

分类号: \_\_\_\_\_

密 级: \_\_\_\_\_

UDC: \_\_\_\_\_

单位代码: \_\_\_\_\_

安徽工业大学

# 硕士学位论文

论文题目: 面向订单的冷轧薄板生产计划方法及其应用研究



学 号: 1420190398

作 者: 茹 伟

专 业 名 称: 工业工程

2016 年 05 月 29 日



# 安徽工业大学硕士学位论文

论文题目: 面向订单的冷轧薄板生产计划方法及其应用研究

Research and Application of Production Planning Method for  
MTO-based Cold-rolled Sheet Enterprises

作 者: 茹 伟 学院: 管理科学与工程学院

指 导 教 师: 潘瑞林 单位: 安徽工业大学

协助指导教师: \_\_\_\_\_ 单位: \_\_\_\_\_

\_\_\_\_\_ 单位: \_\_\_\_\_

论文提交日期: 2016 年 05 月 29 日

学位授予单位: 安 徽 工 业 大 学

安徽马鞍山 243002



## 摘要

本文综述了冷轧企业组批与生产计划的研究现状，总结了钢铁企业冷轧生产系统的特点，根据我国冷轧企业的实际生产情况，以国内某钢铁企业的冷轧生产线为研究对象，研究了冷轧企业的合同组批优化与生产计划问题。主要研究内容如下：

(1) 针对冷轧企业大批量生产模式和多品种、小批量的市场需求之间存在的矛盾，本文建立了以合同交货期差异度、工艺路线差异度和调整次数最小化为目标，同时满足批次重量、出(入)口宽度、出(入)口厚度、抗拉强度等工艺约束的冷轧组批计划模型，并构建了基于粒子群优化的模糊聚类算法进行求解。

(2) 针对冷轧生产流程存在多阶段库存和批次多流向的特点，本文在满足工艺约束的基础上建立了以合同超期和提前、中间库存量超出和不足总惩罚最小并确保成品库成本最低的多阶段多流向冷轧生产计划模型，并利用改进的遗传算法求解。算法通过参数的合理设定，解决了交叉、变异概率在一定区间内取固定值、断点处函数值跳跃大和出现不合理值的问题，从而两种概率可根据适应值进行自适应调整。

(3) 为了使本文所提出的模型和算法得以推广和应用，本文采用了基于 J2EE 的 B/S 模式开发了冷轧薄板订单生产计划系统。该系统为冷轧薄板企业的生产提供了指导，从而提高了冷轧企业的生产管理水平。

本文所建立的组批与生产计划数学模型和相应的算法都经过了实际的生产数据进行了验证，实验表明，所提出的模型和方法都切实可行。

**关键词：**冷轧；生产计划；组批；多阶段；多流向；改进遗传算法；模糊聚类；粒子群算法

## Abstract

In this dissertation, the state of batch and production planning is reviewed and the characters of the cold rolling production system are also summarized in cold rolling enterprises. Taking a cold rolling production line of national certain Iron and steel enterprise as study object, batch and production planning problems of cold rolling are solved by considering the actual production situation in cold rolling enterprises. This dissertation has mainly carried on the following research.

(1) Nowadays, one of the most important challenges faced by cold-rolling mills industries is the adjustment of the production mode in order to satisfy market requirements subjected to fluctuations over time, mainly product varieties, demands, etc. To do so, we propose a batch planning model including three objectives, which are (i) to minimize the difference of delivery, (ii) to minimize the difference of process route, and (iii) to minimize adjustment times of mills of all the batches while meeting the requirements of process constraints such as weight, width of outlet (inlet), thickness of outlet (inlet), and tensile strength. And then a fuzzy clustering algorithm based on the particle swarm optimization which is used to solve the model is established.

(2) Cold rolling production process is characterized as multi-stage inventory and multi-flow directions. This paper presents a production planning model. The model meets process constraints, minimizing the total penalties caused by undue or overdue contracts, sufficient or insufficient intermediate inventory, and guaranteeing the minimum inventory costs of finished products. Genetic algorithm is improved by setting reasonable parameters to avoid the occurrences where crossover probability and mutation probability take fixed values within a certain range and the function values appear big jumps at some breakpoints or are unreasonable. These two probabilities can be adjusted adaptively by the fitness of individuals.

(3) In order to make the presented model and algorithm to get the promotion and application, developing a contract production system of cold rolling sheet based on B/S model of J2EE. This system can provide guidance for production in cold rolling sheet enterprises. Therefore, it will improve the management level for cold rolling enterprises.

The formulated mathematical models and the corresponding algorithms for production planning and batch are tested by practical data. The numerical analysis has shown that the proposed model and algorithm are feasible and effective.

**Key Words:** cold rolling; production plan; batch; multi-stage; multi-flow directions; improved genetic algorithm; fuzzy clustering; particle swarm algorithm

# 目录

摘要.....	I
Abstract.....	II
目录.....	III
1 绪论.....	1
1.1 研究背景.....	1
1.1.1 研究目的.....	2
1.1.2 研究意义.....	2
1.2 国内外研究现状及分析.....	2
1.2.1 国内研究现状.....	2
1.2.2 国外研究现状.....	4
1.2.3 问题提出.....	5
1.3 论文主要研究内容及结构.....	6
1.3.1 研究内容.....	6
1.3.2 论文框架.....	7
1.4 本章小结.....	7
2 某钢铁企业冷轧生产工艺及计划编制规则.....	9
2.1 酸轧工艺.....	9
2.2 退火工艺.....	9
2.3 生产计划编制规则.....	10
2.3.1 原料选择基准.....	10
2.3.2 轧制周期原则.....	10
2.3.3 计划顺序编制原则.....	11
2.3.4 宽度跳跃原则.....	11
2.3.5 原料厚度跳跃原则.....	12
2.3.6 原料强度跳跃原则.....	12
2.4 本章小结.....	12
3 基于模糊聚类的冷轧组批优化方法.....	13
3.1 冷轧组批模型.....	13
3.1.1 组批编制流程.....	13
3.1.2 相关数学符号定义.....	14
3.1.3 数学模型.....	15
3.2 基于粒子群优化的冷轧组批聚类算法.....	16
3.2.1 模糊 C 均值算法和粒子群算法简介.....	16
3.2.2 基于粒子群优化的冷轧组批聚类算法关键技术设计.....	17
3.2.3 算法步骤.....	18
3.3 应用实例.....	19
3.3.1 实验环境和数据.....	19
3.3.2 实验结果及分析.....	21
3.4 本章小结.....	24
4 考虑多阶段多流向的冷轧生产计划.....	25
4.1 冷轧生产计划问题描述.....	26

4.1.1 冷轧生产工艺路线.....	26
4.1.2 相关数学符号定义.....	26
4.1.3 冷轧生产计划数学模型.....	27
4.2 基于改进遗传算法的冷轧计划模型的求解.....	28
4.2.1 交叉、变异概率的自适应调整策略.....	29
4.2.2 改进遗传算法的求解步骤.....	30
4.3 应用实例.....	31
4.3.1 实验环境和数据.....	31
4.3.2 实验结果及分析.....	32
4.4 本章小结.....	35
5 系统设计与实现.....	36
5.1 系统开发技术与平台.....	36
5.1.1 Tomcat 服务器.....	36
5.1.2 MyEclipse 简介.....	36
5.2 数据库设计.....	37
5.3 系统核心代码.....	40
5.3.1 染色体基因编码.....	40
5.3.2 适应值计算.....	40
5.3.3 选择操作.....	43
5.3.4 交叉操作.....	45
5.3.5 变异操作.....	46
5.4 系统主界面.....	48
5.5 系统运行环境.....	49
5.6 本章小结.....	50
6 总结与展望.....	51
6.1 总结.....	51
6.2 展望.....	52
参考文献.....	53
图清单.....	56
表清单.....	57
攻读硕士学位期间参加的科研项目与发表学术论文情况.....	58
致谢.....	59

## 1 绪论

### 1.1 研究背景

冷轧企业是国民经济的支柱产业，为汽车、机械、建筑和其它工业提供基本的原材料，是钢铁企业一体化生产中的重要组成部分。冷轧产品在民生国计中占有着非常重要的地位，因此，钢铁企业都将冷轧产品加入到其产品生产链中。从钢铁企业的整个生产体系来看，冷轧生产位于生产体系的最末端，是对热轧产品进行再次深加工的复杂过程，其生产工艺也是在钢铁生产中技术含量和产品附加值较高的生产过程。而冷轧生产计划问题是冷轧生产管理的核心内容，主要是通过对产品加工和生产库存的合理有效管理，在满足客户订单要求的基础上，保证产品按期交货，降低企业生产成本，充分发挥并提高设备产能和利用率，加强各生产阶段间的相互协调，确保机组间的物流平衡和连续生产，实现生产计划的高效制定，最终达到资源的优化配置，提升企业的利润和核心竞争力的目的。

冷轧薄板企业属于资金密集型，制定高质量切实可行的生产计划，保证机组连续生产且合理的控制库存，实现企业的满负荷生产，并要求设备产能和物流能够保持平衡是冷轧企业生产经营和发展的一个重要目标。冷轧薄板产品的规格较多，生产工艺较复杂，不同品种和规格的产品的生产成本差别也较大，而企业的生产效率和产能的利用率都直接决定了企业的收益。同时冷轧薄板的生产具有多产品品种、多生产批量的特点，根据企业的客户订单，考虑各设备生产不同规格产品的生产能力和生产效率进行有效的管理，从而作出决策确定企业的生产计划，保证企业获取最佳的收益，对企业的决策者来说面临着非常复杂的问题。所以制定高质量的企业生产经营和作业生产计划，对企业生产将起到重要的指导作用，是企业管理的需要，更是企业持续发展的基础。

为了适应复杂的市场环境和个性化的客户需求，冷轧薄板企业的生产方式也由小品种、大批量转变为多品种、小批量的生产方式。由库存生产转变为订单生产的方式，同时对企业准时交货的能力提出了更高的要求。因此，冷轧企业要占有较高的市场份额，必须保证在信息化和自动化建设发展的基础上，实施准时和高效的生产管理战略，使企业生产计划与交货期和库存管理相匹配，并能够满足客户和产品生产工艺的要求。同时，冷轧薄板企业生产线各生产工序之间以中间库连接且有最低与最高库存限制，生产过程中各生产阶段间相互制约，并且有库存限制和工艺规范的要求，也加大了制定生产计划的难度。

本文以某钢铁企业冷轧薄板生产线的实际生产情况为背景，根据冷轧薄板企业生产系统的特点，并分析了冷轧企业生产中合同组批和生产排产出现的问题，研究了冷轧生产系统中合同组批和生产计划的建模及求解方法，并将研究成果应用到企业的生产实践中。



### 1.1.1 研究目的

对于冷轧企业，切实可行的生产计划依赖于客户个性化的需求和原材料供给，必然要求生产计划与交货期和库存安排进行相应的配合。研究合同交货期和库存成本可以解决企业因合同交货期设置不合理而遭受惩罚，因库存量与安全库存量不匹配而造成的成本和场地占用的问题，从而保证按期交货，降低生产成本，提高设备利用率和加强各生产阶段的协调，实现生产计划的高效制定，最终达到资源的优化配置，以提高企业的利润和核心竞争力的目的。

通过综述冷轧生产过程的特点，对合同组批和生产计划问题进行深入分析和研究，提出冷轧生产线的合同组批与生产计划编制方法，并将本文研究成果应用到某钢铁冷轧企业。

### 1.1.2 研究意义

冷轧企业的合同组批与生产计划问题比较复杂，研究以保证合同交货期、降低生产成本、优化生产排产、准时、高效为目标的钢铁企业冷轧合同组批和生产计划问题对深入研究合同组批与生产计划以及对冷轧企业的持续发展都具有深远的理论和实践意义。

- (1) 合理且高质量的冷轧生产计划是保证企业生产过程的连续性和产品质量的基础条件；
- (2) 建立可行的生产计划方法，将大大改进冷轧机组的生产能力，提高整个企业的生产能力和经济效益；
- (3) 研究冷轧生产计划，可以提高企业处理生产突发事件的能力，保证冷轧生产线的连续生产与优化；
- (4) 通过优化合同生产计划，可以进一步保证按期交货；
- (5) 采用合同组批的方式来组织生产可以提高生产效率，从而降低库存和生产成本。

## 1.2 国内外研究现状及分析

本文以冷轧企业合同组批和生产计划为研究对象，通过组批优化方法和生产计划建模两方面分别对国内外的研究现状进行了综述，并对冷轧组批优化方法和生产计划建模方面存在的问题进行了简明扼要的分析。

### 1.2.1 国内研究现状

#### 1) 组批优化方法

赵珺等研究了将生产合同组批以提高合同交货准确度的问题，建立了合同组

批数学模型并采用邻域搜索与改进的遗传算法相结合的方法对模型求解<sup>[1]</sup>。王志刚等提出了将生产合同按产品种类和交货期组批处理合同批量调度方法,建立了可以模糊处理生产时间的数学模型,并采用粒子群优化算法求解模型<sup>[2]</sup>。杨静萍等提出了基于空间和容量约束的聚类分析的组批方法,为组批生产计划提供了质量方面的量化方法<sup>[3]</sup>。唐成龙等建立了引入自适应度向量和自适应指数的自适应模糊聚类模型,实验表明该聚类模型能得到更好的聚类结果<sup>[4]</sup>。刘华强等采用模糊聚类与综合评判相结合的算法,建立了全连轧的轧制优化方法,并在实际应用中取得了良好的效果<sup>[5]</sup>。张晓平等解决了冷轧机的拼卷问题,建立了多目标多背包的数学模型,并构建了聚类方法与遗传算法相结合的算法对模型求解<sup>[6]</sup>。王利等研究了钢卷的宽度跳跃、入口厚度跳跃和出口厚度跳跃等多个工艺约束的冷轧机组生产计划编制问题,建立了冷轧生产计划多目标模型<sup>[7]</sup>。刘国莉等研究了冷轧生产可延迟供货的问题,构建了以库存保存成本、拖期惩罚和启动费用最小化为优化目标建立多阶段库存模型<sup>[8]</sup>。李庆予等提出了将客户的合同组批与企业的原材料组批相结合,进而实现客户合同的组批优化并取得了良好的效果<sup>[9]</sup>。彭威等介绍了一种合同组批的选料模式和自动生成要料单的优化方法以保证产能平衡<sup>[10]</sup>。谢志强等解决了两个不同加工时间的工序组批调度问题,并构建了嵌套批调度算法<sup>[11]</sup>。贾瑞玉等提出了将蚁群算法与遗传算法相结合的聚类算法,从而改善了过早收敛的问题<sup>[12]</sup>。张强等构造了遗传算法与模糊 C 均值聚类算法相结合的混合聚类算法,仿真实验结果表明构造的算法取得了良好的效果<sup>[13]</sup>。王利等提出了混合模型子空间聚类方法,以合同中待加工钢卷的宽度、冷轧机组的入口厚度、出口厚度以及合同的交货期为约束,进行生产合同组批<sup>[14]</sup>。

## 2) 生产计划建模与求解

唐立新深入研究了流程工业中的生产计划与调度的优化问题,并通过建立优化数学模型和利用智能优化算法解决钢铁企业生产过程中存在的计划与调度问题<sup>[15]</sup>。李耀华等以钢铁企业的实际生产情况为背景,研究了生产计划的编制流程,并建立了准时制的合同生产计划数学模型并采用改进的遗传算法求解<sup>[16]</sup>。安玉伟等研究了柔性作业车间生产计划与调度的问题,建立了生产计划与调度的集成优化数学模型,并构建了基于拉格朗日松弛分解的优化算法求解<sup>[17]</sup>。王志刚等为了解决冷轧多机组的生产排产问题,以各机组合同生产延期惩罚和生产类型切换虚拟成本最小化为目标,建立了合同生产计划的优化数学模型并采用启发式蚁群算法求解<sup>[18]</sup>。成国庆等以库存控制和设备更换前故障次数为变量,以运行成本最小化为目标建立数学模型,并利用演化算法对模型求解<sup>[19]</sup>。李铁克等建立了多阶段、多品种带有中间库的批量计划数学模型并采用粒子群优化算法求解<sup>[20]</sup>。严蕾为加快编制生产计划,缩短库存保存时间,改进了“见料排制计划”模式,实现了降低能耗、减少等待时间、降低库存成本的目标<sup>[21]</sup>。纪鹏程等研究了多品种多供应商的库存成本问题,建立了多品种多供应商的库存成本数学模型,并利用智能优

化方法对模型求解<sup>[22]</sup>。王林等研究了多目标的库存优化问题,以成本、缺货率和缺货量最小化为目标建立数学模型,并构建了混合智能优化算法对模型求解<sup>[23]</sup>。孙玲以冷轧生产过程中的设备产能、物料平衡、合同交货期等为约束,建立混合整数规划数学模型,并采用模拟退火算法对模型求解<sup>[24]</sup>。张涛等通过分析钢铁企业的生产过程,建立了合同生产计划的整数规划数学模型,并使用改进的遗传算法对模型进行求解<sup>[25]</sup>。蒋海波等通过对某工厂具体情况的分析对该工厂的生产计划进行了优化设计,运用 *Mathematics* 求解动态规划问题,并提出了合理的建议<sup>[26]</sup>。王文鹏等研究了多级多品种的批量生产计划问题,建立了以生产成本和库存保存成本最小化为目标的数学模型,并构建了两阶段优化算法对模型求解<sup>[27]</sup>。王洪锋等研究了考虑产能约束的订单生产排序问题,建立了混合整数规划数学模型,并使用运筹学方法求解<sup>[28]</sup>。赵艳刚分析了钢铁企业的库存情况,探讨了原材料和库存控制问题,提出了钢铁企业库存优化的方法并有效解决了问题<sup>[29]</sup>。

### 1.2.2 国外研究现状

#### 1) 组批优化方法

WANG 等建立了考虑轧制规程约束的成本函数,提出了基于冷轧优化调度的启发式遗传算法,实验表明所提出的方法能够显著改善以经验得出的解决冷轧问题的方法<sup>[30]</sup>。MOON 等研究了钢卷在进行批处理退火过程中的生产计划问题,并建立了混合的整数线性规划数学模型<sup>[31]</sup>。Amir 等针对混合和分类数据集发现了数据子空间聚类并提出了一种新的 K-均值聚类算法,并用真实的数据集进行测试和比较<sup>[32]</sup>。Theam 等提出了一种选择最佳子集的基于 Choquet 模糊积分的子空间聚类方法,并可以提高以聚类为基础的模式识别,实验结果说明所提出的方法有效<sup>[33]</sup>。Liu 等针对罩式退火过程中多机多阶段折返的问题,提出了一种改进的遗传算法与离散事件仿真技术相结合的优化调度方法<sup>[34]</sup>。Wang 等建立了一种多目标规划模型来解决订单组批问题,并使用基于局部自适应聚类算法的顺序分组方法以降低问题的复杂度<sup>[35]</sup>。Zhang 等提出了一种基于约束传播技术和聚类分析相结合的求解方法解决多个过程的批量计划优化问题<sup>[36]</sup>。TANG 等解决了钢铁生产过程中考虑作业运输和退化的单机分批计划排序问题。以最小化完工时间和批次数量为目标,确定工件运输和批次加工的顺序<sup>[37]</sup>。XU 等以最小化并行批处理生产的完工时间和最大延迟时间为目标建立模型,并提出了基于多目标蚁群优化的调度算法求解<sup>[38]</sup>。

#### 2) 生产计划建模与求解

Boyett 等指出合同交货期的预测在生产计划中是至关重要的因素,论述了 9 种模型使用利用率和存货清单数据预测铝、镁、钢铁、和钛产品的合同交货期,指出利用基于简单排队的非线性转换模型对预测合同交货期具有显著意义<sup>[39]</sup>。

Slotnick 建立了以交货期为目标的生产系统模型,对交货期准确性的预测是对客户交货期的良好承诺,指出一个综合钢厂的瓶颈是最小批量大小的处理,考虑加工时间,排队时间和订单量到来的时间是完成最小批量大小的要求<sup>[40]</sup>。Ramezani 等为了降低成本和库存的不稳定性,建立了混合整数线性规划模型,并用遗传和禁忌搜索算法求解<sup>[41]</sup>。Matsuda 等针对库存分配问题作为一个组合优化问题,提出根据每个订单选择厚度和板型不同的热轧板,高效实用原料,减少浪费,以满足订单优先级和收益率的两个目标进行优化,建立系统并用遗传算法来实现最优分配<sup>[42]</sup>。Mirmohammadi 等研究了通过两种优化方法相结合的方式解决多目标函数优化的问题。第一个优化阶段是基于内尔德和米德单纯形法,实现功率、压力和压下量分配的规划平衡。然后,利用基于实数编码的遗传算法作为外循环优化能耗和生产效率,并将实验结果应用到五机架冷连轧机上<sup>[43]</sup>。Chen 以改善冷连轧机的计算机在线过程控制的性能为目标,提出了一种基于成本函数,同时考虑了极限轧制力、轧制力平衡、前后张力、减速比、功率平衡和板形控制等因素的多目标负荷分配优化模型。应用结果表明,采用该负荷分配优化模型的两级过程控制系统具有运行稳定,精度高的特点;并得到了合理的轧制规程<sup>[44]</sup>。Pindrock 描述了美国钢铁和阿尔斯通电力转换最近现代化 84 英寸冷连轧在 Mon Valley 致力于世界一流标准,以满足汽车和家电行业的需求<sup>[45]</sup>。Moylan 研究了轧制过程中批量生产计划的编制问题,建立了线性规划的数学模型,并利用启发式算法对模型求解<sup>[46]</sup>。As'ad 等研究了钢铁轧钢厂经常遇到的实际的生产计划问题,采用模糊集理论的方法建立以成本最小为目标的生产计划模型,并利用罚函数算法对模型求解<sup>[47]</sup>。Huegler 等研究从现有库存满足钢板订单问题,指出半加工的库存板坯可以重新加热、轧制、切割来满足客户订单需求,并从经济学的角度阐述了对处理客户订单的作用<sup>[48]</sup>。LIU 等研究了冷轧生产过程中的拖期交货的问题,建立了以总库存成本、拖期惩罚和调整成本最小化为目标的多阶段的生产库存数学模型并提出了分割变量的拉格朗日松弛法对模型求解<sup>[49]</sup>。YANG 等研究了冷轧生产中的订单生产计划问题,以最小化换辊和调整退火温度所产生的转换成本为目标,建立了基于运筹学方法的非线性约束的数学模型<sup>[50]</sup>。

### 1.2.3 问题提出

#### 1) 组批优化方法方面

从上述研究的文献来看,模型中的约束条件大多没有同时体现合同的入口宽度、出口宽度、入口厚度、出口厚度、抗拉强度等关键因素,目标函数中大多都没有考虑将工艺路线作为评价指标。另外,虽然目前优化算法和聚类算法两种方法有一定的结合,但是适用于冷轧企业组批方面的却很少。从钢铁企业调研过程中发现,组批操作多采用人机交互的方式,由于组批时考虑的因素较多,容易造



成宽度、厚度以及抗拉强度等因素跳跃较大，违反生产工艺约束等问题。

## 2) 生产计划建模方面

目前国内外关于冷轧生产计划的研究大多只是在满足生产工艺的基础上，以交货期或库存单方面为研究目标建立生产计划数学模型。文献[24]研究了冷轧全流程生产计划编制问题，其建立的数学模型虽然考虑了交货期和中间库存，但模型中的库存控制是以物料等待加工时间的惩罚费用为目标，而且没有考虑成品库库存成本和多阶段库存问题。文献[20]建立的冷轧生产计划数学模型仅考虑了中间库存，而且约束中的库存平衡公式并未体现出生产批次多流向的特点；其库存平衡公式只提到了机组的加工速率，并没有乘以机组的加工时间，所建立的库存平衡公式并不平衡；而且，在优化目标函数中并没有考虑成品库的库存保存成本。

## 1.3 论文主要研究内容及结构

### 1.3.1 研究内容

为了保证企业在生产过程中能够按期交货，降低生产成本，提高设备利用率和加强各生产阶段的协调，高质量且合理的制定生产计划，最终达到企业资源的优化配置，从而实现提高企业的利润和核心竞争力的目的。本文基于面向订单的冷轧薄板生产计划方法及其应用存在的问题展开了研究，以冷轧企业中的合同组批和生产计划为研究对象，分别在数学模型和优化算法两方面进行了研究，主要包括如下内容：

#### 1) 在模型方面

(1) 本文以合同交货期差异度、工艺路线差异度和调整次数最小化为目标，以合同中待加工钢卷的批次重量、出(入)口宽度、出(入)口厚度、抗拉强度为约束建立冷轧组批数学模型。

(2) 以合同交货期惩罚、中间库存量惩罚以及成品库库存成本最小化为目标建立冷轧计划数学模型，侧重考虑冷轧生产系统多阶段库存（包括中间库和成品库）、批次多流向问题，并在文献[20]的基础上设计了考虑批次多种流向的库存平衡公式。

#### 2) 在算法方面

(1) 在求解冷轧组批数学模型过程中，针对合同交货期和工艺路线两个属性分别提出了余弦差异度和工艺路线差异度的计算方法，并构造了基于粒子群的冷轧组批模糊聚类优化算法对模型求解。所构造的算法将待组批的生产合同抽象为样本数据，在聚类过程中首先利用粒子群优化算法来寻找最优的聚类中心，接着根据最优聚类中心使用模糊聚类算法求解隶属度矩阵，然后结合约束条件得到组批结果。

(2) 在求解冷轧生产计划模型过程中，针对文献[51]所提出的自适应遗传算

法在一定区间内取固定值和断点处的函数值跳跃大以及出现不合理值,而且不能完全自适应调整的问题。改进的遗传算法通过合理设定参数  $A$ 、 $B$  和  $r$ , 改善了分段函数(交叉、变异概率公式)在一定区间内取固定值和断点处的函数值跳跃大以及出现不合理值的问题,能够根据个体适应值得到相应的交叉概率和突变概率,从而实现在种群进化过程中交叉概率和突变概率的自适应调整。

### 1.3.2 论文框架

本文通过对冷轧企业生产线的工艺流程和生产业务充分调研并分析的基础上,将企业实际问题抽象为数学模型并利用智能优化方法得出最佳决策方案。

本篇论文共分为六章,其主要章节内容架构如下所示:

第一章:提出了本文的研究目的及其意义,回顾了钢铁企业生产计划研究的现状,并给出本文的主要研究内容以及文章结构。

第二章:介绍了钢铁企业冷轧生产工艺及生产计划编制规则,描述了某钢铁企业酸轧工艺流程和连续退火工艺流程。

第三章:对合同组批问题进行了分析和研究,并描述了企业在合同组批中存在的问题,从合同交货期差异度、工艺路线差异度和调整次数的角度建立了冷轧组批数学模型,并构造了基于粒子群的冷轧组批模糊聚类优化算法对模型求解。

第四章:对冷轧企业生产计划问题进行了研究,描述并分析了目前对冷轧生产计划的研究存在的问题,从合同交货期和库存控制两方面建立了冷轧生产计划数学模型,并提出了改进的遗传算法求解。

第五章:冷轧薄板订单生产计划系统的设计与实现,从冷轧企业的实际需求出发,使用 Myeclipse 开发平台并设计出本系统所需要的数据库,介绍了系统的主要功能和作用。

第六章:归纳和总结了本篇论文的主要研究内容,并且对接下来的研究内容作出了展望。

论文架构图如下图 1.1 所示。

### 1.4 本章小结

本章阐述了研究的背景、目的及意义,以冷轧企业中的合同组批和生产计划两方面为研究对象,对国内及国外的研究现状分别进行了综述,提出并分析了所存在的问题。针对合同组批优化方法和生产计划建模存在的问题分别在数学模型和优化算法两方面分别进行了深入研究,并给出了论文的整体框架。

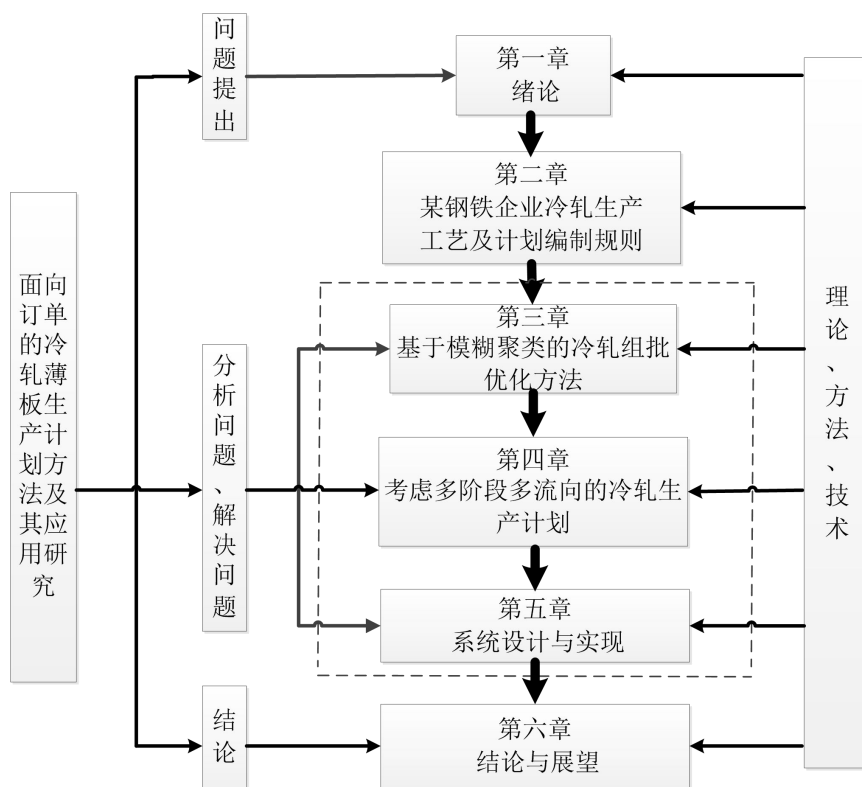


图 1.1 论文框架图

## 2 某钢铁企业冷轧生产工艺及计划编制规则

为了研究冷轧企业的合同组批和生产计划问题,本章以某钢铁企业实际冷轧生产线为背景,分别简要介绍并分析了冷轧生产中的酸轧和退火工艺以及制定生产计划的编制规则。

### 2.1 酸轧工艺

酸轧是冷轧企业生产中的最重要的工序,为了保证带钢的质量,带钢在冷轧之前必须除去氧化铁皮即除磷,除磷的方法主要以酸洗为主。带钢经酸洗后较快的通过冷轧机,生产出板型和表面质量良好的产品。为了提高生产效率,一般冷轧车间都设有连续酸轧加工线。

某钢铁企业酸轧工艺流程图如图 2.1 所示:

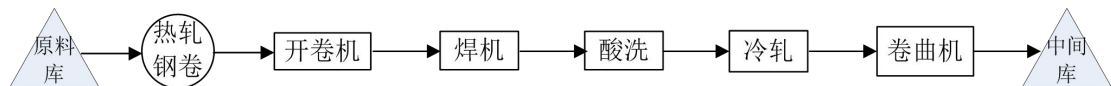


图 2.1 酸轧工艺流程图

该企业采用连续式酸洗线和串列式冷轧机的联合机组,从而可以为企业生产出高质、高产和高收益的冷轧卷。

酸轧机组是以热轧卷为原料,在冷轧生产线的入口处,原料钢卷通过钢卷小车运送到开卷机上对钢卷进行开卷操作,并将前后已开卷的带钢首尾用焊机焊接在一起,以满足向冷轧生产线连续供卷的要求。之后,将带钢通过传送装置输送进入酸洗机组。带钢在酸洗机组上首先进行破鳞处理,然后酸洗清除原料钢卷表面上被氧化的铁皮,最后酸洗后的带钢由烘干机烘干带钢表面后即被输送到冷轧机组上进行轧制操作。冷轧是冷轧生产线上核心工序,现场操作人员会根据客户的要求对冷轧机的轧制参数进行调整。

带钢经冷轧机组轧制后会输送到冷轧机的出口的卷取机上进行卷曲操作,将带钢卷成卷便于储藏和运输,因此卷取机也是冷轧生产线上的重要设备。对带钢进行卷曲操作完成后对其进行称重打捆,然后运送到中间库等待下一道工序或等待发货。

### 2.2 退火工艺

退火是将带钢缓慢加热到一定温度,保持足够时间,然后以适宜速度冷却的热处理工艺。目的是改善或消除钢铁在铸造、锻压、轧制和焊接过程中所造成的各种组织缺陷以及残余应力,防止带钢变形、开裂;降低硬度,软化带钢以便进行切削加工;稳定尺寸,减少变形与裂纹倾向;细化晶粒,改善组织以提高带钢



的机械性能；均匀材料组织和成分，改善材料性能或为以后热处理做组织准备。

某钢铁企业退火工艺流程图如图 2.2 所示：

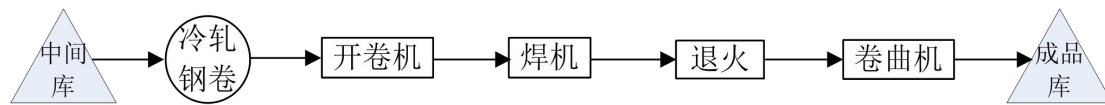


图 2.2 退火工艺流程图

退火机组以上道工序的酸轧机组的成品卷（冷轧钢卷）为原料钢卷。酸轧机组的成品卷存放于与连退工序之间设置的中间库内。

进行退火生产时，冷轧卷由吊车将要退火的钢卷从中间库内依次吊放到退火生产线入口的步进梁上，由钢卷小车将冷轧卷运送到开卷机上进行开卷。开卷后的带钢被运送到焊机上，将开卷后的带钢焊接在一起。焊好后的带钢要对其表面进行清洗，以除去带钢表面上油污。再用烘干器烘干，去除带钢表面上的残水。之后将带钢输送到连续退火炉中执行连续退火操作。连退是退火生产线上的核心工序，带钢在退火炉内随着温度的升高而完成恢复和再结晶的过程。带钢经过退火工序退火后被输送到卷取机上进行卷取操作以方便带钢的储藏和运输。带钢卷成钢卷后对其称重和打捆，之后被运送至成品库储藏或等待发货。

## 2.3 生产计划编制规则

生产计划的编制需要满足生产工艺、机组产能等约束的前提下，一般冷轧机组计划编制应遵循如下原则：原料选择基准、轧制周期原则、计划顺序编制原则。

### 2.3.1 原料选择基准

对于冷轧企业，冷轧机组的原料为热轧钢卷，需要钢铁企业为冷轧企业提供原材料，冷轧企业才能编制生产计划，进行生产运作。因此，冷轧机组在生产计划编制中，对于进行编制生产计划的原料，必须满足以下条件：

- (1) 原料必须已在冷轧原料库中；
- (2) 原料不在封闭状态；
- (3) 与原料相联的生产订单不在修改或关闭状态；
- (4) 在保证排产规则的前提下，原料库中存放时间较长的原料应优先排入生产计划中。

### 2.3.2 轧制周期原则

冷轧机组具有产能的限制，根据机组产能约束，编制生产计划一般设置为一个轧制周期。轧制周期由工作辊的轧制吨位决定，原则上 2200 吨（正负 200 吨）为一个轧制周期。机组轧制一个周期后需要更换轧机工作辊。

典型轧制周期和周期轧制吨位如图 2.3 所示

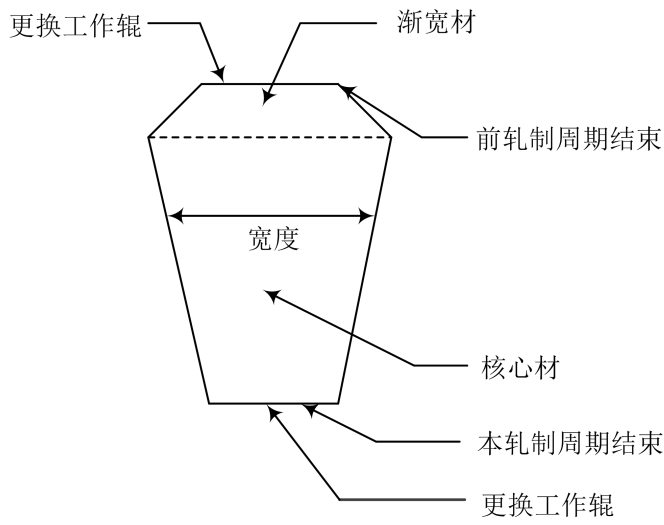


图 2.3 典型轧制周期和周期轧制吨位图

2.3.3 计划顺序编制原则

在生产原材料符合原料选择基准和轧制周期原则后，并确认生产的产品符合本机组生产能力的前提下，编制生产计划应遵循厚度、宽度、强度等优先级，计划顺序编制优先级如下表 2.1 所示。

注：数字越小，优先级越高。

表 2.1 计划顺序编制优先级

热板厚度范围 mm	产品宽度	切边量	原料厚度	产品厚度	原料强度
2.0~6.0	1	2	3	4	5

2.3.4 宽度跳跃原则

根据客户对产品的要求，每个钢卷的轧制宽度会不断变化，为保证企业生产的连续性并确保冷轧辊无损，必须对钢带的轧制宽度进行平稳的过渡，因此在轧制过程中要遵循宽度跳跃原则。

1) 核心材料渐窄原则，如表 2.2 所示。

表 2.2 核心材料渐窄原则

产品目标厚度	出口宽度变窄量	
	常规	极限
0.800~2.500	≤150mm（最好 60-80mm）	≤200mm
0.200~0.799	≤130mm	≤150mm

2) 渐宽材料

渐宽材料宽度的允许范围：

前后宽度为由厚变薄：原料宽度差 $\leq 200\text{mm}$ ；

前后宽度为由薄变厚：原料宽度差 $\leq 100\text{mm}$ 。

渐宽材须注意：

- (1) 应选择产品表面质量等级为 S2 的作为渐宽材料；
- (2) 不得将轧硬外卖卷的原料作为渐宽材料；
- (3) 禁止连续 2 个相同宽度的钢卷作为渐宽材；
- (4) 优先考虑 CQ\DQ(原则上用 CQ，当没有 CQ 时才用 DQ)；
- (5) 优先考虑较宽材料，减少渐宽材的使用。

### 2.3.5 原料厚度跳跃原则

原料厚度在经过轧制工序时会根据客户的不同要求对轧辊进行调整，在保护轧辊并满足生产工艺的前提下，轧辊在调整时需要进行平缓的变化，即应该遵循原料厚度跳跃原则。

- 1) 原料厚度差：( $D$ 、 $d$  分别为较厚带钢厚度和较薄带钢厚度。)

原料厚度 $> 3.2\text{mm}$  时， $|d-D| \leq 0.8\text{mm}$ ；

原料厚度 $\leq 3.2\text{mm}$  时， $|d-D| \leq 0.5\text{mm}$ 。

- 2) 产品厚度  $h > 1.5\text{mm}$ ，允许厚度差 $\Delta h \leq 0.6\text{mm}$ ；

产品厚度  $h \leq 1.5\text{mm}$ ，允许厚度差 $\Delta h \leq 0.5\text{mm}$ 。

如果计划编制不能满足上述出口厚度原则，操作人员可根据跳跃前后的轧制工艺参数变化情况决定是否需要重新穿带。

- 3) 相同产品厚度集中生产。

### 2.3.6 原料强度跳跃原则

钢卷具有抗拉强度的特性，不同抗拉强度的钢卷强度不同，其用途也不同。不同强度的钢卷在轧制中要满足原料强度跳跃的原则以保证产品质量。

- 1) 抗拉强度最大比值  $\sigma_{b1}/\sigma_{b2} \leq 1.2$  ( $\sigma_b$  为带钢抗拉强度)；

- 2) 同一钢种尽量集中生产。

## 2.4 本章小结

本章阐述了某钢铁企业冷轧薄板厂的生产工艺，分别详细概述了酸轧工艺流程和退火工艺流程，以及介绍了某钢铁企业冷轧生产计划编制规则，为下面冷轧合同组批和生产计划问题的研究打下了基础。

### 3 基于模糊聚类的冷轧组批优化方法

通过对冷轧组批优化方法的研究文献分析可知,模型中的约束条件大多没有同时体现合同的入口宽度、出口宽度、入口厚度、出口厚度、抗拉强度等关键因素,目标函数大多没有考虑将工艺路线作为评价目标。另外,虽然目前优化算法和聚类算法两种方法有一定的结合,但是适用于冷轧企业组批方面的却很少。从钢铁企业调研过程中发现,组批操作多采用人机交互的方式,由于组批时考虑的因素较多,容易造成宽度、厚度以及抗拉强度等因素跳跃较大,违反工艺约束等问题。

为此,本章以合同交货期差异度、工艺路线差异度和调整次数最小化为目标,以合同中待加工钢卷的批次重量、出(入)口宽度、出(入)口厚度、抗拉强度为约束建立冷轧组批数学模型。在模型求解过程中,提出了合同交货期余弦差异度和工艺路线差异度的计算方法,并设计了基于粒子群优化的冷轧组批模糊聚类算法。算法将待组批合同抽象为样本数据,聚类过程中首先利用粒子群优化算法寻找最优的聚类中心,接着根据最优聚类中心使用模糊聚类算法求解隶属度矩阵,然后结合约束条件得到组批结果,最后以国内某冷轧企业为案例对模型和算法的进行了验证。

#### 3.1 冷轧组批模型

##### 3.1.1 组批编制流程

合同组批即根据合同的规格、交货期以及生产工艺等因素将其划分为多个批次,然后按批次进行排产,以减少批次间以及批次内的调整次数,从而充分发挥各机组的产能优势,提高现有资源的利用率。从 CIMS 的框架来看,合同组批位于生产计划的上层,是编制生产计划的前提条件,组批结果的好坏直接决定了生产计划的执行效率。若组批时引入生产计划需要考虑的诸如重量、宽度、厚度和抗拉强度等因素,可以实现组批和生产计划的无缝集成,利于生产管理者的理解和接受,可操作性较强。



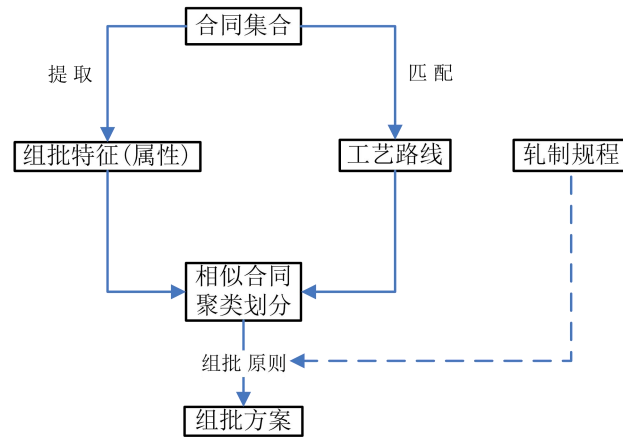


图 3.1 冷轧合同组批编制流程

图 3.1 为冷轧合同组批编制流程，需要考虑的主要因素有合同规格、轧制规程、工艺路线等，其中合同规格的主要属性有：钢种、入口厚度、出口厚度、入口宽度、出口宽度和抗拉强度等。由于钢种和抗拉强度具有较强的耦合关系，本文选取抗拉强度作为组批属性，如果不同钢种的抗拉强度在一定范围内，则可以进行组批。

可见，上述组批问题可转化为聚类问题，将一个合同集合根据组批原则划分为若干个子集合，每个子集合就称为一个批次。本章采用模糊聚类方法的思想建立组批优化数学模型。

### 3.1.2 相关数学符号定义

设  $i=1, 2, \dots, I$  表示聚类形成的批次编号， $j=1, 2, \dots, N$  表示合同编号。 $U_i$  表示第  $i$  个批次所包含的合同集合， $|U_i|$  表示批次  $i$  中包含的合同数。 $\omega_{ij}$  表示合同  $j$  对聚类中心  $i$  的隶属程度， $\Omega$  表示隶属度矩阵， $\Omega=\{\omega_{ij}, 1 \leq i \leq I, 1 \leq j \leq N\}$ ， $d_{ij}$  表示聚类(中心) $i$  到合同  $j$  的距离。 $\lambda_j$  是拉格朗日算子， $m$  是模糊指数。 $W_j$  为合同  $j$  的原料入口宽度， $F_i$  为批次  $i$  内所有合同原料入口的最大宽度差， $W'_j$  为合同  $j$  的产品出口宽度， $F'_i$  为批次  $i$  内的产品出口最大宽度差， $H_j$  为合同  $j$  的原料入口厚度， $a$  为原料入口厚度分界点， $Q_i$ 、 $Q'_i$  为批次  $i$  内的原料入口最大厚度差， $H'_j$  为合同  $j$  的产品出口厚度， $b$  为产品出口厚度分界点， $Y_i$ 、 $Y'_i$  为批次  $i$  内的产品出口最大厚度差， $\sigma_{bj}$  为合同  $j$  的抗拉强度， $Z$  为批次内抗拉强度最大比值。 $M_j$  表示合同  $j$  的重量， $P$  表示每个批次的重量限制。 $G_i$  表示批次  $i$  中相邻合同的交货期重叠指数。 $S_i$  表示批次  $i$  中包含不同规格合同时导致生产机组调整的次数。设  $j, k$  表示批次  $i$  中的合同编号， $R_{j,k}$  表示合同  $j$  与合同  $k$  工艺路线的差异度。 $\alpha$ 、 $\beta$  与  $\gamma$  分别表示合同交货期差异度、机组调整次数、工艺路线差异度的权重。

### 3.1.3 数学模型

基于冷轧生产合同和生产工艺的要求,分别建立聚类过程目标函数与聚类评估目标函数。

目标函数:

$$E(\Omega) = \sum_{i=1}^I \sum_{j=1}^N \left( (\omega_{ij})^m \cdot d_{ij}^2 \right) + \sum_{j=1}^N \lambda_j \cdot \left( 1 - \sum_{i=1}^I \omega_{ij} \right) \quad (3.1)$$

$$E(G, S, R) = \min \left( \sum_{i=1}^I \left( \alpha \cdot G_i + \beta \cdot S_i + \gamma \cdot \left( \sum_{j,k \in U_i} R_{j,k} \right) / C_{|U_i|}^2 \right) \right) \quad (3.2)$$

约束条件:

$$\alpha + \beta + \gamma = 1, \quad (3.3)$$

$$\sum_{j \in U_i} M_j \leq P, \quad (3.4)$$

$$\max_{j \in U_i} W_j - \min_{j \in U_i} W_j \leq F_i, \quad (3.5)$$

$$\max_{j \in U_i} W'_j - \min_{j \in U_i} W'_j \leq F'_i, \quad (3.6)$$

$$\begin{cases} \max_{j \in U_i} H_j - \min_{j \in U_i} H_j \leq Q_i, H_j > a \\ \max_{j \in U_i} H_j - \min_{j \in U_i} H_j \leq Q'_i, H_j \leq a \end{cases}, \quad (3.7)$$

$$\begin{cases} \max_{j \in U_i} H'_j - \min_{j \in U_i} H'_j \leq Y_i, H'_j > b \\ \max_{j \in U_i} H'_j - \min_{j \in U_i} H'_j \leq Y'_i, H'_j \leq b \end{cases}, \quad (3.8)$$

$$\max_{j \in U_i} \sigma_{b,j} / \min_{j \in U_i} \sigma_{b,j} \leq Z. \quad (3.9)$$

目标函数式(3.1)是聚类过程目标函数,第1项表示生产合同的各维属性与组批聚类中心的距离,第2项为 $\omega_{ij}$ 的控制项。式(3.2)是聚类评估目标函数,保证组批结果中合同的交货期相近,工艺路线相似并且最小化机组的调整次数,第1项表示批量合同交货期差异度,第2项表示批量合同引起机组调整次数,第3项表示批量合同工艺路线差异度。式(3.3)表示式(3.2)中交货期差异度、调整次数、工艺路线差异度的权重约束。式(3.4)是根据冷轧机组工艺要求,组批形成的批量合同的重量限制。式(3.5)是原料入口宽度差约束。式(3.6)是原料出口宽度差约束。式(3.7)是原料入口厚度差约束。式(3.8)是原料出口厚度差约束。式(3.9)是原料抗拉强度的变化约束。

其中,合同交货期差异度 $G_i$ 由式(3.10)求得,合同的交货期属性以区间表示。 $g_j$ 表示合同 $j$ 的交货期窗口大小, $g_j \cap g_{j+1}$ 表示相邻合同的重叠交货期, $g_j \cup g_{j+1}$ 表示相邻合同的最大交货期, $G_i$ 越小,则批次 $i$ 内相邻两个合同的交货期相似度

越大。

$$G_i = \sum_{j=1}^{|U_i|-1} \cos\left(\frac{\pi}{2} \cdot \frac{g_j \cap g_{j+1}}{g_j \cup g_{j+1}}\right). \quad (3.10)$$

工艺路线差异度  $R_{j,k}$  由式(3.11)求得, 式中  $L_{j,k}$  表示合同  $j$  与合同  $k$  工艺路线中公共子序列的长度,  $\text{len}(j)$ ,  $\text{len}(k)$  依次表示合同  $j$ ,  $k$  的工艺路线长度。  $R_{j,k}$  越大表示合同  $j$ ,  $k$  工艺路线的差异度越大。

$$R_{j,k} = 1 - \frac{L_{j,k}}{\max(\text{len}(j), \text{len}(k))}. \quad (3.11)$$

式(3.10)和(3.11)用以计算交货期和工艺路线两个区间型变量的距离, 而对于宽度、厚度和抗拉强度连续型变量的距离宜采用欧式距离公式来计算。因此, 本文根据实际问题设计了式(3.12)实现式(3.1)中的聚类中心  $i$  到合同  $j$  的距离  $d_{ij}$ 。设  $d=1, 2, \dots, D$  表示合同的属性维度序号,  $v_{id}$  表示第  $i$  个聚类中心的第  $d$  维属性,  $x_{jd}$  表示第  $j$  个合同的第  $d$  维属性, 设  $p$  为包含交货期和工艺路线属性的集合, 综合式(3.10)和(3.11)可得到混合型样本间距离公式(3.12)。

$$d_{ij} = \sqrt{\sum_{\substack{d=1 \\ d \notin p}}^D (v_{id} - x_{jd})^2} + \cos\left(\frac{\pi}{2} \cdot \frac{g_i \cap g_j}{g_i \cup g_j}\right) + \left(1 - \frac{L_{i,j}}{\max(\text{len}(i), \text{len}(j))}\right) \quad (3.12)$$

## 3.2 基于粒子群优化的冷轧组批聚类算法

### 3.2.1 模糊 C 均值算法和粒子群算法简介

模糊 C 均值聚类算法是一种广泛使用的划分方法。考虑一个样本集合  $X=\{x_1, x_2, \dots, x_n\}$ , 利用模糊聚类算法将其划分为  $I$  个子集合,  $I$  是大于 1 的正整数。对于任一子集合, 可以用隶属程度  $\omega_{ij}$  和聚类中心  $v_i$  两个参数来描述, 其中  $\omega_{ij}>0$ ,  $\sum_{i=1}^I \omega_{ij}=1, 1 \leq j \leq n$ 。对于任一个样本, 根据该样本到所有聚类中心的隶属度值的大小判断该样本划分到某一聚类中。

粒子群优化算法是一种源于对鸟群和鱼群群体觅食行为模拟的群体智能方法。该算法的基本思想是随机初始化粒子的位置和速度, 每个粒子代表一个可能的解; 通过迭代公式(3.13)和(3.14)不断更新自己的位置和速度, 最后找到问题空间的最优解。在迭代过程中, 粒子群算法需要不断更新每一个体的最好位置(个体极值)和种群的最好位置(全局极值), 每个粒子在飞行时根据个体极值和全局极值对飞行速度进行动态调整。速度和位移的迭代公式如下:

$$v_i(t+1) = w \cdot v_i(t) + c_1 \cdot r_1(t) \cdot (p_i(t) - x_i(t)) + c_2 \cdot r_2(t) \cdot (p_g(t) - x_i(t)) \quad (3.13)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (3.14)$$

其中:  $v_i$ ,  $x_i$  分别表示第  $i$  个粒子的速度和位移;  $c_1$  和  $c_2$  是学习因子;  $r_1$  和

$r_2$  是均匀分布在(0,1)区间的随机数;  $w$  是惯性权重;  $p_i$  表示第  $i$  个粒子个体极值,  $p_g$  表示粒子群的全局极值;  $t$  表示当前的迭代次数。

式(3.13)由三个部分构成, 共同决定了粒子的空间搜索方向。其中, 第一部分是粒子上一次迭代的速度, 表明了粒子的当前状态; 第二部分表示粒子的独立思考能力, 使粒子有了足够强的全局搜索能力, 避免局部极小; 第三部分体现了粒子间的信息是互通、共享的。

### 3.2.2 基于粒子群优化的冷轧组批聚类算法关键技术设计

模糊聚类算法实现简单, 快速而且能够有效处理大数据集, 但是存在诸多问题, 如算法对初始聚类中心敏感, 收敛速度慢, 易陷入局部最优等。基于种群操作的粒子群优化算法初始为均匀分布于解空间的若干可能解, 具有很强的全局搜索能力, 不易陷入局部区域, 收敛速度很快。与遗传算法相比, 粒子群算法简单容易实现、需要设置的参数较少、不需要经过交叉、变异等操作, 而优化结果甚至优于遗传算法。此外, 在解决一些复杂问题时遗传算法存在收敛速度慢、易早熟的缺陷。基于上述分析, 本文提出了基于粒子群优化的冷轧组批模糊聚类算法, 既能充分发挥粒子群优化算法的全局搜索能力, 又能避免模糊聚类算法陷入局部最优。首先利用粒子群算法的优化搜索能力寻找全局最优的聚类中心, 然后利用模糊聚类算法进行聚类。该算法中粒子编码、适应度函数、隶属度矩阵更新三个关键技术的设计如下。

#### (1) 粒子编码

聚类算法的关键是确定聚类中心, 因此本文提出的算法利用粒子群优化算法的全局搜索能力寻找全局最优的聚类中心。在粒子编码时, 随机选取  $I$  个样本点作为初始聚类中心, 并由这些聚类中心组成一个粒子。按照这种方式即可生成指定规模的初始粒子群。

#### (2) 适应度函数

算法中任一粒子都是由聚类中心直接组成, 因此每个粒子代表一种聚类方式, 其适应度值大小表示样本集在该聚类方式下聚类效果的优劣。本文建立的数学模型中, 目标函数(3.2)是计算所有批次内的合同交货期差异度、调整次数、工艺路线差异度之和, 其作用相当于对聚类结果的评价, 因此选取目标函数(3.2)式作为算法的适应值函数。

#### (3) 隶属度矩阵更新

隶属度可用于描述样本对聚类中心的隶属程度, 也是模糊聚类算法分析结果的依据。本文提出的算法不仅使用粒子群算法获得全局最优的聚类中心, 而且也使用模糊聚类算法获得聚类结果, 因此目标函数(3.1)式作为该算法的聚类过程目标函数。当使用粒子群算法获得聚类中心  $v_i$  后, 目标函数(3.1)式中的  $d_{ij}$  可根据



式(3.12)计算得到,此时它可视为一常数。为了求隶属度 $\omega_{ij}$ ,利用 Lagrange 乘数法对(3.1)式进行求导可得:

$$\begin{cases} \frac{\partial E}{\partial \lambda_j} = \sum_{j=1}^N \left( 1 - \sum_{i=1}^I \omega_{ij} \right) = 0 \\ \frac{\partial E}{\partial \omega_{ij}} = \sum_{j=1}^N \sum_{i=1}^I \left( m \cdot d_{ij}^2 \cdot \omega_{ij}^{m-1} \right) - \sum_{j=1}^N \lambda_j = 0 \end{cases},$$

$$\text{整理可得: } \begin{cases} 1 - \sum_{i=1}^I \omega_{ij} = 0 \\ \sum_{i=1}^I \left( m \cdot d_{ij}^2 \cdot \omega_{ij}^{m-1} \right) - \sum_{j=1}^N \lambda_j = 0 \end{cases},$$

$$\text{求得: } \omega_{ij} = \begin{cases} \frac{1}{\sum_{k=1}^I \left( \frac{d_{ij}}{d_{kj}} \right)^{\frac{2}{m-1}}}, d_{ij} \neq 0 \\ 1, d_{ij} = 0, k = j \\ 0, d_{ij} = 0, k \neq j \end{cases}. \quad (3.15)$$

公式(3.15)中的第2项和第3项表明,当聚类中心 $i$ 到合同 $j$ 的距离 $d_{ij}=0$ 时,合同 $j$ 的位置与聚类中心的位置重合,此时合同 $j$ 一定完全隶属于聚类中心 $i$ ,即 $\omega_{ij}=1$ ,合同 $j$ 对于其他的聚类中心的隶属度 $\omega_{ij}=0$ ;当聚类中心 $i$ 到合同 $j$ 的距离 $d_{ij} \neq 0$ 时,根据式中的第1项计算隶属度矩阵可使得目标函数(3.1)达到最小值。

### 3.2.3 算法步骤

基于粒子群优化的冷轧组批模糊聚类算法的具体步骤设计如下:

步骤1: 初始化算法参数,包括聚类数目 $I$ 、粒子种群大小 $pnum$ 、学习因子 $c_1$ 和 $c_2$ 、惯性权重 $w$ 、最大迭代次数 $iterations$ 、模糊指数 $m$ 的值;

步骤2: 初始化粒子群。假定需要组批的合同具有 $D$ 维属性,则每个合同即是一个样本, $N$ 个合同构成了一个 $N \times D$ 维的样本集。首先从样本集中随机选择 $I$ 个样本点作为初始聚类中心并构成聚类中心集合 $V = \{ \{ v_{11}, v_{12}, \dots, v_{1D} \}, \{ v_{21}, v_{22}, \dots, v_{2D} \}, \dots, \{ v_{I1}, v_{I2}, \dots, v_{ID} \} \}$ ,即为一个粒子,其编码为 $(v_{11}, v_{12}, \dots, v_{1D}, v_{21}, v_{22}, \dots, v_{2D}, \dots, v_{I1}, v_{I2}, \dots, v_{ID})$ 。重复这一过程直至产生 $pnum$ 个粒子组成初始粒子群为止;

步骤3: 计算样本到聚类中心的距离。对于任一粒子 $j$ ,首先将粒子编码根据其生成规律解析成 $I$ 个聚类中心编码;然后根据公式(3.10)和公式(3.11)计算聚类中心 $i$ 和合同 $j$ 的交货期差异度和工艺路线差异度;最后根据公式(3.12)计算粒子到聚类中心的距离 $d_{ij}$ ;

步骤 4: 更新隶属度矩阵。根据 3.2.2 节的分析可知, 当隶属度矩阵按照公式(3.15)进行迭代时, 目标函数(3.1)式的取值最小。因此结合步骤 3 得到的距离  $d_{ij}$  和公式(3.15)计算隶属度  $\omega_{ij}$  并更新隶属度矩阵  $\Omega$ ;

步骤 5: 解析聚类结果。根据步骤 4 所得到的新隶属度矩阵, 结合模型中的式(3.4)~(3.9)对同一批次内的重量、入口宽度、出口宽度、入口厚度、出口厚度和抗拉强度的约束, 得到该隶属度矩阵所对应的组批结果;

步骤 6: 适应值评价。根据步骤 5 解析得到的聚类结果, 按照目标函数(3.2)计算每个批次内合同交货期差异度、调整次数和工艺路线差异度, 其中交货期差异度根据公式(3.10)求得, 工艺路线差异度根据公式(3.11)求得。本文规定, 当合同规格不同时需要调整一次, 因此批次内的合同种类即为该批次的调整次数。最后将适应值函数(3.2)所得结果作为粒子的适应度值;

步骤 7: 更新粒子的最好位置。将每个粒子的适应值与该粒子的最好适应值进行比较。如果该粒子的当前适应值较好, 则将该粒子的当前适应值作为该粒子的最好适应值并将该粒子的当前位置作为最好位置;

步骤 8: 更新种群的最好位置。将种群中每个粒子经过的最好位置与种群所经历的最好位置进行比较, 如果某一粒子所经过的最好位置较好, 则将该粒子的最好位置作为种群的最好位置;

步骤 9: 更新粒子的速度。按照公式(3.13)调整粒子的飞行速度, 并判断是否超过了最大速度  $v_{max}$ , 如果超过则用  $v_{max}$  取代。

步骤 10: 更新粒子的位移。按照公式(3.14)更新粒子的位移, 并判断是否超过论域空间边界, 如果超过则该粒子用原值取代。

步骤 11: 终止条件判断。若算法达到了指定的迭代次数, 算法结束; 否则返回步骤 3。

步骤 12: 结果输出。包括最佳粒子、最佳粒子适应值、组批划分等。

### 3.3 应用实例

#### 3.3.1 实验环境和数据

采用 Java 语言在 PC Corei5/CPU2.5GHz/RAM4.0GB 计算机上编程实现, 实验环境为 myEclipse8.5。在借鉴相关文献的基础上, 根据对算法的多次试验结果, 确定了算法的基本参数。为了便于描述, 本文将模糊 C 均值聚类算法称为“FCM 算法”, 将基于粒子群算法的冷轧组批聚类算法称为“本文算法”。

实验参数: (1) FCM 算法参数: 模糊指数  $m=2$ ; 聚类数目  $I=4$ ; 迭代次数  $iterations=100$ 。(2) 本文算法参数: 一般来说, 随着种群规模的增大, 算法的收敛速度越慢、搜索精度越高、稳定性也越好。但是, 种群规模太大会造成运行时间过长。实际应用中, 需要根据具体问题进行综合考虑和分析。因此, 在考虑精

度、稳定性和运行时间之间的权衡关系以及实际问题维度的情况下，设定种群规模  $pnum=100$ ；惯性权重  $w$  较大时，算法的全局搜索能力较强，较小时，算法的局部搜索能力较强。经过多次实验最终设定  $w=0.6$ ；学习因子  $c_1$ 、 $c_2$  分别调节向个体和全局最优解方向飞行的最大步长。小的值允许粒子被拉回前可以在目标区域外徘徊，而大的值会导致粒子突然冲向或越过目标区域。合适的  $c_1$ ， $c_2$  可以加快收敛速度且不易陷入局部最优，结合本文实际令  $c_1=c_2=2$ 。其他参数与 FCM 算法参数相同。(3) 模型参数：两种算法均用以求解同一冷轧组批模型，参数设置完全一致。交货期差异度权重  $\alpha=0.5$ ，调整次数权重  $\beta=0.2$ ，工艺路线差异度  $\gamma=0.3$ ；原料重量限制  $P=2200t$ ；为方便起见，本文将每个批次的同一约束的最大值设定为相同值。因此，原料入口最大宽度差  $F_i=220mm$ ；产品出口最大宽度差  $F_i'=80mm$ ；原料入口厚度  $H_j$  的分界点  $a=3.2mm$ ，当  $H_j>3.2mm$  时原料入口最大厚度差  $Q_i=0.8$ ；当  $H_j\leq 3.2mm$  时原料入口最大厚度差  $Q_i'=0.5$ ；产品出口厚度  $H_j$  的分界点  $b=1.5mm$ ，当  $H_j>1.5mm$  时产品出口最大厚度差  $Y_i=0.6$ ；当  $H_j\leq 1.5mm$  时产品出口最大厚度差  $Y_i'=0.5$ ；抗拉强度最大比值  $Z=1.2$ 。

根据国内某冷轧企业的生产实际整理并截取了 45 组实验数据，受篇幅所限表 3.1 中只列出部分数据。表 3.1 中工艺路线编号和名称对应关系如表 3.2 所示。

表 3.1 相关的生产实际数据

订单号	钢种	抗拉 强度 (MPa)	入口 厚度 (mm)	出口 厚度 (mm)	入口 宽度 (mm)	出口 宽度 (mm)	重量 (kg)	工艺 路线	最早交货 时间	最晚交货 时间
1	SPCC	270.00	4.50	1.15	1310	1285	202720	1	8 月 19 日	8 月 22 日
2	M250P1	440.00	4.50	1.40	1345	1320	228400	1	8 月 18 日	8 月 21 日
3	M250P1	440.00	4.50	1.40	1345	1320	203760	1	8 月 18 日	8 月 21 日
4	M250P1	440.00	4.50	1.40	1345	1320	248380	1	8 月 18 日	8 月 21 日
5	M170P1	340.00	5.00	1.20	1286	1260	203130	1	8 月 20 日	8 月 23 日
6	M250P1	440.00	4.50	1.40	1345	1320	249370	1	8 月 18 日	8 月 21 日
7	M210P1	390.00	4.80	1.50	1345	1320	159040	4	8 月 18 日	8 月 21 日
8	SPCC	270.00	4.50	1.15	1310	1285	275990	1	8 月 19 日	8 月 22 日
9	M210P1	390.00	5.00	1.50	1291	1266	250200	5	8 月 19 日	8 月 22 日
10	M210P1	390.00	5.00	1.50	1291	1266	158240	5	8 月 19 日	8 月 22 日
11	SPCC	270.00	4.50	1.20	1330	1305	278190	5	8 月 19 日	8 月 22 日
12	M210P1	390.00	5.00	1.50	1291	1266	199760	5	8 月 19 日	8 月 22 日
...	...	...	...	...	...	...	...	...	...	...

表 3.2 工艺路线编号和名称对照表

工艺路线编号	工艺路线名称
1	酸洗→冷轧
2	酸洗→冷轧→脱脂→罩式退火→二次冷轧
3	酸洗→冷轧→脱脂→罩式退火→涂镀
4	酸洗→冷轧→连续退火→涂镀
5	酸洗→冷轧→脱脂→罩式退火→平整

### 3.3.2 实验结果及分析

为了验证本文所提出的模型和算法的有效性，分别采用两种算法进行测试与对比，结果见表 3.3。

表 3.3 两种算法的聚类结果

算法	聚类结果		
	批次号	批次	批次重量( <i>t</i> )
FCM 算法	1	{14,19,26,27}	939.000
	2	{3,6,7,9,10,12,32,35,36,38}	2082.400
	3	{2,4,17,24,28,31,34,37,39,41}	2131.600
	4	{1,8,11,15,16,18,25}	1635.030
	5	{5,13,20,21,22,23,29,30,33,40,42,43,44,45}	2786.540
本文算法	1	{20,21,22,23,34,39,41,42,43}	1789.880
	2	{2,3,4,6,7,17,28,36,38,40}	2198.810
	3	{1,8,11,15,16,18,19,26,27}	2190.820
	4	{5,9,10,12,13,24,31,33,35,37,45}	2195.850
	5	{14,25,29,30,32,44}	1199.210

聚类结果中每个批次内的重量、宽度、厚度和抗拉强度的约束情况如表 3.4 所示。

表 3.4 各批次的约束情况

算法	批次	与最大重量的	与最大入	与最大出	与最大入	与最大出	与最大抗
		差值 ( <i>kg</i> )	口宽度差 的差值 ( <i>mm</i> )	口宽度差 的差值 ( <i>mm</i> )	口厚度差 的差值 ( <i>mm</i> )	口厚度差 的差值 ( <i>mm</i> )	拉强度比 值的差值 ( <i>MPa</i> )
FCM 算法	1	1261.000	200.000	60.000	0.000	0.150	0.200
	2	117.600	155.000	15.000	0.300	0.200	0.072
	3	68.400	145.000	5.000	0.000	0.000	0.072
	4	564.970	200.000	60.000	0.300	0.250	0.200
	5	-586.540	169.000	30.000	-0.700	-0.100	-0.094
	平均值	502.993	175.000	35.000	0.150	0.150	0.136
	均方差	478.491	25.249	25.249	0.150	0.094	0.064
本文算法	1	410.120	165.000	25.000	0.300	0.200	0.072
	2	1.190	155.000	15.000	0.000	0.000	0.072
	3	9.180	185.000	45.000	0.000	0.150	0.200
	4	4.150	181.000	40.000	0.000	0.000	0.053
	5	1000.790	195.000	55.000	-0.200	0.000	-0.244
	平均值	106.160	171.500	31.250	0.075	0.088	0.099
	均方差	175.515	12.114	11.924	0.130	0.089	0.059

对于表 3.3 和表 3.4 中的结果作如下说明：

(1) 表 3.4 中的数值均为各属性的约束上限减去组批后的实际值所得，如 FCM 算法结果中批次 1 的“与最大重量差值”一项取值为“1261.000”，表示批次 1 的组批后重量(939.000*t*，见表 3.3)低于批次最大重量限制(2200.000*t*)为

1261.000t。正值(负值)表示低于(超出)约束上限的值,正值(负值,用斜体表示)是指该属性的取值满足(不满足)约束,正值或负值的绝对值越小表示结果越好。

(2) 两种算法在聚类过程中均指定聚类数目为4个,但最终均得到5个批次。这是由于两种算法中的第5个批次是由某些合同因不同程度的违反了约束条件(存在负值)而不能聚到前4个批次造成的,并不是由算法直接聚类所得,将这类合同组成一个新的批次5。因此,表3.4中均值和均方差由前4个批次计算得到。

对表3.3和表3.4分析得到如下结论:

(1) 表3.3中,FCM算法中5个批次包含的合同数分别为4、10、10、7、14;本文算法中5个批次包含的合同数分别为9、10、9、11、6。可见,FCM算法的结果中各批次内的合同数分布较为参差,而本文算法结果中各批次内合同数的分布更为均匀。

(2) 对比表3.4中前4个批次,本文算法得到的聚类结果的均值以及均方差均小于FCM算法,这表明本文算法所得聚类结果波动较小,比较稳定且接近最大约束,即在满足企业生产的各项工艺约束的同时,能够最大限度的利用现有资源。而FCM算法的结果中各属性的值分布较离散且与最大约束的差距较大。例如,本文算法的聚类结果中,批次2、3和4不仅满足宽度、厚度和抗拉强度的约束,而且重量均达到2190t以上,非常接近批次最大重量2200t的要求;FCM算法的聚类结果中,批次2和3的重量相对满意,但与本文算法还有一定差距,而且批次1、4与最大重量相差较大。

(3) 另外,两种算法所得到的前4个批次均满足约束(正值),而批次5均不完全满足约束。FCM算法对应的批次5合同数量较多,有4个属性不满足约束,而且数值的绝对值很大,其中,“与最大批次重量的差值”一项达到-586.54;而本文算法的第5批次合同数量远少于FCM算法,只有2个属性不满足约束,而且绝对值很小,表示所得的结果略有超出约束条件。本文算法批次5中包含了一些抗拉强度在270~390MPa范围内的合同,导致“与最大抗拉强度比值差”一项比值较大微劣于FCM算法。但是在控制各批次内各属性满足约束条件的能力方面,本文算法明显优于FCM算法。

两种聚类算法所得结果对应的各项优化子目标对比情况如表3.5所示。

表3.5 两种算法的聚类结果各优化子目标对比

优化目标	求解算法	批次					平均 值	均方 差
		1	2	3	4	5		
交货期差异度	FCM 算法	0.588	0.588	1.176	0.588	1.176	0.735	0.2546
	本文算法	1.176	0.588	0.588	0.588	0.588	0.735	0.2546
工艺路线差异度	FCM 算法	0.433	0.418	0.418	0.371	0.349	0.410	0.0233
	本文算法	0.350	0.380	0.422	0.393	0.280	0.386	0.0259
单次调整对应的 合同数(合同数/ 调整次数)	FCM 算法	4/4=1.000	10/7=1.429	10/8=1.250	7/3=2.330	14/8=1.750	1.502	0.5016
	本文算法	9/5=1.800	10/7=1.429	9/5=1.800	11/6=1.830	6/5=1.200	1.715	0.1654

表 3.5 中的数据有以下说明:

(1) 均值和均方差是根据前 4 个批次的聚类结果计算得到。

(2) 由于不同算法所得结果中各批次的合同数量不同,因此不宜将调整次数直接作为评价指标。故本文将每个批次中“单次调整对应的合同数”作为评价指标,显然该值越大说明生产相同数量的合同所需的调整次数越少,则优化效果会更好。

对表 3.5 分析,可得如下结论:

(1) 交货期差异度方面,两种算法的前 4 个批次的均值和均方差都相同,但从整体来看,FCM 算法的聚类结果中交货期差异度有 2 个批次为 1.176,3 个批次为 0.588;本文算法中 4 个批次均为 0.588,仅有第 1 个批次为 1.176,其原因可能是该批次内的合同数量较多,导致交货期差别较大。故本文算法在交货期方面的整体优化效果优于 FCM 算法。

(2) 工艺路线差异度方面,本文算法聚类结果中前 4 个批次的均值小于 FCM 算法,均方差略大于 FCM 算法,但差距极小。说明本文算法所得结果批次内的工艺路线差异度整体较小,批次间的工艺路线差异度稍大。整体来看,本文算法中仅有一个批次的工艺路线差异度高于 0.4,且第 5 批次不足 0.3;FCM 算法中有 3 个批次高于 0.4,最小的差异度也高于 0.3;故本文算法对工艺路线差异度目标的优化效果略优于 FCM 算法。

(3) 单次调整对应的合同数方面,本文算法前 4 个批次的均值大于 FCM 算法,均方差明显小于 FCM 算法,说明本文算法单次调整对应的合同数较多,且聚类结果较稳定。FCM 算法中批次 4 的值较大的原因是该批次的合同数量较少,且合同规格差距很小,调整次数相对较少,导致比值较大(前面已经描述该结果在满足约束方面较差);而本文算法前 4 个批次聚类效果较好,剩余的合同数量较少,故批次 5 的值小于 FCM 算法;因此,本文算法对调整次数的优化效果明显优于 FCM 算法。

由上述分析可以看出,本文算法在进行合同组批时整体聚类结果较优且更符合企业的实际生产情况,更有利于解决市场的多品种小批量和企业大批量生产的矛盾。

最后,本文遵循冷轧企业实际生产时同批次内原料保持渐窄的原则,确定了批次内合同的生产顺序,见表 3.6。批次 1 中的 20(1325)表示合同编号 20 对应的原料入口宽度为 1325mm。

表 3.6 各批次内合同生产顺序

批次号	合同生产顺序
1	20(1325)→22(1325)→23(1315)→39(1280)→21(1275)→42(1275)→43(1275) →34(1270)→41(1270)
2	2(1345)→3(1345)→4(1345)→6(1345)→7(1345)→28(1310)→17(1291) →36(1285)→38(1280)→40(1280)
3	26(1345)→27(1335)→11(1330)→15(1330)→16(1330)→18(1325)→19(1325) →1(1310.00)→8(1310.00)
4	24(1315)→35(1305)→31(1305)→9(1291)→10(1291)→12(1291)→33(1286)→ 5(1286)→13(1286)→37(1285)→45(1276)
5	14(1330)→44(1326)→25(1310)→29(1306)→30(1306)→32(1305)

### 3.4 本章小结

为了解决大批量生产和市场需求个性化之间的矛盾,本章建立了冷轧组批计划模型。目标函数中引入了合同交货期差异度和工艺路线差异度的概念,并分别提出了可行的计算方法。约束中加入了更加符合冷轧生产合同组批实际要求的条件,如批次重量、出(入)口宽度、出(入)口厚度和抗拉强度等。构建了基于粒子群优化的模糊聚类算法对模型进行优化求解,该算法集成了粒子群算法和 FCM 算法的优点,因此整体优化结果要优于 FCM 算法。采用某冷轧企业实际生产数据对模型和算法的验证结果表明,该方法在满足约束条件的能力以及各子目标优化等方面较一般聚类算法有较强的优势,所得结果更满足企业的实际需求。



## 4 考虑多阶段多流向的冷轧生产计划

目前国内外关于冷轧生产计划的研究大多只是在满足生产工艺的前提下,以交货期或库存单方面为目标建立生产计划与调度模型。

文献[24]研究了冷轧全流程生产计划编制问题,其建立的数学模型的目标函数如下:

$$\min \lambda_1 \sum_{i=1}^{|N|} \max \{c_i |J^i| - d_{i|J^i|}, 0\} + \lambda_2 \sum_{m=1}^{|M|} T_m + \lambda_3 \sum_{i=1}^{|N|} \sum_{j=1}^{|J^i|} (s_{ij} - r_{ij})$$

其中,第一项为交货期提前或拖期惩罚,第二项为调整费用惩罚,第三项为库存占用惩罚。虽然目标函数考虑了交货期和中间库存,但库存控制是以物料等待加工时间的惩罚费用为目标,即  $s_{ij} - r_{ij}$  为物料在中间库等待加工的时间,而且没有考虑成品库库存成本和多阶段库存问题。

文献[20]将冷轧生产计划与调度问题抽象为多阶段、多品种并带有中间库的批量计划与调度问题,其建立的冷轧生产计划数学模型中的目标函数如下:

$$\max T \sum_{i=1}^N \sum_{k \in K'(i)} \sum_{t=1}^T h_{ik} I_{ikt} + U \sum_{i=1}^N \sum_{\substack{j=1 \\ i \neq j}}^N \sum_{k \in K'(i)} a_{ijk} Z_{ijk}$$

其中,第一项为中间库存保存费用,第二项为机组调整费用。

其约束中的库存平衡公式为:

$$I_{ik(t-1)} + Y_{i(k-1)t} v_{i(k-1)} - Y_{i,k+1,t} v_{i,k+1} = I_{ikt}$$

式中右边为  $t$  时段结束,物料  $i$  在库房  $k$  的库存量。左边第一项为  $t-1$  时段结束,物料  $i$  在库房  $k$  的库存量;第二项为前一机组产出的量;第三项为后一机组消耗的量。该库存平衡公式并未体现出生产批次多流向的特点,其库存平衡公式中  $Y_{i(k-1)t}$  与  $Y_{i(k+1)t}$  分别表示  $t$  时段物料  $i$  在机组  $k-1$  与  $k+1$  上加工。 $v_{i(k-1)}$  与  $v_{i(k+1)}$  分别表示物料  $i$  在机组  $k-1$  与  $k+1$  上的加工速率。可知,公式中只提到了机组的加工速率,并没有乘以机组的加工时间,所建立的库存平衡公式并不平衡;而且,优化目标仅考虑了中间库存,没有考虑成品库的库存成本。

本章系统分析了冷轧生产计划的制定过程,在综合考虑合同交货期、中间库存量、成品库成本等目标的基础上,建立满足冷轧生产工艺、机组产能、成材率与库存平衡等约束的冷轧生产计划数学模型。模型重点考虑了冷轧生产系统的多阶段、多流向问题。所谓多阶段是指生产系统具有中间库、成品库等多个阶段的库存,多流向是指同一批次存在多种流向(可视为工艺路线);进而采用改进的遗传算法对模型进行求解;最后以某钢铁企业的冷轧薄板厂的生产实际为案例,对所提出的模型和算法进行验证。

## 4.1 冷轧生产计划问题描述

### 4.1.1 冷轧生产工艺路线

一般冷轧生产工艺流程如图 4.1 所示。

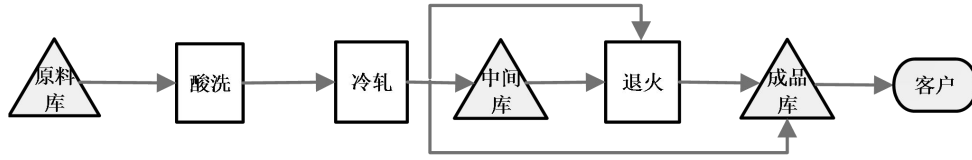


图 4.1 冷轧生产工艺流程图

由图 4.1 可知冷轧生产系统存在多阶段的库存,如中间库(可视情况自主设置多个)和成品库。为了适应冷轧企业多品种、小批量订单共线生产的模式,机组多存在切换、调整时间。同时,系统中包括多个批次流向,例如冷轧后的半成品可经中间库后退火或直接进行退火,产成品可直接进入成品库。

因此,一般冷轧生产系统是一个复杂的混合型系统,具有多阶段,多流向,多品种,多切换等特征。本文针对这些特点,将冷轧生产系统的生产计划问题抽象为多阶段、多流向的数学规划模型。

### 4.1.2 相关数学符号定义

设  $i$  表示批次数,  $i=1,2,\dots,I$ 。  $t$  为时段编号,  $t=1,2,\dots,T$ ,  $L_t$  为  $t$  时段的长度;  $\rho$  为批次成材率; 为方便描述, 本文将中间库、成品库以及机组统称为设备, 设  $k$  为工艺路线中的设备号,  $k=1,2,\dots,Z$ ,  $k=Z$  时表示成品库;  $K'$  表示机组集合,  $|K'|$  表示机组个数,  $K''$  表示库房集合 (包括中间库和成品库),  $|K''|$  表示库房个数,  $K$  表示所有设备集合, 满足:  $K' \cap K'' = \emptyset, K' \cup K'' = K$ 。  $j$  表示批次流向编号,  $A_{in,i,k}$  表示批次  $i$  流入库房  $k$  的流向集合,  $|A_{in,i,k}|$  表示批次  $i$  流入库房  $k$  的流向个数,  $A_{out,i,k}$  表示批次  $i$  流出库房  $k$  的流向集合,  $|A_{out,i,k}|$  表示批次  $i$  流出库房  $k$  的流向个数,  $k \in K''$ ,  $k \neq Z$ 。

设  $x_i$  为批次  $i$  的开始加工时间,  $T_{i,k,t}$  为  $t$  时段批次  $i$  在  $k$  机组上的加工时间,  $T_{i,k}'$  为批次  $i$  在  $k$  机组上的调整时间,  $\lambda_{i,k}$  表示批次  $i$  是否在机组  $k$  上调整,  $d_i$  为批次  $i$  的交货期。 设  $V_{i,k,t}$  为  $t$  时段批次  $i$  在机组  $k$  上的平均加工速率,  $k \in K'$ ;  $V_{i,k,j,t}$  为  $t$  时段批次  $i$  在  $j$  流向上流向库房  $k$  的速率,  $k \in K''$ ,  $k \neq Z$ 。  $C_{i,k,t}$  为  $t$  时段批次  $i$  在  $k$  机组下的产量,  $k \in K'$ ;  $R_i$  为批次  $i$  的供应量,  $G_i$  为批次  $i$  对应产品的产量。 设  $MaxU_k$ ,  $MinU_k$  分别为中间库  $k$  的正常安全库存的上、下限,  $H_{i,k,t}$  为  $t$  时段批次  $i$  在库房  $k$  中的库存量,  $k \in K''$ ,  $k \neq Z$ 。  $\alpha$  为合同超期单位时间惩罚费用,  $\beta$  为合同提前单位时间惩罚费用,  $h_1$  为超出中间库库存上限时单位时间单位重量的惩罚费用,  $h_2$  为低于中间库库存下限时单位时间单位重量的惩罚费用,  $h_3$  为成品库单

位时间单位重量的保存成本,  $F$  为总费用, 包括合同交货期惩罚费用、中间库存量惩罚费用以及成品库库存成本三部分。 $\omega_1$ 、 $\omega_2$ 、 $\omega_3$ 、 $\omega_4$  和  $\omega_5$  分别为合同超期惩罚费用、合同提前惩罚费用、中间库存量超出惩罚费用、中间库存量不足惩罚费用以及成品库库存成本的权重系数, 并满足  $\omega_1 + \omega_2 + \omega_3 + \omega_4 + \omega_5 = 1$ 。

#### 4.1.3 冷轧生产计划数学模型

为了便于描述问题, 给出了如下基本假设:

假设①: 优化目标建立在批次已确定的基础上;

假设②: 成品库库容要求未纳入考虑范畴;

假设③: 中间库持有成本未纳入考虑范畴;

假设④: 机组计划维修情况未纳入考虑范畴。

本文将在满足轧制过程工艺约束的同时, 以合同的超期和提前、中间库库存超出和不足总惩罚费用最小并确保成品库库存成本最低为优化目标。基于以上分析, 考虑多阶段、多流向的冷轧生产计划模型建立如下:

决策变量:  $x_i$  为批次  $i$  的开始加工时间。

目标函数:

$$\begin{aligned} \min F = & \omega_1 \cdot \left\{ \sum_{i=1}^I \alpha \cdot \max \left\{ 0, \left( x_i + \sum_{k=1}^{|K'|} \left( \sum_{t=1}^T T_{i,k,t} + \lambda_{i,k} \cdot T'_{i,k} \right) \right) - d_i \right\} \right\} + \omega_2 \cdot \left\{ \sum_{i=1}^I \beta \cdot \max \left\{ 0, d_i - \left( x_i + \sum_{k=1}^{|K'|} \left( \sum_{t=1}^T T_{i,k,t} + \lambda_{i,k} \cdot T'_{i,k} \right) \right) \right\} \right\} \\ & + \omega_3 \cdot \left\{ \sum_{k=1}^{|K''|} \sum_{t=1}^T h_1 \cdot \max \left\{ 0, \sum_{i=1}^I H_{i,k,t} - \text{Max}U_k \right\} \right\} + \omega_4 \cdot \left\{ \sum_{k=1}^{|K''|} \sum_{t=1}^T h_2 \cdot \max \left\{ 0, \text{Min}U_k - \sum_{i=1}^I H_{i,k,t} \right\} \right\} + \omega_5 \cdot \left\{ \sum_{k=Z}^{|K''|} \sum_{t=1}^T \sum_{i=1}^I h_3 \cdot H_{i,k,t} \right\}. \end{aligned} \quad (4.1)$$

约束条件:

$$x_{i+1} \geq x_i + \sum_{t=1}^T T_{i,k,t} + \lambda_{i,k} \cdot T'_{i,k}, \quad i = 1, 2, \dots, I-1, \quad t = 1, \quad k = 1, \quad (4.2)$$

$$G_i \leq \rho \cdot R_i, \quad i = 1, 2, \dots, I, \quad (4.3)$$

$$H_{i,k,(t-1)} + \frac{|A_{in,i,k}|}{\sum_{j=1}^{|K''|} V_{i,k,j,t} L_t} - \frac{|A_{out,i,k}|}{\sum_{j=1}^{|K''|} V_{i,k,j,t} L_t} = H_{i,k,t}, \quad i = 1, 2, \dots, I, \quad t = 1, 2, \dots, T, \quad k \in K'', k \neq Z, \quad (4.4)$$

$$V_{i,k,t} \cdot T_{i,k,t} \leq C_{i,k,t}, \quad i = 1, 2, \dots, I, \quad t = 1, 2, \dots, T, \quad k \in K', \quad (4.5)$$

$$x_i \geq 0, \quad i = 1, 2, \dots, I, \quad (4.6)$$

$$\lambda_{i,k} = \begin{cases} 1, & \text{批次 } i \text{ 在机组 } k \text{ 上调整}, i = 1, 2, \dots, I, k \in K' \\ 0, & \text{其它} \end{cases} \quad (4.7)$$

目标函数式(4.1)左边为总费用, 右边第 1 项为合同超期惩罚费用, 即所有时段内所有批次的开始加工时间、加工时间以及调整时间之和与交货期差值的惩罚; 第 2 项为合同提前惩罚费用, 即所有时段内所有批次的交货期与开始加工时间、加工时间以及调整时间差值的惩罚; 第 3 项为中间库存量超出惩罚费用, 即

所有时段内所有批次在中间库  $k$  的库存总量与中间库  $k$  的上限差值的惩罚；第 4 项为中间库存量不足惩罚费用，即中间库  $k$  的下限与所有时段内所有批次在中间库  $k$  的库存总量差值的惩罚；第 5 项为成品库库存成本费用，即所有时段内所有批次在成品库  $Z$  的库存量保存费用。约束(4.2)表示各批次之间的加工顺序约束，即为保证机组连续生产，下一批次的加工开始时间应大于当前批次在第 1 时段第 1 机组的完工时间与调整时间之和。约束(4.3)为原料约束，即批次  $i$  对应产品的产量应不大于批次  $i$  的供应量。约束(4.4)为满足多种批次流向的库存平衡公式，式中左边第 1 项为上一时段末批次  $i$  在库房  $k$  的库存量，第 2 项为当前时段批次  $i$  在  $j$  流向上流入库房  $k$  的量，第 3 项为当前时段批次  $i$  在  $j$  流向上流出库房  $k$  的量，式中右边为当前时段批次  $i$  在库房  $k$  的库存量。约束(4.5)保证生产量在机组产能范围之内。约束(4.6)表示批次  $i$  的开始时间约束；约束(4.7)表示批次  $i$  是否在机组  $k$  上调整。

## 4.2 基于改进遗传算法的冷轧计划模型的求解

遗传算法于 1975 年被首次提出，创始人是美国 Michigan 大学的 John Holland 教授，它是一种随机搜索的算法，是在遗传机理与生物自然选择的基础上发展而来的算法，能够很好的解决最优问题。其基本思想是：首先产生初始种群，然后对那些选中的染色体进行交叉、变异等操作后产生后代，通过计算适应度值来选择符合条件的后代，对于不符合条件的后代则进行淘汰。经过若干代遗传后，该算法会最终收敛于条件最好的染色体，也就是得到最优解。

一般遗传算法实现步骤如下图 4.2 所示。

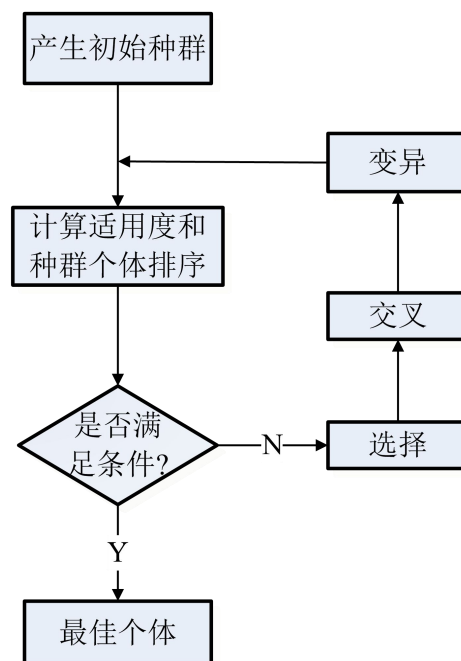


图 4.2 遗传算法流程图

#### 4.2.1 交叉、变异概率的自适应调整策略

遗传算法已经被广泛地应用于各种优化问题的求解中<sup>[51-53]</sup>。随着问题规模的不断扩大和模型复杂度的提高，传统遗传算法的不足之处主要表现在：(1)相关参数的设置十分重要，而传统的遗传算法在每一代进化时都采用相同的交叉和突变概率，这使得在进化的后期，优良个体的基因可能会丢失、优良性状可能无法遗传给子代；(2)当处理多峰复杂函数的优化问题，传统遗传算法常常收敛于局部最优解。

针对传统遗传算法的不足，涌现出了一些改进的遗传算法。文献[51]提出了一种改进的自适应遗传算法，交叉和突变概率随着种群的进化过程不断发生变化，力求达到全局最优。其交叉和突变概率公式如下：

$$P_c = \begin{cases} P_{c1} * \frac{1}{(P_{c1} - P_{c2}) + \exp\left(\frac{f' - f_{avg}}{f_{max} - f_{avg}}\right)}, f' \geq f_{avg} \\ k_1 * P_{c1}, f' < f_{avg} \end{cases} \quad (4.8)$$

$$P_m = \begin{cases} P_{m1} * \frac{1}{(P_{m1} - P_{m2}) + \exp\left(\frac{f - f_{avg}}{f_{max} - f_{avg}}\right)}, f \geq f_{avg} \\ k_2 * P_{m1}, f < f_{avg} \end{cases} \quad (4.9)$$

其中， $f_{avg}$  是种群中所有个体的平均适应度值， $f_{max}$  是种群中最大个体适应度值， $f'$  是两个要交叉个体中适应度值较大个体的适应度值， $f$  是要变异个体的适应度值。 $k_1, k_2, P_{c1}, P_{c2} \in [0.5, 1]$ ,  $P_{m1}, P_{m2} \in [0, 0.1]$ ，且有  $P_{c1} > P_{c2}$ ,  $P_{m1} > P_{m2}$ 。

上述改进的交叉和突变概率公式主要存在以下问题：

(1) 当分母太小或者为 0 时会造成计算错误，出现一些无意义的值。

(2) 进化过程中，上述公式并非总是让交叉和突变概率自适应地调整。例如，公式(4.8)和(4.9)中，对于小于平均适应度值的个体，交叉和突变概率总取一个固定的值，不能做到完全的自适应调整。

为了改善以上问题，本文在其基础上提出了改进的交叉和突变概率公式：

$$P_c = \begin{cases} P_{c2} * \frac{1}{(P_{c1} - P_{c2}) + \exp\left(A * \left(\frac{f_{avg} - f'}{f_{avg} - f_{min} + r}\right)\right)}, f' < f_{avg} & (4.10-1) \\ P_{c1} * \frac{1}{(P_{c1} - P_{c2}) + \exp\left(A * \left(\frac{f_{max} - f'}{f_{max} - f_{avg} + r}\right)\right)}, f' \geq f_{avg} & (4.10-2) \end{cases}$$

$$P_m = \begin{cases} P_{m2} * \frac{1}{(P_{m1} - P_{m2}) + \exp\left(B * \left(\frac{f_{avg} - f}{f_{avg} - f_{min} + r}\right)\right)}, & f < f_{avg} \\ P_{m1} * \frac{1}{(P_{m1} - P_{m2}) + \exp\left(B * \left(\frac{f_{max} - f}{f_{max} - f_{avg} + r}\right)\right)}, & f \geq f_{avg} \end{cases} \quad (4.11-1)$$

改进后的交叉、变异概率公式有以下特点：

(1) 当模型求最小值时，交叉和突变概率能够随着适应度函数的增大而逐渐增大；

(2) 在公式(4.8)和(4.9)的分母中加入了一个很小的正数  $r$ ，解决了文献[51]中两种概率公式会出现无意义值的问题；

(3) 通过设定常数  $A$ 、 $B$ ，使分段函数在断点处的函数值差值尽可能小，两侧取值能够平滑的过渡，解决了文献[51]中交叉、变异概率公式在断点处的函数值跳跃大的问题；

(4) 通过改进交叉和变异概率公式，解决了文献[51]中当  $f'$  和  $f$  小于  $f_{avg}$  时，两种概率为固定值的问题，能够实现种群进化过程中两种概率的自适应调整。

#### 4.2.2 改进遗传算法的求解步骤

改进遗传算法的步骤设计如下：

步骤 1：初始化种群

染色体采用整数编码方式，按照问题的批次规模  $N$  生成  $1 \sim N$  的随机序列作为一个个体。依次产生种群大小  $Q$  个个体并形成初始种群。

步骤 2：适应度值计算

将模型中的目标函数(4.1)作为种群的适应度函数，计算种群中每个个体的适应度值。

步骤 3：执行选择操作

采用轮盘赌选择和精英替换混合策略。首先，根据步骤 2 计算的适应值，使用轮盘赌选择方法得到一个种群规模相同的子代种群；然后，从父代种群中选择出最优的一部分个体放入精英库中，同时在子代中选出同等数量的最差个体；最后，用精英库中的父代精英个体替换子代的最差个体，得到选择之后的最终子代种群。

步骤 4：执行交叉操作

将步骤 3 得到的子代种群作为交叉操作的父代，从中随机选择两个个体，设其中的最小适应值为  $f'$ 。若  $f' < f_{avg}$ ，交叉概率根据公式(4.10-1)计算，否则根据公式(4.10-2)计算。然后根据所获得的交叉概率执行交叉操作。重复步骤 4 直到

得到一个相同规模的新种群。

#### 步骤 5: 执行突变操作

从获得的新种群中选择一个个体作为突变操作的对象, 设该个体的适应值为  $f$ 。若  $f < f_{avg}$ , 突变概率根据公式(4.11-1)计算, 否则根据公式(4.11-2)计算。然后根据所获得突变概率执行突变操作。重复执行步骤 5 直到得到一个相同规模的新种群。

#### 步骤 6: 算法终止条件判断

若此时种群已经达到指定的遗传代数或者精度, 则执行步骤 7, 否则执行步骤 2。

#### 步骤 7: 确定最优解

计算最终子代种群中每个个体的适应度值, 然后选择适应值最小的个体作为模型的最优解, 其编码序列就是对应的批次生产加工顺序。

#### 步骤 8: 输出各批次加工开始时间

根据步骤 7 得到的各批次的加工顺序以及在机组上的加工时间和约束条件(4.2), 计算所有批次的最早开始加工时间并输出, 即对应的决策变量  $x_i$  的取值。

### 4.3 应用实例

#### 4.3.1 实验环境和数据

为了方便对比, 文献[51]的算法称为“改进前算法”, 本文提出的算法称为“改进后算法”。改进前后算法均采用 Java 语言在 PC Corei5/CPU2.5GHz/RAM4.0GB 计算机上编程实现, 开发工具为 myEclipse8.5, 数据库为 MySQL5.6, 应用服务器为 Tomcat8.0。在查阅并借鉴相关文献以及经过大量实验的基础上, 确定了算法的基本参数。

实验参数: (1)改进前算法的参数:  $P_{c1}=0.9$ ;  $P_{c2}=0.8$ ;  $P_{m1}=0.1$ ;  $P_{m2}=0.05$ ;  $k_1=0.35$ ;  $k_2=0.30$ ; 种群规模  $Q$  为 50; 遗传代数为 70。(2)改进后算法的参数: 为了便于对比,  $P_{c1}$ 、 $P_{c2}$ 、 $P_{m1}$ 、 $P_{m2}$  以及种群规模、遗传代数与改进前算法的参数设置相同; 另外,  $A=0.13$ ;  $B=0.7$ ;  $r=0.000001$ 。其他参数设置为: 中间库存下限  $MinU_k$  为 0, 上限  $MaxU_k$  为 16000 吨; 合同超期惩罚费用  $\alpha$  为 250 元/天, 提前惩罚费用  $\beta$  为 50 元/天; 超出库存上限的惩罚费用  $h_1$  为 40 元/(天·吨), 低于库存下限的惩罚费用  $h_2$  为 150 元/(天·吨), 库存保存成本  $h_3$  为 40 元/(天·吨), 调整时间为 1.5 分钟/次; 权重系数  $\omega_1=0.50$ ,  $\omega_2=0.10$ ,  $\omega_3=0.10$ ,  $\omega_4=0.25$ ,  $\omega_5=0.05$ 。由于该企业采用面向订单生产的模式, 因此合同超期的惩罚费用和权重系数设置较高。管理和决策者可通过权重系数的调节来实现各子目标之间的协调。

实验数据是根据国内某大型钢铁公司冷轧薄板厂的实际生产数据整理而来, 从中截取了 80 组数据, 受篇幅所限本文只列出部分数据, 见表 4.1。



表 4.1 相关的生产实际数据

批次号	订单号	钢号	入口厚度 (mm)	出口厚度 (mm)	重量 (kg)	是否连退	交货期 (d)
1	3001022485	SPCC	4.8	1.3	25380	是	1
2	3001020434	M250P1	4.5	1.4	22840	是	2
3	3001020434	M250P1	4.5	1.4	22640	是	2
4	3001020434	M250P1	4.5	1.4	22580	是	2
5	3001020434	M250P1	4.5	1.4	22570	是	2
6	3001020434	M250P1	4.5	1.4	22670	是	2
7	3001022911	M210P1	4.8	1.5	22720	是	1
8	3001022677	SPCC	4.5	1.2	22560	是	1
9	3001020515	SPCC	4.5	1.2	25020	是	1
10	3001020515	SPCC	4.5	1.2	19780	否	1
11	3001020515	SPCC	4.5	1.2	25290	是	1
12	3001020515	SPCC	4.5	1.2	24970	是	1
...	...	...	...	...	...	...	...

#### 4.3.2 实验结果及分析

为了验证冷轧生产计划模型和改进后算法的有效性,根据表 4.1 中的数据依次设定了 40( $P_1$ )、60( $P_2$ )和 80( $P_3$ )个批次的排产问题进行测试。改进前后两种算法的性能及各优化目标的对比结果见表 4.2。表 4.2 中的“交货期惩罚”表项对应模型中合同交货期超期和提前惩罚值之和,“中间库存惩罚”对应模型中中间库存超出和不足惩罚值之和。

表 4.2 两种遗传算法的性能及各优化目标对比

问题	改进前算法					改进后算法				
	运行时间 (s)	最优值	交货期 惩罚	中间库 惩罚	成品库存 惩罚	运行时间 (s)	最优值	交货期 惩罚	中间库 惩罚	成品库存 惩罚
$P_1$	3.649	771	72.0	0.13	698.4	3.278	760	72.0	0.13	687.7
$P_2$	5.391	1238	64.8	0.13	1173.2	4.891	1192	64.8	0.13	1126.6
$P_3$	7.453	2083	741.6	0.11	1241.6	7.299	1824	453.6	0.15	1367.0

对表 4.2 的实验结果进行分析,得到如下结论:

(1) 遗传算法在解决 NP-hard 问题时具有良好的计算能力。当问题达到一定规模(如 80 个批次)时也能在 8s 之内给出结果,明显优于人工编制计划,完全能够满足实际需求,有利于减轻计划人员的工作强度。

(2) 改进后算法在寻找最优值方面明显优于改进前算法,且问题规模越大其优势体现的越明显,同时在时间性能上也优于改进前算法。

(3) 问题  $P_3$  中,改进后算法的交货期惩罚值明显低于改进前算法,说明目标函数更侧重于交货期的控制,这与目标函数(4.1)中交货期惩罚的权重 $\omega_1$ 的设置较大相吻合。

图 4.3 和图 4.4 分别为针对问题  $P_1$  求解所得的最优适应值和平均适应值的变

化曲线。其中蓝色曲线代表改进前算法，红色曲线为改进后算法。

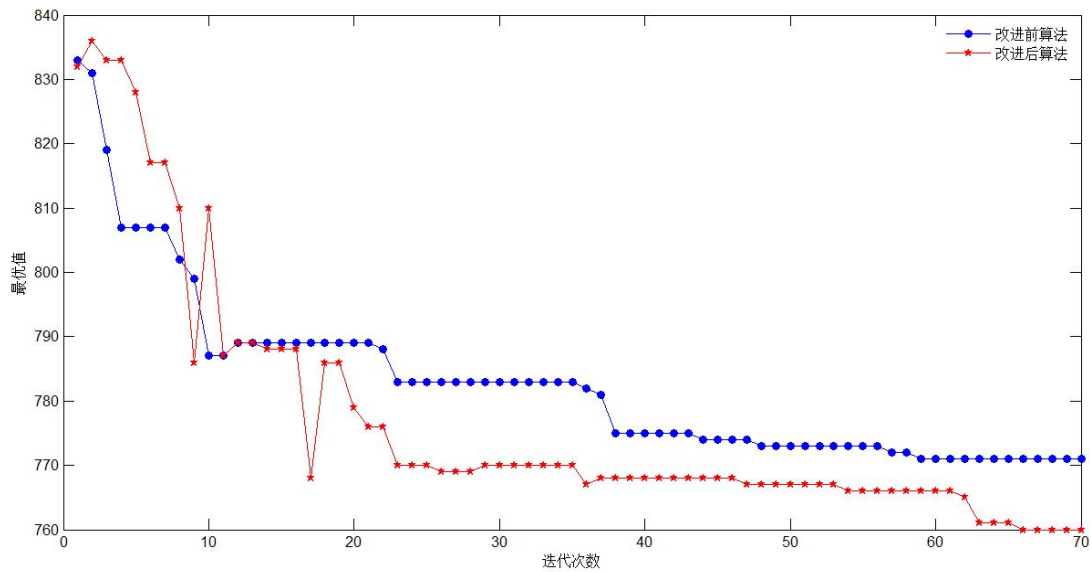


图 4.3 两种遗传算法的最优适应值变化曲线

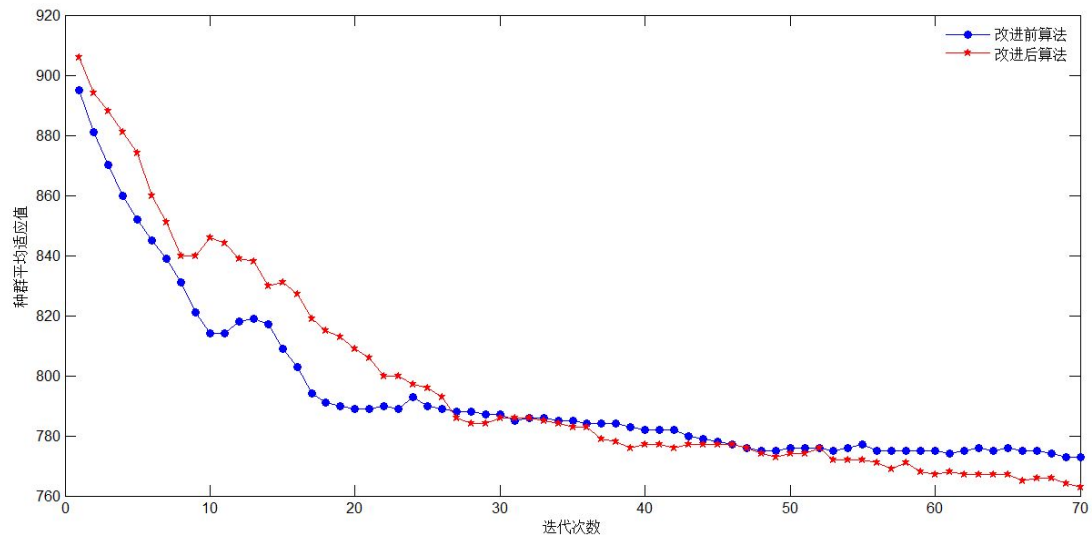


图 4.4 两种遗传算法的平均适应值变化曲线

从图 4.3 和图 4.4 可以看出，随着迭代次数的增加，改进前后算法中种群的最优(平均)适应值逐渐降低。当迭代次数较少时，改进前算法的最优值稍优于改进后算法；当迭代次数达到一定代数后，改进后算法的最优值明显优于改进前算法，同时种群的平均适应值也小于改进前算法。这是由于在种群进化初期，改进前算法中适应值较大个体(较差值)对应的交叉和突变概率大于改进后算法，种群更易于产生新的优良个体，故算法的收敛速度优于改进后算法；但是随着迭代次数的增加，改进后算法中当个体适应值较小(较优值)时，交叉和突变概率也能自适应调整，且小于改进前算法始终保持的一个固定值，改进后算法更能使最优个体保持稳定，故改进后算法的最优值和平均适应值优于改进前算法。

表 4.3 是问题  $P_1$  两种遗传算法的排产结果对比表，批次号与表 4.1 是一致的。排产后各批次的实际的加工顺序可根据表 3 中各批次的加工时间得出。从表 4.3

中改进后算法的排产表可以看出各批次是否经过了调辊或退火工序。例如批次 1 的调辊开始时间为  $28.0min$ ，结束时间为  $29.5min$ ，说明该批次经过了调辊且时间为  $1.5min$ ；批次 2 的调辊开始和结束时间均为  $249.5min$ ，说明该批次没有经过调辊。图 4.5 是根据表 4.3 中改进后算法的排产结果得到的甘特图。纵坐标表示批次，横坐标表示相应批次的加工时间；红色、蓝色和绿色图标依次表示冷轧、退火和调辊时间。由表 4.3 和图 4.5 可以看出，各批次在机组上的加工时间是连续的，没有间断生产且各批次都经过了冷轧工序。例如第 2 顺序生产的批次号为 32 的调辊开始和结束时间均为  $7.0min$ ；冷轧开始时间为  $7.0min$ ，结束时间为  $13.0min$ ；连退开始时间为  $52.6min$ ，结束时间为  $95.4min$ 。第 3 顺序生产的批次号为 22 的调辊开始时间为  $13.0min$ （批次 32 的冷轧结束时间），结束时间为  $14.5min$ ；冷轧开始时间为  $14.5min$ ，结束时间为  $20.5min$ ；连退开始时间为  $95.4min$ （批次 32 的连退结束时间），连退结束时间为  $138.2min$ 。

表 4.3 两种遗传算法的排产结果对比表(问题 P<sub>1</sub>)

批次号	改进前算法						改进后算法					
	调辊		冷轧		退火		调辊		冷轧		退火	
	开始	结束	开始	结束	开始	结束	开始	结束	开始	结束	开始	结束
	时间 (min)	时间 (min)	时间 (min)	时间 (min)	时间 (min)	时间 (min)	时间 (min)	时间 (min)	时间 (min)	时间 (min)	时间 (min)	时间 (min)
1	156.0	157.5	157.5	163.5	893.8	945.0	28.0	29.5	29.5	35.5	189.4	240.6
2	85.5	85.5	85.5	92.5	508.6	554.2	249.5	249.5	249.5	256.5	1279	1324.6
3	108.5	108.5	108.5	115.5	637.0	679.8	122.5	122.5	122.5	129.5	711.4	754.2
4	278.0	278.0	278.0	284.0	1284.6	1324.6	145.5	145.5	145.5	151.5	834.2	874.2
5	163.5	165.0	165.0	173.0	945.0	985.0	129.5	129.5	129.5	137.5	754.2	794.2
6	131.5	133.0	133.0	141.0	765.4	808.2	220.5	220.5	220.5	228.5	1147.8	1190.6
7	173.0	174.5	174.5	181.5	985.0	1030.6	0.0	0.0	0.0	7.0	7.0	52.6
8	13.0	13.0	13.0	21.0	103.8	143.8	44.0	44.0	44.0	52.0	286.2	326.2
9	149.0	149.0	149.0	156.0	848.2	893.8	181.5	181.5	181.5	188.5	1011.0	1056.6
10	255.0	255.0	255.0	263.0	1199.0	1199.0	206.5	206.5	206.5	214.5	1096.6	1096.6
11	53.0	54.5	54.5	60.5	326.2	377.4	20.5	22.0	22.0	28.0	138.2	189.4
12	212.5	214.0	214.0	221.0	1113.4	1159.0	242.5	242.5	242.5	249.5	1233.4	1279.0
13	78.5	78.5	78.5	85.5	463.0	508.6	90.5	92.0	92.0	99.0	537.4	583.0
14	101.0	102.5	102.5	108.5	594.2	637.0	115.0	116.5	116.5	122.5	668.6	711.4
15	245.5	247.0	247.0	255.0	1199.0	1199.0	173.5	173.5	173.5	181.5	1011.0	1011.0
16	0.0	0.0	0.0	7.0	7.0	52.6	166.5	166.5	166.5	173.5	965.4	1011.0
17	263.0	263.0	263.0	271.0	1199.0	1239.0	137.5	137.5	137.5	145.5	794.2	834.2
18	284.0	284.0	284.0	290.0	1324.6	1324.6	256.5	256.5	256.5	262.5	1324.6	1324.6
19	230.5	232.0	232.0	238.0	1159.0	1159.0	280.5	282.0	282.0	288.0	1324.6	1324.6
20	271.0	271.0	271.0	278.0	1239.0	1284.6	35.5	37.0	37.0	44.0	240.6	286.2
...	...	...	...	...	...	...	...	...	...	...	...	...

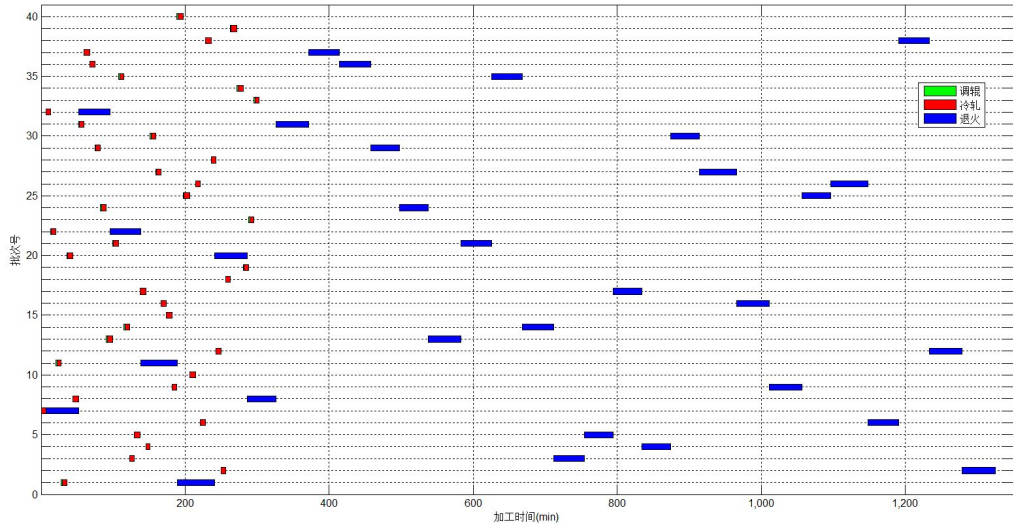


图 4.5 冷轧批次排产甘特图

#### 4.4 本章小结

本章在分析、总结冷轧生产系统特点的基础上，综合了交货期和库存控制等目标，建立了多阶段库存、多批次流向(重点体现在改进的库存平衡公式)的冷轧生产计划模型；改进的遗传算法通过设定常数  $A$ 、 $B$  和  $r$ ，改善了分段函数(交叉、变异概率公式)在一定区间内取固定值和断点处的函数值跳跃大以及出现不合理值的问题，能够根据个体适应值得出相应的交叉和突变概率，以实现种群进化过程中两种概率的自适应调整；设计了源于实际生产数据的仿真实验，结果表明，该生产计划模型可以用于指导生产现场作业，从而达到按期交货、降低库存成本的目的。重点研究了考虑多阶段库存、多批次流向的冷轧生产计划模型的构建和算法的实现。

## 5 系统设计与实现

为了使本文所提出的模型和算法得以推广和应用,本章设计了系统数据库,并以 MyEclipse 为环境平台开发了冷轧薄板订单生产计划系统,本系统的设计与实现对冷轧企业订单的生产排产具有重要的指导作用和实践意义。

### 5.1 系统开发技术与平台

系统的总体设计与开发采用了基于 J2EE 的 B/S 模式, B/S 即 Browser/Server (浏览器/服务器)。为企业提供了一个跨平台、开放的和松耦合的应用框架,能够有效地支持系统的进一步扩展、升级、修改以及与第三方应用程序的集成,可维护性较强。

#### 5.1.1 Tomcat 服务器

Tomcat 服务器是一个 JSP/Servlet 引擎,是 Apache Jakarta 项目的子项目之一。它是由 Apache、Sun 及其他公司和个人共同组织开发的免费且开放源代码的 Web 应用服务器。Sun 公司参与并支持 Tomcat 的开发和维护工作,因此,在 Tomcat 中总能够体现出所有最新的 Servlet 和 JSP 规范,它也是 Servlet 和 JSP 容器,能够支持 JSP 和 Servlet 的开发使用。由于 Tomcat 具有技术先进、性能稳定、完全免费以及支持各种操作系统的特点,成为了业界 JSP 程序开发人员和调试人员的首选,并得到了众多软件开发商的支持,是目前较为广泛使用的 Web 应用服务器之一。

运行 Tomcat 很简单,只要在机器上将 Tomcat 服务器安装配置正确后,启动 Tomcat 服务器即可使用它请求访问 HTML 页面。虽然 Tomcat 服务器是 Apache 服务器的延展,但 Tomcat 服务器执行时是独立与 Apache 服务器单独运行的。Apache 是为 HTML 页面等静态资源服务的,而 Tomcat 是为运行 JSP 页面和 Servlet 等动态资源服务的。Tomcat 与 IIS 等所有的 Web 服务器一样,可以处理 HTML 页面。但是,对于处理静态的 HTML, Tomcat 的能力不如 Apache 服务器。

#### 5.1.2 MyEclipse 简介

MyEclipse 是一个基于 Java 的、开源的以及可扩展的 IDE,它是在 eclipse 的基础上开发的功能强大的企业级集成开发环境平台,主要用来开发 Java、Java EE 以及相关移动应用产品。具有完美的编码、调试、测试和发布的功能,支持 HTML, JSP, CSS, Struts, Javascript, SQL 等框架技术。同时, MyEclipse 是一个 Eclipse 支持下的 J2EE 插件。Myeclipse 插件基本上包含了 Eclipse 插件的功

能，提供了很强大的 Web 开发支持，通过 MyEclipse 可以新建 Web 项目和 JSP 页面，简单和方便的进行 Web 开发，能够满足我们的各种应用需求。MyEclipse 是一个完备的、功能强大的集成开发环境，支持也较为广泛，可以完整支持 Java Servlet, JSP, Struts, JDBC 数据库链接工具等多项技术，几乎涵盖了目前所有主流开源产品的 eclipse 开发工具。

## 5.2 数据库设计

数据库是用来存储和管理系统数据的，是为系统运行服务的，是系统运行的前提，因此数据库是整个系统的核心部分。数据库中表与表之间的关联也就是系统在实际应用中的实体与实体之间的联系，数据表的关系图如图 5.1 所示。本系统实现所需要的数据都是在企业实习调研所获得的真实数据，系统在运行过程中所需要的主要信息表包括表 5.1 为原始数据表结构，表 5.2 为原始数据表，表 5.3 为物料批次信息表结构，表 5.4 为物料批次信息表，表 5.5 为批次加工时间表结构，表 5.6 为物料批次的各机组加工时间表。

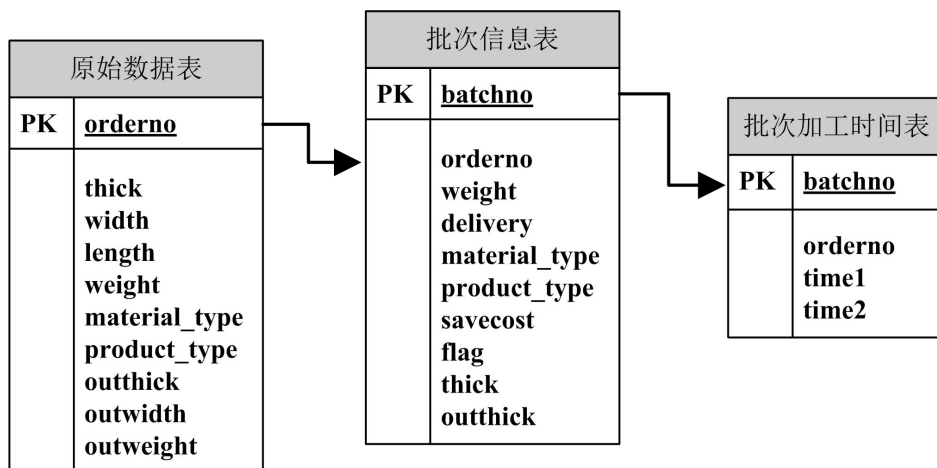


图 5.1 数据表关系图

表 5.1 原始数据表结构

列名	数据类型	允许为空	说明
orderno	varchar2(15)	no	订单号
thick	number(7,2)		入口厚度
width	number(7,2)		入口宽度
length	number(7,2)		入口长度
weight	number(7,2)		入口重量
material_type	varchar2(20)		原料钢种
product_type	varchar2(20)		产品钢种
outthick	number(7,2)		出口厚度
outwidth	number(7,2)		出口宽度
outweight	number(7,2)		出口重量

表 5.2 原始数据表

序号	入口 厚度	入口 宽度	入口 长度	入口 重量	原料钢种	订单号	产品钢种	出口 厚度	出口 宽度	出口 重量
1	4.8	1345	514	25380	MRTRG00101	3001022485	SPCC	1.3	1320	25666
2	4.5	1345	496	22840	M RTP125001	3001020434	M250P1	1.4	1320	25666
3	4.5	1345	491	22640	M RTP125001	3001020434	M250P1	1.4	1320	25666
4	4.5	1345	490	22580	M RTP125001	3001020434	M250P1	1.4	1320	25666
5	4.5	1345	490	22570	M RTP125001	3001020434	M250P1	1.4	1320	25666
6	4.5	1345	492	22670	M RTP125001	3001020434	M250P1	1.4	1320	25666
7	4.8	1345	458	22720	M RTP121001	3001022911	M210P1	1.5	1320	25666
8	4.5	1335	486	22560	MRTRG00101	3001022677	SPCC	1.2	1310	25666
9	4.5	1330	540	25020	MRTRG00101	3001020515	SPCC	1.2	1305	25666
10	4.5	1330	429	19780	MRTRG00101	3001020515	SPCC	1.2	1305	25666
11	4.5	1330	547	25290	MRTRG00101	3001020515	SPCC	1.2	1305	25666
12	4.5	1330	536	24970	MRTRG00101	3001020515	SPCC	1.2	1305	25666
13	4.5	1330	539	24980	MRTRG00101	3001020515	SPCC	1.2	1305	25666
14	4.5	1330	490	22690	MRTRG00101	3001020515	SPCC	1.2	1305	25666
15	4.5	1330	542	25010	MRTRG00101	3001020515	SPCC	1.2	1305	25666
16	4.5	1330	536	24900	MRTRG00101	3001020515	SPCC	1.2	1305	25666
...	...	...	...	...	...	...	...	...	...	...



表 5.3 批次信息表结构

列名	数据类型	允许为空	说明
batchno	varchar2(5)	no	批次号
orderno	varchar2(15)		订单号
weight	number(7,2)		重量
delivery	number(7,2)		交货期
material_type	varchar2(20)		原料钢种
product_type	varchar2(20)		产品钢种
savecost	number(7,2)		单位时间的单位保存成本
flag	int		是否连退
thick	number(7,2)		入口厚度
outthick	number(7,2)		出口厚度

表 5.4 批次信息表

批次号	订单号	重量	交货期	原料钢种	产品钢种	保存成本	是否连退	入口厚度	出口厚度
1	3001022485	25380	1	MRTRG00101	SPCC	39	1	4.8	1.3
2	3001020434	22840	2	MRTP125001	M250P1	50	1	4.5	1.4
3	3001020434	22640	2	MRTP125001	M250P1	50	0	4.5	1.4
4	3001020434	22580	2	MRTP125001	M250P1	50	1	4.5	1.4
5	3001020434	22570	2	MRTP125001	M250P1	50	1	4.5	1.4
6	3001020434	22670	2	MRTP125001	M250P1	50	1	4.5	1.4
7	3001022911	22720	3	MRTP121001	M210P1	12	1	4.8	1.5
8	3001022677	22560	3	MRTRG00101	SPCC	39	0	4.5	1.2
9	3001020515	25020	1	MRTRG00101	SPCC	39	1	4.5	1.2
10	3001020515	19780	1	MRTRG00101	SPCC	39	0	4.5	1.2
11	3001020515	25290	1	MRTRG00101	SPCC	39	1	4.5	1.2
12	3001020515	24970	1	MRTRG00101	SPCC	39	1	4.5	1.2
13	3001020515	24980	1	MRTRG00101	SPCC	39	1	4.5	1.2
14	3001020515	22690	1	MRTRG00101	SPCC	39	1	4.5	1.2
15	3001020515	5010	1	MRTRG00101	SPCC	39	1	4.5	1.2
...	...	...	...	...	...	...	...	...	...

表 5.5 批次加工时间表结构

列名	数据类型	允许为空	说明
batchno	varchar2(5)	no	批次号
orderno	varchar2(15)		订单号
time1	number(7,2)		酸轧时间
time2	number(7,2)		连退时间

表 5.6 批次加工时间表

批次号	酸轧时间	连退时间
1	9	31.6
2	8	28.8
3	8	0
4	8	28.8
5	8	28.8
6	8	28.8
7	8	26
8	8	0
9	9	34.4
10	7	0
...	...	...

### 5.3 系统核心代码

#### 5.3.1 染色体基因编码

```

public static int[] encode(int batches){
    int[] individual = new int[batches] ;
    boolean[] flags = new boolean[batches] ;
    Random random = new Random() ;
    int temp ;

    for(int i = 0 ; i < flags.length ; i ++){
        flags[i] = false ;
    }

    for(int i = 0 ; i < individual.length ;){
        temp = random.nextInt(batches) ;
        if(!flags[temp]) {
            individual[i++] = temp+1 ;
            flags[temp] = true ;
        }
    }
    return individual ;
}

```

#### 5.3.2 适应值计算

```

public double getFitness1(int[] individual){
    double[][] start = new double[individual.length][2] ;
    //每个批次的物料在每道机组加工的开始时间

```

```

        double[][] finish = new double[individual.length][2];
//每个批次的物料在每道机组加工的结束时间
        double[] weights1 = new double[weights.length];
        double[] saveTime = new double[individual.length];
        double adjustTime = 1.5; //调辊时间
        double storages = 0; //中间库的库存量

        /*
         * beyond:超出最大库存的数量
         * machineTime:机器等待时间(由于中间库的库存量低于最小库存导致)
         * aheadTime:超前时间
         * behindTime:拖延时间
         */
        double beyond = 0, machineWaitTime = 0, aheadTime = 0, behindTime = 0; //所有批次的总超前时间和总延期时间
        double fitness = 0; //适应值
        int temp, pre = -1; //temp 表示当前批次号, pre 表示当前批次上一个批次号

        /*
         * 一维数组的初始化操作
         */
        ArrayUtil.init(saveTime);
        ArrayUtil.init(finish);

        /*
         * 计算每个批次的加工开始时间和加工结束时间
         */
        for(int i = 0; i < individual.length; i++){
            temp = individual[i] - 1; //当前批次号

            if(i == 0) {
                //若该批次是第一个投入生产, 则完成时间就是加工时间
                start[temp][0] = 0;
                finish[temp][0] = start[temp][0] + operationTime[temp][0];
            }
            else {
                pre = individual[i-1] - 1; //pre 表示当前批次的紧前批次
                start[temp][0] = finish[pre][0];
                if(first_thicks[temp] != last_thicks[pre]){
                    //相邻两个批次的厚度不同时需要调辊
                    finish[temp][0] = start[temp][0] + operationTime[temp][0] + adjustTime;
                }
                else finish[temp][0] = start[temp][0] + operationTime[temp][0];
            }
        }
    }

```

```

    }

    weights1[temp] = weights[temp] * productivity ;

    if(flags[temp]){ //进行连退操作,则存入中间库
        storages = storages + weights1[temp] ;
    }

    if(storages > maxStore) beyond += storages - maxStore ;

    if(i == 0){
        //第一个批次的连退加工开始时间就是第一个批次的酸轧加工结束时间
        start[temp][1] = finish[temp][0] ;
        machineWaitTime += finish[temp][0] ;
    }
    else{
        /*
            * 若当前批次的酸轧结束时间大于前一批次的连退加工结束时间, 则连退机组会出现等待
            * 若当前批次的酸轧结束时间小于前一批次的连退加工结束时间, 则物料需要在中间库中等待
        */
        if(finish[pre][1] < finish[temp][0]) {
            machineWaitTime += finish[temp][0] - finish[pre][1] ;
        }

        //连退的开始加工时间取当前批次的酸轧结束时间和前一批次的连退结束时间的最小值
        start[temp][1] = Math.max(finish[temp][0], finish[pre][1]) ;
    }
    finish[temp][1] = start[temp][1] + operationTime[temp][1] ;
    //连退的加工完成时间

    if(flags[temp]) storages -= storages - weights1[temp] ;
    //只有需要连退的批次才会影响中间库存
}

/*
    * 根据每个批次的物料超期延期状况计算惩罚值
*/
for(int i = 0 ; i < individual.length ; i++){
    temp = individual[i] - 1 ;
    if((finish[temp][1] - delivery[temp] > 0) && (finish[temp][1] - delivery[temp] < 60 * 24)){

```

```

        behindTime += 1 ;
    } else if((delivery[temp] - finish[temp][1] > 60 * 24)) {
        aheadTime += 1 ;
    }
}

/*
 * 计算每个批次物料的成品库保存成本
 * 1.计算每个批次物料在成品库中的保存时间；
 * 2.计算物料在成品库中的保存成本
 */

for(int i = 0 ; i < individual.length ; i++){    //物料在中间库中的保存成本
    temp = individual[i] - 1 ;
    if((delivery[temp] - finish[temp][1] > 0) && (delivery[temp] -
finish[temp][1] < 60 * 24)){
        saveTime[temp] = delivery[temp] - finish[temp][1] ;
    } else if((delivery[temp] - finish[temp][1] > 60 * 24)){
        saveTime[temp] = 60*24 - finish[temp][1] ;
    }
}
machineWaitTime = machineWaitTime/(60*24) ;

double saveCost = 0 ;
for(int i = 0 ; i < individual.length ; i++){
    temp = individual[i] - 1 ;
    saveCost += (PRICE * saveTime[temp] * weights1[temp])/(60*24*1000) ;
}

//计算个体的适应值
fitness = 0.12 * aheadCost * aheadTime + 0.52 * behindCost * behindTime +
0.12 * aboveCost * beyond + 0.18 * belowCost * machineWaitTime + 0.06 *
saveCost ;

return fitness ;
}

```

### 5.3.3 选择操作

```

public int[][] select(int[][] population){
    int count = 0 ;    //计数变量
    int[][] newPopulation = new int[population.length][] ;
    //选择之后的新种群
}

```

```

        int[][] genius = new int[population.length/10][] ;
//每一代种群中最优的 1/10 的个体
        double[] fitness = new double[population.length] ;
        for(int i = 0 ; i < population.length ; i ++){
            fitness[i] = getFitness1(population[i]) ;
        }

//将最优的 1/10 个体选择到 genius 中
        for(int i = 0 ; i < genius.length ; i ++){
            genius[i] = population[ArrayUtil.getMinIndex(fitness)] ;
            fitness[ArrayUtil.getMinIndex(fitness)] = -1 ;
        }

/*
 * 产生和原种群规模相同的新种群
 */
        while(count < pop_size){
            int index ; //被选中个体下标
            double[] percentage = new double[population.length] ;
//种群适应值比例

            /*
 * 轮盘赌选择，计算被选中个体的下标再将被选中个体复制到新种
群中
 */
            percentage = gambleWheel(population) ;
            double t = rand.nextDouble() ;

            /*
 * 查找被选中的个体
 */
            for(index = 0 ; index < percentage.length - 1 ; index ++){
                if(percentage[index] >= t) break ;
            }

            newPopulation[count ++] = population[index] ;
        }

        for(int i = 0 ; i < population.length ; i ++){
            fitness[i] = getFitness1(newPopulation[i]) ;
        }

        for(int i = 0 ; i < genius.length ; i ++){
            newPopulation[ArrayUtil.getMaxIndex(fitness)] = genius[i] ;
        }

```

```

        fitness[ArrayUtil.getMaxIndex(fitness)] = -1 ;
    }

    return newPopulation ;
}

```

### 5.3.4 交叉操作

```

public int[][] crossover(int[] individual1, int[] individual2 , double fmax , double
fmin , double favg) {
    /* 交叉操作的操作步骤
    * 譬如:参与遗传操作的两个个体的编码序列为
    * individual1: 3 4 8 10 6 | 7 5 1 9 2
    * individual2: 5 3 9 10 8 | 2 4 7 1 6
    * step1:利用随机函数产生一个交叉点,如 5;
    * step2:将 individual1 中 2 4 7 1 6 所对应的位置置为-1,即 3 -1 8 10 -1 | -1
    5 -1 9 -1
    * step3:将 individual2 中 2 4 7 1 6 按照相对位置插入 individual1 中的-1
    所在的位置,得到的最终子代编码序列为 3 2 8 10 4 | 7 5 1 9 6
    */

    /*
    * ind1:交叉操作的其中一个个体
    * ind2:交叉操作的另一个个体
    * child1:交叉之后的子代 1
    * child2:交叉之后的子代 2
    * child:将两个子代返回的数组
    */
    int[] ind1 = new int[individual1.length];
    int[] ind2 = new int[individual2.length];
    int[] child1 = new int[individual1.length] ;
    int[] child2 = new int[individual2.length] ;
    int[][] child = new int[2][] ;

    /*
    * 数组的初始化
    */
    ArrayUtil.copy(individual1, ind1);
    ArrayUtil.copy(individual2, ind2);

    /*
    * 采用自适应交叉概率
    * 操作步骤如下:
    * step1:计算两个父代个体 ind1 和 ind2 的适应值并将二者的最大值赋值

```



```

给 f
    * step2:如果 f<favg,则 crossRate =
PC2*Math.exp((f-favg)/(A*(favg-fmin+R))) ;
    *      否则 crossRate = PC1*Math.exp((f-fmax)/(A*(fmax-favg+R))) ;
    */
double f = 0 ;
double crossRate = 0 ;    //交叉概率
//f 取父代个体适应值的最大值
f=Math.min(getFitness1(ind1), getFitness1(ind2)) ;

/*
    * 自适应调整交叉概率
    */
if(f >= favg){
    crossRate = PC1/((PC1-PC2)+Math.exp(A*((fmax-f)/(fmax-favg+R)))) ;
} else {
    crossRate = PC2/((PC1-PC2)+Math.exp(A*((favg-f)/(favg-fmin+R)))) ;
}

if(rand.nextDouble() <= crossRate){
    int pos = rand.nextInt(ind2.length) ;    //随机产生交叉点

    /*
        * 获得交叉之后的两个子代个体
        */
    child1 = ArrayUtil.replace(ind1, ind2, pos) ;
    child2 = ArrayUtil.replace(ind2, ind1, pos) ;
} else {
    child1 = ind1 ;
    child2 = ind2 ;
}

child[0] = child1 ;
child[1] = child2 ;

return child ;

}

```

### 5.3.5 变异操作

```

public void mutate(int[][] population , int index , int[] individual , double fmax ,
double fmin , double favg){

```

```

/*
 * 突变操作的步骤
 * 譬如,突变个体的编码序列为 10 3 5 2 9 7 4 1 8 6
 * step1:利用随机函数产生两个突变点位置,如 3 和 7,该位置对应的编码
为 5 和 4
 * step2:将相应位置的编码互换位置,得到新的个体 10 3 4 2 9 7 5 1 8 6
 */
double fit = 0 , mutateRate = 0 ;
/*
 * 采用自适应突变概率
 * 操作步骤如下:
 * step1:计算突变个体的适应值 fit
 * step2:如果 fit<favg,则 mutateRate =
PM2*Math.exp((fit-favg)/(favg-fmin+R)) ;
 * 否则 mutateRate = PM1*Math.exp((fit-fmax)/(fmax-favg+R)) ;
 */
fit = getFitness1(individual) ;

if(fit >= favg){
    mutateRate =
PM1/((PM1-PM2)+Math.exp(B*(fmax-fit)/(fmax-favg+R))) ;
} else {
    mutateRate = PM2/((PM1-PM2)+Math.exp(B*(favg-fit)/(favg-fmin+R))) ;
}

if(rand.nextDouble() <= mutateRate){
    //产生两个突变点
    int pos1 = rand.nextInt(individual.length) ;
    int pos2 = rand.nextInt(individual.length) ;

    int t = individual[pos1] ;
    individual[pos1] = individual[pos2] ;
    individual[pos2] = t ;

}

}

```

## 5.4 系统主界面

系统主界面如图 5.2 所示, 主要包括加工订单数, 生产批次数, 初始种群大小, 种群遗传代数, 酸轧成材率, 交叉概率, 突变概率, 中间库最大库存量, 低于最小库存的惩罚系数, 高于最大库存的惩罚系数, 超前惩罚系数, 延期惩罚系数。其中, 酸轧成材率、交叉概率、突变概率、中间库最大库存量、低于最小库存的惩罚系数、高于最大库存的惩罚系数、超前惩罚系数和延期惩罚系数为系统默认值, 一般情况下不进行修改。加工订单数和调度批次数根据实际情况填写, 初始种群大小默认为 50, 遗传代数默认为 50。

冷轧薄板订单生产计划系统			
加工订单数	45	调度批次数	4
种群遗传代数(代)	50	酸轧成材率	0.98
交叉概率	0.07	中间库最大库存量	16000
高于最大库存的惩罚系数	0.2	超前惩罚系数	0.2
		延期惩罚系数	0.9
<a href="#">提交数据</a>			

图 5.2 系统主界面

生产计划结果显示界面如图 5.3 所示, 界面的上侧为订单的组批结果, 左侧为种群的编码序列, 右侧为该编码序列对应的适应值大小。下侧的编码序列为实际生产的生产批次号, 适应值为生产所需的费用。

生产计划结果显示			
订单组批结果显示		查看相应的订单数据	
第1批次:[1, 8, 11, 15, 16, 18, 19, 25, 27] 第2批次:[3, 20, 21, 22, 34, 35, 37, 38, 41, 42, 43] 第3批次:[5, 7, 10, 13, 17, 24, 32, 33, 39, 44, 45] 第4批次:[2, 4, 6, 9, 12, 28, 31, 36, 40] 第5批次:[14, 23, 26, 29, 30]			
初始种群编码序列	5 2 1 3 4 3 4 1 2 5 2 4 5 2 1 5 2 3 1 4 4 3 5 2 1 2 3 4 3 1 4 5 1 3 2 3 1 4 2 5 1 4 2 5 3 5 3 1 4 2 2 5 1 4 2 2 4 3 1 5 3 4 2 5 1 3 1 2 4 5 5 1 4 2 3 5 3 4 1 2 5 1 4 2 3	初始种群个体适应值	个体1的适应值=8563 个体2的适应值=8170 个体3的适应值=7974 个体4的适应值=8383 个体5的适应值=8293 个体6的适应值=7850 个体7的适应值=8670 个体8的适应值=8193 个体9的适应值=8683 个体10的适应值=8676 个体11的适应值=8116 个体12的适应值=8067 个体13的适应值=7949 个体14的适应值=8047 个体15的适应值=9034 个体16的适应值=8634 个体17的适应值=8215
最终子代编码序列	3 2 5 4 1 3 2 5 4 1 3 2 5 4 1 3 2 5 4 1 3 2 5 4 1 3 2 5 4 1 3 2 5 4 1 3 2 5 4 1 3 2 5 4 1 3 2 5 4 1 3 2 5 4 1 3 2 5 4 1 3 2 5 4 1 3 2 5 4 1 3 2 5 4 1 3 2 5 4 1 3 2 5 4 1	最终子代个体适应值	个体1的适应值=7758 个体2的适应值=7758 个体3的适应值=7758 个体4的适应值=7758 个体5的适应值=7758 个体6的适应值=7758 个体7的适应值=7758 个体8的适应值=7758 个体9的适应值=7758 个体10的适应值=7758 个体11的适应值=7758 个体12的适应值=7758 个体13的适应值=7758 个体14的适应值=7758 个体15的适应值=7758 个体16的适应值=7758 个体17的适应值=7758
最佳个体编码序列	3,2,5,4,1		
最佳个体的适应值	7758		
<a href="#">查看对应的甘特图</a>			

图 5.3 生产计划结果显示界面

批次中订单的相关信息如图 5.4 所示, 包括批次号、订单号、原料钢种、产品钢种、重量、是否连退、酸轧时间、连退时间、单位保存费用、交货期。

订单信息显示									
批次号	订单号	原料钢种	产品钢种	批次重量	是否连退	酸轧时间(分)	连退时间(分)	单位保存费用	交货期(天)
1	3001022485	MRTG00101	SPCC	202720.0	是	9.0	31.6	39.0	2.0
2	3001020434	MRTP125001	M250P1	228400.0	是	8.0	28.8	50.0	1.0
3	3001020434	MRTP125001	M250P1	203760.0	否	8.0	0.0	50.0	1.0
4	3001020434	MRTP125001	M250P1	248380.0	是	8.0	28.8	50.0	1.0
5	3001020434	MRLR117001	M170P1	203130.0	是	8.0	28.8	65.0	3.0
6	3001020434	MRTP125001	M250P1	249370.0	是	8.0	28.8	50.0	1.0
7	3001022911	MRTP121001	M210P1	159040.0	是	8.0	26.0	12.0	1.0
8	3001022677	MRTG00101	SPCC	275990.0	否	8.0	0.0	39.0	2.0
9	3001020515	MRTP121001	M210P1	250200.0	是	9.0	34.4	12.0	2.0
10	3001020515	MRTP121001	M210P1	158240.0	否	7.0	0.0	12.0	2.0
11	3001020515	MRTG00101	SPCC	278190.0	是	9.0	34.4	39.0	2.0
12	3001020515	MRTP121001	M210P1	199760.0	是	9.0	34.4	12.0	2.0
13	3001020515	MRLR117001	M170P1	274780.0	是	9.0	34.4	65.0	3.0
14	3001020515	MRTG00101	SPCC	181520.0	是	8.0	34.4	39.0	2.0
15	3001020515	MRTG00101	SPCC	225090.0	是	9.0	34.4	39.0	2.0
16	3001020515	MRTG00101	SPCC	224100.0	是	9.0	34.4	39.0	2.0

图 5.4 订单信息显示界面

图 5.5 是根据子代种群中最优个体所对应的生产加工顺序绘制的甘特图即生产计划甘特图。图中蓝色矩形表示为调辊时间，红色矩形表示为酸轧，黄色矩形表示为连退时间。

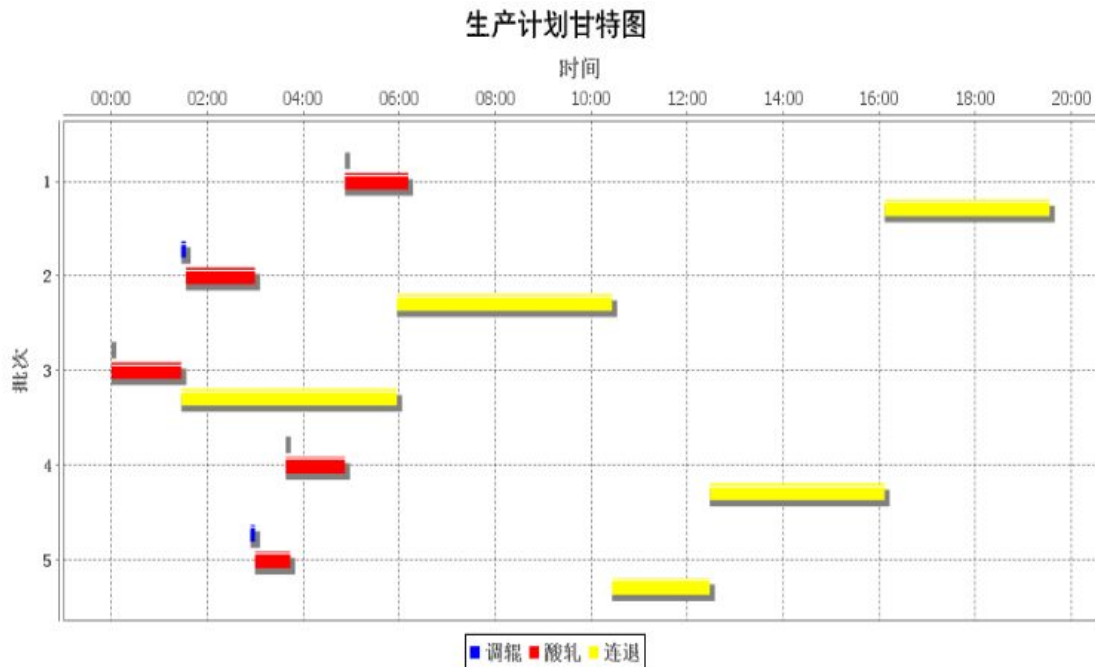


图 5.5 生产计划甘特图

## 5.5 系统运行环境

本系统是在 MyEclipse 平台上进行设计并且仿真的。

运行环境为:

(一) 软件部分

操作系统: Microsoft Windows 7

开发工具: MyEclipse2015

(二) 硬件部分

CPU: Intel(R) Core(TM)i3 CPU 2.27GHz

内存: 2GB

硬盘：300G

## 5.6 本章小结

本章采用了基于 J2EE 的 B/S 模式，通过开发工具 MyEclipse 对冷轧薄板订单生产计划系统进行设计与实现，进一步描述和分析了冷轧薄板企业的订单排产过程，设计了系统所使用的数据库，并简单介绍了冷轧薄板订单生产计划系统的功能和作用。

## 6 总结与展望

### 6.1 总结

目前,针对国内大多数冷轧企业现存的交货期超期或提前、库存超出或不足等问题。研究高质量且合理的合同组批与冷轧生产计划优化方法,为企业决策者提供了一个非常重要的管理思路和方法。由于冷轧生产过程经过多道工序和多台生产设备,同时考虑到复杂的资源和工艺约束,冷轧组批与生产计划问题已经成为冷轧企业进行自动化和信息化建设过程中亟待解决的重要课题。该课题问题的关键在于如何制定合理的合同组批和生产排产计划,同时在满足各种工艺约束的基础上,保证前后工序在库存和交货时间上的协调,从而实现合同交货期惩罚、中间库存惩罚、成品库保存成本最小化等优化目标。本文以合同交货期和库存控制为研究对象,对冷轧企业合同组批和生产计划两个方面分别进行了研究并开发和设计了冷轧薄板订单生产计划系统,其主要研究成果如下:

(1) 本文以合同交货期差异度、工艺路线差异度和调整次数最小化为目标,以合同中待加工钢卷的入口宽度、出口宽度、入口厚度、出口厚度、抗拉强度为约束建立冷轧组批数学模型。在模型求解过程中,针对合同交货期和工艺路线两个属性分别提出了余弦差异度和工艺路线差异度的计算方法,并提出了基于粒子群优化的冷轧组批模糊聚类算法。算法在聚类过程中将合同抽象为样本数据,聚类时首先利用粒子群优化算法寻找最优的聚类中心,接着根据最优聚类中心使用模糊聚类算法求解隶属度矩阵,然后结合约束条件得到组批结果。

(2) 本文系统分析了冷轧生产计划的制定过程,在综合考虑合同交货期、中间库存量、成品库成本等目标的基础上,建立满足冷轧生产工艺、机组产能、成材率与库存平衡等约束的冷轧生产计划数学模型。模型重点考虑了冷轧生产系统的多阶段、多流向问题。所谓多阶段是指生产系统具有中间库、成品库等多个阶段的库存,多流向是指同一批次存在多种流向(可视为工艺路线);进而采用改进的遗传算法对模型进行求解。

(3) 为了使本文所提出的模型和算法得以推广和应用,本文采用了基于 J2EE 的 B/S 模式开发了冷轧薄板订单生产计划系统。该系统为冷轧薄板企业的生产提供了指导作用,提高了冷轧企业的生产管理水平。

最后以某冷轧企业的生产实际为案例,对全文所提出的模型和方法进行了验证。结果表明,所建立的合同组批和生产计划模型可以用于指导生产现场作业,从而达到按期交货、降低生产成本的目的。

## 6.2 展望

冷轧企业的合同组批与生产计划问题比较复杂，研究合同组批与生产计划对冷轧企业的持续发展具有深远的理论和实践意义。根据目前现有的研究成果和企业现状，今后仍需要在以下几个方面继续做深入的研究：

(1) 进一步优化合同组批方法，使合同组批在冷轧生产计划优化问题中发挥更重要的作用。

(2) 本文侧重于考虑多阶段多流向的冷轧生产计划，下一步重点研究含有逆向批次流向的生产计划问题研究，使编制的生产计划更加切实合理。

(3) 本文中所设计和开发的冷轧薄板订单生产计划系统的功能与模块仍需要作进一步完善，进而使其更好的在冷轧企业中进行推广与应用。

## 参考文献

- [1] 赵珺, 王伟, 刘全利. 冷轧薄板生产线组批调度模型与算法[J]. 计算机集成制造系统, 2008, 14(10): 1957-1965.
- [2] 王志刚, 赵珺, 王伟. 冷轧合同批量调度的模糊 Job Shop 模型及算法[J]. 控制与决策, 2009, 24(10): 1455-1462.
- [3] 杨静萍, 刘晓冰, 王宇春, 等. 基于多约束聚类的钢铁合同组批质量设计方法[J]. 计算机集成制造系统, 2009, 15(11): 2104-2110.
- [4] 唐成龙, 王石刚. 基于数据间内在关联性的自适应模糊聚类模型[J]. 自动化学报, 2010, 36(11): 1544-1556.
- [5] 刘华强, 杨荃, 唐荻, 等. 模糊聚类和模糊评判在冷连轧机轧制策略优化中的应用[J]. 钢铁研究学报, 2008, 20(1): 54-58.
- [6] 张晓平, 王伟, 赵珺. 基于系统聚类和自适应量子遗传算法的钢卷拼卷方法[J]. 计算机集成制造系统, 2009, 15(7): 1421-1429.
- [7] 王利, 王伟, 高宪文, 等. 冷轧机组批量作业计划模型与算法[J]. 控制理论与应用, 2010, 27(5): 582-588.
- [8] 刘国莉, 张博, 唐立新. 可延迟供货的冷轧生产库存问题之建模与优化[J]. 控制理论与应用, 2012, 29(11): 1512-1516.
- [9] 李庆予, 朱宏. 钢铁企业供应链客户订单的组批优化技术研究[J]. 工业工程与管理, 2003, 2: 51-53.
- [10] 彭威, 刘爱国. 冷轧企业负荷平衡原料计划模式和方法[J]. 计算机集成制造系统, 2004, 10(6): 662-666.
- [11] 谢志强, 刘长海, 杨静. 2 个加工时间不同工序组批的嵌套批综合调度算法[J]. 机械工程学报, 2012, 48(24): 164-173.
- [12] 贾瑞玉, 王会颖. 基于改进蚁群算法的聚类分析[J]. 计算机应用与软件, 2010, 27(12): 97-100.
- [13] 张强, 李淼. 基于遗传算法和遗传模糊聚类的混合聚类算法[J]. 计算机工程与应用, 2007, 43(3): 164-165.
- [14] 王利, 高宪文, 王伟, 等. 基于模型的子空间聚类与时间段蚁群算法的合同生产批量调度方法[J]. 自动化学报, 2014, 40(9): 1991-1997.
- [15] 唐立新. 基于智能优化的钢铁生产计划与调度研究[J]. 管理学报, 2005, 2(3): 263-267.
- [16] 李耀华, 宁树实, 王伟, 等. 基于准时制的轧钢厂生产计划模型及算法[J]. 控制工程, 2004, 11(4): 321-324.
- [17] 安玉伟, 严洪森. 柔性作业车间生产计划与调度集成优化求解策略[J]. 自动化学报, 2013, 39(9): 1476-1491.
- [18] 王志刚, 邬静东, 赵珺, 等. 冷轧多机组合同优化排产模型及算法(上)[J]. 冶金自动化, 2010, 34(4): 10-15.
- [19] 成国庆, 周炳海, 李玲, 等. 考虑缓冲区库存的退化系统最优维修更换策略[J]. 计算机集成制造系统, 2015, 21(6): 1593-1600.
- [20] 李铁克, 施灿涛. 冷轧生产批量计划与调度问题模型及算法[J]. 管理学报, 2008, 5(1): 64-69.



- [21] 严蕾. 冷轧机组作业预计划技术及应用[J]. 冶金自动化, 2014, 38(4): 25-29.
- [22] 纪鹏程, 宋士吉, 吴澄, 等. 钢铁企业复杂库存环境下的精确库存成本建模[J]. 计算机集成制造系统, 2010, 16(2): 293-298.
- [23] 王林, 顿彩霞, 杨锐. 基于混合差分进化算法和熵权 TOPSIS 的多目标 $(Q,r)$ 库存模型[J]. 控制与决策, 2011, 26(12): 1913-1916.
- [24] 孙玲. 冷轧全流程生产计划编制方法研究[J]. 冶金自动化, 2013, 37(4): 39-43.
- [25] 张涛, 王梦光, 唐立新, 等. 钢厂合同计划的模型与算法[J]. 控制理论与应用, 2000, 17(5): 711-715.
- [26] 蒋海波, 何莉, 李恩. 生产计划的优化模型[J]. 成都大学学报(自然科学版), 1996, 15(3): 26-32.
- [27] 王文鹏, 杨再步, 李铁克. 冷轧生产线的批量计划与调度方法[J]. 冶金自动化, 2006, 30(5): 11-15.
- [28] 王洪锋, 李洪宇. 生产计划和车间排程问题的整数规划解法[J]. 现代制造工程, 2008, 07: 92-95.
- [29] 赵艳刚. 钢铁企业原材料库存控制优化机制研究[J]. 现代商贸工业, 2015, 04: 44-45.
- [30] WANG D.D., TIEU A.K., BOER F.G., et al. Toward a heuristic optimum design of rolling schedules for tandem cold rolling mills[J]. Engineering Application of Artificial Intelligence, 2002, 13(4): 397-406.
- [31] MOON S, HRYM AK A N. Scheduling of the batch annealing process deterministic case [J]. Computers and Chemical Engineering, 1999, 23(9): 1193-1208.
- [32] Amir A, Lipika D. A K-means type clustering algorithm for subspace clustering of mixed numeric and categorical data sets[J]. Pattern Recognition Letters, 2011, 32(7): 1062-1069.
- [33] Theam Foo Ng, Pham T D, Xiuping Jia. Feature interaction in subspace clustering using the Choquet integral[J]. Pattern Recognition, 2012, 45(7): 2645-2660.
- [34] Liu Quanli, Wang Wei, Hong-ren Zhan, et al. Optimal scheduling method for a bell-type batch annealing shop and its application[J]. Control Engineering Practice, 2005, 13(10): 1315-1325.
- [35] Wang L, Zhao J, Wang W. Order planning model and algorithm for wholeprocess of cold rolling process[J]. ICIC Express Letters, 2009, 3(3): 657-662.
- [36] Zhang Wenxue, Li Tieke. Batch plan optimization of steel integrated production with multiple processes[J]. Computer Integrated Manufacturing Systems, 2013, 19(6): 1296-1303.
- [37] TANG Lixin, GONG Hua, LIU Jiyin, et al. Bicriteria scheduling on a single batching machine with job transportation and deterioration considerations[J]. Naval Research Logistics, 2014, 61(4): 269-285.
- [38] XU Rui, CHEN Huaping, LI Xueping. A bi-objective scheduling problem on batch machines via a Pareto-based ant colony system[J]. International Journal of Production Economics, 2013, 145(1): 371-386.
- [39] Boyett J, White C R. Forecasting lead times with economic data[J]. Production Planning and Control, 1993, 4(1): 27-34.
- [40] Slotnick, Susan A. Optimal and heuristic lead-time quotation for an integrated steel mill with a minimum batch size[J]. European Journal of Operational Research, 2011, 210(3): 527-536.
- [41] Ramezani R, Rahmani D, Barzinpour F. An aggregate production planning model for two phase production systems: Solving with genetic algorithm and tabu search[J]. EXPERT SYSTEMS WITH APPLICATIONS, 2012, 39(1): 1256-1263.
- [42] Matsuda K, Yoshida K, Sasaki S, et al. Optimization of order allocation to in-stock slabs by

- genetic algorithm[J]. Transactions of the Society of Instrument and Control Engineers, 1992, 33(2): 118-126.
- [43] Mirmohammadi M H, Haddad H, Naghavi S M. A hybrid set-up optimization model for tandem cold rolling mill[C]. Proceedings of the 6th International Conference on Informatics in Control, Automation and Robotics. Intelligent Control Systems and Optimization, 2009, 01: 49-54.
- [44] Chen J S. Load distribution algorithm of process control system in tandem cold rolling[J]. Materials Research Innovations, 2015, 19(4): 220-224.
- [45] Pindrock B.J., Readal G.J., Ketter W.J. et al. Cold mill modernization at U.S. Steel's Mon Valley Works[J]. AISE Steel Technology, 2002, 79(4): 32-38.
- [46] Moylan P. J. A connective stability result for interconnected passive systems [J]. IEEE Trans on Automatic Control, 1980, 25(4): 812-813.
- [47] As' ad R, Demirli K. Production scheduling in steel rolling mills with demand substitution: Rolling horizon implementation and approximations[J]. International Journal of Production Economics, 2010, 126(2): 361-369.
- [48] Huegler, P.A.; Hartman, J.C. Fulfilling orders for steel plates from existing inventory[J]. Journal of the Operational Research Society, 2007, 58(9): 1156-1166.
- [49] LIU Guoli, ZHANG Bo, TANG Lixin. Modeling and optimization for a cold rolling production-inventory problem with backlogging[J]. Control Theory & Applications, 2012, 29(11): 1512-1516.
- [50] YANG Yang, CHEN Lijun. Splittable order planning problem with consideration of changeover costs[C]. 2015 27th Chinese Control and Decision Conference, 2015: 1252-1256.
- [51] 张京钊, 江涛. 改进的自适应遗传算法[J]. 计算工程与应用, 2010, 46(11): 53-55.
- [52] 任子武, 伞冶. 自适应遗传算法的改进及在系统辨识中应用研究[J]. 系统仿真学报, 2006, 18(1): 41-43.
- [53] 乔阳. 基于改进遗传算法的图像分割方法[D]. 成都:电子科技大学, 2013.

## 图清单

图 1.1 论文框架图.....	8
图 2.1 酸轧工艺流程图.....	9
图 2.2 退火工艺流程图.....	10
图 2.3 典型轧制周期和周期轧制吨位.....	11
图 3.1 冷轧合同组批编制流程.....	14
图 4.1 冷轧生产工艺流程图.....	26
图 4.2 遗传算法流程图.....	28
图 4.3 两种遗传算法的最优适应值变化曲线.....	33
图 4.4 两种遗传算法的平均适应值变化曲线.....	33
图 4.5 冷轧批次排产甘特图.....	35
图 5.1 数据表关系图.....	37
图 5.2 系统主界面.....	48
图 5.3 生产计划结果显示界面.....	48
图 5.4 订单信息显示界面.....	49
图 5.5 生产计划甘特图.....	49

## 表清单

表 2.1 计划顺序编制优先级.....	11
表 2.2 核心材料渐窄原则.....	11
表 3.1 相关的生产实际数据.....	20
表 3.2 工艺路线编号和名称对照表.....	20
表 3.3 两种算法的聚类结果.....	21
表 3.4 各批次的约束情况.....	21
表 3.5 两种算法的聚类结果各优化子目标对比.....	22
表 3.6 各批次内合同生产顺序.....	24
表 4.1 相关的生产实际数据.....	32
表 4.2 两种遗传算法的性能及各优化目标对比.....	32
表 4.3 问题 $P_1$ 两种遗传算法的排产结果对比表.....	34
表 5.1 原始数据表结构.....	38
表 5.2 原始数据表.....	38
表 5.3 批次信息表结构.....	39
表 5.4 批次信息表.....	39
表 5.5 批次加工时间表结构.....	39
表 5.6 批次加工时间表.....	40

## 攻读硕士学位期间参加的科研项目与发表学术论文情况

### 参加的科研项目

[1] 安徽省科技厅软科学重大项目“安徽省智能制造业关键技术选择研究”。

### 发表学术论文

[1] 考虑多阶段多流向的冷轧生产计划模型及算法，计算机集成制造系统(EI 检索期刊)，已录用，2016。

## 致谢

在论文完成之际，我要衷心地感谢在两年硕士研究生期间对我学习和生活给予帮助和鼓励的老师、同学及家人们，正是他们对我的帮助和关爱使我顺利地完成了研究生学业。

感谢潘瑞林老师对本篇论文的大力支持和认真指导。在攻读硕士学位期间，潘老师踏实的解惑授业态度和严谨的科研教学风格给本人留下了深刻的印象，并从中受益匪浅，必将在今后的学习和工作中产生积极的影响。在此，谨向潘老师表示感谢。

感谢马钢（合肥）冷轧薄板厂给我实习调研的平台，感谢在实习期间给予帮助的老师、企业领导和基层员工，让我学习到了宝贵的工作经验和企业生产运营管理知识。

感谢在校期间伴我一起学习和成长的同学以及师兄弟们。感谢同伴们在学习和生活中真诚的关心和鼓励。

感谢安徽工业大学管理科学与工程学院给予的关心和帮助。

最后，感谢评阅和参加论文答辩的各位专家和教授。