

杭州电子科技大学

硕士学位论文

题 目：缝纫机在线监测系统的设计与实现

研 究 生 吴文楷

专 业 电子与通信工程

指导教师 陈科明 副教授

企业导师 曾 佳

完成日期 2022 年 5 月

杭州电子科技大学硕士学位论文

缝纫机在线监测系统的设计与实现

研 究 生： 吴文楷

指导教师： 陈科明 副教授

2022 年 5 月

**Dissertation Submitted to Hangzhou Dianzi University
for the Degree of Master**

Sewing machine online monitoring system design and implementation

Candidate: Wu Wenkai

Supervisor: Associate Prof. Chen Keming

May, 2022

杭州电子科技大学

学位论文原创性声明和使用授权说明

原创性声明

本人郑重声明：所呈交的学位论文，是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不含任何其他个人或集体已经发表或撰写过的作品或成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。

申请学位论文与资料若有不实之处，本人承担一切相关责任。

论文作者签名：



日期：2022 年 5 月 31 日

学位论文使用授权说明

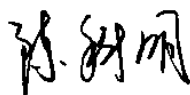
本人完全了解杭州电子科技大学关于保留和使用学位论文的规定，即：研究生在校攻读学位期间论文工作的知识产权单位属杭州电子科技大学。本人保证毕业离校后，发表论文或使用论文工作成果时署单位名称仍然为杭州电子科技大学。学校有权保留送交论文的复印件，允许查阅和借阅论文；学校可以公布论文的全部或部分内容，可以允许采用影印、缩印或其它复制手段保存论文。（保密论文在解密后遵守此规定）

论文作者签名：



日期：2022 年 5 月 31 日

指导教师签名：



日期：2022 年 5 月 31 日

摘 要

如今随着智能制造发展，传统制造业需要通过智能化的解决方案来实现产业升级，提高生产效率。缝纫机作为一个传统制造业已经有一百多年的历史，目前缝纫机系统普遍存在以下问题：不能及时同步设备运行参数和状态；对缝纫机设备故障识别不准确；整个缝纫机生产线工序分配不均衡等问题，极大地影响整个系统的工作效率。

本文针对目前缝纫机系统存在的以上缺点，实现了基于物联网技术设计的一套缝纫机在线监测系统：用自主研发的智能网关与缝纫机的控制器对接，对控制器终端进行实时采集设备参数，通过 Wi-Fi 无线通信技术将数据上传到物联网平台进行处理分析。在整个系统运行过程中，会将实时采集的数据及情况可视化的展示在 Web 端和小程序页面上，通过信息可视化的展示给用户提供数据透明化，操作便捷化的体验。在必要时候，也可远程对缝纫机进行控制。该系统实现了远程监测，生产线管理，故障统计等需求。

针对传统缝纫机轴承故障诊断方案中存在的信号处理产生模态混叠及检测效果不佳等缺点，本文设计了基于集成经验模态分解(Ensemble Empirical Mode Decomposition, EEMD)和改进粒子群算法的支持向量机(Support Vector Machine, SVM)故障检测方案对其进行优化。将采集到的轴承信号通过改进 EEMD 算法分解，特征值选择能量熵，并作为分类器的输入向量，通过自适应惯性权重粒子群算法优化参数 c 和 g 。经过测试表明，改进的算法可以将故障检测准确率提高到 98.6%，大于标准粒子群算法的 92.25%，保证了故障检测的正确率。

针对生产线工序分配不均衡问题，对标准遗传算法进行改进和研究，提出一种基于改进遗传算法的生产线平衡优化的方案。首先引入基于拓扑排序的编码来保证整个种群有可行解，使用竞争策略，进而能使个体的子代比个体的父代更优秀，再使用保留策略，进而使最好的父代复制到子代。使用多种群之间的种群迁移策略，避免了局部最优来达到全局最优。测试表明，比原先不进行改进的生产线，平衡率提高了近 20%，比传统工程工业优化提高了 10%左右。有效地提高生产效率且保证人员操作的动态调整。

关键词：缝纫机系统、工业物联网、生产线平衡、故障检测、改进粒子群算法

ABSTRACT

Nowadays, with the development of intelligent manufacturing, traditional manufacturing industries need intelligent solutions to achieve industrial upgrading and improve production efficiency. Sewing machine has been a traditional manufacturing industry for more than 100 years. At present, the following problems commonly exist in sewing machine system: failure to synchronize equipment operation parameters and status in time; inaccurate identification of sewing machine equipment faults; uneven process distribution of the whole sewing machine production line, which greatly affects the efficiency of the whole system.

In this thesis, a sewing machine online monitoring system based on IoT technology is designed to address the above shortcomings of the current sewing machine system: a self-developed intelligent gateway is docked with the sewing machine controller, collecting device parameters in real time from the controller terminal, and upload the data to the IoT platform for processing and analysis through Wi-Fi wireless communication technology. During the whole system operation, the real-time collected data and situation will be displayed visually on the web terminal and applet page, providing users with a transparent data and convenient operation experience through the visualization of information. When necessary, the sewing machine can also be controlled remotely. The system realizes the needs of remote monitoring, production line management and fault statistics.

In response to the shortcomings of the traditional sewing machine bearing fault diagnosis scheme, such as the signal processing produces modal confusion and poor detection effect, this paper designs a support vector machine (SVM) based on the integrated empirical mode decomposition (EEMD) and improved particle swarm algorithm. Machine (SVM) fault detection scheme based on the integrated Empirical Mode Decomposition (EEMD) and improved particle swarm algorithm is designed to optimize it. The collected bearing signals are decomposed by the improved EEMD algorithm, the energy entropy is selected for the feature values and used as the input vector of the classifier, and the parameters c and g are optimized by the adaptive inertia weight particle swarm algorithm. Tests show that the improved algorithm can increase the fault detection accuracy to 98.6%, which is greater than 92.25% of the standard particle swarm algorithm, ensuring the correct fault detection rate.

To address the problem of unbalanced process allocation in production lines, the standard genetic algorithm is improved and studied, and a scheme for balanced optimization of production

lines based on the improved genetic algorithm is proposed. Firstly, a topological sorting-based encoding is introduced to ensure that the whole population has feasible solutions, a competition strategy is used, which in turn enables the children of an individual to be better than the parents of an individual, and then a retention strategy is used, which in turn enables the best parent to be replicated to the children. Using multiple inter-population migration strategies avoids the local optimum to reach the global optimum. Tests show that the equilibrium rate increases by nearly 20% over the original production line without improvements, and by about 10% over the traditional industrial engineering optimization. It effectively improves the production efficiency and ensures the dynamic adjustment of personnel operation.

Keywords: Sewing machine system, Industrial Internet of Things, Production line balancing, Fault detection, Improved particle swarm algorithm

目 录

摘 要.....	I
ABSTRACT.....	II
目 录.....	IV
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
1.2.1 缝纫机系统发展现状.....	2
1.2.2 故障诊断发展现状.....	4
1.2.3 生产线平衡发展现状.....	5
1.3 本文的主要工作及章节安排.....	6
第二章 缝纫机在线运维监测系统设计.....	7
2.1 系统总体设计.....	7
2.1.1 需求分析.....	7
2.1.2 系统总体框架.....	8
2.2 物联网网关设计.....	9
2.2.1 无线通信方式的选择及介绍.....	9
2.2.2 网关硬件设计.....	10
2.2.3 网关软件设计.....	12
2.2.3.1 网关软件整体设计.....	12
2.2.3.2 网关联网模块设计.....	14
2.2.3.3 网关数据处理.....	16
2.3 通信协议设计.....	18
2.3.1 与下挂设备通信协议设计.....	18
2.3.2 与设备云通信协议.....	20
2.4 应用层设计.....	21
2.4.1 云平台设计.....	21
2.4.1.1 设备云.....	21
2.4.1.2 业务云.....	22
2.4.2 Web 设计.....	23
2.4.3 小程序设计.....	24
2.5 整体效果展示.....	25

2.5.1 设备部分效果展示.....	25
2.5.2 云平台部分效果展示.....	25
2.5.3 Web 端部分效果展示	26
2.5.4 小程序部分效果展示.....	28
2.6 本章小结.....	29
第三章 基于 EEMD 和改进粒子群算法的 SVM 故障检测方案.....	30
3.1 缝纫机轴承故障分析.....	30
3.2 故障检测方案分析.....	31
3.2.1 方案介绍和分析.....	31
3.2.1.1 EMD 分解和小波变换.....	32
3.2.1.2 RBF 神经网络	32
3.2.2 优化方向分析.....	32
3.3 基于 EEMD 和改进粒子群算法的 SVM 故障检测设计与实现.....	33
3.3.1 总体流程设计.....	33
3.3.2 信号预处理.....	34
3.3.2.1 EEMD 分解算法	34
3.3.2.2 EEMD 阈值降噪处理	35
3.3.3 特征提取.....	38
3.3.4 基于改进粒子群的 SVM 算法.....	42
3.3.4.1 SVM 分类器	42
3.3.4.2 粒子群算法.....	44
3.3.4.3 SVM 参数意义	45
3.3.4.4 改进粒子群优化 SVM 参数.....	45
3.4 测试和结果分析.....	49
3.4.1 基于 EMD 和 EEMD 算法故障识别结果对比	50
3.4.2 基于 EEMD 的优化参数 SVM 故障识别结果对比.....	50
3.5 本章小结.....	56
第四章 基于改进遗传算法的生产线平衡问题优化.....	58
4.1 生产线流程分析.....	58
4.2 标准遗传算法方案分析.....	61
4.2.1 标准遗传算法步骤.....	61
4.2.2 优化方向.....	62
4.3 改进遗传算法的生产线平衡优化设计与实现.....	62
4.3.1 总体流程设计.....	63
4.3.2 初始化种群和选择实现.....	64
4.3.3 交叉、变异操作实现.....	65

4.3.4 译码实现.....	67
4.4 测试和结果分析.....	68
4.4.1 基于 Jackson 平衡算例问题验证.....	68
4.4.2 缝纫机生产线平衡改进遗传算法优化结果.....	71
4.4.3 优化结果分析.....	74
4.4.3.1 确定工作站个数.....	74
4.4.3.2 确定工作站结果分析.....	76
4.4.4 整体结果分析.....	76
4.5 本章小结.....	77
第五章 总结与展望.....	78
5.1 总结.....	78
5.2 展望.....	79
参考文献.....	80
致 谢.....	84
附 录.....	85

第一章 绪 论

1.1 研究背景与意义

自缝纫机发明两百多年以来，给人们的生活带来了巨大变化。上世纪 90 年代，随着我国工业迅速发展，从原先简单的工序发展到现如今功能各异，设备齐全的生产线。相应地，我国缝纫机行业也发展迅速，目前，缝纫机出口已占世界的八成以上^[1]。但是，至今我国出口的缝纫机仍基本集中在中低端。同时，缝纫机行业同质化严重，在机械方面进行创新提高，难度很大。为了适应当前经济发展，下游企业更加注重“质”而不是“量”。在工厂中，因缝纫机分布范围大，生产环境恶劣，不能实时、准确地获取缝纫机设备的生产情况，难以对缝纫机整体生产线情况进行监控而花费巨大人力成本。

近些年，中国平均劳动力成本逐年上升，导致企业需要通过各类智能技术来提高生产效率和降低成本。且随着中国制造 2025 的提出，国家更加大力支持和发展产业数字化。在《中国十四个五年规划和 2035 年远景目标纲要》^[2]中，提出要大力推动缝纫机械等产业向智能化方向发展，完成产业新的突破。在此背景下，缝纫机设备对数字化，智能化的升级需求日益提升。数字化赋能缝纫机设备示意图如图 1.1 所示。

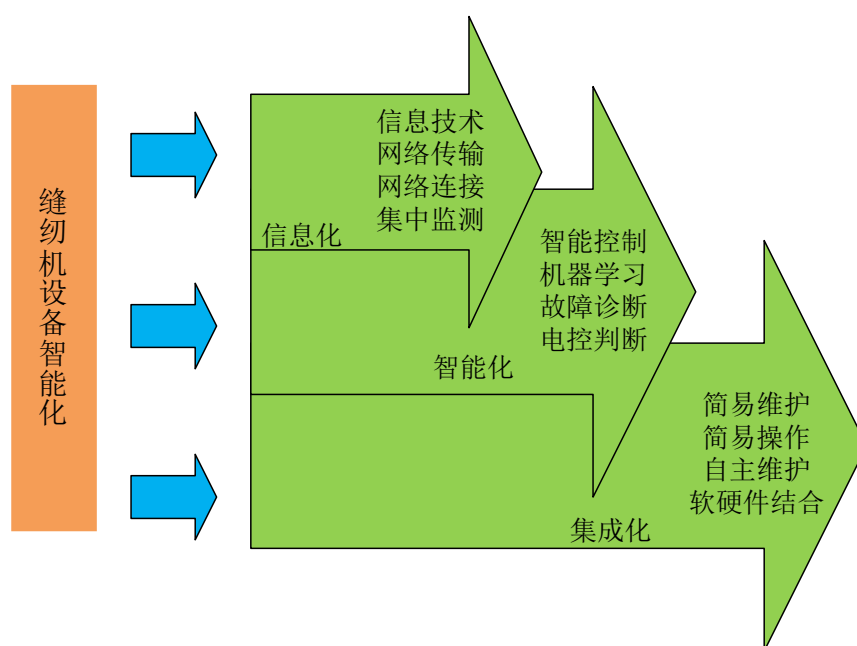


图 1.1 数字化赋能缝纫机设备示意图

在各数字化技术赋能下的缝纫机设备主要以信息化，智能化，集成化升级为主，其中，信息化升级可使缝纫机设备实现无线网络互联，对大量缝纫机设备与数据进行控制和监测；

集成化集中体现在各部分都是智能的且通过信息化形成一个整体，并使业务和实际生产成为很好的闭环。同时缝纫机系统可通过结合机器学习等技术进一步提高设备自动化水平和降低维护难度。因此，在现有的基础加入物联网技术和人工智能算法来提升产能和生产效率就显得极为重要。并且以此形成的数字经济也能够对中国工业的发展提供新的动力。

当前工厂仍存在一些困扰企业发展的痛点问题，较为突出和常见的，例如：

1. 管理人员对生产线工作效率不清晰，不能对各个工序的进展有很好的把握。
2. 在切换生产线同时，需要对生产线上的设备手动逐台进行参数更新，效率低下，或者大批量更新某设备参数时，操作繁琐。对设备参数信息不能进行实时获取。
3. 凭借经验对生产工序顺序随意安排，导致各工序生产效率不平衡的问题，没有一个标准来指导改进。
4. 根据经验性判断运转的设备是否出现故障，准确率较低，导致生产效率受到影响。
5. 工厂时常面临着工价基于经验设定，实际差异较大；工时基于样衣采集，实际生产因素考虑少的情况^[3]。需要有标准来指导矫正标准工时和工价。

因此，本文根据上述现有工厂出现的一些问题进行改进优化，实现了一套缝纫机生产线在线监测的系统，来满足工厂使用者和管理员对工厂缝纫机的远程监控，数据分析的需求，同时还着重对其中工厂遇到急需解决的痛点进行优化：

1. 在缝纫机正常工作过程中，能及时对故障问题进行准确的识别，保证机器能在运行时正常工作，出现问题时能立刻停止工作，减少机器损伤，并上报故障信息。
2. 为了能使缝纫机在加工作业时能更大提升生产效率和节省成本，需要对生产线安排进行优化，做到对时间和效率共同利益的最大优化。

1.2 国内外研究现状

1.2.1 缝纫机系统发展现状

自 18 世纪末世界上第一台手摇缝纫机诞生以来，标志着缝纫设备从手工时代跨入了机械时代^[4]。近四十年来，缝纫设备的用途越来越广，从简单缝制衣服到现在能够满足不同形式缝纫要求；缝制速度也从之前需要的几天到现在的几分钟，缝制速度大大提升，同时缝纫机已经在安全性，可靠性方面有着极大提升^[5, 6]。

缝制机械又可以根据工序分为缝前设备，缝中设备，缝后设备。缝前设备主要是针对缝制所需要的材料进行整理和裁剪。缝中设备是整个系统最重要的部分，通过平缝机，包缝机等不同的缝纫机种类，可以实现不同的缝制需求，并且不同种类的机种能相互配合使用，模块化工作。当前对缝纫效率提升主要通过对缝中设备进行智能化升级，在单台设备上完成之前多项工作任务。最后将缝制完成的产品进行熨烫和包装。这样就形成一套完整的流水线工作流程。现代缝制系统示意图如图 1.2 所示。

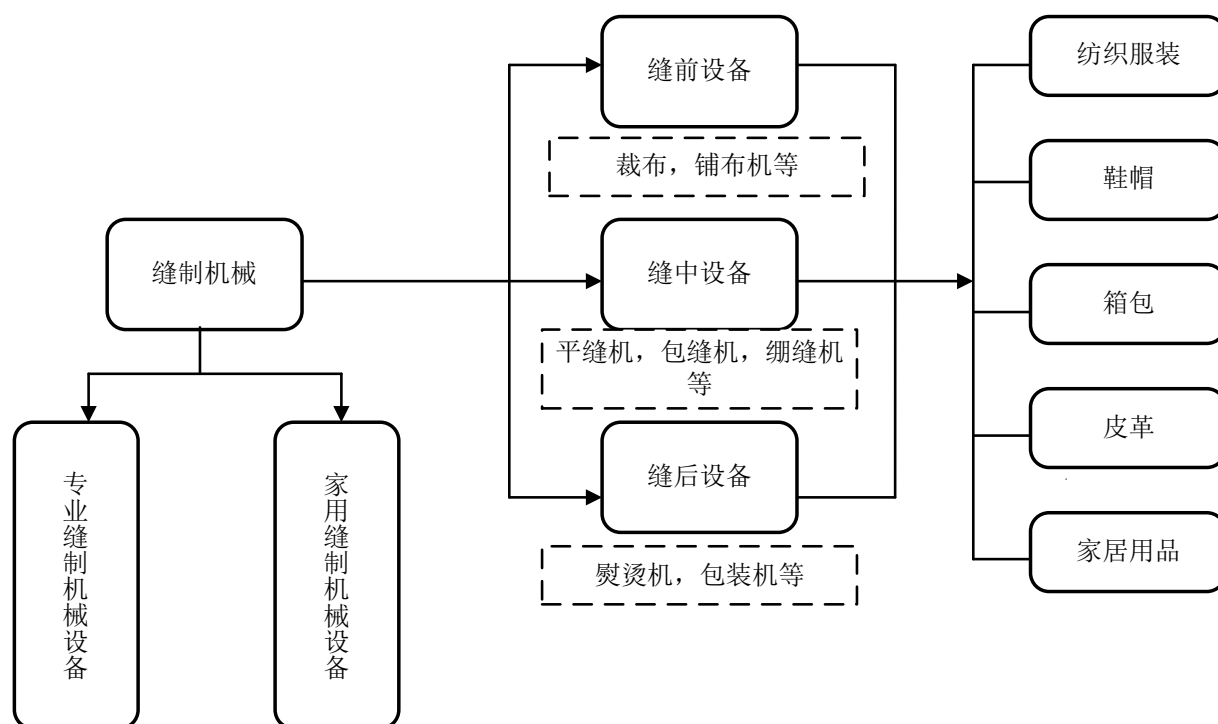


图 1.2 现代缝制系统示意图

如今缝纫机的种类多种多样，同时缝纫机也可根据功能不同组成了各式的生产线。国内，杰克缝纫机作为缝纫机龙头公司，近些年加快数字化转型，建立云端统一数据库，将整个生产流程统一规划和控制。并辅以物联网技术打造智能车间，实时掌握进度，使生产可知，可视化。但是国内大部分工厂或者研究仍局限于单台缝纫机设备的自动化升级过程，并没有对整个缝纫机生产线做一个统筹的利用。例如：吴德文在缝纫机远程监控系统中采用主从网络端模式上传数据，但多人不能共享数据或浏览^[7]；刘紫健采用 PID 算法对直流无刷缝纫机进行控制^[8]；罗立峰使用 S 型加速算法对缝纫机自动化进行优化，但是并没对整个模板化安装实现自动化^[9]；赵新潮采用基于模糊控制对缝纫机进行智能化控制，但没有对控制的数据实现实时展示和参数反馈^[10]。

国外在这方面相对于国内做的更突出一些，例如日本的兄弟工业株式会社不但对单台缝纫机进行智能化改进，既有大屏展示，又能对整个生产线实时监测并进行远程控制，但是只对数据进行了采集，并没有对数据进行分析处理。Olga Kadnikova 采用 DSP 对缝纫机控制部分进行改进，来达到产品设计要求^[11]。Adel 等人采用的是 PID 补偿的自适应控制^[12]。Carvalho 等人对于送料机采用模糊 PID 控制算法^[13]。国外主要是针对缝纫机本身进行改进，并没有考虑到整个生产流程中的智能化。

综上所述，目前无论是国内和国外市场，主要方向都是发展智能化缝纫机，但是对工厂而言，不仅是单台缝纫机实行智能化，更需要的是整个缝纫机系统体系的智能化，通过无线通信技术实时获取缝纫机的运行状态及参数，并进行完善的统一管理，在特定情况下，对设备进行远程控制并进行数据分析。

1.2.2 故障诊断发展现状

对于轴承的故障诊断研究开始于上世纪 60 年代。随着技术的发展,诊断的技术也不断的创新发展。近些年,对于故障诊断主要分为两个步骤,首先是信号的预处理,再是模式识别。前者还包括特征提取,后者更多的是对故障的诊断。但是在诊断过程中,两者需要紧密相连。

在信号预处理方面,起初研究过程中,通常是对获得信号的均值,方差等以时间序列模型分析后的参数,根据时域的公式进行分析,通过时域特征参数对故障进行识别^[14];再之后,基于频谱分析的方法得到广泛应用,此时,轴承的故障判断标准为根据将变换之后的频域信号求出某些数学指标,例如:功率或峰值等,只要不达标就被认为故障^[15]。但是该方法的缺点就是准确性不高。Wei 等人对在变速下的轴承进行分析,将采集信号从时域不是转换到频域而是角域,目的是减少轴承转动速度不均匀的影响,并对各阶次信号定量进行分析并建立模型仿真^[16]。Hasan Yetis 等人将电机轴承的信号和预期的信号做一个差分处理,经过滤波器得到的值应用于特征提取^[17]。Li 等人提出了采用改进的小波包分析方法,使用多层小波包算法对信号分解,将能量占大比重的信号单分支重构分解的信号,提取小波包能量熵,形成特征向量^[18]。

在故障诊断模型方面,李海涛在其文章中,对 SVM 进行改进,使用的是基于量子万有引力搜索算法^[19]。为了提高万有引力算法对参数收敛的情况,将两者算法进行整合来解决 SVM 模型过拟合问题。国红波使用径向基神经网络模型对缝纫机分解信号各频段能量,关联到奇异熵建立特征向量到故障种类的映射来进行故障诊断^[20]。Song Xudong 等人提出了一种基于长短期记忆网络训练故障诊断模型的故障诊断算法,利用少量目标域数据进行微调模型参数^[21]。最后,构建了基于改进弹性网络迁移学习的变负载工况故障诊断模型。Wang 等人针对实际工业应用提出了一种新颖的三阶段智能故障诊断方法,提出了用于聚类的自适应密度峰值搜索(ADPS)算法,该算法可以自适应地确定截止距离参数,取得了优于 K-means 等常用聚类算法的效果^[22]。

综上所述,信号预处理有很多方法,但是在处理信号时,时域和频域的方法并不能很好地处理非平稳信号,小波变换在处理时需要选择合适的小波基函数和分解次数,不能做到简单易行;同时在故障诊断模型方法,神经网络模型在一定程度上可以很好进行诊断,但对于参数调整还是繁琐的,在初始化时需设置阈值及权值的大小,在训练中也需不断进行调整;而且训练过程不能很直观地进行观察;模型训练时间过长会出现过拟合情况。聚类算法对某些值较敏感。所以需要选取一种既能处理非平稳信号,又能自适应分解信号的信号提取方法;通过提取的特征,能建立简洁清晰的模型对数据进行分类。

1.2.3 生产线平衡发展现状

生产线是现代工厂中主流的生产方式，下游工厂通过对原材料的机械加工，完成一系列的模块化工作等生产过程^[23]。生产线平衡问题是企业生产效率提高必须要解决的问题。当生产线中各工作站作业周期大致平衡时，就可以极大地降低由于作业周期不一致导致的时间浪费，保证工厂生产维持连续、均衡的工作，更好地获得企业的生产效益^[24]。

研究生产线平衡的意义在于能够有效地降低企业的生产成本，同时也可以提高生产的效率和产品质量。这对企业来说是个能获取利润的最重要途径之一。对于生产线平衡问题，也有几种不同的解决方法来处理，分别是最优化方法，启发式方法和元启发式算法。

最优化方法：对需解决的问题进行抽象化处理，抽象成一个适合的数学模型并求得结果。殷旅江等人的论文综合了汽车生产线装配线，结合各个工作的流程和工序之间的互斥关系，使用第二类生产平衡线的解决方法，建立数学优化模型来进行求解^[25]。在国外也有相关的研究，Bukchin Y 和 Raviv T 在 2018 年提出使用约束规划来解决生产线平衡的问题，作为一种通用的解法，在生产线任务较多情况下，能有更好的性能，能得到相对来说更好的最优解^[26]。

启发式方法：该算法的基础是根据经验来构造的，在给定的范围内，得到问题较好的一个解。同时，该方法也存在缺点，不一定可以得到最优解，因此常与其他算法结合使用，用来改进算法的效率^[27]。国内方面，李明等人首先采用工作时间定界压缩规则等四种方法融合解决第二类生产线平衡问题^[28-30]，在这之后，提出工作交换规则来优化之前的组合算法，虽然说优化手段比较简单，但是对数据量非常大的装配线平衡问题还是可以求得比较好的解。相比于国内，在国外，Ibrahim Kucukkoc 针对混合装配线提出使用新型蚁群优化算法，并在大部分测试中找到接近最优的解决方案^[31]。再之后由 S.G.Ponnambalam 根据流行的 6 种启发式算法进行总结分析并进行评估，最后使用遗传算法来求解问题^[32]。

元启发式方法：是对启发式算法的改进，主要思想也是对局部搜索算法的优化，对问题有更全面的扩展。元启发式算法包括一般为群体算法或者智能算法等^[33]。在国内，毛凌翔等人使用蚁群算法，通过贪婪和迭代对第一类生产线平衡进行了求解，再比较局部和全局的信息，综合得到一个较优的解^[34]。蒋玲玲等人在解决第二类平衡问题时使用混合粒子群和模拟退火相结合的算法，通过这两个算法的结合不但保留了混合粒子群的优势，还能避免得到的只是局部最优，提高了算法的通用性^[35]。张晨威对比各种算法之后，采用了改进人工鱼群算法，并建立 MATLAB 仿真模型，通过搜索生产过程中最优的生产节拍，来确定生产优先的关系，并使用改进的鱼群算法进行编码来进行优化^[36]。

综合上述论文，可以得出的结论是，要根据不同的生产线类型来选取不同的算法。而这些年采用的元启发算法在这些论文中体现出的是需要应用到更大的生产线规模中去，而不是局限于部分的生产线或者中小规模生产线。在改进算法的过程中，要注重算法的稳定性和适用性。

1.3 本文的主要工作及章节安排

本文基于当前缝纫机系统中需改进之处，根据目前常用的无线通信应用并结合工厂的实际情况，对其进行优化调整，设计了基于 Wi-Fi 的缝纫机在线监测系统。实现对缝纫机数据展示，远程控制，故障检测及生产线管理等功能，通过自主研发的网关将缝纫机接入互联网中，设计云平台，Web 端和小程序，实现了对数据采集上报，给予用户一定的反馈体验。同时，通过对缝纫机轴承振动信号进行处理和分析，采用基于 EEMD 和改进粒子群的 SVM 算法对缝纫机的轴承进行故障检测，保证了系统能够稳定正常的运行。并根据生产线的进度，设计了一种基于改进的遗传算法来优化整个生产线的工序时间分配不均等问题，既提高了生产效率，又有效地减少工厂生产成本。全文共分为五章，主要研究内容如下：

第一章：绪论，介绍论文的研究背景及意义，对当前缝纫机系统发展做了简要概述，总结了现有研究方案的不足，分别从缝纫机系统，故障诊断，生产线平衡三个方面，阐述了国内外研究现状并进行分析，为接下来设计提供方向。

第二章：缝纫机在线监测系统设计，根据系统的需求，设计系统总体框架。并根据端和云的顺序对整体系统进行分模块讲解。首先对网关的硬件及软件设计进行详细设计阐述，在应用层方面，对云平台设计和 Web 端及小程序的设计框架进行介绍。最后对整体系统效果进行实物展示。

第三章：基于 EEMD 和改进粒子群算法的 SVM 故障诊断方案，本章首先对缝纫机轴承故障展开分析，接着对传统的故障诊断方案进行分析，针对方案中的信号分解和分类算法不足改进，并设计了新的故障诊断方案。使用 EEMD 对信号进行分解，通过对比选取预处理信号的能量熵作为特征向量，并结合自适应惯性权值优化的 SVM 分类算法进行故障诊断，最后对优化前和优化之后的方案进行评估测试。

第四章：基于改进遗传算法的生产线平衡问题优化。首先以衬衫生产线流程为例展开分析，再对标准的遗传算法进行介绍剖析，针对标准遗传算法的不足之处，应用改进的遗传算法对该生产线问题进行解决，并用经典算例验证算法的有效性，最后根据实际生产线进行优化测试。

第五章：总结与展望，本章对缝纫机在线监测系统总体设计和使用的关键技术做出总结，并就本设计方案中存在的不足之处，提出分析和展望。

第二章 缝纫机在线运维监测系统设计

针对当前缝纫机工厂出现的对数字化工厂的需求，且不能及时掌握缝纫机工作时的运行状态，故障检测功能不完善，生产线不平衡等一系列问题。本文提出并设计一种基于无线通信技术的缝纫机运维在线监测系统。通过定制自主开发的物联网网关和工厂各种缝纫机进行数据交互，对缝纫机运行时重要参数进行实时获取，并通过无线通信技术上报到物联网云平台。平台对数据进行处理和整合之后，将实时数据及生产线信息通过 Web 端方式进行展示，对故障预警通过及时通知发送到小程序提醒工厂管理员。便于工厂进行管理和远程控制。

2.1 系统总体设计

2.1.1 需求分析

针对当前缝纫机工厂存在的信息化程度低，故障检测不及时等问题。本文对于要设计的缝纫机运维监测系统，对于需求分析整理如下所示：

1. 实时监测，能及时获取到设备运行时详细信息参数，例如机身编号，机型，参数设置等。还可以获取该设备的历史记录，能对过去任意时间段内生产信息进行查询，包括任务完成数量，设备号等，以及该设备过去故障报警查询。
2. 联网稳定，智能终端的连接是多频段连接，有多种措施保证可以正常连接网络。当连接断开时，能保证 30s 内进行重新连接。且能缓存断网时数据，在重新连接网络时，将之前未上报的数据进行发送。
3. 远程控制，可以对监测的设备进行开关机操作，充分考虑安全情况。对设备参数可以批量设置，包括针距，针数，缝纫模式等重要参数。不仅可以帮助生产人员在更换设备时，批量调节并设置工艺参数、习惯参数，控制订单品质。也能不在现场时，通过手机远程控制、批量下达指令。
4. 系统运维，对设备故障进行实时预警，自动通过小程序提醒管理人员，并对故障进行分类统计，有助于对故障进行智能化检测功能。还可以对设备进行保养管理，对设备过去保养情况进行查询。
5. 生产线管理，对生产过程中工序进度进行实时上报，对“瓶颈工序”及时把握，快速找到效率低的工序，为制定生产线平衡方案提供数据基础。
6. 数据可视化分析，选取相应模块，投放到大屏可以动态查看数据。对设备产能信息以图表形式进行展示，更加清晰明确，更加工厂需求，分析过往数据，提出与预测性建议。对历史数据进行占比统计分析。

2.1.2 系统总体框架

依照以上的需求分析，对缝纫机运维在线监测系统进行了整体设计，系统设计架构图如图 2.1 所示。

整体架构大体分为云和端两侧，端又分为数据端和采集端。物联网网关属于采集端，通过 RS485 和工业缝纫机进行数据通信，数据端是 Web 端和客户端，将采集到的数据以可视化方式展现给用户和管理员，且可以通过数据端对采集端进行命令下发控制。云平台是两个端数据交互媒介，通过 MQTT 物联网协议进行通信，且承担业务处理，数据分析等功能。

1. 采集端：通过硬件层的通用 RS485 接口和工业缝纫机进行数据通信，通信协议采用的是自有的数据协议和 Modbus 通用协议进行对接的。对采集到的数据先进行边缘性的数据处理，之后通过无线通信技术上报至云平台。物联网网关和云平台进行的长连接，保证数据收发的持续性。
2. 云平台：云平台根据业务功能可以分为设备云和业务云两部分。设备云主要选用移动 OneNET 物联网平台和电信云平台，负责和物联网网关进行对接，对上报的数据以监控点形式展示且存储。业务云主要负责对业务逻辑进行处理，通过设备云提供的接口从设备云获取数据信息，并同时提供 API 接口供客户端进行调用。
3. 数据端：通过 Web 端和移动客户端的形式对数据进行可视化展示，包括一些历史数据，设备管理和故障预警信息等，并提供使用者在某些特定情况下进行远程控制等功能。

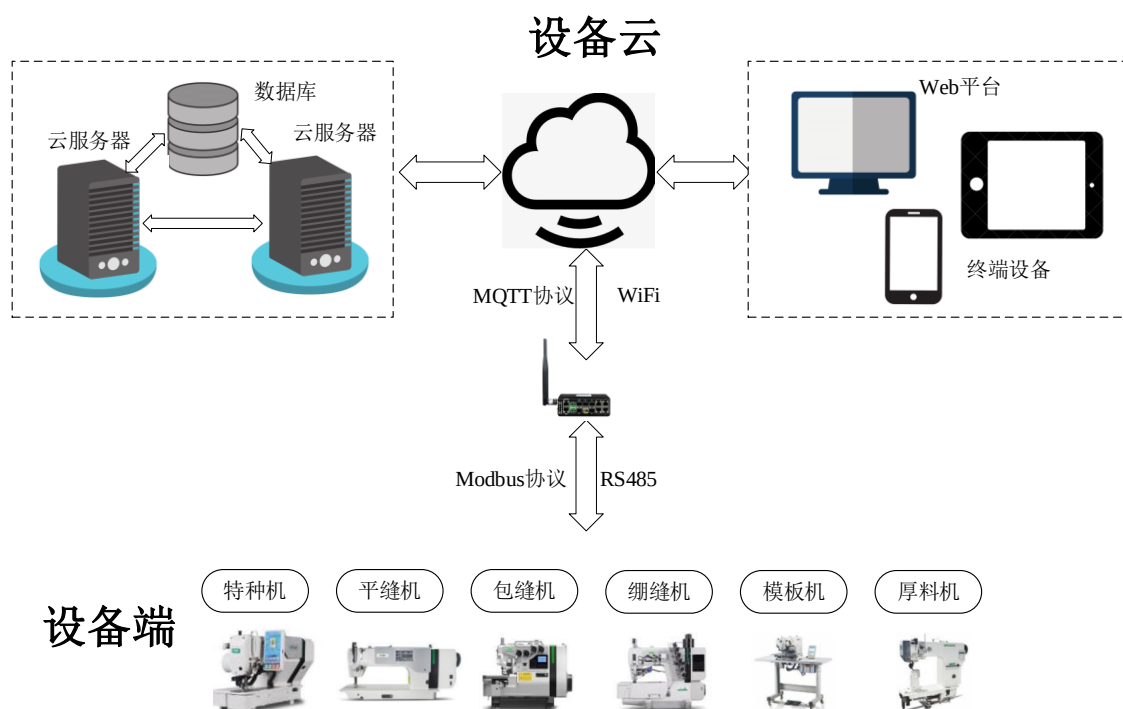


图 2.1 缝纫机在线监测系统架构图

2.2 物联网网关设计

物联网网关是整个系统最关键的部分，通过与下挂设备使用硬件接口进行交互。再将数据进行一个整合处理，与设备云通信进行数据上传，同时又是设备云端下发指令的有效通道。本节就无线通信方式的选择，硬件部分设计和软件设计做一个详细的阐述。

2.2.1 无线通信方式的选择及介绍

当前，物联网技术中常用的无线通信技术分别是 ZigBee，Wi-Fi 和 NB-IOT。ZigBee 作为短距离无线通信技术，优势非常明显，可以在各种局域网设备之间都能进行数据交互^[37, 38]，另外，不需要频繁的发送数据，间断性发送数据节省功耗也是另一点优势。Wi-Fi 在我们实际中也应用广泛，其由于有着传输速度快，覆盖面广，使用简单，使用的频段不受限制等多种特点^[39, 40]，成为物联网开发者最常用的通信方式。NB-IOT 作为近些年兴起的另一种物联网技术，凭借着窄带宽和大规模的节点容量迅速占据了市场部分份额^[41, 42]，但在实际使用中，受限于窄带通信的缺点，通信速率也是相对于其他来说较慢，不能有效做到大量数据的同时上传和下发。三种无线通信对比如表 2.1 所示。

表 2.1 物联网中常用的无线网络通信

	ZigBee	Wi-Fi	NB-IoT
组网方式	基于 ZigBee 网关	无线路由器	移动网络
部署方式	中继+节点	路由器+节点	节点
网络延时	不到 1s	不到 1s	6s-10s
网络节点容量	200-500 个	50 个	20 万个
传输距离	<10m	<50m	>10Km
频段	未认证 2.4G 频段	2.4G 和 5G	认证频段 运营商频段
传输速度	理论是 100kbps	2.4G: 1M-11M 5G: 1M-500M	理论低于 100kbps, 受限于串
适用场景	室内	室内室外均可	室外, 大规模传感器网络

根据以上对无线通信技术的分析，再结合工厂现有的条件情况。本文选择的是双频 Wi-Fi+蓝牙的通信方式来接入设备云。该技术相比于 ZigBee 有着更广的通信范围，且覆盖范围更广。相比于 NB-IOT 技术，Wi-Fi 有着更快的上报速率。同时双频 Wi-Fi 分别有 2.4GHz 和 5GHz 频段，既可以向下兼容旧的协议和路由器，又能在一个路由器下尽可能多的接入设备。Wi-Fi 也是我们日常常用的无线通信技术，在易操作性和区域覆盖性有着巨大优势。现对 Wi-Fi 模块的各种模式进行介绍。

1. STA 模式：STA 是 STATION 的缩写，原意是站点。在该模式下，Wi-Fi 模组是作为一个网络中的终端连入网络中，模组作为一个客户端和路由器进行通信。STA 模式示意图如图 2.2 所示：

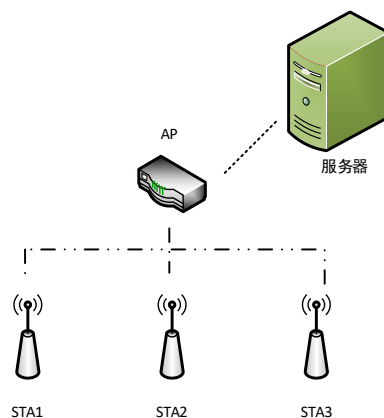


图 2.2 STA 模式示意图

2. AP 模式:AP 是 Access Point 的缩写,是作为一个小的局域网中的集中器,作为网络结构中的服务器和下面的设备进行通信。AP 模式示意图如图 2.3 所示:

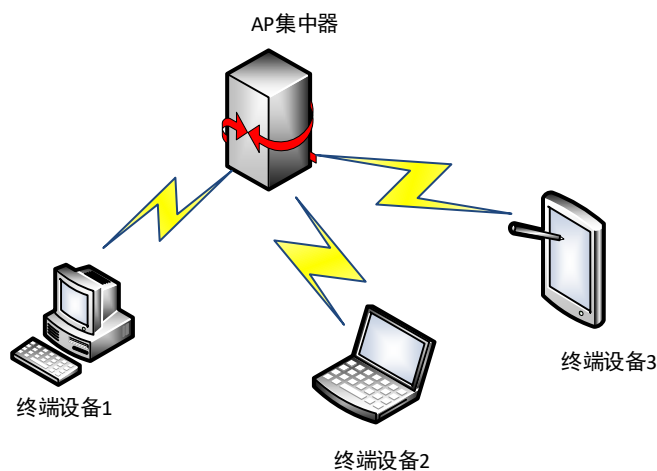


图 2.3 AP 模式示意图

3. STA+AP 模式:该模式是上述两种模式的混合模式,此模式的前提是该模组有两张网卡,通过对不同网卡配置不同的 IP 地址,可以实现这种模式。

2.2.2 网关硬件设计

网关硬件模块主要包括:Wi-Fi 模组,MCU 模块,RS485 模块,传感器检测模块,串口通信模块等。主控芯片通过和无线通信模组以串口相连,通过配置 AT 命令控制模组,再通过 RS485 和下挂设备进行异步传输通信。传感器检测模块通过 UART 与主控芯片相连。通过 MCU 控制 LED 来指示网关的联网状态和下挂设备通信状态。按键模块可用作对网关进行重启操作和 AP 配网模式的选择。物联网网关硬件设计模块图如图 2.4 所示。

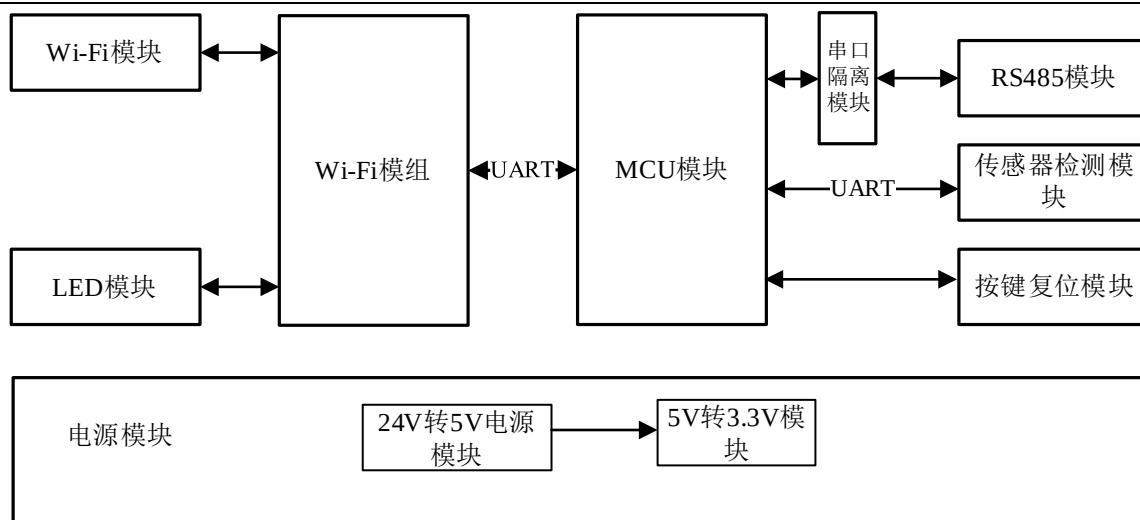


图 2.4 网关硬件模块设计框图

Wi-Fi 模组：采用的是瑞昱 RTL8720D 芯片模组，该模组是双频段+蓝牙模组。支持多种 Wi-Fi 连接模式，是一款低功率芯片。通过对网卡 IP 地址设置可以使模组处于不同模式。可通过该模组将数据上报到云平台。

MCU 模块：采用的是 GD32F103RCT6 芯片，该芯片功能强大可以很好地处理与下挂设备通信和传感器的数据，例如，将串口和 RS485 芯片 IO 管脚相连，方便收发数据。在接收数据时，用到 DMA 缓存队列等。

RS485 模块：该接口与缝纫机设备相连，目的是获取到缝纫机数据。对于工业领域使用，需考虑复杂的应用环境，因此加入光耦隔离部分，增强抗干扰能力。

传感器检测模块：使用加速度计传感器采集轴承振动数据。采用 ACC345 型号的加速度传感器的数字加速度计，通过串口和 MCU 相连。

电源管理模块：主要功能是将适配器 24V 转为 5V，是一个 Buck 电路；再将 5V 转为 3.3V，是一个 LDO 电路。电源部分原理图如图 2.5 所示。

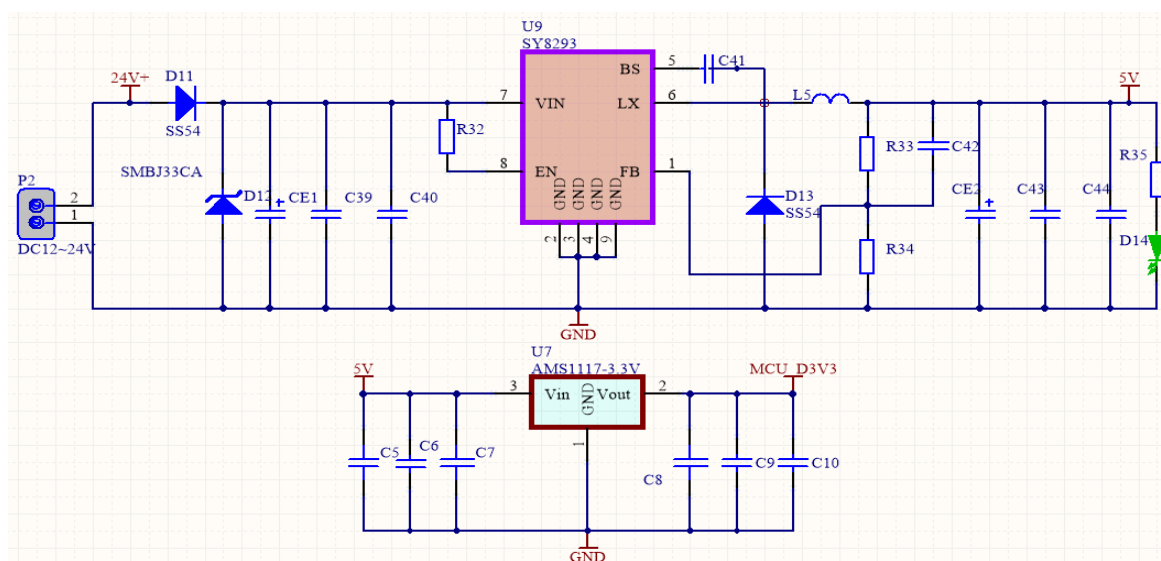


图 2.5 电源模块电路原理图

2.2.3 网关软件设计

本系统在运行过程中，需处理多种不同优先级的事件，既要保证任务的实时性又要保证之前任务准确性。因此有必要引入实时操作系统作为对软件设计的基础支撑。FreeRTOS 作为一套开源操作系统，其功能强大包括任务调度、资源管理、外设配置等^[43]。本系统采用 FreeRTOS，对多个线程进行调度，根据线程的优先级进行分配调度，对中断也能及时响应。具有很好的实时性。

在网关的设计过程中，需要满足以下几种功能：

1. 定时对设备的数据进行轮询，并在云平台控制和设备轮询之间取平衡点，做到能及时响应云端的命令。
2. 为了减少网络通道负载，采用差异化上报的方式，并能保证云端数据成功的处理和发送。
3. 保证网关与设备云之间长时间稳定的连接，当网络连接断开时，需自动重新连接。并在断网时，将需要发送的数据进行缓存直到联网成功。
4. 能够对下挂设备的串口通信做出及时响应和处理。

2.2.3.1 网关软件整体设计

本网关基于 FreeRTOS 操作系统开发，因此根据功能需求，对网关任务分配进行梳理，这样对于实时操作系统分模块开发有着极大帮助。网关程序整体框图如图 2.6 所示：

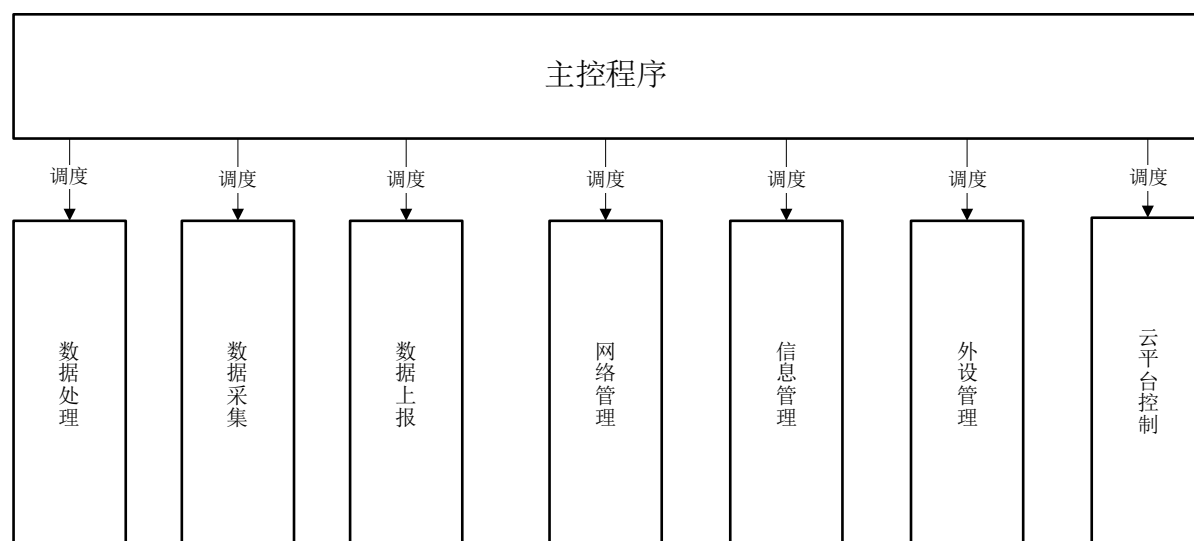


图 2.6 网关程序整体框图

网关应用程序主要由 4 个线程组成，分别是网络管理线程，云平台数据收发线程，RS485 和外部接口数据处理和信息处理线程。这些线程在实时操作系统的线程管理情况下，根据实时性和优先级大小被调度执行，实现上图中网关功能。

在这些功能中，根据优先级排序，网络管理是优先级最高的，若无网络，则无法和设备云进行数据交互。当网络连接成功之后，在初次使用时网关需要注册和连接到设备云服务器。当以上阶段都成功之后，就可以进行有业务层的数据处理。

云平台数据收发包括下发命令的处理和上报的数据处理。通信数据是以 MQTT 格式封装的，本地将接收的数据进行解析操作，并根据不同的指令做出相应的处理，指令包括命令下发，消息订阅，上传成功等。处理之后的数据是 JSON 格式的，根据其规则对数据进行处理，将需要操作的监控点和数据存入到监控点的链表中，以便轮询链表时，对相应的监控点按照指令进行操作。

网关整体工作流程图如图 2.7 所示：

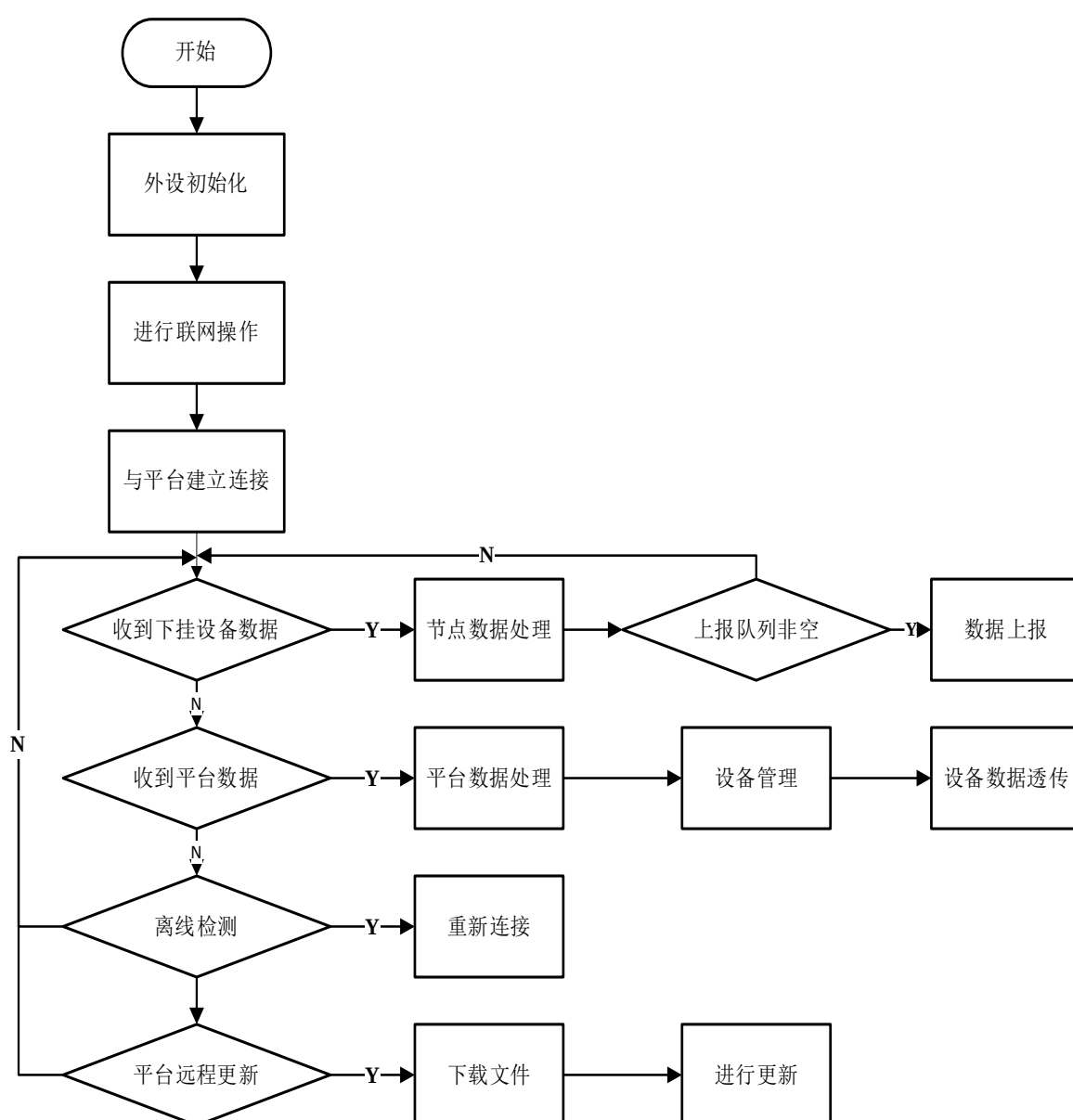


图 2.7 网关工作流程图

2.2.3.2 网关联网模块设计

本系统无线通信采用的是 Wi-Fi 联网的无线通信方式。作为一个 Wi-Fi 模组，可以进入多种模式，同时这也意味着承担着不同的功能。联网是整个数据端最重要的模块，当未能联上网时，所有采集到数据都不能被可视化，且下发的控制命令也不能得到应答。所以必须采取多种措施保证使用者能连上网络。Wi-Fi 配网需要一些重要信息，比如：

SSID：路由器对于用户的一个标识。**PASSWORD：**路由器的进入密码。**CHANNEL：**Wi-Fi 的通信信道，不同的信道代表不同的频段。

本模块设计是通过 3 种方式不同的获取 Wi-Fi 配网信息，当获得配网信息之后，通过 STA 模式接入到可连接的路由器中。分别是本地模式配网，AP 模式配网，蓝牙配网。

1. STA 模式配网：

在该模式下进行配网，首先保证是 STA 模式，再通过读取保存在文件中的 Wi-Fi 信息去连接路由器，服务器会根据当前路由器的局域网的 IP 地址通过 DHCP 服务器对 Wi-Fi 模组分配一个当前的 IP 地址，供给模组进行联网。当模组连接成功之后，会将该 Wi-Fi 信息以文件形式写入到 FLASH 中，保证下次连接时不需其他方式联网，可以自动连接网络。STA 模式连接流程图如图 2.8 所示。

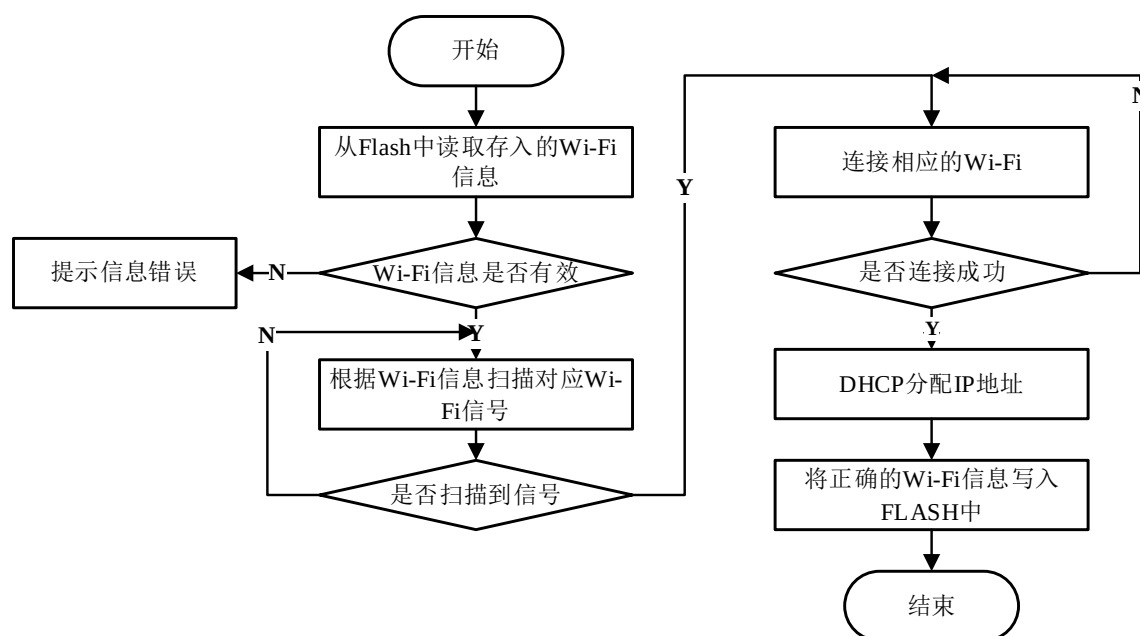


图 2.8 STA 模式联网流程图

2. AP 模式配网：

若通过之前存在文件中的 Wi-Fi 信息不能顺利连上 Wi-Fi。那么采取 AP 模式进行配网。为避免与蓝牙配网冲突，使用时对缝纫机进行连续断电重启 3 次操作（每次操作需在 20s 内完成）来表示进入 AP 配网模式。当进入 AP 配网模式时，用户终端通过小程序会扫描到一个以机身编码为 SSID 的热点名称。当连入此热点，相当于 Wi-Fi 模组和终端设备组成一个局域

网，在局域网中可以通过 UDP 通信，手机终端通过 UDP 将可以连入的 Wi-Fi 信息发给 Wi-Fi 模组。若 UDP 通信失败，通过重发机制，重发 3 次。此时，Wi-Fi 模组收到数据之后，对数据进行解析处理，进入到 STA+AP 模式，进行配网。AP 模式连接流程图如图 2.9 所示。

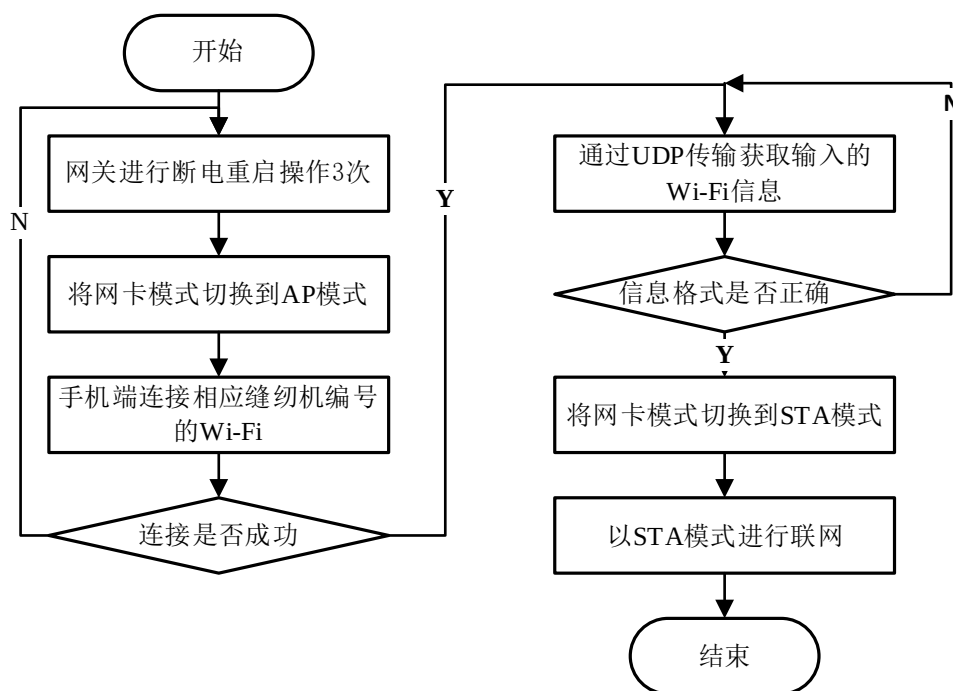


图 2.9 AP 模式联网流程图

3. 蓝牙配网：

此模式和 AP 配网模式都属于是辅助配网，蓝牙配网示意图如图 2.10 所示。

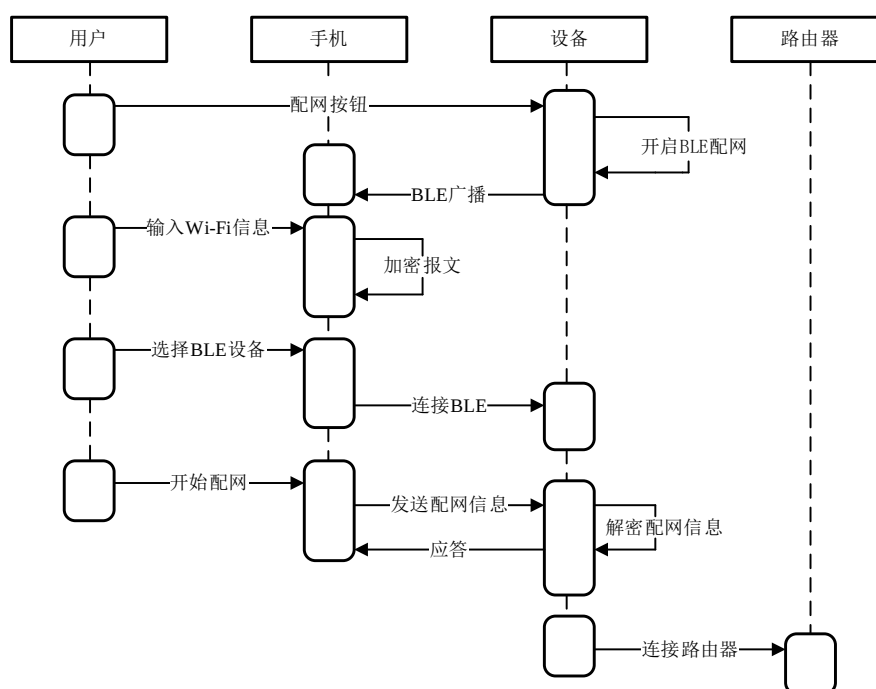


图 2.10 蓝牙配网示意图

蓝牙配网在操作上更加方便可行。Wi-Fi 模组有蓝牙功能，用户可以打开手机终端蓝牙功能，根据缝纫机机身编号进行蓝牙连接，通过蓝牙将 Wi-Fi 信息发给模组设备，进行连接。

一般情况来说，若第一次联网成功之后，再次启动网关，会通过 STA 模式自动连入之前连接成功的 Wi-Fi，若出现连接失败情况，再使用其他方式进行配网。网关整体联网流程图如图 2.11 所示：

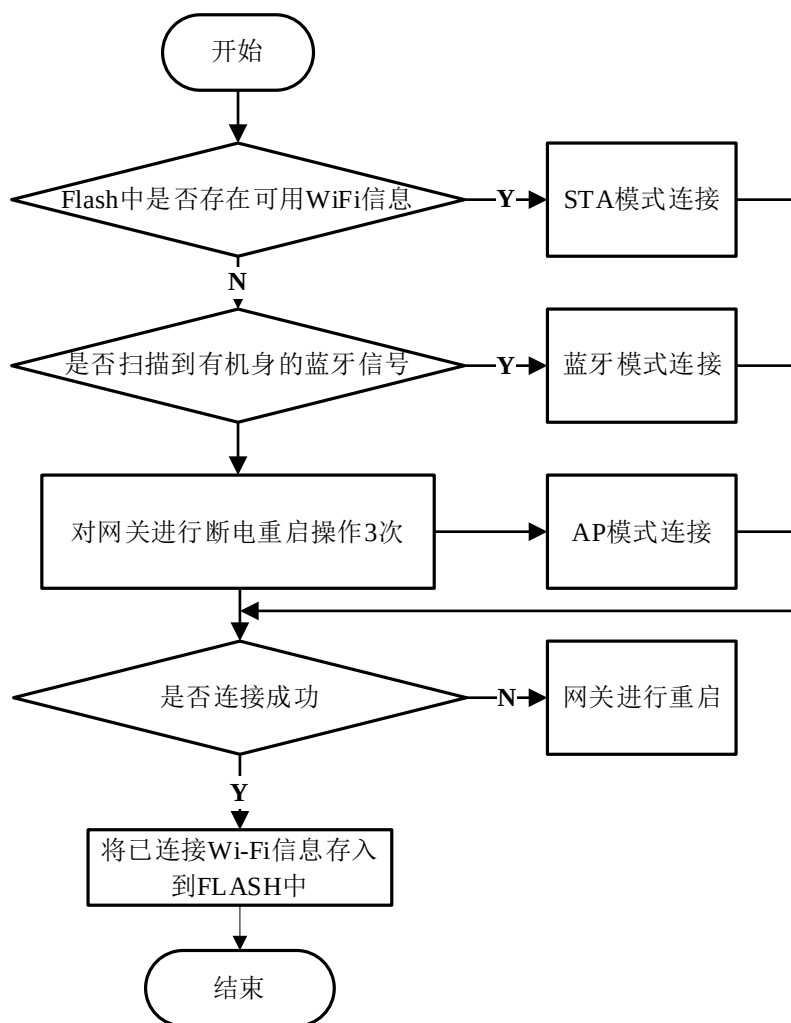


图 2.11 网关整体联网流程图

2.2.3.3 网关数据处理

对于缝纫机数据，将缝纫机数据以监控点形式进行上报和修改。上报的监控点属于读监控点，修改的监控点属于写监控点。写监控点优先级高于读监控点。每个设备下面都会有两种监控点，因此需要维护一个设备链表来方便管理。为适配下挂多个设备的情况，根据边缘计算思想，将一部分数据进行处理之后，再上传至云平台。网关处理数据整体框图如图 2.12 所示，左侧是设备端，将需要上报的数据以 RS485 通信方式上报至网关端，网关端维护一个设备链表，每个设备链表节点开辟一块独立于其他的内存空间，用来存放和处理该设备的参

数信息，达到和其他设备处理解耦效果。在本地对一些数据进行处理完之后，通过 TCP/IP 通信将数据上传至设备云端。

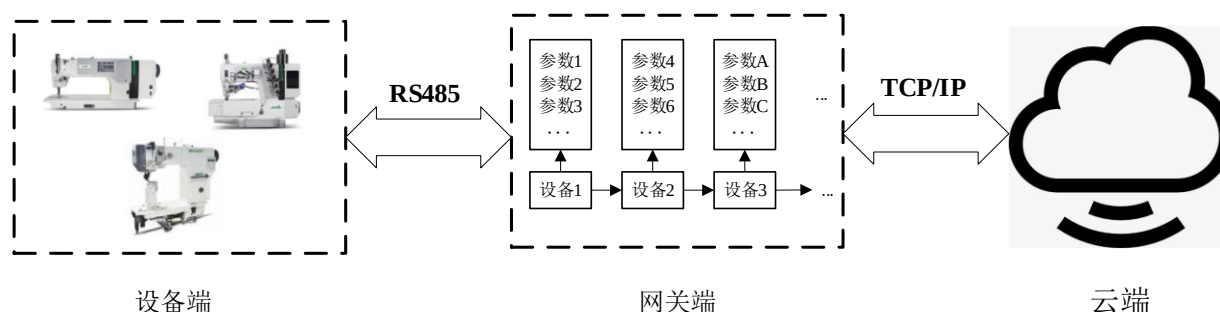


图 2.12 网关数据处理框图

对数据监控点进行读取的操作称为读操作，对数据监控点进行修改的操作称为写操作。读操作是根据设备的节点进行轮询操作，写操作是云服务器有命令下发时进行修改。和下挂设备通信是通过 RS485 设备，因为和设备是问答机制，所以建立一个 RS485 串口接收队列，对接收到的数据进行存储，防止断包。RS485 数据处理示意图如图 2.13 所示。

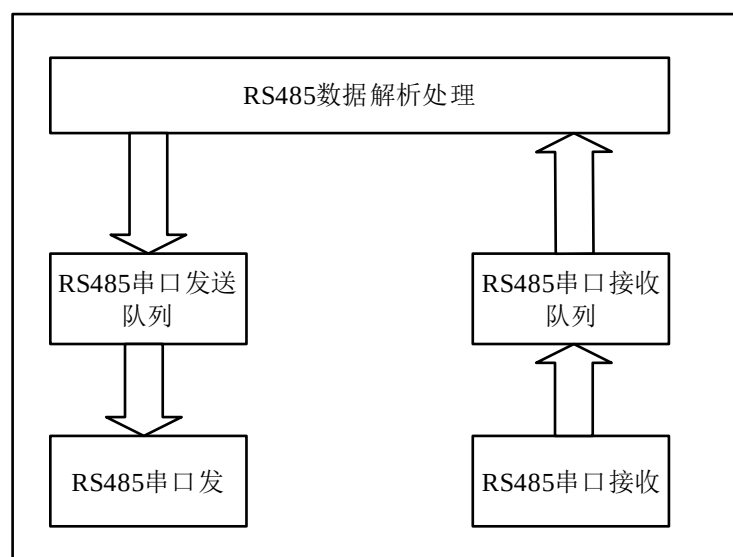


图 2.13 RS485 数据处理流程图

在进行 RS485 数据通信操作前，会获取当前操作的设备节点指针，再从指针指向的设备结构体中，获取读操作监控点指针和写操作监控点指针。定义对监控点修改操作的优先级高于读取操作，所以获取了读操作监控点指针和写操作监控点指针以后，确认此时可以进行数据发送，那么会优先判断写操作监控点指针是不是为空，不为空则会先进行写操作，然后才会进行读操作。

当处于等待回复的状态时，该模块通过确认 RS485 串口接收队列是否有数据，来判断串口是否有回复，如果超过了 3 秒钟，仍没有接收到数据，那么此时判断设备离线，进行下一

台设备的读写操作。如果接收到数据，根据上一次发送的数据类型，对该数据进行分类处理（读操作或写操作），然后指针指向下一个要操作的监控点节点，需要注意的是，需要指向的下一个节点为未在本次连续操作的下一个节点指针。对网关端数据读写流程图如图 2.14 所示：

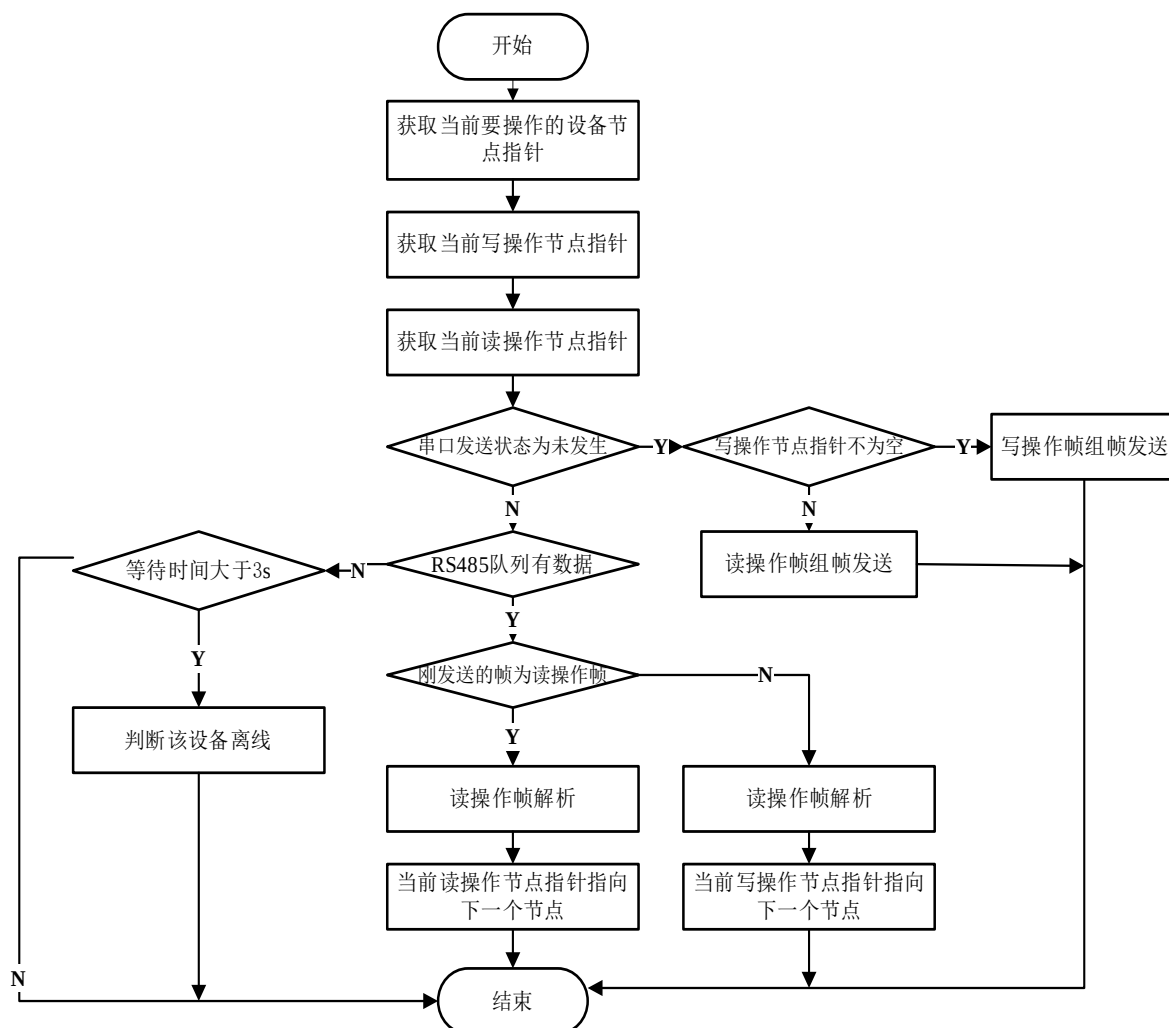


图 2.14 网关数据读写流程图

2.3 通信协议设计

2.3.1 与下挂设备通信协议设计

网关和与之连接的缝纫机电控系统通过 RS485 总线连接，采用的是 Modbus 通信协议。Modbus 是一类工业上常用的串行通信协议。在总线上，不同设备的有两种模式，分别是主机/从机模式，Modbus 因其开源和可支持多种接口，逐渐成为一种标准的工业通信协议。现在工业设备都广泛使用的是 Modbus 协议^[44, 45]。

Modbus 是区分主机和从机的，属于应用层的协议。在本系统中，智能网关属于主机，

下挂的缝纫机控制板属于从机。网关需要不断发送 Modbus 帧数据进行读监控点的查询。Modbus 标准协议，有着 ASCII 格式和 RTU 两种不同的数据格式。不论选择哪种格式，在同一总线上的传输模式一定得是相同。ASCII 和 RTU 均使用大端模式。RTU 格式的 Modbus 帧结构如图 2.15 所示：

地址 8位	功能码 8位	数据域 N×8位	校验位 16位
----------	-----------	-------------	------------

图 2.15 RTU 格式 Modbus 协议帧

在上述的帧格式中，地址一般指的是下挂设备从机的地址，功能码是该数据帧需要完成的功能。例如功能码为 05，表示需要修改单线圈的值，功能码 04，表示需要连续读取寄存器的值。若出现读取错误的情况，功能码的高位置 1，表示此次无响应或者通信出错。前两个字段通常大小和为 2 个字节。数据域值有多种形式，一般会随着功能码的要求而改变，数据域值一般包含是否需要应答，是否有连续读取操作，连续读取数量等。校验的方式方面，RTU 格式和 ASCII 是不同的，RTU 采用的 CRC16 的校验方式，一般高位在后，低位在前。

在本系统中，因为涉及到寄存器数量巨大，且连续读取操作次数也较多，为了方便读取和存储。提出了一种监控点和寄存器的地址自适应的方法，该方法的示意图如图 2.16 所示。

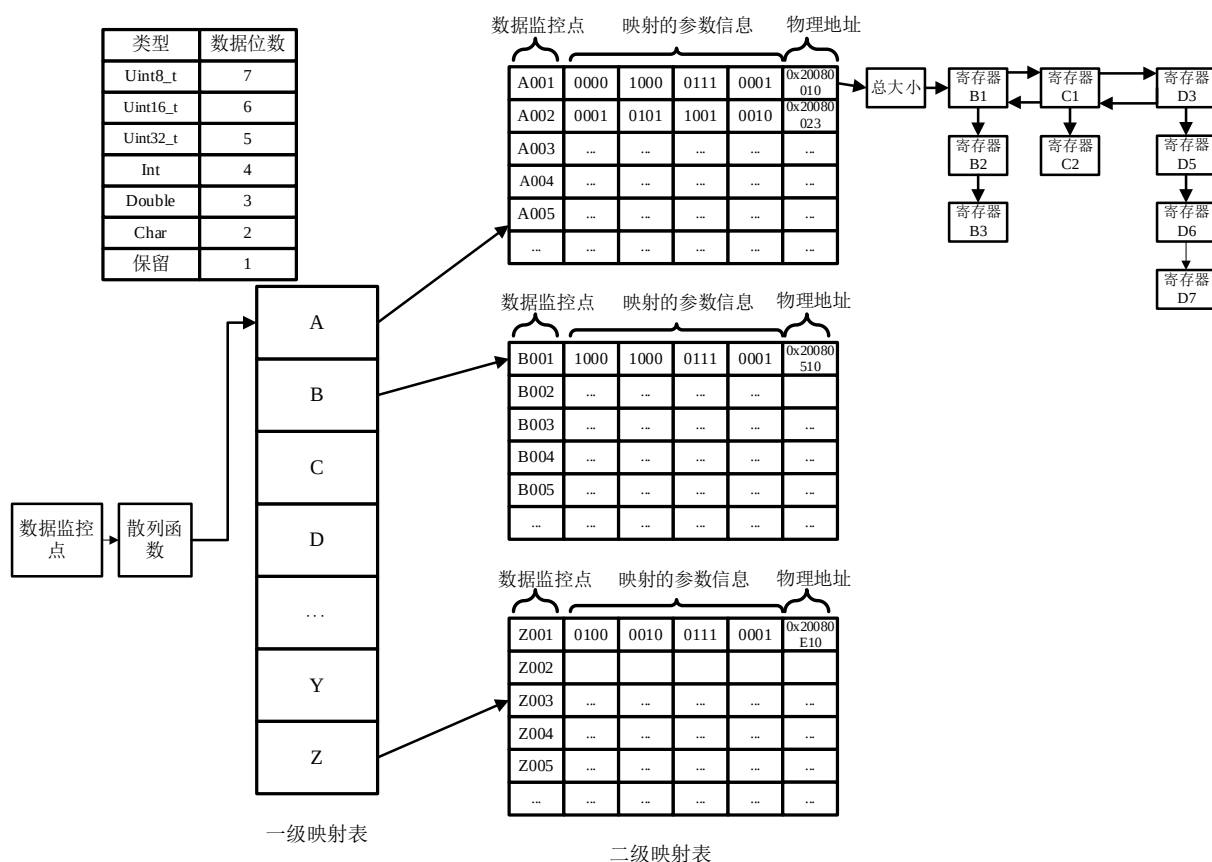


图 2.16 监控点和寄存器的地址自适应的方法示意图

当进行连续读取的数据帧时，采用数据位的首位来标识是哪一组的数据，因为同一个监控点存在多组数据。如此可以复用一部分的功能码和地址码。

该方法具体操作是：首先对数据监控点的类型通过哈希散列函数进行分类，从而建立一级映射表来提升查找的效率，其次有了一级映射表之后，可以再次对相同种类的数据监控点进行统一规划。这样就可以形成二级映射表，二级映射表中存有相应的数据监控点的映射参数信息和存放对应寄存器链表的物理地址。本方法通过建立二级映射表，将物联网监控点的对应寄存器信息整合统一，提高查找效率和代码的通用性。若未进行对应的数据监控点，也可以做到自适应添加。

2.3.2 与设备云通信协议

在与云设备通信中使用的是 MQTT 协议，MQTT 协议由 IBM 公司 1999 年推出，在 2013 年对其进行了标准化，开发之初为了保证石油运输管道与卫星之间能够保持通信顺畅^[46, 47]。之后随着物联网技术不断的发展，MQTT 协议在智能家居、智能工厂及智能城市等领域得到了广泛的应用。MQTT 协议主要被广泛使用在嵌入式设备和网络中间件通信过程中，适用于可用网络资源较少的场景，网络开销较少。采用的异步通信方式，通过 TCP 建立连接，可以和设备多对多进行通信，其可以反向控制，能进行长连接也是与设备云通信的优势。

MQTT 协议数据包格式共有三部分，第一部分是固定头部，固定头部用来标识数据包类型和数据包的分组。第二部分为可变头部，常用来表示是否此次的操作的功能意义，是不一定需要的。最后部分为消息体，消息体即有效负荷，是另一端收到数据的详细内容，也就是数据域内容。MQTT 协议数据包格式如图 2.17 所示：

7	6	5	4	3	2	1	0
MQTT报文类型				标识位			
可变头部（可选）							
消息体							

图 2.17 MQTT 数据包格式

在实际使用 MQTT 协议的时，为了加快发送速度和解析速度，发送和接收阶段均使用 JSON 格式。JSON 格式具有很好的层次结构，可以将数据更清晰展示，同时也可使用其他的语言对其进行解析，通用性和兼容性强。JSON 格式的数据因其占用传输带宽小，非常适合用于嵌入式网络设备中。其次该格式便于客户端进行读取和解析。和设备云之间数据通信流程如图 2.18 所示：

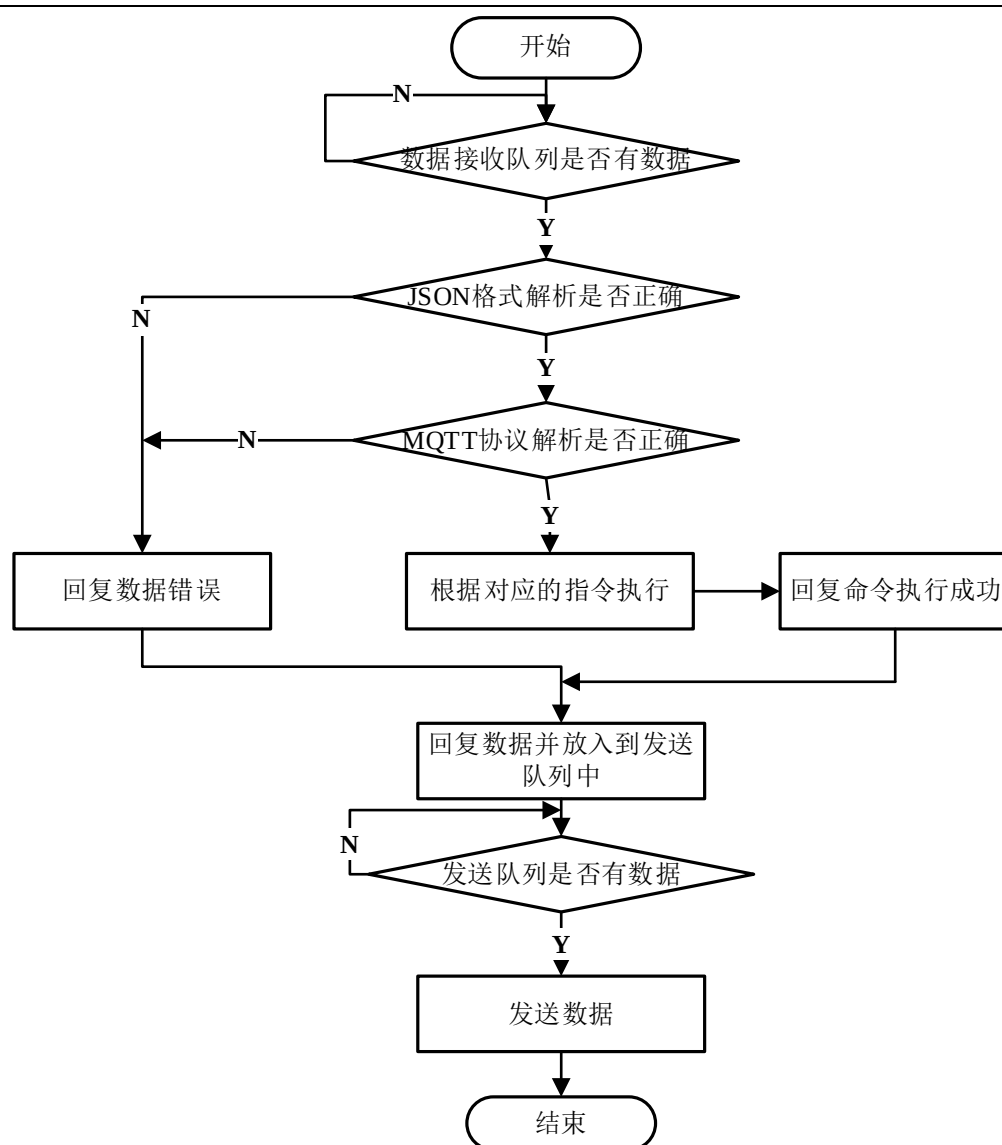


图 2.18 与设备云之间数据通信流程

2.4 应用层设计

2.4.1 云平台设计

2.4.1.1 设备云

设备云是集硬件设施和软件服务为一体的平台，是将采集的数据与业务云交互中的媒介。本系统通过选取分析并根据需求，最后选取中国移动 OneNET 物联网云平台，该平台在国内物联网第三方平台中技术较为领先。该平台功能强大，其可以为终端设备提供包括在线调试、远程升级维护在内等多种功能。当调试硬件端和业务云接口时，还可以提供模拟数据功能。设备可通过创建、注册，获得唯一的 IMEI 码，通过此标识与业务云进行对接，交互数据。设备的监控点信息也在设备云中进行展示，云平台架构图如图 2.19 所示：

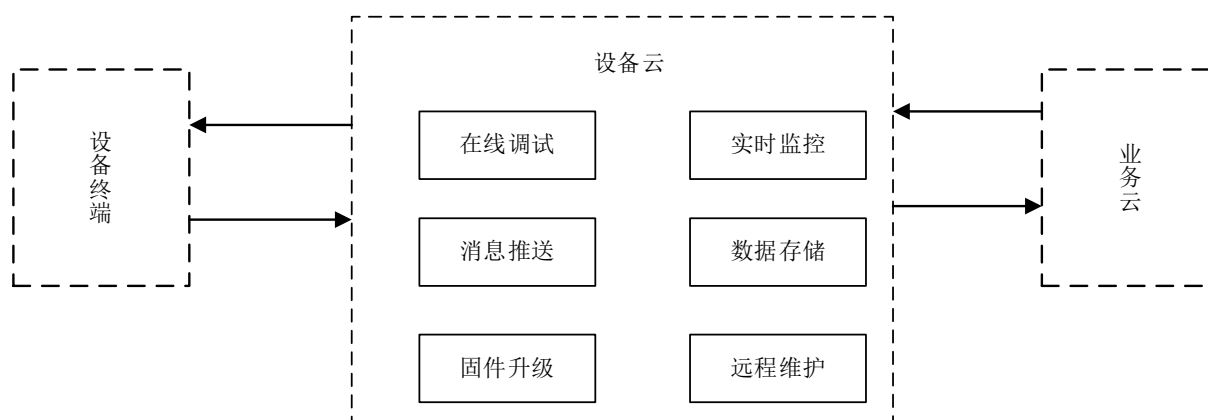


图 2.19 云平台架构

网关和云平台通过 MQTT 协议进行通信，同时也提供了 API 接口给用户使用，用户通过接口可以实现例如注册，定位等功能。通过产品 ID 来区分项目。通过 Master-APIkey 保证数据的安全性。开发者也可以根据需求选择通信格式和数据连接方式，触发器的设置等功能。

2.4.1.2 业务云

当外部有多个业务请求时，本系统采用 Nginx 负载均衡，将不同的请求分发到不同服务器上。Nginx 能保证接受高并发的同时系统不易瘫痪。业务分为数据赋能层和数据存储层，通过 Redis 来缓存缝纫机中不需经常修改的参数，这样既能缓解数据库压力，又能提供更好的服务质量。在 IaaS 基础服务上，选取第三方平台，在其上运行 Tomcat 服务，其中包括 Spring Boot，因其易部署等优点常被使用，给使用者更好的操作体验。业务云架构如图 2.20 所示。



图 2.20 业务云架构

2.4.2 Web 设计

Web 端实现多种功能且复杂，故采用 Vue 框架。Vue 是以数据为驱动的，可以更方便地对相应数据进行处理，同时是轻量级的框架，同时在组件化方面有更好的优势，有着更好的扩展性。满足不同功能的需求。Web 设计的框图如图 2.21 所示，整体框架以 Vue 为主体，通过第三方可视化组件将复杂的数据转化为易懂的图表。同时在网络方面与业务端通过 HTTP 异步请求，极大地提高了性能和速度。

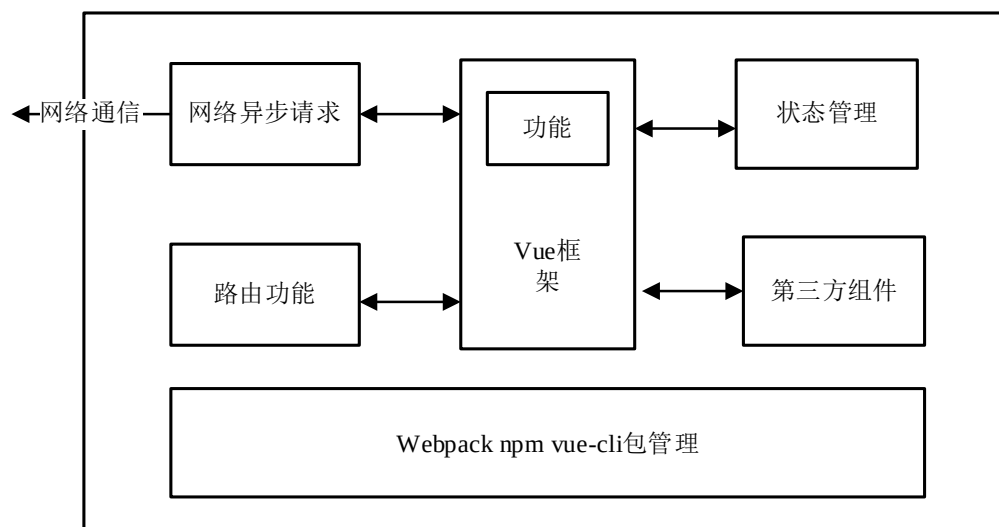


图 2.21 Web 端整体框架图

Web 端根据业务需求对整体页面进行功能设计如图 2.22 所示。在用户管理上对用户的登录，注册等进行实现。为了用户能更加直观的看到数据，对数据进行可视化大屏的展示，不仅能看到当前的数据，同时对历史数据也能逐天进行查看。还有实现业务逻辑功能比如网关的注册和删除包括监控点的添加，删除等操作。当网关出现故障可以在页面进行醒目提示。

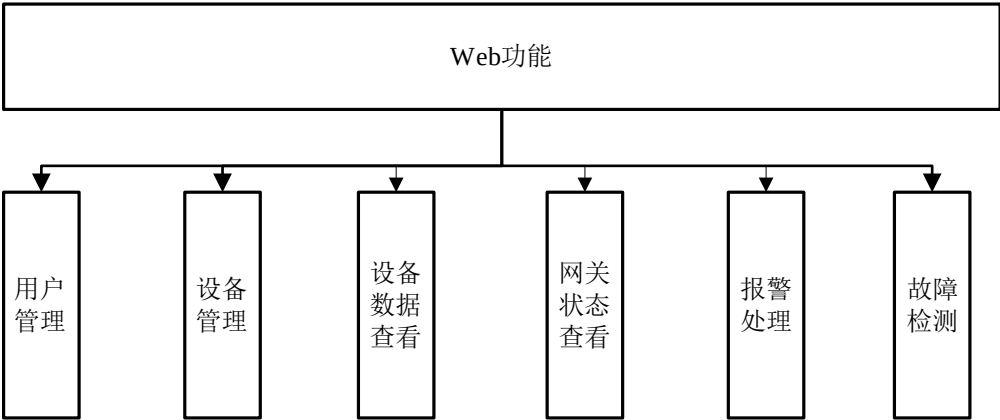


图 2.22 Web 功能设计图

2.4.3 小程序设计

小程序采用类 Web+原生组件形式，提高开发效率。通过将逻辑层和渲染层分开使用多线程来实现两部分之间的交互。通信模型如图 2.23 所示。

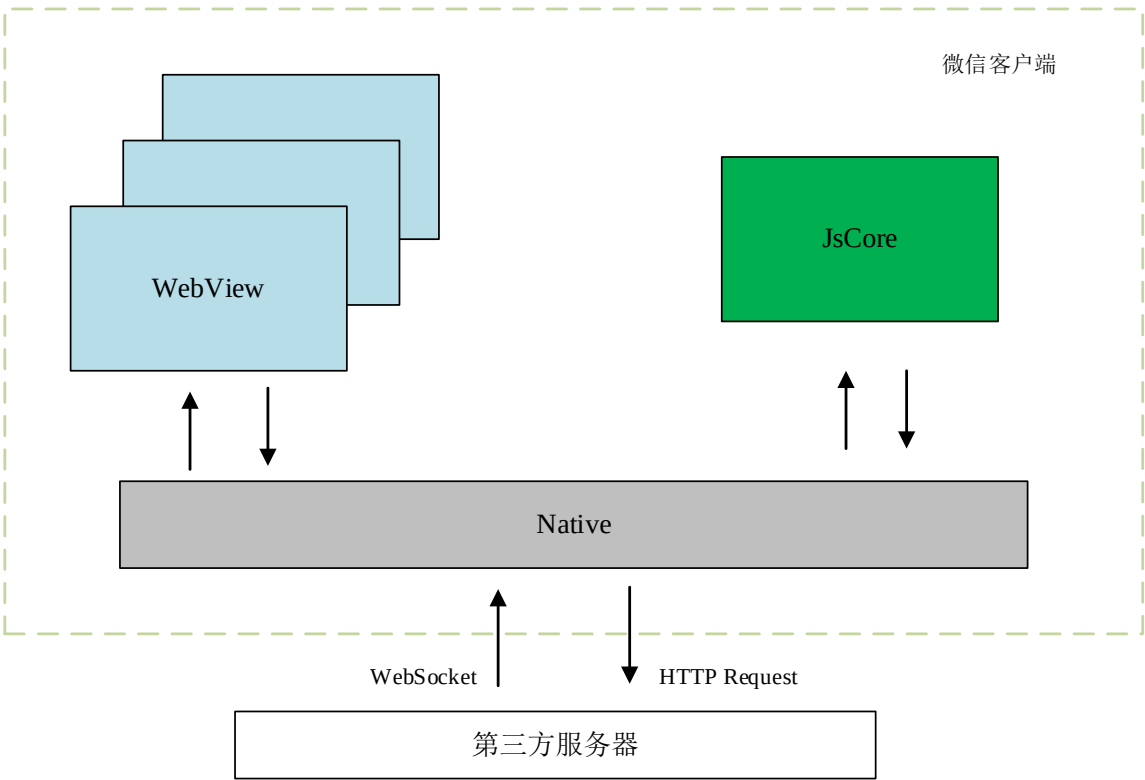


图 2.23 小程序通信模型图

2.5 整体效果展示

2.5.1 设备部分效果展示

图 2.24 为缝纫机整体展示图；图 2.25 为智能网关展示图；图 2.26 为缝纫机控制器部分展示图；图 2.27 为缝纫机控制器整体展示图。



图 2.24 缝纫机整体展示图



图 2.25 智能网关展示图

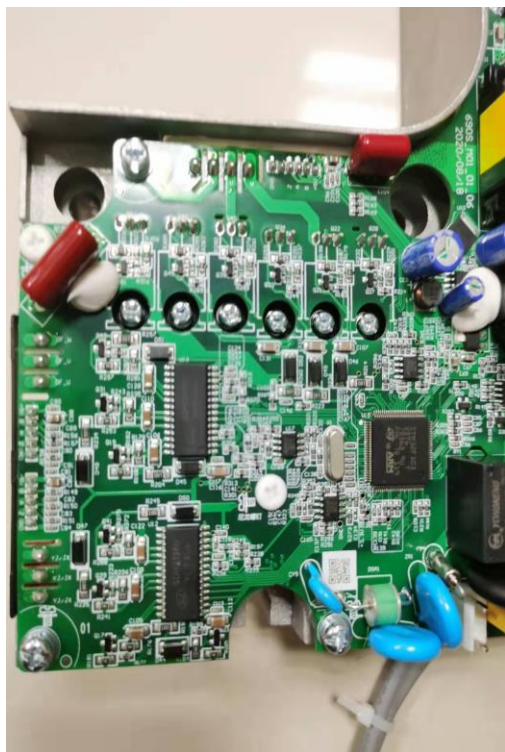


图 2.26 缝纫机控制器部分展示图



图 2.27 缝纫机控制器整体展示图

2.5.2 云平台部分效果展示

图 2.28 为示为缝纫机部分在线设备展示，该缝纫机目前已有 744 台在线使用。

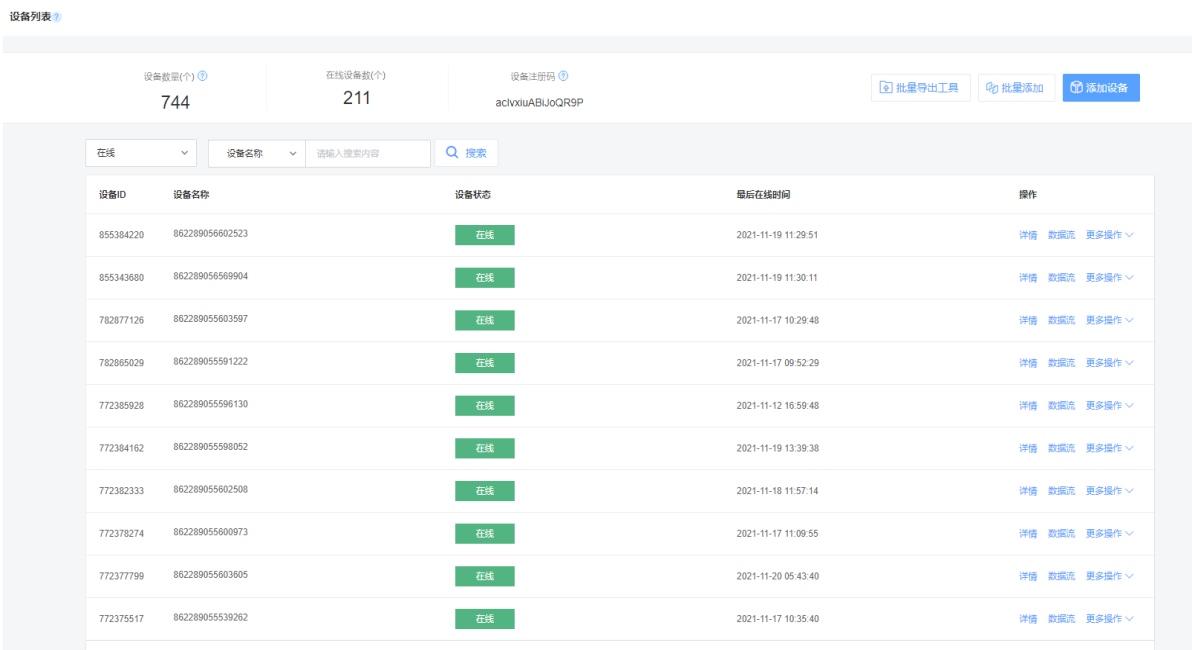


图 2.28 缝纫机在线设备图

如图 2.29 所示为缝纫机单台设备的监控点信息，包括网关版本号，异常状态、连接信息，Wi-Fi 名称等重要信息。

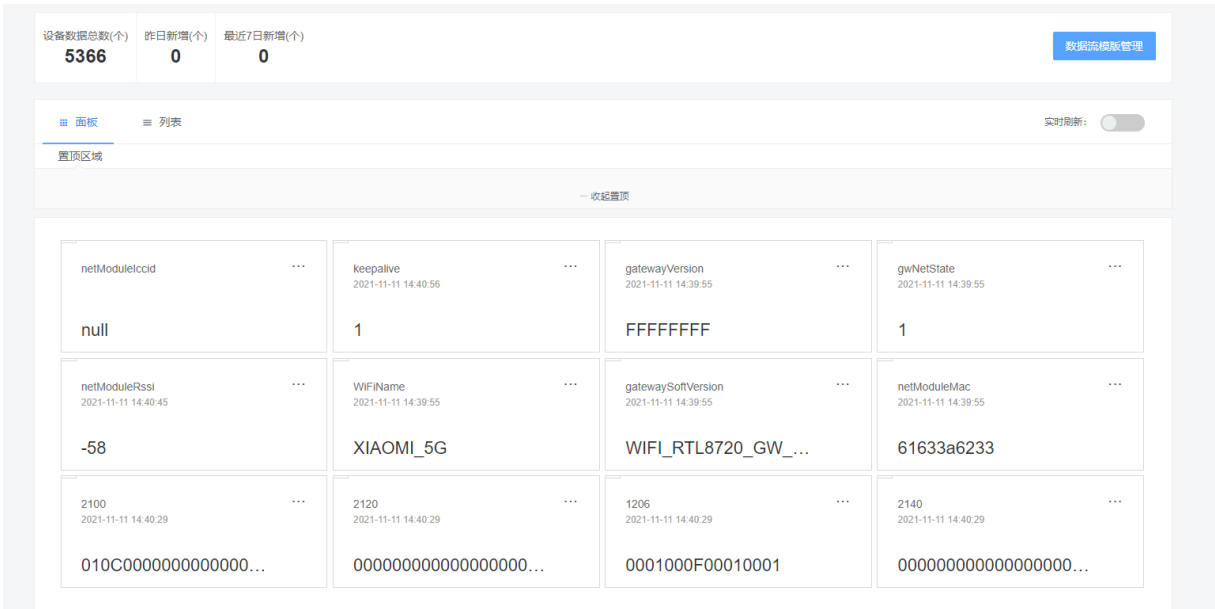


图 2.29 缝纫机监控点信息

2.5.3 Web 端部分效果展示

Web 端页面展示如图 2.30 所示，主要显示信息包括设备运行统计情况，产线的产能，缝纫机的类型及当前产线的生产进度等主要信息。



图 2.31 所示为对缝纫机参数进行远程设置实际效果图。

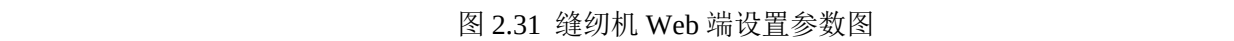
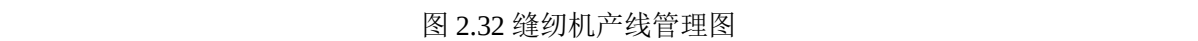


图 2.32 所示是对产线的管理信息汇总。包括当前进度，完成的数量和目标数量等信息。



27

图 2.33 所示是设备故障信息及相关故障代码：

公司选择: 电控厂家:

设备分组

 设备名称:

设备名称

 时间选择:

开始日期

 至

结束日期

查询

重置

序号	设备类型	设备编码	电控厂家	故障代码	故障内容	故障次数	年故障占比
1	包缝机	01070130A	瑞星	E17	缝台保护开关没到正确位置	20	50
2	包缝机	01070130A	瑞星	E16	抽压脚保护开关没到正确位置	18	100
3	包缝机	01070130A	瑞星	E17	缝台保护开关没到正确位置	12	30
4	包缝机	01060130A	瑞星	E17	缝台保护开关没到正确位置	7	17.5
5	平缝机	000000000	沪龙	P.oFF	断电显示	2	100
6	包缝机	01060130A	瑞星	E14	编码器信号异常	2	100
7	缝缝机	17AA00100	瑞星	E05	控速器接触异常	2	50
8	平缝机	17AA00100	瑞星	E87	倒缝步进电机堵转	1	100
9	平缝机	17AA00100	瑞星	E05	控速器信号异常	1	25
10	包缝机	01070130A	瑞星	E05	控速器接触异常	1	25

图 2.33 缝纫机设备故障信息图

2.5.4 小程序部分效果展示

小程序页面展示如下所示，图 2.34 为终端配网，图 2.35 为设备运行选取，图 2.36 为设备配置参数。



图 2.34 配网图



图 2.35 设备运行图



图 2.36 设备配置参数图

2.6 本章小结

本章对缝纫机在线监测系统设计进行了整体的概述，并对各部分进行详细描述，包括物联网网关的整体设计，之后根据硬件，软件模块分块介绍。并针对通信协议设计，云平台及Web 端设计进行介绍，最后分模块对系统整体效果进行展示。

第三章 基于 EEMD 和改进粒子群算法的 SVM 故障检测方案

工业缝纫机是生产服装的重要设备，随着机械化生产的发展，同时增大了机器振动和噪声，严重影响了缝纫机的生产质量。因此，需要在缝纫机生产工作中进行故障检测。

故障检测主要是通过对采集到的设备信息进行数据分析，能够在设备正常运行过程中对设备是否有故障进行检测。在上世纪 60 年代，以快速傅里叶变换为基本原理的频谱分析仪对轴承振动波形进行故障诊断。近些年来，随着技术发展，有着更智能方法进行故障检测。国红波根据缝纫机轴承振动信号特点选择 EMD 方法进行分解并通过 RBF 神经网络进行故障诊断且取得了较好的效果^[48]，不过，该方案仍存在可以优化的地方。首先 EMD 方法会造成信号混叠情况导致误差，使得检测精度不足，同时 RBF 神经网络建模过程较其他算法更复杂。张荣川等人使用 CEEMD 方法在缝纫机轴承振动信号分解精度方面进行改进，并采用最小二乘向量机完成分类并进行故障检测^[49]。

本文针对此类工厂机器故障识别正确性不高的问题，提出一种基于改进的粒子群支持向量机故障检测方案。首先对缝纫机轴承故障展开分析，并对传统技术方案进行分析，针对方案中的信号分解技术和分类算法进行合适选取分析，对支持向量机的参数选取方案进行改进。根据机械特点和检测精度选用 EEMD 方法进行分解，应用改进的粒子群优化 SVM 参数来提高轴承故障检测的准确率。

3.1 缝纫机轴承故障分析

缝纫机运转时在周期性变化的力矩和力的作用下在垂直和水平方向上都将产生振动^[50]。剧烈的振动会使缝纫机的寿命降低，零部件磨损严重。缝纫机振动包含低频分量和高频分量，是很宽的谐波分量。此类振动不能消除，只能减小机器的振动。同时机器产生噪声的来源也很多，例如：齿轮的冲击噪声，驱动电机的电磁噪声等。轴承的剖面图如图 3.1 所示。图中 1 为轴承的外圈，2 为轴承的滚动体，3 为轴承的内圈。在平时使用中，故障共可分为三类，分别为：

内圈故障通常是因为轴承游隙过小，导致内圈膨胀严重，引起了内圈故障。内圈故障均方根值会因为较大的能量冲击而大大增加，同时在频谱上 200Hz 时，振幅大于正常情况信号。

外圈故障通常是因为长时间受力导致孔径变大，致使打滑，不及时进行修复而长时间磨损，引起外圈故障。外圈故障在时域中无法较明显看出和正常情形的区别，在频谱上 120Hz 时，可以看出振幅有区别于正常情况下信号。

滚动体故障通常是因为内外圈经过长时间工作，挤压磨损变形，出现擦痕，引起滚动体转动不流畅。滚动体故障在时域上振动趋势和正常情况下类似，但显得更加离散，且在振幅

上更加明显。

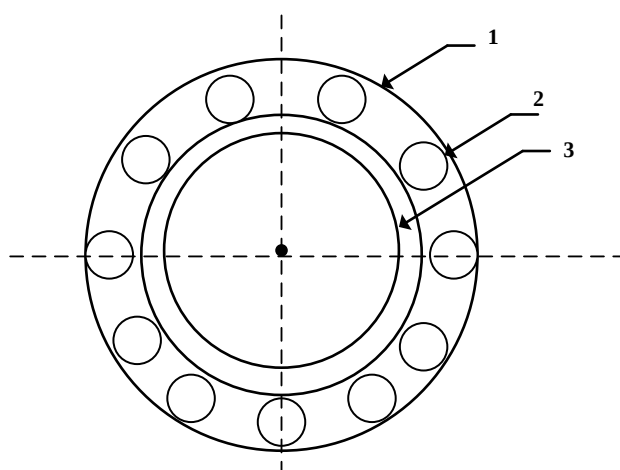


图 3.1 滚动轴承剖面图

缝纫机振动信号有以下几个特点：

1. 振动信号频谱范围相对于其他的机械设备较宽。
2. 缝纫机组件较复杂，直接获得振动信号难以实现。
3. 信号源较多，易受到干扰，信号均为非平稳信号，获取有效信号难度大。

总之，工业缝纫机由于机械旋转、往复、冲击运动所产生的信号混杂在一起，一般不直接用采集的信号用作分析对象，频率范围较宽，导致最终采集的信号是非平稳信号。而是通过现代信号分析方法对采集的信号进行处理，将无法表达故障的特征信息和反映故障的特征信息分离开，得到能表征该信号的状态的特征信息，之后寻找此特征信息和状态的关联。因此对其实施故障诊断比旋转机械困难，需要对采集到的信号分解再进行分析。

3.2 故障检测方案分析

3.2.1 方案介绍和分析

在文献[48]中，作者是采用两种对信号分解的方法来处理获得的信号并进行对比，一种是小波变换，另一种是经验模态分解。最后通过训练 RBF 神经网络对其进行检测识别。其中处理之后的振动信号选取了奇异值熵作为 RBF 的输入特征向量。

传统故障检测方案如图 3.2 所示。

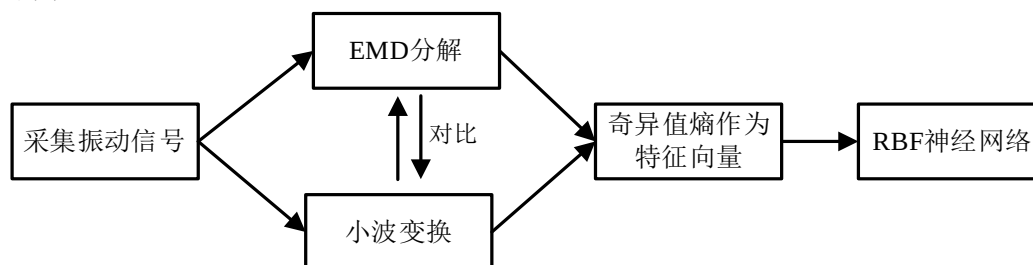


图 3.2 传统故障检测方案图

3.2.1.1 EMD 分解和小波变换

1. 经验模态分解：简称为 EMD。是一种将时域和频域结合起来分析信号的方法，信号可根据自身特点进行分解。可以针对任何类型信号，将采集到的信号波形分解成能够反映本信号特征的本征模态函数(Intrinsic Mode Function, IMF)的总和。这些分量是分布在不同频率上的，且波形平稳。在一定程度上，可以有效反应出信号原本的固有特性。通过计算由信号的所有极值点构成的包络线，使得分解产生的信号的平均值为零，将原始信号减去平均值曲线获得此次分解的分量，再以剩余分量作为下一次的原始信号。同时该方法可以消除部分周围噪声对信号的影响。
2. 小波变换：是上世纪后期发展起来的时频理论，通过时频域的局部变换，对处理之后信号作出更多尺度的精细分类，可以有效地解决传统傅里叶变化无法处理的问题。

EMD 分解虽然可以自适应分解信号，且能很好地保证信号的完备性，但是仍会存在相同的特征分量存在于不同分量中，不同的分量仍保留在单一的分量中。同时，在极值点需要拟合的包络线会随着振动信号的振荡而发生剧烈改变。形成了尺度的混杂，导致后续需要分解的 IMF 信号有着严重的误差。

小波变换虽然能很好地从信号中获取有效信息，但对于小波基的选择是比较复杂的，且只要选择了确定的基函数，在变换中不能随着信号不同而变换，不是非常灵活。分解算法对比如表 3.1 所示。

表 3.1 分解算法对比

分解算法	不足
EMD 分解	存在模态混叠问题
小波变换	小波基选择困难，固定基函数，恒定分辨率

3.2.1.2 RBF 神经网络

RBF 神经网络有三层结构，最左侧一般为输入层，负责的是获取信息；中间为隐藏层，一般有多层，是将数据分割的层，具有重要意义；最右侧为输出层。之所以称为 RBF，是因为中间神经元是基于径向基函数构建的，根据点到中心的距离，通过线性变换等方式映射到输出层。该论文通过有监督学习（网络学习法）学习三个参数。虽然 RBF 神经网络运行快，输出简单，但是由于对要学习的参数依赖极高，不能很好地对函数进行收敛。且有监督学习参数会使整个网络结构变得臃肿，计算量变大。

3.2.2 优化方向分析

目前传统方案对于缝纫机轴承振动信号分解主要是基于 EMD 算法，但是 EMD 算法会

出现多个频率差距较大信号在一个分量中，称之为模态混叠，且末端效应会产生影响分解的效果，从而在整体上影响最终的信号分解进度。同时小波变换存在基函数选择困难和不能灵活地适应不同的数据的问题。对于分类算法来说，RBF 神经网络的网络学习法会使网络结构变得庞大和复杂，加大计算量。综合目前这些方案存在的缺点，优化的方面可以分为以下几点：

1. 针对现有基于 EMD 算法缝纫机轴承振动信号分解产生的模态混叠和末端效应的问题，采用集成经验模态分解（EEMD）算法，对原始信号进行处理，保证了 IMF 分量的连续性。且选取特征可以进行很好聚类的特征向量。
2. 针对 RBF 神经网络的计算量大和复杂等问题。使用支持向量机（SVM）来进行分类，可以提高模型的泛化能力。
3. 支持向量机虽然模型建立简单且能解决非线性数据问题，但是有时候会陷入局部最优解的情况，需要对参数选择方式进行优化，提出自适应惯性权值粒子群算法优化 SVM 的参数选择。

本节根据上文所述缝纫机传统信号分解算法和分类算法的技术缺点和优化方向分析，设计了基于 EEMD 和改进粒子群算法的 SVM 故障检测方案。首先通过集成经验模态分解算法对缝纫机轴承信号进行分解处理，然后对处理之后的信号进行特征提取，最后通过改进粒子群算法的 SVM 对故障进行识别检测。具体方案如图 3.3 所示：

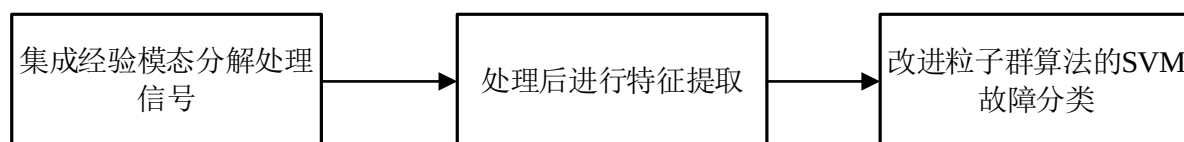


图 3.3 基于 EEMD 和改进粒子群算法 SVM 故障检测方案示意图

3.3 基于 EEMD 和改进粒子群算法的 SVM 故障检测设计与实现

3.3.1 总体流程设计

根据上小节对方案的介绍，故障检测流程同样也分成三个部分：通过 EEMD 对信号进行预处理，之后根据不同特征向量的聚类结果选择合适的特征，最后通过改进粒子群对 SVM 参数进行优化处理。具体步骤如下：

1. 通过传感器采集信号。
2. 通过 EEMD 算法对采集到的轴承信号进行分解，且对处理之后的信号采取阈值降噪操作。
3. 对经过预处理之后的信号，计算不同频率的特征向量，选取聚类效果较好的特征向量作为输入。
4. 通过改进的粒子群算法来获得 SVM 分类器较优参数。

5. 将采集到的信号分类，取前三分之二的的数据来训练。
 6. 用剩下的数据来测试训练之后的模型，得到四种状态结果。
- 具体流程如图 3.4 所示。

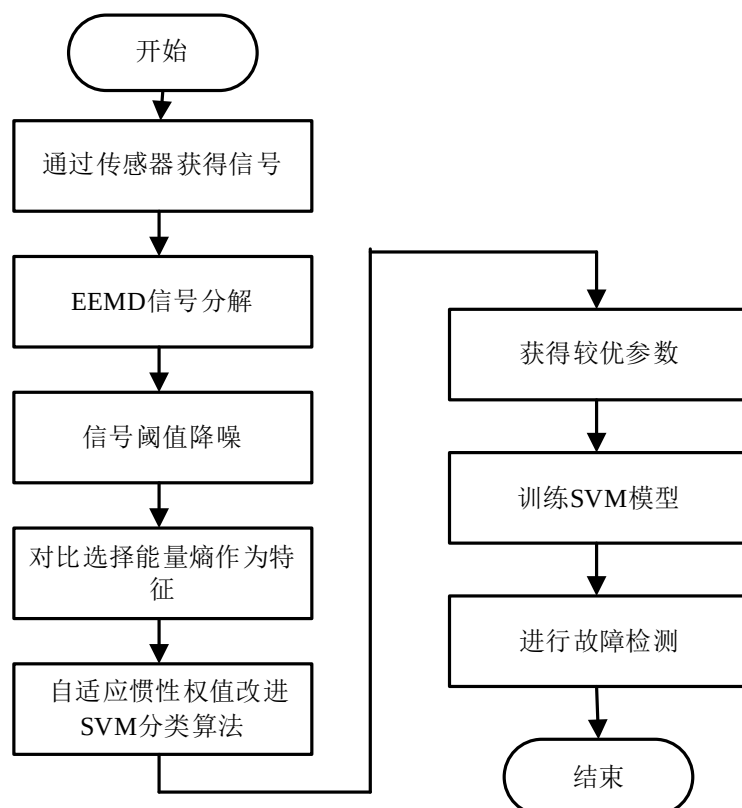


图 3.4 算法具体流程图

3.3.2 信号预处理

3.3.2.1 EEMD 分解算法

根据经验模态分解存在的混叠问题，E.Huang 针对噪声和混叠的缺陷，发现在需要分解的信号中会缺少高频的分量信号，需加入均值一定的白噪声，来填充一些被淹没的高频分量。此方法叫做集成经验模态分解算法。

该算法的原理是，因为原始信号极值点影响了 IMF 分解，导致的模态混叠，而白噪声频谱是均匀分布的，那么白噪声会使要求分解的信号分布到合适的尺度上，且白噪声具体均值为零的特性。随着白噪声加入次数增加之后，两者组成的复合信号逐渐抵消。这样获得的分解结果可以看作是最后的结果，随着白噪声的不断加入，复合信号和原来信号差距越小。集成经验模态分解算法步骤如下：

- (1) 假设原始信号为 $x_0(t)$ ，将一组均值为零，方差相等的白噪声序列 $n_j(t)$ ($j=1,2,\dots,n$) 加入到原始信号中，得到混合信号 $x_i(t)$ ($i=1,2,\dots,m$)，混合信号共有 m 个。混合信号 $x_i(t)$

可表示为：

$$x_i(t) = x_0(t) + n_j(t) \quad (3.1)$$

- (2) 对复合信号的每组都进行 IMF 分解，可以得到每阶 IMF 分量和每组最后剩余的分类 $r_k(t)$ 。 $r_k(t)$ 是分解后的残余分量。每组分解结果为：

$$x_i(t) = \sum_{i=1}^k c_i(t) + r_k(t) \quad (3.2)$$

上式中 $c_i(t)$ 为第 i 个 IMF 分量。

- (3) 分析分解之后的信号，若次数没达到标准，那么继续步骤 1 和 2，为得到 IMF 分量矩阵 C ，残余分量矩阵 Res 。可通过增加功率相同但序列不同的白噪声。

$$C = \begin{pmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & \ddots & \vdots \\ c_{m1} & \cdots & c_{mn} \end{pmatrix} \quad Res = \begin{pmatrix} Res_{1n} \\ Res_{2n} \\ \cdots \\ Res_{mn} \end{pmatrix} \quad (3.3)$$

- (4) 因为白噪声均值为零，可以得到 IMF 各分量为：

$$\begin{cases} C_1 = \frac{1}{m} \sum_{i=1}^m c_{i1} \\ C_2 = \frac{1}{m} \sum_{i=1}^m c_{i2} \\ \cdots \\ C_n = \frac{1}{m} \sum_{i=1}^m c_{in} \end{cases} \quad Res = \frac{1}{m} \sum_{i=1}^m Res_{in} \quad (3.4)$$

- (5) 根据以上分析，原始信号可表示为：

$$x_0(t) = \sum_{i=1}^n C_i + Res \quad (3.5)$$

3.3.2.2 EEMD 阈值降噪处理

在处理原始信号的时候，EEMD 分解加入了均值为零的白噪声来减小模态混叠的问题，但是仍然会存在噪声影响特征提取。所以将分解之后的 IMF 信号通过阈值降噪的算法对其进行降噪处理，提高分类的准确性。

该算法的原理是：通过计算各 IMF 分量的相关系数来判断。相关系数的计算为 IMF 分类能量平均值和其对应的平均周期乘积。若相关系数和判断条件相近，那么该分量判断为是噪声，若相差较大，则该分量中含有有效信息，可以用来进行特征提取。

IMF 能量的平均值和其对应的平均周期乘积公式如式 (3.6)。

$$C = \overline{E_n T_n} \quad (3.6)$$

式 (3.7) 所示为判断条件, 其中 a 一般选取 0.25 至 0.3 范围:

$$\frac{|\ln \overline{E_n} + \ln \overline{T_n}|}{\ln \overline{T_n}} \geq a \quad (3.7)$$

EEMD 阈值降噪步骤如下, 流程图如图 3.5 所示:

1. 通过 EEMD 分解对原始信号 $x_0(t)$ 预处理, 以得到各阶 IMF 分量。
2. 计算各阶 IMF 能量平均值和其对应平均周期乘积。
3. 根据式 (3.7) 进行判断, 若乘积和式 (3.7) 相近, 那么认为是噪声。反之, 则含有有效信息。
4. 对能表达有效信息的 IMF 分量信号进行重组。

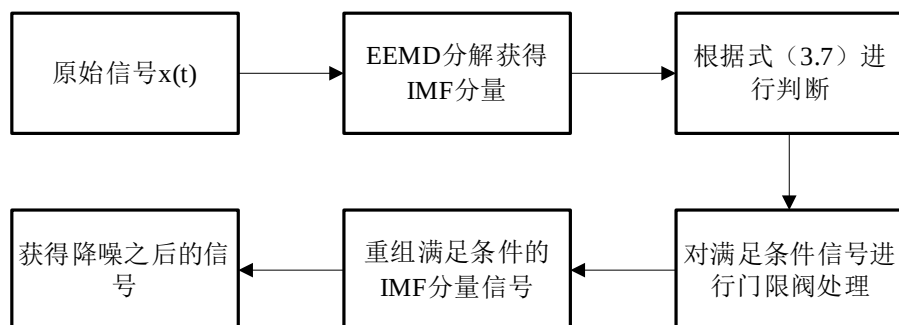


图 3.5 EEMD 阈值降噪流程图

本文采用的 EEMD 算法是在原始信号中加入幅度值为 0.2, 总平均次数为 200 的多组白噪声。如图所示是正常状态下原信号波形和加入噪声波形, 图 3.6 所示是正常状态原始信号和加噪声信号, 图 3.7 为内圈故障原始信号和加噪声信号:

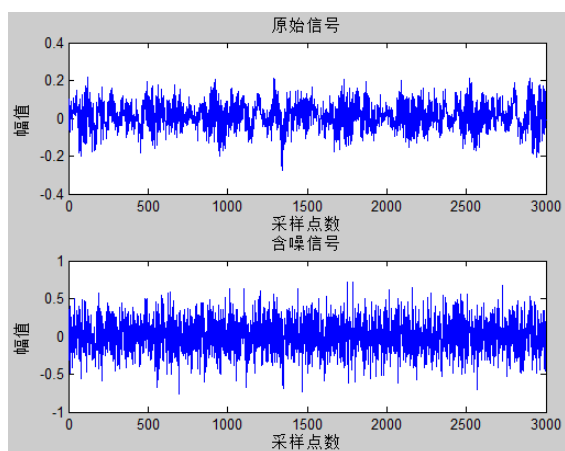


图 3.6 正常状态原始信号和加噪声信号

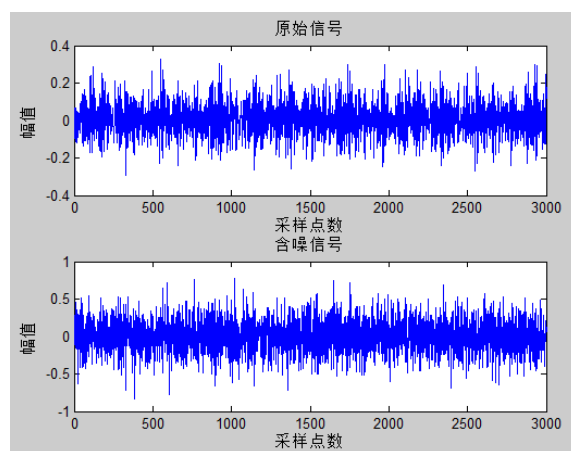


图 3.7 内圈故障原始信号和加噪声信号

图 3.8 为正常状态信号 EEMD 分解的各 IMF 分量，图 3.9 为内圈故障状态信号 EEMD 分解的各 IMF 分量。

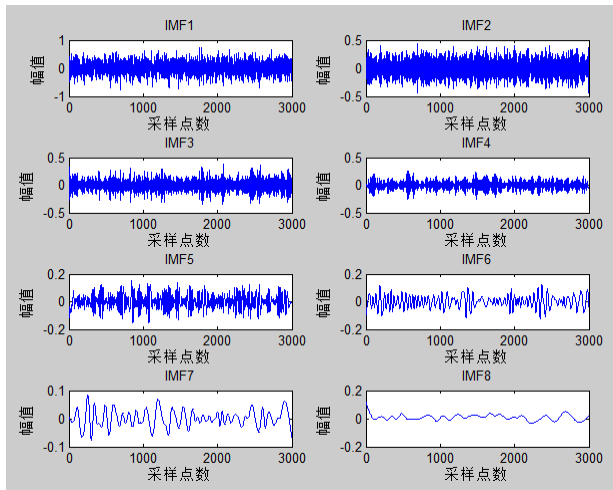


图 3.8 正常状态信号 EEMD 分解

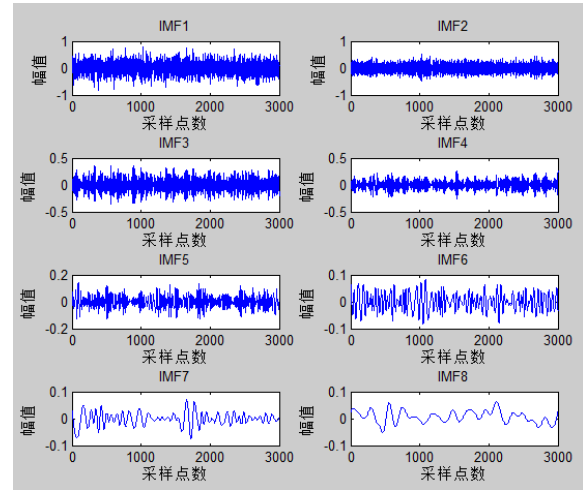


图 3.9 内圈故障信号 EEMD 分解

在实际 EEMD 分解中，因为每个 IMF 分量需多次迭代分解，不同频率的分量信号不能按顺序进行排列，会产生频率重叠的现象。对于某些分量又会出现过分解的情况，这样会导致很多低频率信号的出现。因此，在某个信号多个 IMF 分量中，不是所有的分量都能反映出该信号的特征，将这些信号称为伪 IMF 分量。对于处理伪 IMF 分量可以很大程度提高分解的效率。通过计算和原信号的相关性来判断是否是伪 IMF 分量，一般当低于 0.01 就被认为是无用信息。计算相关性公式如下：

$$Q_{xy} = \frac{\sum_{k=0}^n x(k)y(k)}{\sqrt{\sum_{k=0}^n x^2(k) \sum_{k=0}^n y^2(k)}} \quad (3.8)$$

通过公式计算相关性之后可得，IMF 分解的阶数越多，与原信号的相关性就越弱，越不能体现信号特征。根据计算可得轴承的各种状态和原始信号相关性情况如表 3.2 所示：

表 3.2 IMF 各分量相关系数

故障分类	相关系数							
	IMF1	IMF2	IMF3	IMF4	IMF5	IMF6	IMF7	IMF8
正常	0.4981	0.6941	0.4398	0.4377	0.5293	0.3244	0.1243	0.0275
内圈故障	0.672	0.7842	0.5862	0.3366	0.2312	0.0541	0.0292	0.0052
外圈故障	0.8287	0.7754	0.2344	0.0432	0.0376	0.0275	0.0087	0.0123
滚动体故障	0.6433	0.949	0.415	0.2379	0.1746	0.0955	0.0523	0.0112

根据上表可以得出，外圈故障和滚动体故障在 IMF7 分量和原信号相关系数小于 0.01，说明不能很好表现出原信号特征，故选择前 6 阶的 IMF 分量进行特征提取。

EEMD 算法示意图如图 3.10 所示：

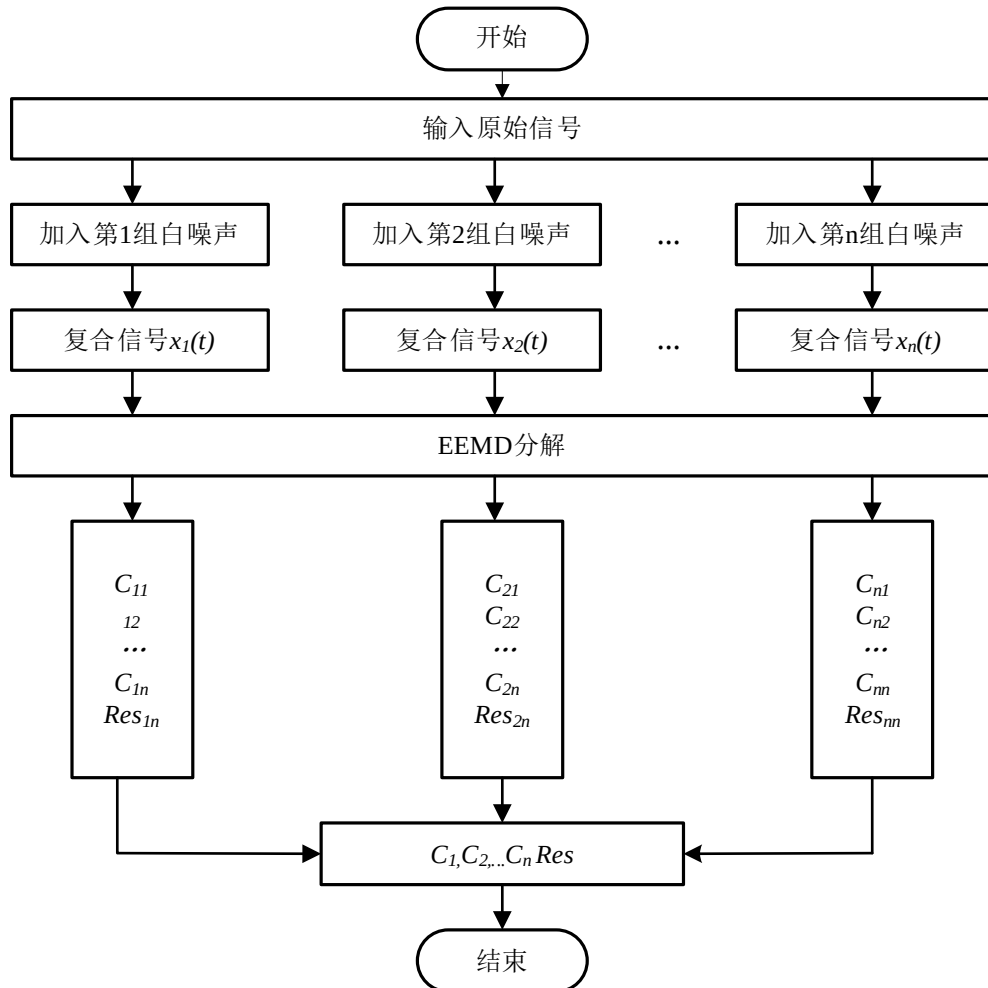


图 3.10 EEMD 算法流程图

3.3.3 特征提取

当对缝纫机轴承原始信号进行预处理之后，得到多个不同频率的分量。但是阶数不同的分量代表的信息也是不同的，若把这些特征信息均放入分类器中进行识别诊断，种类繁多，且不能很好根据特征信息进行识别，降低了识别正确率。

在如何将 IMF 分量转化为分类器可以识别的特征向量上，常用的向量有：能量熵，峭度，功率谱熵，模糊熵等特征向量。根据实验测试，在阶数较小的情况下，四者均可以很好进行故障分类，但随着阶数增大，后两者对于不能很好将故障分离开。因此，本小节对能量熵和峭度两种特征进行选取。

峭度是一种常被用来当作特征的参量，该数值和峰值变化的趋势是具有相似性的，尤其在早期未完全发生故障时，有很好的敏感性，尤其是冲击信号。峭度值计算如下所示：

$$\beta = \frac{1}{N} \sum_{i=1}^N x_i^4 \quad (3.9)$$

根据式(3.9)，可以分别计算出轴承在四种状态下的各阶 IMF 分量的峭度值。

根据上小节，可以得知通过对 IMF 前 6 阶的分量分析，从中能获取到有用的特征信息。提取特征的依据是需要在各分量的都能有较好的区分效果。

为了更好地看出聚类效果，将前 6 阶分量计算的结果使用三维散点图进行表示，图 3.11 的 x 轴表示的是第一阶 IMF；y 轴表示的是第二阶 IMF；z 轴表示的是第三阶 IMF，图 3.12 的 x 轴表示的是第四阶 IMF、y 轴表示的是第五阶 IMF、z 轴表示的是第六阶 IMF，从前 3 阶分量分类中可以看出，四种状态都能很好地被区分开，但后 3 阶 IMF 分量在内圈故障和滚动体故障分类上有大量重叠，不能很好地将四种状态分离开。因此，通过峭度值分类，在高阶分量部分，有较好聚类效果，但在低阶分类，没有很好聚类效果。

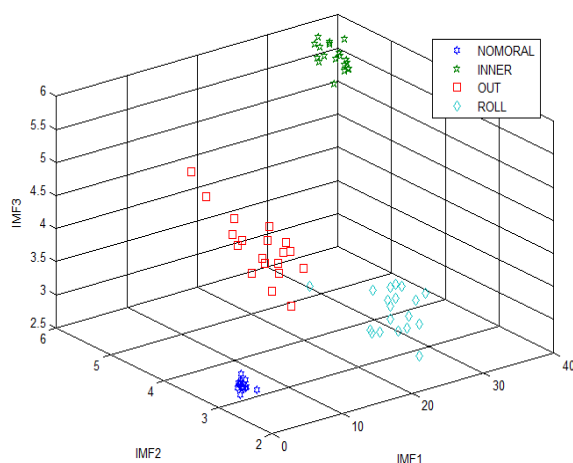


图 3.11 前 3 阶分量峭度值散点图

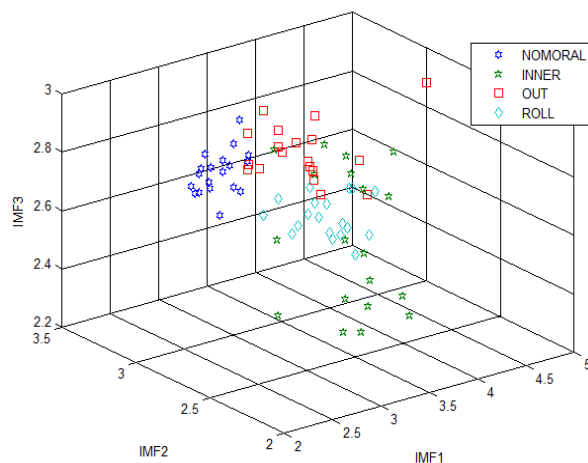


图 3.12 后 3 阶分量峭度值散点图

根据上式计算某样本峭度特征提取结果如表 3.3 所示：

表 3.3 峭度特征样本提取结果

选取特征	故障类型	IMF1	IMF2	IMF3	IMF4	IMF5	IMF6
峭度	正常	2.272	2.873	2.8753	2.8421	2.79340	2.8548
	内圈故障	36.2863	5.3134	5.5369	3.6298	2.35520	2.7743
	外圈故障	19.9213	4.9765	3.1208	4.8323	2.8878	2.9234
	滚动体故障	21.4216	2.789	3.072	3.8327	2.7314	2.6913

在轴承振动信号上每个 IMF 分量的能量都代表一定的信息，不同频率的能量是不同的，某个或多个信号能量能代表一种故障信息。因此，不同频率信号的能量可以作为分类器的输入特征向量。根据 EEMD 分解的信号计算 IMF 分量的能量：

$$E_i = \int |c_i(t)|^2 dt \quad (3.10)$$

根据能量熵的公式，可得 IMF 分量的能量熵为：

$$H_e n = \sum_{i=1}^n p_i \lg p_i \quad (3.11)$$

式 (3.11) 中， p_i 为第 i 个 IMF 分量的能量在总能量中的占比。

同时可以通过本小节原理计算可得通过能量熵来提取特征样本结果如表 3.4 所示：

表 3.4 能量熵特征样本提取结果

选取特征	故障类型	IMF1	IMF2	IMF3	IMF4	IMF5	IMF6
能量熵	正常	0.298	0.372	0.1652	0.2543	0.3145	0.1364
	内圈故障	0.3681	0.3761	0.3075	0.1298	0.1232	0.0143
	外圈故障	0.3023	0.3634	0.0873	0.0084	0.0072	0.0179
	滚动体故障	0.1502	0.1436	0.1422	0.1151	0.0717	0.0014

根据式 (3.11)，可以计算出轴承在四种状态下的各阶 IMF 分量能量熵的数值。

将前 6 阶分量能量熵计算结果使用三维散点图表示，图 3.13 的 x 轴表示的是第一阶 IMF；y 轴表示的是第二阶 IMF；z 轴表示的是第三阶 IMF，图 3.14 的 x 轴表示的是第四阶 IMF；y 轴表示的是第五阶 IMF；z 轴表示的是第六阶 IMF，从两图中可以看出，以能量熵为特征值，对于分解的 IMF 分量，均能很好地将四种状态区分开，说明能量熵具有很好聚类效果，能够完成对状态的分类。

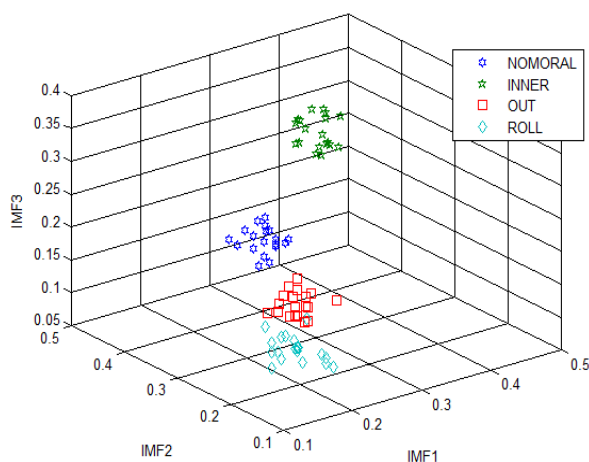


图 3.13 前 3 阶分量能量熵散点图

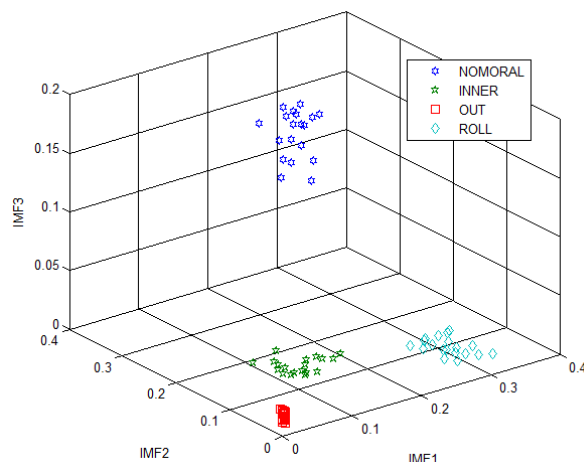


图 3.14 后 3 阶分量能量熵散点图

从上两个表中，可以分析得出能量熵和峭度在前 3 阶分量时有很好的聚类性，峭度效果

更好。但是随着阶数增加，峭度在后 3 阶效果不佳，很难有分类的效果。因为特征选取时聚类效果越好，后面分类效果越好。综合以上因素，选取能量熵作为特征向量是正确的。

若直接将 EEMD 分解之后的信号进行能量熵特征提取，虽然也有比较好的效果，但是仍会有一些随机噪声对 IMF 分量干扰，影响聚类的结果。现根据之前所介绍的阈值降噪法对信号进行降噪，去除一些噪声的干扰，再进行能量熵的特征提取，结果如表 3.5 所示：

表 3.5 阈值降噪处理之后的能量熵特征提取结果

选取特征	故障类型	IMF1	IMF2	IMF3	IMF4	IMF5	IMF6
能量熵	正常	0.2621	0.3627	0.1752	0.2531	0.3289	0.1314
	内圈故障	0.3675	0.3661	0.3075	0.1293	0.1235	0.01233
	外圈故障	0.1323	0.1347	0.1345	0.1132	0.0701	0.0179
	滚动体故障	0.1382	0.1331	0.1442	0.1153	0.0692	0.0012

对比进行降噪前后的能量熵处理的结果，可以发现当进行降噪处理之后，前 6 阶各分量都有变化，特征值都相对的提高了，可以说明进行阈值降噪操作之后分类效果更好了。整个特征提取的流程图如图 3.15 所示：

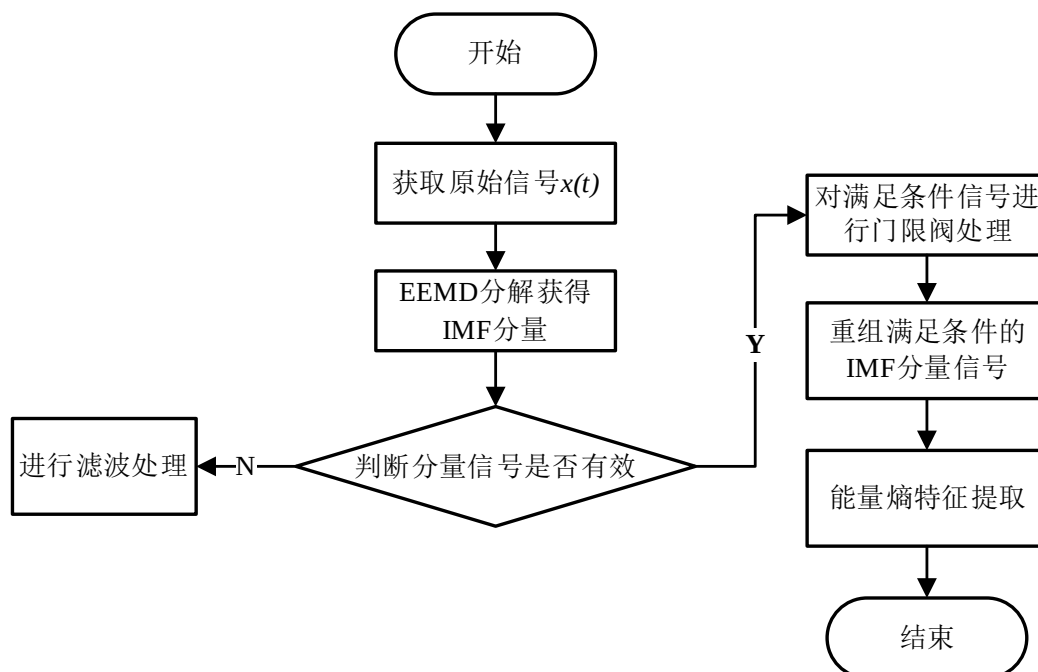


图 3.15 阈值降噪特征提取流程图

3.3.4 基于改进粒子群的 SVM 算法

3.3.4.1 SVM 分类器

SVM 是支持向量机的英文缩写。SVM 最开始是为了二值分类所设计的，后来随着研究人员越来越多，被推广到超平面进行研究学习。根据空间中的变量最大间隔空间原理找到最好的超平面，将空间中的数据进行分类。

SVM 向量机算法思想：在 n 维空间中有 P_0, P_1 两个点集，若空间中存在 n 维向量 w_i 和实数 b ，使得所有在点集 P_0 中的点均满足 $w_i x_i + b < 0$ ，同样地，对于点集 P_1 中的所有点均满足 $w_i x_i + b > 0$ ，那么可以认为在这个 n 维空间中， P_0 和 P_1 为线性可分。且已找到了可以分类的超平面。其原理如图 3.16 所示：

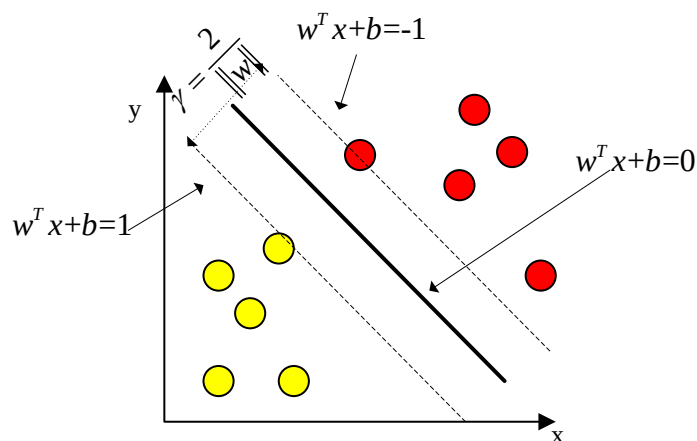


图 3.16 支持向量机原理图

线性分类：

在 n 维欧氏空间中，任意的超平面用公式描述为：

$$w^T x + b = 0 \quad (3.12)$$

式 (3.12) 中， $w = \{w_1, w_2, w_3, \dots, w_n\}$ 是最优超平面的法向量， b 是偏移值。

根据之前的最优超平面的定义，SVM 问题转化为最大化距离 d ，根据之前支持向量定义，

最终需要最大化 $\frac{2}{\|w\|}$ 。

最大化问题可以转化为最小化问题，那么 SVM 优化问题可转化为以下模型：

$$\min \frac{1}{2} \|w\|^2 \quad \text{s.t.} \quad y_i (w^T x_i + b) \geq 1 \quad (3.13)$$

为了解决这个优化问题，引入拉格朗日乘数法对其进行优化，对式 (3.13) 采用乘数法，

$$L(w, b, \beta) = f(w) + \sum_{i=1}^n \beta_i h_i(w) = \frac{1}{2} \|w\|^2 + \sum_{i=1}^n \beta_i (1 - y_i (w^T x_i + b)) \quad (3.14)$$

式中 β_i 为拉格朗日算子，对上述两边同时求偏导，可得到线性支持向量机的通用模型为：

$$\max_{\beta} \sum_{i=1}^n \beta_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \beta_i \beta_j x_i^T x_j y_i y_j \quad \text{s.t.} \quad \sum_{i=1}^n \beta_i x_i y_i, \beta_i \geq 0 \quad (3.15)$$

以上讨论的是线性分类情况，但是在实际情况中，非线性情况还是大多数的，不可能对所有数据都能完全地进行分类，因此在一定范围能使用不是完全分类。

那么为了处理这些不能很好分类的数据，引入松弛变量 ξ_i ，松弛变量即为不符合条件的样本点到本分类堆中的直线距离。将此模型进行修改如下式。那么软间隔 SVM 的优化模型变为：

$$\min \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n \xi_i \quad \text{s.t.} \quad y_i (w^T x_i + b) \geq 1 - \xi_i \quad (3.16)$$

在上式中，引入新变量惩罚因子 C ，该参数是为了适应松弛变量的，为了保证模型分类的正确性。若 C 的值很大，那么为了分类更准确，松弛变量就得变小，说明对不符合样本点要求更高，反之，若 C 值很小，对不符合样本点要求就松。

在加入了松弛变量之后，根据拉格朗日乘数法，将式(3.14)改为如下：

$$L(w, b, \beta, \xi_i, \mu) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n \xi_i + \sum_{i=1}^n \beta_i (1 - \xi_i - y_i (w^T x_i + b)) - \sum_{i=1}^n \mu_i \xi_i \quad (3.17)$$

其中 β_i 和 μ_i 均为拉格朗日乘子，其他参数均为变量。现对 w , b , ξ_i 进行求偏导。

$$\max_{\beta} \sum_{i=1}^n \beta_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \beta_i \beta_j x_i^T x_j y_i y_j \quad \text{s.t.} \quad \sum_{i=1}^n \beta_i x_i y_i, \beta_i \geq 0, \mu_i \geq 0, C = \beta_i + \mu_i, i = 1, 2, \dots, n \quad (3.18)$$

线性不可分：

对于不能简单地用直线或平面进行分类的问题，若要解决该问题，需引入一个函数，映射各个变量，将其转为线性情况进行分析。因此针对这些空间中不可再分的点，需要通过该函数将其映射到大于该空间的维度中，抑或是无限空间中，再通过软间隔等方式处理其线性可分的情况。我们对空间中需要分类的点 x ， $\Phi(x)$ 表示映射到更高维度的新向量。那么这个超平面可以表示为：

$$f(x) = w^T \Phi(x) + b \quad (3.19)$$

根据之前支持向量机问题表达，可以将其转换为：

$$\min \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n \xi_i \quad \text{s.t.} \quad y_i (w^T \Phi(x) + b) \geq 1 - \xi_i \quad (3.20)$$

根据拉格朗日乘数法和对参数进行求导，可以得到类似于式(3.15)的式子。

$$\max_{\beta} \sum_{i=1}^n \beta_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \beta_i \beta_j \Phi(x_i)^T \Phi(x_j) y_i y_j \quad \text{s.t.} \quad \sum_{i=1}^n \beta_i y_i = 0, 0 \leq \beta_i \leq C, i=1, 2, \dots, n \quad (3.21)$$

因需要进行高维度的点乘运算，计算量太大。这时引入核函数，在该函数中，高维度的点的数量积不易计算，但等于原始空间中的数量积大小。这样就不需要大量的数量积运算了。核函数表示为 $K(x, y) = (\Phi(x), \Phi(y))$ 。通常选择为高斯核函数。

非线性支持向量机以核函数形式表达为：

$$\max_{\beta} \sum_{i=1}^n \beta_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \beta_i \beta_j y_i y_j K(x_i, x_j) \quad \text{s.t.} \quad \sum_{i=1}^n \beta_i y_i = 0, 0 \leq \beta_i \leq C, i=1, 2, \dots, n \quad (3.22)$$

那么支持向量机的决策函数为：

$$f(x) = \sum_{i=1}^n \sum_{j=1}^n a_i^* y_j \exp\left(-\frac{\|x - x_i\|}{2\sigma^2}\right) + b^* \quad (3.23)$$

在式（3.23）中， σ 为高斯核的宽度。

在 SVM 进行分类时，分类的准确度与核宽度 σ 和惩罚因子 C 有关。在实际分类过程中，设置好这两个值有利于提高分类的准确率。

3.3.4.2 粒子群算法

1995 年，美国学者 Kennedy 等人提出粒子群优化算法（PSO），这是一种以迭代为基础的优化算法^[51]。该算法模拟鸟类自然界行为，该行为就是觅食行为，在觅食过程中，鸟类中个体相互合作共享食物信息进行觅食。鸟类在空间中搜寻食物，在算法中，该空间就是算法的搜索空间。种群对应于鸟群，粒子对应于鸟群中各只鸟。

该算法思想是：首先对种群中的粒子进行初始化，进行不断迭代，在每次迭代中，粒子在搜寻目标的同时，更新自己的位置信息，在自己搜寻过程中也关注群体的最优位置，两者共同作用，直到找到最优解。

假设问题搜索空间是多维的，粒子群的大小是 m 。粒子群中的每个粒子都有位置和速度这两个向量。设粒子的所在的位置为 X_i ，速度为 V_i 。每个粒子经过最优的位置为 P_i ，整个群体经过最优的位置为 G_i 。在每次迭代中，每个个体均通过以下两个式子不断改变自身的最优位置和速度。

$$V_i(t+1) = wV_i(t) + c_1 r_1 (P_i - X_i(t)) + c_2 r_2 (G_i - X_i(t)) \quad (3.24)$$

$$X_i(t+1) = X_i(t) + V_i(t+1) \quad (3.25)$$

在上两式中，其中 w 为惯性权值， c_1 为自我学习常数， c_2 为社会认知常数。 r_1, r_2 是区间[0,1]内的随机数。

从上式中，可以看出影响粒子群搜索的参数有三个：

- (1) 惯性权值：反映当前粒子速度对粒子的影响，粒子按照 w 的数值沿着自身速度方向进行搜索，可以动态地根据适应度的值调整粒子搜索的方向。
- (2) 学习因子：反映粒子经过的最优位置对粒子的影响，粒子会按照 P_i 来调整自己搜索方向，说明粒子对位置有记忆性，使粒子有局部最优搜索能力。
- (3) 社会认知常数：反映整个群体最优位置对粒子的影响，体现搜索过程是通过群体互相共享最优位置，加快粒子的收敛速度。

3.3.4.3 SVM 参数意义

影响 SVM 分类的精度有两个参数，分别是惩罚因子 C 和核参数 g 。其中（ $g = \frac{1}{2\sigma^2}$ ）核宽度 σ 和核参数 g 成反比，对两个参数进行优化对分类精度有重大意义。

惩罚因子 C 是对分类的宽限容忍程度，通过影响松弛变量 ξ 来控制风险程度，继而影响整个分类模型的泛化程度。当 C 的值小的时候，说明对错误的容忍程度较大，惩罚力度小，但是会出现欠拟合的情况。随着 C 值不断变大，对错误的容忍程度变小，惩罚力度大，那分类正确性就提高，但是会出现过拟合情况，泛化能力会下降。

核宽度 σ 对模型的泛化能力也起到很重要的作用， σ 表示的是核函数的径向范围， σ 的值越大，那么径向范围越大。从初始阶段，慢慢增大 σ 的值，模型泛化能力会变好，当达到某个值时，泛化能力最好，再继续增大 σ 的值，泛化能力又会下降。

总之，惩罚因子 C 决定了对错误分类的容忍程度，核宽度 σ 决定了范围大小，二者共同作用影响模型的泛化能力。

3.3.4.4 改进粒子群优化 SVM 参数

SVM 虽然能通过惩罚因子和核宽度很好地将各类问题进行分类，不过这两个参数的设置还是随机的，并不能找到最合适的。而标准的粒子群算法可以通过个体最优和群体最优综合来优化参数，有着较好的搜索能力，却会出现某个个体早熟，或出现容易陷入局部最优解，过早进入收敛的情况，因此需要有个标准能很好地引导粒子向着全局最优部分进行搜索，更大程度地寻找最优解。

在此前，Eberhart 等人提出对惯性权值进行线性改变，进而能提升粒子在后期的搜索能力^[52]，惯性权值的计算公式如下所示：

$$w = w_{\max} - \frac{t}{T_{\max}} \times (w_{\max} - w_{\min}) \quad (3.26)$$

在式（3.26）中， w 是惯性权值， $w_{\max}$ 是惯性权值的最大值， $w_{\min}$ 是惯性权值的最小值， t 是迭代代数， $T_{\max}$ 为迭代代数最大值。

该方法在一定程度上优化了参数的查找，但是该方法属于是线性的递减，不能很好反应粒子的寻找轨迹空间，使得寻找的精度不够高。

根据以上的描述，为了更好的模拟粒子的寻找方向，需要粒子自适应的改变寻找的方向。本文提出一种根据粒子自身的适应度的权值来进行动态调整寻找方向，所以改进的惯性权值的公式为：

$$w = w_{\max} - \frac{(w_{\max} - w_{\min})}{1 + 2^k} \quad k = \frac{f_i - f_{avg}}{f_g - f_{avg}} \quad (3.27)$$

式（3.27）中， f_i 为粒子当前适应度， f_g 为种群中最优适应度， f_{avg} 为种群平均适应度。

当 w 值较大时，粒子在全局中活动范围变化较大，可到达的范围广，在全局搜索能力强，在局部范围不能很好地寻参；当 w 值较小时，粒子在全局中活动范围变化较小，可到达的范围窄，在局部搜索能力变强，在局部范围更好地寻参，算法更易收敛。不会在整个搜索过程保持较线性的搜索能力，从而更灵活的适应参数寻优。因此，这种自适应粒子群算法不再是线性地搜索参数，而是动态改变权值，使 w 随着代数非线性改变。

为了更好描述粒子移动情况，移动示意图如图 3.17 所示。其中 x_i 为粒子在多维空间中的一个起始点， p_i 为个体的最优情况， g_i 为群体最优情况。而粒子群速度有三部分决定，分别是惯性速度 $w \cdot v(t)$ ，具有记忆性； $c_1 r_1 (P_i - x_i)$ 为个体经验的部分， $c_2 r_2 (G_i - x_i)$ 为种群的经验部分共同作用。下一个位置由现有位置和速度矢量共同决定。

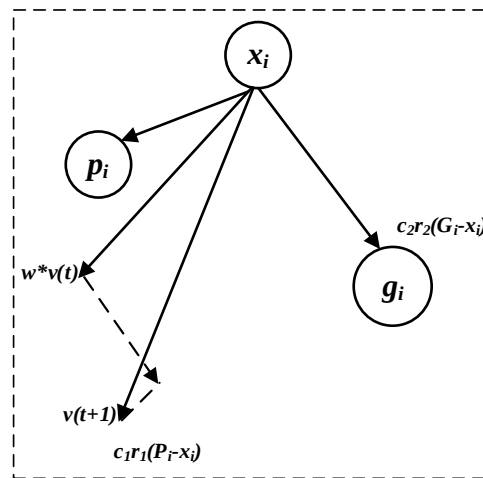


图 3.17 粒子移动示意图

具体搜索流程是：用粒子代表核参数和惩罚因子，在有解空间中寻找合适解，每个粒子都有两个属性，一个是适应度另一个是速度，为防止速度过快，可以限制粒子的速度。对粒子初始化完之后，粒子开始迭代运行，根据式 (3.27) 的方法不断进行更新。改进的自适应粒子群算法如图 3.18 所示。

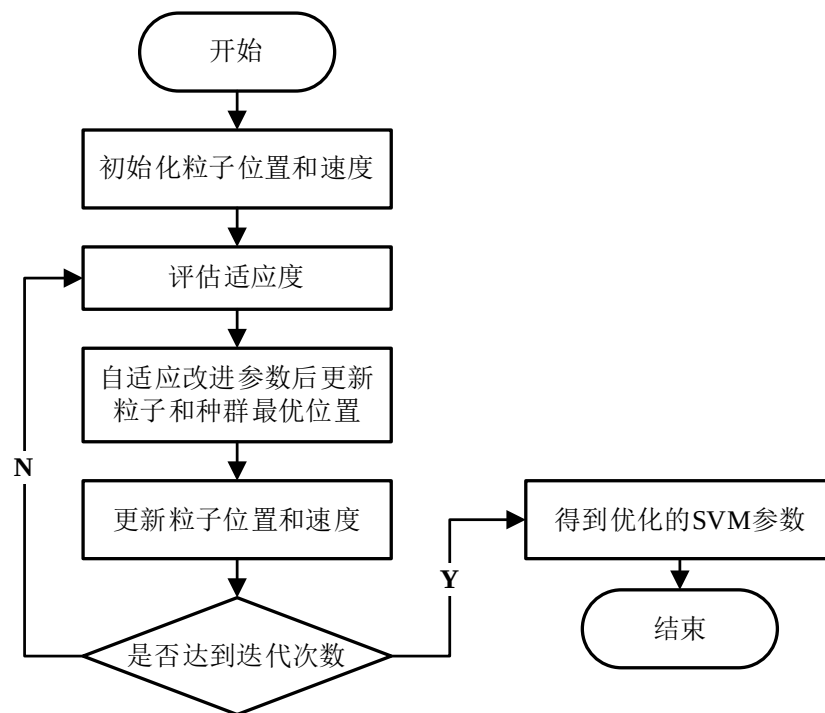


图 3.18 改进粒子群算法流程图

为证明自适应优化参数 SVM 的正确性，现使用标准测试函数 Rastrigin 函数，Griewank 函数，Ackley 函数对其进行验证。其公式和性质如表 3.6 所示：

表 3.6 测试函数公式和性质

函数	表达式	定义域	维度
Rastrigin	$f_1(x) = \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i) + 10)$	$[-5.12, 5.12]$	50
Griewank	$f_2(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \left(\frac{x_i}{\sqrt{i}} \right) + 1$	$[-600, 600]$	50
Ackley	$f_3(x) = -20e^{-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}} - e^{\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)} + 20 + e$	$[-5, 5]$	50

现通过标准粒子群算法（以下简称 PSO），基于线性惯性权值的粒子群算法（以下简称 LPSO），基于自适应惯性权值粒子群算法（以下简称 SA-PSO）。在 PSO 和 LPSO 中， $c_1 = c_2 = 2$ ，在 SA-PSO 中， $c_1 = c_2 = 2.05$ 。种群大小 $N = 20$ ，最大权值 w_{max} 为 0.9，最小权值 w_{min} 为 0.4， r_1, r_2 为 $[0, 1]$ 区间内的随机数。服从正态分布。在 MATLAB 中独立运行 20 次并取平均值，并进行分析。

三个测试函数适应度曲线如图 3.19 所示：

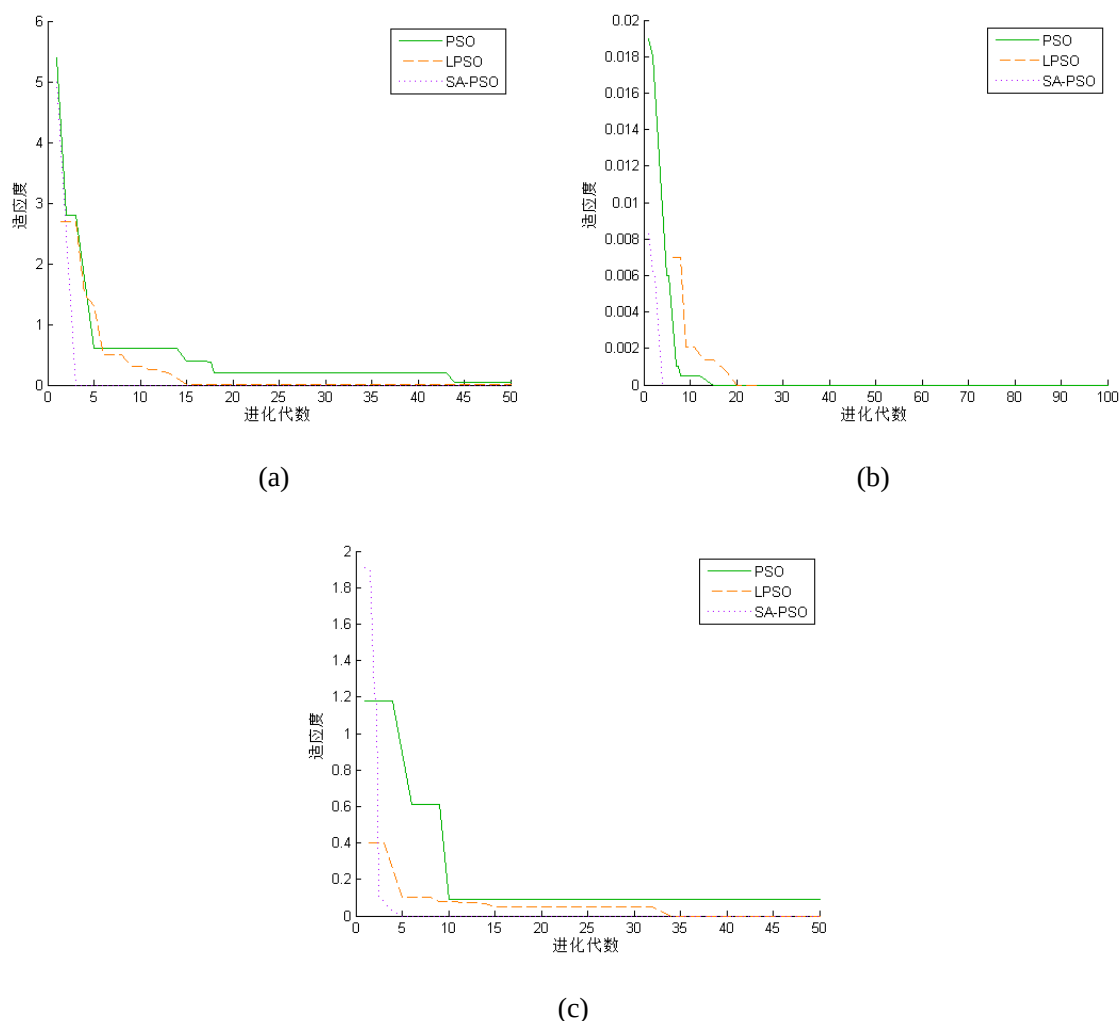


图 3.19 各函数适应度测试结果

(a) R 函数适应测试结果 (b) G 函数适应测试结果 (c) A 函数适应测试结果

从图 3.19 可以看出，相对于前两者，SA-PSO 在三个测试函数表现中，可以更早的函数收敛，在较少的迭代次数得到函数最优值，尤其在 Rastrigin 和 Ackley 函数中，表现的更为明显。

表 3.7 最优解测试结果

函数	算法	最优值	平均值	极差
Rastrigin	PSO	8.2023e-04	0.0313548922	0.1382
	LPSO	1.8496e-05	0.0372351243	0.0083
	SA-PSO	1.2311e-06	2.434274e-04	0.0020
Griewank	PSO	5.9632e-07	4.127537e-04	0.0073
	LPSO	1.4311e-09	4.672384e-07	4.2494e-06
	SA-PSO	1.0311e-14	6.325697e-09	7.5465e-08
Ackley	PSO	0.0079	0.032655	0.0812
	LPSO	7.5683e-04	0.0072124765	0.0163
	SA-PSO	1.3311e-05	2.473278e-05	5.3456e-04

由表 3.7 可以看出，对测试函数寻优方面自适应惯性权值粒子群算法明显优于其他两种算法，更贴近函数最优解 0。尤其是对于 Griewank 函数，有着更好的效果，得到的最优值相对于其他算法提高了好几个数量级。从平均值和极差来看，SA-PSO 在性能上也是优于其他两种算法，有效性可以得以验证。

3.4 测试和结果分析

总体诊断模型是通过对采集到的振动信号，先进行 EEMD 的信号分解，将分解完的信号再进行阈值降噪处理，对符合要求的 IMF 分量计算其能量熵作为 SVM 分类器的输入特征向量。对现有标准的 SVM 两个参数 C 和 σ 进行参数优化，根据粒子群特点，对粒子群惯性权值的值进行适当优化，更有利于全局搜索，不局限于局部最优值。总体诊断的流程模型如图 3.20 所示。

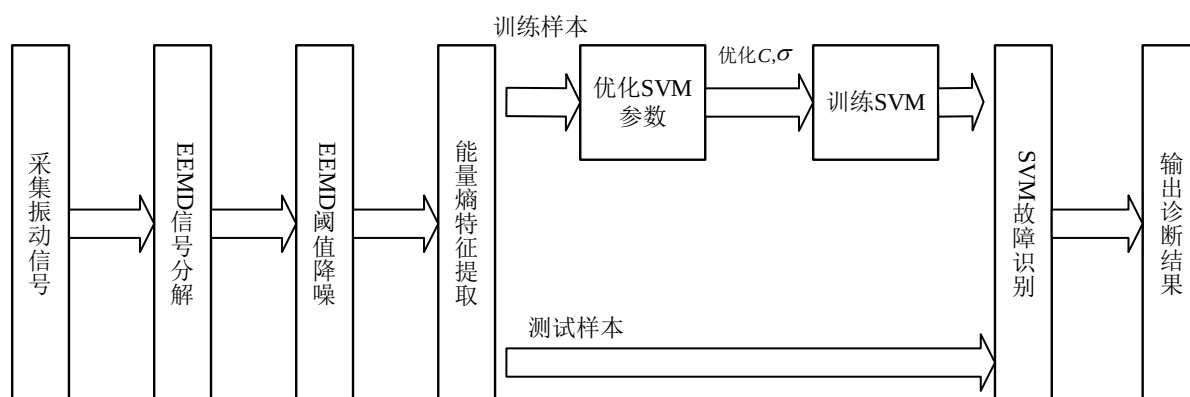


图 3.20 总体诊断流程模型

本节测试结果分为两部分，第一部分是通过对在 SVM 随机参数下，对信号预处理分为 EMD 和 EEMD 故障检测结果对比；第二部分是通过对信号预处理均为 EEMD，使用 PSO, LPSO, SA-PSO 对 SVM 参数调优的故障检测结果对比。

在测试中，分别把轴承的四种状态标为 1, 2, 3, 4。共选取 240 组数据，选取前三分之二作为训练数据，再选取剩余组数作为测试数据，具体情况如表 3.8 所示。

表 3.8 数据标签

状态	总数据量	训练数据	测试数据	标签
正常状态	240	160	80	1
内圈故障	240	160	80	2
外圈故障	240	160	80	3
滚动体故障	240	160	80	4

3.4.1 基于 EMD 和 EEMD 算法故障识别结果对比

在国红波等人论文中,对缝纫机轴承信号分解采用的是 EMD 算法。此次测试通过 EEMD 分解算法与其进行的对比, SVM 参数是随机设置。通过测试的数据来对比两个算法分类的准确性。

图 3.21 为基于 EMD 的 SVM 在测试集上的结果。图 3.22 为基于 EEMD 的 SVM 在测试集上的结果。表 3.9 为两种算法分类准确度。

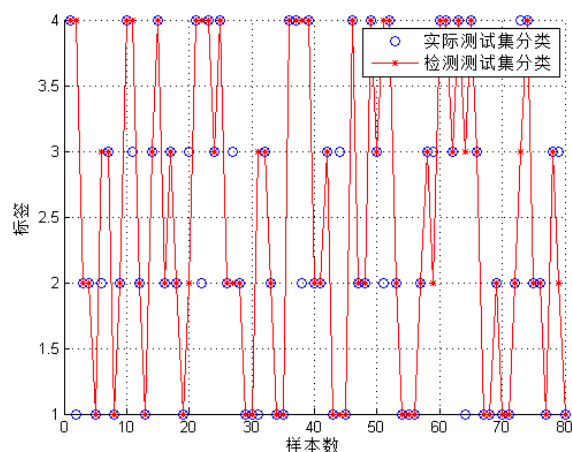


图 3.21 基于 EMD 的 SVM 测试结果

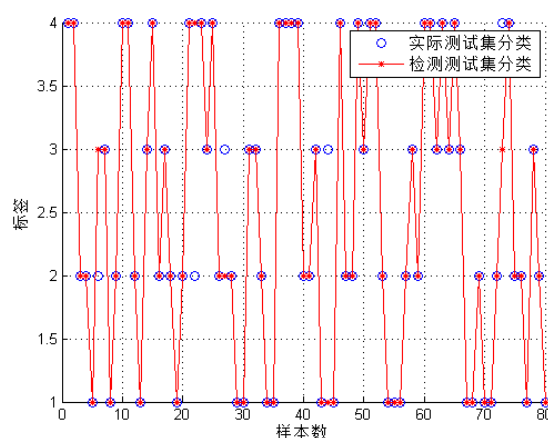


图 3.22 基于 EEMD 的 SVM 测试结果

表 3.9 两种算法分类准确度

	1	2	3	4	平均准确率
基于 EMD 分类	90%	89%	90%	92%	90.25%
基于 EEMD 分类	97%	94%	95%	96%	95.5%

通过上图和上表可知,在测试样本表现中,使用 EEMD 处理采集的信号,并经过阈值处理之后,在使用相同的 SVM 参数情况下,可以获得更好的识别结果。同时,在对正常状态识别中,有着较优的效果,正确率提高了 7%。且对四种状态识别的平均正确率提升较大,从而证明了使用 EEMD 处理信号相比于 EMD 有着更优的效果。

3.4.2 基于 EEMD 的优化参数 SVM 故障识别结果对比

现通过给标准粒子群 SVM 分类器,线性惯性权重的粒子群 SVM 分类器和自适应惯性权重粒子群 SVM 分类器训练 160 组样本,并将 80 组数据作为测试样本进行测试。通过分类准确度和进化代数来表现算法的实用性。

首先通过传感器采集轴承运行时的信号,再通过 EEMD 分解及与阈值降噪处理获得不同信号各个分类的结果。

四种状态下的原始信号部分波形如图 3.23 所示。

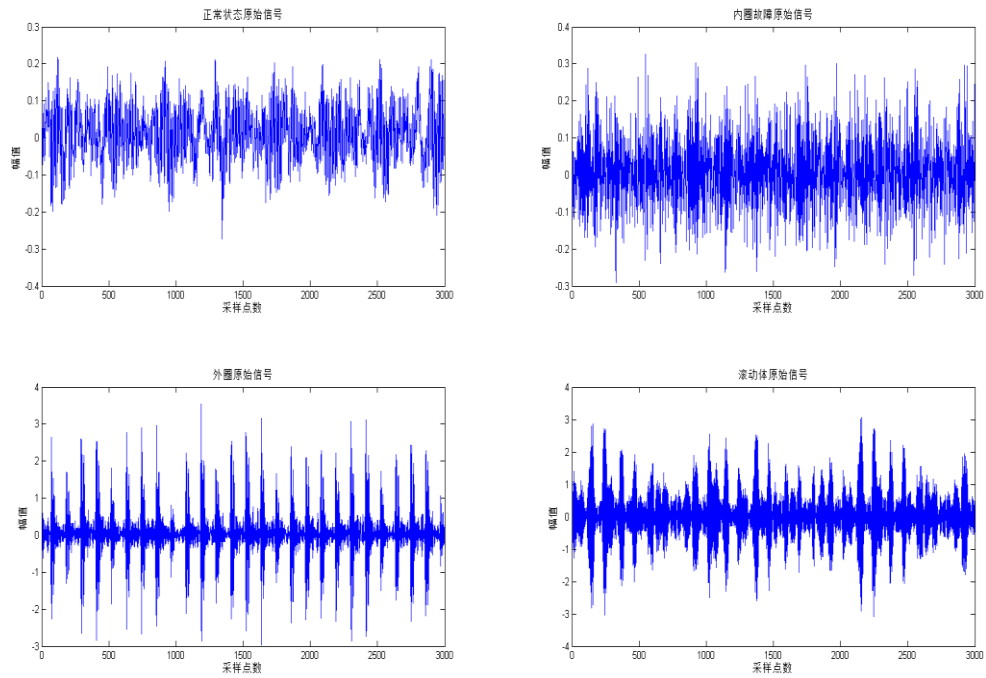


图 3.23 四种状态下原始信号

对采集到的信号加入 0.2 倍白噪声，对整个复合信号进行研究，对数据进行采样，每组共 3000 个数据点，之后对信号进行 EEMD 分解。并根据之前小节的对比，采取前 6 个分量进行测试。四种状态结果如图 3.24，图 3.25，图 3.26，图 3.27 所示。

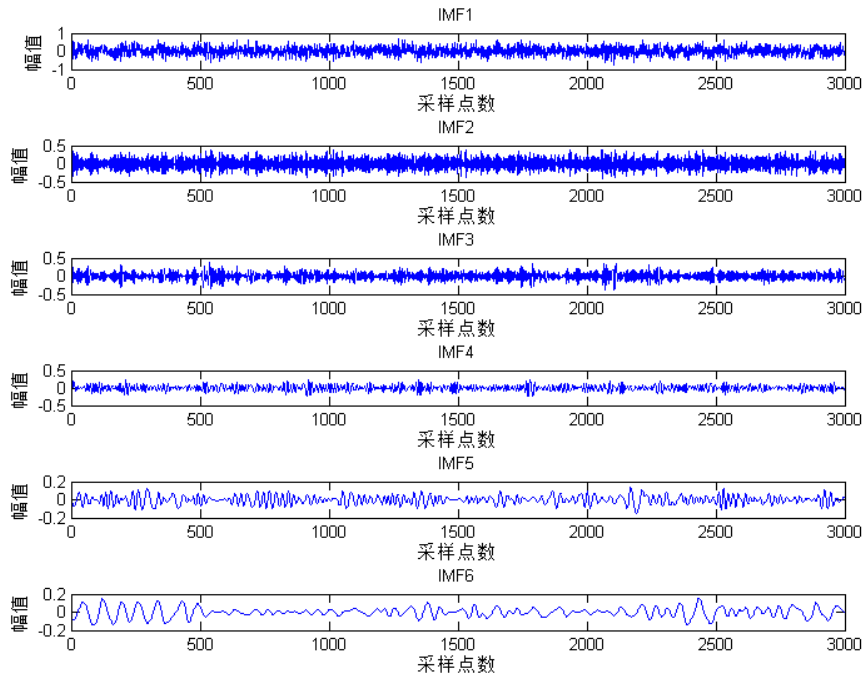


图 3.24 正常状态前 6 阶 IMF 分量

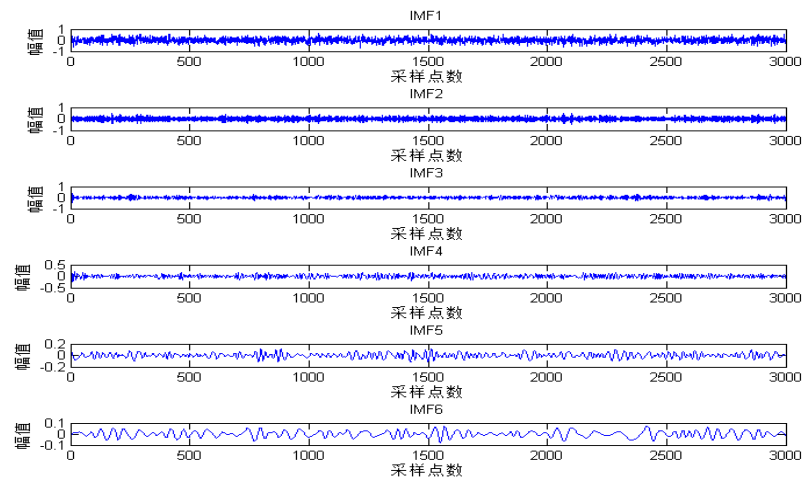


图 3.25 内圈故障前 6 阶 IMF 分量

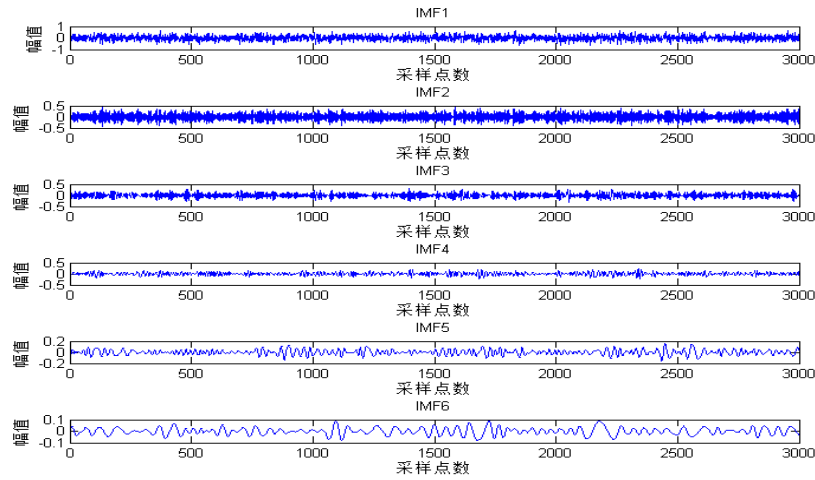


图 3.26 外圈故障前 6 阶 IMF 分量

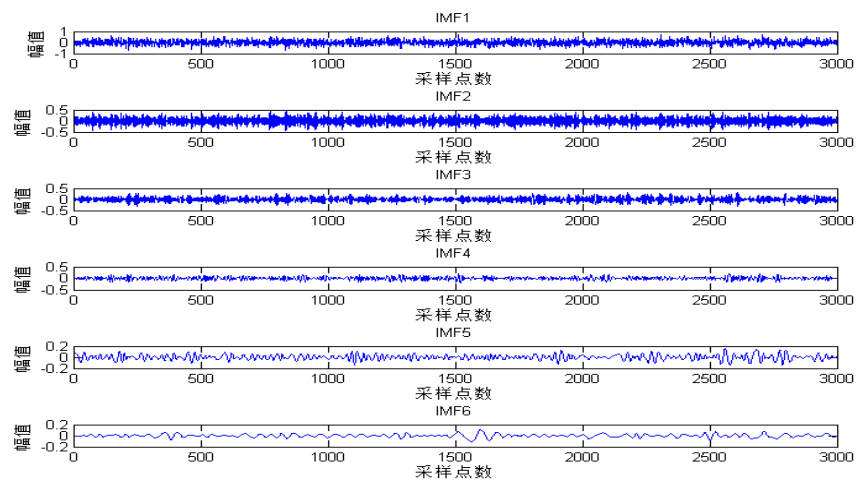


图 3.27 滚动体故障前 6 阶 IMF 分量

当获得了 IMF 分量之后,选取能量熵对信号进行特征提取处理。对能量熵部分测试数据如表 3.10 所示:

表 3.10 能量熵部分测试数据

	IMF1	IMF2	IMF3	IMF4	IMF5	IMF6
正常	0.2682	0.3632	0.1739	0.2541	0.3386	0.1315
正常	0.2625	0.3891	0.1788	0.2298	0.3107	0.1335
正常	0.2743	0.3586	0.1994	0.2604	0.3573	0.1669
内圈故障	0.3875	0.3654	0.3193	0.1234	0.1237	0.01302
内圈故障	0.3852	0.3373	0.3534	0.1235	0.1498	0.01178
内圈故障	0.3591	0.3448	0.3672	0.1383	0.1177	0.01571
外圈故障	0.1244	0.1327	0.1325	0.1111	0.0751	0.0143
外圈故障	0.1417	0.1310	0.1367	0.1172	0.0713	0.0136
外圈故障	0.1136	0.1295	0.1422	0.1089	0.0749	0.0140
滚动故障	0.1389	0.1331	0.1442	0.1153	0.0692	0.0013
滚动故障	0.1338	0.1358	0.1401	0.1123	0.0685	0.0011
滚动故障	0.1404	0.1412	0.1477	0.1162	0.0674	0.0022

优化参数的基本流程是:进行初始化之后,通过适应度函数评价每个粒子行为是否更接近目标,该指标也可称为适应度,通过适应度和全局最优目标进行比较,若优于全局目标,那么将此适应度作为全局最优,更新粒子速度和位置,如此也能获得个体及群体最优位置。最后,若达到迭代次数则停止,否则,继续进行寻优操作。

在初始化时,将种群进化代数设置为 100,种群数量为 20。最大权值 w_{max} 设置为 0.9,最小权值 w_{min} 设置为 0.4。粒子群位置和速度成线性关系,系数为 0.6。具体参数设置如表 3.11 所示:

表 3.11 初始化参数

参数	设置
种群大小	20
最大迭代次数	100
学习因子	PSO-SVM: $c1=c2=1.5$
	LPSO-SVM: $c1=c2=1.5$
	SA-PSO-SVM: $c1=c2=2$
惯性权值	$w_{max}=0.9, w_{min}=0.4$
惩罚因子	$0.01 \leq C \leq 100$
核宽度	$0.01 \leq \sigma \leq 10$

为验证选择合适的初始值对粒子群寻参的结果的影响，现通过纵向对比，选取测试集对参数进行验证，测试中对核参数 g 的值以 0.1 间隔变化。横坐标表示核参数 g 的取值变化，左纵坐标表示针对数据集的分类正确率；右纵坐标表示适应值 F 的变化。两曲线分别表示随着核参数 g 和适应值 F 变化趋势。从中可以看出，正确率随着适应度的改变而改变，并且两者的走势基本吻合。因此，适应度准确地反映数据集的可分度，也与分类器的分类效果相关。结果如图 3.28 所示。因为本实验中选取核参数初始值为 0.9472。

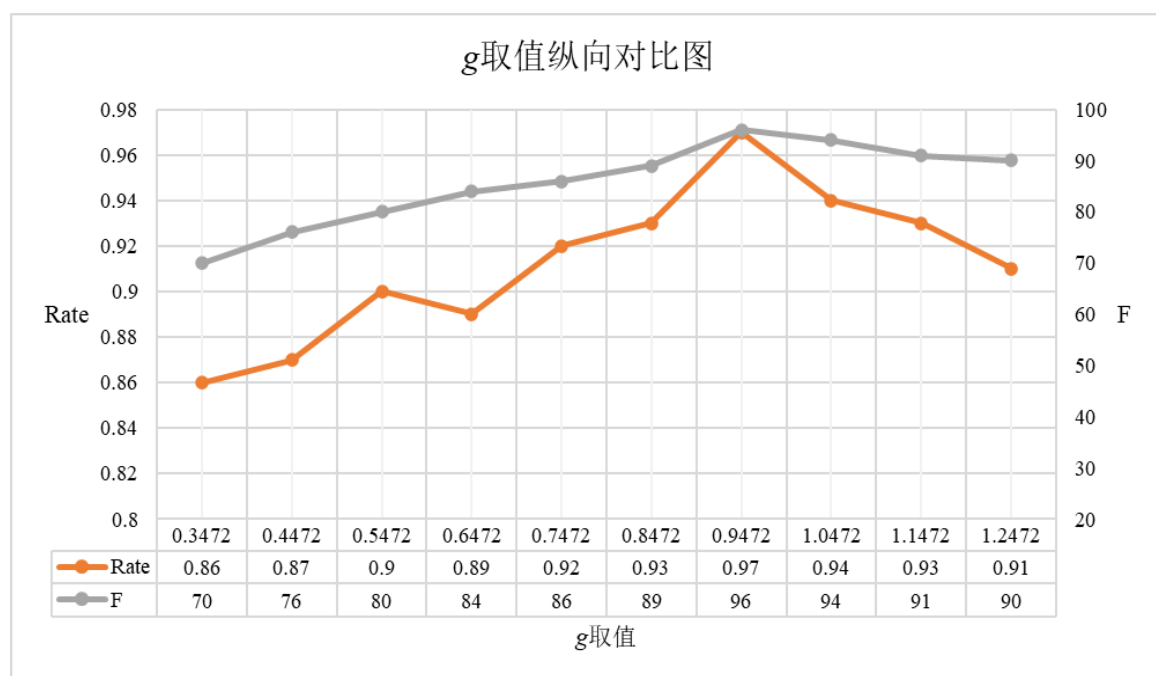
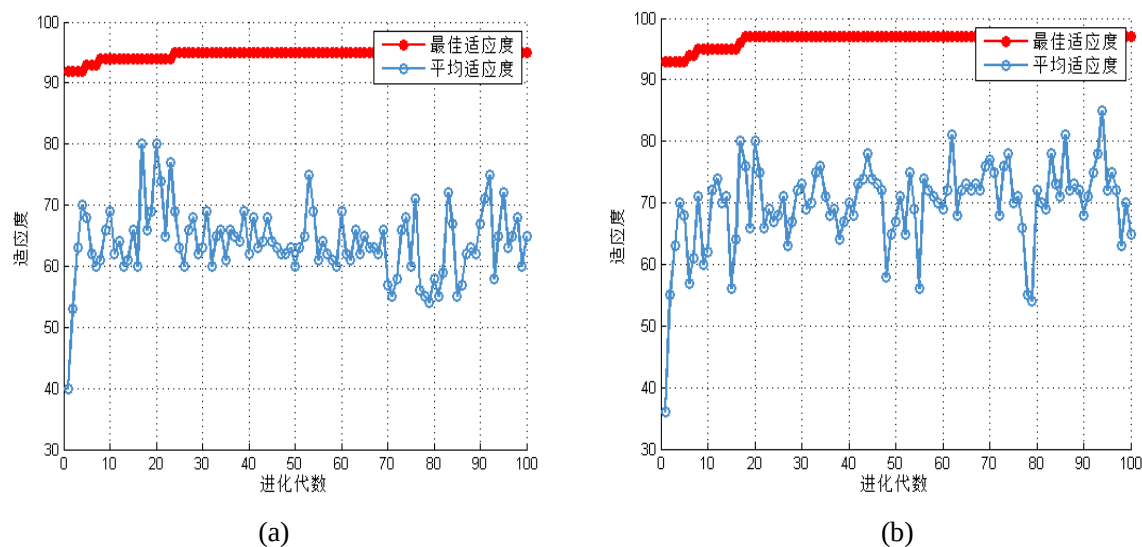
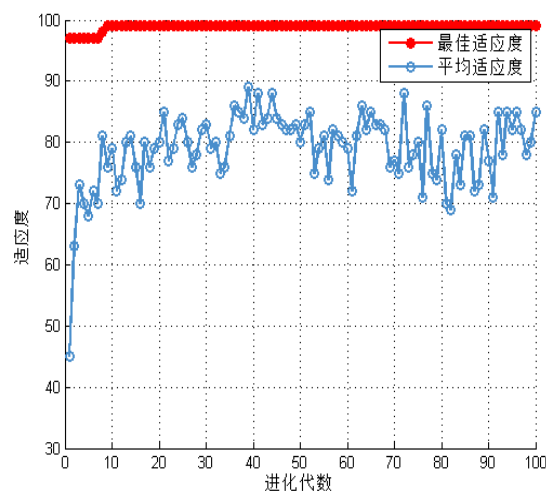


图 3.28 g 取值纵向对比图

现通过多次对标准粒子群算法，线性惯性权值粒子群，自适应惯性权值粒子群对 SVM 进行参数优化。

三种算法的适应度曲线如图 3.29 所示：





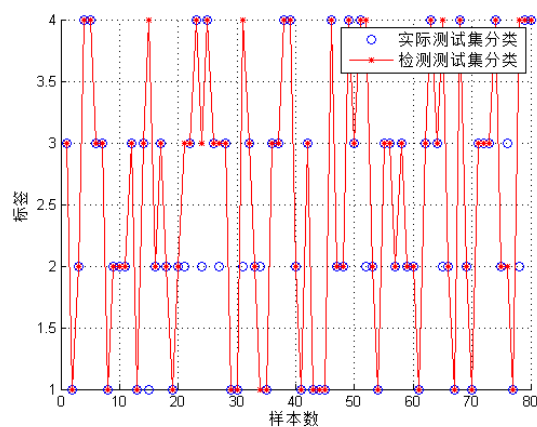
(c)

图 3.29 各算法适应度曲线

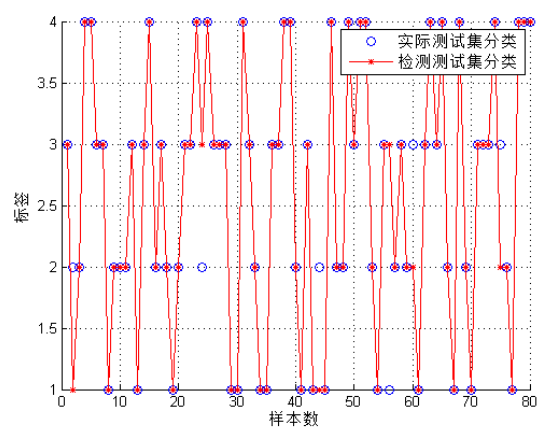
(a) PSO 适应度曲线 (b) LPSE 适应度曲线 (c) SA-PSO 适应度曲线

在图 3.29 中,可以看出随着进化代数增多,适应度的值不断变大。因为若粒子当前适应度的值大于需要优化全局最优的值,那么该值就需要作为最后优化的目标。从图中还能看出使用 PSO 算法优化 SVM, 平均适应度在 65%作用, 最佳适应度是在 95%。使用基于 LPSE 算法优化 SVM, 平均适应度在 72%作用, 最佳适应度是在 97%。使用 SA-PSO 算法优化 SVM, 平均适应度在 82%作用, 最佳适应度是在 99%。

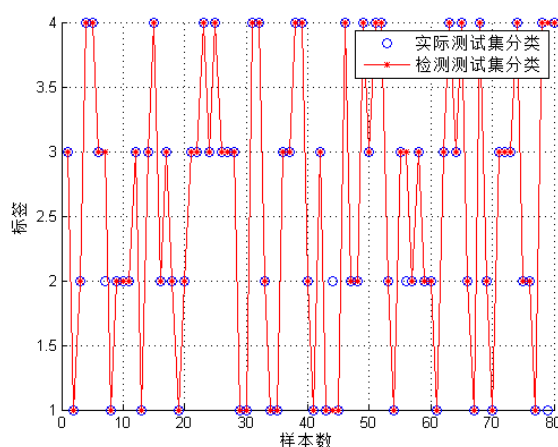
三种算法分类结果如图 3.30 所示, 三种算法平均准确度如表 3.12 所示:



(a)



(b)



(c)

图 3.30 各算法分类结果

(a) 基于 PSO 分类结果 (b) 基于 LPSO 分类结果 (c) 基于 SA-PSO 分类结果

表 3.12 三类算法平均准确率

	1	2	3	4	平均准确率
PSO 准确率	94%	92%	91%	92%	92.25%
LPSO 准确率	97%	98%	97%	97%	97.25%
SA-PSO 准确率	100%	99%	98%	99%	98.6%

从上图中可以看出，通过 80 组的测试数据，相比于后两者，标准粒子群优化参数的误分类的概率较大，集中在内圈故障诊断方面。横向对比三个图，可以看出自适应惯性权值粒子群优化的 SVM 参数的算法有着较好的分类结果。

从上表中可以看出三种算法对正常状态的检测准确率最高，平均准确率在 96.3%。PSO-SVM 对四种不同故障平均准确率是 92.25%，LPSO-SVM 对四种不同故障平均准确率是 97.25%，SA-PSO-SVM 对四种不同故障平均准确率是 98.6%。通过对比验证，本方案中设计的自适应惯性权值粒子群优化的 SVM 参数相比于标准的粒子群和线性惯性权值优化的 SVM 参数有着较好的提升且有效。

3.5 本章小结

本章首先对缝纫机轴承故障进行介绍，并对现有故障诊断方案中的 EMD 分解算法和分类算法进行分析，针对这两方面进行优化改进，设计了一种基于 EEMD 和改进粒子群算法 SVM 的故障检测方案。将采集到的信号通过改进 EEMD 算法分解，选择能量熵作为特征值，作为分类器的输入向量，并通过自适应惯性权值来优化参数 c 和 g ，同时通过测试函数验证了基于自适应惯性权值优化 SVM 参数可以提高算法的收敛速度。最后通过对比分析 EMD 算法和改进 EEMD 算法在对分类结果的影响，改进 EEMD 算法可以将分类正确率较大幅度地

提高，且通过对比基于 EEMD 的三个优化 SVM 参数算法在故障诊断中分类结果，自适应惯性权值粒子群算法有着更好的表现，对各种轴承状态的平均诊断正确率在 98.6%。在原来方案基础上性能有着一定程度的提升。

第四章 基于改进遗传算法的生产线平衡问题优化

自从流水线技术应用到生产中之后，极大地提升了生产效率。但由于每道工序时间不均匀，会出现部分产品堆积在某道工序上，造成生产效率不能最大化。由此引申出生产线平衡问题。针对此类问题，本章首先对生产线流程展开分析，之后对传统解决生产线平衡问题的算法进行分析，针对其不能很好获得最优解情况，设计了一种基于改进的遗传算法方案并结合实际情况对生产线工序进行优化，使得各工序更加平衡，有效地提高了该公司生产效率。

4.1 生产线流程分析

Z 公司是一家生产衬衫的大型企业，现以该公司生产衬衫生产线流程为例对生产过程进行分析。在生产过程中，将缝制过程总体分为三大部分：缝前过程，缝中过程，缝后过程。缝制过程如图 4.1 所示。

其中缝前过程为面料的裁剪及面料的整理；缝中过程是对各种面料进行加工；缝后过程是加工之后的成品熨烫与包装。

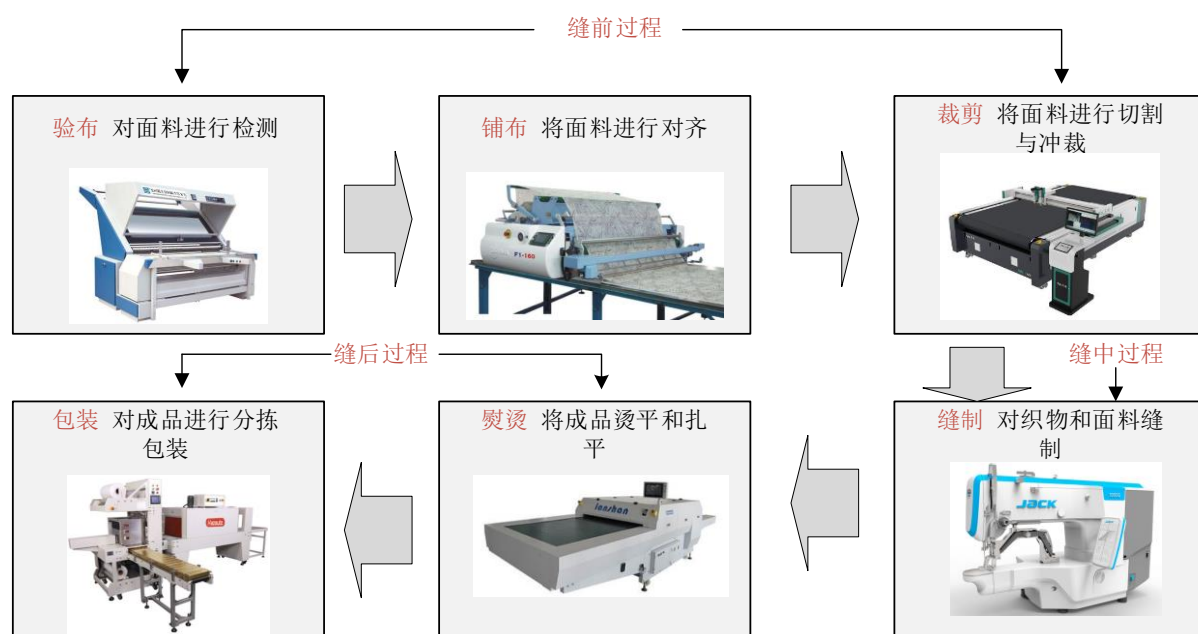


图 4.1 缝制过程示意图

在衬衫生产过程中，同样类似以上过程，且缝制阶段是整个生产线最重要的组成部分，先对需要缝制的部分进行适当的裁剪，在缝制阶段需要对前后衣片，袖子，领口等进行处理，在分别处理完这些部分之后，再对其进行整体的整合和卷边底工作，最后进行整体的熨烫和

包装。衬衫生产具体流程如图 4.2 所示。

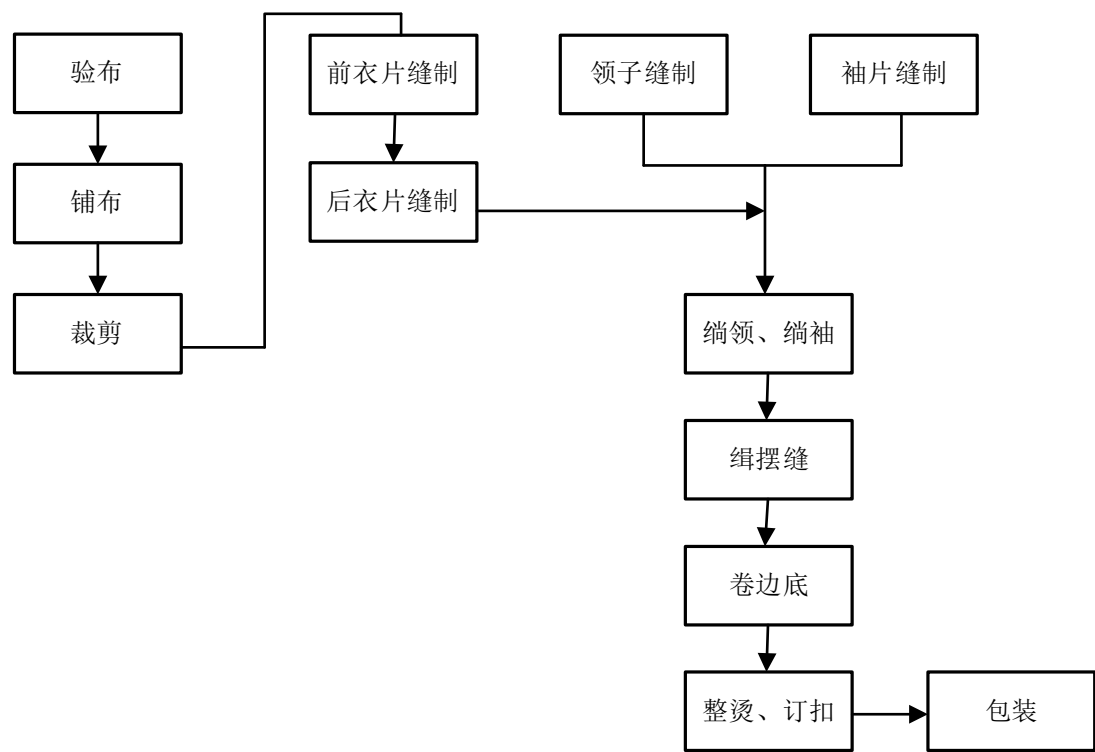


图 4.2 衬衫实际生产流程图

其中生产流程和工作站安排如表 4.1。

表 4.1 衬衫生产流程划分

工作站序号	工作序号	工作单元
1	1	验布
	2	铺步
	3	裁剪
	4	缝布袋口
2	5	缉袋口
	6	合缉止口
	7	合肩缝
	8	缉明线
3	9	勾上口
	10	绱领
	11	合前袖
	12	合后袖
4	13	绱袖
	14	合侧缝
	15	勾缉领口
	16	固定胸衬
5	17	缝垫肩
	18	卷边底

工作站序号	工作序号	工作单元
6	19	整烫、订扣
	20	包装

在缝制过程中，需要将领口，前后袖子和前后衣分别缝制，当各部分都缝制完成之后，将其进行组合，进行合缝处理，和绱操作，最后整体再进行整合和细节上的处理。在实际作业流程中，常会出现绱领，绱袖工序中，有大量材料堆积，而负责袖片缝制工人总是能相对清闲，导致其他工序工人抱怨颇多，各工序负荷极度不平衡。因此，急需解决生产线平衡问题。

生产线平衡问题就是需要解决一类在约束条件下因各工作站之间产能不均衡所引起的低效率生产问题。该约束条件是：各个工序之间要遵循先后顺序且每个工作站负荷总和不超过生产一件正常产品所用的时间。其目的是提高生产线效率，并降低各工序之间消耗带来的损耗以实现利益最大化^[53]。根据上述的约束条件可以看出，优化生产线平衡问题就是将现存的生产线之间的工序进行优化和组合，使其尽量能达到现有情况下的最优解。

在生产线平衡问题上，有以下参数可以衡量生产线效率。

1) 平衡率(Balance Rate)：

平衡率是各工序时间相加之后和节拍的比值，表示的是整条生产线工作的有效利用率。节拍指的是正常工作情况下，完成一个新产品所用的时间，通常用 c 来表示。

$$\partial = \frac{\sum_{i=1}^n t_i}{s \times \max(ST_i)} \times 100\% \quad (4.1)$$

在式(4.1)中：

∂ ：平衡率。

s ：工作站个数，是指完成一道或多道工序，工作人员在某个特定位置完成工作。

$\max(ST_i)$ ：将所有工位中时间最大值作为生产节拍。

2) 均衡指数(Smooth Index)：

通过计算平方根来反映各个工作站之间的负荷差异，这种离散情况值越大，那么各个工作站之间负荷作业差距就越大。若 SI 值达到 0 时，说明生产线均衡了。

$$SI = \sqrt{\frac{\sum_{i=1}^j (\max ST_i - ST_i)^2}{s}} \quad (4.2)$$

在式(4.2)中：

SI 指的是均衡指数。

4.2 标准遗传算法方案分析

上世纪七十年代, Bagley J.D 提出了遗传算法这个概念^[54, 55]。之后, John H.Holland 进行深入研究, 提出了一个数学模型, 该模型允许这种复杂相互作用的非线性关系, 专注于具有以非线性方式相互作用的多个因素的系统^[56, 57]。遗传算法目前在工业控制, 人工智能, 生物工程领域被广泛应用。

4.2.1 标准遗传算法步骤

遗传算法是常见的启发式类算法之一, 中心思想与自然进化理论是一致的。该算法体现自然选择的过程, 每个种群都希望能保留种群的特征, 故选择种群中优秀个体与相似种群进行繁殖, 对适合生存的个体基因进行变异重组, 可以获得更优个体。这是遗传算法的核心思想^[58, 59]。

遗传算法是对一个种群中的所有个体进行操作, 再建立数学模型。这个模型基础是带有一定特性的种群, 该特性类似于生物学上的染色体。在这个种群中, 一种染色体对应一种解。之后根据问题的优化方向需要构造一个适应度函数, 根据此函数, 对种群中的个体进行复制, 交叉, 突变, 繁殖下一代的个体。对繁殖生成的个体进行筛选, 若此代个体达到足够精确的进度, 那么再重复之前繁殖的过程。直到产生符合条件的个体。此个体就是该问题的最优解。

在模型中具体操作分为以下几个部分:

- 1) 编码方式: 和人类的染色体相同, 在染色体上会有一定的基因排列。在该数学模型中, 通过对种群中的个体进行编码, 在初始化种群时进行编码。方便对问题的处理。
- 2) 译码方式: 一般情况下, 译码方式是对整个过程进行一个判断方式, 是根据我们需要优化目标来选择的。
- 3) 选择方式: 通过某种标准, 在种群中筛选出更优的个体, 进行繁殖, 有利于种群适应性。
- 4) 交叉: 和生物中的染色体交叉原理相似, 通过对父代中的编码根据一定概率进行交换, 从而得到基因优秀的子代。
- 5) 变异: 在生物进化的过程中, 若只有染色体交叉遗传, 那么生物不会形成多样性。在遗传过程中, 会一些点位会有一定概率发生基因变异, 会形成和原始种群不同的群体, 填补在交叉过程中缺失的基因。变异示意图如图 4.3 所示:

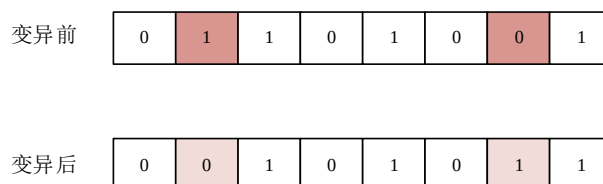


图 4.3 个体变异示意图

遗传算法基本步骤示意图如图 4.4 所示。

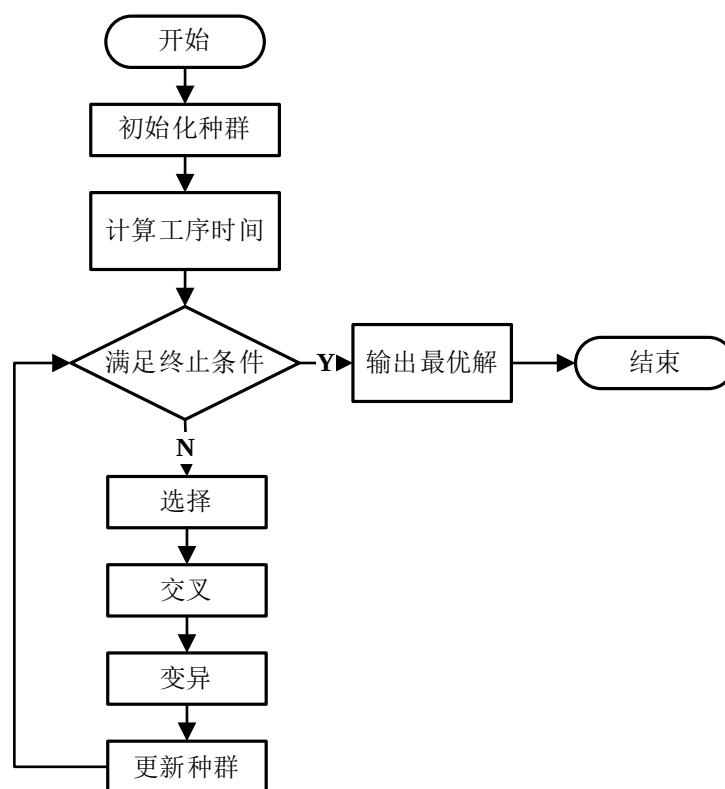


图 4.4 标准遗传算法基本步骤

4.2.2 优化方向

根据传统的遗传算法，在单种群进行一定概率的交叉，变异，在前期的效果还是可以的，种群中的差异会表现出比较大的差异，但随着遗传，交叉等操作次数增多，剩下的个体都比较优良且很多都是近亲。这样随着近亲繁殖，会导致繁殖的后代性能下降，可搜索的空间不能再扩大，容易过早进入收敛状态这样获得局部解的概率提高。若要提高搜索性能，通过产生两个种群，提高搜索范围。并通过选取合适的适应度函数来选择需要再次繁殖的个体。

4.3 改进遗传算法的生产线平衡优化设计与实现

研究此类问题根据最后需要优化的目标不同，可分为 3 大类。

第一类生产线平衡问题：这类情况一般是在生产线还没完全装配成功之前进行研究的，对工作站个数进行优化，需要考虑到生产的数量和人力成本等问题。

第二类生产线平衡问题：这类情况一般是生产线已经在运行中，设备和人员情况基本稳定不变，但是由于外界客观因素需要缩短生产线生产的时间。通过优化生产节拍来提高单位时间的生产效率。

第三类生产线平衡问题：此类问题对工作站个数和生产节拍都不做限制要求。可以根据

延迟率计算公式，得出一个数对 (s, c) ，使得 $s \cdot c$ 这个值最小即可。这类情况结果不确定，可根据设备具体情况进行分析。

本文着重对第二类生产线平衡问题进行研究。

4.3.1 总体流程设计

双种群遗传算法是多种群遗传算法中的一种。选取两个独立遗传的群体，两个群体互不关联，分别有独立的进化方式和变异算子。当两个群体各自繁衍之后，根据一定的规则进行群体交换，以此能获得更大的群体可搜索性和多样性。设种群的交叉概率为 P_c ，突变概率为 P_m 。其中子种群 1 的 P_{c1} 和 P_{m1} 值较大，称为探测子种群，因为该种群交叉和突变概率较大，所以在进化过程中能更好地在搜索空间中搜索，不断获得新的超平面，可以保持种群高效搜索范围，相反地子种群 2 的 P_{c2} 、及 P_{m2} 值较小，称为开发子种群，在搜索全局过程中，获取局部最优的个体，并保留下来。改进遗传算法流程图如图 4.5 所示。

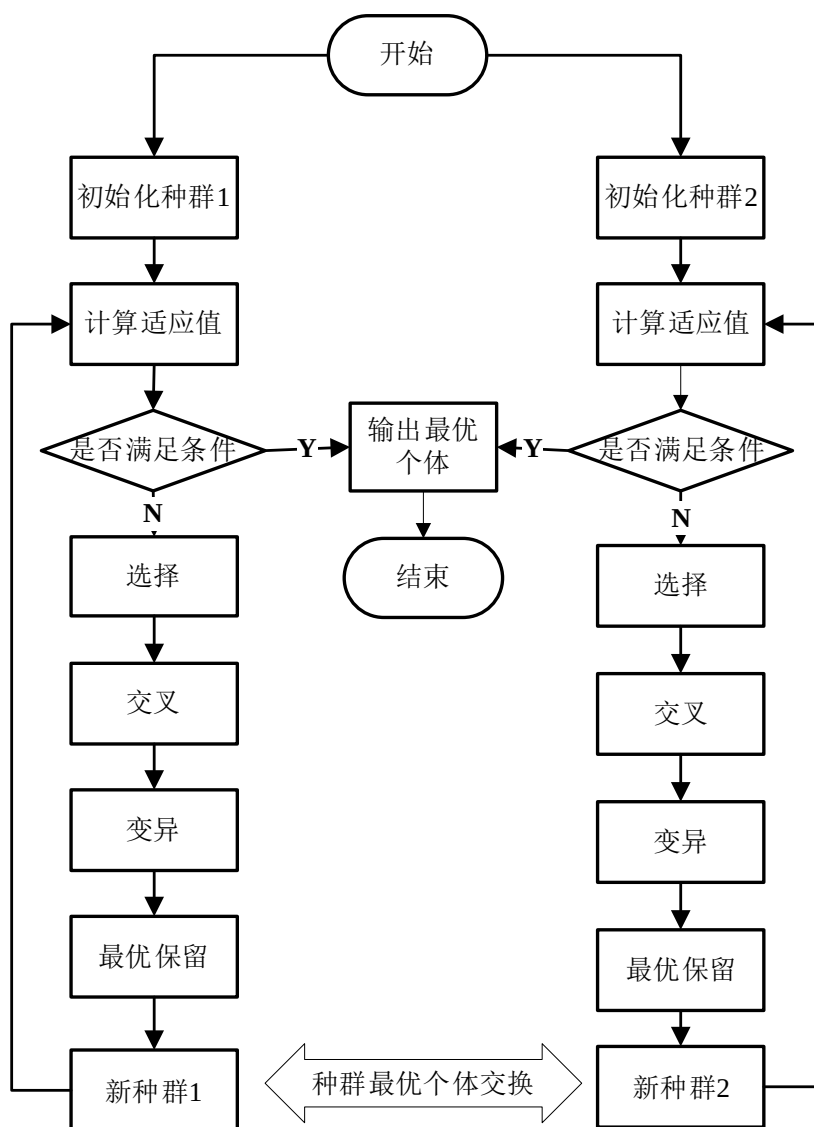


图 4.5 改进遗传算法流程图

4.3.2 初始化种群和选择实现

(1) 编码操作

在生产过程中，工序是有一定的先后顺序，形成一定的约束条件。也就是说，只有当前面工序操作完成之后，后面工序才能开始被执行操作。又被称为优先级关系或先后顺序。

根据生产线的流程画出一个生产线的有向图，在有向图中，每个结点的位置和工序的优先级相对应。每个结点位置对应每道工序，保证基因组的完备性。本节所阐述的方法将实际生产线抽象为有向图。图 4.6 是抽象化生产线示意图：

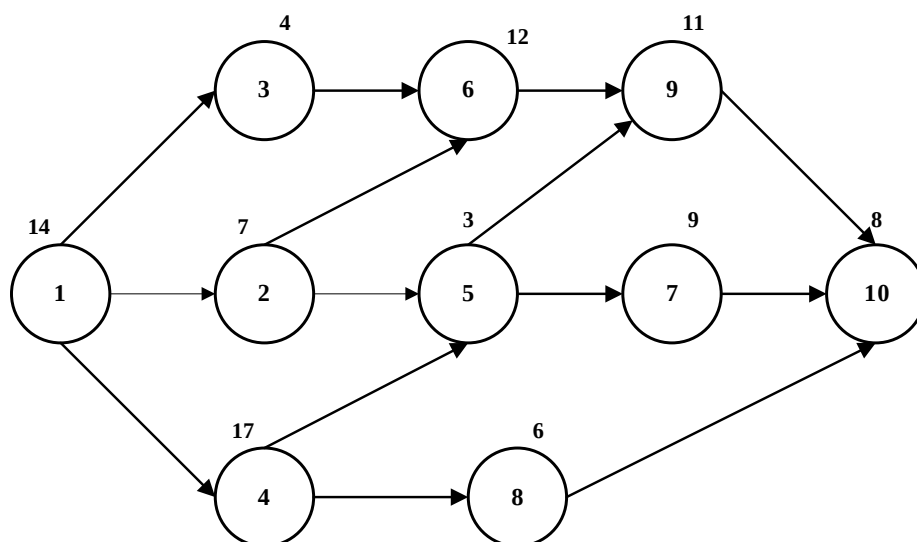


图 4.6 抽象化生产线示意图

根据上述的方法，进行随机抽取工序按照生产的先后顺序组成整条生产线的一组染色体。因为有 10 道工序，那么按照要求排列组合之后形成的染色体是[1,3,2,4,6,5,9,7,8,10]。也就是说，1 号工序是第一次被分配的，3 号工序是第 2 次被分配的，2 号工序是第 3 次被分配的，以此类推。直到 10 号工序最后一个被分配成功。采用这类编码方式能有效降低计算量，提高编码速度和搜索速度。

(2) 选择操作

本节采用的选择方式是轮盘赌方式，此方法容易操作，选择新的染色体的概率与其适应度成线性关系，且成正比，目的是为了减小不符合条件染色体选中的概率，使得种群的基因有更强的适应度。类似于在一个按照一定比例分配的轮盘中选择。每个轮盘的大小是一个百分比值，通过计算当前的适应度占总的适应度的比值。此方式的概率计算公式如下所示：

$$P(x_i) = \frac{f(x_i)}{\sum_{j=1}^m f(x_j)} \quad (4.3)$$

(3) 适应度函数选择

适应度函数需要根据整体流程来选择，此函数是用来衡量每个群体中个体基因是否良好的重要依据。种群通过适应度函数会改变种群的整体结构。一般来说，适应度函数均为非负，再根据目标函数要求，若目标函数求最大值，那么适应度函数值越大，适应能力越强。反之亦然。

现考虑到优化第二类生产线平衡问题，其目标是最小化节拍和平滑指数，所以适应度函数需要根据这个需求进行设置。适应函数如式(4.4)所示，第一部分是生产线平衡的公式，因为该公式是用来评价生产线效率，同时还有一部分是各工序之间的方差的倒数，用来评价各工序之间的均衡程度。

$$Fit = \frac{\sum_{i=1}^n t_i}{s \times \max(ST_i)} + \frac{1}{\sqrt{\sum_{i=1}^s (c - T_i)^2 + 0.01}} \quad (4.4)$$

上式中 t_i 为工序时间，表示完成一道工序所用的时间；

上式中分母加 0.01 是为了使后面部分有意义，保证分母不为 0。同时可以限制后面部分的最大值。

4.3.3 交叉、变异操作实现

(1) 交叉操作

交叉方式有多种选择，一般来说，单点交叉应用于二进制编码中比较多，相对于这种根据优先级编码的情况，选择采用基于子路径交叉的交叉方式来进行操作。

具体实现思想是：在共有 n 个基因的染色体中，随机选取从第 1 个到第 $n-1$ 个点位中任意一点作为一个交叉点，对其进行标记，记为 *Junction1*，再对 *Junction1* 之后的基因 [*Junction1*, n] 随机生成另一个交叉点，记为 *Junction2*。所以至此，整个染色体被分为 3 部分，交叉操作是针对中间部分，图 4.7 是针对图 4.6 中编码的例子展示子路径交叉的具体步骤：

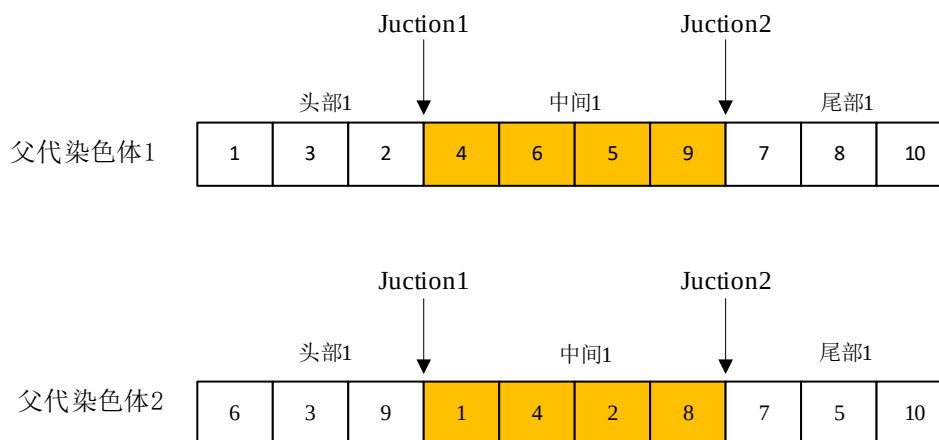


图 4.7 染色体交叉点选择示意图

在父代染色体 1 中根据随机交叉点产生的中间部分按照顺序是 4, 6, 5, 9, 那么需要到父代染色体 2 中找这几个基因, 根据搜寻, 在父代染色体找到是上述相关的基因, 在父代染色体 2 中发现顺序是 6, 9, 4, 5。那么在子代染色体 1 中, 将 6, 9, 4, 5 作为中间部分。同理, 按照相同的方式, 可以得出子代染色体 2 中的中间部分基因是 1, 2, 4, 8。如图 4.8 所示:

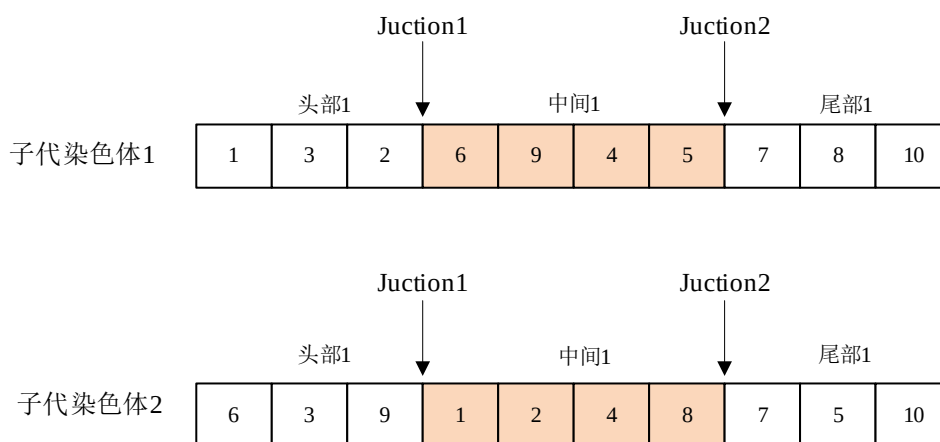


图 4.8 交叉操作之后的子代染色体

(2) 变异操作

在遗传过程中, 变异是整个种群生存进化的关键点之一。通过变异, 改变种群基因多样性, 同时获得更多的超平面以此来增大搜索范围, 避免总是在部分解中循环遗传, 获得局部解的情况。

在变异的过程中, 不会对所有染色体的所有基因进行变异操作, 会保留部分的基因完整地遗传到子代中, 但对某些染色体的有些基因按照一定规则进行变异。主要思想是: 选择一个变异点, 对变异点之前的基因不做任何修改, 依然遗传到子代染色体中, 对变异点之后基因进行变异, 该变异是按照一定规则的重组。这里的一定规则指的是, 在顺序矩阵中 (所谓的顺序矩阵指的是为了便于计算机对有向图进行处理, 可以通过转化为矩阵进行处理。称为顺序矩阵, 该矩阵通过 0, 1 数值来表示工序之间的先后顺序)。将变异点之前的基因根据行和列的关系全部清除, 基于种群初始化的过程, 通过有向图的拓扑排序方法构造出一个可行解中的基因片段。然后将这个构建的可行方案与未突变的基因片段相结合, 生成一个新的染色体。当然, 有多种方法可以来构造这个可行解, 只需选择一个符合要求的即可。图 4.9 是图 4.8 中的父代染色体 1 变异操作的具体过程:

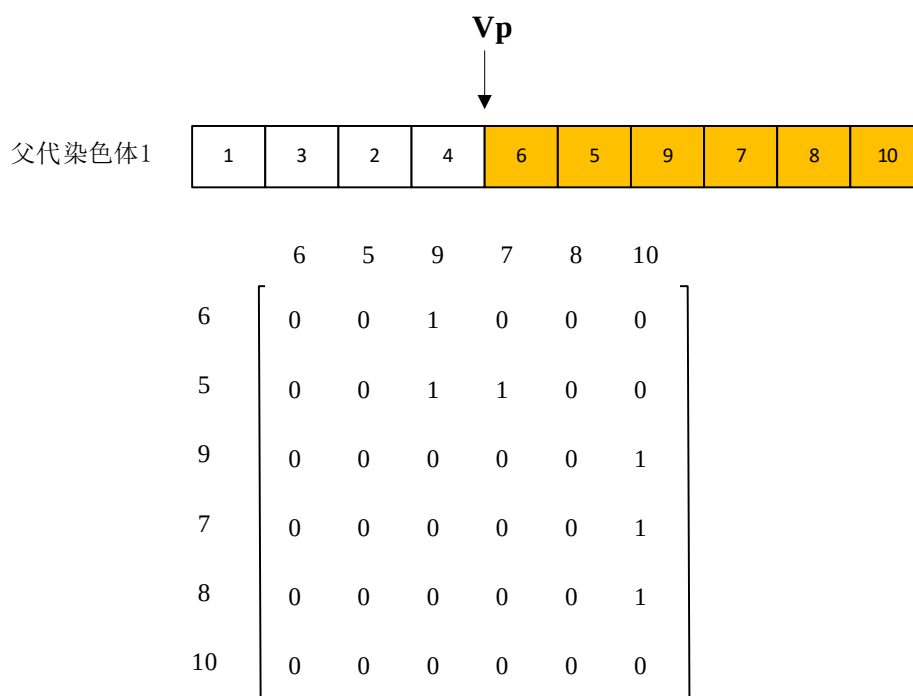


图 4.9 变异基因的顺序矩阵

根据父代染色体 1，假设从第 4 个基因点位开始变异，那么参照之前所述规则，1，3，2，4 这 4 个基因原封不动遗传至子代染色体中，重新对剩余基因按照顺序矩阵进行排列。

排列组合之后的结果如图 4.10 中的矩阵。再随机选取按照顺序可以排列到前面的基因，例如选择基因 5，若选取其他基因的话，会受到紧前工序的制约。那么此时，新的子代染色体前 5 个基因分别是 1，3，2，4，5。再将基因 6 所在的行列从顺序矩阵中删除，选取下一个变异的基因。按照此类原则，对后序基因进行重组，最终得到的子代染色体基因排列是 1，3，2，4，5，6，7，9，8，10。如图 4.10 所示：

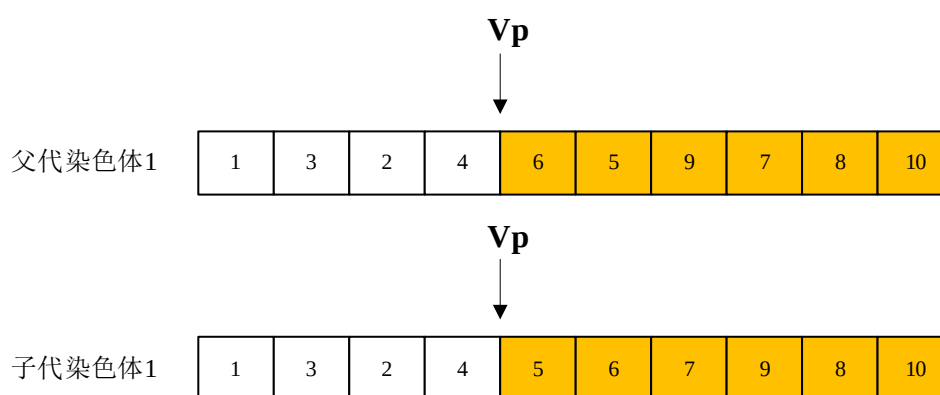


图 4.10 变异操作之后父代子代染色体对比

4.3.4 译码实现

译码操作相对于编码操作是一个逆过程。对于第一类生产线平衡问题，已知的是节拍 c ，

需要使得工作站 s 最小。这样在分配工序时会根据节拍来分配，若超过该值，就可以将工序安排到下一工序。但是第二类平衡问题的节拍 c 是不明确的，所以在刚开始译码时会有一个理论标称值。具体步骤如下：

- 1) 初始化工作站节拍 c' ：根据节拍计算公式 $c = \sum_{i=1}^n T_i / s$ ，获得 c 的值，令 $c' = c$ 。其中 c' 为理论节拍值。
- 2) 将所有工序按照作业顺序分配到各个工作站中。计算第 i 个工作站的负荷 $ST_i (i = 1, 2, \dots, s)$ 。若 ST_i 均小于等于 c' ，那么此 c' 为最小节拍值，搜索结束。否则，继续搜索。
- 3) 计算各工作站潜在增量 Δt_i ，其中 $\Delta t_j, (j=1, 2, \dots, s)$ 的值是第 $j+1$ 个工作站第一个工序的操作时间。一般采用的增量是单位时间，但这样会加大计算量。
- 4) 令 $c = \max(ST_j)$ ， $c' = \min(ST_j + \Delta t_i)$ ，若 c 小于等于 c' ，那么 c' 就是最小节拍，搜索结束。否则，回到步骤 2)。

4.4 测试和结果分析

根据之前对遗传算法的改进，现使用 MATLAB 对多种群遗传算法进行代码实现，并根据具体实例对算法有行性进行验证。

4.4.1 基于 Jackson 平衡算例问题验证

1956 年，Jackson James R. 在其论文中提出了使用枚举法解决生产线平衡问题，在论文中，Jackson 以了一个生产线模型为例作为算法的用例。这就是著名的 Jackson 生产平衡算例^[60]。因为该问题提出较早且模型简单，具有代表性的特点。在之后研究人员常使用该算例对改进算法进行验证。

Jackson 算例的先后顺序关系图如图 4.11 所示：

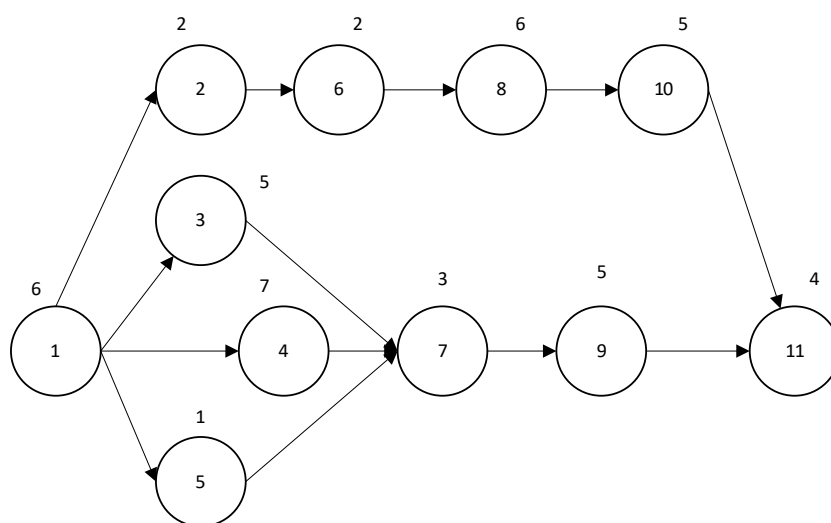


图 4.11 Jackson 模型先后顺序关系图

在上图中，可以看出共有 11 道工序，每道工序的上方数字代表的工序所用的时间。左侧的工序优先级高于右侧的工序。在 MATLAB 计算过程中，除了需要使用先后关系矩阵之外，还需要每道工序的工作时间，用来进行译码和适应度的计算。

综合之前的对先后关系矩阵的定义和时间矩阵定义，可以得到两个矩阵结果如下所示：

$$[Time] = [6 \ 2 \ 5 \ 7 \ 1 \ 2 \ 3 \ 6 \ 5 \ 5 \ 4]$$

$$[Matrix] = \begin{pmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

在 MATLAB 运行中，初始化种群时需设置种群参数，总遗传代数设置为 250 代。设置参数如表 4.2 所示：

表 4.2 各工作站运行结果

	种群 1	种群 2
数量	100	100
交叉概率	0.8	0.2
变异概率	0.2	0.05

为了更好的和 Jackson 的枚举法进行对比，设置相同的工作站数 $s = 5$ 。将以上的各个参数初始化之后，放入 MATLAB 中运行，结果如图 4.12 所示：

根据下图结果，进行分析：用柱状图形式来表示工序的先后顺序和工序所用的时间。柱状图的个数为工作站的个数，纵坐标为所用的时间。每个柱状图的左侧代表的是工序的编号，对应的是图 4.11 中的工序编码，右侧数字代表的是当前工序所用的时间。工序的执行顺序是至下往上的。例如：对于工作站 1，其工序的先后顺序是 $1 \rightarrow 2 \rightarrow 5$ ，相对应的工序 1 所用的时间是 6 秒，工序 2 所用的时间是 2 秒，工序 5 所用的时间是 1 秒。所以，工作站 1 的工序所用的总时间是 9 秒。

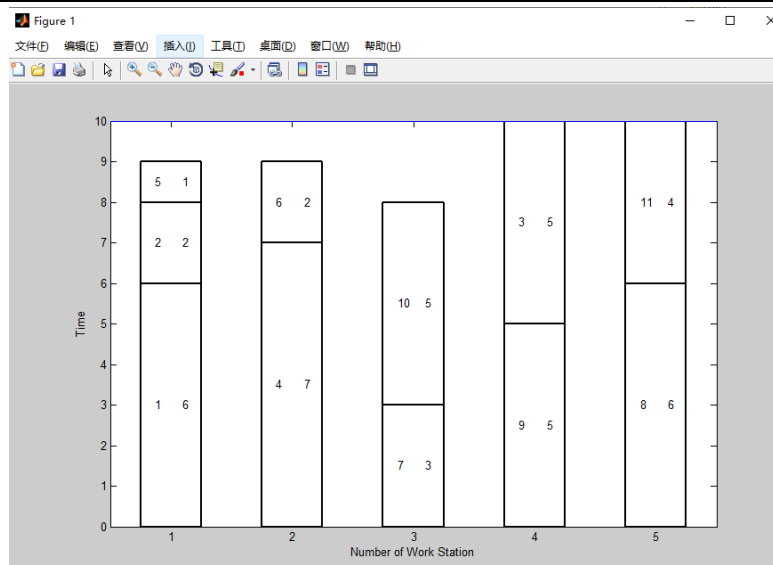


图 4.12 Jackson 改进遗传算法结果图

各个工作站所用的总时间分别是：9，9，8，10，10。生产节拍 $c = 10$ ，根据式（4.1）和式（4.2）可分别得出平衡率和均衡指数：

$$\text{平衡率: } \partial = \frac{\sum_{i=1}^n t_i}{n \times \max(ST_i)} \times 100\% = \frac{9+8+10+10+9}{5 \times 10} = 92\%$$

$$\text{均衡指数: } SI = \sqrt{\frac{\sum_{i=1}^j (\max ST_i - ST_i)^2}{j}} = \sqrt{\frac{(10-9)^2 + (10-8)^2 + (10-10)^2 + \dots + (10-10)^2}{5}} = 1.09$$

根据 Jackson 使用枚举法得到的结果，当工作站数为 5 的时候，各个工作站所用的总时间分别是 10，7，10，10，9。生产节拍 $c = 10$ 根据式（4.1）和式（4.2）可分别得出平衡率和均衡指数：

平衡率 $\partial = 92\%$ ，均衡指数 $SI = 3.16$ 。

改进遗传算法和 Jackson 的枚举法各个参数之间对比见下表：

表 4.3 Jackson 算例两种算法对比

算法名称	工作站工序分配					工作站 个数	生产 节拍	平衡率	均衡指 数
	工作站1	工作站2	工作站3	工作站 4	工作站 5				
枚举算法	1, 2, 6	4	5, 7, 8	3, 10	9, 11	5	10	92%	3.16
改进遗传算法	1, 2, 5	4, 6	7, 10	3, 9	8, 11	5	10	92%	1.09

从上表可以得知，在工作站工序分配上，两种算法结果是不相同的，但是整个生产线的总时间是相同的，所以根据式（4.1）得出的平衡率是相同的。虽然平衡率相同，但是改进遗传算法的均衡指数为 1.09，远小于 Jackson 的枚举法的 3.16，说明改进的遗传算法的各个工作站之间更加平衡，没有差异特别巨大的工作站。说明改进的遗传算法在各个工作站负荷平衡上要优于 Jackson 的枚举法。

4.4.2 缝纫机生产线平衡改进遗传算法优化结果

在缝纫机实际的工作生产过程中，通常是对面料的裁剪，缝制，钉扣等过程。裁剪和缝制又分为不同针距的缝制。在某公司缝纫机生产衬衫流程一共 20 道工序，包括了布料裁剪，上下线缝制，锁眼定位等过程。原工作站个数是 6 个，现通过改进的遗传算法对此生产线进行优化。

实际生产线测试环境如图 4.13 所示：



图 4.13 衬衫生产线实际环境

衬衫生产线先后顺序关系图如图 4.14 所示，各工序所需的工作时间如表 4.4 所示。

表 4.4 各工序所需工作时间

工序	时间(min)	工序	时间(min)	工序	时间(min)	工序	时间(min)
1	9.5	6	6	11	4	16	8
2	15	7	10	12	8.5	17	17
3	10.5	8	16	13	13.5	18	13
4	8	9	12	14	16	19	6.5
5	5.5	10	13	15	20	20	9

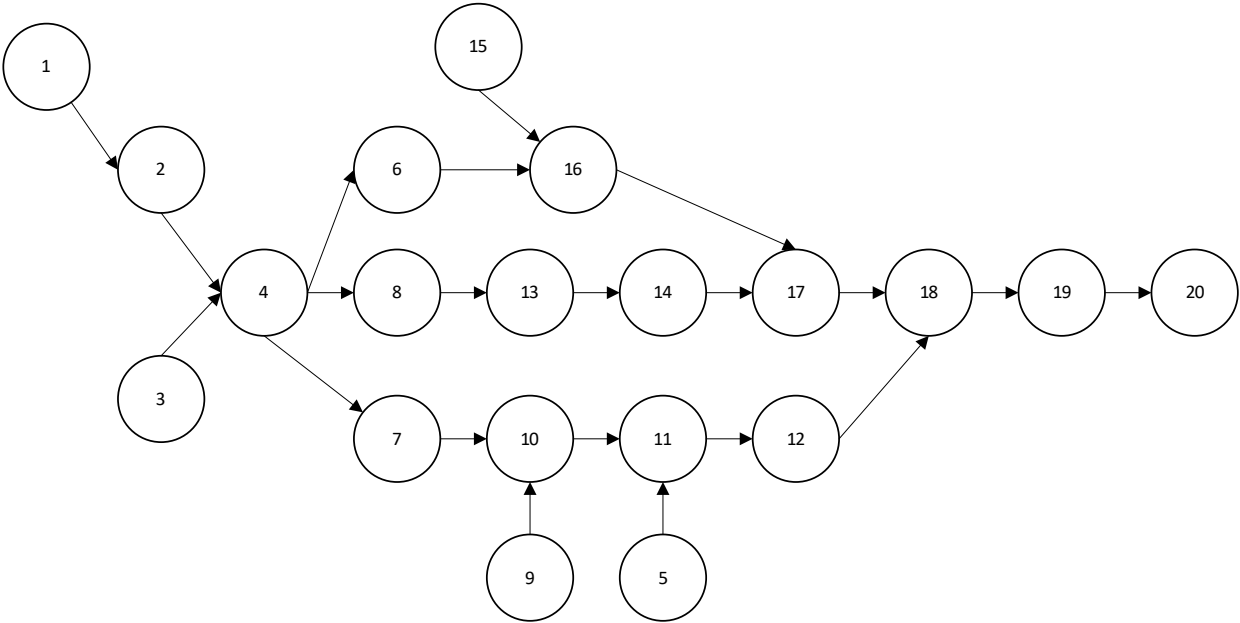


图 4.14 衬衫生产线先后顺序图

现运行的生产线一共 6 个工作站，各个工作站分配工序及时间如下表所示：

表 4.5 各工作站对应的工序

工作站	工序	时间（min）
1	1, 2, 3	35
2	4, 6, 9, 7	36
3	8, 13, 14	45.5
4	10, 5, 11, 15	42.5
5	12, 16, 17	33.5
6	18, 19, 20	28.5

从上面可以看出，各工作站之间非常不平衡，不同工作站工作的时间很不均衡，各工作站会出现互相等待产品的结果，导致生产效率低下。现根据衬衫生产线先后顺序图及表 4.5 得出如下矩阵：

Time = [9.5 15 10.5 8 5.5 6 10 16 12 13 4 8.5 13.5 16 20 8 17 13 6.5 9];

现对不同工作站个数进行分类讨论，计算平衡率和均衡指数。根据综合结果，选择一个适宜的工作站个数。

通过 MATLAB 软件进行仿真，工作站个数分别是 4，5，6，7 个。仿真结果如下图 4.15 所示，其中（a）为 4 个工作站的仿真情况，（b）为 5 个工作站的仿真情况，（c）为 6 个工作站的仿真情况，（d）为 7 个工作站的仿真情况。

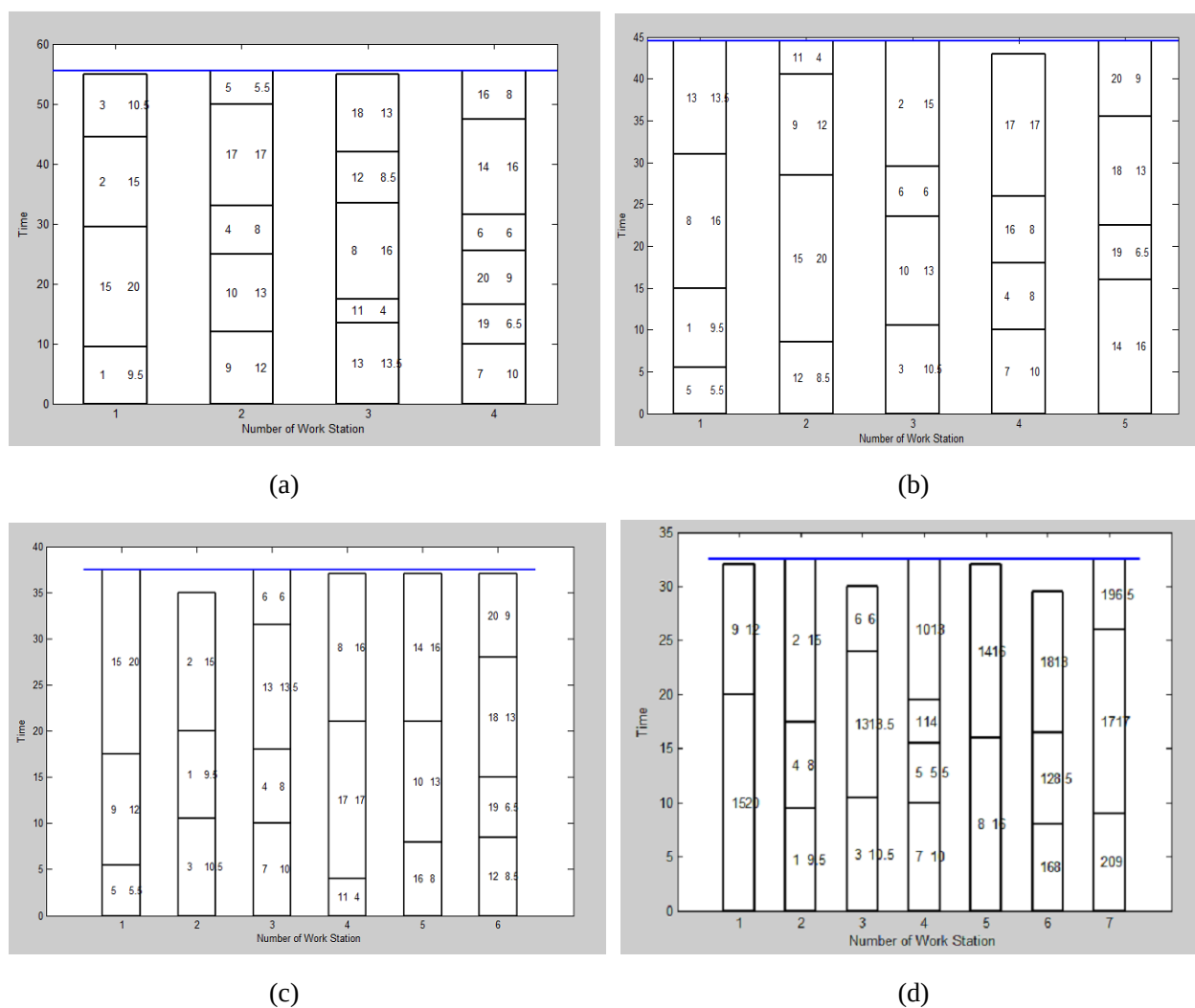


图 4.15 不同工作站仿真结果

当工作站个数为 4 时，各个工作站的负荷[55,55.5,54,54.5], 生产节拍 c 为 55.5。根据式 (4.1) 和式 (4.2) 可分别得出平衡率和均衡指数。

$$\rho = \frac{\sum_{i=1}^n t_i}{n \times \max(ST_i)} \times 100\% = \frac{55+55.5+54+54.5}{4 \times 55.5} \times 100\% = 98.6\%$$

$$SI = \sqrt{\frac{\sum_{i=1}^j (\max ST_i - ST_i)^2}{j}} = \sqrt{\frac{(55.5-55)^2 + (55.5-55.5)^2 \cdots + (55.5-54.5)^2}{4}} = 0.94$$

当工作站个数为 5 时，各个工作站的负荷[44.5,44.5,44.5,43,44.5], 生产节拍 c 为 44.5。根据式 (4.1) 和式 (4.2) 可分别得出平衡率和均衡指数。

$$\partial = \frac{\sum_{i=1}^n t_i}{n \times \max(ST_i)} \times 100\% = \frac{44.5+44.5+44.5+43+44.5}{5 \times 44.5} \times 100\% = 99.3\%$$

$$SI = \sqrt{\frac{\sum_{i=1}^j (\max ST_i - ST_i)^2}{j}} = \sqrt{\frac{(44.5-44.5)^2 + (44.5-44.5)^2 \cdots + (44.5-43)^2}{5}} = 0.67$$

当工作站个数为 6 时，各个工作站的负荷[37.5,35,37.5,37,37,37], 生产节拍 c 为 37.5。根据式 (4.1) 和式 (4.2) 可分别得出平衡率和均衡指数。

$$\partial = \frac{\sum_{i=1}^n t_i}{n \times \max(ST_i)} \times 100\% = \frac{37.5+35+37.5+37+37+37}{6 \times 37.5} \times 100\% = 98.2\%$$

$$SI = \sqrt{\frac{\sum_{i=1}^j (\max ST_i - ST_i)^2}{j}} = \sqrt{\frac{(37.5-37.5)^2 + (37.5-35)^2 \cdots + (37.5-37)^2}{6}} = 1.08$$

当工作站个数为 7 时，各个工作站的负荷[32,31.5,31,31,30,33,32.5], 生产节拍 c 为 33。根据式 (4.1) 和式 (4.2) 可分别得出平衡率和均衡指数。

$$\partial = \frac{\sum_{i=1}^n t_i}{n \times \max(ST_i)} \times 100\% = \frac{32+31.5+31+31+30+33+32.5}{7 \times 33} \times 100\% = 95.7\%$$

$$SI = \sqrt{\frac{\sum_{i=1}^j (\max ST_i - ST_i)^2}{j}} = \sqrt{\frac{(33-32)^2 + (33-31.5)^2 \cdots + (33-32.5)^2}{7}} = 1.72$$

4.4.3 优化结果分析

4.4.3.1 确定工作站个数

在此次优化过程中，根据表 4.2 所设置的初始化参数对多种工作站个数不同的情况进行

模型，运行结果如下表：

表 4.6 各工作站运行结果

仿真序号	程序参数设定									运行结果		
	工作站个数	种群1规模	种群2规模	种群1交叉概率	种群2交叉概率	种群1变异概率	种群2变异概率	繁殖代数	交换基因数目	生产节拍	平衡率	均衡指数
1	4	100	100	0.8	0.2	0.2	0.05	250	5	55.5	98.6%	0.94
2	5									44.5	99.3%	0.67
3	6									37.5	98.2%	1.08
4	7									33	95.7%	1.72

通过上表可知，通过改进的遗传算法可以有效的优化生产节拍，提高平衡率，降低均衡指数。在上述仿真过程中，平衡率均高于 95%，均衡指数均低于 2。这表明改进的遗传算法起到很好的效果。

在各个工作站之间进行横向对比，在初始参数设定相同情况下，对比结果如图 4.16 所示：

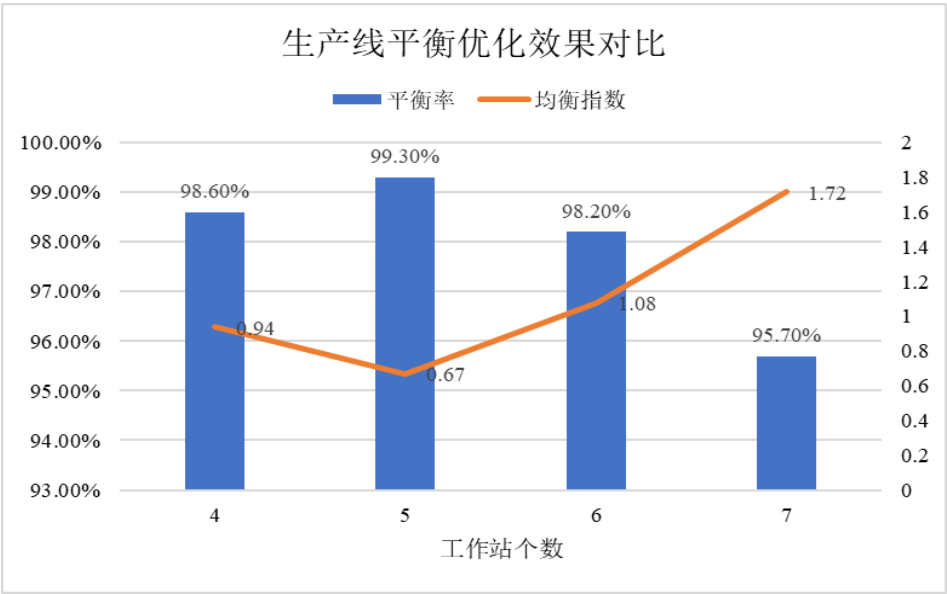


图 4.16 生产线平衡优化效果对比

从上述表 4.6 可得知，当工作站数目 $s = 4$ 时，平衡率为 98.6%，工作站数目 $s = 5$ 时，平衡率高达 99.3%，工作站数目 $s = 6$ 时，平衡率下降，为 98.2%，之后随着工作站数目增加，平衡率下降趋势变大。从上述表 4.6 可知，当工作站数目 $s = 5$ 时，均衡指数最小，随着工作站数目增加，均衡指数快速增长。

从缝纫机实际生产线来看，当工作站数目 $s \leq 4$ 时，工序会比较集中于几个工作站中，每个工作站生产时间较多，导致生产效率降低，不能满足客户需求。当工作站数目 $s \geq 7$ 时，平衡率数值明显小于工作站数目少的时候，均衡指数急剧上升，导致各工序之间不均衡，工人效率下降；且工作站数量过多，导致生产线长度大于设定值，不便于管理，提升了生产成本。

综合以上各因素的考虑，选取工作站数量为 5，不仅小于原先的工作站个数，降低了成本，既能极大地提高生产效率，又能使各工序之间平衡且有序。

4.4.3.2 确定工作站结果分析

根据确定好的工作站数量 5，各工作站工序分配如下：

工作站 1 = [5,1,8,13]

工作站 2 = [12,15,9,11]

工作站 3 = [2,3,10,6]

工作站 4 = [4,7,16,17]

工作站 5 = [14,18,19,20]

结合现实缝纫机生产线，如本小节刚开始设置的工序顺序在实际中实现有一定困难，如在现实操作中，卷下摆和下摆加固需要在一个工作站中完成，目的是为了避免犯错。又如锁眼和钉扣工序，将这两个工序安排在前后顺序，会更便于工人进行操作，提升工人的工作效率。

根据此原则，现对某些工序进行人工手动调整：

工作站 1 = [1,2,15]

工作站 2 = [3,4,5,12,9]

工作站 3 = [6,11,14,17]

工作站 4 = [7,10,16,13]

工作站 5 = [8,18,19,20]

根据式（4.1）和式（4.2）可分别得出平衡率为 98.1%和均衡指数为 1.12, 和人工调整之前无很大差异，且符合客户需求。

4.4.4 整体结果分析

在未进行优化分析之前，原生产线共有工作站 6 个，各工作站负荷如表 4.5 所示。根据式（4.1）和式（4.2）可分别得出平衡率为 80.95%和均衡指数为 25.4。通过传统工业工程优化之后和改进遗传算法对其进行优化，并通过人工手动对生产线进行优化，将它们优化结果进行比较展示，结果如图 4.17 所示：

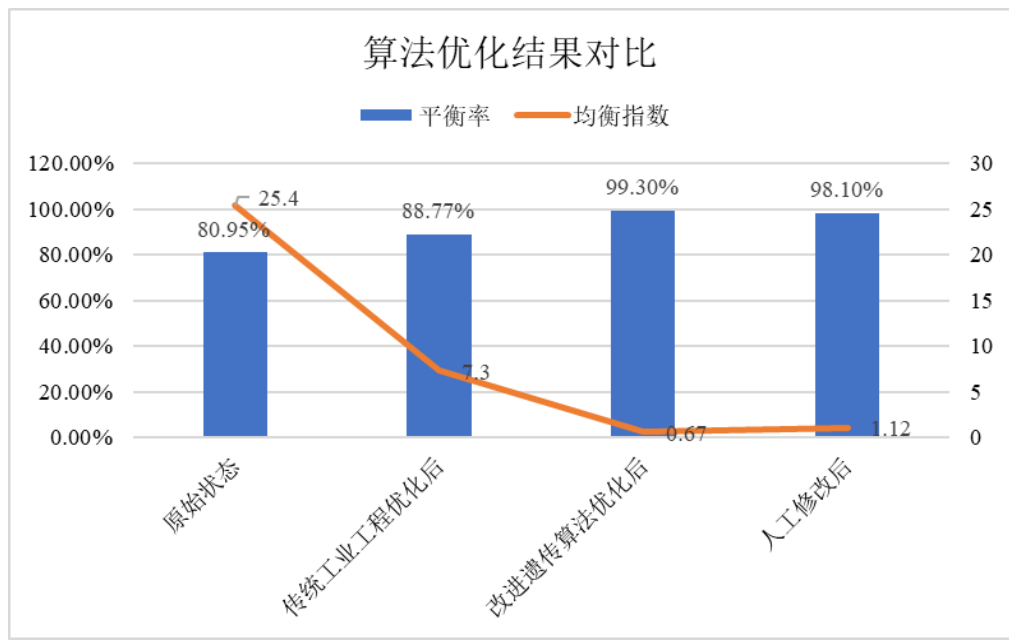


图 4.17 优化结果对比

表 4.7 改进前后误差对比

	原始状态	改进遗传算法	差值
平衡率	80.95%	99.3%	18.35%
均衡指数	25.4	0.67	-24.73

从上图可以看出，未进行改进之前生产线非常不平衡，且生产效率低下。通过传统的遗传算法对其进行优化，有了显著的提升，生产效率提高了 10%，但是均衡指数仍有很大的改善空间。使用改进的遗传算法后，将平衡率提高到 99.3%，且均衡指数可以降到 0.67，使得各工作站负荷基本一致。结合实际生产工序的情况，手动调整工序结构，使得生产线更加平稳。

对不同工作站个数通过 MATLAB 运行改进的遗传算法，综合各种因素得出工作站数为 5 时会产生较好效果，优于现有运行的生产线。

4.5 本章小结

本章对生产线平衡概念通过数学模型进行介绍，同时对基本遗传算法步骤做了描述。并设计了一种改进的遗传算法，对多种群的遗传算法设计和一些重要操作进行详细的分析。使用 MATLAB 对该算法进行测试，采用经典的 Jackson 算例验证程序的正确性，并对缝纫机实际生产线进行模拟求解，得出不同工作站数量下的平衡率和均衡指数。最后，通过综合对比，得出最优的工作站数量且进行人工调整，和原先生产线相比，平衡率提高了近 20%，符合工厂提出优化的需求。

第五章 总结与展望

5.1 总结

针对当前传统的缝纫机工厂存在的数字化程度不高，对缝纫机工作状态不能实时获取且操作的问题。本文基于无线通信技术研发了一套智能缝纫机在线监测系统，通过了解工厂目前存在的痛点和需求，根据需求，设计了包括云和端的整体框架，物联网网关与缝纫机控制器通过工业总线进行连接，实时获取缝纫机工作时的各类参数和工作状态，通过 Wi-Fi 无线通信技术与设备云进行交互，将数据发送至云端，并通过 Web 页面进行展示，给用户直观的体验，形成一整套完善的系统。

以监测系统总体为基础，将平台收到的数据，进行筛选分析，对有价值数据进行处理，并对缝纫机故障检测和生产线平衡问题进行深入研究，为缝纫机工厂转型起到关键作用。具体关键技术包括 EEMD 分解算法和改进粒子群的支持向量机方案，基于改进遗传算法对生产线平衡的优化。

本文设计的系统符合市场需求，极大的提高了生产效率和缩减工厂成本。其中主要工作和技术包括：

1) 对当前缝纫机工厂的需要解决的问题进行分析：无法通过可视化页面掌握缝纫机实时的工作状态，对获取的数据不能保证能长时间存储，不能长时间连接或者修改设备的参数信息，对当前效率不高生产线无法做到有效性的修改意见。

2) 针对无法通过可视化页面掌握缝纫机实时的工作状态的问题，本文设计一套结合工业总线和无线通信技术，将缝纫机设备入网，获取生产状态，并将数据存在云端，可以保证长时间存储。在某些情况下，可以对缝纫机设备进行远程控制，保证系统整体的安全运行。

3) 针对当前缝纫机故障检测不准确，本文设计一种基于 EEMD 和改进粒子群算法的 SVM 故障检测方案。首先将采集到的轴承信号通过改进 EEMD 算法分解，再通过对比能量熵和峭度在不同分量中的聚类效果，选择前者作为分类器的输入向量，通过改进粒子群算法来优化参数 c 和 g 。经过测试对比表明，可以很大的提升故障检测准确率。

4) 针对当前缝纫机工厂由于生产线工序组合问题引起的生产效率低的问题。本文根据对实际生产线分析，并就传统的遗传算法的阐述分析，根据其容易在相似的种群中不断寻找解，不容易跳出的情况，容易陷入局部最优。以此设计了多种群遗传算法，将两个种群独立遗传，之后进行交叉繁衍获取更优个体，并使用经典算例进行验证，结合实际生产线工序进行改进，大大提高了生产线的平衡率和均衡率。提升了工厂生产的效率。

5.2 展望

本文开发的缝纫机在线监测系统在功能和使用上基本满足了设计的需求，并很大程度降低了成本，但由于整体系统过于复杂且庞大，再加上个人经验不足，整个系统仍存在一些不足：

1) 因缝纫机设备参数过多，可使用边缘计算能技术对数据进行处理，更好的为平台侧分担压力，提高系统整体效率，尽可能实现本地化服务；并可以结合工厂的订单，做好工单的分配处理及售后服务。

2) 对于故障检测的分类模型，没有更多的使用神经网络模型对其进行仿真分析。且可以通过对缝纫机故障其他信号进行提取或许能更大程度上提供检测效率；对于信号特征的选择上，可以同时选取多种特征作为模型的输入参数；

3) 本文仅考虑工厂生产方法，并未将生产过程中各个工作站布局纳入考虑范围，缺少思考其对整体生产效率的影响。且只考虑生产某种产品的情况，若要更好的应用到实际中，需要考虑大规模混合生产的情况。

参考文献

- [1] 胡睿. 缝纫机的现状与发展趋势[J]. 商业文化(上半月), 2011(09): 316-317.
- [2] 中国十四个五年规划和 2035 年远景目标纲要[J]. 2021.
- [3] 谭新民. 缝纫机的计时方法/系统、存储介质、缝纫机及计时网络[P]. 杰克缝纫机股份有限公司, 2021.
- [4] 王永忠, 王妍萍. 服装设备自动化的现状和发展[J]. 电子科技, 2013, 26(10): 169-172.
- [5] 陈毅钊. 国内外缝纫线的发展现状及对扩大我国产品出口的探讨[J]. 缝纫机科技, 1992(4): 17-19.
- [6] 卿浩. 缝纫机用无刷直流电机控制器研究[D]. 武汉: 华中科技大学, 2006.
- [7] 吴德文, 张团善. 基于 Qt 的模板缝纫机远程监控系统设计[J]. 科技视界, 2020, (19): 264-265.
- [8] 刘紫健. 基于直流无刷电机的工业缝纫机控制系统的研制[D]. 广州: 华南理工大学, 2012.
- [9] 罗立峰. 全自动缝纫机控制系统的设计与开发[D]. 杭州: 浙江工业大学, 2014.
- [10] 赵新潮. 基于模糊控制的智能缝纫机器人[D]. 天津: 河北工业大学, 2017.
- [11] Olga Kadnikova, Gulnara Altynbayeva, Potential analysis of implementation of developed technology for processing of sewing and knitting fabrics[J]. Energy Procedia, 2017, 128: 411-417.
- [12] Adel A. El-samahy, Mohamed A. Shamseldin, Brushless DC motor tracking control using self-tuning fuzzy PID control and model reference adaptive control[J], Ain Shams Engineering, 2018, 9(3): 341-352.
- [13] Carvalho.H, Soares F, Silva, Guhr.F, Adaptive control of an electromagnetically actuated presser-foot for industrial sewing machines[P]. Emerging Technologies and Factory Automation, 2010. IEEE Conference on 2010.
- [14] V. K. Rai, A. R. Mohanty., Bearing fault diagnosis using FFT of intrinsic mode functions in Hilbert-Huang transform [J]. Mechanical Systems & Signal Processing, 2006, 21(6): 2607-2615.
- [15] 王炳和. 现代数字信号处理[M]. 西安: 西安电子科技大学出版社, 2011.
- [16] Wei D C, Robert X. Gao, et al. The Error Caused by Order Analysis for Rolling Bearing Fault Signal[J] Advanced Materials Research, 2013, 2717(819-819): 254-258.
- [17] H. Yetis, M. Karakose, I. Aydin, E. Akin, Bearing Fault Diagnosis in Traction Motor Using

- the Features Extracted from Filtered Signals[C], 2019 International Artificial Intelligence and Data Processing Symposium (IDAP), 2019: 1-4.
- [18] Li. S L, Liu. Z L. Application of improved wavelet packet energy entropy and GA-SVM in rolling bearing fault diagnosis[J]. IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC), 2018: 1-5,
- [19] 李海涛, 何玉珠, 宋平. 基于量子万有引力搜索的 SVM 自驾故障诊断[J]. 北京航空航天大学学报, 2016, 042(006): 1093-1098.
- [20] 国红波, 马晓建, 郑子明. 基于 LabVIEW 与 EMD 的工业缝纫机故障诊断[J]. 机械工程师, 2009(2): 60-62.
- [21] Song X D, Zhu D J, Liang P, An L. A new bearing fault diagnosis method using elastic net transfer learning and LSTM[J]. Journal of Intelligent & Fuzzy Systems, 2021, 40(6).
- [22] Wang Y, Wei Z, Yang J. Feature Trend Extraction and Adaptive Density Peaks Search for Intelligent Fault Diagnosis of Machines[J]. IEEE Transactions on Industrial Informatics, 2019, 15(1): 105-115.
- [23] 李保东. 装配生产线平衡分析与应用研究——以 Z(重庆宗申动力机械)公司为例[D], 重庆: 重庆交通大学, 2014.
- [24] 王公臻. 大规模生产线平衡问题的模型及求解研究[D]. 北京: 北京交通大学, 2018.
- [25] 殷旅江, 何波, 杨立君. 多类指派约束下汽车总装装配线平衡优化[J]. 制造业自动化, 2014, 36(24): 41-44.
- [26] Bukchin Y, Raviv T. Constraint programming for solving various assembly line balancing problems[J]. Omega, 2018, 78(1): 57-68.
- [27] 张晓玲. 装配线平衡问题不同算法性能比较研究[D]. 衡阳: 南华大学, 2016.
- [28] 郑巧仙, 李明, 唐秋华. 多约束装配线平衡问题的可行解存在性分析[J]. 计算机集成制造系统. 2019, 25(3): 619-628.
- [29] 李明, 唐秋华, 郑巧仙. 第 2 类装配线平衡问题的改进规则组合算法[J]. 计算机集成制造系统. 2015, 21(1): 88-93.
- [30] 李明, 李珊, 夏绪辉. 大规模多工位装配线平衡问题的规则组合算法[J]. 计算机集成系统, 2013, 19(11): 2780-2787.
- [31] Ibrahim Kucukkoc et.al. Balancing of mixed-modele two-sided assembly line with underground workstations : A mathermathical model and ant colony optimization algotithm[J]. International Journal of Production Economics, 2018, 205: 228-243.
- [32] Ponnambalam S G, Aravindan P, Naidu G M. A comparative evaluation of assembly line balancing heuristics[J]. International Journal of Advanced Manufacturing Technology. 1999, 15(8): 577-586.

- [33] 高书冰. 带有并行机的多行设备布局问题及其超启发式算法[D]. 北京: 北京邮电大学, 2018.
- [34] 毛凌翔, 郑永前. 蚁群算法求解装配线平衡第一类问题[J]. 计算机系统应用, 2010, 19(1): 140-143.
- [35] 鲁建厦, 蒋玲玲, 李修琳. 基于混合粒子群算法求解装配线第二类平衡问题[J]. 中国机械工程, 2010, 21(4): 420-424.
- [36] 张晨威. 某装配车间生产线平衡问题研究[D]. 沈阳: 沈阳工业大学, 2017.
- [37] 王军宇. 基于 ZigBee 的智能停车场管理系统的研究[D]. 长春: 吉林建筑大学, 2017.
- [38] 汤明, 郑婧, 黄文婷, 等. 基于 ZigBee 的无线振动传感器设计与实现[J]. 传感技术学报, 2018, 31(2): 312-318.
- [39] 徐文. 基于 WiFi 与 Android 的智能家居监控系统设计[D]. 成都: 西南交通大学, 2017.
- [40] 鲁勇, 吕绍和. 基于 WiFi 信号的人体行为感知技术研究综述[J]. 计算机学报, 2018:1-22.
- [41] 穆志洋. 基于 NB-IOT 的城市声光污染监测系统研究[D]. 杭州: 浙江大学, 2018.
- [42] 祝恩国, 邹和平, 巫钟兴. 一种应用 NB-IOT 的智能电能计量装置设计[J]. 电气传动, 2019, 49(3): 92-96.
- [43] 熊一鹏, 岳伟. FreeRTOS 多任务调度机制在监控单元中的应用[J]. 单片机与嵌入式系统应用, 2021, 21(9): 64-66.
- [44] 王计波. 面向智能断路器的多协议通信技术研究[D]. 天津: 河北工业大学, 2013.
- [45] 吴晨红. 基于 Modbus 通信协议的信号采集系统[D]. 合肥: 合肥工业大学, 2021.
- [46] 张英. 基于 MQTT 协议的数据安全过滤组件的研究与实现[D]. 北京: 华北电力大学, 2021.
- [47] 徐侃, 丁强. 一种基于 MQTT 协议的物联网通信网关[J]. 仪表技术, 2019(1): 1-4.
- [48] 国红波. 工业缝纫机动态特性测试系统的研究[D]. 上海: 东华大学, 2009.
- [49] 张荣川, 郭文飞. 基于 CEEMD-LSSVM 的缝纫机轴承故障诊断[J]. 2020, 07(6): 81-86.
- [50] 朱兴统. 旋转机械智能故障诊断方法的研究[D]. 广州: 广东工业大学, 2020.
- [51] Kennedy J, Eberhart R. Particle swarm optimization[C]. Proceedings of International Conference on Neural Networks. Piscataway: IEEE, 1995: 1943-1948.
- [52] Shi Y, Eberhart R. A Modified Particle Swarm Optimizer[C]. IEEE World Congress On Computation Intelligence, Anchor- age, AK, USA, 1998: 69-73.
- [53] Al-Hawari T, Alim M, Al-Araidah O, et al. Development of a genetic algorithm for multi-objective assembly line balancing using multiple assignment approach[J]. The International Journal of Advanced Manufacturing Technology, 2015, 77(5-8): 1419-1432.
- [54] 杨晓华, 陆桂华, 杨志峰, 等. 格雷巧加速遗传算法及其理论研究[J]. 系统工程理论与实践, 2003, 23(3): 100-106.

- [55] Qian J, Tao F, Yan D, Xin Z and Wang J Y. Servo Control of Fast Steering Mirror Based on Improved Genetic Algorithm[C]. Annual International Conference on Data Science and Business Analytics (ICDSBA), 2020: 134-138.
- [56] Holland J. Adaptation in natural and artificial systems[M]. Bradford: A Bradford Book 1975.
- [57] Hou K, Ren W, Chang S. Liu Q, Design of orthogonal discrete frequency coding waveforms via improved genetic algorithm[C]. IET International Radar Conference (IET IRC 2020), 2020. 868-873.
- [58] Qi X, Khattak S, Zaib A and Khan I. Energy Efficient Resource Allocation for 5G Heterogeneous Networks Using Genetic Algorithm[J]. IEEE Access, 2021,9: 160510-160520.
- [59] Zhang L, Zhang L, Yang J, Gao M and Li Y. Application Research of Fuzzy PID Control Optimized by Genetic Algorithm in Medium and Low Speed Maglev Train Charger[J]. IEEE Access, 2021,9: 152131-152139.
- [60] Jackson, J.R. Acomputing procedure for a line balancing problem. Management Science[J].1956, 2: 261-271.

致 谢

未觉池塘春草梦，阶前梧叶已秋声。研究生求学生涯，转眼间就迎来了句号。在杭州电子科技大学电子信息学院的每一次学习都是实实在在的经历，都是最珍贵的记忆。过去的无法挽回，感动之余，我也有很多感谢的话想对研究生期间遇到的老师、同学和朋友们诉说：

首先，感谢我的研究生导师陈科明副教授对我的谆谆教诲。感谢陈老师对我科研上的严格要求，为我们提供良好的科研环境，让我学到了知识，也获得了实践锻炼的机会；感谢陈老师生活上的平易近人，对我们的学习生活也是关怀的无微不至；感谢陈老师对我视如己出的教导，用老师亲身经历为我未来的发展规划分享宝贵经验……正是陈老师在我读研生活中亦师亦友的教诲，帮助我培养起高水平的为人处事能力。在此谨向我最敬爱的陈老师表达我最诚挚的敬意和衷心的感谢。

感谢我的同门张百军、刘剑、徐志杰、胡晓双、蔡坤、黄盼盼、小邢同学，他们是我学习、工作和生活上的伙伴，也是我面对困难和挑战时的战友；感谢我的师兄吴耀增，给与我无私的帮助和鼓励，让我学到很多；也感谢师弟师妹，为我提供的帮助。

感谢我的舍友赵鑫、黄盛淼，正是你们平时对我生活上的帮助和照顾，才能让我有更多精力投入的科研学习中。

感谢我的父母及家人多年来对我的一贯关心、爱护和帮助，无论任何时候都没有给我压力，给了我一个轻松的求学环境。正是你们的无私奉献，让我更加充满希望、没有后顾之忧地去热爱生活、去奋斗拼搏。他们的支持、理解和鼓励是我努力完成学业的精神支柱。

最后，感谢论文评阅老师和答辩委员会的各位老师！感谢培养教育我的杭州电子科技大学，电子信息学院浓厚的学术氛围，舒适的学习环境我将终身难忘！

附 录

作者在读期间发表的学术论文及参加的科研项目

参加的科研项目：

[1] 中捷缝纫机工业互联网平台开发，横向课题(KYH043121031M) .2021.4-2022.3

发表的专利：

- [1] 陈科明, 吴文楷, 张百军, 吴耀增. 一种物联网数据监控点和寄存器地址自适应映射方法 [P] 浙江省: CN114281408A, 2022-04-05, (已公开, 2/4, 导师第一发明人).
- [2] 陈科明, 吴文楷, 庞若雷, 齐玲玲. 基于EEMD和改进PSO-SVM的轴承故障诊断方法[P] 浙江省: 202210494296.1, (已申请, 2/4, 导师第一发明人).
- [3] 陈科明, 张百军, 吴文楷, 刘剑. 一种六轴优化的磁力计在线校准方法[P] 浙江省: CN114279426A, 2022-04-05, (已公开, 3/4, 导师第一发明人).