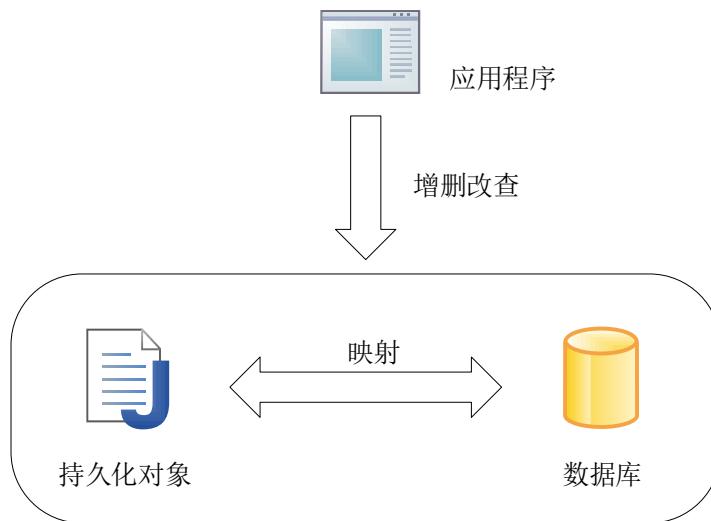


图 4.1 ORM 框架工作示意图

Fig. 4.1 Diagrammatic Sketch of ORM

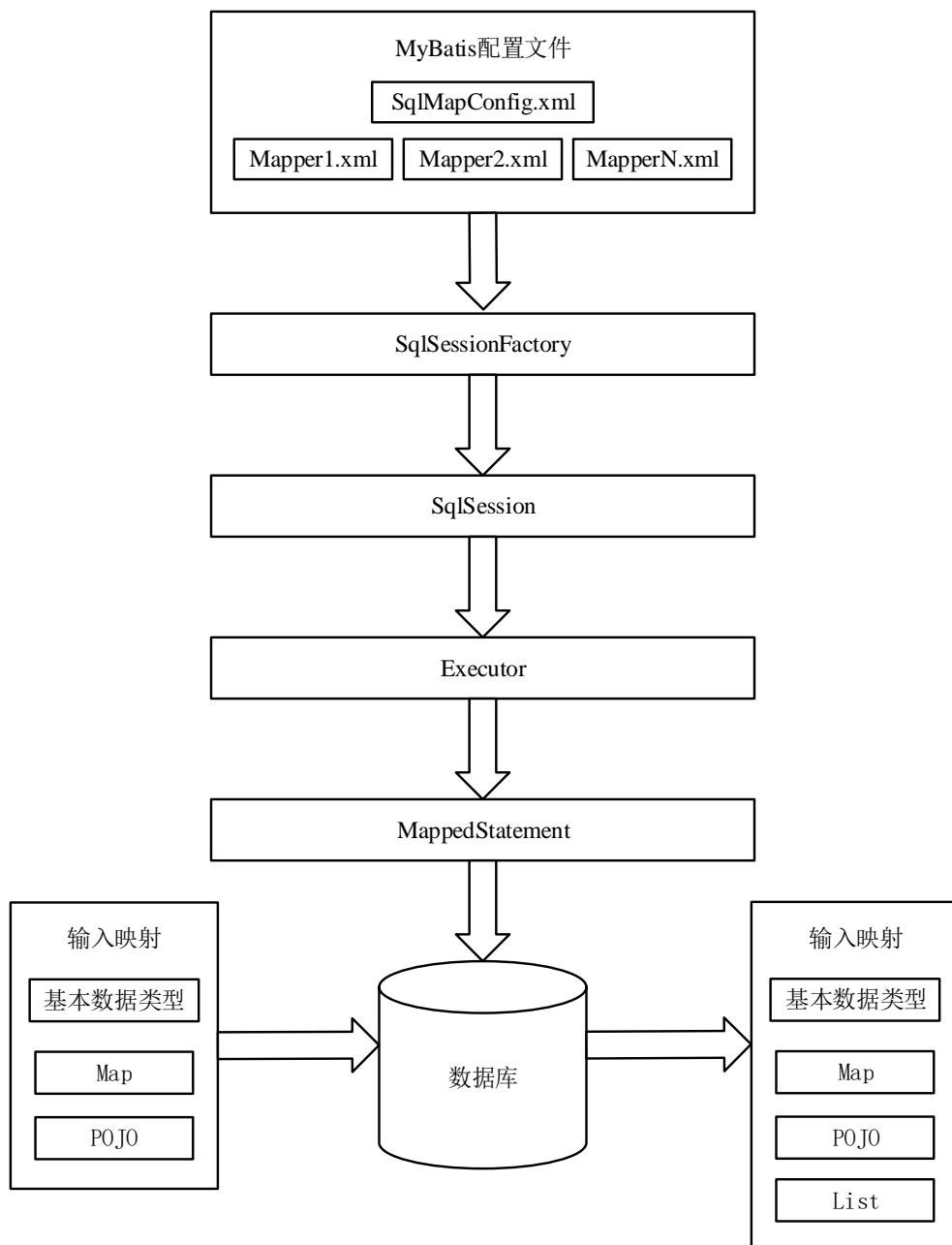
严格地说，数据库仅仅是存放数据的地方。数据库管理系统以数据为基础，为用户提供访问、操作数据库的接口。数据库管理系统的全称时 Database Management System，简称 DBMS。DBMS 是所有数据的知识库，负责管理数据的存储、安全、一致性、并发、恢复和访问等操作。通常所说的数据库，既包括用户存储的数据，也包括数据库管理系统。

在数据库的发展过程中，出现了网状型数据库、层次型数据库、关系型数据库和面向对象数据库 4 种类型的数据库系统。其中，关系型数据库是目前理论最为成熟，应用最为广泛的数据库。如今的关系型数据库在大数据的查找、排序上非常迅速，并且对数据库系统的并发、隔离有着非常完善的解决方案。MySQL 数据库非常小巧，操作方便，因此广泛应用于项目的开发阶段。

传统的 JDBC 操作数据的流程包括注册驱动、建立连接、创建 statement、手动设置参数等。对于一个 Web 系统，需要与数据进行频繁互动，系统每次操作数据库都需要重复编写上述流程的相关程序，不仅开发过程繁琐，而且严重影响系统的开发效率^[34]。

MyBatis 作为一个优秀的持久层框架，将传统 JDBC 操作进行了封装。这样开发者就可以将精力完全集中在 SQL 语句的设计上，而无需理会数据库的连接过程。MyBatis 通过配置的方式将 Java 的 POJO 类映射成数据库中的记录。相比于其他 ORM 框架， MyBatis 并不刻意契合 ORM 的思想，而是更加专注于简洁方便地数

据库访问方式，以减少开发人员的工作量。MyBatis 架构中可以配置大量的 SQL 语句，增加了系统的灵活性。MyBatis 程序架构如图 4.2 所示。其中 Sql Session 是 MyBatis 的顶层 API 接口，作为会话访问，完成 CRUD 的功能；Executor 是 MyBatis 的核心，用于 SQL 动态语句生成和查询缓存的维护。



4.2 MyBatis 程序架构图
Fig. 4.2 Program Architecture Diagram of MyBatis

4.1.2 Spring MVC 简介

Spring 是一个一站式 Java EE 开发框架，它由 Rod Johnson 创建。由于互联网的崛起，各行各业的公司都需要开发自己的企业级应用。Spring 的出现大大简化了企业级应用的开发。

Spring 基于依赖注入的核心机制，使得各组件耦合性大大降低。Spring 中控制反转（IoC）即是依赖注入思想，依赖对象无需主动初始化被依赖对象，只需要声明被依赖对象，然后被动接受 Spring 容器所赋予的初值即可。当应用中某一组件需要更新或更换时，只需对这一组件进行修改，应用其他组件中所有声明该组件都会被初始化为新版本。如此大大降低了各组件之间的耦合性，提高了系统整体的可维护性。

Spring 支持基于面向切面编程的声明式事务管理。通过使用声明式事务管理，开发者无需在 Java 代码中书写相关事务管理代码，只需要通过 XML 文件配置 AOP 代理即可。这种方式可以让应用在不同的事务策略之间进行灵活的切换，同样无需修改 Java 代码，只要修改配置文件。面向切面编程，提高了程序的复用性，大大减少了应用系统的代码量；更换事务策时应用程序无需任何更改，这种低侵入式的编程方式也避免了代码污染；同时，应用开发过程中无需考虑事务处理，能够更加专注于业务逻辑本身的开发。

此外作为一站式开源框架，Spring 本身提供了从持久层到表现层的各组件解决方案，而不局限于专注某一层组件的解决方案。同时它并不强制使用自身的解决方案，可以与已有框架进行完美整合，高度的开放性使 Spring 框架具有了强大的生命力^[35]。图 4.3 为 Spring 框架结构。其中 Spring Core Container 是 Spring 的核心由 org.springframework.beans、org.springframework.core、org.springframework.context、org.springframework.expression 四个包组成。在使用 Spring 框架时，这四个包缺一不可。

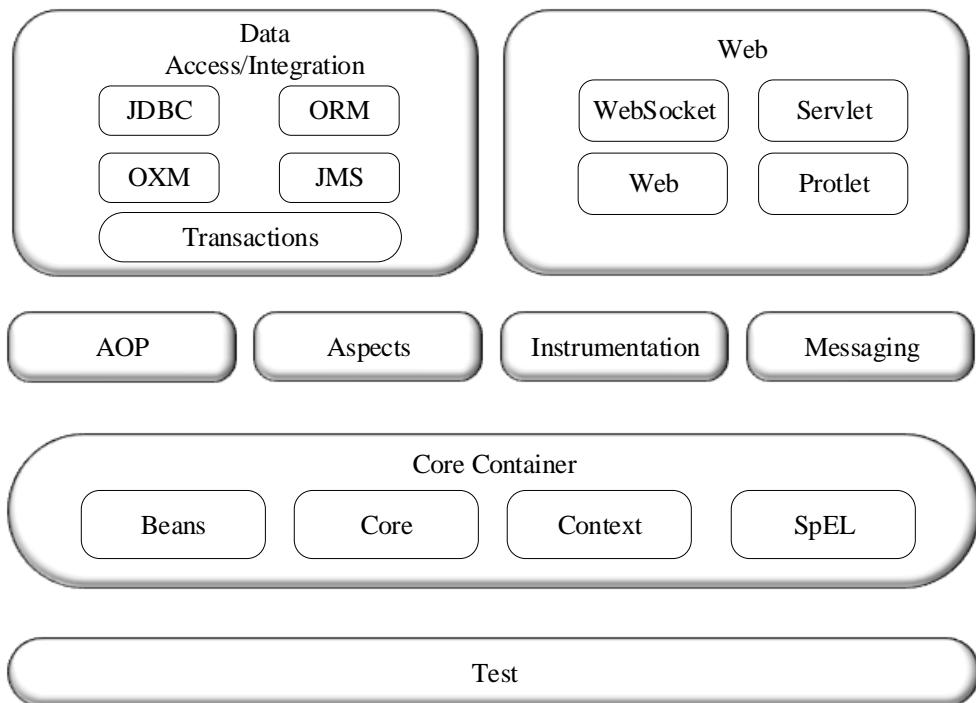


图 4.3 Spring 架构结构图
Fig. 4.3 Diagram of Spring Architecture

MVC 思想时所有面向对象程序语言都具备的规范，并不是 Java 所特有的思想。MVC 即模型（Model）、视图（View）、控制（Controller）。系统应用被分为以上三层，大大降低了系统应用内部的耦合性，方便了系统应用日后的更新和扩展。MVC 思想类似于设计模式中的观察者模式，MVC 应用会为每个模型注册视图，当某个模型发生改变时，将向所有已注册的视图下发通知。然后视图会根据模型所下发的通知进行更新。一个模型可以拥有多个视图，目的是减少程序代码量，同时提高系统的可维护性。应用被分为三层，而且层与层之间的代码联系有限，提高了系统的可扩展性。此外，MVC 思想更符合工程化管理特征，每层指责分工明确，便于管理。遵循 MVC 模式的 Java Web 运行流程图如图 4.4 所示。

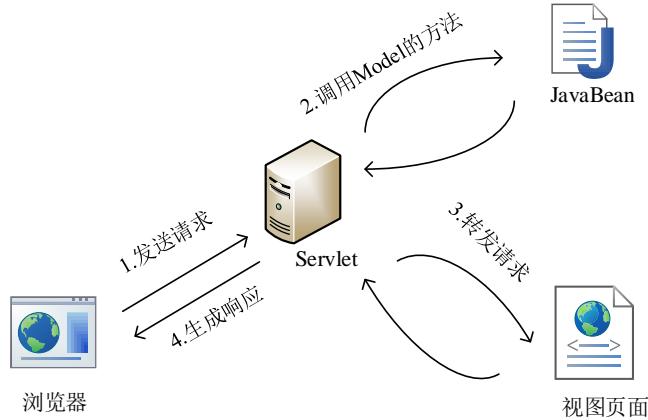


图 4.4 MVC 运行流程图
Fig. 4.4 Running flow chart of MVC

Spring MVC 是 Spring 框架提供的构建 Web 应用的全功能 MVC 模块。SpringMVC 提供了前端控制器 DispatcherServlet 用于派发请求，无需开发者额外开发控制器；Spring MVC 支持多种视图技术，最常见的是 JSP，Spring 提供了强大的 JSP 标签库，降低了 JSP 编写难度；Spring MVC 分离了控制器、模型对象、分派器以及处理对象；Spring MVC 拥有很好的灵活性、非侵入性和可配置性。

图 4.5 是 Spring MVC 程序基本执行流程图。

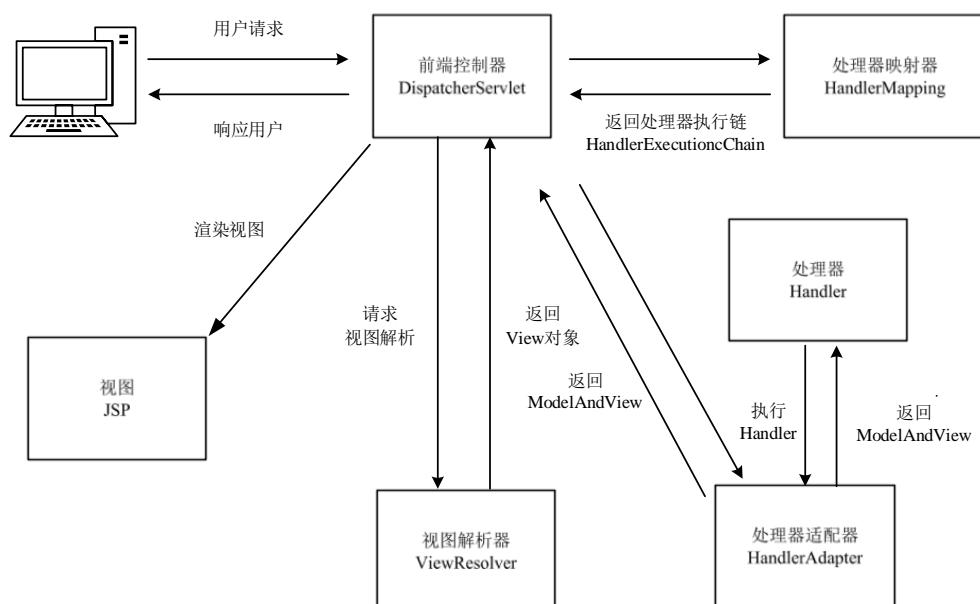


图 4.5 Spring MVC 程序流程图
Fig. 4.5 Program Flow Diagram of Spring MVC

具体执行流程如下：

- (1) 用户通过 url 发送请求。
- (2) 前端控制器 Dispatcher Servlet 接收到请求，通过处理器映射器配置找到对应的 Handler，并将 url 映射的控制器 Controller 返回给核心控制器。
- (3) 通过前端控制器找到相应的处理器适配器。
- (4) 由找到的处理器适配器，调用实现对应接口的处理器，并将结果返回给适配器，结果中包含数据模型和视图对象，再由适配器返回给前端控制器。
- (5) 前端控制器将获取的数据和视图结合的对象传递给视图解析器，获取解析得到的结果，并由视图解析器响应给前端控制器。
- (6) 前端控制器将结果返回给客户端

4.2 软件平台总体功能设计

智慧油田数据管理与故障诊断软件平台以故障诊断功能为核心功能；对井场操作区以及其中的单井进行管理；借助单井上部署的远程终端单元对单井运行参数进行持久化。此外，作为一款应用软件，从功能性上的角度考虑，需要将数据全方位、立体化地呈现给用户；从非功能性的角度上考虑，要求软件系统具有较强的稳定性、易用性以及可扩展性；从软件程序设计的角度考虑，在满足基础业务逻辑的基础之上，要求代码书写规范，逻辑清晰，简洁优雅。

根据油田生产的需求，为软件平台五个功能模块。这五个功能模块分别为用户及权限管理模块、参数汇总模块、报表查询模块、曲线分析模块以及作业区管理模块。软件平台整体功能模块图如图 4.6 所示。

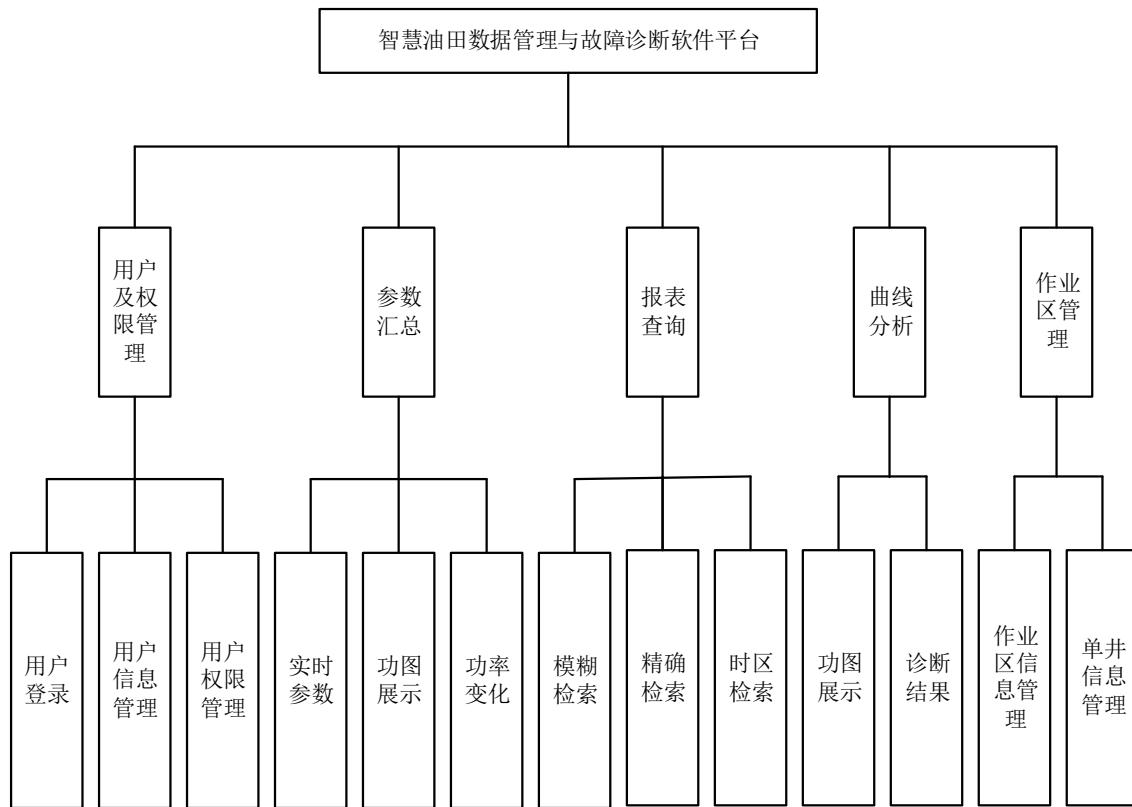


图 4.6 软件平台整体功能模块图

Fig. 4.6 Overall Function Module Diagram of Software Platform

用户及权限管理模块主要包括用户登录、用户信息管理以用户权限校验及用户权限管理等功能。其中用户登录可对用户登录信息进行校验；提供记住密码功能，在安全的使用环境下简化用户登录操作；拦截未登录用户的一切请求，提高整个系统的安全性。权限管理功能，将用户划分为三个等级，每个等级所拥有的权限，可由管理员按照实际需要进行设置。用户信息管理模块对用户信息进行增删改查操作。普通用户只能修改自身基本信息和密码，拥有权限的用户可以对用户进行管理，包括新增或删除用户以及修改系统内所有用户的信息。

参数汇总模块直观展示了某一单井的实时信息，功能主要包括实时参数展示、最新示功图展示以及最近 1 天功率（包括有功功率和无功功率）变化曲线，检索方式分为手动输入和通过作业区选择两种方式。

报表查询模块，是对某一单井某一时段的参数汇总，参数包括井口油压、冲程、冲次、最大载荷、最小载荷、A 相电流、B 相电流、C 相电流、A 相电压、B 相电压、C 相电压、有功功率、无功功率、功率因数等，支持精确检索和模糊检索。

曲线分析模块借助前一章节所介绍的算法对示功图进行故障诊断，计算出的特征向量值、与各种故障的关联度并最终给出诊断结果。

作业区管理模块功能用于作业区信息管理以及单井信息管理。作业区信息管理可以查询所有作业区信息并提供增加、修改、删除功能；单井信息管理可查询某一作业区内所有单井的基本信息，并提供增加、修改、删除功能。

4.3 软件平台系统架构分析

程序设计方面，系统采用 Spring、Spring MVC、MyBatis 架构实现，按照经典 Java EE 设计对系统进行结构划分：

表现层：利用 Bootstrap 快速搭建，由一系列的 JSP 页面组成。

控制器层：使用 Spring MVC 框架。

业务逻辑层：由业务逻辑组件构成，利用 Spring 的 IoC 容器进行管理。

DAO 持久层：由 DAO 组件组成，使用 MyBatis 技术实现 ORM 映射。

领域对象层：由 Domain Object 组件组成。

数据库服务层：使用 MySQL 数据库存储持久化数据。

层次清晰，模式简单、直接，便于实际开发。软件平台系统架构如图 4.7 所示。

其中持久层数据库中，定义了 role、user、operation_area、well_info、well_data 等 5 个物理表，用于将系统数据进行持久化。

领域对象层，根据系统需求提取出系统中的对象，再将这些对象抽象为类，其中，需要进行持久化保存的类成为持久化对象（PO）。

本系统共设计 5 个持久化类，具体包括：

User。对应用户表，包括用户名、功号、登录密码等属性。

Role。对应角色表，包括不同角色各自的权限。

OperationArea。对应作业区表，包括作业区名称、地址、联系人、联系电话等属性。

WellInfo。对应单井基本信息表，包括所属作业区、单井 ID、井号、状态、联系人、联系电话等属性。

WellData。对应单井数据表，包括单井 ID、井口油压、冲程、冲次、最大载荷、最小载荷、A 相电流、B 相电流、C 相电流、A 相电压、B 相电压、C 相电压、有功功率、无功功率、功率因数、功图中 200 个点载荷以及对应的 200 个点位移等属性。

前端层	JSP+BootStrap			
拦截器	BrowseInterceptor LoginInterceptor OperationareaManaInterceptor UserManaInterceptor			
控制器层	WellInfoController UserController OperationAreaController WellDataController RoleController			
业务逻辑层	WellInfoService UserService OperationAreaService WellDataService RoleService			
DAO象层	WellInfoMapper UserMapper OperationAreaMapper WellDataMapper RoleMapper			
领域对象层	IndicatorDiagramAnalysisResult OperationArea PageBean Position WellInfo User UserPageBean WellData Role			
持久层	数据库			

图 4.7 软件平台系统架构图

Fig. 4.7 System Architecture Diagram of Software Platform

以上类中部分对象存在关联关系：Role 和 User 存在多对一关系，即一种角色下可以存在多名用户，但是每名用户只允许拥有一种角色。OperationArea 和 WellInfo 存在一对多关系，即一个作业区可以包含多口单井，但是每口单井只能属于某一作业区。WellInfo 和 WellData 存在一对多关系，即一口单井会有不同采集时间的多组数据参数，而每组数据参数只属于某一单井。

MyBatis 通过定义接口来完成 SQL 语句的映射，该接口可直接作为 DAO 组件来使用。DAO 组件只负责实现对数据库的基本的 CRUD 操作，是业务逻辑实现的基础。

本系统中设计了 RoleMapper、UserMapper、OperationAreaMapper、WellInfoMapper、WellDataMapper 等 5 个 DAO 组件，接口中定义了数据的各种增

删改查方式，以及满足特殊功能的连表查询方式，用来提供不同业务模块的逻辑需求，同时通过 XML 配置了对应的 SQL 语句映射。

DAO 组件设计完成后，需要对 Spring 与 MyBatis 进行整合。整合的目的是让 Spring IoC 容器管理 DAO 组件，方便 Service 组件进行依赖注入。为此，MyBatis 社区开发了 Spring-MyBatis 中间件来满足需求，通过配置 SqlSessionFactoryBean 即可实现将 SqlSessionFactory 纳入 Spring IoC 容器管理。在 Spring 管理 DAO 组件之前，必须要提供对应的数据源，本系统使用 C3P0 数据源。

业务逻辑组层由业务逻辑组件构成，业务逻辑组件的构成相当于对 DAO 组件进行封装，换句话说，业务逻辑组件依赖于 DAO 组件。本系统设计 RoleService、UserService、OperationAreaService、WellInfoService、WellDataService 等 5 个业务逻辑组件，用于实现对不同主体操作的业务逻辑。

业务逻辑组件设计完成后，同样需要将管理权限交给 Spring IoC 容器。本系统中通过注解@Service 以及配置包扫描的方式实现。

控制器层和前端层可合并为一层并称之为 Web 层。Web 层的设计主要包括控制器的设计以及 JSP 页面的设计。JSP 页面发送请求，控制器根据接收到的请求调用相关的业务逻辑组件。所以通常而言，这两部分是同时进行开发的。从另一个角度上说，控制器接到 JSP 发送的请求后并不会亲自处理请求，只负责调用业务逻辑组件来处理请求。因此控制器可以看作是 JSP 与业务逻辑组件之间桥梁。此处的控制器指的是核心控制器 DispatcherServlet 和业务控制器 Controller。控制器层与前端层之间添加 BrowseInterceptor、LoginInterceptor、OperationareaManaInterceptor、UserManaInterceptor 等 4 个拦截器。其中 UserManaInterceptor 拦截器用于鉴别发出请求的用户是否为已登录状态，并拦截一切非法请求访问；BrowseInterceptor、LoginInterceptor、OperationareaManaInterceptor 用于进行用户权限校验。

4.4 数据库设计

从系统需求和功能出发，设计数据库物理表。用户及权限管理模块包括用户管理和权限管理。其中，用户管理模块，涉及新增用户，删除用户，修改用户信息，查询用户信息以及校验用户名密码是否配等功能；权限管理模块可设置不同级别的用户拥有的具体权限，同时需要查询不同级别用户所拥有的权限，作为请求拦截的条件。

参数汇总模块主要执行的是查询操作，查询内容包括井号以及该单井的实时参数即最近一条参数数据、最近一张示功图以及当天电功率的变化曲线。井号属于单井的基本信息，而其他查询内容均属于单井数据。报表查询通过井号和时间两个限定条件来查询单井数据，需要支持精确查询以及模糊查询。

曲线分析模块对示功图进行分析，进而给出抽油机故障诊断结果，此功能以查询示功图数据为基础。

作业区管理模块主要包括作业区信息管理和单井信息管理。作业区信息管理包括新增作业区、删除作业区，修改作业区信息，查询作业区等功能。需要注意的是，删除作业区需要级联操作。即删除某一作业区，会同时删除作业区内所有单井信息。而单井信息管理包括新增单井、删除单井，修改单井信息，根据所属作业区查询单井等功能。与作业区信息管理相同，删除单井信息也需要级联操作，即删除某一单井信息，会同时时删除该单井所有采集数据。

根据上述功能逻辑分析，数据库中定义了 role、user、operation_area、well_info、well_data 等 5 个物理表分别用于存储角色信息、用户信息、作业区信息、单井信息以及单井数据。各个物理表中定义的字段名称、字段类型、约束条件及备注由下表给出。

表 4.1 用户表
Table 4.1 User Table

字段名	字段类型	约束条件	是否为空	备注信息
id	INT(11)	PRIMARY KEY AUTO_INCREMENT	NOT NULL	
username	VARCHAR(32)	UNIQUE KEY	NOT NULL	用户名
password	VARCHAR(256)		NOT NULL	用户密码
name	VARCHAR(10)		NOT NULL	姓名
pid	VARCHAR(13)		NOT NULL	工号
phone	VARCHAR(20)		NOT NULL	手机
role	INT(2)	FOREIGN KEY	NOT NULL	角色
createdate	DATETIME			创建时间

表 4.2 角色表
Table 4.2 Role Table

字段名	字段类型	约束条件	是否为空	备注信息
id	INT(2)	PRIMARY KEY AUTO_INCREMENT	NOT NULL	
browse	VARCHAR(2)		NOT NULL	浏览信息
operationareamana	VARCHAR(2)		NOT NULL	作业区管理
usermana	VARCHAR(2)		NOT NULL	用户及权限管理

表 4.3 作业区表
Table 4.3 Operation Area Table

字段名	字段类型	约束条件	是否为空	备注信息
id	INT(11)	PRIMARY KEY AUTO_INCREMENT	NOT NULL	
area_id	VARCHAR(11)	UNIQUE KEY	NOT NULL	作业区 ID
name	VARCHAR(32)	UNIQUE KEY	NOT NULL	作业区名称
address	VARCHAR(256)		NOT NULL	地址
linkman	VARCHAR(32)		NOT NULL	联系人
phone	VARCHAR(32)		NOT NULL	联系电话
note	VARCHAR(256)		NOT NULL	备注

表 4.4 单井信息表
Table 4.3 Well Information Table

字段名	字段类型	约束条件	是否为空	备注信息
id	INT(11)	PRIMARY KEY AUTO_INCREMENT	NOT NULL	
well_id	INT(11)	FOREIGN KEY	NOT NULL	单井 ID
oil_pressure	DOUBLE(6,2)		NOT NULL	井口油压
stroke	DOUBLE(6,2)		NOT NULL	冲程
times	DOUBLE(6,2)		NOT NULL	冲次

表 4.4 单井信息表（续）

Table 4.3 Well Information Table (Cont)

字段名	字段类型	约束条件	是否为空	备注信息
load_max	DOUBLE(6,2)		NOT NULL	最大载荷
load_min	DOUBLE(6,2)		NOT NULL	最小载荷
ac	DOUBLE(6,2)		NOT NULL	A 相电流
bc	DOUBLE(6,2)		NOT NULL	B 相电流
cc	DOUBLE(6,2)		NOT NULL	C 相电流
av	DOUBLE(6,2)		NOT NULL	A 相电压
bv	DOUBLE(6,2)		NOT NULL	B 相电压
cv	DOUBLE(6,2)		NOT NULL	C 相电压
active_power	DOUBLE(6,2)		NOT NULL	有功功率
reactive_power	DOUBLE(6,2)		NOT NULL	无功功率
power_factor	DOUBLE(6,2)		NOT NULL	功率因数
displacement	VARCHAR(2000)		NOT NULL	200 个位移点
load200	VARCHAR(2000)		NOT NULL	200 个载荷点
current200	VARCHAR(2000)		NOT NULL	200 个电流点
collection_date	DATETIME		NOT NULL	采集时间

表 4.5 单井数据表

Table 4.5 Well Data Table

字段名	字段类型	约束条件	是否为空	备注信息
id	INT(11)	PRIMARY KEY AUTO_INCREMENT	NOT NULL	
well_id	INT(11)	FOREIGN KEY	NOT NULL	单井 ID
oil_pressure	DOUBLE(6,2)		NOT NULL	井口油压
stroke	DOUBLE(6,2)		NOT NULL	冲程
times	DOUBLE(6,2)		NOT NULL	冲次
load_max	DOUBLE(6,2)		NOT NULL	最大载荷
load_min	DOUBLE(6,2)		NOT NULL	最小载荷

表 4.5 单井数据表（续）
Table 4.5 Well Data Table(Cont)

字段名	字段类型	约束条件	是否为空	备注信息
ac	DOUBLE(6,2)		NOT NULL	A 相电流
bc	DOUBLE(6,2)		NOT NULL	B 相电流
cc	DOUBLE(6,2)		NOT NULL	C 相电流
av	DOUBLE(6,2)		NOT NULL	A 相电压
bv	DOUBLE(6,2)		NOT NULL	B 相电压
cv	DOUBLE(6,2)		NOT NULL	C 相电压
active_power	DOUBLE(6,2)		NOT NULL	有功功率
reactive_power	DOUBLE(6,2)		NOT NULL	无功功率
power_factor	DOUBLE(6,2)		NOT NULL	功率因数
displacement	VARCHAR(2000)		NOT NULL	200 个位移
load200	VARCHAR(2000)		NOT NULL	200 个载荷
current200	VARCHAR(2000)		NOT NULL	200 个电流
collection_date	DATETIME		NOT NULL	采集时间

为了方便系统以后的更新与修改，在这 5 个数据表中均单独设置了没有实际含义的、与数据和逻辑无关的主键。考虑到数据的物理意义与软件系统的具体需求，为了减少数据冗余、保证数据的一致性以及实现级联操作，上述表格通过外键建立关联。具体关联关系如图 4.8 所示。角色表与用户表存在一对多关系；作业区信息表与单井信息表存在一对多关系；单井信息表与单井数据表存在一对多关系。

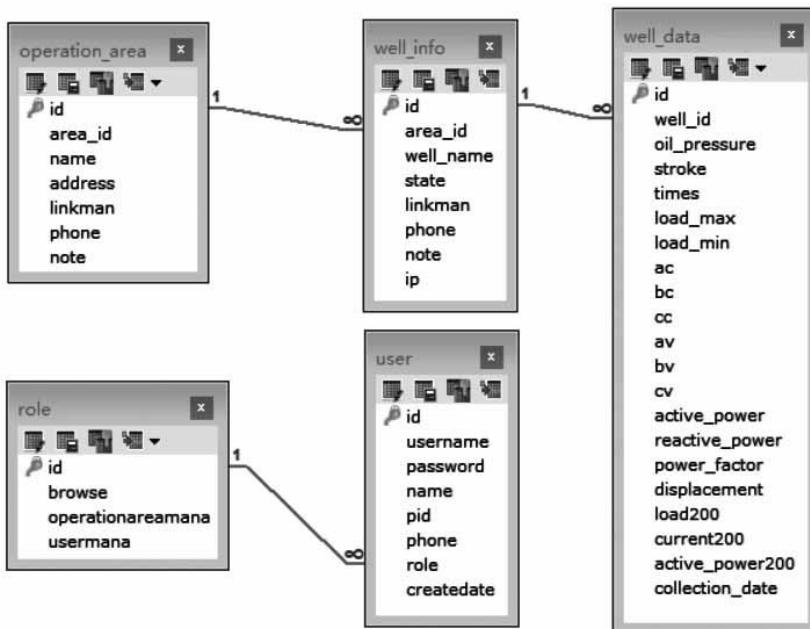


图 4.8 数据库物理表关联图

Fig. 4.8 Association Diagram of Database Physical Table

4.5 用户及权限管理模块设计

4.5.1 用户及权限管理流程设计

用户及权限管理模块提供用户登录、用户管理、权限校验、权限管理等功能。下面从流程设计的角度对该模块的功能进行详细阐述。

(1) 用户管理

用户管理可分为个人信息管理和对系统中所有用户的管理。所有用户均可在个人信息管理中修改自己的用户名、姓名、工号、电话等基本信息以及登录密码。对于所有用户的管理需要使用该功能的用户拥有权限，此功能用于新增、修改或删除用户以及检索用户。检索方式包括按姓名检索以及按工号检索，两种检索方式均兼容精确查询和模糊查询。考虑到用户管理功能流程相对简单且类似，这里仅给出新增用户的业务流程图。如图 4.9 所示。

(2) 权限管理

系统用户权限有三种，分别为浏览权限、作业区管理权限以及权限管理权限，与之相对应，将用户权限划分成三个等级，系统初始状态下，1 级用户拥有所有权

限，2 级用户拥有浏览权限和作业区管理权限，而 3 及用户仅拥有浏览权限。1 级用户拥有权限管理权限，可修改不同等级的用户所拥有的权限。

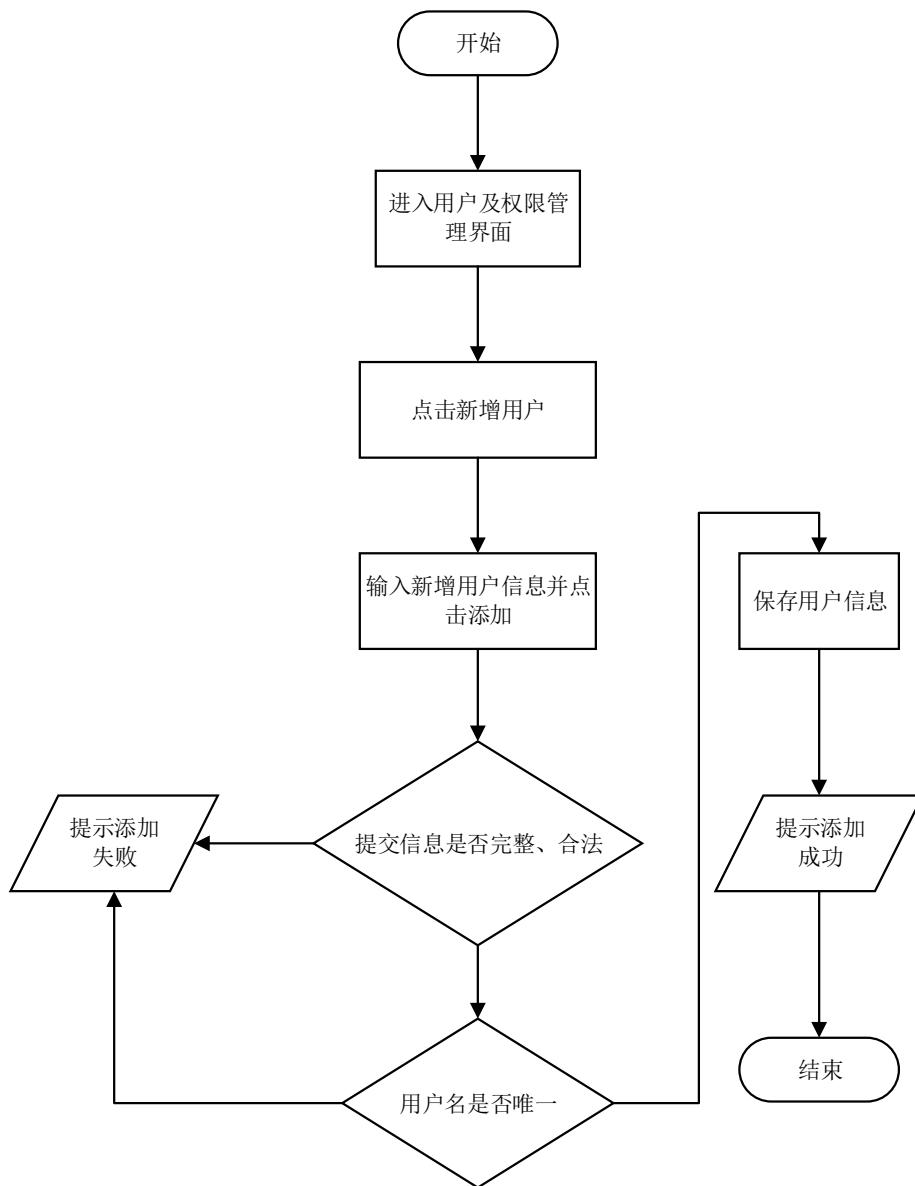


图 4.9 新增用户业务流程图
Fig. 4.9 Business Flow Chart of Add User

4.5.2 用户及权限管理程序设计

用户及权限管理模块核心类包括 UserController 类、UserService 接口、 UserServiceImpl 类、UserMapper 接口、RoleController 类、RoleService 接口、

RoleServiceImpl 类以及 RoleMapper 接口。用户及权限管理模块 UML 类图如图 4.10 所示。

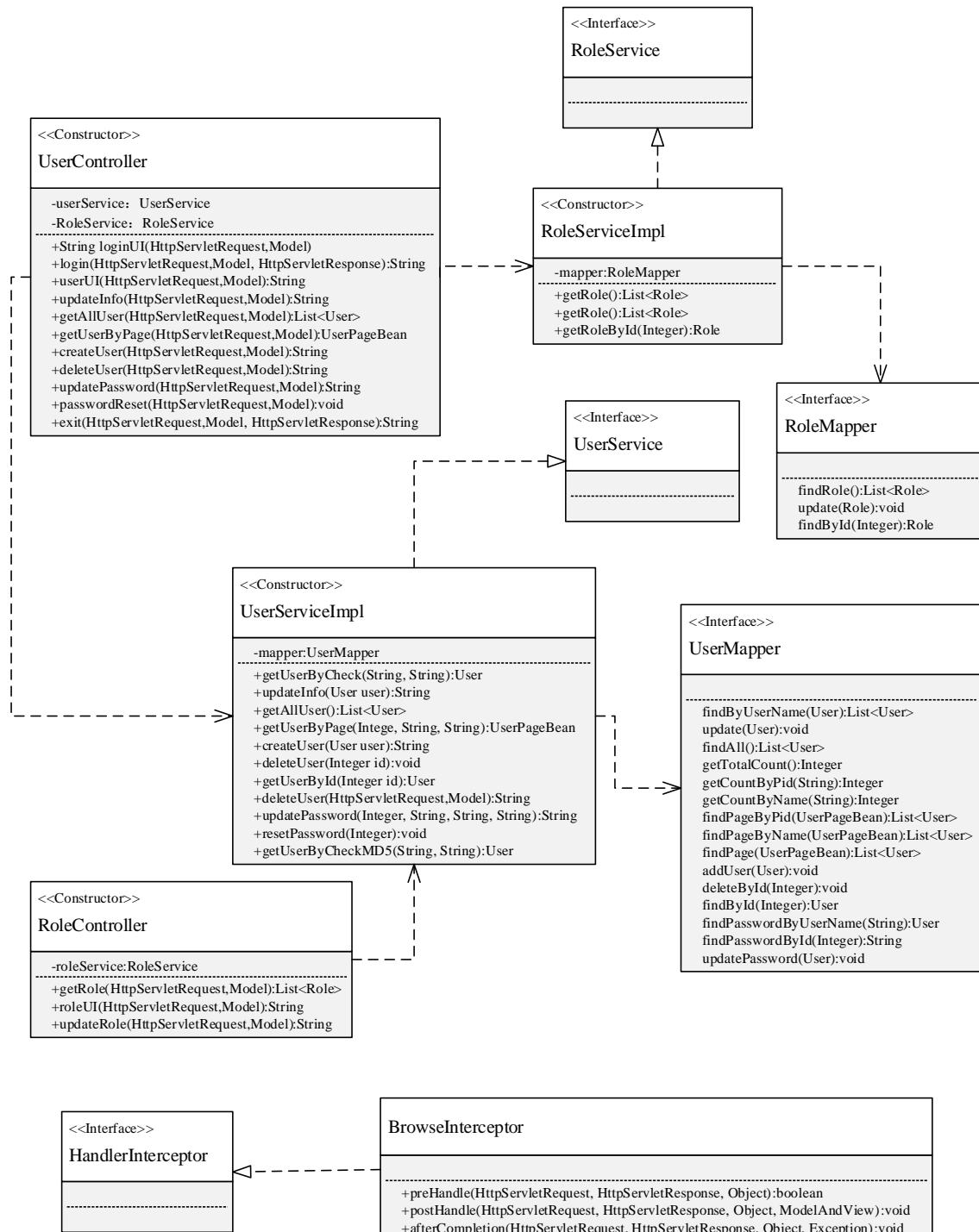


图 4.10 用户及权限管理模块核心类图

Fig. 4.10 Class Diagram of User Management and Role Management Model

UserController 类和 RoleController 类用于处理与用户管理和权限相关的请求，通过 login()、createUser()、getUserByPage() 等方法来实现对请求的处理。

其中 login() 方法用于处理用户登录请求。前端请求参数中，除了基本的用户名、密码等信息，还包含了记住密码的状态信息。在校验用户名密码成功后，会判断是否记住密码。

记住密码功能通过 Cookie 实现，Cookie 是由服务器放置到客户端的少量信息，实质上是一组键值对。新建包含用户信息的 Cookie 并设置其有效时长，将这个 Cookie 通过 HttpServletResponse 的 addCookie() 方法放置到客户端，当用在 Cookie 的有效时长内登录该系统，用于登录的拦截器将提取该 Cookie 的登录信息，若信息校验通过，将直接放行用户请求，不需要重新登录。

在系统调试过程中，可以使用 Chrome 浏览器，利用开发者工具的记录网络日志功能，可以查看客户端所添加的 Cookie，如图 4.11 所示。

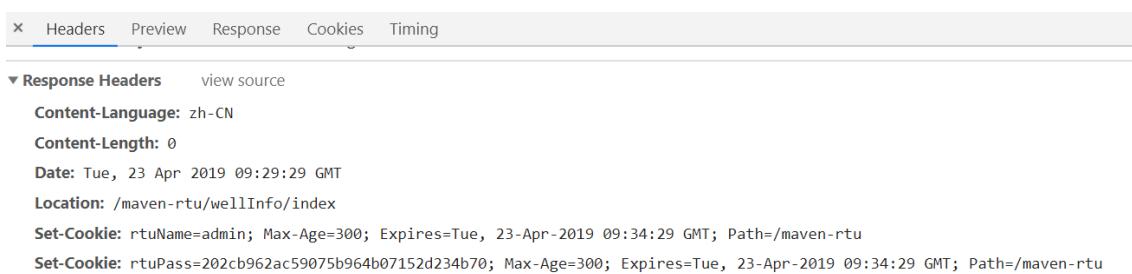


图 4.11 响应头信息

Fig. 4.11 Response Headers

exit() 方法用于注销用户，执行该方法，新建同名 Cookie 并设置有效时长为 0，将该 Cookie 通过 HttpServletResponse 的 addCookie() 方法放置到客户端，会替换客户端原有 Cookie，实现取消自动登录功能。

UserServiceImpl 类和 RoleServiceImpl 类负责用户管理与权限管理相关的业务逻辑实现。为了保护用户隐私，提升密码安全度， UserServiceImpl 类中所有关于密码的处理，均采用 MD5 加密技术。

UserServiceImpl 中的 getUserByPage(Integer, String, String) 方法用于通过用户名或工号检索用户。考虑到检索结果可能较多，为了方便用户查看，采取分页方式进行查询。为此，以 User 类为基础，在领域对象层新建一 UserPageBean 类，用于封装分页操作所需要的信息。UserServiceImpl 首先检索出满足条件的数据总数，根

据每页的数据量计算出总页数；然后通过当前页数以及每页的数据量，调用 Dao 层分页查询方法来查询当前页所有用户信息；将上述查询信息通过 setter 方法为 UserPageBean 的属性赋值并将该实体返回给调用该方法的 Controller 类。用户分页查询前端界面效果如图 4.12 所示。

用户管理							
		按姓名检索		按工号检索	重置	新增用户	
用户ID	用户名	姓名	工号	电话	角色	创建时间	操作
1	admin	张三	001	13245678902	1	2018-05-03 23:16:57.0	修改 删除
2	user1	李四	002	13245678901	2	2018-05-03 23:16:57.0	修改 删除
3	user4	王五	003	13245678901	2	2018-05-03 23:16:57.0	修改 删除
4	user3	赵六	004	1861113344	2	2018-05-03 23:16:57.0	修改 删除
7	user6	李白	007	13223223232	2	2018-05-03 23:16:57.0	修改 删除

图 4.12 分页查询界面

Fig. 4.12 Paging Query Interface

UserMapper 接口和 RoleMapper 接口处于 DAO 层，用于定义完成 SQL 语句的映射。业务逻辑层是 DAO 组件的封装，这样可以分离业务层与 DAO 层的功能：业务逻辑组件仅专注于实现业务逻辑，而 DAO 组件负责持久化技术的变化。

4.5.3 用户及权限管理时序设计

用户及权限管理模块主要用于管理用户和用户所拥有的权限。更新用户信息在该模块中是一种比较常用的功能。故以此功能为例，来介绍用户及权限管理时序设计。

Spring MVC 中的 DispatcherServlet 在接收到用户发送的查看用户信息的请求后，拦截器将校验该用户是否拥有修改信息的权限。若校验通过，则将该请求派发给 UserController 类。UserController 类调用其 getUserByPage()方法来处理该请求。通过 UserServiceImpl 和 UserMapper 的交互，最终得到 UserPageBean 实体并将其返回。前端界面借助 Ajax 将 UserPageBean 内的信息异步加载，呈现给用户。

当 DispatcherServlet 接收到用户发出的修改信息请求时，同样会通过拦截器权限校验，将该请求派发给 UserController。UserController 类调用其 getUserById()方法，该方法调用了 UserServiceImpl 的同名方法，并通过 UserServiceImpl 与

UserMapper 的交互返回指定 id 的 user 实体。前端界面将该实体中的信息异步加载至相应的文本框，等待用户修改信息。

用户点击保存按钮提交修改信息，DispatcherServlet 将请求派发给 UserController。UserController 中的 updateInfo()方法调用 UserServiceImpl 的同名方法，首先校验更新的用户信息的合法性，若信息不合法，将返回错误提示信息；若信息合法，则会通过 UserMapper 将该信息更新并返回更新成功信息。前端部分通过 Javascript 代码的相关函数来判断返回信息，并对不同信息做出对应的响应，包括更新提示框颜色、文本信息等。图 4.13 为修改用户信息时序图。

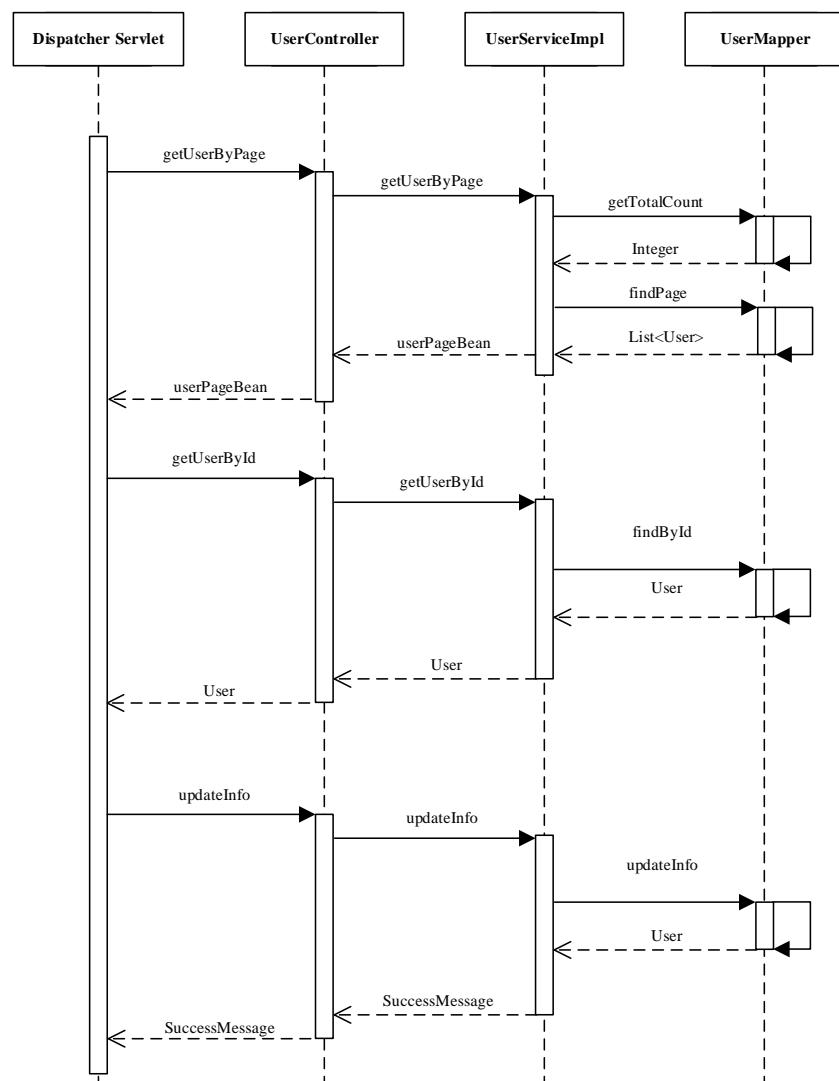


图 4.13 修改用户信息时序图

Fig. 4.13 Sequence Chart of Updating User Infomation

4.6 参数汇总模块设计

4.6.1 参数汇总流程设计

参数汇总模块，用于查看实时参数的汇总信息，可全方位了解某一单井的运行状态。该模块仅提供多种方式的查询功能，流程相对简单。参数汇总执行业务流程图如图 4.14 所示。进入参数汇总界面后，输入要查看的单井井号，点击查询。若井号存在，则显示该单井最新一组参数信息；若该井号不存在，则显示数据库中存储的最新一组参数信息，而不再提示错误信息。

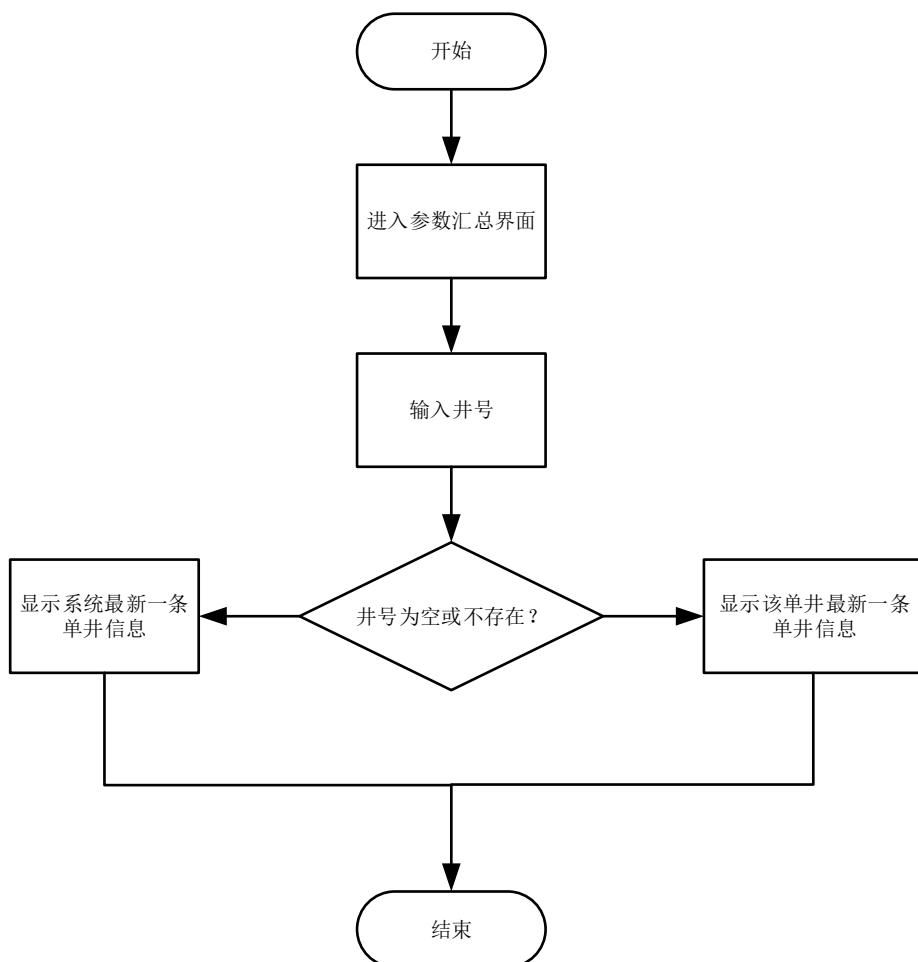


图 4.14 参数汇总业务流程图

Fig. 4.14 Business Flow Chart of Parameters Summary

4.6.2 参数汇总程序设计

参数汇总模块核心类包括 WellInfoController 类、OperationAreaController 类、WellInfoService 接口、WellDataService 接口、WellInfoServiceImpl 类、WellDataServiceImpl 类、WellInfoMapper 接口以及 WellDataMapper 接口。参数汇总模块类图如图 4.15 所示。

WellInfoController 类用于处理与单井信息相关的请求。参数汇总是单井数据的汇总，但同时需要显示单井井号等属于单井信息的参数，故将参数汇总相关请求交给 WellInfoController 类来处理，由其中的 getIndexData() 方法负责实现。OperationAreaController 类在此功能中仅负责响应查询系统中所有作业区的请求。

业务逻辑层中，WellDataServiceImpl 组件用于收集参数汇总所需信息并将其封装返回给 WellInfoController。

WellInfoServiceImpl 组件则是为了实现作业区-单井的联动查询，方便用户根据作业区精确查找某一单井。联动查询由前端 JSP 发起，页面加载完成后查询所有作业区并将它们添加到作业区选择框；对作业区选择框进行 onChange 动作监听，当用户选择某一作业区后，会发送带作业区 id 参数的请求，WellInfoController 接到请求后调用 WellInfoServiceImpl 类的 getWellNamesByAreaId() 方法查询该作业区下的所有单井并将单井名异步加载至单井下拉框，实现作业区-单井的联动查询。

DAO 层组件 WellInfoMapper 和 WellDataMapper 为业务逻辑组件提供实现参数汇总功能的多种查询方法。

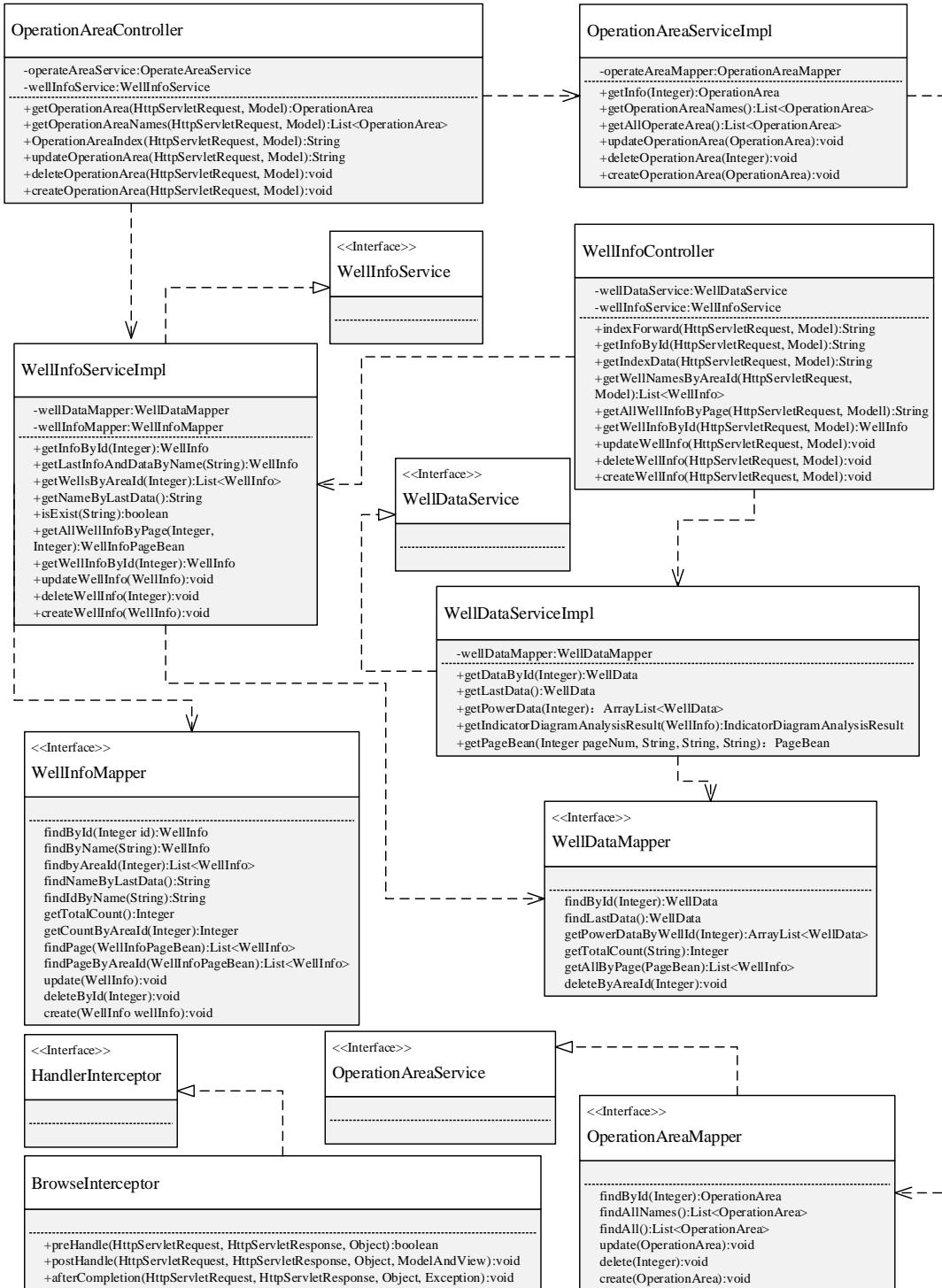


图 4.15 参数汇总模块核心类图

Fig. 4.15 Class Diagram of Parameter Summary Model

4.6.3 参数汇总时序设计

参数汇总模块汇总了单井的井口油压、冲程、冲次、最大载荷、最小载荷、三相电流、三相电压、有功功率、无功功率、功率因数、功图信息以及 1 天功率变化曲线。参数汇总界面也是整个系统的首页。由于汇总信息的时间维度并不相同，所以需要业务逻辑层按照首页不同模块的格式需求，将数据封装好。下面介绍参数汇总模块时序设计。

用户选择单井所属作业区后，程序监听到下拉框的变化，向系统发送查询改作业区内所有单井的井号请求。DispatcherServlet 将该请求派发给 WellInfoController。WellInfoController 类调用其 getWellNamesByAreaId() 方法，该方法通过 UserServiceImpl 与 UserMapper 的交互返回该作业区内所有单井。前端界面将该实体中的信息异步加载至相应的下拉框，等待用户进行下一步选择。

用户点击查询，Spring MVC 中的 DispatcherServlet 在接收到用户发送的查询参数汇总信息的请求后，拦截器将校验该用户是否拥有修改信息的权限。若校验通过，则将该请求派发给 WellInfoController 类。WellInfoController 类调用其 getIndexData ()方法来处理该请求。该方法内通过判断请求参数中是否包含单井井号来决定调用 WellInfoServiceImpl 内的具体方法。点击参数汇总，默认不包含单井井号，页面将呈现数据库中最新一组参数信息。图 4.16 为参数汇总时序图。

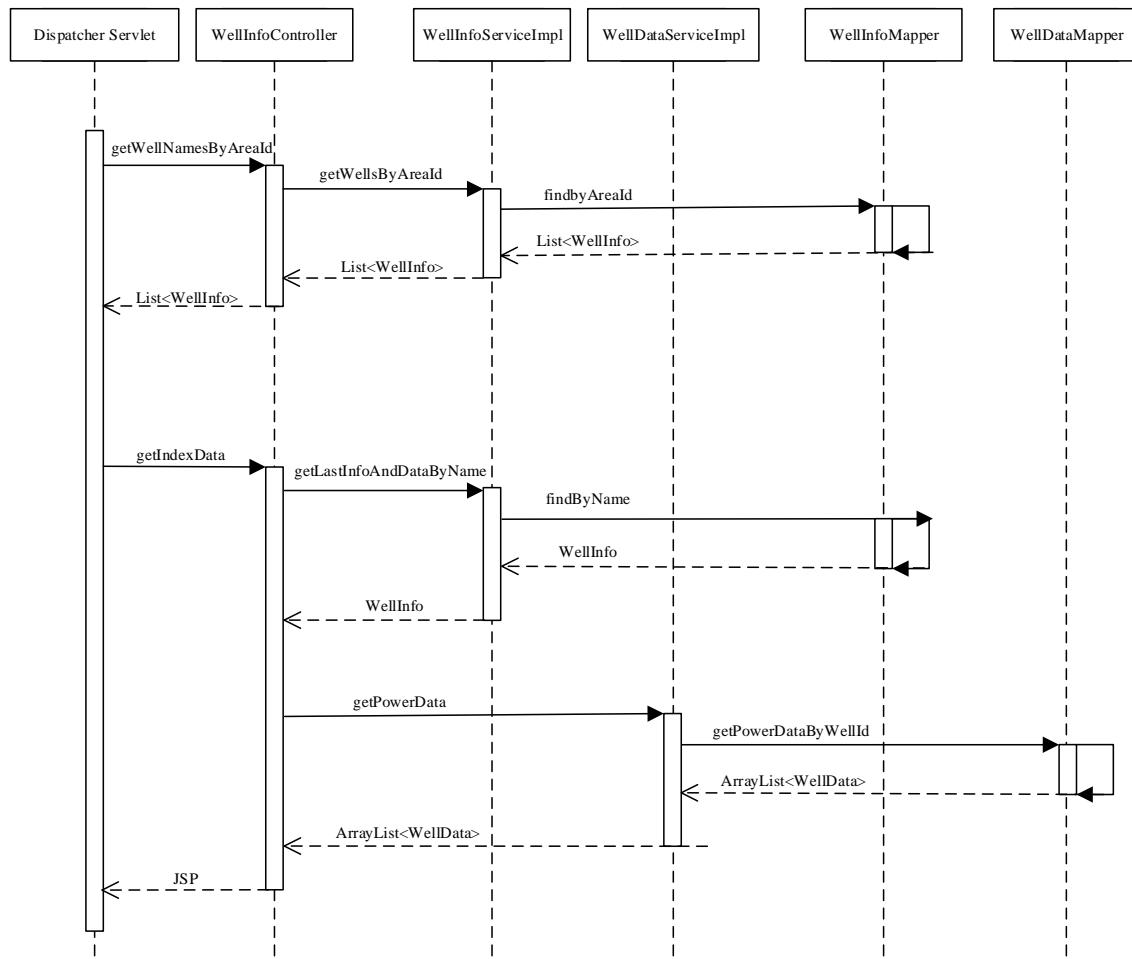


图 4.16 参数汇总时序图

Fig. 4.16 Sequence Chart of Parameter Summary

4.7 曲线分析模块设计

该模块基于第3章所述的基于灰度矩阵的示功图诊断算法。算法接口程序流程图如图4.17所示。

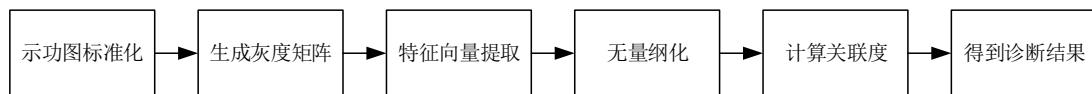


图 4.17 算法接口程序流程图

Fig. 4.17 Flow Chart of Algorithm Interface Program

首先将表 3.1 中典型故障示功图特征向量以故障名-特征向量的形式录入 Map 集合，作为示功图诊断的参考模板。然后对形参传入的待测功图进行标准化，连通区域划分判断每点的相对位置，按照等高线原则获取灰度矩阵，通过灰度直方图进而得到待测示功图的特征向量。此时将参考特征向量与得到的待测示功图特征向量放到一起进行无量纲化处理，然后得到待测示功图与各个标准示功图之间的关联度。对关联度进行排序，得到最终的诊断结果。Domain 层中 IndicatorDiagramAnalysisResult 类封装了诊断信息，成员变量包括待测示功图的 6 个特征值，关联度数组存储了待测示功图与各个标准示功图之间的关联度，以及最大关联度。程序中将数值转化为 BigDecimal，利用 BigDecimal 中的方法来实现算法中的公式，这样做可以保证运算的精度。曲线分析模块核心类图如图 4.18 所示。

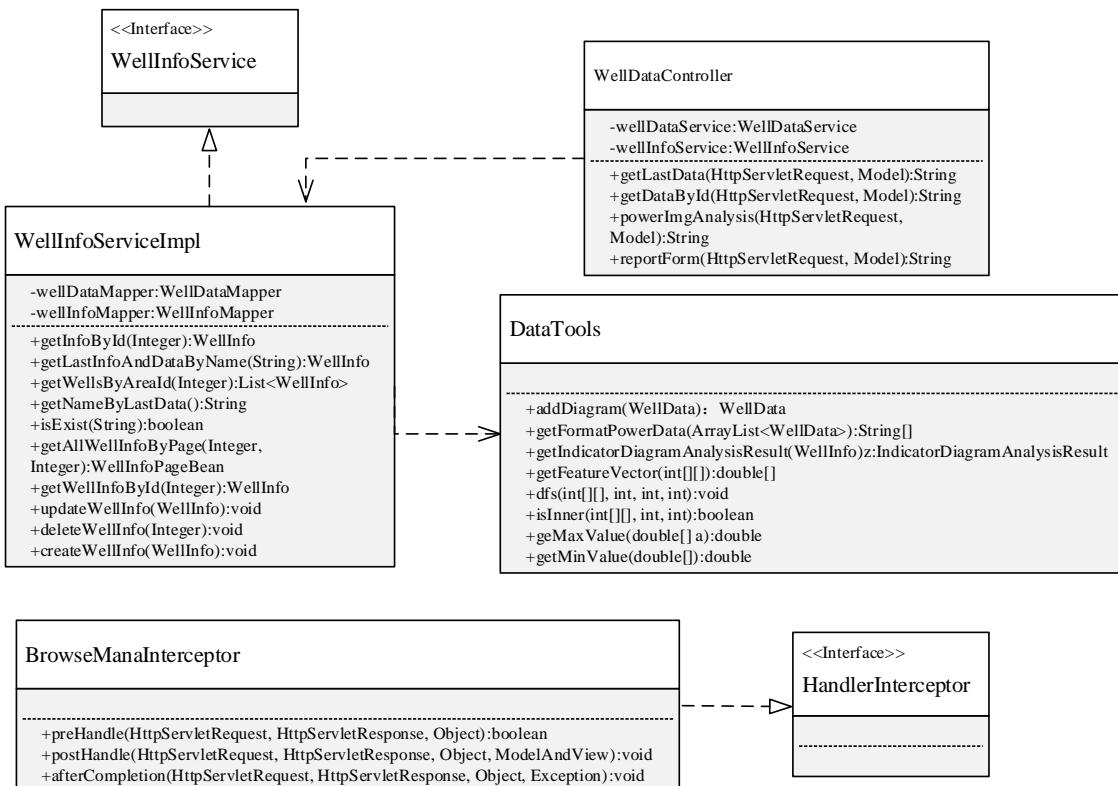


图 4.18 曲线分析模块核心类图

Fig. 4.18 Class Diagram of Curve Analysis Model

4.8 其他模块设计

系统其他模块为报表查询模块和作业区管理模块。其中报表查询模块技术细节包括分页查询以及联动查询，在前文所述中已有所介绍；作业区管理主要是对作业区和单井信息进行 CRUD 操作，功能上与用户管理类似。故仅给出这三个模块的核心类图，不做详述。但是从功能角度上讲，系统五个模块同等重要。图 4.19 为报表查询模块核心类图。图 4.20 为作业区管理模块核心类图。

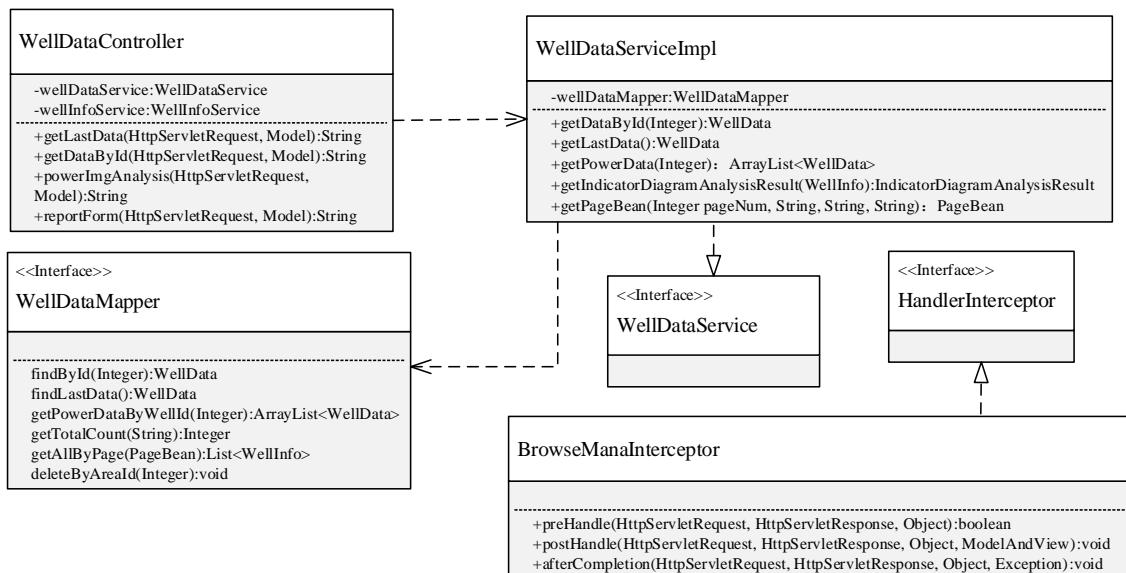


图 4.19 报表查询模块核心类图

Fig. 4.17 Class Diagram of Report Query Model

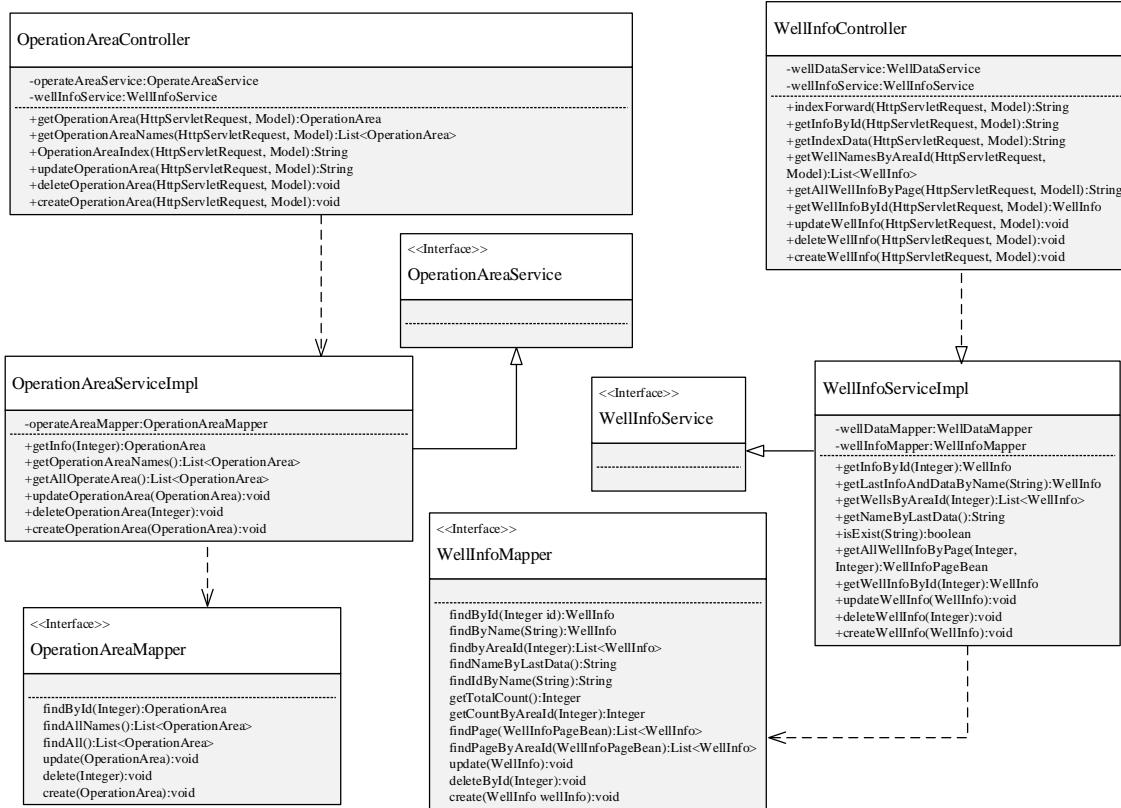


图 4.20 作业区管理模块核心类图

Fig. 4.18 Class Diagram of Operation Area Management Model

4.9 本章小结

本章详细介绍了智慧油田数据管理与故障诊断软件平台。首先简要介绍搭建系统所应用的技术，包括 Spring、Spring MVC、MyBatis 等。然后分别从功能角度和程序设计角度对软件平台整体设计进行了阐述。最后详述了系统中 5 个模块中部分功能的操作流程、程序设计以及时序设计，并在此过程中给出了记住密码、分页查询、联动查询等技术细节的实现方法。

第5章 远程终端单元设计

5.1 远程终端单元总体设计方案

远程终端单元的在系统中的职责是汇总单井所部署的各类传感器所采集的参数，并将这些参数按照既定协议进行打包上传。实际应用中，RTU 向下以串口的方式与传感器进行通信，遵循标准的 Modbus/RTU 协议；RTU 向上通过 TCP/IP 的底层协议与上位机通信，遵循标准的 Modbus/TCP 协议。远程终端单元整体设计架构如图 5.1 所示。

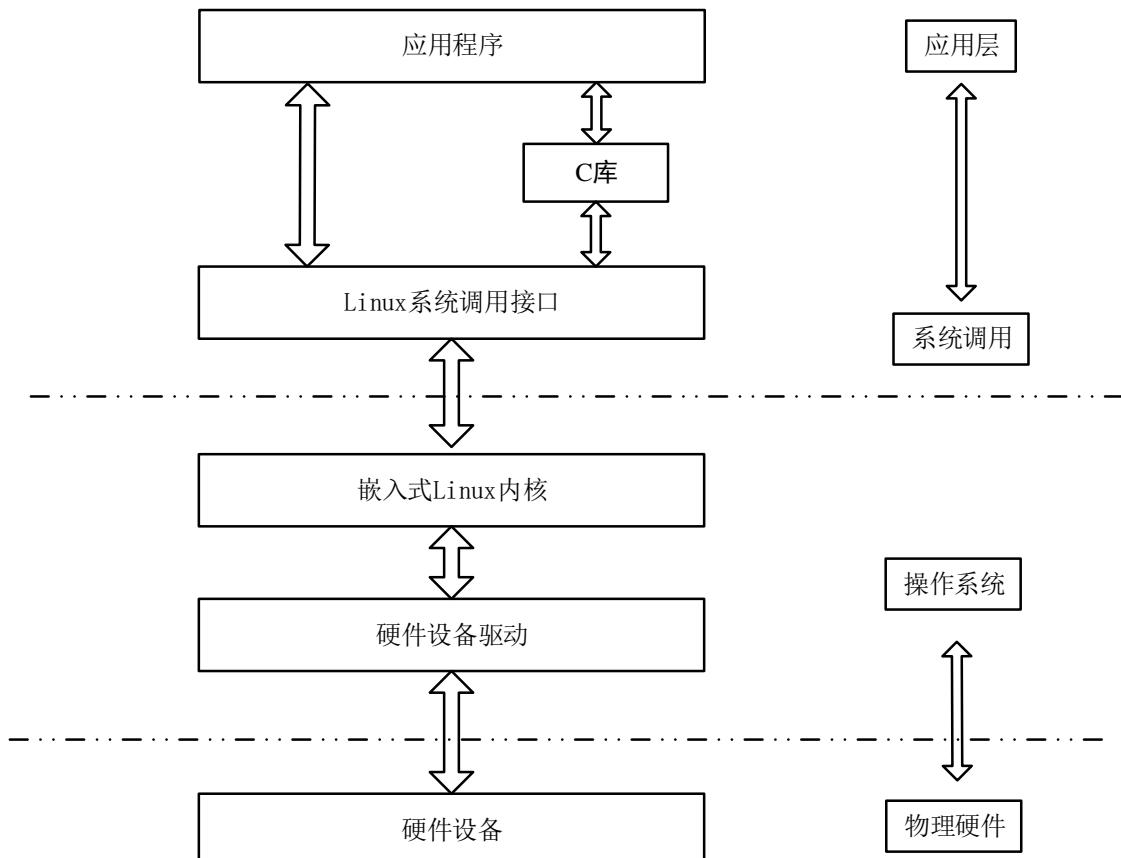


图 5.1 远程终端单元整体设计架构

Fig. 5.1 Overall Design Architecture of Remote Terminal Unit

RTU 操作系统采用嵌入式 Linux 操作系统，具体内核版本为 Linux 2.6.3.2。嵌入式 Linux 操作系统的优点在于源代码开放，可以根据实际需要对内核进行裁剪，

这样可以充分利用处理器有限的资源。在配置好系统内核后需要配置相关硬件设备驱动，包括串口驱动、USB 驱动、网卡驱动等。

应用软件采用多线程的设计方式，在功能设计中充分考虑到了实际运行过程中可能发生的意外情况。例如在接收多包数据帧的过程中，会对包序号进行校验；丢弃不完整的数据，保留重复数据包其中的一包数据等策略。避免线程因数据不完整而长时间等待和重复数据导致存储位置错乱等情况，保障数据的真实可靠。

5.2 远程终端单元应用程序设计

根据 RTU 多任务的需求，采用多线程的方式实现应用程序。系统中的线程包括：主线程、网络通信子线程、电参传感器通信子线程、433MHz 无线接收模块通信子线程。其中网络通信子线程基于 modbus/TCP 通信协议；电参传感器通信子线程、433MHz 无线接收模块通信子线程都基于 modbus/RTU 通信协议。

5.2.1 主线程设计

首先系统启动进行初始化，其中包括配置 RPC 通信参数、串口通信参数、内存缓冲区以及仪器仪表参数等。然后创建系统中各个子线程。在主函数的 while 循环中添加 RPC 服务，目的是完成系统远程启停功能。主线程程序流程图如图 5.2 所示。

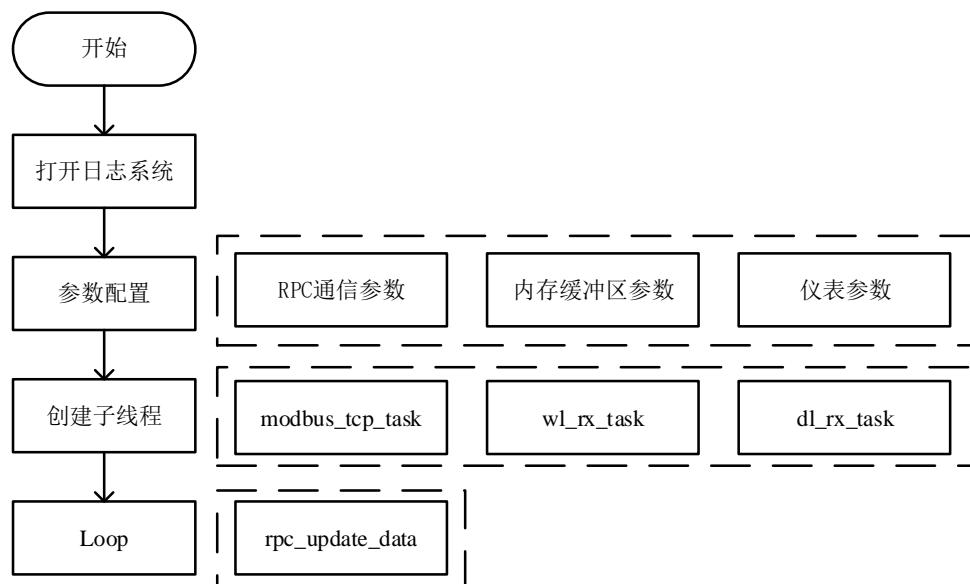


图 5.2 RTU 主线程程序流程图
Fig. 5.2 Main Thread Flow Chart of RTU

5.2.2 Modbus/RTU 通信子线程设计

电参传感器通信子线程和 433MHz 无线接收模块通信子线程都基于 Modbus/RTU 通信协议。由于功能类似，以 433MHz 无线接收模块通信子线程为例进行介绍。

实际应用中，RTU 与 433MHz 无线收发模块通过 485 串口通信。系统运行后，433Mhz 无线收发模块定时接收一体化示功仪和压力变送器所采集的参数，并将数据通过串口即时发送给 RTU。RTU 只需要不间断监听串口，接收数据即可。

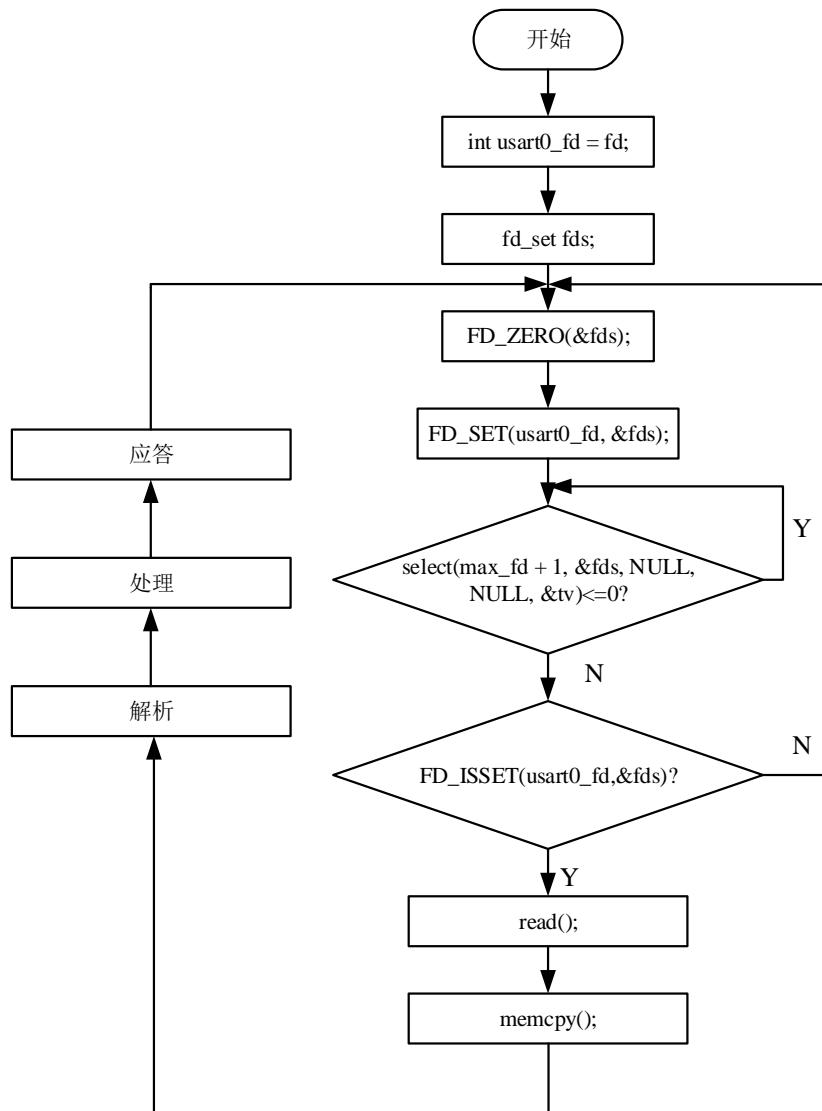


图 5.3 Modbus/RTU 通信子线程程序流程图

Fig. 5.3 Modbus/RTU Communication Subthread Flow Chart

Modbus/RTU 通信子线程程序流程图如图 5.3 所示。按照 Linux 的“一切皆文件”的思想，将串口设备当作文件，通过文件描述符进行操作。将串口文件描述符加入文件描述符集。文件描述符集中每一位代表一个文件描述符。在执行业务逻辑之前，利用 FD_ZERO()函数将文件描述符集的所有位置 0，然后依次添加系统所关心的文件描述符。通过 select 函数监听该文件描述符集。当其中数据发生改变，该函数将返回一个大于 0 的值。进一步通过 FD_ISSET()函数判断指定串口是否接收到数据。若串口接收到数据，则借助 read()函数读取数据请求并将其存储在缓冲区。然后对该请求进行解析处理应答。应用到 433MHz 无线接收模块通信子线程，该线程主要负责接收并解析采集到的油压、功图数据并将它们存储到 Modbus 对应的 4 区主站存储地址，不需要给传感器应答。

5.2.3 Modbus/TCP 通信子线程设计

基于 Modbus/TCP 协议的网络通信子线程接收并解析客户端发送的请求，实现数据打包和上传功能。在网络通信中，RTU 为服务器，上位机为客户端。利用 TCP 协议三次握手机制可以保障数据传输的可靠性。下面结合 TCP 三次握手机制，从 RTU 角度详解嵌入式 Linux 下的网络编程。

在 Modbus/TCP 通信子线程中，首先新建 socket，系统会为其分配一个文件描述符。配置 socket 的 IP 地址和端口号并使之可监听。然后利用 epoll 机制监听 socket。相比于 select 的触发机制，epoll 的优势在于它的效率并不会随监听事件数量的增加而降低。最后通过 accept()函数阻塞等待客户端连接。当客户端连接成功时，accept()会返回一个新的 socket 文件描述符，该 socket 即可用于客户端与服务器之间的通信。

客户端连接成功后，epoll 依然监听 socket 文件描述符。当监听到客户端发来请求时，通过 recv()函数接收请求，然后利用 make_modbus_ack()函数来解析请求并生成响应，最后使用 send()函数将相应结果发送给客户端。其中 make_modbus_ack()函数封装了 RTU 与上位机通信的所有业务逻辑。Modbus/TCP 通信子线程程序流程图如图 5.4 所示。

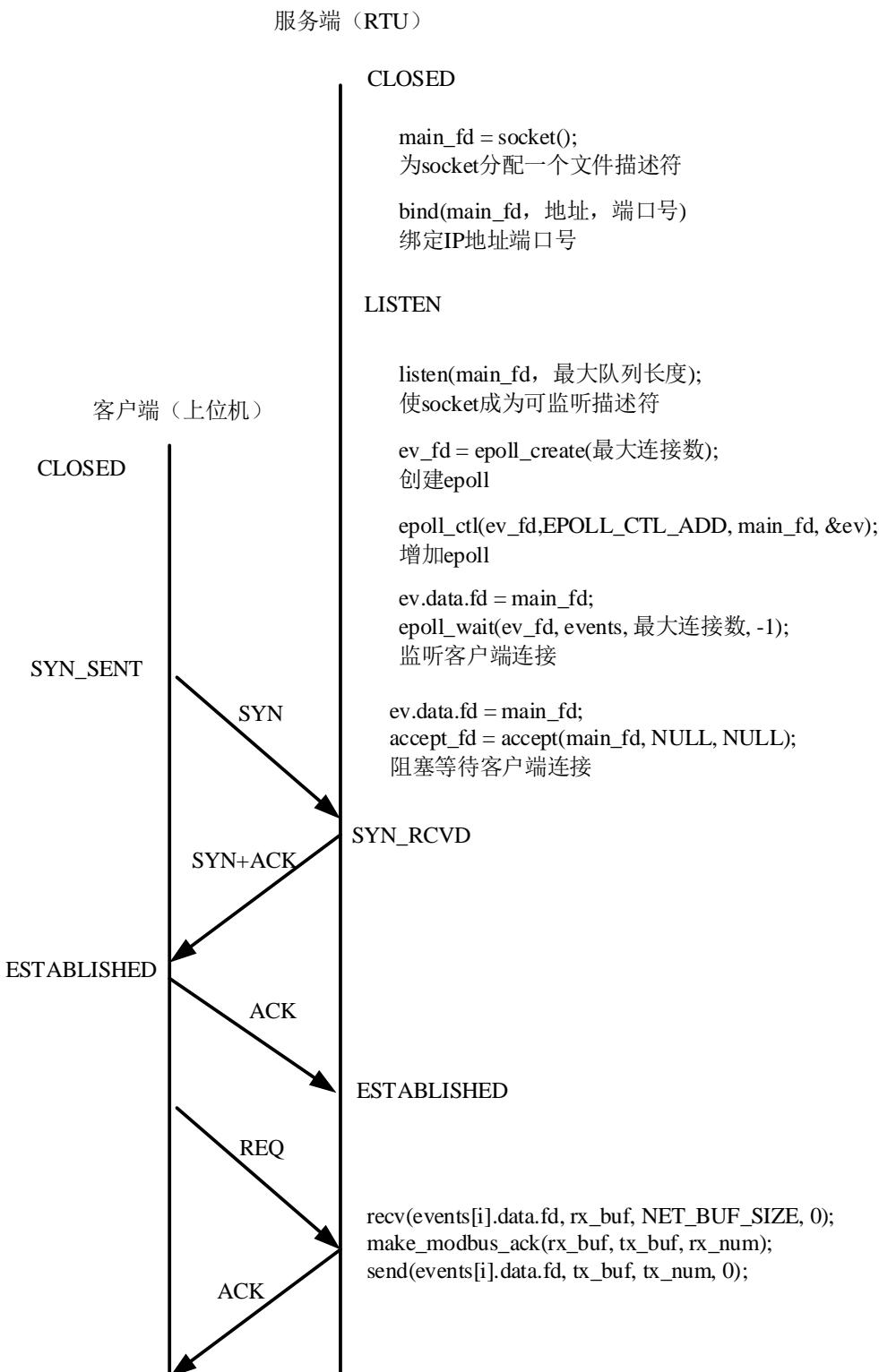


图 5.4 Modbus/TCP 通信子线程程序流程图
Fig. 5.4 Modbus/TCP Communication Subthread Flow Chart

5.3 本章小结

本章首先总体介绍了远程终端单元的设计方案。然后详细阐述了远程终端单元的应用程序设计。应用程序采用多线程方式进行开发，主要包含主线程、网络通信子线程、电参传感器通信子线程、433MHz 无线接收模块通信子线程等四个线程。按照功能类型进行划分，并给出了各个线程的设计流程图。

第6章 系统调试和效果展示

6.1 算法实现优化

按照第3章的算法实现步骤，对示功图样本进行故障诊断测试。选取油田实际采集示功图，作为测试变量，结果发现部分诊断结果与实际情况相差较大。为了找到问题的原因，在程序中添加日志记录，用于打印了所有与算法相关的中间变量。

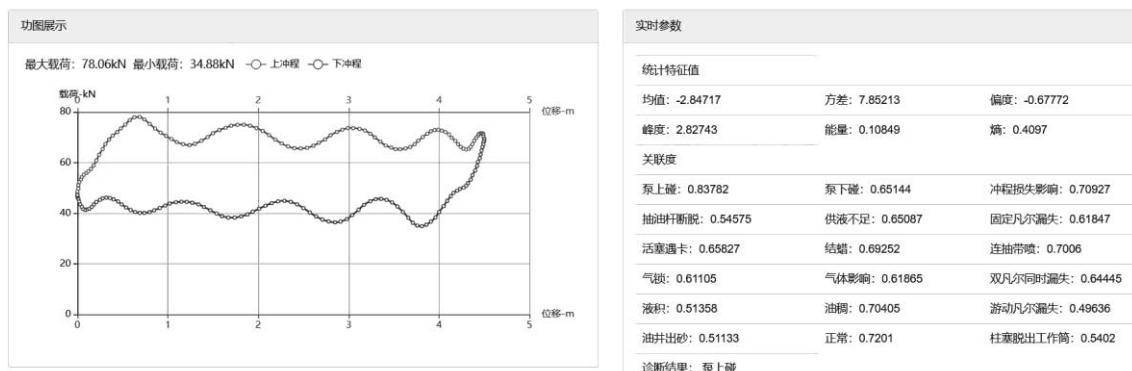


图 6.1 错误诊断结果
Fig. 6.1 Misdiagnosis Result

通过查看系统运行的日志信息，找到了问题出现的根源。在示功图标准化结果如图所示。在示功图的边界出现了断点，如图中标记所示。究其原因，示功仪采集的200组示功图数据是按照相同的时间间隔进行采集，而不是相同的位移差。由此可知，示功图的位移点并不是均匀分布的。当某两个坐标点位移差太大时，应用公式3.3进行位移标准化，就很有可能出现上述断电点问题。同理对于纵向的载荷点，当两点间的载荷差过大时，也会导致断点问题发生。而示功图边界连续，是联通块划分的必要条件。示功图曲线不闭合会导致深度优先搜索算法通过断点“泄露”，联通块区域划分失败，如图6.2所示。后续的特征向量提取也必然失效，最终所得到的诊断结果与理论上的结果相差较大。

图 6.2 不闭合示功图曲线
Fig. 6.2 Misdiagnosis Result

图 6.3 错误的区域划分

为了解决上述问题，对程序中的算法步骤进行了进一步的优化。示功图标准化之前，对原始数据进行预处理，对原始数据进行 1 次线性插值，将 200 组数据变为 400 组，如此可以减小两组坐标点的间距，进而减小断点问题的出现次数。最后给

出算法优化后的结果，从图中可以看出，对于插值后的数据进行标准化，原本断开点被补上，联通块区域划分正常，诊断结果也符合预期。算法优化后的诊断结果如图所示。

图 6.4 优化后标准化示功图

Fig. 6.4 Optimized Standardized Indicator Diagram

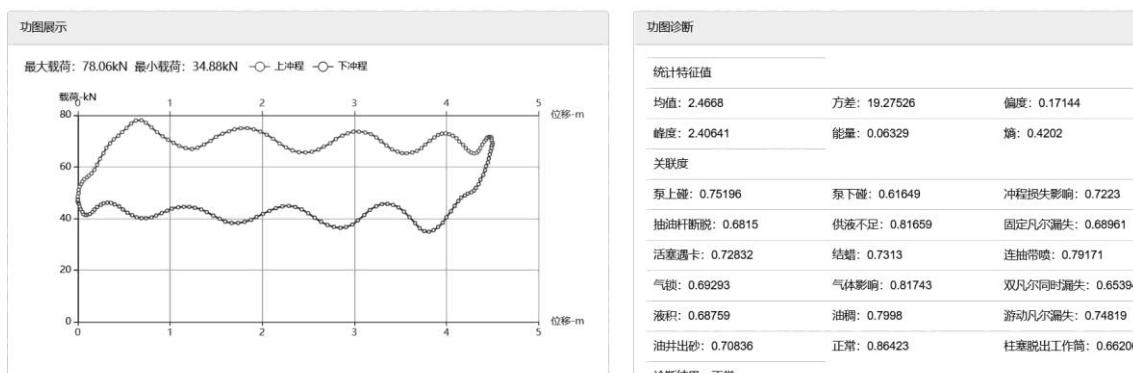


图 6.5 优化后诊断结果

Fig. 6.5 Optimized Misdiagnosis Result

6.2 远程终端单元测试

首先对 RTU 进行通信测试。在连接之前需要确保测试 RTU 与电脑主机的 IP 地址处于同一网段。使用 USB 转串口线将主机与 RTU 相连，如图 6.6 所示。利用超级终端 SecureCRT 对 RTU 进行网络配置，具体配置在 rcS 文件中如图 6.7 所示。配置完成后终端键入 ifconfig 命令验证结果，如图 6.8 所示。然后，将 RTU 与电脑通过网线以点对点的方式进行连接。使用 ModScan32 软件查看 RTU 各模块通信情况。经过分析，整个系统通信正常。图 6.9 为 ModScan32 数据显示界面。



图 6.6 RTU 串口连接图

Fig. 6.6 RTU Serial Port Connection Diagram

```
#!/bin/sh
mount -t proc none /proc
mount -t sysfs none /sys

export PATH=/bin:/sbin:/usr/bin:/usr/sbin:/usr/local/bin
export QTDIR=
export QWS_MOUSE_PROTO="TPanel:/dev/input/event1"
export T_ROOT=/usr/tslib
export LD_LIBRARY_PATH=$T_ROOT/lib
export TSLIB_CONSOLEDEVICE=none
export TSLIB_FBDEVICE=/dev/fb0
export TSLIB_TSDEVICE=/dev/input/event1
export TSLIB_CALIBFILE=/etc/poinercal
export TSLIB_CONFIGFILE=$T_ROOT/etc/ts.conf
export TSLIB_PLUGINDIR=$T_ROOT/lib/t
export TSLIB_TSEVENTTYPE=''
# Start a new shell in the console
/sbin/udevd -daemon &

/sbin/ifconfig eth0 192.168.241.34 netmask 255.255.255.248 &
/usr/local/bin/a11_rtu.bin &
ifconfig eth0 up
/sbin/udevstart &
mount -t devpts devpts /dev/pts
sleep 5
ifconfig eth0 up
route add default gw 192.168.241.33
/sbin/ifconfig eth0 add 240c:806c:0000:0016::f/64 $>
/sbin/route -A inet6 add default gw 240c:806c:0000:0016::1 &
```

图 6.7 RTU 网络配置

Fig. 6.7 Network Configuration of RTU

```
/ # ifconfig
eth0      Link encap:Ethernet Hwaddr 12:34:56:78:90:00
          inet addr:192.168.241.34  Bcast:192.168.241.39  Mask:255.255.255.248
              inet6 addr: fe80::1234:56ff:fe78:9000/64 Scope:Link
                  inet6 brd: 240c:8000:0:100::f/64  scope:Global
                      UNR BROADCAST MULTICAST MTU:1500 Metric:1
                      RX packets:122 errors:7 dropped:0 overruns:0 frame:3
                      TX packets:45 errors:0 dropped:0 overruns:0 carrier:0
                      collisions:0 txqueuelen:1000
                      RX bytes:14517 (14.1 Kib)  TX bytes:2278 (2.2 Kib)
          Interrupt:21 Base address:0xc000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
              inet6 addr: ::1/128 Scope:Host
                  UNR LOOPBACK RUNNING MTU:16436 Metric:1
                  RX packets:12 errors:0 dropped:0 overruns:0 frame:0
                  TX packets:12 errors:0 dropped:0 overruns:0 carrier:0
                  collisions:0 txqueuelen:0
                  RX bytes:1020 (1020.0 B)  TX bytes:1020 (1020.0 B)
```

图 6.8 RTU 网络配置结果

Fig. 6.8 Network Configuration Result of RTU

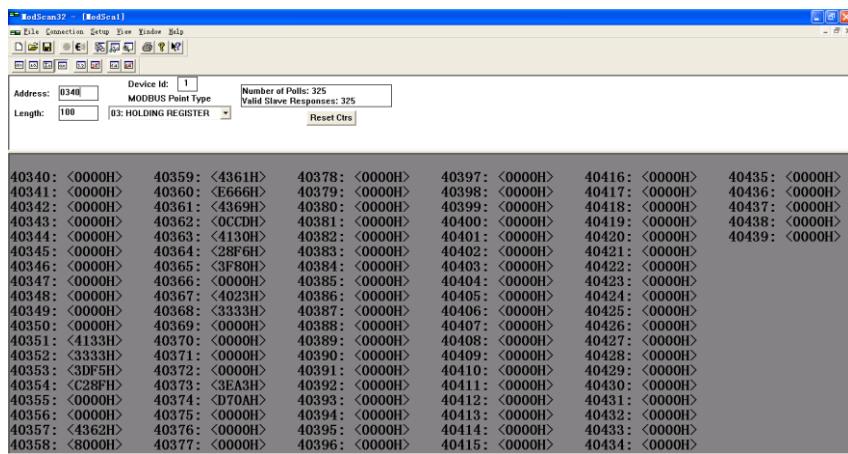


图 6.9 ModScan32 数据显示界面

Fig. 6.9 Data Display Interface in ModScan32

通信测试完成后，对 RTU 进行联调测试。如图 6.10 所示，远程测控机柜中主要包括 RTU 及其 5V 供电电源、RTM 及其 12V 供电电源、电量传感器和 433MHz 无线收发模块。电量模块电压接口接入三项电，电流接口介入电流互感器。433MHz 无线收发模块与无线示功仪、无线压力变送器配对。



图 6.10 RTU 调试机柜

Fig. 6.10 RTU Debugging Cabinet

测试过程中，无线压力变送器每五分钟发送一次压力数据，无线示功仪每十五分钟发送一次模拟示功图数据。利用专用上位机对数据进行采集并写入配置文件，该配置文件内的数据即数据库的数据来源。上位机界面如图 6.11 所示。上位机生成配置文件如图所示。测试结果正常。上位机生成配置文件内容如图 6.12 所示。为了测试 RTU 运行的稳定性，该测试系统进行为期一周的无间断运行测试，结果测试期间运行稳定。



图 6.11 RTU 上位机界面

Fig. 6.11 RTU PC Software Interface

图 6.12 上位机生成的配置文件

Fig. 6.12 Configuration File Generated by PC Software

6.3 软件平台测试和效果展示

功能测试方面，系统运行后对所有功能进行测试。除了按照正常流程的操作之外，还进行了非常规操作，用来测试系统是否存在逻辑上的漏洞。例如，在用户管理模块中，进入修改用户信息，不做任何修改直接提交，界面显示信息未修改；在报表查询模块中只输入查询的开始时间，而不输入结束时间，系统会自动以当前时间作为结束时间；客户端手动删除 Cookie，则再次进入网站转入登录界面。无论时常规流程还是非常规操作，软件平台均通过了测试。表 6.1 为软件平台所有测试用例。

表 6.1 软件平台测试用例

Table 6.1 Software Platform Test Cases

测试功能	描述	预期结果	测试结果
用户登录	输入正确用户名和密码	跳转至系统首页	通过
	输入不存在的用户名	提示用户不存在	通过
	输入错误密码	提示密码错误	通过
	用户未经登录直接访问 系统功能界面	跳转至登录页面	通过
	记住密码后重启浏览器，访问系统功能界面	正常访问	通过
	记住密码后手动删除 cookie，重启浏览器， 访问系统功能界面	跳转至登录页面	通过

表 6.1 软件平台测试用例（续）
Table 6.1 Software Platform Test Cases(Cont)

修改个人信息	输入新用户信息后提交	提示修改成功	通过
	保持原有信息直接提交	提示信息未修改	通过
	修改已存在用户名提交	提示用户名已存在	通过
修改密码	输入正确原密码、新密码以及确认密码然后提交	提示修改成功	通过
	输入错误原密码提交	提示原密码错误	通过
	输入新密码与确认密码不同	提示密码不一致	通过
查找用户	输入用户名点击查找	查找出所有用户名中包含输入信息的用户	通过
	输入工号点击查找	查找出所有工号中包含输入信息的用户	通过
	输入空的用户名信息点击查找	显示系统内所有用户信息	通过
	输入空的工号信息点击查找	显示系统内所有用户信息	通过
删除/添加用户	点击删除	该用户被删除	通过
	输入用户信息点击添加	提示添加成功	通过
	输入信息中用户名重复	提示用户名已存在	通过
参数汇总	输入正确井号点击查询	显示该单井最新参数汇总信息	通过
	选择作业区	下拉框显示该作业区内所有单井	通过
	选择单井点击查询	显示该单井最新参数汇总信息	通过
	直接点击查询	显示数据库最新一组参数汇总信息	通过
	仅选择作业区而不选择单井点击查询	显示数据库最新一组参数汇总信息	通过
报表查询	选择正确的作业区单井和时间段点击查询	显示该单井在该时间段内的所有数据	通过
	仅选择时间段而不选择单井	显示在该时间段内所有单井数据	通过
	仅选择开始时间不选择结束时间	显示开始时间至系统时间所有单井数据	通过

表 6.1 软件平台测试用例（续）
Table 6.1 Software Platform Test Cases(Cont)

曲线分析	输入正确井号点击查询	显示该单井最新功图和曲线分析结果	通过
	选择作业区	下拉框显示该作业区内所有单井	通过
	选择单井点击查询	显示该单井最新功图和曲线分析结果	通过
	直接点击查询	显示数据库最新功图和曲线分析结果	通过
	仅选择作业区而不选择单井点击查询	显示数据库最新功图和曲线分析结果	通过
新增作业区	输入正确作业区信息点击添加	提示添加成功	通过
	输入已存在作业区 ID 点击添加	提示作业区 ID 已存在	通过
删除作业区	点击删除	删除作业区信息、作业区下所有单井信息以及单井采集数据	通过
修改单井信息	输入新单井信息后提交	提示修改成功	通过
	保持原有信息直接提交	提示信息未修改	通过
	修改已存在井号提交	提示井号已存在	通过
新增单井	输入正确单井信息点击添加	提示添加成功	通过
	输入已存在井号点击添加	提示井号已存在	通过
删除单井	点击删除	删除单井信息以及该单井所有采集数据	通过
查询单井	选择作业区点击查询	显示该作业区内所有单井信息	通过
	直接点击查询	显示系统内所有单井信息	通过

为了保障软件平台可靠运行，使用专用测试工具 Jmeter 对软件平台的并发性进行测试。测试分为 3 组，每组测试中，设定 5 个 HTTP 请求，这 5 个请求对应 5 个功能模块。用于测试的 5 个功能模块请求如图所示。第一组测试中，设定总线程数为 1000，ramp-up period 为 10 秒，即 10 秒内启动 1000 个线程。测试结果如图 6.13 所示。

Label	# 样本	平均值	最小值	最大值	标准偏差	异常 %	吞吐量	接收 KB/sec	发送 KB/sec	平均字节数
报表查询	1000	2	1	15	1.22	0.00%	100.1/sec	1629.49	27.86	18671.0
参数汇总	1000	2	1	9	1.01	0.00%	100.1/sec	2613.75	24.15	26738.0
曲线分析	1000	9346	96	16546	4348.61	0.00%	39.5/sec	710.73	9.98	18448.0
作业区管理	1000	2	1	7	0.77	0.00%	100.1/sec	1911.87	24.44	19558.0
用户管理	1000	1	1	12	0.68	0.00%	100.2/sec	86.95	22.49	889.0
总体	5000	1871	1	16546	4213.24	0.00%	17.5/sec	281.55	4.35	16460.8

图 6.13 Jmeter 第 1 组测试结果

Fig. 6.13 The First Set of Test Results in Jmeter

第二组测试，将总线程数改为 2000，运行测试用例，测试结果如图 6.14 所示。

Label	# 样本	平均值	最小值	最大值	标准偏差	异常 %	吞吐量	接收 KB/sec	发送 KB/sec	平均字节数
报表查询	2000	0	0	2	0.50	0.00%	200.1/sec	244.90	57.07	1253.0
参数汇总	2000	1	1	12	0.85	0.00%	200.2/sec	5226.97	48.29	26738.0
曲线分析	2000	23101	251	42507	11921.49	0.00%	38.5/sec	693.02	9.73	18448.0
作业区管理	2000	1	1	4	0.66	0.00%	200.3/sec	3826.43	48.91	19558.0
用户管理	2000	1	0	4	0.44	0.00%	200.2/sec	173.82	44.97	889.0
总体	10000	4621	0	42507	10667.72	0.00%	25.7/sec	335.95	6.42	13377.2

图 6.14 Jmeter 第 2 组测试结果

Fig. 6.14 The Second Set of Test Results in Jmeter

第三组测试，总线程数改为 5000，同样是 10 秒内全部启动，测试结果如图 6.15 所示。

Label	# 样本	平均值	最小值	最大值	标准偏差	异常 %	吞吐量	接收 KB/sec	发送 KB/sec	平均字节数
报表查询	5000	9	1	409	45.85	0.00%	500.5/sec	8147.47	139.29	16671.0
参数汇总	5000	1	1	28	0.93	0.00%	500.2/sec	13060.89	120.65	26738.0
曲线分析	5000	63776	2246	121182	34289.59	0.00%	38.1/sec	687.13	9.65	18448.0
作业区管理	5000	1	0	44	2.09	0.00%	500.5/sec	9559.36	122.19	19558.0
用户管理	5000	1	0	27	0.90	0.00%	499.7/sec	433.78	112.23	889.0
总体	25000	12758	0	121182	29763.74	0.00%	29.6/sec	476.56	7.36	16460.8

图 6.15 Jmeter 第 3 组测试结果

Fig. 6.15 The Third Set of Test Results in Jmeter

分析三组测试结果可以发现，对于报表查询、参数汇总、作业区管理、用户管理这四个功能模块，即使每秒钟的访问量达到 500，响应时间依然是毫秒级别的。而对于曲线分析模块，由于算法步骤较多，导致响应较慢，每秒钟的访问量为 1000 时，平均响应时间接近 1 秒。而访问量增至 500/秒时，系统响应时间已经达到了 6 秒，最大响应时间甚至达到了 12 秒。从吞吐率这一项数据看，曲线分析每秒响应的请求数量至多为 38。由测试结果可知，曲线分析功能决定了整个软件平台承受访问压力的上限，算诊断算法流程还需要日后进一步优化。另一方面，在这三组测试中，事务出错率均为 0%，这验证了软件平台的可靠性。

最后给出前端页面效果图。图为用户登陆界面；图为参数汇总界面；图为报表查询界面；图为曲线分析界面；图为作业区管理界面；图为用户管理界面。

④ 智慧油田数据采集与故障诊断研究与实现 软件平台



图 6.16 用户登录界面
Fig. 6.16 User Login Interface

④ 智慧油田数据采集与故障诊断研究与实现 软件平台

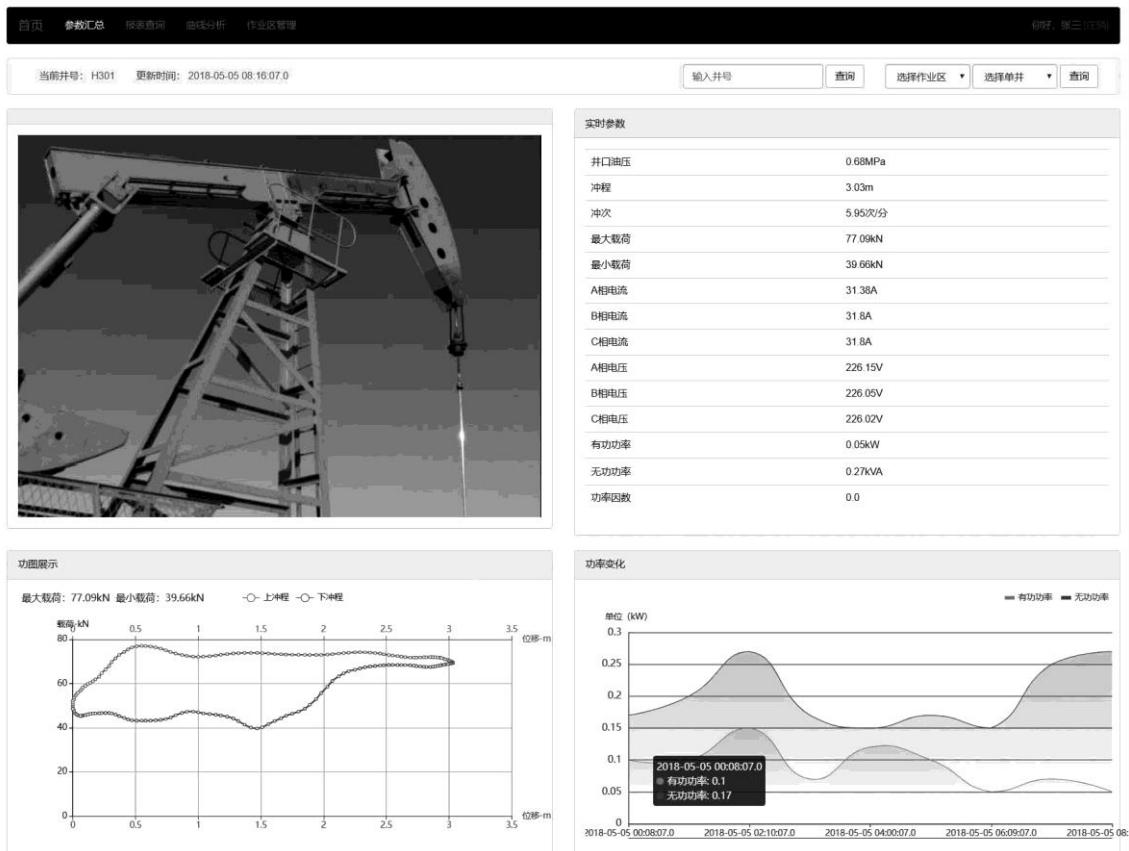


图 6.17 参数汇总界面
Fig. 6.17 Parameter Summary Interface

智慧油田数据采集与故障诊断研究与实现 软件平台

首页 参数汇总 报表查询 曲线分析 作业区管理 你好，张三 | 登录

选择作业区 ▼ 选择单井 ▼ 请选择开始日期 请选择结束日期 查询

井号	井口油压	冲程	冲次	最大载荷	最小载荷	A相电流	B相电流	C相电流	A相电压	B相电压	C相电压	有功功率	无功功率	功率因数	时间
A10-15	0.67	3.35	4.5	79.4	48.48	28.75	29.67	29.67	227.47	227.65	226.57	2.57	19.55	-0.13	2018-04-28 14:01:18.0
A10-15	0.69	3.51	4.49	73.46	46.11	29.07	29.83	29.83	230.12	230.27	229.02	1.27	20.02	0.06	2018-04-28 15:01:29.0
A10-15	0.69	3.51	4.49	73.46	46.11	29.17	30.14	30.14	231.2	231.52	230.32	1.55	20.27	-0.07	2018-04-28 16:00:29.0
A10-15	0.64	3.55	4.5	82.65	55.06	28.75	29.58	29.58	228.2	228.47	227.12	1.17	19.57	0.06	2018-04-28 17:00:29.0
A10-15	0.7	3.36	4.5	80.95	52.1	30.76	31.63	31.63	226.65	226.85	225.65	5.67	19.95	0.27	2018-04-28 18:00:23.0
A10-15	0.63	3.27	4.5	83.52	54.94	29.4	30.12	30.12	228.52	228.77	227.52	3.0	19.87	0.15	2018-04-28 19:05:23.0
A10-15	0.63	3.32	4.5	82.53	55.15	33.91	34.81	34.81	227.2	227.32	226.2	9.72	20.8	0.42	2018-04-28 20:12:23.0
A10-15	0.68	3.37	4.49	82.25	54.9	29.28	30.08	30.08	228.2	228.45	227.32	2.65	19.7	0.13	2018-04-28 21:53:23.0
A10-15	0.65	3.35	4.5	82.55	55.13	28.64	29.37	29.37	228.95	229.17	228.0	1.55	19.6	0.08	2018-04-28 22:16:23.0
A10-15	0.7	3.36	4.49	82.35	55.14	29.72	30.52	30.52	230.87	231.02	229.72	2.2	20.2	0.1	2018-04-28 23:10:23.0

第1/15页 下一页

图 6.18 报表查询界面

Fig. 6.18 Report Query Interface

智慧油田数据采集与故障诊断研究与实现 软件平台



图 6.19 曲线分析界面

Fig. 6.19 Curve Analysis Interface

智慧油田数据采集与故障诊断研究与实现 软件平台

作业区信息管理

作业区ID	作业区名	作业区地址	联系人	联系电话	备注
7181	作业区A	北京市昌平区	张三	13715486207	修改 删除
7182	作业区B	北京市海淀区	李四	18810254963	修改 删除
7183	作业区T				修改 删除

单井信息管理

井号	ip地址	所属作业区	状态	联系人	联系电话	备注	操作
A10-15	192.168.241.12	作业区A	1	王五	18812547896		修改 删除
A10-16	192.168.241.10	作业区A	1	王五	18812547896		修改 删除
A10-17	192.168.241.9	作业区A	1	王五	18812547896		修改 删除
A11-20	192.168.241.8	作业区A	1	王五	18812547896		修改 删除
A11-30	192.168.241.7	作业区A	1	王五	18812547896		修改 删除

第1/3页 下一页

图 6.20 作业区管理界面
Fig. 6.18 Operating Area Management Interface

 智慧油田数据采集与故障诊断研究与实现 软件平台

首页 参数汇总 报表查询 曲线分析 作业区管理 你好, 张三 [注销]

个人信息管理

用户名:	admin	姓名:	张三	工号:	001
角色:	1	电话:	13245678902	保存	修改密码

用户管理

		按姓名检索		按工号检索		重置	新增用户
用户ID	用户名	姓名	工号	电话	角色	创建时间	操作
1	admin	张三	001	13245678902	1	2018-05-03 23:16:57.0	修改 删除
2	user1	李四	002	13245678901	2	2018-05-03 23:16:57.0	修改 删除
3	user4	王五	003	13245678901	2	2018-05-03 23:16:57.0	修改 删除
4	user3	赵六	004	18611133344	2	2018-05-03 23:16:57.0	修改 删除
7	user6	李白	007	13223223232	2	2018-05-03 23:16:57.0	修改 删除

第1/3页下一页

权限管理

角色	权限	操作
角色1	/浏览信息/操作区管理/用户管理	<input type="checkbox"/> 浏览信息 <input type="checkbox"/> 操作区管理 <input type="checkbox"/> 用户管理 <input checked="" type="button"/> 修改
角色2	/浏览信息/操作区管理	<input type="checkbox"/> 浏览信息 <input type="checkbox"/> 操作区管理 <input type="checkbox"/> 用户管理 <input checked="" type="button"/> 修改
角色3	/浏览信息	<input type="checkbox"/> 浏览信息 <input type="checkbox"/> 操作区管理 <input type="checkbox"/> 用户管理 <input checked="" type="button"/> 修改

图 6.21 用户管理界面
Fig. 6.19 User Management Interface

6.4 本章小结

本章首先针对算法应用过程中遇到的问题进行优化。在示功图标准化之前进行了1次线性插值，增加点的密度，减少断点情况的发生。然后对RTU硬件进行调试，保障RTU通信正常。最后对软件平台进行了功能性测试和压力测试。软件平台通过了所有的测试用例。软件性能测试结果显示，算法运算效率还有待提高。

第7章 总结与展望

7.1 工作总结

本文首先对石油行业的发展现状以及现有功图法进行了调研，进而提出了建立基于功图法的远程诊断系统的重要意义。然后对现有功图特征提取方法做了简要的介绍，在此基础提出了基于灰度矩阵的示功图特征提取算法。根据上述特征提取算法设计了集数据管理与故障诊断的软件平台和用于数据采集传输的远程终端单元。最后对整个系统进行了调试，并最终实现了一套应用于智慧油田的数据采集与故障诊断系统。

其中详细介绍了示功图标准化方法，在灰度矩阵提取过程中所应用的基于深度优先搜索的连通区域划分算法和基于广度优先搜索的灰度等高线划分算法，特征向量的提取方法。

从业务流程、程序设计以及时序设计三个角度详细阐述了软件平台的实现方案，并介绍了远程终端单元的软件、硬件实现原理。

具体的研究成果如下：

(1) 在查阅大量文献的基础上，分析了智慧油田数据采集与故障诊断系统的需求之后，制定了系统的整体实现方案。

(2) 设计了基于灰度矩阵的示功图特征向量提取算法。该算法以网格法为基础结合了模式识别相关理念。特征提取过程中应用的算法主要包括：对于待测示功图的预处理即示功图标准化算法；基于深度优先搜索的连通区域划分算法；基于广度优先搜索的灰度等高线划分算法。通过上述算法得到示功图的灰度矩阵，并将该矩阵的 6 个特征值作为示功图特征向量。以特征向量为依据，计算待测示功图于示功图模板的关联度，即可达到故障诊断的目的。

(3) 以故障诊断算法为核心，设计并实现了智慧油田数据管理与故障诊断软件平台。软件平台采用 SSM 架构进行开发，围绕系统五个核心模块，完成了从持久层到前端层的设计并实现了预期功能。最后通过测试验证了系统的可靠性。

(4) 完成远程终端单元的硬件设计，在 Linux 平台上完成系统应用程序的开发。

(5) 对设备进行现场调试，并根据调试结果对硬件应用程序的逻辑进行优化。

7.2 工作展望

在现有系统中，远程终端单元的数据采集是通过上位机来实现的。上位机通过轮询的方式采集数据，将数据存储于指定位置的配置文件中。再由软件平台读取配置文件后对数据进行持久化。实际上上位机的功能完全能够整合到软件平台。

对于软件平台，目前只实现了基本功能。虽然可以满足实际需求，但可以对功能进行扩展。比如现场视频监控，但是这需要硬件的支持。此外，为了提高系统的响应速度，算法的实现方式还有待于优化。

远程终端单元方面，目前的设备只适用于参数的实时传输，对于传输线路故障的情况没有应对策略。可以在程序中设计本地数据库，或者使用消息队列。

参 考 文 献

- [1] BP Statistical Review of World Energy. 2012
- [2] 张旭.中国石油开采的现状及问题探析[J].化工管理,2016(35):120.
- [3] 郭昱. 国际原油价格冲击对中国股票回报率的时变影响研究[D].湖南大学,2018.
- [4] Gibbs S G, Neely A B. Computer Diagnosis of Down-hole Conditions in Sucker Rod Pumping Wells. J. Pet. Tech. (January 1966)91-98.
- [5] Gibbs S G, Neely A B. Method of Determining Sucker Rod Pump Performance. U. S. Patent 3, 343, 409. Spet 26, 1967.
- [6] 刘珂.人工智能在档案管理中的应用探析[J].兰台世界,2019(02):19-23+12.
- [7] Foley W L and Svinos J G Exprod.Expert Advisor Program For Rod Pumpings.Chevron Oil field Research Co SPE 16920.
- [8] Rogers J D,Guffey C G,Oldham W J B. Artificial Neural Networks for Identification of Beam Pump Dynamo-meter Load Card[J].SPE2061,1990.
- [9] Nazi G M. Application of Artificial Neural Network to Pump Card Diagnosis[J]. SPE25420,1993.
- [10] 谢宗旺. 非现场监管分析系统设计与实现[D].南京航空航天大学,2011.
- [11] 张勇. 基于嵌入式技术的物联网网关设计[D].中国石油大学(北京),2013.
- [12] 崔振华, 余国安.有杆抽油系统.北京:石油工业出版社, 1994.
- [13] 王鸿勋, 张琪等.采油工艺原理.北京:石油工业出版社, 1985:1-89.
- [14] 宋云祖.采油工程.北京:石油工业出版社, 1991.
- [15] HB 布雷德利 (美) .石油工程手册.北京:石油工业出版社, 1992.
- [16] 抽油井示功图的测试和分析编写小组.抽油机示功图的测试和分析.石油工业出版社, 1983. 05
- [17] 刘益江, 张学臣, 李伟.抽油井示功图综合解释.油气田地面工程, 2007, 26 (8) : 3-5.
- [18] 孙维娟, 马来增, 汤恒斌.有杆泵抽油系统常见井下故障分析.石油矿场机械, 2002,31(4):59-61.
- [19] 吴伟, 王玮.机械式有杆泵抽油井示功图的识别.石油矿场机械, 2007, 36 (4) :16-18.
- [20] 任广武, 杨凤波特高含水期抽油机井示功图变异特征.油气田地面工程, 2007, 26 (1) :26-27.
- [21] 余国安, 彭勇.应用模式识别技术实现抽油机井下故障的自动诊断.西安石油学院学报, 1990, 5 (1):23-32.

- [22] 韩国庆, 吴晓东, 张庆生. 示功图识别技术在有杆泵工况诊断中的应用. 石油钻采(艺), 2003, 25 (5):70-74.
- [23] 苑顺仓. 简明示功图判别法. 油气井测试, 2005, 14 (1) : 49-50
- [24] 李正勤, 黎洪生. 基于示功图面积变化的抽油机故障诊断模型. 油气田地面工程, 2008,27 (9):3-4.
- [25] J F Lea. Boundary Condition used Dynamic Models of Beam Pump Performance. Southwestern Petroleum Short Course,1988:251-263.
- [26] 杨顺辉. 差分曲线法在示功图诊断中的应用. 石油钻探技术, 2002, 30 (2) : 63-64.
- [27] 麦莹. 基于灰色理论的抽油井泵功图诊断技术研究[J]. 微电子学与计算机,2005(03):77-80.
- [28] 李正勤, 黎洪生. 基于示功图面积变化的抽油机故障诊断模型. 油气田地面工程, 2008, 27 (9):3-4.
- [29] 余国安, 彭勇. 应用模式识别技术实现抽油机井下故障的自动诊断. 西安石油学院学报, 1990 5 (1):23-32.
- [30] 周传宏,陈郭宝,王怀虎,康少博. OpenCV 在条烟视觉检测系统中的应用[J].机械设计与制造,2011(11):72-74.
- [31] 张海浪,程涛,李景,何伟荣,兰海涛. 抽油井功图诊断过程中灰度法数学模型的校正[J]. 青海石油,2012,30(01):17-22.
- [32] 王然. 基于 S2SH 的飞行器试验数据处理系统设计与实现[D]. 哈尔滨工业大学,2011.
- [33] 蔡洁. 基于 S2SH 框架的监控管理客户端的设计与实现[D]. 浙江工业大学,2015.
- [34] 赵英军. Hybrid 模式系统框架的设计与应用[D]. 大连海事大学,2018.
- [35] 李文杰. 基于 SSH 框架的客户响应系统设计与实现[D]. 电子科技大学,2014.

致 谢

时光匆匆，又到了毕业季。七年的石大时光让我早已熟悉这里的一切。到了要离别的时刻，还真是有些不舍。石大见证了我的成长，感谢石大为我提供的优越的学习环境。愿母校在未来的日子里发展越来越好。

研究生三年，对我帮助最大的无疑是我的导师曹旭东老师。无论在学术上还是在生活上，曹老师都给予我很大的帮助。老师在关键时候的引导让我在成长过程中少走了不少弯路。同时感谢这三年里遇到的每一位老师，他们都是各自研究领域的佼佼者。感谢他们为我解决了学习过程中遇到的种种问题。

最后还要由衷感谢和我一起奋斗的同学们。三年的时光里我们一起攻克了学术竞赛、项目研发中的道道难关，同时也结下了深厚的友谊。愿大家未来一切都好。

攻读硕士学位期间参加的科研项目和学术成果

1. 参与的科研项目

- [1] 二连高寒 IPV6 自动化前端软硬件升级, 华北油田项目
- [2] 过套管电阻率测井系统设计与解释方法研究, 中石化集团公司项目

2. 学术成果

- [1] 王森,曹旭东,魏文龙,吴晓珊.基于 ARM+Linux 架构智能油气水井物联网系统设计[J].计算机测量与控制,2018,26(08):262-265.
- [2] 曹旭东,王森,吴晓珊,魏文龙.警用摩托车管理系统（软件著作权, 登记号: 2018SR133401）
- [3] 曹旭东,吴晓珊,王森,魏文龙.过套管电阻率测井数据采集与处理软件（软件著作权, 登记号: 2018SR184069）