

研究报告

作品名称 面向军工装备制造业的智能优化排产软件

申报人 王妍、刘琦、刘源波、李邈、

梁啟鏘、倪大元

类别 科技发明制作 A 类

2013 年 6 月

摘要

“多品种、变批量、产研结合、混线生产”是我国军工装备制造业独特的生产模式。这种模式产生了大量的共用生产线，导致零件的加工路线需要跨多条生产线（即多个车间）频繁转移。在我国三代主战坦克的生产中，有 50% 以上的零件需要跨车间协作完成，在延迟零件中有 75% 以上属于跨车间协作零件。排产软件（即生产计划调度软件）在各行各业的应用已很普遍，但在我国大型军工装备制造企业中，复杂零部件的生产却没有使用排产软件。其原因是：普通排产软件只面向车间内部，不能解决军工装备制造业跨车间的独特问题。因此，本作品的目标是：根据国防军工的特点专门开发一款能够解决跨车间问题的智能排产软件。

本作品的核心和难点在于如何设计跨车间协作的调度算法。在国内外现有的研究成果中，跨车间调度算法大多基于元启发式算法或精确求解算法，只适用于较小的问题规模。而面对军工装备制造业上千台机器、数千种零件的问题规模，计算效率成了阻碍现有算法投入应用的重要瓶颈。

为此，我们基于超启发式方法设计了新颖的跨车间调度算法，通过高层元启发式算法搜索合适的启发式规则，再应用得到的组合规则求得调度解。更进一步地，针对当前的超启发式算法需要事先指定规则集的限制性，将超启发式算法扩展为三个阶段：第一阶段通过遗传编程生成规则集，第二阶段通过遗传算法从规则集中选择合适的规则，最后运用得到的组合规则求得调度解。

结果表明，我们设计的算法在数千种零件、数万道工序的问题规模下，只需 3~15 分钟就能得到计算结果，突破了实际应用的瓶颈。并且与现有研究成果相比，零件延迟率降低 10% 以上，平均延迟时间降低 60% 以上，零件跨车间率降低 30% 以上。

在此基础上我们采用 J2EE 技术架构开发了智能优化排产软件，实现了普通排产软件所不具备的功能，具体包括（1）在跨车间条件下制定车间计划和分批计划；（2）在数千种零件的规模下实现跨车间生产调度，并能预测窄口位置、实现物流平衡。另外还实现了设备管理、生产日历管理、加班及班次管理等实用的管理功能。

本作品已经在兵器一机成功应用，在北奔投入试运行，并且在航空航天、海洋工程、轨道交通等装备制造业具有广阔的应用前景。

关键词：军工装备制造业；生产计划与调度；排产软件；超启发式算法

Abstract

The production mode adopted by the military equipment manufacturing industry of China can be described as *multiple products and varying batches in mixed production lines, combined with research*. Such a peculiar production mode results in large amounts of shared production lines, which incurs frequent intercell moves for parts among various production lines or workshops. For the 3rd generation of the main battle tanks of China, more than 50% parts need to visit more than one workshop, and among all the tardy parts, more than 75% are caused by inefficient intercell cooperation.

However, in contrast to other mechanical manufacturing industries in which production arrangement softwares, i.e. production planning and scheduling softwares, have been widely used, in the military equipment manufacturing enterprises, the usage of such softwares is still very few. The main reason lies in that, those common production arrangement softwares are designed to solve the internal problems of a single workshop and cannot deal with the peculiar intercell scheduling problems arisen from the military equipment manufacturing industry. Therefore, our goal is to design and develop an intelligent production arrangement software especially for the intercell scheduling problems to meet the requirements of our military equipment manufacturing industry.

The most important and difficult part of this work is to design the intercell scheduling algorithm. With respect to the current researches, meta-heuristics are usually adopted to solve the intercell scheduling problems, which are suitable only for small dimension problems. However, hundreds of machines and thousands of parts are often involved in some complicated military equipments, which makes the computational efficiency the bottleneck of applying the related scheduling algorithms.

Therefore, in order to obtain computation efficiency and optimization performance simultaneously, we design a novel intracell scheduling algorithm based on hyper-heuristics, using meta-heuristic to search appropriate heuristic rules, and then using the obtained combinatorial rules to schedule parts and machines. Moreover, due to the limitation of hyper-heuristics that the set of the heuristic rules have to be determined in advance, we extend our algorithm into three stages. In the first stage, genetic programming is introduced to generate the set of the heuristic rules; in the second stage, genetic algorithm is adopted to search appropriate rules from the generated set; and finally, the obtained combinatorial rules are used to schedule parts and machines.

Computational results show that, for large dimension problems with thousands of parts, it spends only 3 to 15 minutes for our algorithm to schedule all the operations and machines, and compared with the current researches, the ratio of tardy parts is decreased more than 10%, the mean tardiness is decreased more than 60%, and the ratio of the tardy parts is decreased more than 30%.

After conquering the most important bottleneck of the intercell scheduling algorithm, we develop the intelligent production arrangement software using J2EE. This software has implemented the following functions that others did not achieve, (i) production plans and

batching plans under the circumstances of frequent intercell moves, and (ii) schedule thousands of parts with intercell moves, and predict the bottleneck to achieve logistics balance. In addition, other useful functions such as machine management and calendar management are also implemented in this software.

This software has been applied to the production management of two large equipment manufacturing enterprises, and has a good application prospect in the production management of military vehicles, aircrafts, and marine equipments, etc.

Key words: military equipment manufacturing industry; production planning and scheduling; production arrangement software; hyper-heuristics

目录

第 1 章 绪论	1
1.1 引言	1
1.2 国内外研究现状	3
1.2.1 跨车间调度问题研究现状	3
1.2.2 超启发式算法研究现状	4
1.3 作品的创新点	5
1.4 作品的科学性、先进性和实用性	6
1.4.1 科学性	6
1.4.2 先进性	6
1.4.3 实用性	6
1.5 本章小结	7
第 2 章 基于超启发式的跨车间调度算法	8
2.1 引言	8
2.2 问题的数学模型	8
2.3 启发式搜索遗传算法	12
2.3.1 编码方案	12
2.3.2 解码方案	15
2.3.3 遗传算子	16
2.4 基于遗传编程的改进算法	17
2.4.1 GP 流程	18
2.4.2 表示方案	19
2.4.3 参数设置	19
2.4.4 适应度评价	23
2.4.5 整体算法	23
2.5 HSGA 的性能分析	24
2.5.1 实验设计	24
2.5.2 参数分析	25

2.5.3 性能比较实验	29
2.5.4 敏感度分析	34
2.5.5 计算效率	36
2.6 改进算法的性能分析	38
2.6.1 实验设计	38
2.6.2 数据集	38
2.6.3 性能比较实验	38
2.7 真实数据验证	49
2.8 本章小结	50
第 3 章 基于能力平衡的作业计划优化算法	51
3.1 引言	51
3.2 算法设计	51
3.3 数据分析	55
3.4 本章小结	58
第 4 章 智能优化排产软件开发	59
4.1 软件功能介绍	59
4.2 软件系统结构	64
4.2.1 网络结构	64
4.2.2 技术架构	64
4.3 模块详细设计	66
4.3.1 生产计划	66
4.3.2 生产调度	68
4.3.3 生产管理	73
4.4 软件使用说明	76
4.4.1 安装与初始化	78
4.4.2 生产计划	80
4.4.3 生产调度	88
4.4.4 生产管理	104
4.5 实际生产应用	121
4.5.1 单线生产	121

4.5.2 混线生产	123
4.6 本章小结	124
结论	126
参考文献	127
作品成果清单	129
致谢	130
附件清单	131

第 1 章 绪论

1.1 引言

“多品种、变批量、产研结合、混线生产”是我国军工装备制造业独特的生产模式。在这种生产模式下，产生了大量的共用生产线，导致零件的加工路线需要跨多条生产线（即多个车间）频繁转移。

我们在 617（内蒙古第一机械集团有限公司，我国三代主战坦克的研制和生产单位）进行暑期社会实践中发现，在我国三代主战坦克的生产中，有 50% 以上的零件需要跨车间协作完成（如表 1.1 所示），在延迟零件中有 75% 以上属于跨车间协作零件；再加上加工路线非常复杂，仅综合传动装置就涉及九百余台设备，七千多种零件，六万余道工序，问题复杂性和问题规模都非常大。在这样的情况下如何制定生产计划和调度方案就成了非常棘手的问题。

表 1.1 综合传动装置跨车间零件比例统计

车型	跨车间零件比例	跨车间次数/工序总数
A	46.74%	11.21%
B	46.29%	10.14%
C	63.76%	13.09%
D	48.50%	11.63%
E	47.76%	9.22%
平均值	50.61%	11.06%

作为计算机专业的学生，我们想，能否运用我们的专业知识解决这一实际问题呢？于是就有了这个作品，面向军工装备制造业的智能优化排产软件。

最初在调研中最令我们不解的是，现在国内外有许多生产计划调度软件，但是为什么像 617 这样的大企业，在它最关键的零部件（例如综合传动装置等）生产中却没有借助软件进行排产呢？在更大范围内进行调研后我们得到了这样的结论：现有的排产软件（即生产计划调度软件）不能解决军工装备制造业的独特问题。以汽车工业为例，各车型的生产线都是独立的，零件跨车间转移的情况非常罕见，排产软件只须面向生产线内部或车间内部。因此，**就像我们的国防工业要自主发展一样，国防军工的排产软件也要根据其特点专门开发。**

研制这款软件的技术关键在于，怎样设计跨车间调度算法？要解决这个问题，首先要弄清楚跨车间调度算法和传统的车间内部调度算法有什么本质区别。于是我们分析了跨车间为生产带来的影响。

（1）跨车间加剧了物流不平衡

大型生产系统中的车间之间通常存在生产能力差异。在我们的调研中（如表 1.2 所示），综合传动装置的 11 个车间中平均工序时间最短的 413 车间只有 3.71 分钟，而最长的 409 车间达 71.01 分钟。在跨车间条件下，409 车间极易成为工艺路线上的窄口（即瓶颈）。

表 1.2 综合传动装置各车间零件加工工时比较

车 间	最长工时	最短工时	平均工时
402	713.51	0.54	57.13
403	618.56	0.1	18.20
404	339.42	0.04	11.59
405	1097.6	0.06	36.04
406	1076.2	0.02	10.54
408	3967.9	0.6	61.99
409	6923.1	0.1	71.01
410	1495.3	0.16	44.22
411	303.25	0.72	25.72
412	854.46	0.01	17.21
413	333.33	0.01	3.71

（2）跨车间使各车间排产相互依赖和制约

在跨车间条件下如果各车间仍进行封闭式的独立排产决策，如图 1.1 所示，401 车间执行完某零件的第 1 道工序、并将其转到 402 车间后，无法获知它何时能回到 401 车间执行第 4 道工序。同样，402 车间也无法获知该零件何时到来，无法将它和其他零件一起统筹调度。调研显示，一个零件的加工过程平均需要跨车间 4 次，最长达 11 次。由于产生了车间之间的依赖和制约，使得各车间的作业计划无法与其他车间割裂而独立存在。

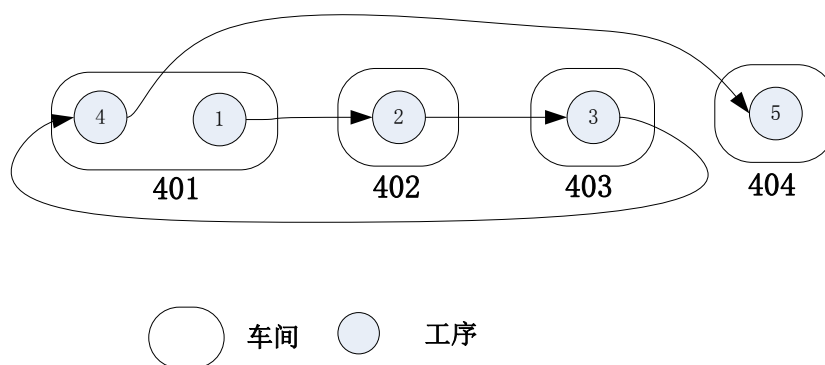


图 1.1 跨车间零件工艺路线示例

(3) 跨车间增加了排产问题规模

传统的生产调度局限于车间内部，而跨车间协作使零件调度必须面向多个车间组成的平面结构，导致问题规模急剧增加，而在跨车间条件下又不能分解为多个车间的子问题独立求解。对于军工装备复杂产品上千台设备、数千种零件、数万道工序的问题规模来说，一般的算法很难满足计算效率的要求。

基于上述分析我们得到这样的结论：**跨车间调度算法要兼具较好的优化能力和较高的计算效率，这也是本作品的技术关键和难点。**

超启发式算法（Hyper-Heuristic）是近年来新兴的一类优化算法，通过高层的元启发式方法选择低层的启发式规则来解决复杂计算的搜索问题，兼有元启发式方法的寻优能力和分派规则的计算效率。本作品基于元启发式算法设计了跨车间调度算法，并改进了现有超启发式算法的局限性以获得更好的优化性能。在此基础上研制开发了专门针对我国军工装备制造业的智能优化排产软件。

1.2 国内外研究现状

下面分别从跨车间调度问题和超启发式算法两个方面介绍当前国内外的研究现状。

1.2.1 跨车间调度问题研究现状

据统计，在大中型制造企业中，有 72.9% 的企业实行单元生产模式，但零件的平均跨车间率达 20% 以上，且只有 10% 的车间无需其他车间协作。尽管上世纪 90 年代 Garza 等学者就指出，由于跨车间转移难以避免，理想的单元制造系统 (Cellular Manufacturing System, CMS) 将难以实施^[1]。但直到近年来，随着 CMS 的实施逐渐深

入并遇到困难，这一问题才开始被关注^[2-8]，相关研究可以分为跨流水车间调度和跨作业车间调度两种类型。

(1) 跨流水车间调度

Yang 等人^[2]的问题模型是零件至多在两个单元间移动，且只移动一次。采用分支定界法和启发式算法解决每个单元内工序的加工顺序。Solimanpur 等人^[3]考虑了特殊零件在两个以上单元间的转移问题。首先忽略零件的跨单元特点，确定零件在单元内的加工顺序。然后按权重递减的顺序排列单元间的顺序。Tavakkoli 等人^[4]和 Gholipour 等人^[5]均采用基于分散搜索的元启发式算法解决零件在单元内和单元间的多目标调度问题。

(2) 跨作业车间调度

唐加福等人假设每个零件都有一条已知路径，特殊零件在若干个单元之间转移，采用分散搜索方法设计单元构建^[6]并解决跨单元调度问题^[7]。基于此问题模型，Elmi 等人^[8]允许零件可重入，采用模拟退火的方法，同时利用邻近结构优化最终解。

现有方法大多基于分散搜索、模拟退火、蚁群优化等元启发式算法或分支定界等精确求解算法，适用于较小的问题规模（20~60 个零件、10~40 台机器、2~8 个车间）。而在军工装备复杂产品的生产中，往往涉及上千台机器、数千种零件、数万道工序。在这样的条件下，问题规模大大增加，现有方法难以满足计算效率的要求。**计算效率已经成为阻碍跨车间调度算法相关研究成果投入实际应用的重要瓶颈。**

1.2.2 超启发式算法研究现状

超启发式算法（Hyper-Heuristic）是一种通过选择或生成启发式规则以解决复杂计算搜索问题的优化方法^[9]，因其适用于解决大规模组合优化问题^[10,11]，近年来引起国内外的关注。

超启发式算法由高层启发式方法和低层启发式规则构成。高层的启发式方法通过搜索低层的启发式规则来完成对全局问题解的搜索。当高层启发式采用元启发式方法，低层启发式方法采用分派规则时，超启发式算法将兼有元启发式方法较强的寻优能力和分派规则的高效计算性能。通过高层元启发式算法选择合适的启发式规则，然后用得到的组合规则求得调度解^[12-16]。

目前，用于生产调度领域的超启发式算法可以分为两类，即面向工序的超启发式算法和面向机器的超启发式算法。

(1) 面向工序的超启发式算法

面向工序是指高层元启发式算法是为工序选择启发式规则的。Dordorf 和 Pesch^[15]通过遗传算法搜索一个最优的启发式规则序列用于选择每次调度的工序。对于多目标作业车间调度问题，Vazquez-Rodriguez 和 Petrovic^[14]基于混合启发式规则的遗传算法，该算法同时为工序搜索合适的启发式规则序列和每个启发式规则调度的工序数目。

(2) 面向机器的超启发式算法

面向机器是指高层元启发式算法是为机器搜索启发式规则的，每当机器空闲时，机器利用该规则调度其缓冲区的零件。Fayad 和 Petrovic^[12]采用遗传算法为模糊作业车间的每台机器搜索合适的启发式规则。Ponnambalam 等人^[13]为多目标作业车间设计了一种类似的方法。

对我们的作品来说，问题更为复杂。具体来说就是，由于柔性加工路径的存在，要求我们的超启发式算法要同时考虑零件和机器，即同时为分派问题和排序问题搜索合适的启发式规则。这对我们的算法设计提出了更高的要求。

1.3 作品的创新点

本作品的创新点可以归纳为：算法具有创新性，软件具有实用性。

(1) 通过超启发式算法大幅提高了跨车间调度算法的计算效率，突破了阻碍相关研究投入实际应用的重要瓶颈。

(1-1) 大幅提高了跨车间调度算法可承受的问题规模

在国内外公开发表的研究成果中，跨车间调度问题的规模是：20~60 个零件、10~40 台机器、2~8 个车间，难以满足实际生产中大规模问题的计算效率要求。我们基于超启发式设计的跨车间调度算法，在数千种零件、数万道工序的问题规模下，3~15 分钟内就能得到计算结果。

(1-2) 改进了现有超启发式算法的局限性

当前的超启发式算法通常包括两个阶段，第一阶段通过元启发式算法在规则集中选择合适的规则，第二阶段应用得到的规则进行调度。其局限性在于，规则集是事先指定的，缺少科学依据。为此我们设计了三阶段的超启发式算法，即在原有的两阶段之前，通过遗传编程（Genetic Programming, GP）生成规则集。我们的算法与国内外现有跨车间调度研究成果相比，零件延迟率降低 10% 以上，平均延迟时间降低 60% 以上，零件跨车间率降低 30% 以上。

(2) 针对实际问题设计开发了智能优化排产软件，在我国军工装备制造业生产管理中具有很强的实用性。

作品实现了普通排产软件所不具备的功能，具体包括：

(2-1) 在跨车间条件下制定车间计划和分批计划；

(2-2) 在数千种零件的规模下实现跨车间生产调度，并能预测窄口位置、实现物流平衡。

1.4 作品的科学性、先进性和实用性

1.4.1 科学性

刚刚接触国防军工企业的时候，大家都说排产问题特别复杂，但是包括企业的专家在内，都只能给出一个模糊而粗略的描述。我们基于实地调研收集整理了大量数据，对问题进行定量分析，从众多因素中找到影响排产的主要原因，即跨车间协作问题。然后通过文献调研发现，这也是一个研究领域的科学问题。在学习当前研究成果的基础上我们基于超启发式算法设计了跨车间调度算法，并开发了智能优化排产软件。应用结果表明，在解决了跨车间协作问题之后，排产效率和效果大为改善，这表明我们的设计过程和作品本身都具有科学性。

1.4.2 先进性

本作品的先进性集中体现在核心调度算法上。

(1) 大幅提高了跨车间调度算法的计算效率。国内外公开发表的跨车间调度算法只适用于数十台机器、上百个零件的规模。而我们的算法在九百余台设备、七千余种零件、六万余道工序的问题规模下，可以在 3~15 分钟以内得到调度结果。

(2) 改进了跨车间调度算法的优化能力。与国内外公开发表的跨车间调度算法相比，我们的算法表现出更好的优化性能，零件延迟率降低 10% 以上，平均延迟时间降低 60% 以上，零件跨车间率降低 30% 以上。

产生的专利和论文等见成果清单及附件。

1.4.3 实用性

由于抓住了军工装备制造业的生产特点，解决了影响排产的主要问题，即跨车间协作问题，我们的作品在兵器一机集团取得了良好的应用效果。投入应用 1 年来，该集团比上一年多完成了 18% 的任务量，带来 7000 余万元的新增产值；在制品库存降

低 28%，节约成本 1000 余万元；另外在生产周期、设备利用率等方面也都有较大改善。

另外，我们的作品今年 5 月起在北奔投入试运行。

作品在军用车辆、航空航天、海洋工程等装备制造业具有广阔的应用前景，目前中船信息中心、航天五院等多家企业也向我们表达了应用意向。

相关应用证明见附件。

1.5 本章小结

本作品针对我国军工装备制造企业独特的跨车间协作问题设计并实现了一款专门的智能优化排产软件，其中的核心算法与国内外同类研究成果相比，具有更高的计算效率和更好的优化能力，尤其是对计算效率的提升，突破了相关成果投入实际应用的重要瓶颈。

第 2 章 基于超启发式的跨车间调度算法

2.1 引言

基于第1章的分析，现有的跨车间调度研究成果适用于小规模问题，与军工装备制造复杂产品数千个零件、数万道工序的问题规模相去甚远。要解决这样的大规模问题，通过传统的元启发式算法或精确求解算法是不行的。

在众多的调度方法中，分派规则因具有简单高效和灵活性而成为实际应用的首选。然而制造系统是复杂多变的，没有任何一个规则能在所有环境和评价标准下性能都优于别的规则，也就是说，分派规则对调度环境和调度目标具有强烈的依赖性^[17]。因此，一些研究提出了组合分派规则决策（Combinatorial Dispatching Rules Decision），即每个阶段使用不同的分派规则以获取更优的调度解性能^[18-20]。这些方法都可以归类为通过枚举策略求解合适的启发式规则组合，但是枚举策略导致计算开销剧增，难以在可接受时间内求得满意的规则组合。

为了快速的求得合适的规则组合，超启发式（Hyper-Heuristic）算法应运而生。超启发式算法分为高层启发式和底层启发式两部分。高层启发式一般采用元启发式算法，用来搜索合适的底层启发式规则。

我们首先以超启发式为框架，设计了启发式搜索遗传算法（Heuristic Search Genetic Algorithm, HSGA）。然后针对超启发式算法需要事先指定规则集的限制性，将遗传编程（Genetic Programming, GP）引入超启发式框架，用于生成规则集，对 HSGA 进行了改进。

2.2 问题的数学模型

本文研究的问题模型可描述如下： N 个零件在 k 个车间 ($k \geq 3$) 组成的单元制造系统（Cellular Manufacturing System, CMS）中加工。每个车间含有多台非等同（non-identical）的离散机（对应机加工序）或者批处理机（对应热处理工序），其中仅有一个车间只由批处理机构成。在批处理车间（通常进行热处理），多个零件将组成一批由批处理机同时加工，然后再经过后续的若干离散机阶段加工直到完成加工

任务。零件具有不同的到达时间和交货期。一个含有两个零件的三阶段 HFS 的调度实例如表 2.1 所示。

表 2.1 问题模型的一个调度实例

加工时间		机器						
		车间 1			车间 2		车间 3	
零件	工序	M1	M2	M3	B4	B5	M6	M7
零件 1	1	13	24	15	-	-	-	-
	2	-	-	-	135	145	-	-
	3	-	-	-	-	-	12	15
零件 2	1	15	8	27	-	-	-	-
	2	-	-	-	125	133	-	-
	3	-	-	-	-	-	22	17

M 表示离散机，B 表示批处理机

作品采用文献[21]中对问题模型的描述方法，用 $\alpha|\beta|\gamma$ 的三元组描述调度问题模型，其中 α 表示车间的结构信息，包括阶段数和每个阶段的机器数等； β 表示问题的约束和假设； γ 表示调度目标。公式 (2.1) 和公式 (2.2) 中的 α 和 β 是相同的，其中 HF_K 表示含有 K 个阶段的 HFS； $(PM^{(k)})_{k=1}^K$ 表示每个阶段都含有并行机； r_j 表示零件具有不同的到达时间； S_{snd} 表示准备时间与次序无关； $batch^{(k')}$ 表示阶段 k' 为批处理机。公式 (2.1) 和公式 (2.2) 中的 γ 部分（即调度目标）不相同，其中公式 (2.1) 中的 C_{max} 表示以最小化最大完工时间为目标，公式 (2.1) 中的 TWT 表示以最小化总加权延迟时间为目标。

$$HF_K, (PM^{(k)})_{k=1}^K | r_j, S_{snd}, batch^{(k')} | C_{max} \quad (2.1)$$

$$HF_K, (PM^{(k)})_{k=1}^K | r_j, S_{snd}, batch^{(k')} | TWT \quad (2.2)$$

在正式给出数学模型前，先定义以下符号变量。

索引：

- j 零件索引 ($j = 1, \dots, N$)
- m 机器索引 ($m = 1, \dots, M$)
- k 阶段索引 ($k = 1, \dots, K$)
- b 批次索引 ($b = 1, \dots, B$)

参数：

t_c	当前时间
t	时间片 ($t = 1, 2, \dots, T$)
CM_m	机器 m 的容量
p_{jkm}	零件 j 在阶段 k 机器 m 上的处理时间
p_{jk}	零件 j 在阶段 k 的处理时间
st_{jkm}	零件 j 在阶段 k 机器 m 上的生产准备时间
C_{jk}	零件 j 在阶段 k 的处理时间完工时间
d_j	零件 j 的交货期
r_j	零件 j 的到达时间
w_j	零件 j 的权重
O_{jk}	零件 j 在阶段 k 上处理的工序, 即零件 j 的第 k 道工序
TO_{jk}	零件 j 的工序 O_{jk} 的类型
TM_m	机器 m 的类型

类型变量:

$$CM_m = \begin{cases} 1, & \text{机器 } m \text{ 是离散机} \\ N^+ (\geq 2), & \text{机器 } m \text{ 是批处理机} \end{cases}$$

$$TO_{jk} = \begin{cases} 1, & O_{jk} \text{ 是批处理工序} \\ 0, & O_{jk} \text{ 是机加工序} \end{cases}$$

$$TM_m = \begin{cases} 1, & \text{机器 } m \text{ 是批处理机} \\ 0, & \text{机器 } m \text{ 是离散机} \end{cases}$$

决策变量:

$$X_{jkb} = \begin{cases} 1, & \text{如果零件 } j \text{ 在阶段 } k \text{ 被分配至批次 } b \\ 0, & \text{其他} \end{cases}$$

$$Y_{jkm} = \begin{cases} 1, & \text{如果零件 } j \text{ 在阶段 } k \text{ 在机器 } m \text{ 上加工} \\ 0, & \text{其他} \end{cases}$$

$$S_{jkt} = \begin{cases} 1, & \text{如果零件 } j \text{ 在阶段 } k \text{ 的加工在时刻 } t \text{ 开始} \\ 0, & \text{其他} \end{cases}$$

基于上述假设和符号定义，为我们要解决的调度问题模型建立一个数学模型。该问题模型具有两个目标函数，分别是 1) 最小化最大完工时间 (C_{max})；2) 最小化加权延迟总和 (Total Weighted Tardiness, TWT)。目标函数的数学表达式如公式 (2.3) 和公式 (2.4) 所示。

$$\min \sum_{j=1}^N \max\{C_{jk}\} \quad (2.3)$$

$$\min \sum_{j=1}^N (w_j \max\{C_{jk} - d_j, 0\}) \quad (2.4)$$

实际生产调度具有很多特有特征，需要受各种约束条件约束。问题模型的约束条件如下。

$$\sum_{m=1}^M Y_{jkm} = 1, \quad \forall j, k \quad (2.5)$$

$$(1 - TO_{jk}) + TO_{jk} \sum_{b=1}^B X_{jkb} = 1, \quad \forall j, k \quad (2.6)$$

$$\sum_{t=1}^T S_{jkt} = 1, \quad \forall j, k \quad (2.7)$$

$$\sum_{t=1}^T tS_{jkt} \geq r_j, \quad \forall j \quad (2.8)$$

$$(1 - TM_m) S_{jkt} Y_{jkm} \sum_{j'} \sum_{t'=t}^{t+p_{jk}} S_{j'kt'} Y_{j'km} = 0, \quad \forall j, k, m, \text{ and } j \neq j' \quad (2.9)$$

$$\sum_{j=1}^N Y_{jkm} X_{jkb} \leq CM_m, \quad \forall b \quad (2.10)$$

$$p_{jk} \geq (1 - TO_{jk}) \sum_{m=1}^M Y_{jkm} p_{jkm} + TO_{jk} \max \sum_{j=1}^N \sum_{m=1}^M X_{jkb} Y_{jkm} p_{jkm} \quad (2.11)$$

$$\sum_t^T tS_{j(k+1)t} \geq \sum_t^T tS_{jkt} + p_{jk} + \sum_{m=1}^M Y_{jkm} st_{jkm}, \quad \forall j, \text{ and } k = 1 \cdots K - 1 \quad (2.12)$$

$$X_{jkb} \sum_t^T tS_{jkt} = X_{j'kb} \sum_t^T tS_{j'kt}, \quad j \neq j', TO_{jk} = 1, \forall j, k, b \quad (2.13)$$

$$C_{jk} \geq \sum_t t S_{jkt} + p_{jk} + \sum_{m=1}^M Y_{jkm} s t_{jkm}, \quad \forall j, k \quad (2.14)$$

各约束表示的意思分别是：

约束（2.5）表示每个零件一次只能安排到一台机器上加工。

约束（2.6）表示零件在批处理阶段被安排的批次有且只有一个。

约束（2.7）表示零件在每个阶段都必须被处理。

约束（2.8）表示零件的加工不能早于它的到达时间。

约束（2.9）表示离散机每次仅能处理一个零件。

约束（2.10）表示批次大小不能超过批处理机的既定容量。

约束（2.11）表示零件在每个阶段的实际加工时间取决于当前阶段的工序类型。

对于离散工序，保持不变，而对于批处理工序，实际加工时间为组成同一批次的零件的最长加工时间。

约束（2.12）表示仅有当前阶段的加工完成后才能开始下阶段的加工。

约束（2.13）表示批处理阶段，属于同一批次的零件具有相同的开工时间。

约束（2.14）表示零件的完工时间不早于其开工时间和处理时间之和。

2.3 启发式搜索遗传算法

在生产调度领域，现有的超启发式算法可分为两类，分别是面向工序的^[14-15]和面向机器的^[12-13]。但对于本作品来说，这些算法不能直接应用。原因在于，在军工装备复杂产品的生产中，柔性加工路径非常普遍。也就是说，调度问题要同时解决指派和排序两个子问题。相应的，超启发式要为机器和零件同时搜索合适的启发式规则。本作品设计了一种启发式搜索遗传算法（Heuristic-Search Genetic Algorithm, HSGA），通过分段编码为机器和零件同时搜索启发式规则，再使用启发式规则生成调度解。不同于传统的元启发式算法，本算法的搜索空间是启发式规则，远小于调度解空间，因此具有良好计算效率，更适用于解决企业的大规模调度问题。

2.3.1 编码方案

在 HSGA 中，染色体表示方案包含了三个分段，即零件段，机器段和组批段，分别对应指派子问题，排序子问题和组批子问题。每个分段中的基因位是一个整数，表示一个启发式规则的索引号。考虑到环境信息对调度决策的影响，编码方案

的设计是同时面向零件（指派子问题）和机器（排序子问题和组批子问题）的，而不是面向阶段。这是因为尽管机器分布在同一个阶段，但是他们具有各自的环境信息，如机器容量，加工速度和缓冲区待调度零件数量等。类似的，零件也不尽相同，具有各自的到达时间和交货期，不同的加工路径等。因此，考虑多种差异因素，同时为零件和机器设计编码方案，而不是面向阶段，更能体现问题模型的特征和更好全局优化性能。

在 HSGA，候选规则集由 5 个指派规则，11 排序规则和 3 个组批规则构成。对于在批处理机上处理的实体（即批次），该批次的交货期，释放时间，到达时间分别由该批次中所含零件的最小交货期，最小释放时间和最小到达时间决定。与此类似，该批次的加工时间，生产准备时间，剩余加工时间和权重分别由该批次中所含零件的最大加工时间，最大生产准备时间，最大剩余加工时间和最大权重所决定。

(1) 候选指派规则如下所示：

- *First Available (FA)*: 零件被指派至最早可用的机器。
- *Least Utilization (LU)*: 零件被指派至利用率最低的机器。
- *Most Available (MA)*: 零件被指派至缓冲区零件数最少的机器。
- *Shortest Processing Time (SPT)*: 零件被指派至加工该零件时间最短的机器。
- *Earliest Finish Time (EFT)*: 零件被指派至最早可用时间与加工该零件时间的总和最短的机器。

(2) 候选排序规则如下所示，其中大部分是采用 Yang 等人^[16]定义的规则，另外新增加两个排序规则，即 *weighted shortest processing time* 和 *weighted earliest due date*，处理最小化 TWT 目标。

- *First in first out (FIFO)*: 选择最早释放到机器的实体。
- *Time in Shop (TIS)*: 选择在系统逗留时间最长的实体。系统逗留时间 tis_e 可按公式 (2.15) 计算。

$$tis_e = t_c - r_e \quad (2.15)$$

其中， r_e 表示实体 e 的到达时间。

- *Shortest Processing Time (SPT)*: 选择具有最短加工时间的实体。
- *Shortest Remaining Processing Time (SRPT)*: 选择具有最大剩余加工时间的实体。

● *Largest Estimated Flow Time (LEFT)*: 选择具有最大估算流程时间的实体。实体 e 在阶段 k 的估算流程时间 eft_{ek} 可按公式 (2.16) 计算。

$$eft_{ek} = t_c - at_{ekm} + rpt_{ek} \quad (2.16)$$

其中, at_{ekm} 表示实体 e 在阶段 k 到达机器 m 的释放时间。

● *Smallest Processing Time Ratio (SPTR)*: 选择具有最小加工时间比率的实体。实体 e 在阶段 k 的机器 m 加工时间比率 ptr_{ekm} 可按公式 (2.17) 计算。

$$ptr_{ekm} = \frac{p_{ekm}}{t_c - r_e} \quad (2.17)$$

其中, p_{ekm} 表示实体 e 在阶段 k 的机器 m 上的加工时间。

- *Earliest Due Date (EDD)*: 选择具有最早交货期的实体。
- *Minimum Slack (MS)*: 选择具有最小松弛的实体。
- *Critical Ratio (CR)*: 选择具有最小关键度的实体。
- *Weighted Shortest Processing Time (WSPT)*: 选择具有最小加权加工时间的实体。
- *Weighted Earliest Due Date (WEDD)*: 选择具有最小加权交货期的实体。

(3) 候选组批规则如下所示:

- *First In First Out (FIFO)*: 缓冲区零件按先来先到排序后组批。
- *Shortest Processing Time (SPT)*: 缓冲区零件按加工时间升序排序后组批。
- *Earliest Due Date (EDD)*: 缓冲区零件按交货期升序排序后组批。

为了便于描述问题, 给出一个具体例子。对于表 2.1 的调度实例, 一个可能的染色体编码如图 2.1 所示。在零件段染色体中, 基因位 i 的数值表示第 i 个零件的指派规则。例如, 这段染色体第 2 位的基因数值是 3, 则表示对第 2 个零件采用 LU 作为指派规则。机器段染色体和组批段染色体可同理解析。需要说明的是, 离散机仅对应排序规则, 而批处理机不但有排序规则还有组批规则。

→ 工件段 ←		→ 机器段 ←						→ 组批段 ←	
4	3	3	8	2	6	7	9	2	3
MA	LU	EDD	SPT	CR	SL	SPT	TIS	EDD	SPT

图 2.1 染色体一个可能编码示例

2.3.2 解码方案

本小节将介绍解码方案以及零件指派、组批和实体排序算法。

(1) 零件指派

为了便于描述指派算法，定义可指派零件集。

定义1 时刻 t 的可指派零件集 APS_t ，定义为满足一下其中一个条件的零件集合。

(1) O_{jk} 表示为零件 j 的下一道工序，如果 $k=1$ ，即 O_{jk} 是零件 j 的首道工序，则零件 j 属于 APS_t ;

(2) 如果 $O_{j,k-1}$ 已完工， $k=2, 3, \dots, K$ ，则零件属于 APS_t 。

让 $p(i)$ 表示 APS_t 中的第 i 个零件， $AssignmentRule_j$ 表示零件 j 的指派规则。零件指派的启发式算法描述如下：

零件指派启发式算法 (PAHA)

Step 1. 当 t 时刻有新零件到达或者零件的某道工序完工，令 $i = 1$ ，更新 APS_t ;

Step 2. 如果 i 大于 APS_t 所含零件数，则转 Step6;

Step 3. 对于所有可加工 $p(i)$ 的可选机器，根据 $AssignmentRule_j$ 选择最优的机器 m^* ;

Step 4. 把 $p(i)$ 分配至 m^* ;

Step 5. $i = i + 1$ ，转 Step2;

Step 6. 置 APS_t 为空集。算法结束。

(2) 组批

为了便于描述组批算法，定义候选零件集。

定义2 把已根据 PAHA 指派的零件集合定义为候选零件集 CPS_m 。

令 $BatchFormationRule_m$ 表示机器 m 的组批规则， $pos(j)$ 表示根据 $BatchFormationRule_m$ 排序后 CPS_m 中的第 j 个零件。组批的启发式算法描述如下：

组批启发式算法 (BFHA)

Step 1. 置批次集合 BU 为空集， $BU = \emptyset$ ，更新 CPS_m ;

Step 2. 根据 $BatchFormationRule_m$ 对属于 CPS_m 的零件排序;

Step 3. 依次从 CPS_m 中选择不超过 CM_m 个零件组成一个批次 b ,

$b = \{j \mid j \in CPS_m \text{ and } pos(j) \leq CM_m\}$, $BU = BU \cup \{b\}$, $CPS_m = CPS_m - \{j \mid j \in b\}$;

Step 4. 如果 $CPS_m \neq \emptyset$ ，转 Step3;

Step 5. 清空机器 m 的缓冲区，重新分配所有批次到 m ; 算法结束。

(3) 实体排序

为了便于描述实体排序算法，定义可调度实体集。

定义 3 把已根据 PAHA 或者 BFHA 分配到机器 m 的实体集定义为可调度实体集 SES_m 。

令 $SequencingRule_m$ 表示机器 m 的排序规则， e 表示待处理的可调度实体。实体排序的启发式算法描述如下：

实体排序启发式算法 (ESHA)

- Step 1. 更新 SES_m ;
- Step 2. 对于所有属于 SES_m 的实体 e ，根据 $SequencingRule_m$ 选择最佳的实体 e^* ;
- Step 3. 机器 m 调度实体 e^* ;
- Step 4. $SES_m = SES_m - \{e^*\}$ ，并从机器 m 的缓冲区删除 e^* ；算法结束。

(4) 染色体解码

因为 HSGA 是搜索启发式规则而不是调度解，因此需要设计一个离散事件模拟器 (Discrete Event Simulator, DES) 解码染色体中的启发式规则，然后利用这些启发式规则构建调度解，从而获得目标函数值。解码算法描述如下：

解码算法

- Step 1. 参数初始化，置 DES 时钟 $t = 0$;
- Step 2. 对于一条给定的染色体的不同分段，分别把排序规则分配至各机器，指派规则分配至各零件，组批规则分配至各批处理机；
- Step 3. 如果所有零件都已完工，转 Step10;
- Step 4. 对于每一个可分配零件 j ，运行 PAHA;
- Step 5. 如果机器 m 变空闲且 m 是批处理机，则转 Step6; 否则， m 是离散机，则转 Step7;
- Step 6. 运行 BFHA;
- Step 7. 根据 ESHA 调度一个实体;
- Step 8. 记录该实体调度工序的开工时间，完工时间和对应的加工机器;
- Step 9. $t = t + 1$ ，转 Step3;
- Step 10. 利用 Step8 中的记录信息，根据公式 (2.3) 和公式 (2.4) 计算相应的目标函数值。算法结束。

2.3.3 遗传算子

适应度函数是评价一个染色体的依据，适应度值越高，该染色体越好。本文采用公式 (2.18) 定义适应度函数：

$$fit(i) = \frac{1}{obj(i)+1} \quad (2.18)$$

其中， $obj(i)$ 为第 i 条染色体表示的调度解的目标函数值。

我们采用的轮盘赌方法是多种常见选择算子中应用最广泛的一种^[22]。设每条染色体的适应度值是 $fit(i)$ ，则染色体被选择的概率 $prob(i)$ 可按公式 (2.19) 计算：

$$prob(i) = \frac{fit(i)}{\sum_{i=1}^n fit(i)} \quad (2.19)$$

我们采用两点交叉算子。在满足一定交叉概率的情况下，对给出的两条染色体，先随机确定某条染色体的两个位置 $pos1$ 和 $pos2$ ，然后互换这两条染色体位于 $pos1$ 和 $pos2$ 之间的基因段。变异过程采用单点变异，即每段染色体的选定位基因随机替换成合法的规则索引号。

由于本文的染色体有三段不同含义的编码，因此需要分别对三段染色体进行遗传算子操作，这三段染色体共用一个交叉概率和变异概率。

2.4 基于遗传编程的改进算法

上一小节中介绍了 HSGA 的设计，但是 HSGA 的解性能易局限于所提供的规则集，且规则集的选取缺乏依据，因此，有必要设计优秀的规则集以提高 HSGA 的解性能。传统的方法是，根据领域知识人为地设计启发式规则，再通过具体实例仿真该启发式规则的有效性，如此不断循环优化，直到设计出满足性能要求的启发式规则。这个过程由于需要大量的人工参与而十分耗时，因此需要寻求一种具备自动化产生启发式规则的方法。遗传编程满足这种需求。

遗传编程是一种从生物演化过程得到灵感的自动化生成和选择计算机程序来完成用户定义的任务的技术。遗传编程适用于自动产生启发式，原因在于：

(1) 可变长的数据结构。对于一个具体的实际问题，我们难以估计或者确定究竟需要多长的数据结构才合适；

(2) 产生可执行的数据结构。启发式通常被表示为公式，甚至计算机程序或者算法；

(3) 容易从设定的输入变量中发现哪些特征对于问题的求解更具有贡献。

我们将 HSGA 扩展为三个阶段，即在原来的基础上运用 GP 产生启发式规则集，然后再通过 GA 在生成的规则集中选择合适的规则，最后应用规则组合求解调度问题。本小节将介绍如何通过 GP 生成规则集。

2.4.1 GP 流程

为了便于描述，在不产生混淆的情况下，使用术语 GP 规则指代那些利用 GP 产生的启发式规则。

我们的目标是产生合适的启发式规则。启发式规则是一种用来计算优先度的公式，因此非常适合于使用遗传编程生成，除此外，我们所设计的遗传编程算法还具有如下优势：

(1) 耗时的 GP 程序可离线计算。一旦产生了优质的 GP 规则，用于在线或者实时调度的时间花销与常见的规则是相当的；

(2) 因为产生 GP 规则过程是全自动化，并不需要太多的人工参与，因此节省了大量不断调整输入所带来的额外时间开销；

(3) GP 规则可容易地集成到现有的调度算法或者生产制造软件系统中。

GP 算法的典型流程可见图 2.2。

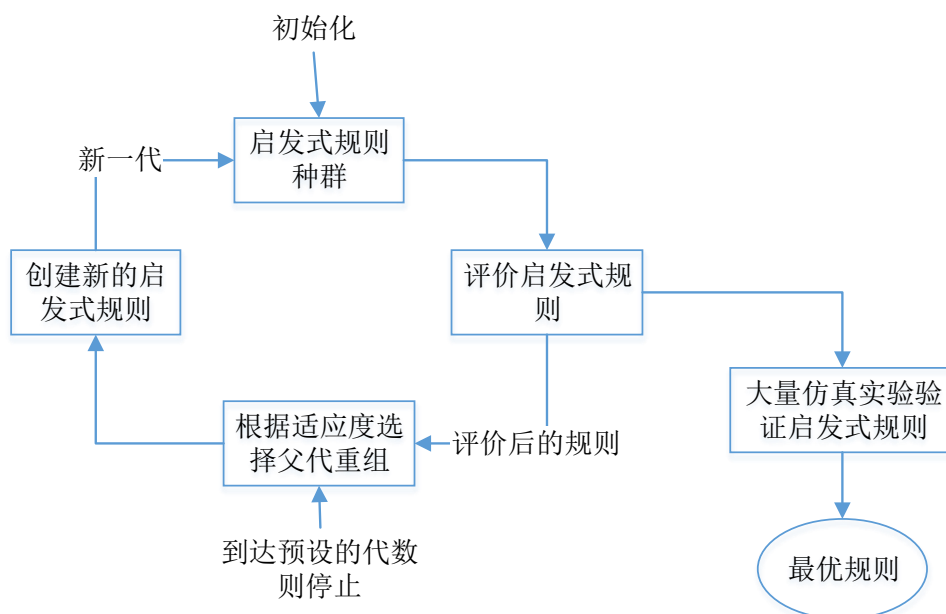


图 2.2 GP 算法的流程

设计一个 GP 算法有几个预置步骤需要完成：

- (1) 表示方案;
- (2) 端点集 (语法树的叶子结点集合), 一般包括常量、随机数和程序输入;
- (3) 函数集 (语法树的非叶子结点集合);
- (4) 适应度的评价方式;
- (5) 算法运行参数。

下文将针对这些步骤描述本文的遗传算法的具体实现。

2.4.2 表示方案

表示方案关注的是如何由零件或者机器的不同信息构建复合启发式规则 (Composite Heuristic Rule), 即 GP 规则。我们采用语法树的数据结构表示一个具体的 GP 规则。语法树的叶子节点都是输入变量, 这些变量与零件或者机器的信息直接相关, 非叶子节点表示的是一个函数。图 2.3 表示的是一个 GP 规则的编码实例, 其中 AT 表示零件的到达时间, W 表示零件的权重, PT 表示零件的加工时间, DD 表示零件的交货期, 则图 2.3 的语法树可等价地表述为公式 (2.20)。

$$\frac{W}{PT + DD} + AT \quad (2.20)$$

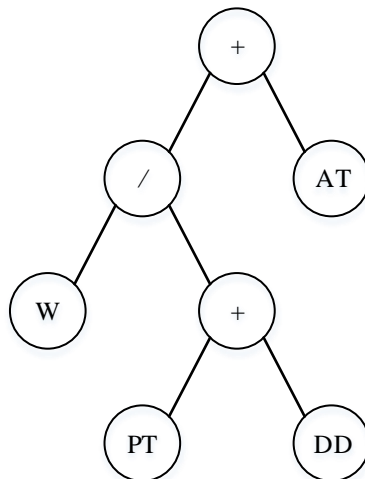


图 2.3 启发式规则对应语法树的示例

2.4.3 参数设置

(1) 函数集

定义了语法树表示方案后，下一步需要关注的是如何定义语法树的叶子结点和非叶子结点。如前所述，叶子结点对应的是启发式规则的组成成分或输入变量，即零件到达时间 AT 和零件的权重 W 等。非叶子结点对应的是启发式规则的函数。此处的函数一般指算术运算函数，如四则基本运算加减乘除，三角函数 \sin 、 \cos 或 \tan ，非线性函数 \max 、 \min 或 if 等，根据领域需求而定义。通过组合运算函数和输入变量，就能表达定义域内的任何一条公式。中序遍历语法树，依次递归计算各子树的值，最终能求得该语法树根节点的值，即启发式规则的对应的函数值。

我们考虑的函数集包含四个基本的算术运算函数，分别是加法 (+)、减法 (-)、保护除法 (/) 和乘法 (*)。当除数为零时，会引致除法非法，程序会因异常而退出，因此修改原始除法为保护除法。在保护除法中，当检测到除数为零时，不执行除法操作，而是直接返回 1。

(2) 端点集

在 GP 中，端点集对应的是语法树的叶子结点，一般为常量，随机数或者公式的组成部分。在本文所研究的调度问题中，端点集将会是与零件或者机器相关的属性。因为我们的问题划分为排序、指派、组批三个子问题，它们的端点集分别如表 2.2、表 2.3 和表 2.4 所示。

表 2.2 用于排序子问题的端点集

记号	描述
AT	实体到达系统的时间
DD	实体的交货期
B	实体的满批率 (实体所含零件数/加工机器容量)
PT	实体的加工时间
RT	实体到达当前机器的释放时间
OP	实体剩余工序数
RPT	实体剩余加工时间
W	实体的权重
#	服从[0,1]均匀分布的常量

表 2.3 用于指派子问题的端点集

记号	描述
L	机器的缓冲区零件数量
NIT	机器的下一次空闲时间
PT	机器加工该零件的时间
#	服从[0,1]均匀分布的常量

表 2.4 用于组批子问题的端点集

记号	描述
<i>DD</i>	零件的交货期
<i>PT</i>	零件在当前机器的加工工时
<i>RT</i>	零件的释放到当前机器的时间
<i>W</i>	零件的权重
#	服从[0,1]均匀分布的常量

(3) 参数表

我们所设计的学习启发式规则的 GP 算法是基于 ECJ20 库（一个关于进化计算的 java 开源库）^[23]开发的。后文中所使用到的 GP 算法的参数设置如表 2.5 所示。

表 2.5 GP 算法的参数设置

参数	描述
种群大小	1024
交叉概率	90%
变异概率	10%
进化代数	51
选择算子	锦标赛（大小为 17）
初始化	Ramped half-and-half（最小深度 2，最大深度 6）
交叉最大深度	17
端点集	见表 2.2，表 2.3 和表 2.4
函数集	+, -, *, /
适应度	$fit(\Delta)$ （见 2.4.4 节）

种群大小设为 1024 保证了足够的多样性。种群的初始化常用的有三种方法，分别是全深度（Full）、增长深度（Grow）和 Koza^[24]提出的 ramped-half-and-half。假设语法树的最大深度为 H，那么对于 Full 初始化方法，所有的叶子节点都具有相同的深度 H；对于 Grow 初始化方法，叶子节点的深度不超过 H；Ramped-half-and-half 是两者的结合，种群中一半的个体使用 Full 初始化，另一半的个体使用 Grow 初始化。这是因为不管是 Full 还是 Grow 初始化方法，都无法保证在初始化的时候提供一个足够多样化（具体来说就是树的大小和深度）的种群。由于启发式规则是基于图 2.4 的语法树创建的，本文通过强类型约束和严格的参数个数约束保证每个 GP 节点的返回值符合语法树的要求。因此，在不违反语法约束的前提下，交叉算子和变异算子才会被执行。对于交叉算子，GP 算法使用子树交叉^[25]，通过任意组合两个被选中的父代的子树产生下一代的新个体。对于变异算子，GP 算法使用子树变异^[25]，通过随机选

择一个个体的任意节点，然后由以该节点为根节点的一个新子树替换。这两个的遗传算子的操作实例如图 2.4 所示。

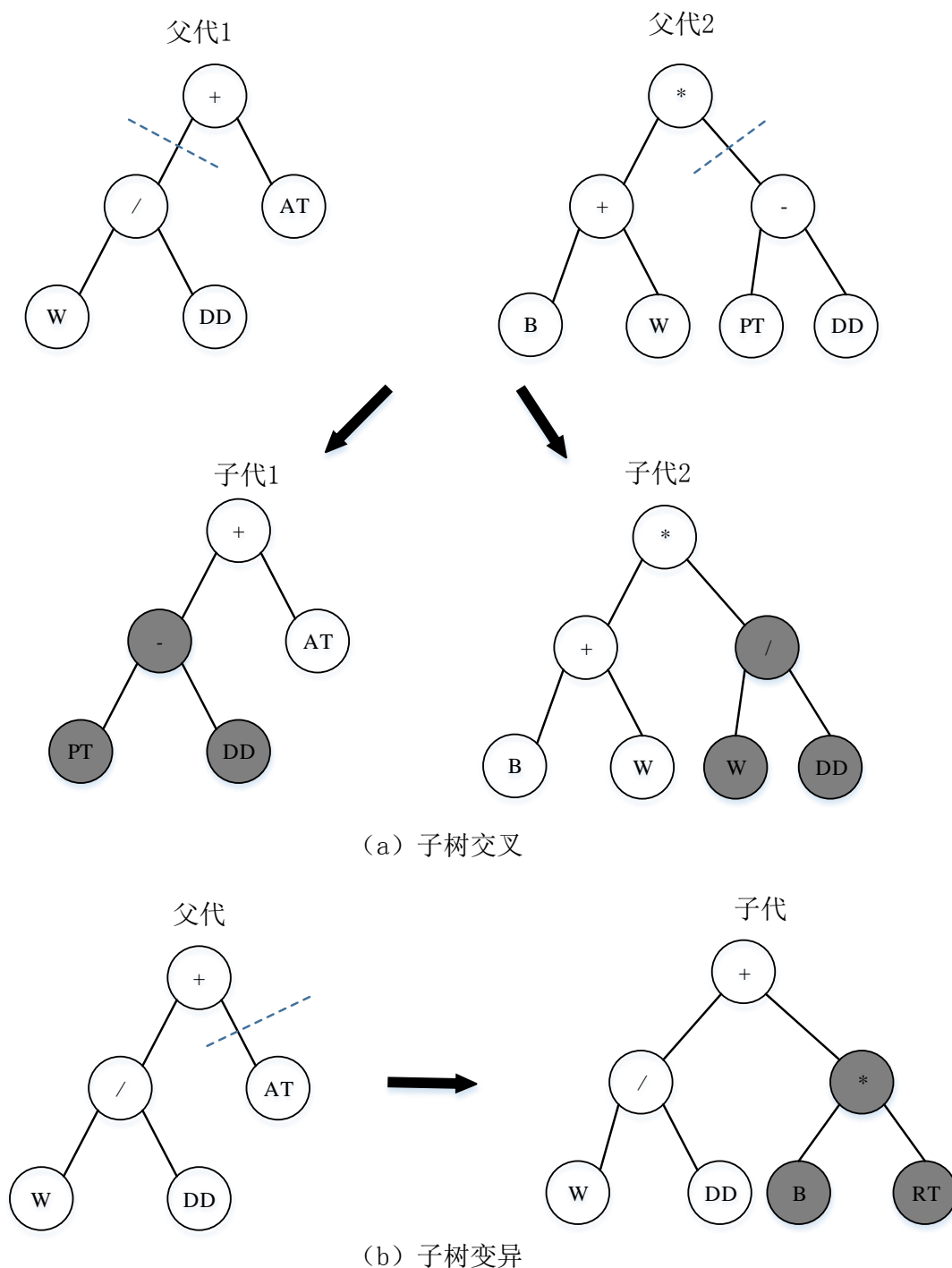


图 2.4 GP 算法子树交叉和子树变异的示例

不管是种群初始化、交叉算子或者变异算子过程中产生的新个体，均受语法树最大深度约束。种群初始化的最大树深为 6，当一个新个体的语法树深度超过 6，该

个体将会被舍弃，重复生成一个新个体，直至符合深度限制。交叉算子对语法树的深度限制为 17，如果通过交叉产生的新个体语法树深度超过 17，则放弃本次交叉操作，保持两个父代不变。增大语法树的深度能使 GP 算法的搜索空间增大而更易于进化出更优的 GP 规则，但是深度的增加意味着计算时间的开销增大，而且会导致进化的 GP 规则过于庞大难以分析，且容易产生过拟合问题。因此本文设置为 17。选择算子使用的大小为 17 的锦标赛选择方法。

2.4.4 适应度评价

为了评估一个 GP 规则的有效性，每个 GP 规则将会求解训练集中的所有实例，并且得到各实例的目标函数值。由于 GP 求解每个实例的目标函数值差距较大（存在数量级上差距），难以直接比较各目标函数值之间的差异，因此通过计算目标函数值与参考值的偏差，以此偏差作为评价该目标函数值的优异程度。偏差可按公式 (2.21) 算得。

$$dev(\lambda, Ins_i) = \frac{obj(\lambda, Ins_i) - Ref(Ins_i)}{Ref(Ins_i)} \quad (2.21)$$

其中， $obj(\lambda, Ins_i)$ 表示应用 GP 规则 λ 求解实例 Ins_n 的目标函数值； $Ref(Ins_i)$ 是实例 Ins_i 的参考目标函数值。

在公式 (2.21) 的基础上，规则 Δ 在训练集上的适应度可按公式 (2.22) 计算。

$$fit(\lambda) = dev_{avg}(\lambda) = \frac{\sum_{i=1}^I dev(\lambda, Ins_i)}{I} \quad (2.22)$$

适应度函数 $fit(\lambda)$ 评价规则在训练集上的平均表现性能。在本文中， $Ref(Ins_i)$ 的值均取自第 2.3 节 HSGA 所求得解。

2.4.5 整体算法

GP 算法的整体算法描述如下：

启发式规则产生算法

输入：训练集 $D = \{Ins_1, Ins_2, \dots, Ins_I\}$

输出：最优 GP 规则 λ^*

Step 1. 随机初始化种群 $P = \{\lambda_1, \dots, \lambda_S\}$;

Step 2. 令 $\lambda^* = null$, $fit(\lambda^*) = +\infty$;

- Step 3. 进化代数 $gen = 0$;
- Step 4. 如果 $gen > MaxGeneration$, 转 Step10;
- Step 5. 对于所有 $\lambda_i \in P$ 和 $Ins_i \in D$, 利用 λ_i 求解 Ins_i , 并根据公式 (2.21) 计算 $dev(\lambda_i, Ins_i)$;
- Step 6. 根据公式 (2.22) 计算 $fit(\lambda_i)$;
- Step 7. 如果 $fit(\lambda_i) < fit(\lambda^*)$, 则 $\lambda^* = \lambda_i$, $fit(\lambda^*) = fit(\lambda_i)$;
- Step 8. 执行选择算子、交叉算子和变异算子产生新的种群 P' , $P = P'$;
- Step 9. $gen = gen + 1$;
- Step 10. 输出 λ^* 。算法结束。

2.5 HSGA 的性能分析

由于目前没有 benchmark 可以借鉴, 我们设计了多组不同规模的测试问题 (test problem) 来验证算法的有效性。仿真实验使用 Java 语言实现 HSGA, 运行在 Core i7-2600 3.40GHz, 4G 内存的 PC 机上。

2.5.1 实验设计

由于现有算法的问题模型均较小, 我们首先设计了规模较小的测试问题, 用以在同等条件下与现有算法进行比较。然后再用真实数据对算法进行验证。

根据不同的机器设置和零件设置, 设计了 18 个不同规模的测试问题, 每个规模下随机产生 10 个算例 (instance), 共有 180 个不同的算例。产生算例的参数设置如表 2.6 所示。每个测试问题按照表 3 参数随机产生 10 个不同的算例。每个算例进行 5 次独立的仿真实验。10 个算例的目标函数值的均值作为此测试问题的性能指标。零件的交货期按式 (2.23) 设置:

$$d_j = r_j + df \sum_{k=1}^K p_{jk} \quad (2.23)$$

其中 df 是交货期因子, 表示交货期的紧张程度, 默认取 3; $\sum_{k=1}^K p_{jk}$ 表示零件 j 在阶段 k 所有可选设备上的总工时的均值。

表 2.6 算例产生的参数表

算例产生参数	取值分布
每个阶段机器数	$\sim U [2,4]$
批处理机容量 CM_m	$\sim U [2,5]$

零件到达时间 r_j		$\sim U [0,50]$
零件权重 w_j		$\sim U (0,1)$
批处理机工序	准备时间 st_{jkm}	$\sim U [10,35]$
	处理时间 p_{jkm}	$\sim U [100,200]$
离散机工序	准备时间 st_{jkm}	$\sim U [5,10]$
	处理时间 p_{jkm}	$\sim U [1,30]$

不同规模的测试问题含有的零件数从 10 到 95 不等，机器数从 8 到 21 不等，车间数从 3 到 7 不等。每个测试问题采用 $j_{n1}m_{n2}s_{n3}$ 的记法，例如 $j3m4s5$ 表示该测试问题是一个 5 车间含有 4 台机器需要调度 3 个零件的问题。

2.5.2 参数分析

遗传算法性能很大程度上依赖于算法参数的设置，因此需要评价不同的参数组合，以确定算法最优参数组合。本文考虑的 HSGA 参数共有四个，分别是种群大小、交叉概率、变异概率和进化代数，并进行全析因设计（full factor design）实验，每个参数作为一个因子（factor），每个因子的水平（treatment）如表 2.7 所示。

表 2.7 HSGA 参数

因子	种群大小 ps	交叉概率 pc	变异概率 pm	进化代数 gm
水平	6,12,24,48	0.05, 0.30, 0.60, 0.90	0.00, 0.02, 0.10, 0.18	25, 50, 100

在置信度水平 95% 下，采用方差分析法（ANalysis Of VAriance, ANOVA）^[26] 分析共 192 个不同参数组合的 HSGA 的性能表现，从而确定最优算法参数组合。

定义响应变量（responsible variable, RV）为 HSGA 的目标函数值相对于基准值的平均性能提升百分比，如公式（2.24）所示：

$$RV = \frac{\sum_{i=1}^{18} \frac{HSGA_{ni} - Ref_i}{Ref_i} \times 100}{18} \quad (2.24)$$

其中 $HSGA_{ni}$ 是第 n 组算法参数下求解第 i 个算例的目标函数值； Ref_i 是求解第 i 个算例的目标函数值的基准值。定义基准值为采用标准参数设置的 HSGA（ $ps=50$ 、

$pc=0.8$ 、 $pm=0.1$ 、 $gm=500$)的目标函数值。基准值的获取方法参考 Ruiz 和 Maroto^[24]。从公式 (2.24) 可知, 响应变量越小, HSGA 解性能越优。

在最小化 C_{max} 目标下, ANOVA 对全析因实验的分析结果如表 2.8 所示。从表 2.8 可以看出, 所有单因子都是显著的 (P 值 ≤ 0.05)。通过比较 F 值大小, 可以识别出对 HSGA 影响最大的因子或者交互因子。每个单因子主效应如图 2.5 所示, 随着 ps 、 pm 、 gm 的增大, HSGA 获得更优的解性能, 而 HSGA 在 $pc=0.6$ 的时获得最优的解性能, 更大或者更小 pc 值会导致解性能下降。另外, 从表 2.9 可知因子组合 $ps*pc$, $ps*pm$, $ps*gm$, $pm*gm$ 间的交互作用存在显著性, 因此还需要分析因子间交互作用对 HSGA 解性能的影响。二阶因子交互作用如图 2.6 所示。根据表 2.8 的 F 值可知因子组合 $ps*pm$ ($F=67.27$) 对解性能的影响最大, 以 $ps*pm$ 为例分析参数的取值。因为 pm ($F=360.05$) 较 ps ($F=261.64$) 对解性能的影响更大, 因此先确定 pm 的值。从图 2.5 可知 $pm=0.18$ 时具有最优解性能, 而 $ps=48$ 时具有最优解性能。这与图 2.6 的结果是一致的。同理, 对于 $ps*gm$, $ps*pc$ 可知 $gm=100$, $pc=0.6$ 时, HSGA 取得最优解性能。综合以上分析, 在最小化 C_{max} 目标下, 本文将 HSGA 的参数取值定为: 种群大小 $ps=48$, 交叉概率 $pc=0.6$, 变异概率 $pm=0.18$, 进化代数 $gm=100$ 。

表 2.8 最小化 C_{max} 目标下的 ANOVA 表

Source	Df	Seq SS	Adj SS	Adj MS	F	P
ps	3	153.515	153.515	51.172	261.64	0.000
pc	3	2.980	2.980	0.993	5.08	0.002
pm	3	211.254	211.254	70.418	360.05	0.000
gm	2	45.784	45.784	22.892	117.05	0.000
$ps*pc$	9	4.870	4.870	0.541	2.77	0.005
$ps*pm$	9	118.403	118.403	13.156	67.27	0.000
$ps*gm$	6	25.303	25.303	4.217	21.56	0.000
$pc*pm$	9	1.555	1.555	0.173	0.88	0.542
$pc*gm$	6	1.228	1.228	0.205	1.05	0.398
$pm*gm$	6	22.952	22.952	3.825	19.56	0.000
误差	135	26.403	26.403	0.196		
合计	191	614.246				
S = 0.442242 R-Sq = 95.70% R-Sq (调整) = 93.92%						

表 2.9 最小化 TWT 目标下的 ANOVA 表

Source	Df	Seq SS	Adj SS	Adj MS	F	P
ps	3	9558.40	9558.40	3186.13	5825.42	0.000
pc	3	1179.42	1179.42	393.14	718.80	0.000
pm	3	672.25	672.25	224.08	409.71	0.000

<i>gm</i>	2	321.26	321.26	160.63	293.69	0.000
<i>ps*pc</i>	9	19.86	19.86	2.21	4.04	0.000
<i>ps*pm</i>	9	145.01	145.01	16.11	29.46	0.000
<i>ps*gm</i>	6	45.94	45.94	7.66	14.00	0.000
<i>pc*pm</i>	9	128.05	128.05	14.23	26.01	0.000
<i>pc*gm</i>	6	2.99	2.99	0.50	0.91	0.490
<i>pm*gm</i>	6	58.09	58.09	9.68	17.70	0.000
误差	135	73.84	73.84	0.55		
合计	191	12205.11				
S = 0.739551 R-Sq = 99.40% R-Sq (调整) = 99.14%						

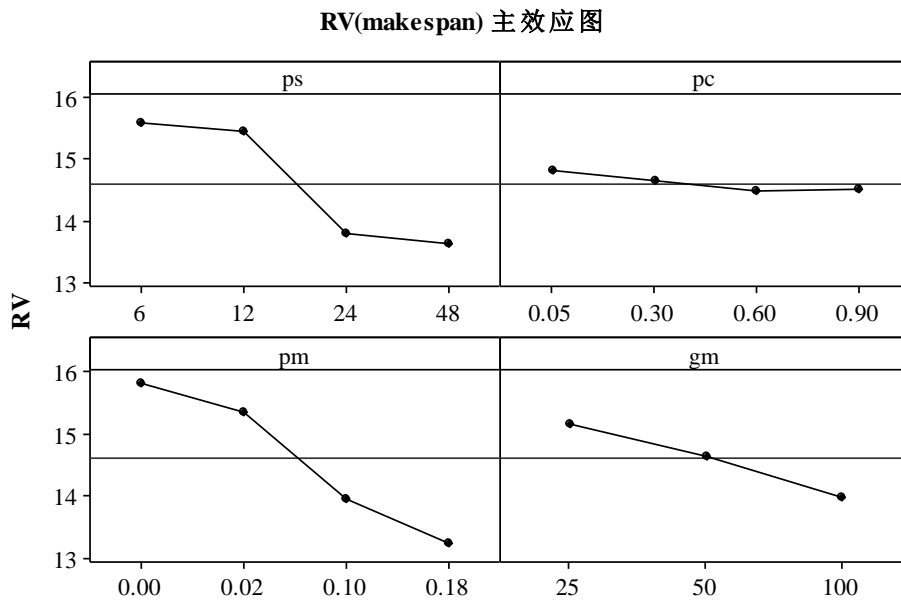


图 2.5 最小化 C_{max} 目标下的单因子主效应

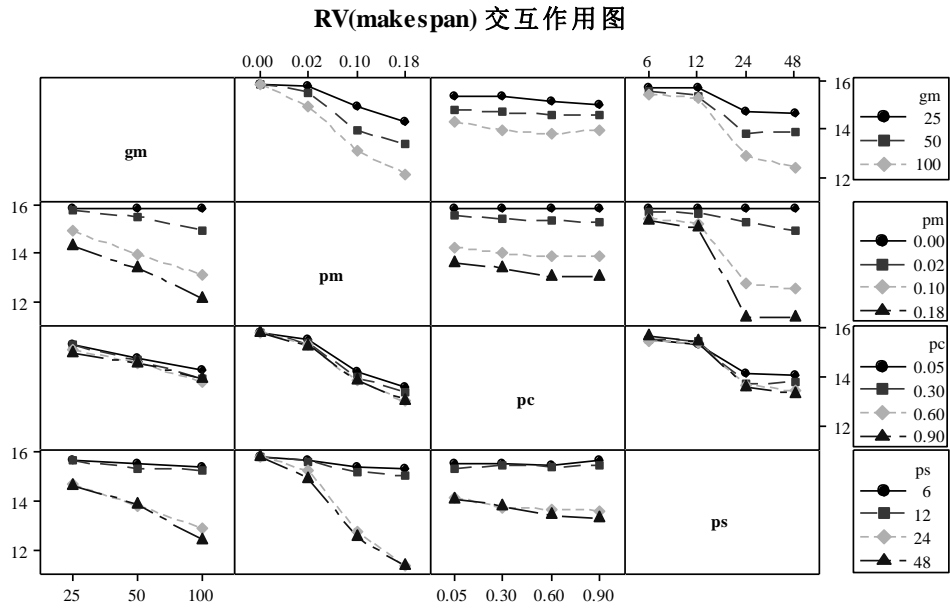


图 2.6 最小化 C_{max} 目标下的因子交互作用图

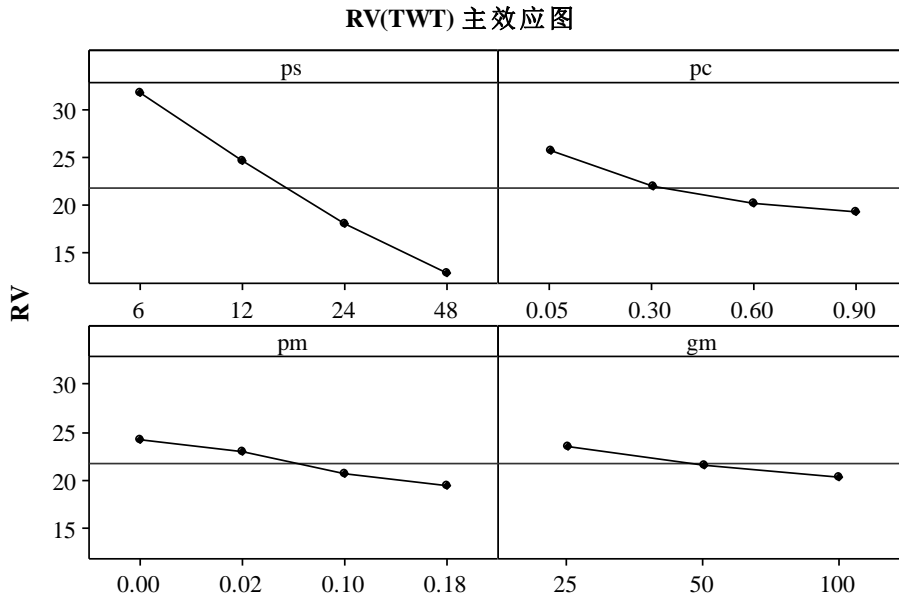


图 2.7 最小化 TWT 目标下的主效应图

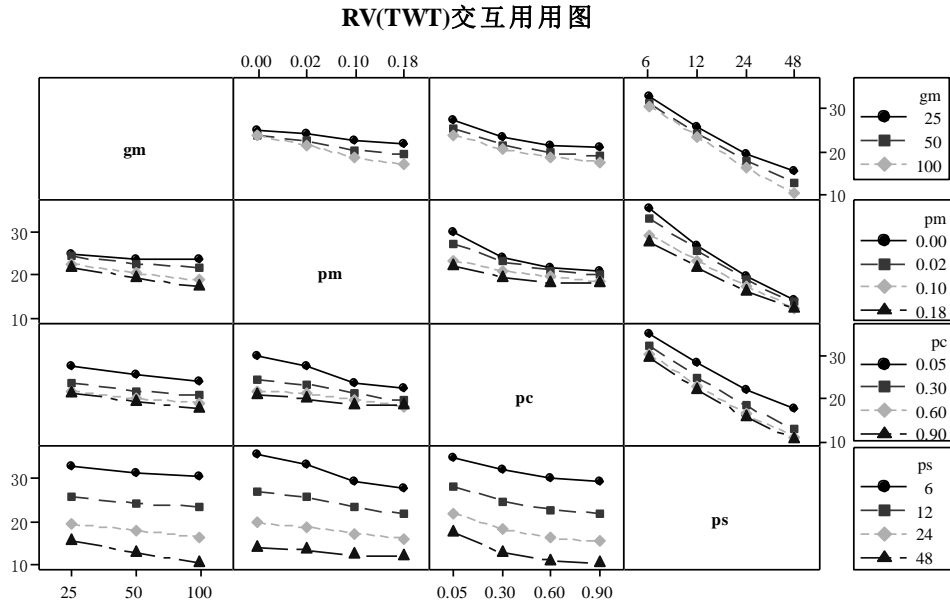


图 2.8 最小化 TWT 目标下的因子交互作用图

同理，在最小化 TWT 目标下，可对表 2.9、图 2.7 和图 2.8 作类似的分析，最后得出参数的取值为种群大小 $ps=48$ ，交叉概率 $pc=0.9$ ，变异概率 $pm=0.18$ ，进化代数 $gm=100$ 。在没有特殊说明的情况下，本节分析得出的参数组合将作为 HSGA 的默认参数设置。

2.5.3 性能比较实验

确定 HSGA 算法参数后，需要验证算法的有效性并分析其效率。本文将从以下三个方面验证算法的有效性：（1）与多个组合规则进行性能比较；（2）不同编码方式之间的性能比较；（3）与 CPLEX 比较。

（1）与组合规则比较

在所有的组合规则中，本文共考虑了 165 种组合规则，分别在最小化 C_{max} 和最小化 TWT 的目标下对每种组合规则进行仿真实验。通过分析实验结果，选取平均性能最优的 5 种组合规则（如表 2.10 所示），与本文提出 HSGA 算法进行对比实验，实验结果如表 2.11 和表 2.12 所示。HSGA 在两种目标下均比组合规则具有更优的解性能。在最小化 C_{max} 目标下，HSGA 较组合规则平均提升了 23.53%；在最小化 TWT 目标下，HSGA 较组合规则平均提升了 55.98%。这表明了利用 HSGA 搜索不同排序、指派和组批规则组合的机制的有效性。

表 2.10 不同调度目标下比对实验的组合规则

最小化 C_{max}				最小化 TWT			
排序规则	指派规则	组批规则	记作	排序规则	指派规则	组批规则	记作
SPTR	MA	SPT	SPMS	SRPT	SPT	SPT	SRSS
WSPT	MA	SPT	WSMS	SPT	SPT	SPT	SSS
MS	MA	EDD	MME	SPTR	SPT	SPT	SPSS
SPT	MA	SPT	SMS	WSPT	SPT	SPT	WSSS
MS	MA	FIFO	MMF	SRPT	SPT	EDD	SRSE

表 2.11 最小化 C_{max} 目标下 HSGA 与组合规则的性能比较

测试问题	HSGA	SPMS	WSMS	MME	SMS	MMF	AVG_GAP%
j10m8s3	377.32	502.20	502.10	420.50	495.20	424.00	24.24
j15m8s3	431.16	577.40	581.90	522.50	597.90	509.40	29.38
j20m11s3	430.64	630.80	649.70	529.20	629.30	528.90	37.84
j25m11s3	523.98	698.60	674.60	597.00	683.60	598.40	24.13
j30m7s3	933.54	1185.90	1191.50	1122.70	1183.90	1110.10	24.13
j35m7s3	1096.22	1370.30	1358.00	1251.60	1392.00	1238.70	20.61
j40m13s5	1075.18	1281.10	1293.80	1155.60	1276.70	1148.00	14.50
j45m13s5	1170.6	1333.00	1360.60	1273.30	1359.20	1268.60	12.67
j50m15s5	1253.86	1728.90	1786.70	1655.10	1765.70	1642.90	36.85
j55m15s5	1347.7	1752.40	1792.10	1717.10	1775.50	1745.60	30.34
j60m16s5	1136.94	1495.90	1492.30	1410.20	1463.90	1421.90	28.14
j65m16s5	1225.9	1552.20	1586.70	1394.70	1549.60	1401.10	22.10
j70m20s7	1293.78	1559.20	1622.30	1468.80	1541.10	1499.70	18.89
j75m20s7	1365.2	1706.30	1706.00	1879.00	1650.20	1935.80	30.05
j80m21s7	1902.6	2224.50	2226.30	1955.70	2250.80	1958.10	11.59
j85m21s7	2022.9	2295.40	2332.50	2190.10	2299.50	2188.60	11.78
j90m21s7	2110.26	2664.80	2584.50	2597.10	2651.20	2600.10	24.13
j95m21s7	2225.58	2747.00	2803.80	2644.60	2724.60	2668.10	22.11
平均值	1217.96	1516.99	1530.30	1432.49	1516.11	1438.22	23.53

表 2.12 最小化 TWT 目标下 HSGA 与组合规则的性能比较

测试问题	HSGA	SRSS	SSS	SPSS	WSSS	SRSE	AVG_GAP%
j10m8s3	0	0	0	1.259	2.638	0	-
j15m8s3	0	0.102	2.021	2.021	0	0	-
j20m11s3	0	0.425	0.425	0.74	0	0	-
j25m11s3	0	30.169	41.99	48.741	43.017	20.152	-
j30m7s3	636.2132	1413.603	1513.809	1441.691	1456.434	1256.42	122.63
j35m7s3	1328.803	2185.576	2325.664	2196.65	2470.435	2183.211	71.00
j40m13s5	847.164	1468.471	1468.645	1499.465	2238.405	1575.993	94.79
j45m13s5	1842.959	2439.28	2404.363	2573.137	2796.57	2375.148	36.61
j50m15s5	2255.434	4272.223	4355.789	4695.691	4826.448	4595.268	101.69
j55m15s5	3294.652	4682.162	4661.971	5190.456	5854.681	5355.182	56.28

<i>j60m16s5</i>	1820.739	2890.945	2952.328	3359.639	3333.276	3112.549	71.89
<i>j65m16s5</i>	2496.315	3329.604	3226.955	3537.037	3750.856	3348.792	37.75
<i>j70m20s7</i>	2760.782	3440.448	3475.803	4169.929	4004.391	3075.313	31.60
<i>j75m20s7</i>	3997.12	6535.316	6701.998	7698.292	7702.756	6885.486	77.75
<i>j80m21s7</i>	10801.1	12513.7	12356.01	12284.69	14866.03	14080.91	22.40
<i>j85m21s7</i>	14457.78	16133.4	16279.15	15995.86	19034.6	18372.18	18.71
<i>j90m21s7</i>	15087.08	17775.28	17680.93	19268.6	20156.03	19044.72	24.51
<i>j95m21s7</i>	18528.34	19734.21	20010.27	22218.46	22705.28	22887.25	16.10
平均值	4453.03	5491.38	5525.45	5899.02	6402.33	6009.37	55.98

(2) 不同编码方式的比较

本节将 HSGA 的解性能与 Yang 等人^[16]按阶段编码的 GA 搜索组合规则的方法进行比较, 将其算法记为 GA_Y。由于 Yang 等人的实验中得出的最优参数设置中种群大小为 24, 最大进化代数为 8, 这两个参数均远小于本文所设置的值, 为避免由于运行时间不均等导致有失公平性, 按照参数分析得出的结果设置 GA_Y 的种群大小和最大进化代数。另外, 由于 Yang 等人对调度数据中的组批情况是给定的, 且未提及对零件指派问题的解决方法, 在对比试验中, GA_Y 的组批问题和指派问题均采用本文提出的方案。

GA_Y 令 T_i 表示第 i 条染色体对应的调度解的目标函数值, F_i 表示适应度函数, 其定义公式 (2.25) 所示:

$$F_i = \max \{T_i\} - T_i \quad (2.25)$$

Yang 等人认为使用 F_i 更易于分辨出好的染色体。但仿真结果显示, 此适应度函数性能表现并不理想, 因此根据公式 (2.18) 对 GA_Y 进行改进, 改进后的算法记作 GA_YI, 实验结果如表 2.13 和表 2.14 所示。与 GA_Y 的适应度函数相比, 本文的适应度函数使 GA_YI 的求解性能在最小化 C_{max} 目标下平均提升了 8.33%, 在最小化 TWT 目标下平均提升了 44.79%, 表明本文的适应度函数更适合解决此类问题。在 C_{max} 和 TWT 两个目标下, HSGA 均优于 GA_YI, 在最小化 C_{max} 目标下, 平均提升了 2.63%, 而在最小化 TWT 目标下, 平均提升了 15.70%。

表 2.13 最小化 C_{max} 目标下 HSGA 与 GA_Y, GA_YI 的比较

测试问题	GA_Y	GA_YI	HSGA	GA_YI 与 GA_Y 的 GAP%	HSGA 与 GA_YI 的 GAP%
<i>j10m8s3</i>	404.37	381.03	377.32	6.12	0.98
<i>j15m8s3</i>	488.10	448.07	431.16	8.93	3.92
<i>j20m11s3</i>	483.63	456.97	430.64	5.84	6.11
<i>j25m11s3</i>	598.67	526.27	523.98	13.76	0.44

<i>j30m7s3</i>	1000.37	950.23	933.54	5.28	1.79
<i>j35m7s3</i>	1161.47	1120.53	1096.22	3.65	2.22
<i>j40m13s5</i>	1158.37	1101.10	1075.18	5.20	2.41
<i>j45m13s5</i>	1270.90	1195.73	1170.60	6.29	2.15
<i>j50m15s5</i>	1485.80	1316.00	1253.86	12.90	4.96
<i>j55m15s5</i>	1582.20	1409.40	1347.70	12.26	4.58
<i>j60m16s5</i>	1274.97	1173.60	1136.94	8.64	3.22
<i>j65m16s5</i>	1391.00	1275.23	1225.90	9.08	4.02
<i>j70m20s7</i>	1481.90	1312.23	1293.78	12.93	1.43
<i>j75m20s7</i>	1577.30	1393.50	1365.20	13.19	2.07
<i>j80m21s7</i>	2034.50	1918.23	1902.60	6.06	0.82
<i>j85m21s7</i>	2183.80	2048.17	2022.90	6.62	1.25
<i>j90m21s7</i>	2307.90	2152.10	2110.26	7.24	1.98
<i>j95m21s7</i>	2428.13	2291.80	2225.58	5.95	2.98
平均值	1350.74	1248.34	1217.96	8.33	2.63

表 2.14 最小化 TWT 目标下 HSGA 与 GA_Y, GA_YI 的比较

测试问题	GA_Y	GA_YI	HSGA	GA_YI 与 GA_Y 的 GAP%	HSGA 与 GA_YI 的 GAP%
<i>j10m8s3</i>	0	0	0	-	-
<i>j15m8s3</i>	0	0	0	-	-
<i>j20m11s3</i>	0	0	0	-	-
<i>j25m11s3</i>	0	0	0	-	-
<i>j30m7s3</i>	1166.37	833.86	636.21	39.88	31.07
<i>j35m7s3</i>	2081.01	1422.74	1328.80	46.27	7.07
<i>j40m13s5</i>	1790.06	1053.63	847.16	69.89	24.37
<i>j45m13s5</i>	3038.28	2077.50	1842.96	46.25	12.73
<i>j50m15s5</i>	4156.73	2953.87	2255.43	40.72	30.97
<i>j55m15s5</i>	5935.80	3821.84	3294.65	55.31	16.00
<i>j60m16s5</i>	3752.39	2100.06	1820.74	78.68	15.34
<i>j65m16s5</i>	4365.95	2946.77	2496.32	48.16	18.04
<i>j70m20s7</i>	5233.14	3230.44	2760.78	61.99	17.01
<i>j75m20s7</i>	7214.17	4727.10	3997.12	52.61	18.26
<i>j80m21s7</i>	13636.13	11645.46	10801.10	17.09	7.82
<i>j85m21s7</i>	18104.96	15032.33	14457.78	20.44	3.97
<i>j90m21s7</i>	20968.67	16028.82	15087.08	30.82	6.24
<i>j95m21s7</i>	24439.44	20539.85	18528.34	18.99	10.86
平均值	6437.95	4911.9	4453.03	44.79	15.70

(3) 与 CPLEX 的比较

为了进一步验证算法的有效性,采用商业优化器 CPLEX 12.4 求解本文的问题模型,以验证 HSGA 的性能。CPLEX 的运行时间上限为 6 小时。实验结果如表 2.15 所

示。对于本文的 18 个测试问题，在最小化 C_{max} 目标下，HSGA 比 CPLEX 平均提升了 49.11%，而在最小化 TWT 目标下，HSGA 比 CPLEX 平均提升了 89.16%，HSGA 的解性能与 CPLEX 的相比有着显著提高。而且，从 $j50m15s5$ 开始 CPLEX 在 6 小时内无法求出可行解，而 HSGA 可以在很短的时间（0.20 秒~7.40 秒）内求解出满意的可行解，表明 HSGA 相当高的计算效率。

表 2.15 HSGA 与 CPLEX 在多台批处理机模型下的比较

测试问题	CPLEX				HSGA				Gap% (C_{max})	Gap% (TWT)
	C_{max}	时间 (h)	TWT	时间 (h)	C_{max}	时间 (s)	TWT	时间 (s)		
$j10m8s3$	692	6	6.28	6	374.00	0.20	0	0.20	-45.95	-100.00
$j15m8s3$	851	6	54.11	6	419.00	0.31	0	0.32	-50.76	-100.00
$j20m11s3$	1170	6	761.75	6	446.20	0.46	0	0.44	-61.86	-100.00
$j25m11s3$	1476	6	1289.79	6	542.00	0.59	0	0.58	-63.28	-100.00
$j30m7s3$	1744	6	4355.03	6	934.40	0.80	578.394	0.77	-46.42	-86.72
$j35m7s3$	1909	6	5290.22	6	1059.20	1.00	1368.008	0.95	-44.52	-74.14
$j40m13s5$	1774	6	4616.65	6	1096.80	1.78	952.662	1.65	-38.17	-79.36
$j45m13s5$	2084	6	7603.51	6	1210.20	2.06	2049.398	1.95	-41.93	-73.05
$j50m15s5$	-	-	-	-	1253.86	2.47	2255.434	2.38	-	-
$j55m15s5$	-	-	-	-	1347.70	2.75	3294.652	2.79	-	-
$j60m16s5$	-	-	-	-	1136.94	2.71	1820.739	2.68	-	-
$j65m16s5$	-	-	-	-	1225.90	3.12	2496.315	3.02	-	-
$j70m20s7$	-	-	-	-	1293.78	4.96	2760.782	4.76	-	-
$j75m20s7$	-	-	-	-	1365.20	5.34	3997.12	5.24	-	-
$j80m21s7$	-	-	-	-	1902.60	5.39	10801.1	5.49	-	-
$j85m21s7$	-	-	-	-	2022.90	5.92	14457.78	5.88	-	-
$j90m21s7$	-	-	-	-	2110.26	6.89	15087.08	6.91	-	-
$j95m21s7$	-	-	-	-	2225.58	7.55	18528.34	7.40	-	-
平均值									-49.11	-89.16

为了验证 HSGA 与最优解的差距，将批处理机的数量缩减至一台，重新设置了 6 个小规模测试问题进行一组新的实验，实验结果如表 2.16 所示。通过缩减问题规模，CPLEX 能够在 6 小时内求得满意的可行解。在最小化 C_{max} 目标下，HSGA 与 CPLEX 的平均 GAP 很小，仅为 0.84%，表明在 HSGA 在该目标下具有与 CPLEX 相当的寻优能力。在最小化 TWT 目标下，HSGA 与 CPLEX 的平均 GAP 较大，为 9.32%，但仍然在 10% 以内，表明 HSGA 在该目标下具备良好的寻优能力。而且，HSGA 求解所有规模的测试问题所需时间均在 1 秒内，与 CPLEX 相比（6 小时），HSGA 是一个高效的算法。

实验结果还呈现一个有趣的现象，在最小化 TWT 目标下 HSGA 与 CPLEX 的 GAP 较在最小化 C_{max} 的大。一个可能的原因是与目标函数的计算方式有关。因为对于最小化 C_{max} ， C_{max} 仅与本次调度最晚完工的零件 j^* 有关，即使把别的零件完工时间提前，只要 j^* 的完工时间不变， C_{max} 就不变。而对于最小化 TWT，TWT 与每个零件的完工时间都有关，因此只要提前任意一个零件的完工时间，都可获得更小的 TWT。所以，与优化 C_{max} 相比，优化 TWT 更容易见效。

综合实验结果表明 HSGA 在保证计算效率的同时，具备较好的寻优能力，是一种高效的算法。

表 2.16 HSGA 与 CPLEX 在单台批处理机模型下的比较

测试问题	CPLEX				HSGA				Gap% (C_{max})	Gap% (TWT)
	C_{max}	时间 (h)	TWT	时间 (h)	C_{max}	时间 (s)	TWT	时间 (s)		
<i>j5m5s3</i>	547	6	240.85	6	513	0.10	241.29	0.09	-6.22	0.18
<i>j8m6s3</i>	718	6	813.97	6	712	0.15	872.34	0.15	-0.84	7.17
<i>j10m7s3</i>	683	6	993.29	6	698	0.17	1087.51	0.18	2.20	9.49
<i>j12m9s3</i>	789	6	1736.59	6	829	0.35	1922.61	0.34	5.07	10.71
<i>j16m10s5</i>	1145	6	2202.12	6	1166	0.53	2568.2	0.47	1.83	16.62
<i>j18m12s5</i>	1269	6	4026.40	6	1307	0.62	4499.56	0.62	2.99	11.75
平均值									0.84	9.32

2.5.4 敏感度分析

在表 2.10 中，分别在两个目标下挑选最优的组合规则，即 SPMS (C_{max}) 和 SRSS (TWT)，作为本小节实验的比较对象。本节分别在不同的到达时间产生模式和交货期设置上对 HSGA 进行敏感度分析。

(1) 到达时间产生模式

本文考虑了四种不同的到达时间产生模式验证 HSGA 的鲁棒性，分别记为 uni50、uni100、exp25 和 exp50。前两个模式分别服从 [0,50] 和 [0,100] 离散均匀分布，后两个模式分别服从均值为 25 和 50 的指数分布。每个实验分别独立运行 5 次。

从表 2.17 可以清楚看出 HSGA 在各模式下均显著优于 SPMS，在最小化 C_{max} 目标下，HSGA 与 SPMS 的 GAP 在模式 uni50、uni100、exp25 和 exp50 下分别是 26.71%、25.57%、24.34% 和 26.10%，这表明在最小化 C_{max} 目标下，HSGA 比 SPMS 对于变化的到达时间产生模式更鲁棒。

表 2.17 不同到达时间产生模式 HSGA 与 SPMS 在最小化 C_{max} 目标下的 GAP

测试问题	uni50	uni100	exp25	exp50
<i>j10m8s3</i>	29.95	47.11	24.11	34.60
<i>j15m8s3</i>	39.92	29.57	36.44	29.71
<i>j20m11s3</i>	42.15	34.21	16.05	43.10
<i>j25m11s3</i>	35.68	33.02	31.13	37.36
<i>j30m7s3</i>	30.18	30.13	27.37	21.87
<i>j35m7s3</i>	11.71	8.36	7.25	23.15
<i>j40m13s5</i>	24.56	21.00	16.70	16.23
<i>j45m13s5</i>	10.83	12.30	27.88	8.11
<i>j50m15s5</i>	45.63	39.26	40.47	28.69
<i>j55m15s5</i>	18.94	29.80	35.61	44.88
<i>j60m16s5</i>	29.19	24.48	37.39	24.98
<i>j65m16s5</i>	22.10	33.70	15.10	29.69
<i>j70m20s7</i>	23.25	29.90	13.98	22.12
<i>j75m20s7</i>	16.97	16.03	15.82	21.38
<i>j80m21s7</i>	19.53	22.14	20.36	21.07
<i>j85m21s7</i>	30.45	13.71	23.10	15.74
<i>j90m21s7</i>	18.80	19.02	29.36	26.75
<i>j95m21s7</i>	31.01	16.61	19.98	20.30
Average	26.71	25.57	24.34	26.10

从表 2.18 可知，在最小化 TWT 目标下，HSGA 与 SPMS 的 GAP 在模式 uni50、uni100、exp25 和 exp50 下分别是 71.06%、55.44%、67.43 和 56.33%，这表明在最小化 TWT 目标下，HSGA 比 SRSS 对于变化的到达时间产生模式更鲁棒。

表 2.18 不同到达时间产生模式 HSGA 与 SRSS 在最小化 TWT 目标下的 GAP

测试问题	uni50	uni100	exp25	exp50
<i>j10m8s3</i>	0.00	0.00	0.00	0.00
<i>j15m8s3</i>	0.00	0.00	0.00	0.00
<i>j20m11s3</i>	0.00	0.00	0.00	0.00
<i>j25m11s3</i>	0.00	0.00	0.00	0.00
<i>j30m7s3</i>	115.97	65.70	74.11	48.83
<i>j35m7s3</i>	23.59	9.42	28.33	34.09
<i>j40m13s5</i>	53.46	75.51	72.82	70.80
<i>j45m13s5</i>	34.15	8.41	35.56	19.59
<i>j50m15s5</i>	223.84	221.74	285.06	234.33
<i>j55m15s5</i>	121.45	100.26	80.45	131.23
<i>j60m16s5</i>	276.78	197.99	222.83	106.02
<i>j65m16s5</i>	134.89	71.58	107.56	107.35
<i>j70m20s7</i>	68.24	64.00	67.93	96.10
<i>j75m20s7</i>	141.15	133.41	146.41	92.12

<i>j80m21s7</i>	12.18	7.44	7.65	8.37
<i>j85m21s7</i>	7.62	7.35	4.02	2.87
<i>j90m21s7</i>	0.28	4.39	13.57	2.03
<i>j95m21s7</i>	65.45	47.64	67.48	64.29
Average	71.06	55.44	67.43	56.33

(2) 交货期设置

由于交货期的设置仅与最小化 TWT 目标相关，因此该部分的实验只在最小化 TWT 目标下进行。交货期的设置是根据公式 (2.23) 且交货期因子 dl 从取值 {1, 2, 3, 4}，分别表示紧张、中等、适中和松弛的交货期设置。

从表 2.19 可知，在不同的交货期设置下，HSGA 与 SRSS 相比，解性能有着显著的提高。而且，一个有趣的趋势是 HSGA 与 SRSS 的 GAP 随着 dl 的增大而增大，即 HSGA 在松弛的交货期设置下能获得更优的解性能。

表 2.19 不同交货期设置 HSGA 与 SRSS 在最小化 TWT 目标下的 GAP

测试问题	$dl=1$	$dl=2$	$dl=3$	$dl=4$
<i>j10m8s3</i>	25.69	0.00	0.00	0.00
<i>j15m8s3</i>	21.95	5924.74	0.00	0.00
<i>j20m11s3</i>	4.53	587.65	0.00	0.00
<i>j25m11s3</i>	32.81	888.77	0.00	0.00
<i>j30m7s3</i>	8.04	10.47	16.87	317.12
<i>j35m7s3</i>	0.90	9.83	14.06	67.04
<i>j40m13s5</i>	13.45	31.53	48.69	262.82
<i>j45m13s5</i>	7.72	16.20	38.83	419.63
<i>j50m15s5</i>	68.72	112.83	263.64	938.17
<i>j55m15s5</i>	35.88	61.90	109.92	355.86
<i>j60m16s5</i>	26.88	66.22	174.19	2252.28
<i>j65m16s5</i>	32.50	60.19	132.64	473.89
<i>j70m20s7</i>	25.57	41.13	52.72	512.34
<i>j75m20s7</i>	41.31	70.28	170.90	488.80
<i>j80m21s7</i>	1.75	1.69	1.02	0.56
<i>j85m21s7</i>	4.31	5.73	4.14	5.15
<i>j90m21s7</i>	7.87	7.54	15.95	24.33
<i>j95m21s7</i>	22.98	37.25	62.99	90.38
Average	21.27	440.78	61.48	344.91

2.5.5 计算效率

采用相同的 18 组测试问题在不同的数据规模上对 HSGA 和 GA_YI 的算法进行比较，在最小化 C_{max} 和 TWT 两个目标下测试了两种算法的 CPU 时间。在解决相同的测试问题时，HSGA 算法较 GA_YI 耗时，这是因为 HSGA 面向机器的编码方式比

GA_YI 面向阶段的编码方式拥有更大的搜索空间。如表 2.13 和表 2.14 所示, 在最小化 C_{max} 和 TWT 目标下, HSGA 的解性能相比 GA_YI 分别有 2.63% 和 15.70% 的提升, 而二者在计算时间上的差距却很小。由此可见, 调度过程中不同的机器环境信息确实对解性能有较大的影响。从图 2.9 和图 2.10 可知, HSGA 求解最大规模的测试问题 $j95m21s7$ 平均不到 8s, 可见 HSGA 能在较短时间内求得近优解。而且, 采用 HSGA 搜索组合规则的方法, 在测试问题规模增长时 CPU 时间呈现近线性增长, 不会出现 CPU 时间因数据规模增大而急剧增长的情况, 这表明 HSGA 具备处理较大规模调度问题的能力。

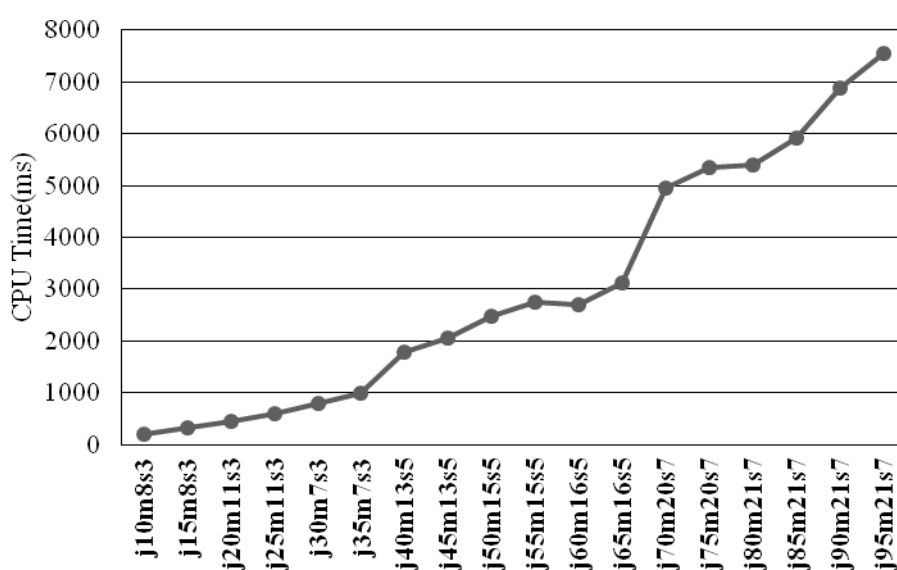


图 2.9 最小化 C_{max} 下 HSGA 的计算时间

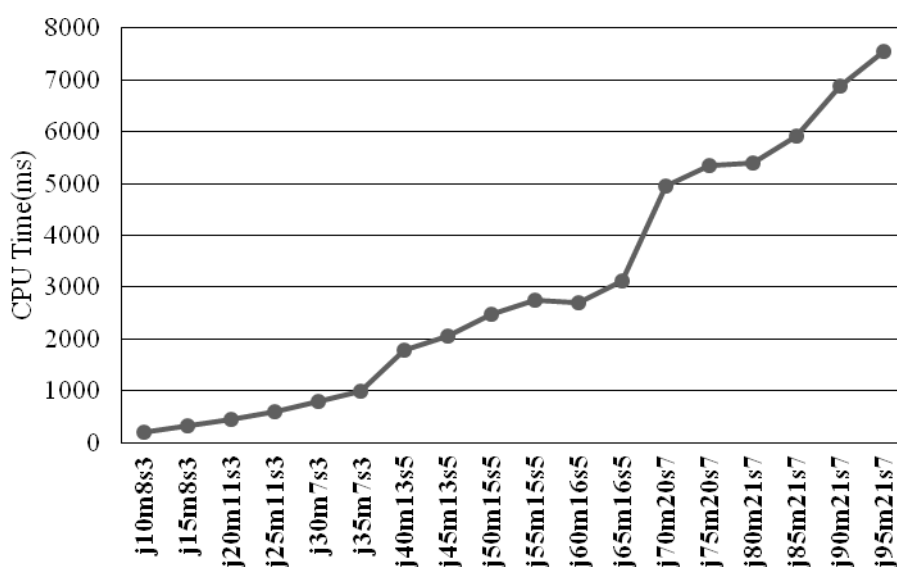


图 2.10 最小化 TWT 下 HSGA 的计算时间

2.6 改进算法的性能分析

2.6.1 实验设计

根据不同的机器设置和零件设置，设计了 22 个不同规模的测试问题，每个规模下随机产生 10 个算例（instance），共有 240 个不同的算例。产生算例的参数设置如表 2.6 所示。每个测试问题按照表 3 参数随机产生 10 个不同的算例。每个算例进行 5 次独立的仿真实验。10 个算例的目标函数值的均值作为此测试问题的性能指标。零件交货期的设置可参照公式（2.23）。新添加的测试问题的机器布局如表 2.20 所示。

表 2.20 新测试问题的机器布局

测试问题	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11
<i>j100m29s9</i>	2	2	4	3	4(B)	4	3	4	3		
<i>j105m29s9</i>	2	2	4	4	4(B)	3	4	3	3		
<i>j110m35s11</i>	4	3	2	4	4	3(B)	4	4	2	2	3
<i>j115m33s11</i>	4	3	4	3	4	4(B)	3	2	2	2	2

B 表示该阶段是由批处理机组成的

2.6.2 数据集

对于 GP 算法，需要指定训练集和测试集。训练集和测试集如表 2.21 所示。

表 2.21 GP 算法的训练集和测试集

数据集	测试问题
训练集	<i>j10m8s3</i> , <i>j20m11s3</i> , <i>j30m7s3</i> , <i>j40m13s5</i> , <i>j50m15s5</i> , <i>j60m16s5</i> , <i>j70m20s7</i> , <i>j80m21s7</i> , <i>j90m21s7</i>
测试集	<i>j15m8s3</i> , <i>j25m11s3</i> , <i>j35m7s3</i> , <i>j45m13s5</i> , <i>j55m15s5</i> , <i>j65m16s5</i> , <i>j75m20s7</i> , <i>j85m21s7</i> , <i>j95m21s7</i> , <i>j100m29s9</i> , <i>j105m29s9</i> , <i>j110m35s11</i> , <i>j115m33s11</i>

2.6.3 性能比较实验

确定 GP 算法参数以及训练集和测试集后，需要验证算法的有效性并分析其效率。本章将从以下四个方面验证算法的有效性：（1）GP 规则与组合规则进行性能比较；（2）与 HSGA 比较；（3）与常见规则组合比较；（4）与 CPLEX 比较。

（1）与组合规则比较

对于每个子问题的启发式规则，分别取其进化过程中（51 代进化）的两个最优 GP 规则，作为性能分析对象，其组合情况表 2.22 所示。

表 2.22 组合 GP 规则

GP 组合规则			记号
排序规则	指派规则	组批规则	
MachineGP1	JobGP1	BatchGP1	GP1
MachineGP1	JobGP1	BatchGP2	GP2
MachineGP1	JobGP2	BatchGP1	GP3
MachineGP1	JobGP2	BatchGP2	GP4
MachineGP2	JobGP1	BatchGP1	GP5
MachineGP2	JobGP1	BatchGP2	GP6
MachineGP2	JobGP2	BatchGP1	GP7
MachineGP2	JobGP2	BatchGP2	GP8

采用公式 (2.21) 定义每个规则的性能表现, 分别在训练集和测试集比较组合 GP 规则与原有组合规则集解性能。如表 2.23 和表 2.24 所示, 在最小化 C_{max} 目标下和最小化 TWT 目标下, 组合 GP 规则比原有组合的解性能具有大幅度的提高。容易发现, 组合 GP 规则在两个目标下, 测试集的解性能平均优于训练集的解性能, 表明通过 GP 学习的启发式规则具有较强的泛化性, 对于未知的测试用例能保持优异的解性能。另外, 一个有趣的现象是, 在最小化 TWT 目标下, 不管是训练集还是测试集, 8 个 GP 组合规则的解性能均优于 HSGA 的解性能, 这表明一方面 GP 规则具有很强的寻优能力, 甚至优于 HSGA; 从另一方面看, HSGA 解性能的好坏与搜索的规则集有着直接的联系, 给 HSGA 提供优质的启发式规则, 将会大大提高 HSGA 的解性能。实验结果表明, 使用 GP 算法能有效地生成优质启发式规则, 同时 GP 规则在测试集的优异表现表明了 GP 规则具备可复用的价值。

表 2.23 最小化 C_{max} 目标下 GP 规则与常用组合规则的解性能

组合规则方案	训练集	测试集
GP1	10.01%	8.65%
GP2	10.09%	8.75%
GP3	10.01%	8.65%
GP4	10.09%	8.75%
GP5	9.88%	8.77%
GP6	10.09%	9.02%
GP7	9.88%	8.77%
GP8	10.09%	9.02%
SPMS	28.77%	24.96%
WSMS	30.06%	25.83%
MME	17.50%	18.22%
SMS	28.40%	25.26%
MMF	17.67%	18.60%

表 2.24 最小化 TWT 目标下 GP 规则与常用组合规则的解性能

组合规则方案	训练集	测试集
GP1	-5.46%	-14.91%
GP2	-6.38%	-15.07%
GP3	-5.36%	-14.32%
GP4	-4.99%	-14.74%
GP5	-4.83%	-15.19%
GP6	-5.73%	-14.98%
GP7	-5.00%	-14.10%
GP8	-4.83%	-14.45%
SRSS	29.27%	28.74%
SSS	29.91%	29.38%
SPSS	44.25%	41.21%
WSSS	49.84%	49.29%
SRSE	34.74%	39.25%

(2) 与 HSGA 比较

HSGA 解性能的好坏依赖于搜索的规则集。因此，在 HSGA 原有的规则集中，加入学习的 GP 规则改进 HSGA，把改进后的 HSGA 记为 GP+HSGA，通过与未改进的 HSGA 比较，验证改进后的效果。在最小化 C_{max} 目标下，GP+HSGA 与 HSGA 的平均 GAP 为-2.03%，GP+HSGA 略优于 HSGA，与组合 GP 规则(9.41%)相比，GP+HSGA (-2.03%)具有大幅度提高。在最小化 TWT 目标下，GP+HSGA 与 HSGA 的平均 GAP 为-36.25%，GP+HSGA 与 HSGA 相比，解性能具有大幅度提高，与组合 GP 规则 (-10.02%) 相比，GP+HSGA (-36.25%) 同样具有大幅度提升。实验结果表明，不管在最小化 C_{max} 目标下还是最小化 TWT 目标下，GP+HSGA 能比 HSGA 提供更好的解性能，而且对于 HSGA 算法来说，提供优质的规则集能极大的改善 HSGA 的解性能。关于最小化 TWT 目标下的优化效果比最小化 C_{max} 明显的原因已在 2.5.3 节分析过了。因此，非常有必要引入 GP 算法改善 HSGA 的解性能。

表 2.25 最小化 C_{max} 目标下 GP+HSGA 与 HSGA 的 GAP

测试问题	HSGA	GP+HSGA	GAP%
<i>j10m8s3</i>	377.32	371.14	-1.64
<i>j15m8s3</i>	431.16	424.3	-1.59
<i>j20m11s3</i>	430.64	421.12	-2.21
<i>j25m11s3</i>	523.98	509.22	-2.82
<i>j30m7s3</i>	933.54	926.62	-0.74
<i>j35m7s3</i>	1096.22	1094.52	-0.16
<i>j40m13s5</i>	1075.18	1060.14	-1.40

面向军工装备制造业的智能优化排产软件

<i>j45m13s5</i>	1170.6	1157.06	-1.16
<i>j50m15s5</i>	1253.86	1258.02	0.33
<i>j55m15s5</i>	1347.7	1367.08	1.44
<i>j60m16s5</i>	1136.94	1107.94	-2.55
<i>j65m16s5</i>	1225.9	1188.08	-3.09
<i>j70m20s7</i>	1293.78	1242.22	-3.99
<i>j75m20s7</i>	1365.2	1331.64	-2.46
<i>j80m21s7</i>	1902.6	1896.6	-0.32
<i>j85m21s7</i>	2022.9	2011.6	-0.56
<i>j90m21s7</i>	2110.26	2072.5	-1.79
<i>j95m21s7</i>	2225.58	2177.7	-2.15
<i>j100m29s9</i>	1861.67	1782.67	-4.24
<i>j105m29s9</i>	1917.00	1836.67	-4.19
<i>j110m35s11</i>	1753.00	1667.67	-4.87
<i>j115m33s11</i>	1780.67	1702	-4.42
平均值	1328.90	1300	-2.03

表 2.26 最小化 TWT 目标下 GP+HSGA 与 HSGA 的 GAP

测试问题	HSGA	GP+HSGA	GAP%
<i>j10m8s3</i>	0	0	-
<i>j15m8s3</i>	0	0	-
<i>j20m11s3</i>	0	0	-
<i>j25m11s3</i>	0	0	-
<i>j30m7s3</i>	636.21	541.66	-14.86
<i>j35m7s3</i>	1328.80	903.23	-32.03
<i>j40m13s5</i>	847.16	674.47	-20.39
<i>j45m13s5</i>	1842.96	1263.86	-31.42
<i>j50m15s5</i>	2255.43	1578.25	-30.02
<i>j55m15s5</i>	3294.65	2187.01	-33.62
<i>j60m16s5</i>	1820.74	1222.33	-32.87
<i>j65m16s5</i>	2496.32	1729.71	-30.71
<i>j70m20s7</i>	2760.78	1439.74	-47.85
<i>j75m20s7</i>	3997.12	1979.17	-50.49
<i>j80m21s7</i>	10801.10	8508.60	-21.22
<i>j85m21s7</i>	14457.78	11234.00	-22.30
<i>j90m21s7</i>	15087.08	9920.96	-34.24
<i>j95m21s7</i>	18528.34	12598.08	-32.01
<i>j100m29s9</i>	12370.32	5863.78	-52.60
<i>j105m29s9</i>	14416.93	6927.42	-51.95
<i>j110m35s11</i>	12775.72	5617.40	-56.03
<i>j115m33s11</i>	14191.93	5964.51	-57.97
平均值	6086.79	3643.37	-36.25

(3) 与常见规则组合比较

本节将改进算法与常用的48种组合规则进行比较，从平均零件流动时间、最大完工时间、平均延迟、延迟零件率和平均跨单元率，五个指标上对算法的有效性进行验证。

(i) 平均零件流动时间

从图2.11中可以看出，算法在所有规模下，其平均零件流动时间值均优于48种规则组合。

从图 2.12 中分析，在其中 35 种规则组合中，不存在某种组合在所有规模下均优于其他组合的情况，并且它们在同一数据规模下的平均零件流动时间没有明显差异。在同种调度规则中，计算不同数据规模下平均零件流动时间的均值，比较算法与 35 种规则组合在该性能指标上的差异发现，均值最大的组合规则是 SPT 和 LPT*TOT 组合，较该算法的均值高 7.11%，均值最小的组合规则是 EAT 和 LPT*TOT 组合，较改进算法的均值高 4.46%。

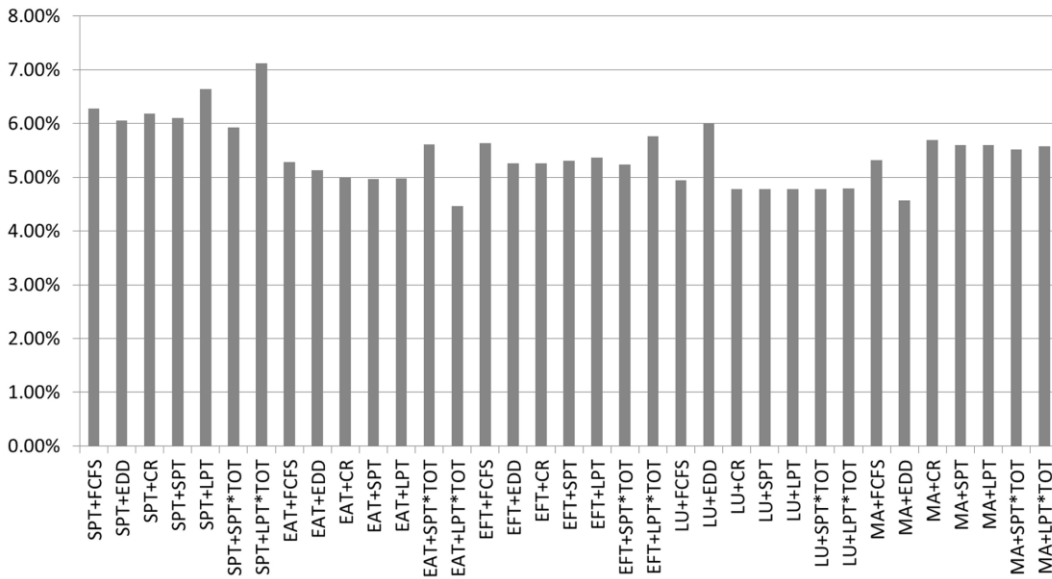


图 2.11 改进算法与组合启发式规则在平均零件流动时间上的 GAP 值

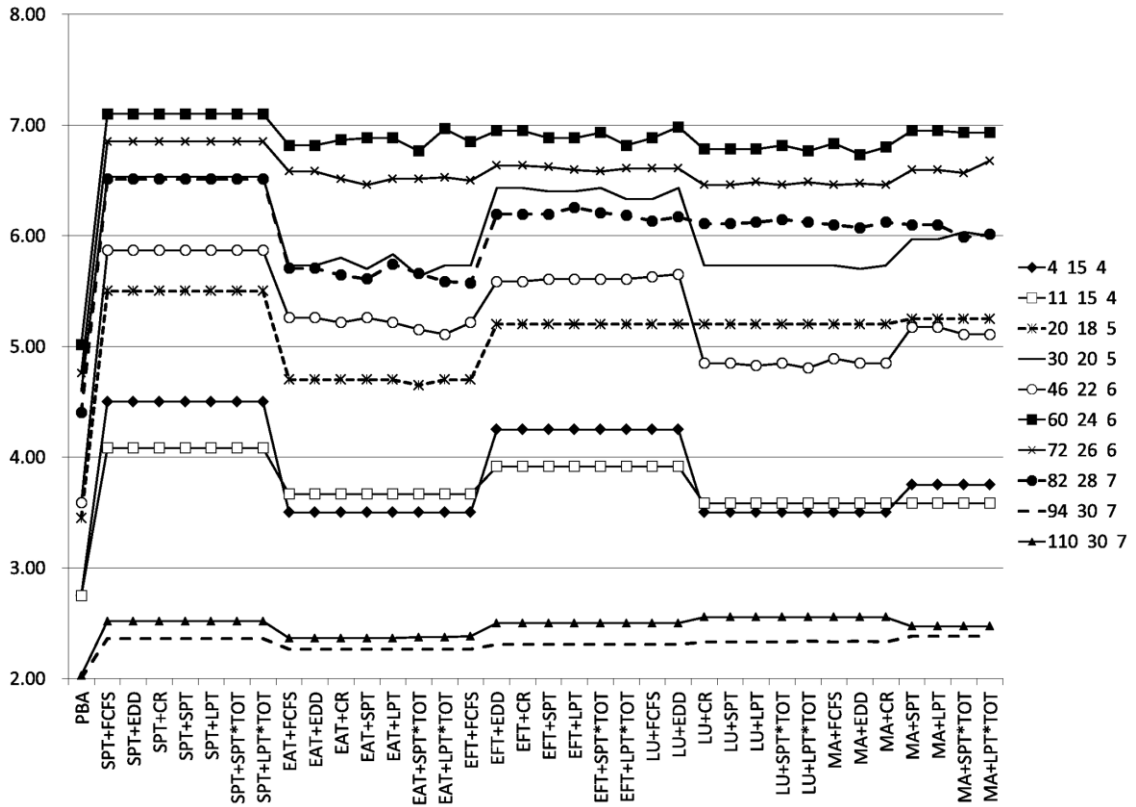


图 2.12 改进算法与组合启发式规则在平均零件流动时间上的对比

(ii) 最大完工时间

不同数据规模下，使用不同调度方法所得到的最大完工时间值如图2.13所示。从实验数据可见，在该性能指标上超启发式算法是最优的，较35种规则组合中性能最好的EAT和LPT*TOT组合低4.39%，较性能最差的SPT和LPT*TOT组合低8.40%。同时，在35种规则组合中不存在某一种组合在所有的数据规模下均优于其他组合的情况。

考察路径选择规则SPT、EAT、EFT分别与7种零件选择规则相组合的情况发现（如图2.14）：SPT+LPT•TOT，EAT+LPT•TOT和EFT+LPT•TOT的性能均劣于它们与其他零件选择规则的组合。SPT是最为广泛使用的零件选择规则之一，而本文的实验结果表明，在大中小10种数据规模下，LPT规则的最大完工时间均优于SPT规则。

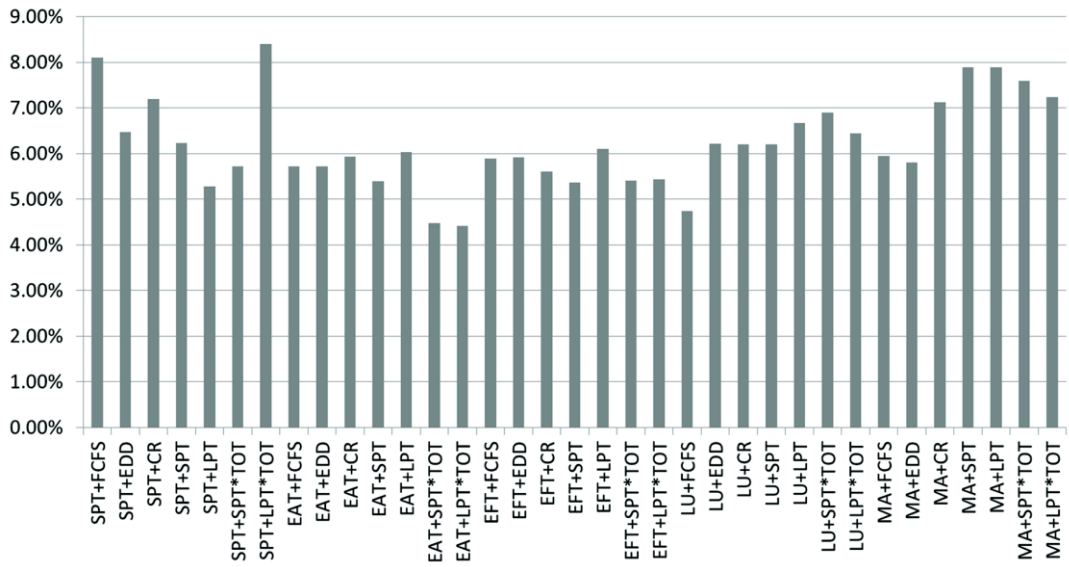


图 2.13 改进算法与组合启发式规则在最大完工时间上的 GAP 值

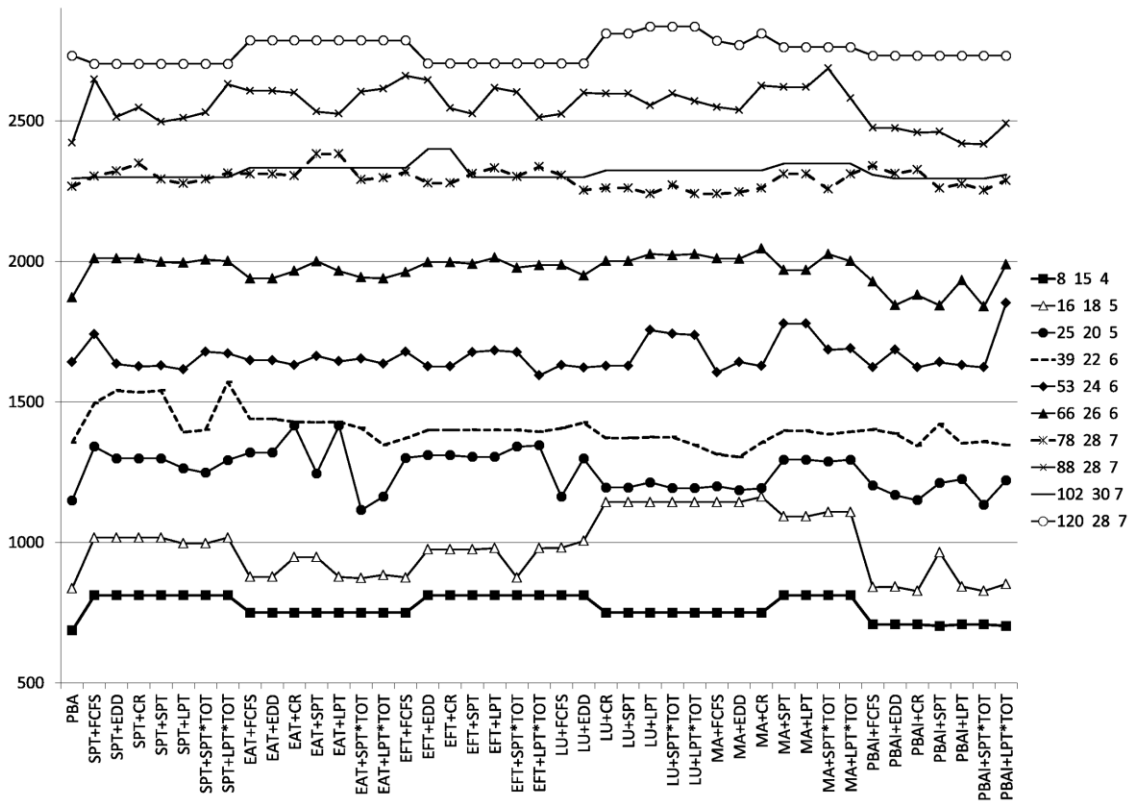


图 2.14 改进算法与组合启发式规则在最大完工时间上的对比

(iii) 平均延迟

从图2.15中可以看出，改进算法在所有数据规模的实验数据下，所获得的平均延迟都是最小的，较35种规则组合中性能最好的EAT和LPT*TOT组合低78.17%，较性能最差的SPT和LPT*TOT组合低78.17%。在35个规则组合中，不存在某种组合在所

有的数据规模下，平均延迟都小于其他组合的情况。

从图2.16中可以观察到，在小规模数据（4,15,4）和（11,15,4）下，路径选择规则 SPT和EFT的平均延迟要大于其他三个规则，但随着数据规模的增大逐渐减小。

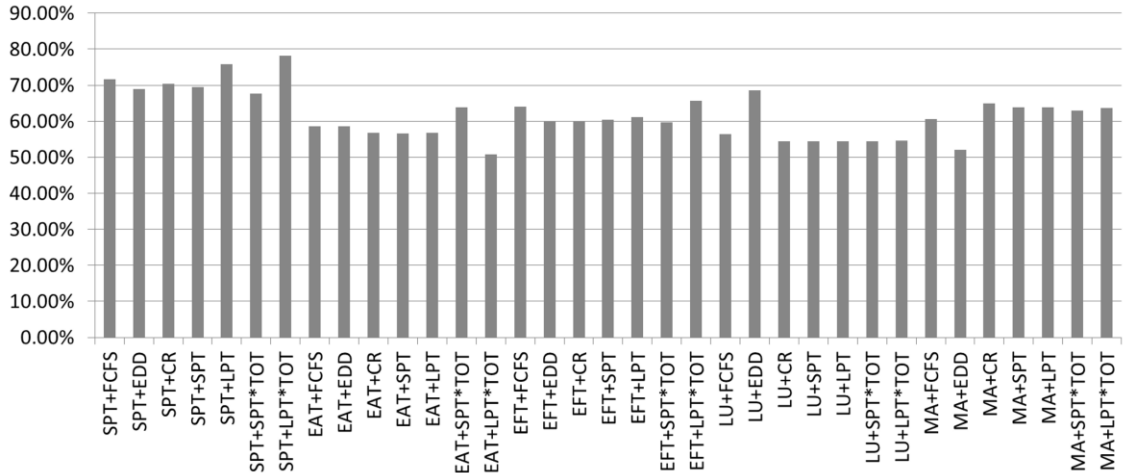


图 2.15 改进算法与组合启发式规则在平均延迟时间上的 GAP 值

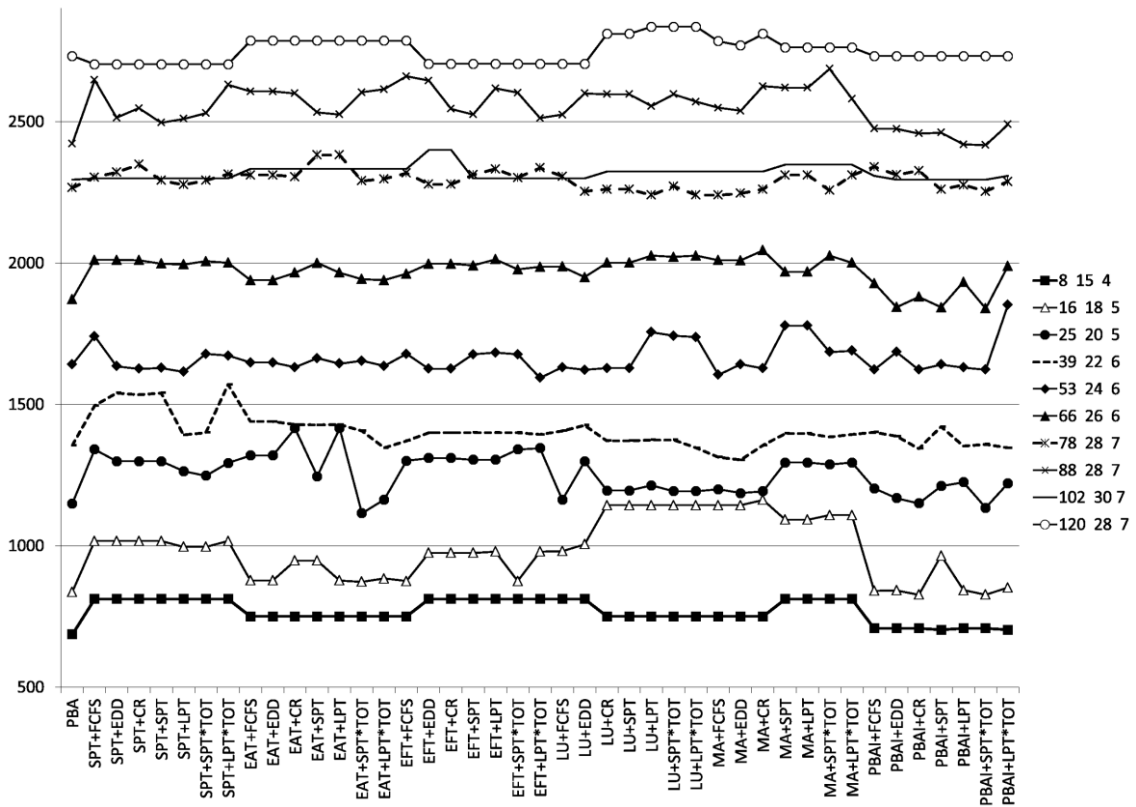


图 2.16 改进算法与组合启发式规则在平均延迟时间上的对比

(iv) 延迟零件率

不同数据规模下的延迟零件率如图2.17所示。从实验数据可见，在该性能指标上改进算法是最优的，较35种规则组合中性能最好的EAT和SPT*TOT组合低25.37%，较性能最差的SPT和EDD组合低61.79%。同时，在35个规则组合中，不存在某种组合在所有的数据规模下均优于其他组合的情况。

观察图2.18可以看出，在5种路径选择规则中，在所有的数据规模下，SPT的延迟零件率均大于其他4种路径选择规则，并且这种劣势随着数据规模的增大逐渐显著。

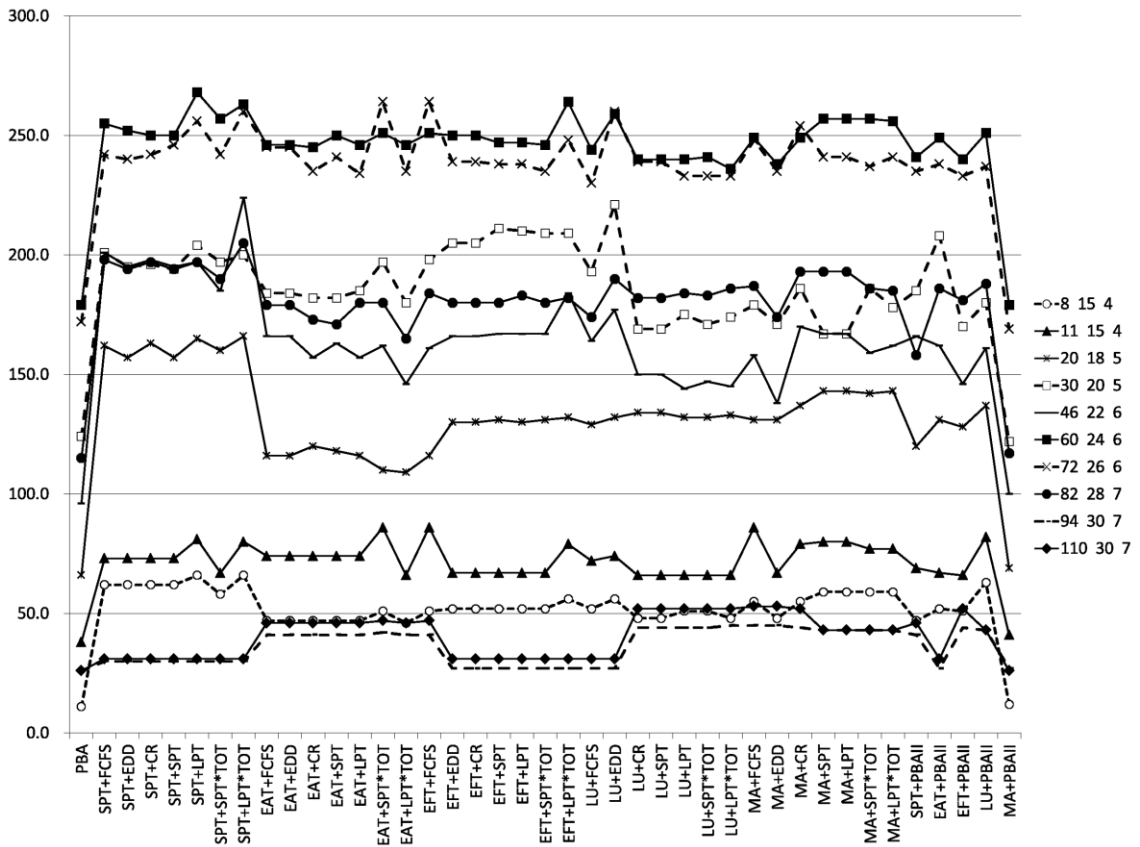


图 2.17 改进算法与组合启发式规则在延迟零件率上的对比

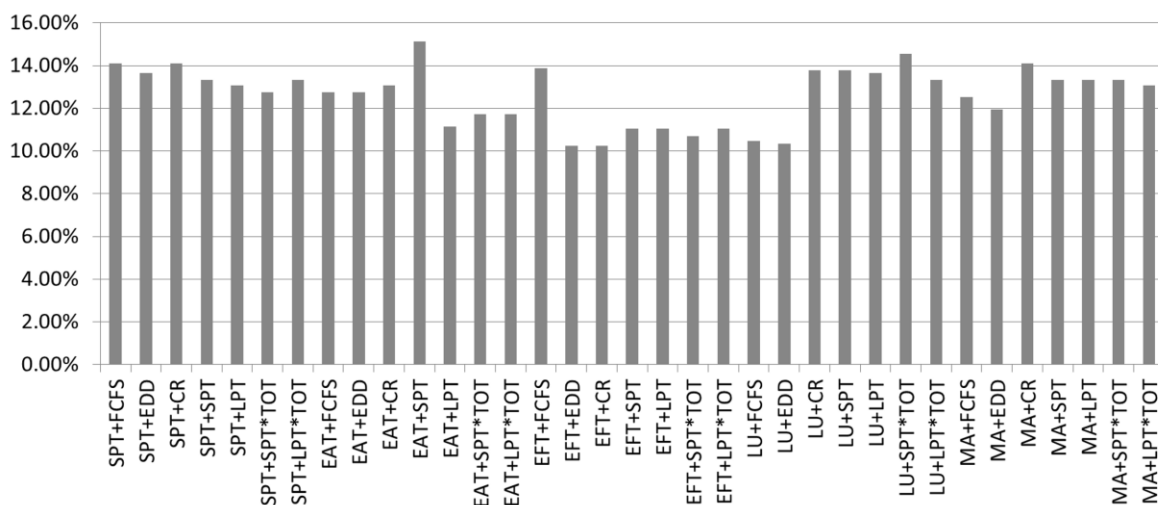


图 2.18 改进算法与组合启发式规则在延迟零件率上的 GAP 值

(v) 平均跨单元率

从图2.19中可以看出，改进算法在所有10种规模的实验数据下，所获得的平均跨单元率都是最优的，较35种规则组合中性能最好的EAT和FCFS组合低42.76%，较性能最差的SPT和FCFS组合低67.30%。。同时，在35种规则组合中，不存在某种组合在所有的数据规模下均优于其他组合的情况。

观察图2.20可以看出，对于5种路径选择规则，在同一数据规模下，SPT与全部7种零件选择规则相组合，所得到的平均跨单元率都是相同的。

对于图2.20，考察路径选择规则EAT与7种零件选择规则的组合发现：虽然在同一数据规模下得到的平均跨单元率不完全相同，但差距很小。当考察路径选择规则EFT、LU和MA时也得到同样的结论。

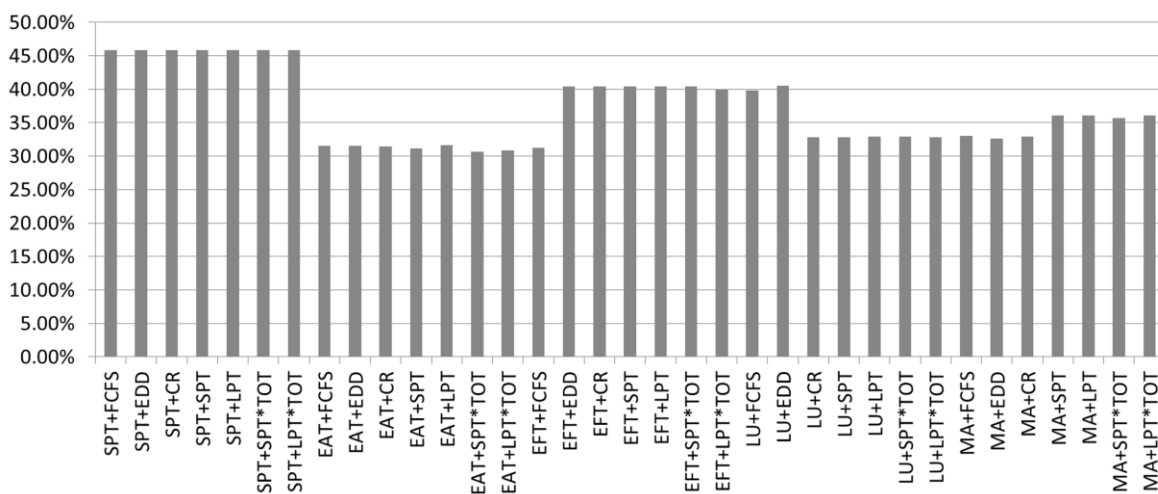


图 2.19 改进算法与组合启发式规则在平均跨单元率上的 GAP 值

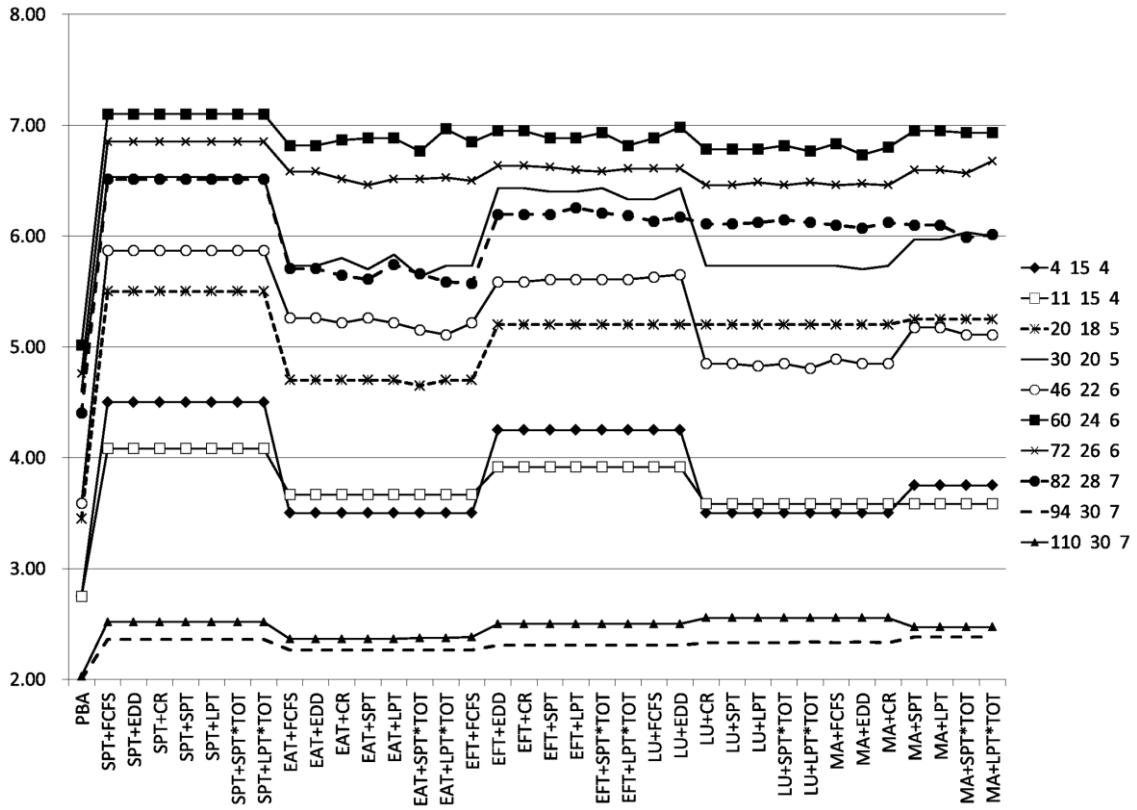


图 2.20 改进算法与组合启发式规则在平均跨单元率上的对比

(4) 与 CPLEX 比较

GP+HSGA 与 CPLEX 的实验结果如表 2.27 和表 2.28 所示。CPLEX 的运行时间设置为 6h。在多台批处理机下，GP+HSGA 与 CPLEX 的 GAP 分别是-50.03% (C_{max}) 和-92.41% (TWT)。多台批处理机的测试问题对于 CPLEX 来说过于复杂，难以在限定时间内求得满意的调度解，从 $j50m15s5$ 开始的较大规模测试问题甚至无法给出可行解。

对于只含单台批处理机的测试问题，在最小化 C_{max} 目标下，GP+HSGA 与 CPLEX 的平均 GAP 很小，仅为 0.44%，表明在 GP+HSGA 在该目标下具有与 CPLEX 相当的寻优能力。在最小化 TWT 目标下，GP+HSGA 与 CPLEX 的平均 GAP 较大，为 8.91%，但仍然在 10% 以内，表明 GP+HSGA 在该目标下具备良好的寻优能力。

表 2.27 多台批处理机下的 GP+HSGA 与 CPLEX 的比较

测试问题	CPLEX			GP+HSGA			Gap for C_{max}	Gap for TWT	
	C_{max}	时间 (h)	TWT	C_{max}	时间 (s)	TWT			
$j10m8s3$	692	6	6.28	371.14	0.20	0.00	0.20	-46.37	-100.00

面向军工装备制造业的智能优化排产软件

j15m8s3	851	6	54.11	6	424.3	0.31	0.00	0.32	-50.14	-100.00
j20m11s3	1170	6	761.75	6	421.12	0.46	0.00	0.44	-64.01	-100.00
j25m11s3	1476	6	1289.79	6	509.22	0.59	0.00	0.58	-65.50	-100.00
j30m7s3	1744	6	4355.03	6	926.62	0.80	541.66	0.77	-46.87	-87.56
j35m7s3	1909	6	5290.22	6	1094.52	1.00	903.23	0.95	-42.67	-82.93
j40m13s5	1774	6	4616.65	6	1060.14	1.78	674.47	1.65	-40.24	-85.39
j45m13s5	2084	6	7603.51	6	1157.06	2.06	1263.86	1.95	-44.48	-83.38
j50m15s5	-	-	-	-	1258.02	2.47	1578.25	2.38	-	-
j55m15s5	-	-	-	-	1367.08	2.75	2187.01	2.79	-	-
j60m16s5	-	-	-	-	1107.94	2.71	1222.33	2.68	-	-
j65m16s5	-	-	-	-	1188.08	3.12	1729.71	3.02	-	-
j70m20s7	-	-	-	-	1242.22	4.96	1439.74	4.76	-	-
j75m20s7	-	-	-	-	1331.64	5.34	1979.17	5.24	-	-
j80m21s7	-	-	-	-	1896.6	5.39	8508.60	5.49	-	-
j85m21s7	-	-	-	-	2011.6	5.92	11234	5.88	-	-
j90m21s7	-	-	-	-	2072.5	6.89	9920.96	6.91	-	-
j95m21s7	-	-	-	-	2177.7	7.55	12598.08	7.40	-	-
j100m29s	-	-	-	-	1782.67	10.3	5863.78	13.1	-	-
9	-	-	-	-	-	85	-	8	-	-
j105m29s	-	-	-	-	1836.67	11.0	6927.42	14.2	-	-
9	-	-	-	-	-	11	-	3	-	-
j110m35s	-	-	-	-	1667.67	12.1	5617.40	14.9	-	-
11	-	-	-	-	-	59	-	8	-	-
j115m33s	-	-	-	-	1702.00	12.2	5964.51	16.0	-	-
11	-	-	-	-	-	69	-	1	-	-
平均值									-50.03	-92.41

表 2.28 单台批处理机下的 GP+HSGA 与 CPLEX 的比较

测试问题	CPLEX			GP+HSGA			Gap for C_{max}	Gap for TWT		
	C_{max}	CPU 时间 (h)	TWT	CPU 时间 (h)	C_{max}	CPU 时间 (s)			TWT	
j5m5s3	547	6	240.85	6	513	0.10	241.29	0.09	-6.22	0.18
j8m6s3	718	6	813.97	6	712	0.15	862.32	0.15	-0.84	5.94
j10m7s3	683	6	993.29	6	696	0.17	1076.73	0.18	1.90	8.40
j12m9s3	789	6	1736.59	6	825	0.35	1904.21	0.34	4.56	9.65
j16m10s5	1145	6	2202.12	6	1156	0.53	2589.59	0.47	0.96	17.60
j18m12s5	1269	6	4026.40	6	1298	0.62	4497.39	0.62	2.29	11.70
平均值									0.44	8.91

2.7 真实数据验证

通过实验验证了算法性能后，我们用实际生产数据对算法进行了测试。数据来源于兵器一机集团三代主战坦克综合传动装置的生产过程，测试结果表明，在九百余台设备、七千余种零件、六万余道工序，零件平均跨车间 4 次、最多跨车间 11 次的条件下，3~15 分钟得到计算结果。与应用之前相比，一年来的任务完成量增加 18%，在制品库存下降 28%，设备利用率提高 20%，生产周期缩短 35%，全员劳动生产率提高 20%，为企业节约成本总计 1000 余万元，新增产值总计 7000 余万元（应用证明见附件）。

由于涉密问题，本节省略对真实数据的具体分析。

2.8 本章小结

本作品的核心和难点在于如何设计跨车间调度算法。我们基于超启发式算法架构设计了HSGA算法，大幅提高了计算效率并获得了较好的优化能力。更进一步的，针对超启发式算法需要事先指定规则集的限制性，将GP引入算法用于生成规则集。数据分析验证了改进算法的效果。应用情况表明我们的算法在计算效率上突破了阻碍跨车间调度相关研究投入实际应用的瓶颈，获得了良好的应用效果。

第3章 基于能力平衡的作业计划优化算法

3.1 引言

跨车间协作生产的存在主要给制定作业计划和调度带来各车间生产能力不平衡，使得各车间难以制定一致的作业计划。

在大型生产系统中，由于各车间生产特点的不同，导致各车间存在生产能力差异。调研结果显示，在兵器一机集团四分厂的十余个车间里，413车间平均加工时间只有2.85分钟/工序，而409车间平均加工时间为146.35分钟/工序。在车间能力不平衡的背景下，跨车间转移给作业计划带来新问题。

假设某跨车间转移零件依次流经413车间和409车间，如果按照413车间的能力下达各车间作业计划，将会加剧409车间的压力，导致其无法完成生产任务；如果按照409车间的能力下达计划，则会造成413车间设备的闲置浪费。因此，由于各车间生产能力的差异，在跨车间转移频繁发生的情况下按照单一原则下达作业计划无法满足不同车间的需求，应该深入分析跨车间转移对不同能力的车间产生的影响，在不超出一定库存费用的约束下，根据各车间的生产能力制定不同的计划，以达到总体生产效率、设备利用率、在制品库存等多方面指标的综合优化。

我们通过迭代优化的方法设计了基于能力平衡的作业计划优化方法，即使是同一种零件，流经各车间时，也会根据不同的生产能力制定不同的生产计划数，以保障物流平衡。同时结合跨车间调度算法，通过在参考数和保障数（即生产计划的模糊上下限）之间迭代，实现计划与调度的一体优化。

3.2 算法设计

首先对算法中涉及到的一些定义进行简要介绍：（1）保障数：作业计划中必须要完成的最低指标数；（2）参考数：作业计划中期待完成的指标数；（3）流入数：零件在进入某车间时的数量。其中，不同车型的同一型号的零件当作不同零件处理；（4）流出数：零件在离开某车间时的数量。其中，不同车型的同一型号的零件当作不同零件处理；（5）窄口：对于工序，窄口是指生产过程中最耗费时间的部分或者明显落后于平均生产节拍的工序；对于设备，窄口是提供的能力无法满足所需

生产力要求的设备。

本作品算法的主要流程描述如下：

定义如下表3.1所示的符号变量：

表 3.1 符号变量

变量名	定义
$i = 1, \dots, N$	零件索引, N 为零件数
$j = 1, \dots, O_i$	工序索引, O_i 为每个零件的工序数
$m = 1, \dots, M$	机器索引, M 为机器数
$w = 1, \dots, W$	车间索引, W 为车间数
O_{ij}	零件 i 的工序 j
P_{ijm}	零件 i 的工序 j 在机器 m 上的处理时间, 以分钟为单位
C_i	作业 i 的完工时间
C_{\max}	$\max C_i$
MTA_m	机器 m 在作业计划中的总的可用时间
G_m	在同一类型机器上加工的工序集合
$BRatio_{ij}$	零件 i 工序 j 的窄口系数
B_i	零件 i 的窄口工序集合
Bm	零件窄口数上限, 为正整数
$round$	迭代寻优的当前轮次
$roundm$	迭代寻优的最大轮次

第 1 步：确定窄口：

每个零件的每道工序 O_j 的加工都对应着一台机器, 在某台机器 m 上加工的所有工序组成一个工序集合 G_m ；假设工厂的工作时间是每天 h 小时, 每周工作 d 天, 生产月以 w 周为单位, 则机器 m 每个生产月的可用时间计算如下：

$$MTA_m = h * d * w * 60(\text{分钟}) \quad (3.1)$$

确定窄口的方法如下：

1) 对每道工序 O_{ij} 归入对应的工序集合 G_m ;

2) 计算每个 G_m 中工序的总加工时间 $Total_Time_m$

$$Total_Time_m = \sum_{i,j} P_{ijm}, O_{ij} \in G_m \quad (3.2)$$

3) 计算每台机器 m 的总可用时间 MTA_m ;

4) 对于每个 G_m , 如果 $Total_Time_m > MTA_m$, 则把 G_m 中的所有工序标注上候选窄口的标记 Tag , 并且计算 $BRatio_{ij}$:

$$BRatio_{ij} = Total_Time_m / MTA_m \quad (3.3)$$

5) 在每个零件 i 的工艺路线上, 选取具有最大 $BRatio_{ij}$ 的至多 Bm 个车间为窄口车间, 同时在窄口车间中选取具有最大 $BRatio_{ij}$ 的工序 O_{ij} 为窄口工序, 并加入该零件的窄口集合 $B_i, B_i = B_i \cup \{O_{ij}\}$; 其中, Bm 的取值为一个不大于总车间数的正整数;

如图3.1所示, 零件的工艺路线跨了3个车间, 选出每个车间中窄口系数 $BRatio_{ij}$ 最大的工序为窄口工序。

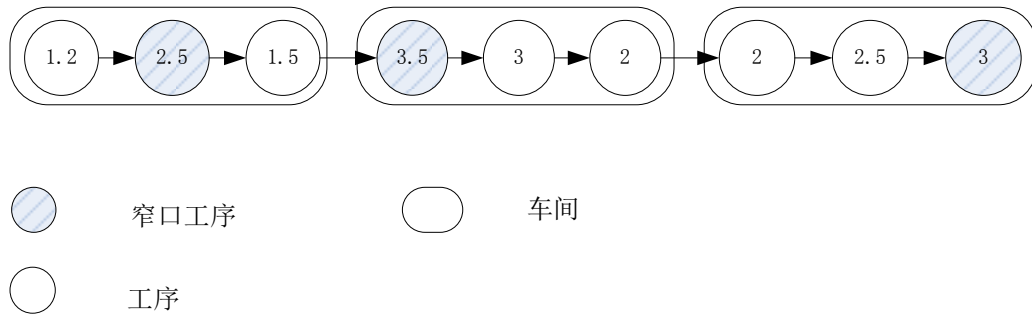


图 3.1 窄口工序的分布示意图

第2步: 如果当前轮次不超过最大迭代轮次, 即 $round < roundm$, 则转入步骤4; 否则转步骤7; 其中, $round$ 为当前迭代寻优轮次, 初始值为1, $roundm$ 为生产管理人员设置的最大迭代轮次, 为一个正整数;

第3步: 采用基于分派规则的调度方法进行预调度, 步骤为:

1) 置可调度集合 S 为空, $S = \emptyset$;

2) 当前时刻, 对于所有未调度的工序 O_{ij} , 如果 O_{ij} 是可调度工序, 则 $S = \{O_{ij}\} \cup S$;

可调度工序是指未调度工序集合中符合以下条件之一的工序: (1) 同一个零件中前一道工序状态为已调度的工序; (2) 子部件或零件的工序状态全部为已调度, 且该工序为本零件的第一道工序; (3) 零件的第一道工序, 且该零件代表零件而非部件;

3) 选择工序: 在可调度集合 S 中选取具有最短加工时间的待调度工序 O_{ij}' , 如果存在多个可选择的工序, 则选择等待时间最长的, 如果还存在多个可选择的工序, 则随机选择一个;

4) 选择机器: 选取最早可用的机器 m' , 如果存在多台可选择的机器, 则选择利用率最低的机器, 如果还存在多台可选择的机器, 则随机选择一台;

5) 在机器 m' 上调度工序 O_{ij}' , 把工序标记为已调度工序, 并且 $S = S - \{O_{ij}'\}$;

6) 如果所有的工序已调度完, 则预调度结束; 否则转步骤 2);

第 4 步: 根据预调度结果调整投产数量, 步骤为;

1) 超期零件集合 $OD = \emptyset$;

2) 对于所有零件 $part_i$, 如果其最后一道工序的结束时间晚于作业计划的交货期 Due_Date , 则把 $part_i$ 加入超期工序集合 OD 中, $OD = OD \cup \{part_i\}$;

3) 对于 OD 中的每个零件, 按照以下策略, 调整工序的投产数量:

(1) 如果该零件没有窄口, 则每次开始调整数量的工序即起始减工序 SO_{ij} 为零件的第一道工序, 从 SO_{ij} 开始约减投产数量, 包括该零件的装配路径上涉及的所有父零件, 从而确定约减工序路径; 否则执行步骤 2);

(2) 如果零件存在窄口, 按照可开始加工时间越晚的工序在前的原则, 对窄口工序集 B_i 进行排序, 并通过 $round \bmod |B_i|$ 获得 SO_{ij} 索引, B_i 中顺序与 SO_{ij} 索引对应的工序即为起始减工序 SO_{ij} , 从 SO_{ij} 开始约减投产数量, 包括该零件的装配路径上涉及的所有父零件, 从而确定约减工序路径; 例如 $B_i = \{O_{i1}, O_{i2}, O_{i3}\}$, 其中 O_{i1} 、 O_{i2} 和 O_{i3} 是一个零件工艺路线上的不同工序, 且 O_{i1} 晚于 O_{i2} , O_{i2} 晚于 O_{i3} , $|B_i|=3$, 那么本次的 SO_{ij} 索引是 $round \bmod |B_i|$; 假如 $round = 1$, 则 $1 \bmod 3 = 1$, 则 $SO_{ij} = O_{i1}$; $round=5$, $5 \bmod 3 = 2$, 则 $SO_{ij} = O_{i2}$, 其他情况依此类推。

(3) 对于通过前两个步骤确定的约减路径上的每一道工序, 按照折半约减和等距约减结合的方法获得调整后的工序数量:

假设当前工序的数量是 $amount$, 则折半约减后的数量按公式 (3.4) 计算:

$$amount1 = \frac{amount + ensure}{2} \quad (3.4)$$

等距约减后的数量按公式 (3.5) 计算:

$$amount2 = \max(ensure, amount - (guide - ensure) / roundm) \quad (3.5)$$

由公式 (3.6) 得出本次调整后的工序数量:

$$amount_new = \min(amount1, amount2) \quad (3.6)$$

其中，*ensure*为作业计划中每个车型的保障数，*guide*为作业计划中每个车型的参考数；

折半约减和等距约减两者结合的方法可以达到快速收敛的效果。

第5步：令 $round = round + 1$ ，转到第2步；

第6步：输出调度结果和对应的投产计划。

该算法取得了以下3方面的效果：（1）有效的解决生产车间能力不平衡问题；（2）有效的解决生产任务跨车间的转移问题；（3）作业计划与调度一体化集成。

3.3 数据分析

实验数据采用某车型A的模拟数据，假设车型A含有2641个零件，21306道工序。

通过对基于实例的数据进行仿真实验统计分析，并对每个零件的窄口工序集合进行统计，得到以下两个表格：

表 3.2 15 台份的车型 A 的窄口分布信息

窄口数	出现次数	所占比例
0	613	23.21%
1	1964	74.37%
2	54	2.04%
3	10	0.38%

表 3.3 20 台份的车型 A 的窄口分布信息

窄口数	出现次数	所占比例
0	611	23.14%
1	1966	74.44%
2	54	2.04%
3	10	0.38%

从表3.2和表3.3可以看出，车型A的窄口数大于1的零件数量只占不到2.5%，绝大部分零件窄口数为1或者没有窄口。因此 *Bm* 的取值为3，能够涵盖所有情况。

在确定窄口之后，按照3.2节中的步骤3-6，对作业计划进行迭代寻优，在每次迭代的过程中都执行一次预调度过程和一次调整投产数量的过程。最大迭代轮次 *roundm* 的取值直接影响了算法的效率。*roundm* 取值太大会极大的增加寻优的时间，太小又会使调度的结果不够优化。因此需要对迭代过程进行仿真，寻找一个有效

$roundm$ 的取值，在不影响总体调度的有效性的前提下，有效的提高算法的运行效率。设单道工序最大数量是 SL ，对于超期零件仅采用折半减的策略，排产台份数为 TS ，理论上最大的迭代次数是：

$$roundm = \lceil \log_2(SL * TS) \rceil \quad (3.7)$$

可以将此理论最大迭代次数作为 $roundm$ 的值。例如，车型A中单道工序最大数量是350，对于超期零件仅采用折半减的策略，那么对于排产15台份的车型A，理论上最大的迭代次数是：

$$roundm = \lceil \log_2(350 * 15) \rceil = 13 \quad (3.8)$$

因此可以将 $roundm$ 的值设为 13。

本算法对 $roundm$ 的设置进行了进一步的优化，首先将理论最大迭代次数作为 $roundm$ 的值，然后按照发明内容的步骤进行仿真，在仿真过程中统计迭代的实际收敛次数，并以此次数作为 $roundm$ 的值。

例如，本算法分别对5台份、30台份的车型A进行仿真，5台份设置 $roundm=20$ ，30台份的设置 $roundm=30$ ，仿真结果如图3.2、图3.3所示。可见，实际值收敛在12次左右，因此取 $roundm=12$ 。

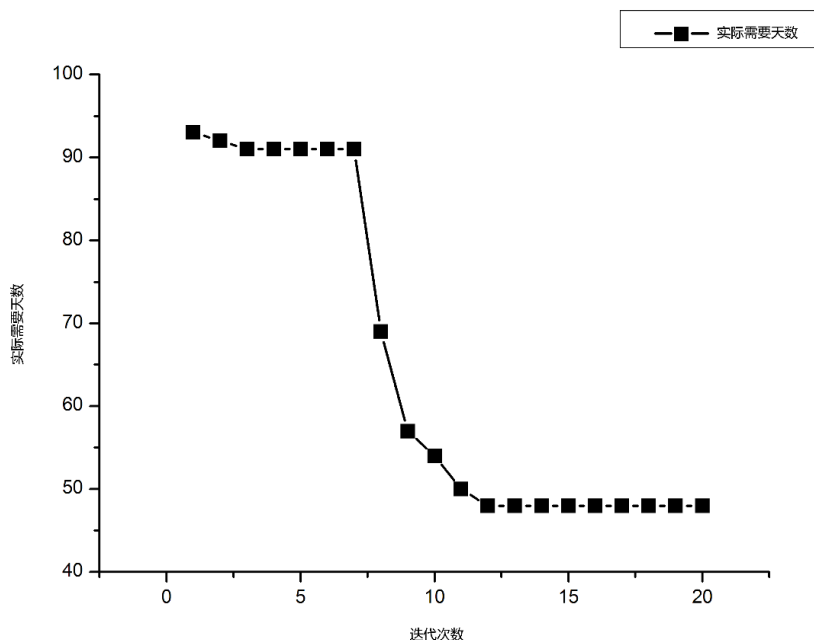


图 3.2 5 台份车型 A 的迭代求优结果

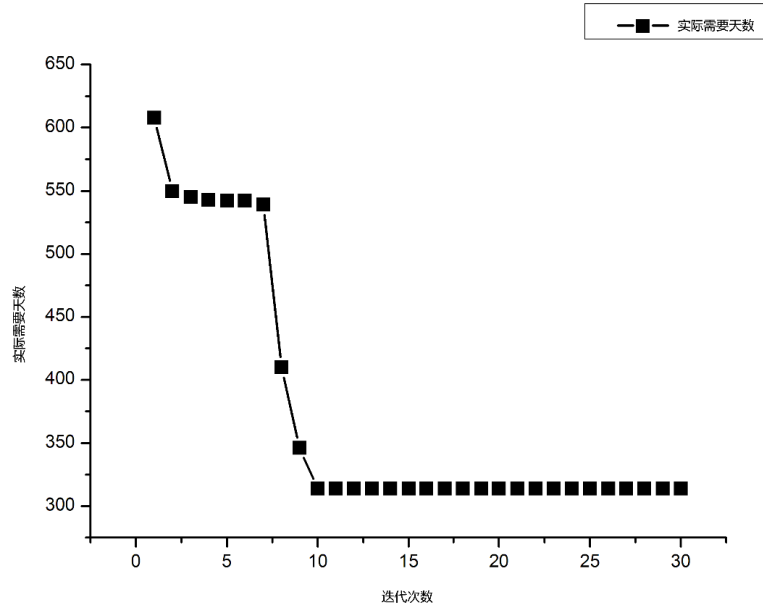


图 3.3 30 台份车型 A 的迭代求优结果

为了对本算法的优化结果进行评估，试验中对零件的收缩分布情况进行了统计。表 3.4、表 3.5 为进行迭代寻优后，对零件收缩分布情况的统计表。

表 3.4 15 台份车型 A 零件收缩分布情况

收缩情况	次数	所占比例
没有收缩	2347	49.33%
未收缩至保障数	130	2.73%
收缩至保障数	2281	47.94%

表 3.5 20 台份车型 A 零件收缩分布情况

收缩情况	次数	所占比例
没有收缩	2060	43.30%
未收缩至保障数	132	2.77%
收缩至保障数	2566	53.93%

从表3.4、表3.5统计结果可以看出，通过迭代寻优后，零件在每个车间的投产数量得到校正，验证了本算法提出的作业计划与调度一体化的方案。以部件P1为代表，调度结果显示如表3.6所示，从最底层的零件9开始，零件5、零件3、零件1的投产数量通过迭代求解后得到一个在参考数和保障数之间的数量，也就是说，通过迭代寻优的调度算法的反馈，重新修正不合理的作业计划。因此，本文在有效解决调度问题的同时，修正了作业计划不合理的部分，实现了作业计划和调度一体化。

表 3.6 P1 的零件投产信息

零件	参考数	保障数	流入数	流出数
1	60	30	45*	45*
2	60	30	60	60
3	60	30	45*	45*
4	120	60	120	120
5	240	120	180*	180*
6	60	30	60	60
7	960	480	960	960
8	960	480	960	960
9	240	120	180*	180*

备注：加*表示需要调整的流入数与流出数

3.4 本章小结

我们的智能优化排产软件除了要实现跨车间条件下的生产调度之外，还要实现相应的制定作业计划的功能，因此要设计作业计划优化算法。跨车间条件下的生产计划与普通生产计划的不同之处在于，由于跨车间的存在，使同一个零件流经不同车间，因此一个车间的生产计划就很可能与其他车间相关联。如果不考虑这种关联性，就极有可能造成物流不平衡。我们考虑各车间的生产能力差异，通过迭代优化生成每种零件在每个车间的最优的流入数和流出数，从而实现了跨车间条件下的作业计划优化。

第4章 智能优化排产软件开发

近年来，随着设备的升级和工艺的革新，生产系统和生产过程的复杂性和精密程度不断提高，为生产管理和调度控制带来新的挑战。目前军工制造企业面临的生产问题是生产过程复杂、加工路线长、零部件数量庞大种类繁多、十余个车间交互频繁。该现状导致生产管理人员在制定生产计划和调度时遇到困难，难以从分厂层面进行统筹规划和过程监控。为此我们研制了智能优化排产软件以协助军工装备制造企业制定生产计划和调度。

4.1 软件功能介绍

我们开发的智能优化排产（见图4.1）能够有效地解决军工制造企业面临的各车间生产能力不平衡以及生产任务跨车间的频繁转移等问题，解决的关键问题如下：



图 4.1 系统欢迎页面

- (i) 生产计划
 - (a) 能够保证各车间生产计划的准确性和可执行性；
 - (b) 能够根据生产任务和资源状态，动态分析出窄口（瓶颈）车间、窄口工序和窄口设备的分布，在此基础上生成基于各车间能力的差异的生产计划（见图4.2）；



图 4.2 窄口分布及窄口收缩



图 4.3 跨车间生产计划

(c) 能够根据零件的跨车间工艺路线，实现跨车间统筹的生产计划，解决车间对零件的盲等待造成的资源浪费问题（见图4.3）。

(ii) 生产调度

(a) 实现了数千种零件、数万道工序、近千台设备、多种工艺类型同时在线的

高效生产调度；

- (b) 实现了紧急订单的动态调度；
- (c) 实现了窄口区域的动态逐级收缩调度；
- (d) 实现了跨车间和跨工艺类型的完整制造链调度。

上述相关页面见图4.4和图4.5。

零件名称	图号	原计划年月	开工时间	完工时间	排产数量	车型	查看工序详情	查看车间详情
泵马达联接螺栓	CB3001.00.003	2012-01	2012-1-1	2012-1-17	40	CS1001	查看工序详情	查看车间详情
耳轴螺栓	CB3001.00.004	2012-01	2012-1-1	2012-1-5	180	CS1001	查看工序详情	查看车间详情
变速器联接螺栓	CB3001.00.009	2012-01	2012-1-1	2012-1-16	360	CS1001	查看工序详情	查看车间详情
传感器盖	CB3001.03.003	2012-01	2012-1-18	2012-1-19	15	CS1001	查看工序详情	查看车间详情
泵组盖	CB3001.03.005	2012-01	2012-1-2	2012-4-3	10	CS1001	查看工序详情	查看车间详情
润滑盖	CB3001.03.006	2012-01	2012-1-11	2012-1-19	15	CS1001	查看工序详情	查看车间详情
变矩器出口盖	CB3001.03.007	2012-01	2012-1-12	2012-5-7	10	CS1001	查看工序详情	查看车间详情
插管螺堵	CB3001.03.008	2012-01	2012-1-1	2012-3-9	10	CS1001	查看工序详情	查看车间详情
变速插管	CB3001.03.009	2012-01	2012-1-1	2012-1-6	20	CS1001	查看工序详情	查看车间详情
主箱体	CB3001.03.031	2012-01	2012-1-2	2012-2-3	10	CS1001	查看工序详情	查看车间详情
螺圈	CB3001.03.043	2012-01	2012-1-10	2012-1-26	20	CS1001	查看工序详情	查看车间详情
螺塞	CB3001.03.044	2012-01	2012-1-10	2012-2-9	20	CS1001	查看工序详情	查看车间详情
插管	CB3001.04.005	2012-01	2012-1-1	2012-1-25	12	CS1001	查看工序详情	查看车间详情
定位套	CB3001.04.006	2012-01	2012-1-1	2012-1-3	60	CS1001	查看工序详情	查看车间详情
螺堵	CB3001.04.008	2012-01	2012-1-2	2012-1-3	60	CS1001	查看工序详情	查看车间详情

图 4.4 三千余种零件的排产结果图表

(iii) 生产管理

(a) 实现了设备管理，将设备分布、加工能力、维修保养等信息应用在排产过程中，将设备利用率、窄口设备分布及负载程度等结果体现在排产方案中（见图 4.6）；

(b) 实现了工具工装的管理，并能针对使用数控刀具的工序给出最优的刀具选配与组装方案；



图 4.5 大规模零件的排产结果甘特图

(c) 实现了企业-车间-设备多级生产日历的管理，能够动态设置企业/车间/设备的作息时间、加班时间、节假日、倒班规则等（见图4.7，图4.8）；

(d) 实现了毛坯、在制品、成品等库存管理。



图 4.6 设备利用率统计

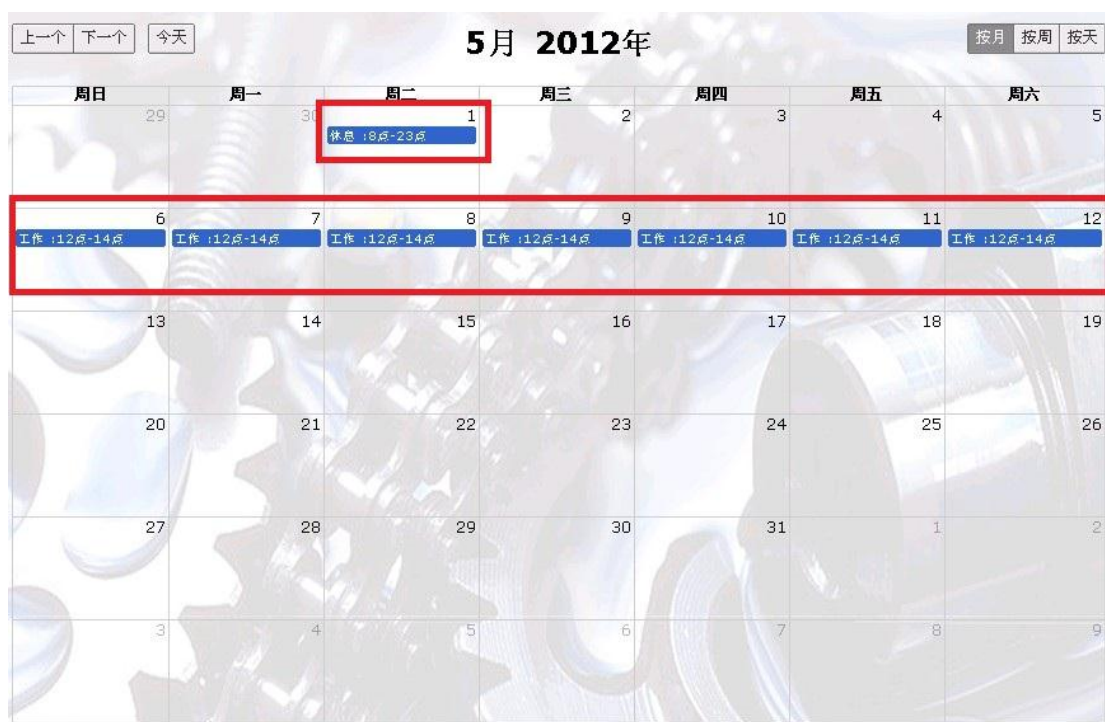


图 4.7 生产日历设置

(iv) 灵活的人机交互

(a) 排产结果能以图表、甘特图等方式进行展示;

(b) 支持按照时间段、车间、设备、零件等多种索引进行查询;

(c) 允许生产管理人员动态定制相关排产策略。



图 4.8 设备班次设置

在研制该智能排产系统之前，企业生产计划的制定缺乏科学依据，制定出的计划不能保证有可行的调度方案；生产调度依赖人工经验，遇到跨车间或工艺路线长等复杂情况只能通过人为沟通，准确性和效率都很不理想；同时生产资源与物流缺乏统一管理。这些方面都为企业提高生产效率、缩短生产周期带来很大障碍。该智能排产系统很好的解决了上述问题。

4.2 软件系统结构

4.2.1 网络结构

在软件的研制开发过程中，我们将实用性与先进性结合，采用成熟技术，我们采用的是B/S架构，B/S最大的优点就是系统部署简单，升级、维护方便。客户端可以在任何地方进行操作而不用安装任何专门的软件，只要有一台能上网的电脑就能使用，客户端零维护。系统的扩展非常容易，只要能上网，再由系统管理员分配一个生产管理人员名和密码，就可以使用了。并且一个客户端可以支持多生产管理人员使用，有利于企业有效的部署客户端，节约成本。只需系统管理员负责日常的维护，就可以保证整个系统的所有终端正常工作，网络部署示意图如图4.9所示。

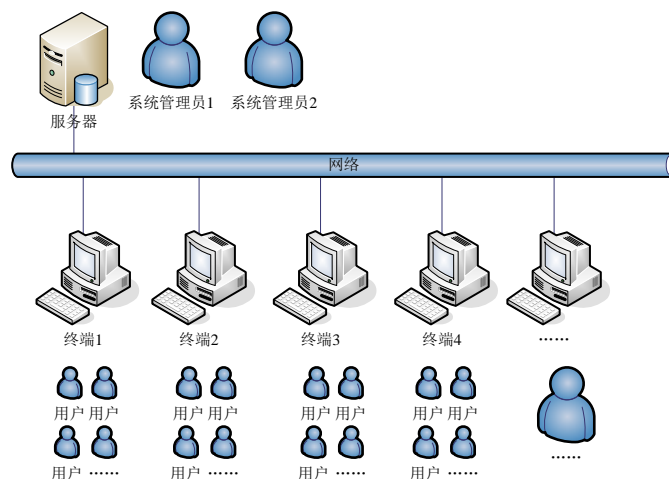


图 4.9 系统网络部署示意图

4.2.2 技术架构

系统设计采用的是J2EE技术架构，应用了Struts框架、Hibernate持久层、XML技术、CSS技术、AJAX技术、SQL2000数据库技术等。J2EE是一个开放的、基于标准的平台，它支持开发、部署和管理N层结构、面向Web的、以服务器为中心的企业级应用。J2EE具有较高的安全性和可移植性，并且表示层和数据访问层分离,层次更

清晰,灵活性好,维护成本低。

J2EE规范包括三个部分：应用组件、服务组件和通信组件。J2EE技术框架如图4.10所示。

(1) 应用组件

应用程序开发者提供applets、servlet、JSP、JavaBeans、EJBs等组件。这些组件运行在J2EE容器里，容器由中间件厂商提供，比如Sun的J2EE，一个J2EE容器需要提供一定的服务和通讯。

(2) 服务器组件

组件可以通过一组API访问的应用服务器功能。例如，一个组件可以通过使用JDBC API访问关系数据库，可以使用JNDI API访问命名服务。需要Security (JAAS)、JTA/JTS、XML支持。XML描述符允许客户方便地自定义组件结构。

(3) 通信组件

J2EE定义了分布式的面向对象的结构。在分布式系统中，组件需要能够彼此通信，所以，容器就需提供相应的通信机制。J2EE支持的通信协议如下：

- (a) 互联网协议：TCP/IP、HTTP1.0、SSL3.0；
- (b) 远程方法调用协议：JRMP (Java远程方法协议)；
- (c) 对象管理组织协议：Java IDL、RMI/IIOP、与CORBA对象交互；
- (d) 消息技术；
- (e) 数据格式：HTML、XML、JAR文件、class文件。

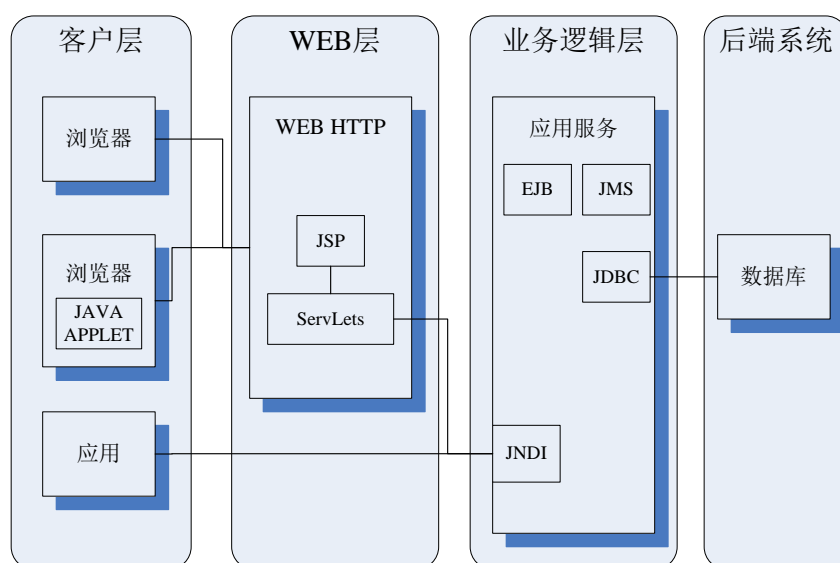


图 4.10 J2EE 技术框架

4.3 模块详细设计

按照排产软件实现功能划分的三大模块，分别是（1）生产计划；（2）生产调度；（3）生产管理。下面分别对各个子模块的具体功能描述、业务逻辑以及类实现细节进行阐述。

4.3.1 生产计划

（1）快速定制排产策略

在“开始排产”页面中，为生产管理人员提供定制排产策略的功能，包括工时调整和动态分批策略。其中工时调整可以定制4个工时区间的压缩比例，生产管理人员可以根据自己的需要调整各个工时区间的压缩比例（0-100%）；动态分批策略中生产管理人员可以调整最大分批数（默认批次为1）、时间临界值（默认为无穷大）和数量临界值（默认为无穷大）。当生产管理人员定制排产策略之后，点击开始排产，系统将按照生产管理人员定制的策略进行排产。实现定制排产策略的具体处理流程如图4.11，业务逻辑图如图4.12：

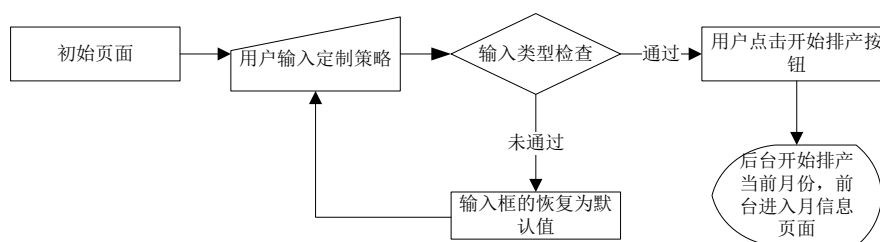


图 4.11 定制排产策略处理过程

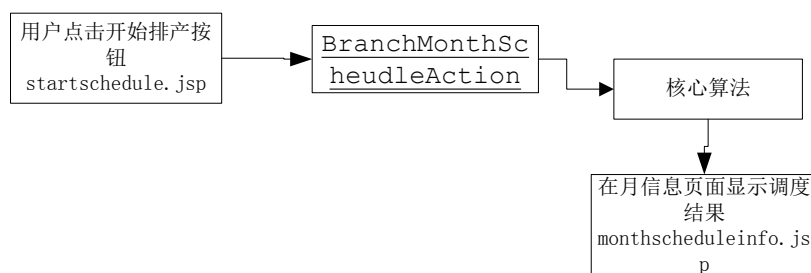


图 4.12 定制排产策略业务逻辑图

（2）查看当前月排产的零件信息

（i）显示当前月排产的零件信息

进入当月生产计划页面后，显示本月生产的零件信息，其处理流程如图4.13所示。

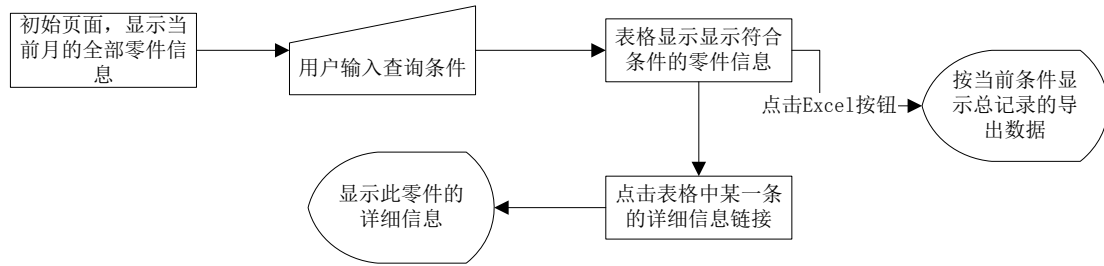


图 4.13 分厂月计划详情显示处理流程

针对上述功能，对查询零件信息设计如下类结构，见图4.14，由底层数据库表 YhYjh 封装成 Dao 类，在向上创建 Service 类，为 Action 类提供具体功能，最后由 startschedule.jsp 页面触发操作，完成零件信息的获取。

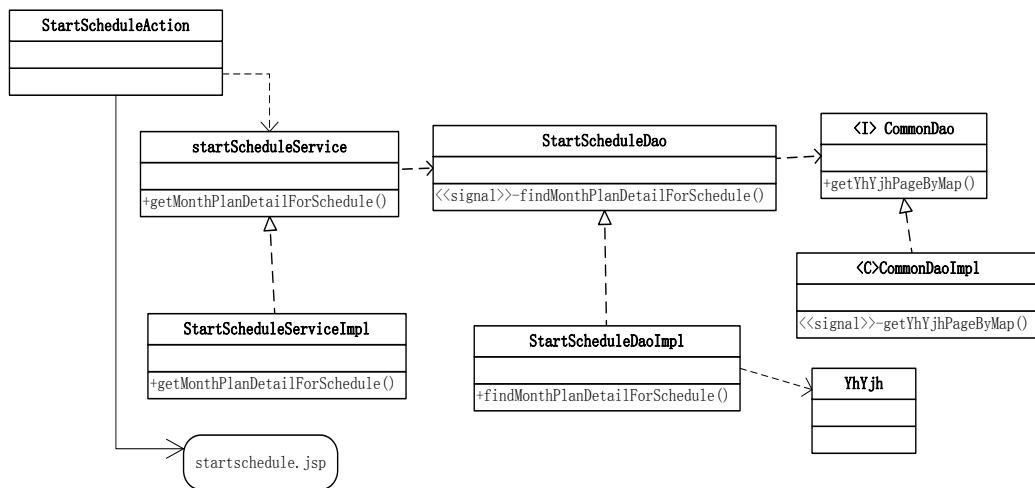


图 4.14 零件信息查看类图

(ii) 显示零件详细信息

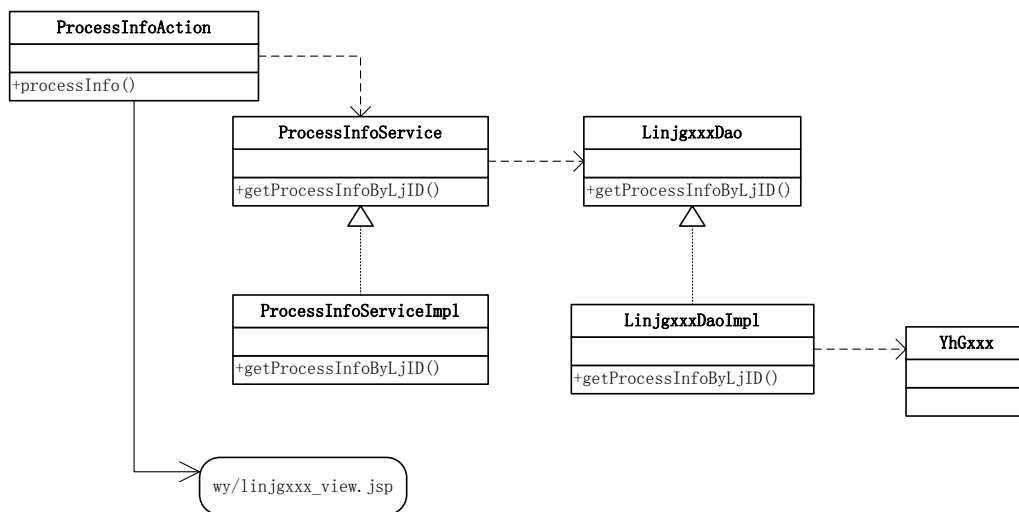


图 4.15 零件详细信息查看类图

每一个零件均可以查看其详细信息，包括每道工序加工的车间号、设备号以及加工时间等，实现类图如图4.15所示。由底层数据库表YhYjh封装成Dao类，在向上创建Service类，为Action类提供具体功能，最后由linjgxxx_view.jsp页面触发操作，完成零件信息的获取。

4.3.2 生产调度

(1) 图表查询排产结果

当某月计划排产结束后，生产管理人员可以在月信息页面，点击查看排产结果链接，进入结果展示页面，默认即为图表查询页面。该模块可以查看零件和查看设备的详细信息。具体处理流程如下：

(i) 零件查询流程

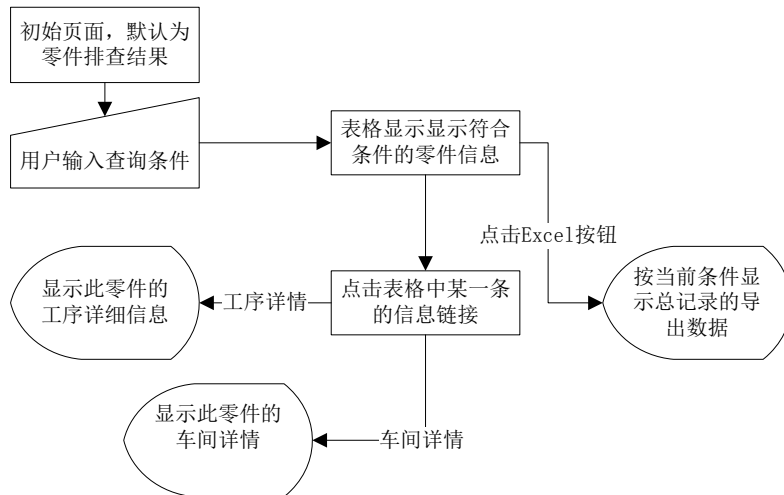


图 4.16 零件查询信息处理过程

(ii) 设备查询流程

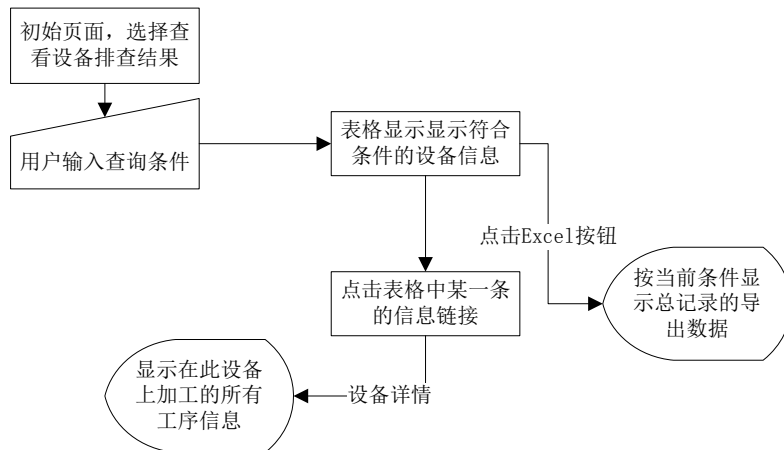


图 4.17 设备查询信息处理过程

针对上述功能，对查询零件信息设计如下类结构，由底层数据库表YhGxpcxq和YhYpcjg封装成Dao类，在向上创建Service类，为Action类提供具体功能，最后由viewWorkpieceInTable.jsp页面触发操作，完成零件信息和设备信息的获取。

(iii) 零件查询类图

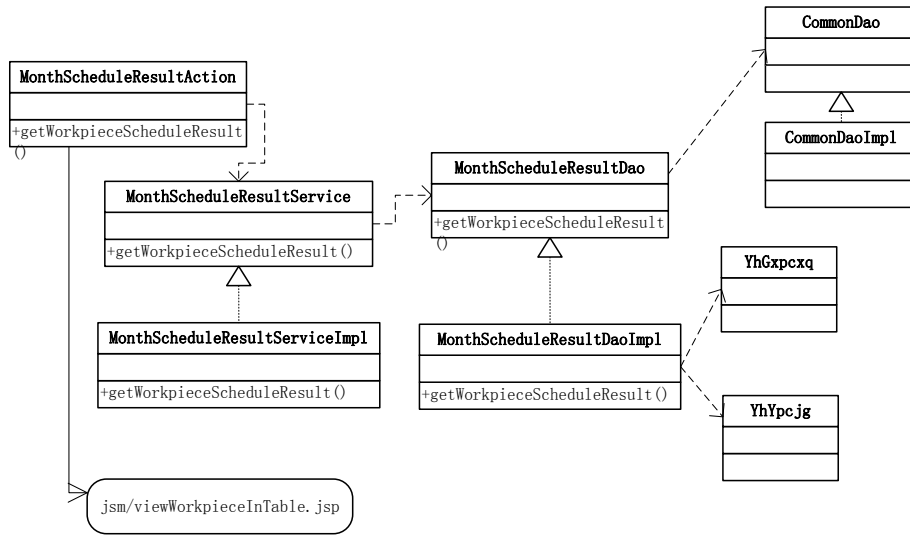


图 4.18 图表查询调度结果中零件查询类图

(iv) 设备查询类图

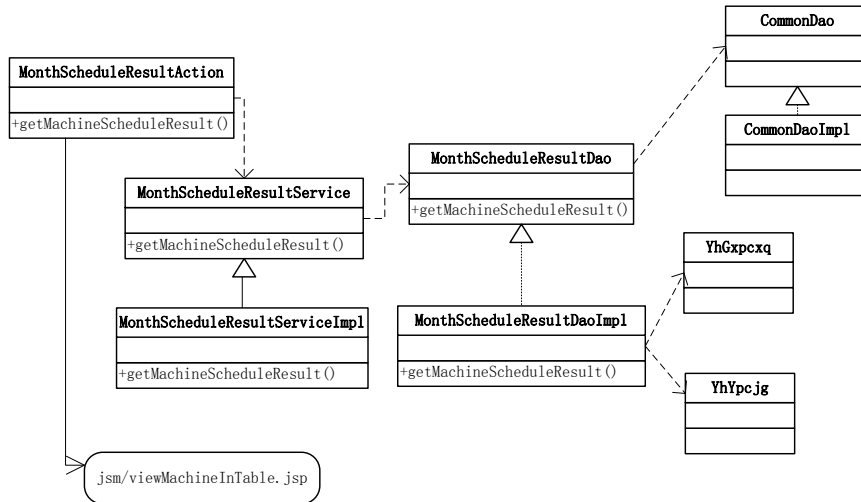


图 4.19 图表查询调度结果中设备查询类图

(2) 甘特图查看排产结果

当某月计划排产结束后，生产管理人员可以在月信息页面，点击查看排产结果链接，进入结果展示页面，以甘特图的形式查看排产结果。该模块同样可以查看零件和查看设备的详细信息。具体处理流程同图表查询，类图设计如下：

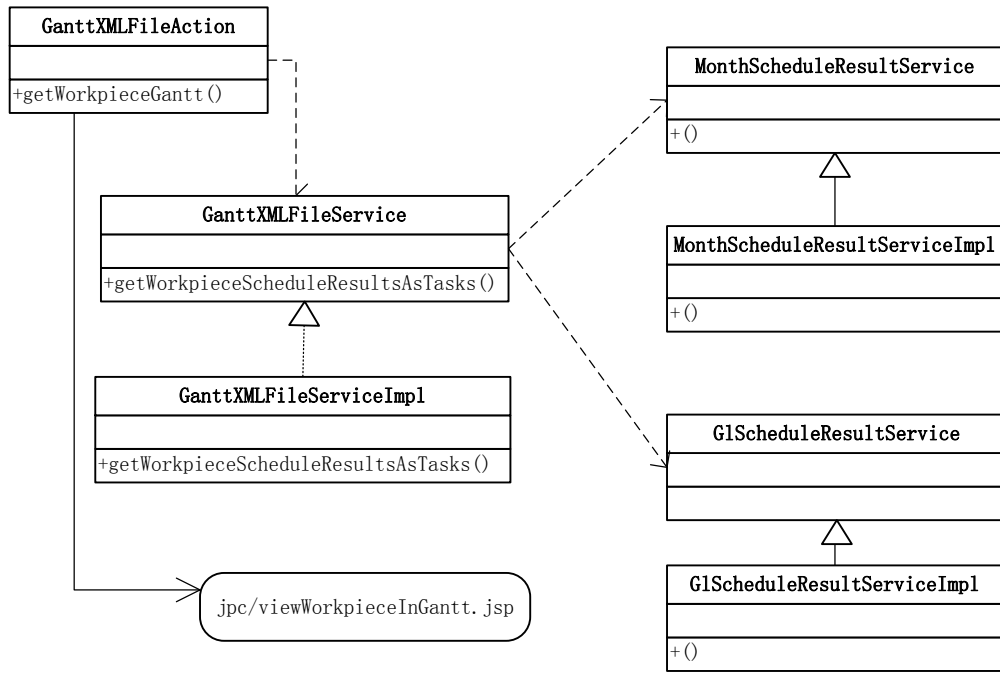


图 4.20 甘特图查询调度结果中零件查询类图

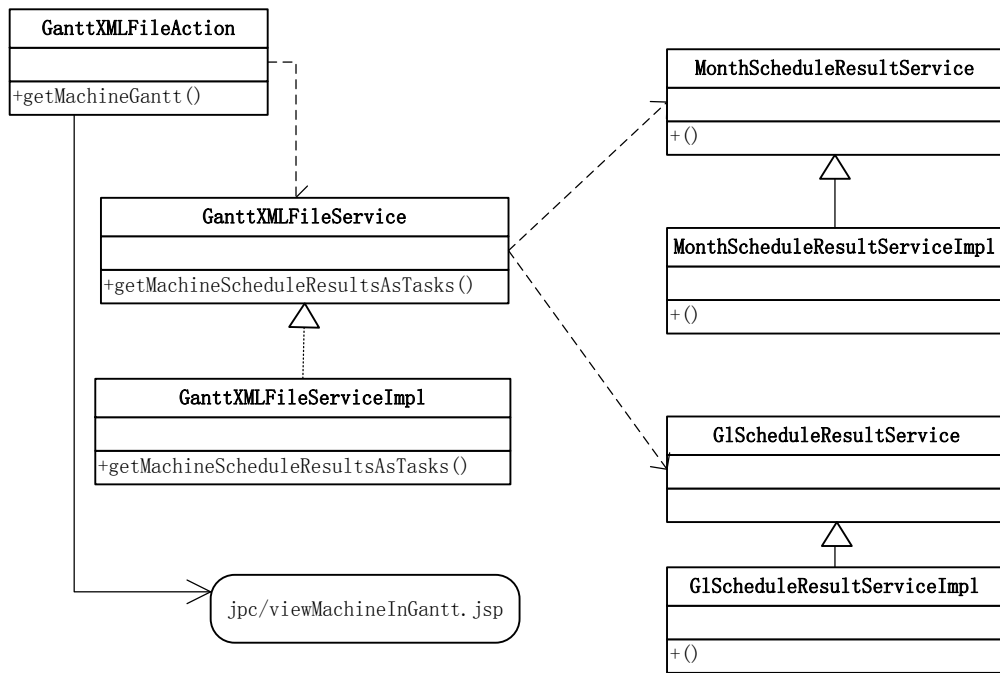


图 4.21 甘特图查询调度结果中设备查询类图

(3) 窄口查看

在零件生产过程中出现严重制约生产计划的瓶颈称为窄口。对于窄口工序，表示为生产过程中最耗费时间部分或者明显落后于平均生产节拍的工序；对于窄口设备，表示为提供能力无法满足所需求的生产力要求的设备。本子模块提供灵活查看

窄口工序和窄口设备的功能。详细操作流程如下：

(i) 工序查询流程

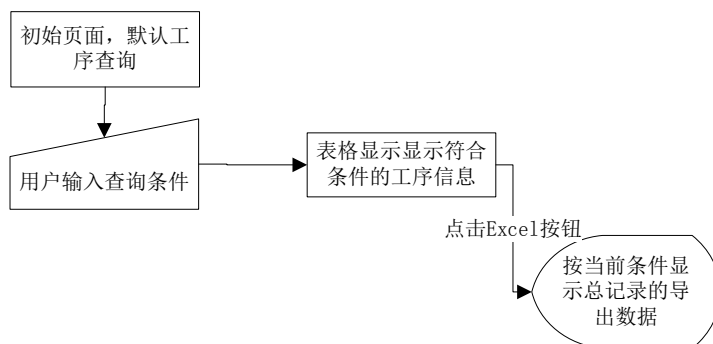


图 4.22 零件查询信息处理过程

(ii) 设备查询流程

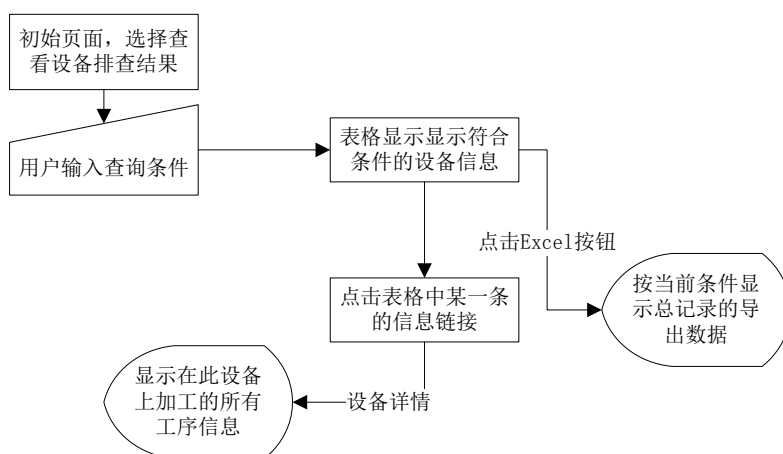


图 4.23 设备查询信息处理过程

根据上述功能，为本模块设计如下类图设计，具体如下。

(iii) 窄口工序查询类图

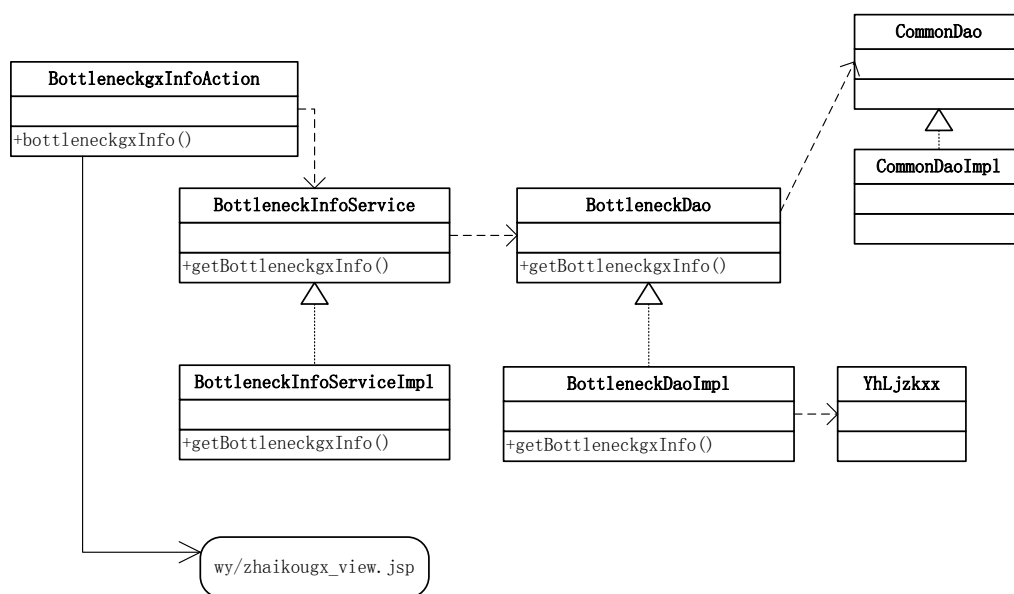


图 4.24 窄口工序查询类图

(iv) 窄口设备查询类图

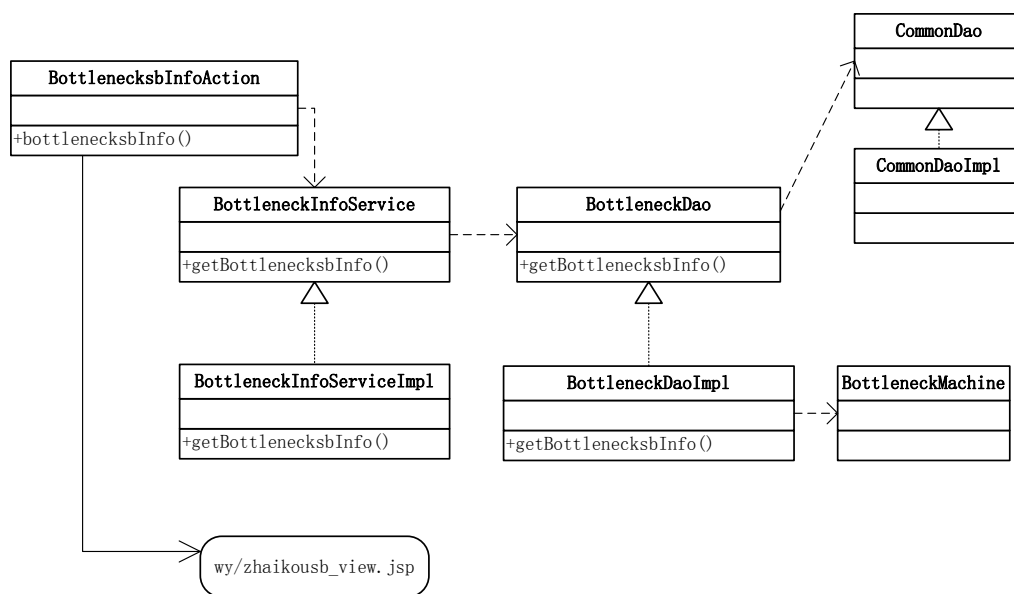


图 4.25 窄口设备查询类图

(4) 车间计划查看

为了更加清晰地呈现软件解决的物流平衡的问题，本模块提供查询车间详细计划的功能，包括零件在各车间的流入数和流出数。生产管理人员可以自由选择查看流入和流出数是否相等的零件。详细设计的类图结构如下：

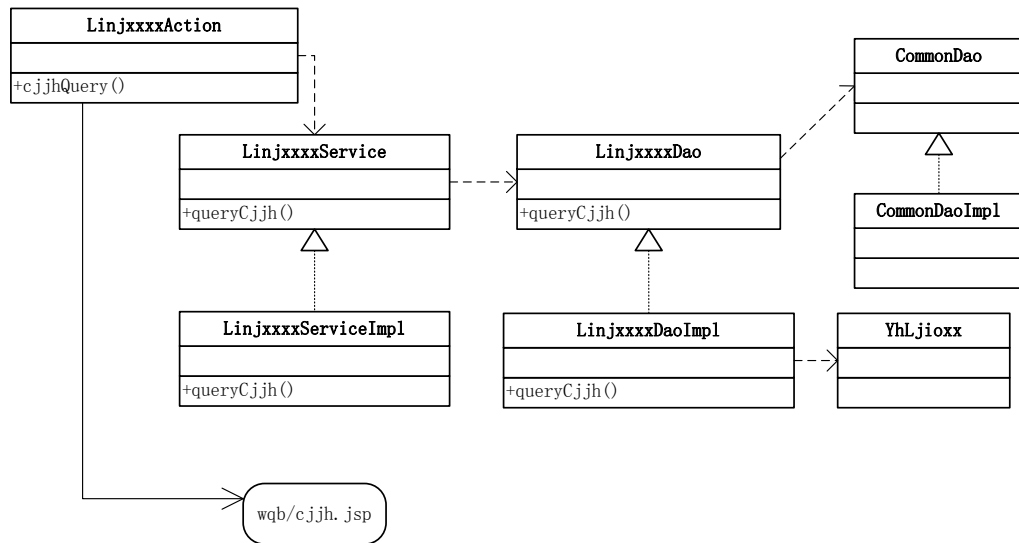


图 4.26 车间计划类图

4.3.3 生产管理

(1) 生产日历查看

生产日历为生产管理人员提供查看生产规则的页面，可以按日历和表格两种形式查看，处理流程如图4.27。生产管理人员可以根据自己的需求选择查询条件，包括车间、设备、时间，其中机器为必选项，选择按日历查看，点击查询生产日历，该页面下方的日历显示生产管理人员查询的生产规则；生产管理人员可以按月、按周、按天详细查看规则的详细情况。详细类图见图4.28所示。

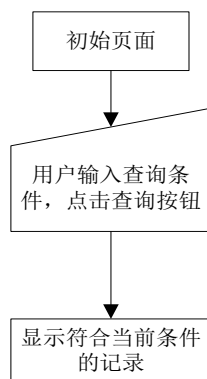


图 4.27 生产日历查看处理流程图

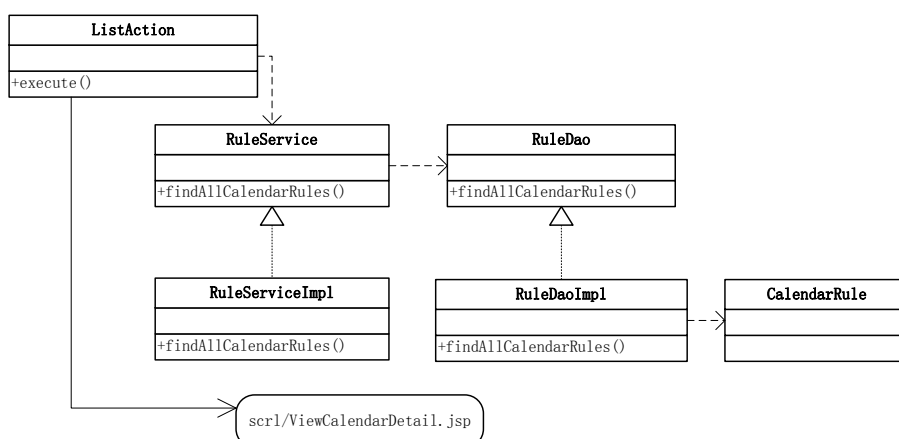


图 4.28 生产日历查看类图

(2) 规则设置

规则设置为生产管理人员提供个性化设定生产规则的页面，生产管理人员可以根据生产实际情况，设置某个时间段工作或者休息，某台设备工作或者休息，主要功能包括新建规则、删除规则和查看工厂日历，其中具有分厂级权限的生产管理人员具有设置所有设备规则的权限，而车间级生产管理人员只能设置本车间内的设备的规则。详细类图设计如下，见图4.29和图4.30所示。

(i) 新建规则

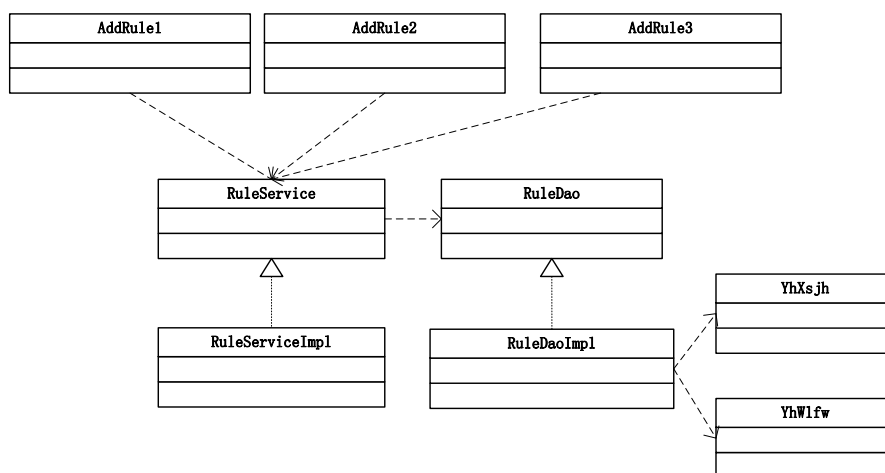


图 4.29 新建规则类图

(ii) 删除规则

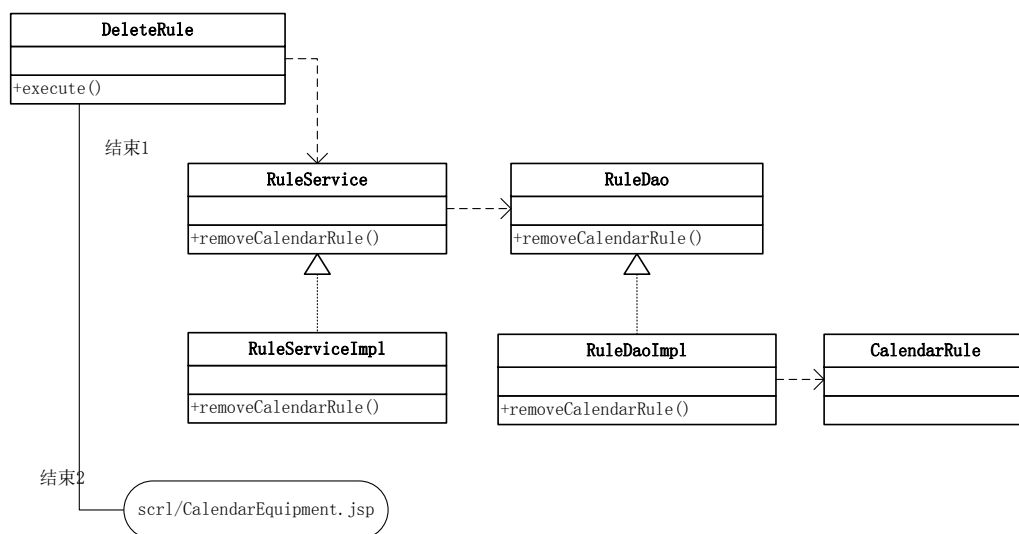


图 4.30 删除规则类图

(3) 班次设置

为生产管理人员提供调整班次时间的页面，班次分为单班、双班和三班，单班包括2个时间区间，双班包括3个时间区间，三班包括1个时间区间，生产管理人员可以根据实际需求，修改各个时间区间的起止时间；点击“重置为默认”，恢复最近一次保存结果（即数据库中的班次设置情况）；点击“保存设置结果”，保存当前对班次的修改情况。详细类图如图4.31和图4.32所示。

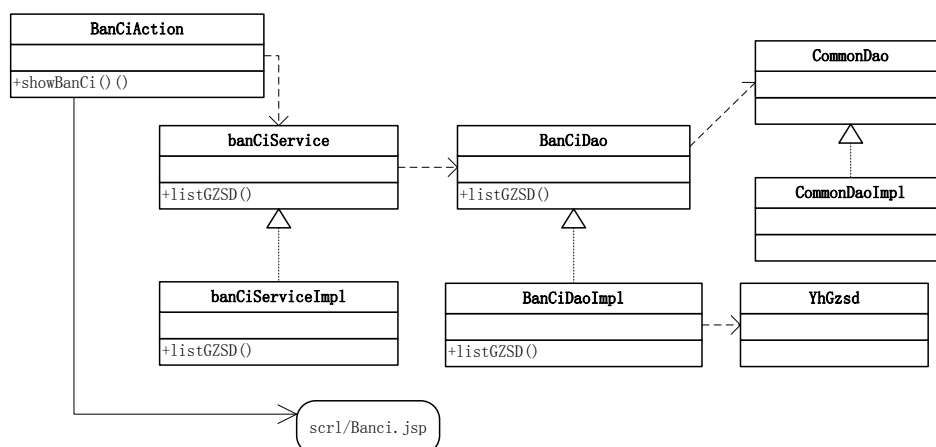


图 4.31 班次重置设置时的类图

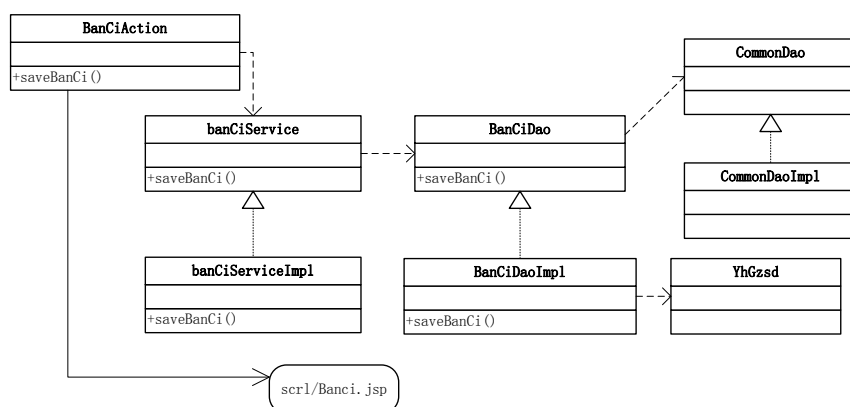


图 4.32 班次保存时的类图

(4) 工作日设置

工作日设置为生产管理人员提供了查看和修改工作日范围的页面，生产管理人员可以从下拉框中选择工作日范围，点击保存，保存当前对工作日的设置。详细类图如图4.33所示。

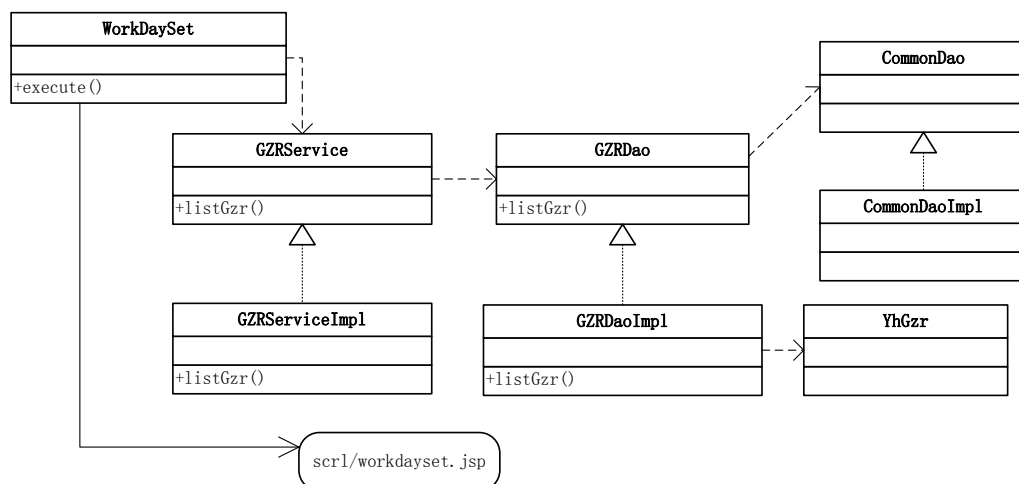


图 4.33 工作日设置类图

4.4 软件使用说明

系统在使用的时候，必须先进入系统登录模块，正确登录系统后，才能使用其余的模块。具有不同权限的生产管理人员登录系统后，能够使用的模块是不相同的。

系统通过各个模块实现的3大类的功能。

1、通过本系统，实现对分厂月计划的信息化、规范化管理，包括：

(1) 生产月设置：包括查看各年生产月的起止时间、修改生产月的起止时间的

功能，满足生产月与自然月不一致的实际需求；

(2) 排产设置：可以对生产月快速定制排产策略，可以调整工时的压缩比例，制定动态分批策略；

(3) 计划完成情况：根据分厂实际完成情况，显示月计划中新追加的计划和剩余计划的完成情况；

2、基于智能优化方法，解决生产调度问题，将排产结果通过多元化方式呈现，包括：

(1) 以图表、甘特图等方式查看零件在各车间中设备上的加工信息；

(2) 查看窄口信息，通过离线计算窄口系数，反映出在生产过程中处于瓶颈的工序和设备；

(3) 车间计划的详细信息，通过计算零件流入车间和流出车间的数量，体现车间之间的物流情况，计算收缩次数进而有效地实现车间之间的物流平衡。

3、实现生产管理中生产日历、设备设置等实用功能，包括：

(1) 查看生产日历：允许生产管理人员以日历或者表格方式查看生产日历，内容包括生产管理人员在规则设置中设定规则的详细信息。

(2) 规则设置：生产管理人员可以根据实际需要个性化创建规则，该规则直接影响到排产结果。

(3) 生产设置：提供设备设置、班次设置和工作日设置的功能，生产管理人员可以根据需要修改设备的班次，调整班次时间以及修改工作日区间。

软件运行的硬件条件包括：

(i) 服务器

服务器使用奔腾4以上的CPU，主频在3.0以上；内存在1G以上。

(ii) 客户机

客户机使用奔腾4以上的CPU，主频在2.8以上；内存在256M以上。

(iii) 网络环境

网络环境为标准以太网传输方式，符合TCP/IP的网络协议。以及运行在TCP/IP协议族之上的HTTP应用协议。网络带宽在10M以上。

支持软件包括：

(i) 服务器

服务器建议使用的浏览器版本IE8.0 兼容模式；操作系统使用Windows Server

2000以上版本。

(ii) 客户机

客户机建议使用浏览器版本IE8.0兼容模式；浏览器分辨率在1024×768以上；操作系统使用Windows2000、WindowsXP；操作系统对Excel格式文件的支持；操作系统支持BMP、JEPG、GIF等格式的图片显示。

4.4.1 安装与初始化

(i) JBoss的安装

解压JBoss的压缩包jboss-4.2.3.GA.zip，解压缩后将其放置到一个不带空格的目录下，如D:\Program Files\jboss-4.2.3.GA

同时在“环境变量设置”中设置名为JBOSS_HOME的环境变量，值为JBoss的安装路径，具体操作如下：

右键点击桌面上“我的电脑”图标，选择“属性”：



图 4.34 属性示意图

在弹出的系统属性设置窗口，选择“高级”选项卡：

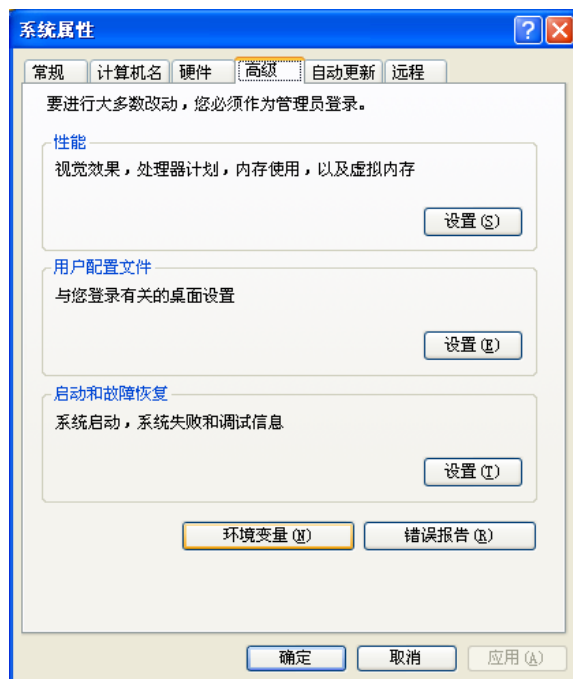


图 4.35 高级选项卡

点击环境变量，在系统环境中新建变量JBoss_HOME，变量值为安装路径，例如D:\Program Files\jboss-4.2.3.GA

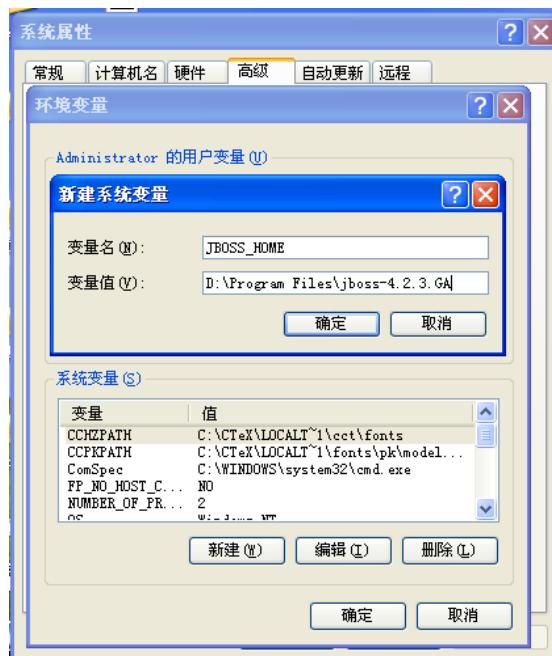


图 4.36 环境变量设置

保存设置后，JBoss的安装工作已经结束，可通过如下方式测试安装是否成功：
运行JBoss安装目录“bin”下的run.bat，如果窗口中没有出现异常，且出现如下信息：

10:16:19,765INFO[Server]JBoss(MX
MicroKernel)[4.2.1GA(build:SVNTag=JBoss_4_2_1_GA date=201112051605)]Start in
30s:828ms字样，则表示安装成功。

生产管理人员可以通过访问：<http://localhost:8080/>进入JBoss的欢迎界面，点击JBoss Management下地JMX Console可进入JBoss的控制台。

若启动失败，可能由以下原因引起：

1. JBoss所用的端口(8080,1099,1098,8083等)被占用。一般情况下可将默认8080端口修改为8088端口，方法为进入JBoss安装目录下...
\\server\\default\\deploy\\jboss-web.deployer文件夹中server.xml文件，在此文件中搜索8080，将其修改成8088；

2. JDK安装不正确；

3. JBoss下载不完全。

(ii) JRE的安装

下载安装jre1.6版本以上的安装包，设置系统环境变量JAVA_HOME为安装目录，保存设置即可。

(iii) 程序的部署

将程序复制到指定的文件夹后，需要配置数据库的连接字符串。打开Optimization文件夹，在Optimization\\src的文件夹中有个名为spring.properties的文件。对该文件进行打开编辑，会看到在其中有如下内容。

```
.....  
db.driverClassName=net.sourceforge.jtds.jdbc.Driver  
  
db.url=jdbc\:jtds\;sqlserver\://10.108.13.1\:1433;DatabaseName\=jxc_6  
db.username=sa  
db.password=!@#$$%^
```

其中红色标记的部分是“sa”为数据库的生产管理人员名，“!@#\$\$%^”为生产管理人员名的密码，10.108.13.1为服务器的IP地址，这些需要更改成和该服务器的数据库配置信息一致。如果服务器的数据库中信息做了更改或调整，此处的配置信息也需要进行调整。

4.4.2 生产计划

(i) 分厂月计划设定

本模块主要功能：具有分厂级权限的生产管理人员可以查看和修改分厂各个生产月计划的开始时间和结束时间；具有车间级权限的生产管理人员只允许查看生产月的起止时间，不能修改时间。详细操作流程如下：

在“请选择年份”的下拉框中选择需要查看的年份，如“2011”，点击“确定”，月划分情况显示2011年12个生产月的开始时间和结束时间。

单击任一输入框，可修改各生产月的起止时间。单击“保存设置结果”，保存生产月设置。单击“重置为默认”，恢复最近一次保存结果（即数据库中的生产月设置情况）。界面如下图所示。

图 4.37 生产月设置

数据输入要求起止时间通过悬浮的日历选择，不提供手动输入的功能，同时满足一下输入要求：

- 1、同一月份的开始时间必须早于结束时间
- 2、本月开始时间必须晚于上一月份的结束时间（支持跨年检测）
- 3、生产派工单的月份无法重新设置生产月的起止时间

(ii) 分厂月排产设置

本模块主要包括五项主要功能：快速定制排产策略、图表查询排产结果、甘特图查询排产结果、窄口查询和车间计划查询。其中具有分厂级权限的生产管理人员具有所有权限，而车间级生产管理人员只能查看本车间数据的排产结果。

具有分厂级权限的生产管理人员查看的分厂月排产设置页面：



图 4.38 分厂级权限的分厂月排产设置页面

具有车间级权限的生产管理人员查看的分厂月排产设置页面：



图 4.39 车间级权限的分厂月排产设置页面

1、快速定制排产策略

(1) 为生产管理人员提供快速定制排产策略的页面：

在“开始排产”页面中，为生产管理人员提供定制排产策略的功能，包括工时调整和动态分批策略。其中工时调整可以定制4个工时区间的压缩比例，生产管理人员可以根据自己的需要调整各个工时区间的压缩比例（0-100%）；动态分批策略中生产管理人员可以调整最大分批数（默认批次为1）、时间临界值（默认为无穷大）和数量临界值（默认为无穷大）。当生产管理人员定制排产策略之后，点击开始排产，系统将按照生产管理人员定制的策略进行排产；

(2) 查询零件信息的页面：

生产管理人员可以根据自己的需求选择性的输入查询条件后，点击查询，在本页面列出所有满足查询条件的记录，记录信息包括：图号、零件名称、车型、参考数、保障数以及该零件的详细信息；其中查询条件包括：a.直接查询，即生产管理人员不用任何输入，直接点击查询，此时列出该生产月中所有的零件信息；b.在零件图号，零件名称和车型三者中，可以任意选择性的输入一个，两个或者三个（输入条件支持模糊查询、忽略字母大小写），点击查询，进行联合查询，此时列出满足查询条件的所有记录；

零件信息默认按照零件图号升序排序，同时为生产管理人员提供按图号、零件名称、车型、参考数、保障数的升序及降序排序功能；提供将查询结果导出为Excel表格的功能；

(3) 查看零件详细信息的页面

当生产管理人员查询完后，在(2)页面列出的记录中，详细信息以超链接的形式显示，当生产管理人员点击某条记录的详细信息时，弹出一个新的页面，显示该零件的具体信息，包括工序号、工序名称、加工时间、准备时间、设备型号和车间。查看记录默认按工序号升序排序，同时为生产管理人员提供按照每列字段升序和降序排序的功能，方便生产管理人员查看记录；提供将查看结果导出为Excel表格的功能。

上述功能的操作流程具体如下：

1、排产过程

分厂月计划设置页面：



图 4.40 分厂月排产设置页面

点击“开始排产”进入开始排产页面：



图 4.41 开始排产示意图

在该页面查看分厂月计划零件详情，提供多种查询条件；点击“详细信息”查看零件的工序详情；

零件基本信息					
零件图号: CS212.40.215 零件名称: CM离合器压板 车型: CS212					
零件详细信息(默认按工序号排序)					
工序号	工序名称	加工时间	准备时间	设备型号	车间
5	划线	8.97	0.0		406
10	钻孔	0.0	0.0	Z35	406
15	去毛刺	0.0	0.0		406
20	粗车	35.87	0.0	C630-1	406
21	中检	0.0	0.0		406
25	热处理	19.34	0.0	RCL-19	404
	喷砂	1.82	0.0	RCL-8	404
30	磨平面	11.21	0.0	M7475	406
35	去毛刺	0.0	0.0		406
40	退磁	0.0	0.0		406
45	车内孔	26.91	0.0	CY6140	406
50	车端面	0.0	0.0	CY6140	406
55	中检	0.0	0.0		406
60	线切割	72.14	0.0	FL310	408
65	检验	0.0	0.0		408

1-15 共 15条

输出excel数据 返回上一级

图 4.42 零件信息

点击“返回上一级”，返回开始排产页面，可以快速定制排产策略，点击“开始排产”，系统将按照生产管理人员定制的排产策略进行排产，同时页面自动跳转到分厂月计划设置页面，排产状态更新为“正在排产”；

分厂月排产状态			
生产月	状态	排产时间	操作
1月	正在排产	-	
2月	未排产	-	开始排产
3月	未排产	-	开始排产
4月	未排产	-	开始排产
5月	未排产	-	开始排产
6月	未排产	-	开始排产
7月	未排产	-	开始排产
8月	未排产	-	开始排产
9月	未排产	-	开始排产
10月	未排产	-	开始排产
11月	未排产	-	开始排产
12月	未排产	-	开始排产

1-12 共 12条

图 4.43 正在排产示意图

等待一段时间后，排产结束，状态更新为“查看排产”的超链接，显示排产时间和“开始排产”、“查看排产结果”、“确认排产结果”的超链接。

生产月	状态	排产时间	操作
1月	查看状态	2011-12-05 19:43:13	开始排产 查看排产结果 确认排产结果
2月	未排产	-	开始排产
3月	未排产	-	开始排产
4月	未排产	-	开始排产
5月	未排产	-	开始排产
6月	未排产	-	开始排产
7月	未排产	-	开始排产
8月	未排产	-	开始排产
9月	未排产	-	开始排产
10月	未排产	-	开始排产
11月	未排产	-	开始排产
12月	未排产	-	开始排产

图 4.44 排产完成示意图

点击“查看状态”，查看某个生产月的排产状态（如2011年1月），会出现以下两种状态：



(a) 排产时间不足的情况

(b) 排产成功的情况

图 4.45 分月排产状态

对于排产时间不足的情况，可以调整排产策略，重新排产；对于排产成功的情况确认无误后，可以点击“确认排产结果”。

生产月	状态	排产时间	操作
1月	查看状态	2011-12-05 19:43:13	查看排产结果 生成派工计划
2月	未排产	-	开始排产
3月	未排产	-	开始排产
4月	未排产	-	开始排产
5月	未排产	-	开始排产
6月	未排产	-	开始排产
7月	未排产	-	开始排产
8月	未排产	-	开始排产
9月	未排产	-	开始排产
10月	未排产	-	开始排产
11月	未排产	-	开始排产
12月	未排产	-	开始排产

图 4.46 确认排产结果

点击“生产派工计划”后，状态更新为“已分发”；

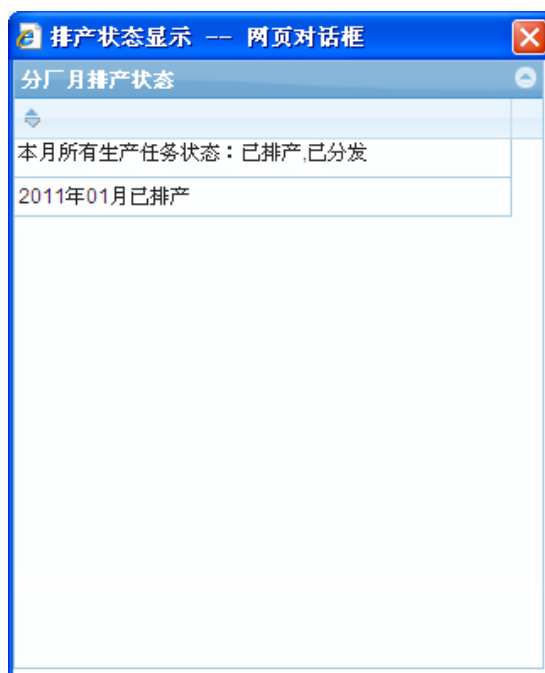


图 4.47 排产状态

4.4.3 生产调度

(i) 查看排产结果

1、图表查询排产结果

生产管理人员可以以图表的形式查询排产结果，提供多种组合查询条件，包括车间、开工时间范围、零件或者设备；其中零件查询条件包括零件图号、零件名称、车型，而设备查询条件包括设备编号、设备名称、设备型号，可以任意选择性

的输入一个、两个或者三个（输入条件支持模糊查询、忽略字母大小写），点击查询，进行联合查询，此时列出满足查询条件的所有记录；

a) 以零件方式查询时，记录包括：零件名称、图号、原计划年月、开工时间、完工时间、排产数量、车型、查看工序详情的超链接和查看车间详情的超链接；

排产结果默认按照车型和零件图号升序依次排序，同时为生产管理人员提供按图号、零件名称、原计划年月、开工时间、完工时间、排产数量和车型的升序及降序排序功能；提供将查询结果导出为Excel表格的功能；

当生产管理人员查询完后，在排产结果页面列出的记录中，查看工序详情和查看车间详情以超链接的形式显示，当生产管理人员点击某条记录的工序详情时，切换一个新的页面，显示该零件的具体信息，包括工序号、工序名称、批次号、开工时间、完工时间、车间、设备编码和排产数量。查看记录默认按工序号升序排序，同时为生产管理人员提供按照每列字段升序和降序排序的功能，方便生产管理人员查看记录；提供将查看结果导出为Excel表格的功能；当生产管理人员点击某条记录的车间详情时，切换一个新的页面，显示该零件的具体信息，包括车型、车间、进入车间的次数、图号、流入数量和流出数量。查看记录默认按照工艺路线排序，同时为生产管理人员提供按照车型、图号、流入数量和流出数量的升序和降序排序的功能，方便生产管理人员查看记录；提供将查看结果导出为Excel表格的功能；

b) 以设备方式查询时，记录包括：设备编号、设备名称、设备型号、设备利用率、设备车间、和设备详情的超链接；

排产结果默认按照设备编号升序排序，同时为生产管理人员提供按各列字段的升序及降序排序功能；提供将查询结果导出为Excel表格的功能；

当生产管理人员查询完后，在排产结果页面列出的记录中，设备详情以超链接的形式显示，当生产管理人员点击某条记录的设备详情时，切换一个新的页面，显示该设备的具体信息，包括设备编号、图号、工序号、工序名称、批次、开工时间和完工时间。查看记录默认按工序开工时间排序，同时为生产管理人员提供按照每列字段升序和降序排序的功能，方便生产管理人员查看记录；提供将查看结果导出为Excel表格的功能；

2、甘特图查询排产结果

生产管理人员可以以甘特图的形式查询排产结果，提供多种组合查询条件，包括车间、开工时间范围、零件或者设备；其中零件查询条件包括零件图号、零件名

称、车型，而设备查询条件包括设备编号、设备名称、设备型号，可以任意选择性的输入一个、两个或者三个（输入条件支持模糊查询、忽略字母大小写），点击查询，进行联合查询，此时满足条件的甘特图等信息；

a) 以零件方式查询时，记录包括：图号、名称、车型、开始时间、结束时间和甘特图；其中甘特图提供放大、缩小、缩小到最小、收缩当页所有记录的功能，同时生产管理人员可以打印甘特图当前屏和打印甘特图当前页；

b) 以设备方式查询时，记录包括：设备编号、名称、型号、开始时间、结束时间和甘特图；其中甘特图提供放大、缩小、缩小到最小、收缩当页所有记录的功能，同时生产管理人员可以打印甘特图当前屏和打印甘特图当前页；

3、窄口查询

窄口查询为生产管理人员提供一个查询窄口工序和窄口设备的页面，提供多种组合查询条件，包括车间、零件或者设备；其中零件查询条件包括零件图号、车型，而设备查询条件包括设备名称、设备型号，可以任意选择性的输入一个或者两个（输入条件支持模糊查询、忽略字母大小写），点击查询，进行联合查询，此时列出满足查询条件的所有记录；

a) 以零件方式查询时，记录包括：图号、工序号、车间、设备型号、车型、收缩次数和窄口系数，记录默认按照窄口系数降序和图号升序依次排序，同时为生产管理人员提供按各列字段的升序及降序排序功能；提供将查询结果导出为Excel表格的功能；

b) 以设备方式查询时，记录包括：车间、设备名称、设备型号、窄口系数和设备详情的超链接，记录默认按照窄口系数降序排序，同时为生产管理人员提供按各列字段的升序及降序排序功能；提供将查询结果导出为Excel表格的功能；

当生产管理人员查询完后，在窄口设备信息页面列出的记录中，设备详情以超链接的形式显示，当生产管理人员点击某条记录的设备详情时，切换一个新的页面，显示该设备的具体信息，包括设备型号、图号、零件名称、工序号和工序名称。查看记录默认按零件图号和工序号依次排序，同时为生产管理人员提供按照每列字段升序和降序排序的功能，方便生产管理人员查看记录；提供将查看结果导出为Excel表格的功能；

4、车间计划查询

车间计划查询为生产管理人员提供一个查询车间内详细计划的页面，提供多种

组合查询条件，包括车间、零件图号、零件名称、车型和流入数与流出数不相等的计划，其中零件图号、零件名称和车型可以任意选择性的输入一个、两个或者三个（输入条件支持模糊查询、忽略字母大小写），点击查询，进行联合查询，此时列出满足查询条件的所有记录；在本页面列出所有满足查询条件的记录，记录信息包括：零件名称、图号、车型、车间、进入车间的次数、流入数量和流出数量；记录默认按照零件图号排序，同时为生产管理人员提供按各列字段的升序及降序排序功能；提供将查询结果导出为Excel表格的功能。

上述功能的操作流程具体如下：

1、查看排产结果

点击“查看排产结果”，可以进入图表查询、甘特图查询、窄口查询和车间计划查询页面。

零件名称	图号	原计划年月	开工时间	完工时间	排产数量	车型	查看工序详情	查看车间详情
------	----	-------	------	------	------	----	--------	--------

图 4.48 图表查询

(1) 图表查询

以零件形式查询排产结果页面：

分厂月排产结果(默认按车型和零件图号排序)								
零件名称	图号	原计划年月	开工时间	完工时间	排产数量	车型	查看工序详情	查看车间详情
通气器体	CS120.08.198-2	2011-01	2011-1-3	2011-1-3	40	CS212	查看工序详情	查看车间详情
螺栓	CS120.08.238	2011-01	2011-1-3	2011-1-3	40	CS212	查看工序详情	查看车间详情
传动箱	CS212.40.00	2011-01	2011-1-25	2011-1-28	20	CS212	查看工序详情	查看车间详情
端盖	CS212.40.002	2011-01	2011-1-4	2011-1-10	20	CS212	查看工序详情	查看车间详情
轴承座	CS212.40.003	2011-01	2011-1-7	2011-1-11	20	CS212	查看工序详情	查看车间详情
套筒	CS212.40.004	2011-01	2011-1-5	2011-1-10	20	CS212	查看工序详情	查看车间详情
套筒	CS212.40.005	2011-01	2011-1-5	2011-1-10	20	CS212	查看工序详情	查看车间详情
齿轮Z28	CS212.40.006	2011-01	2011-1-3	2011-1-7	40	CS212	查看工序详情	查看车间详情
轴1	CS212.40.007-1	2011-01	2011-1-3	2011-1-13	20	CS212	查看工序详情	查看车间详情
套筒	CS212.40.009	2011-01	2011-1-5	2011-1-10	20	CS212	查看工序详情	查看车间详情
轴5总成	CS212.40.01	2011-01	2011-1-17	2011-1-17	20	CS212	查看工序详情	查看车间详情
端盖	CS212.40.010	2011-01	2011-1-5	2011-1-10	20	CS212	查看工序详情	查看车间详情
轴承座	CS212.40.011	2011-01	2011-1-7	2011-1-11	20	CS212	查看工序详情	查看车间详情
轴承盖	CS212.40.013	2011-01	2011-1-5	2011-1-10	20	CS212	查看工序详情	查看车间详情
轴承盖	CS212.40.014	2011-01	2011-1-4	2011-1-10	20	CS212	查看工序详情	查看车间详情

1 - 15 共 210 条

[输出excel数据](#)

图 4.49 零件排产结果

查看工序详情页面：

零件基本信息 零件图号: CS120.08.198-2 零件名称: 通气器体 原计划年月: 2011-01 车型: CS212

工序调度详情(默认按工序号排序)							
工序号	工序名称	批次号	开工时间	完工时间	车间	设备编码	排产数量
5	切断	1	2011-01-03 08:10:58	2011-01-03 08:19:09	406	03110601291	13
5	切断	2	2011-01-03 08:19:09	2011-01-03 08:27:21	406	03110601291	13
5	切断	3	2011-01-03 08:27:21	2011-01-03 08:36:10	406	03110601291	14
10	车端面	1	2011-01-03 10:03:00	2011-01-03 10:15:17	406	03120201118	13
10	车端面	2	2011-01-03 10:03:00	2011-01-03 10:15:17	406	03120201127	13
10	车端面	3	2011-01-03 10:04:32	2011-01-03 10:17:46	406	03120201187	14
15	钻孔	1	2011-01-03 10:17:46	2011-01-03 10:19:39	406		40
20	修毛刺	1	2011-01-03 10:19:39	2011-01-03 10:44:07	406	03140101106	13
20	修毛刺	2	2011-01-03 10:19:39	2011-01-03 10:19:39	406	03140101024	13
20	修毛刺	3	2011-01-03 10:19:39	2011-01-03 10:46:00	406	03140101025	14
25	铣槽	1	2011-01-03 10:46:00	2011-01-03 10:48:50	406		40
30	修毛刺弯曲	1	2011-01-03 10:48:50	2011-01-03 11:09:13	406	03140101106	13
30	修毛刺弯曲	2	2011-01-03 10:48:50	2011-01-03 11:09:13	406	03140101024	13
30	修毛刺弯曲	3	2011-01-03 10:48:50	2011-01-03 11:10:47	406	03140101025	14
35	相差	1	2011-01-03 11:10:47	2011-01-03 11:12:40	406		40

1 - 15 共 17 条

[输出excel数据](#) [返回上一级](#)

图 4.50 工序排产结果

查看车间详情页面：

零件基本信息 零件图号: CS120.08.198-2 零件名称: 通气器体 原计划年月: 2011-01 车型: CS212

车间计划详情(默认按照工艺路线排序)

车型	车间/次数	图号	流入数量	流出数量
CS212	406/1	CS120.08.198-2	40	40

1-1 共 1条

输出excel数据 返回上一级

图 4.51 车间详情查询结果

以设备形式查询排产结果页面：

分厂月排产结果(默认按设备编号排序)

设备编号	设备名称	设备型号	设备利用率	设备车间	设备详情
03110501002	六角车床	1325	16.0%	413	设备详情
03110501005	六角车床	1325	16.1%	413	设备详情
03110501008	六角车床	1K37	35.4%	403	设备详情
03110501016	六角车床	1K37	31.0%	402	设备详情
03110501017	六角车床	1K37	35.8%	402	设备详情
03110501021	六角车床	1K37	50.7%	405	设备详情
03110501024	六角车床	1K36	76.8%	406	设备详情
03110501025	六角车床	1K36	70.4%	406	设备详情
03110501026	六角车床	1K37	32.8%	402	设备详情
03110501030	六角车床	C365L	33.9%	402	设备详情
03110501032	六角车床	C365L	35.4%	403	设备详情
03110501047	六角车床	C365L	0.8%	413	设备详情
03110501055	六角车床	C365L	75.8%	406	设备详情
03110501057	六角车床	C365L	71.7%	406	设备详情
03110501061	六角车床	C365L	77.3%	406	设备详情

1-15 共 264条

输出excel数据

图 4.52 设备形式查询结果

查看设备详情页面：

设备基本信息 设备编号: 03110501002 设备名称: 六角车床 设备型号: 1325 车间: 413

设备调度详情(默认按工序开工时间排序)

设备编号	图号	工序号	工序名称	批次	开工时间	完工时间
03110501002	CS212.40.273	5	车外圆	1	2011-01-03 00:00:00	2011-01-03 17:07:38
03110501002	CS212.64.011	5	切断	1	2011-01-05 14:00:00	2011-01-05 14:18:01
03110501002	CS212.64.011	25	去工艺余头	1	2011-01-05 15:53:05	2011-01-05 16:29:13

1-3 共 3条

输出excel数据 返回上一级

图 4.53 设备详情查询结果

(2) 甘特图查询

甘特图查询页面:



图 4.54 甘特图查询

以零件形式查询排产结果页面：



图 4.55 甘特图查看零件排产结果

可以放大查看：

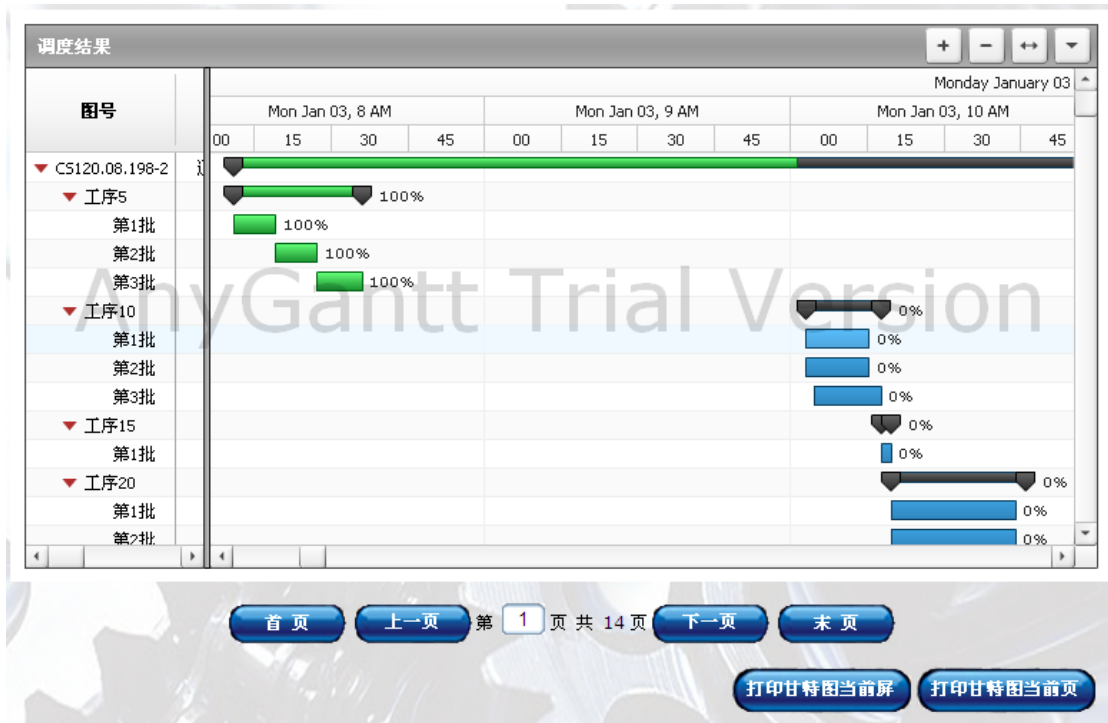


图 4.56 甘特图放大效果图

以设备形式查询排产结果页面：



图 4.57 设备形式查询甘特图

同样也可以放大查看：

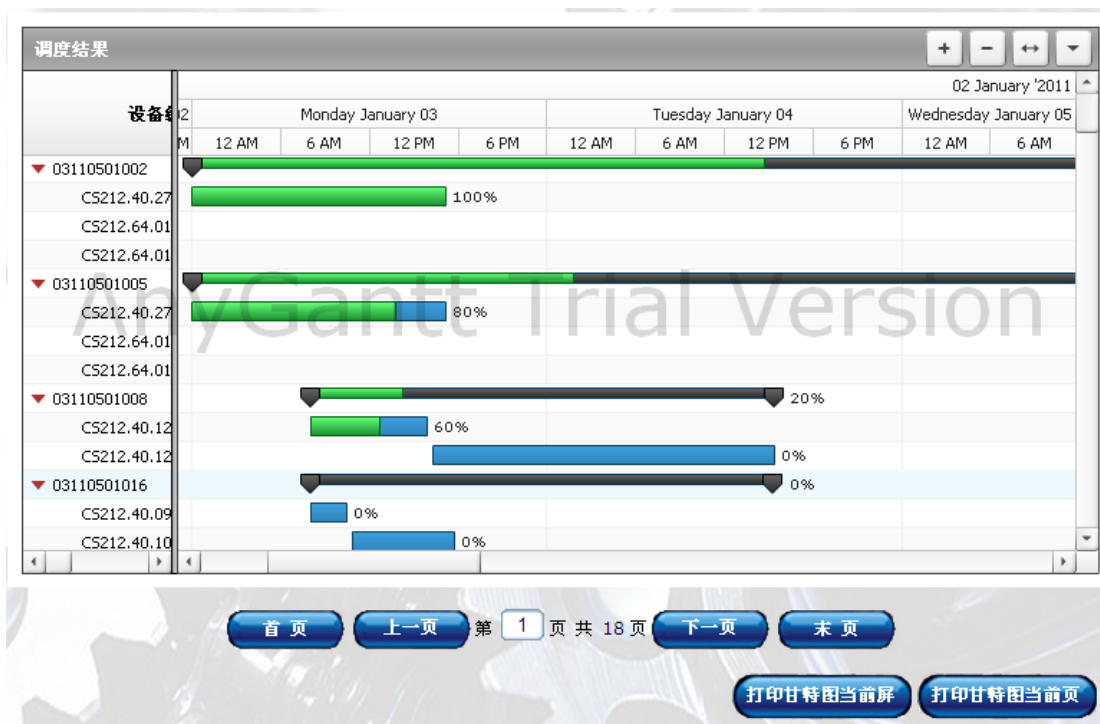


图 4.58 设备形式查询放大图

(3) 窄口查询

窄口查询页面：



图 4.59 窄口查询

以工序的形式查询排产结果页面：

(4) 车间计划查询

车间计划查询页面：



图 4.60 车间计划查询页面

点击查询后，显示查询记录：

车间计划(默认按零件图号排序)					
零件名称	图号	车型	车间/次数	流入数量	流出数量
通气器体	CS120.08.198-2	CS212	406/1	40	40
螺栓	CS120.08.238	CS212	413/1	40	40
传动箱	CS212.40.00	CS212	409/1	20	20
端盖	CS212.40.002	CS212	406/1	20	20
端盖	CS212.40.002	CS212	404/1	20	20
端盖	CS212.40.002	CS212	406/2	20	20
轴承座	CS212.40.003	CS212	406/1	20	20
套筒	CS212.40.004	CS212	406/1	20	20
套筒	CS212.40.004	CS212	404/1	20	20
套筒	CS212.40.004	CS212	406/2	20	20
套筒	CS212.40.005	CS212	406/1	20	20
套筒	CS212.40.005	CS212	404/1	20	20
套筒	CS212.40.005	CS212	406/2	20	20
齿轮Z28	CS212.40.006	CS212	402/1	40	40
齿轮Z28	CS212.40.006	CS212	404/1	40	40

1 共 36页 1 - 15 共 531条

输出excel数据

图 4.61 车间计划查询结果

(ii) 分厂月计划完成情况

本模块主要包括两项功能：查看新追加计划和查看剩余计划，生产管理人员可以查看某月新追加计划和剩余计划的详细信息。其中具有分厂级权限的生产管理人员具有所有权限，而车间级生产管理人员只能查看本车间的月计划完成情况。

1、查看新追加计划

为生产管理人员提供多种组合查询条件，生产管理人员可以根据自己的需求选择性的输入查询条件后，点击查询，在本页面列出所有满足查询条件的记录，记录信息包括：零件名称、图号、车型、参考数量、保障数量、流入数量、流出数量、完工数量、车间、进入车间的次数以及该零件的详细信息；其中查询条件包括：**a.**直接查询，即生产管理人员不用任何输入，直接点击查询，此时列出该生产月中所有的零件信息；**b.**在零件图号，零件名称和车型三者中，可以任意选择性的输入一个，两个或者三个（输入条件支持模糊查询、忽略字母大小写），点击查询，进行联合查询，此时列出满足查询条件的所有记录；

零件信息默认按照车型和图号依次排序，同时为生产管理人员提供按零件名称、图号、车型、参考数量、保障数量、流入数量、流出数量、完工数量的升序及降序排序功能；提供将查询结果导出为Excel表格的功能；

当生产管理人员查询完后，在月计划完成情况页面列出的记录中，详细信息以

超链接的形式显示，当生产管理人员点击某条记录的详细信息时，切换一个新的页面，显示该零件的具体信息，包括图号、零件名称、车型、加工路线、累计入库数量、库存数量、毛坯累计数量和合同号；提供将查看结果导出为Excel表格的功能。

2、查看剩余计划

为生产管理人员提供多种组合查询条件，生产管理人员可以根据自己的需求选择性的输入查询条件后，点击查询，在本页面列出所有满足查询条件的记录，记录信息包括：零件名称、图号、车型、原计划年月、剩余数量、完工数量、车间以及该零件的详细信息；其中查询条件包括：a.直接查询，即生产管理人员不用任何输入，直接点击查询，此时列出该生产月中所有的零件信息；b.在零件图号，零件名称和车型三者中，可以任意选择性的输入一个，两个或者三个（输入条件支持模糊查询、忽略字母大小写），点击查询，进行联合查询，此时列出满足查询条件的所有记录；

零件信息默认按照原计划年月、车型和图号依次排序，同时为生产管理人员提供按零件名称、图号、车型、原计划年月、剩余数量、完工数量、车间的升序及降序排序功能；提供将查询结果导出为Excel表格的功能；

当生产管理人员查询完后，在月计划完成情况页面列出的记录中，详细信息以超链接的形式显示，当生产管理人员点击某条记录的详细信息时，切换一个新的页面，显示该零件的具体信息，包括图号、零件名称、车型、加工路线、累计入库数量、库存数量、毛坯累计数量和合同号；提供将查看结果导出为Excel表格的功能。

上述功能的具体操作流程如下：

输入查询条件后，月计划完成情况页面为：

月计划完成情况(默认按车型和图号依次排序)									
零件名称	图号	车型	参考数量	保障数量	流入数量	流出数量	完工数量	车间/次数	详细信息
通气器体	CS120.08.198-2	CS212	40	20	40	40	0	406/1	详细信息
螺栓	CS120.08.238	CS212	40	20	40	40	0	413/1	详细信息
传动箱	CS212.40.00	CS212	20	10	20	20	0	409/1	详细信息
端盖	CS212.40.002	CS212	20	10	20	20	0	404/1	详细信息
端盖	CS212.40.002	CS212	20	10	20	20	0	406/1	详细信息
端盖	CS212.40.002	CS212	20	10	20	20	0	406/2	详细信息
轴承座	CS212.40.003	CS212	20	10	20	20	0	406/1	详细信息
套筒	CS212.40.004	CS212	20	10	20	20	0	404/1	详细信息
套筒	CS212.40.004	CS212	20	10	20	20	0	406/1	详细信息
套筒	CS212.40.004	CS212	20	10	20	20	0	406/2	详细信息
套筒	CS212.40.005	CS212	20	10	20	20	0	404/1	详细信息
套筒	CS212.40.005	CS212	20	10	20	20	0	406/1	详细信息
套筒	CS212.40.005	CS212	20	10	20	20	0	406/2	详细信息
齿轮Z28	CS212.40.006	CS212	40	20	40	40	0	402/1	详细信息
齿轮Z28	CS212.40.006	CS212	40	20	40	40	0	402/2	详细信息

1 共 36页 1-15 共 531条

[输出excel数据](#)

图 4.62 月计划完成情况页面

详细信息页面为:

零件详细信息							
图号	零件名称	车型	加工路线	累计入库数量	库存数量	毛坯累计数量	合同号
CS120.03.356	垫圈	CS123	GPX-413-6(0)-4K	0	0	0	2011-002

1 共 1页 1-1 共 1条

[输出excel数据](#) [返回上一级](#)

图 4.63 详细信息页面

查看剩余计划的页面:



图 4.64 查看剩余计划页面

输入查询条件后，剩余计划情况页面为：

零件名称	图号	车型	原计划年月	剩余数量	完工数量	车间	详细信息
螺塞座	CS120.03.003-2	CS122R	2011-01	180	0	405	详细信息
接合盘	CS120.03.082-1	CS122R	2011-01	11	0	405	详细信息
排除空气开关	CS120.05.0126	CS122R	2011-01	10	0	409	详细信息
开[排除空气开关]	CS120.05.0128	CS122R	2011-01	10	0	409	详细信息
垫圈	CS120.05.027-1	CS122R	2011-01	20	0	413	详细信息
螺塞	CS120.05.038-A	CS122R	2011-01	20	0	413	详细信息
开关体	CS120.05.398	CS122R	2011-01	10	0	406	详细信息
杆	CS120.05.400	CS122R	2011-01	10	0	413	详细信息
大制动鼓密封盖	CS120.09.004-1C	CS122R	2011-01	20	0	405	详细信息
左分离环	CS120.09.007-2	CS122R	2011-01	10	0	406	详细信息
右分离环	CS120.09.008-2	CS122R	2011-01	10	0	406	详细信息
支撑套	CS120.09.017-1C	CS122R	2011-01	20	0	406	详细信息
固定板	CS120.09.022	CS122R	2011-01	80	0	406	详细信息
密封环	CS120.09.024-1A	CS122R	2011-01	60	0	406	详细信息
密封环	CS120.09.025-2A	CS122R	2011-01	60	0	406	详细信息

图 4.65 剩余计划结果页面

详细信息页面为：

零件详细信息							
图号	零件名称	车型	加工路线	累计入库数量	库存数量	毛坯累计数量	合同号
CS120.05.038-A	螺塞	CS123		0	0	0	2011-002

1-1 共 1条

输出excel数据 返回上一级

图 4.66 详细信息页面

4.4.4 生产管理

(i) 生产日历查看

生产日历为生产管理人员提供查看生产规则的页面，可以按日历和表格两种形式查看，其中具有分厂级权限的生产管理人员具有所有权限，而车间级生产管理人员只能查看本车间的规则。

如果按照日历查看，生产管理人员可以根据自己的需求选择查询条件，包括车间、设备、时间，其中机器为必选项，选择按日历查看，点击查询生产日历，该页面下方的日历显示生产管理人员查询的生产规则；生产管理人员可以按月、按周、按天详细查看规则的详细情况

如果按照表格查看，生产管理人员可以根据自己的需求选择查询条件，包括车间、设备、时间，其中机器为必选项，选择按表格查看，点击查询生产日历，该页面下方的表格显示生产管理人员查询的对应规则，包括时间、规则名和内容。

具体操作流程如下：

点击“选择设备”，弹出设备选择窗口：



图 4.67 设备查询页面

选择全部设备添加到列表，如果是车间级权限登录的生产管理人员，选择设备的窗口会显示该车间的所有设备名称，如下图：



图 4.68 设备选择页面

选择按日历查看，点击“查询生产日历”，日历按月显示如下：

图 4.69 日历形式查看生产日历页面

选择按表格查看，点击“查询生产日历”，日历按月显示如下：

时间	规则名	内容
2010-12-01	all-admin-20111129.1945-规则	休息 08:00-12:00
2010-12-02	all-admin-20111129.1945-规则	休息 08:00-12:00
2010-12-03	all-admin-20111129.1945-规则	休息 08:00-12:00
2010-12-04	all-admin-20111129.1945-规则	休息 08:00-12:00
2010-12-05	all-admin-20111129.1945-规则	休息 08:00-12:00
2010-12-06	all-admin-20111129.1945-规则	休息 08:00-12:00
2010-12-07	all-admin-20111129.1945-规则	休息 08:00-12:00
2010-12-08	all-admin-20111129.1945-规则	休息 08:00-12:00
2010-12-09	all-admin-20111129.1945-规则	休息 08:00-12:00
2010-12-10	all-admin-20111129.1945-规则	休息 08:00-12:00
2010-12-11	all-admin-20111129.1945-规则	休息 08:00-12:00
2010-12-12	all-admin-20111129.1945-规则	休息 08:00-12:00
2010-12-13	all-admin-20111129.1945-规则	休息 08:00-12:00
2010-12-14	all-admin-20111129.1945-规则	休息 08:00-12:00
2010-12-15	all-admin-20111129.1945-规则	休息 08:00-12:00

1.未显示时间均按照默认生产时间设置

图 4.70 表格形式查看生产日历页面

(ii) 规则设置

规则设置为生产管理人员提供个性化设定生产规则的页面，生产管理人员可以根据生产实际情况，设置某个时间段工作或者休息，某台设备工作或者休息，主要

功能包括新建规则、删除规则和查看工厂日历，其中具有分厂级权限的生产管理人员具有设置所有设备规则的权限，而车间级生产管理人员只能设置本车间内的设备的规则。

新规则的创建步骤一共三步，第一步选择物理范围，即选择车间的设备，范围小到一个车间的一台设备，大到所有车间的所有设备；第二步选择时间，为生产管理人员提供按周添加、按日添加和按种类添加三种方式，生产管理人员可以选择至少一条记录添加到列表中；第三步创建新的日模式内容，设置时间段的范围、内容性质为工作或者休息，备注为生产管理人员选填项，点击完成，返回规则设置页面，新建的规则添加到列表中。

具体操作流程如下：



图 4.71 规则页面

点击“新建规则”，进入创建规则第一步：



图 4.72 新建规则页面

选择车间和设备，添加到工厂日历规则设置列表中，例如选择402车间、03110501016设备：



图 4.73 选择物理范围页面

点击下一步，进入创建规则第二步：

当前步骤 第二步, 创建一个日集合

选择年: 2011 年

按周添加 全部月 月 全部周 周 全部日 日

按日添加 全部月 月 全部日 日

按种类添加 全部月 月 全部种类

添加到列表中 删除所选记录

日范围表			
方式	月	周	日

0 共 0 页 无数据显示

上一步 下一步

图 4.74 创建日集合页面

可以按周、按日、按种类添加规则，例如选择2011年按日添加12月5日，点击“添加到列表中”：

当前步骤: 第二步, 创建一个日集合

选择年: 2011 年

按周添加 12月 月 第6周 周 全部日 日

按日添加 12月 月 5号 日

按种类添加 全部月 月 全部种类

日范围表			
方式	月	周	日
按日期	12		5

1 共 1页 1-1 共 1条

图 4.75 添加日范围页面

点击下一步，进入创建规则第三步：

当前步骤：第三步，创建新的日模式内容

*时间段： 点到 点

*内容性质：

备注：

1.未设置时间默认休息，即只需要设置工作时间即可

小时集合表			
内容性质	起始时间(点)	结束时间(点)	备注

0 共 0页 无数据显示

图 4.76 创建日内容页面

设置时间段为8点到12点，休息，备注填写为设备损坏，添加到列表中：

当前步骤：第三步，创建新的日模式内容

*时间段：08 点到 12 点

*内容性质：休息

备注：设备损坏

添加到列表中 删除所选记录

1.未设置时间默认休息，即只需要设置工作时间即可

内容性质	起始时间(点)	结束时间(点)	备注
休息	08	12	设备损坏

1-1 共 1条

上一步 完成

图 4.77 设置日内容页面

点击完成，新规则创建成功。



图 4.78 规则设置完成页面

点击弹出提示框中的确定, 自动返回规则设置页面, 新创建的规则已经添加到列表中:



图 4.79 查看新添加规则页面

选择这条新添加的规则：



图 4.80 删除规则页面

点击删除规则，提示已删除：



图 4.81 删除规则成功页面

点击“查看工厂日历”，跳转到“生产日历查看”页面。

(iii) 生产设置

生产设置为生产管理人员提供机器设置、班次设置和工作日设置的功能。生产管理人员可以根据生产实际情况，调整设备的班次、班次时间区间以及工作日范围。主要功能包括机器设置、班次设置和工作日设置，其中具有分厂级权限的生产管理人员具有修改班次和工作日的权限，而车间级生产管理人员只能修改本车间内的班次，查看工作日范围的权限。

1、机器设置

为生产管理人员提供调整设备班次的页面，提供多种组合查询条件，包括车间、设备编号、设备型号和设备名称；其中设备查询条件包括车间、设备编号、设备型号和设备名称，可以任意选择性的输入一个、两个或者三个（输入条件支持模糊查询、忽略字母大小写），点击查询，进行联合查询，此时列出满足查询条件的所有记录；记录包括：设备编号、设备名称、设备型号、设备车间、班次和确定修改班次的按钮；当生产管理人员选中某行记录时，班次变化为下拉框，可以调整该记录的班次，点击“确定修改”，保存修改；

查询的设备信息默认按照设备编号升序排序，同时为生产管理人员提供按设备编号、设备名称、设备型号、设备车间和班次的升序及降序排序功能；提供将查询结果导出为Excel表格的功能；

2、班次设置

为生产管理人员提供调整班次时间的页面，班次分为单班、双班和三班，单班包括2个时间区间，双班包括3个时间区间，三班包括1个时间区间，生产管理人员可以根据实际需求，修改各个时间区间的起止时间；点击“重置为默认”，恢复最近一次保存结果（即数据库中的班次设置情况）；点击“保存设置结果”，保存当前对班次的修改情况。

3、工作日设置

工作日设置为生产管理人员提供了查看和修改工作日范围的页面，生产管理人员可以从下拉框中选择工作日范围，点击保存，保存当前对工作日的设置。

生产日历管理中各功能具体操作如下：

机器设置页面：



图 4.82 机器设置页面

输入查询条件，点击查询，页面下方显示查询记录：

设备信息(默认按设备编号排序)					
设备编号	设备名称	设备型号	设备车间	班次	确定修改班次
03110401001	仿形车床	CF6140	405	1	确定修改
03110501002	六角车床	1325	413	2	确定修改
03110501005	六角车床	1325	413	2	确定修改
03110501008	六角车床	1K37	403	1	确定修改
03110501016	六角车床	1K37	402	1	确定修改
03110501017	六角车床	1K37	402	1	确定修改
03110501021	六角车床	1K37	405	1	确定修改
03110501024	六角车床	1K36	406	1	确定修改
03110501025	六角车床	1K36	406	1	确定修改
03110501026	六角车床	1K37	402	1	确定修改
03110501030	六角车床	C365L	402	1	确定修改
03110501032	六角车床	C365L	403	1	确定修改
03110501047	六角车床	C365L	413	1	确定修改
03110501055	六角车床	C365L	406	1	确定修改
03110501057	六角车床	C365L	406	1	确定修改

1 共 56页 1-15 共 829条

输出excel数据

图 4.83 查询机器页面

选中某行记录，修改班次：

设备信息(默认按设备编号排序)					
设备编号	设备名称	设备型号	设备车间	班次	确定修改班次
03110401001	仿形车床	CF6140	405	1	确定修改
03110501002	六角车床	1325	413	2	确定修改
03110501005	六角车床	1325	413	2	确定修改
03110501008	六角车床	1K37	403	1	确定修改
03110501016	六角车床	1K37	402	2	确定修改
03110501017	六角车床	1K37	402	1	确定修改
03110501021	六角车床	1K37	405	1	确定修改
03110501024	六角车床	1K36	406	1	确定修改
03110501025	六角车床	1K36	406	1	确定修改
03110501026	六角车床	1K37	402	1	确定修改
03110501030	六角车床	C365L	402	1	确定修改
03110501032	六角车床	C365L	403	1	确定修改
03110501047	六角车床	C365L	413	1	确定修改
03110501055	六角车床	C365L	406	1	确定修改
03110501057	六角车床	C365L	406	1	确定修改

1 共 56页 1-15 共 829条

输出excel数据

图 4.84 修改班次页面

点击确定修改，保存修改记录：



图 4.85 修改班次成功页面

班次设置页面：



图 4.86 班次设置页面

调整时间区间，例如将单班时间区间1修改为从08时30分至11时30分，点击“保存设置结果”：



图 4.87 班次设置成功页面

工作日设置页面：



图 4.88 工作日设置页面

修改工作日范围，例如从周一到周六，点击保存：



图 4.89 工作日设置成功页面

4.5 实际生产应用

在生产组织过程中，生产管理人员经常要决策各产品以怎样的比例进行投产，或按照怎样的策略划分批次，才能获得更高的生产效率或更短的生产周期。该项功能的作用是为生产管理人员的上述决策提供支持，具体分为单线生产决策建议和混线生产决策建议：

4.5.1 单线生产

(1) 各车型的分批策略

实现了单线生产环境下，各车型的最优批次划分策略（其中一种车型的数据分析案例如图4.90所示）。采用单线生产某车型，分析不同临界值下不同批大小所需调度时间以及设备切换次数。从图中分析得到，增加分批数可以缩短完工时间，但同时也增加了设备切换的次数。

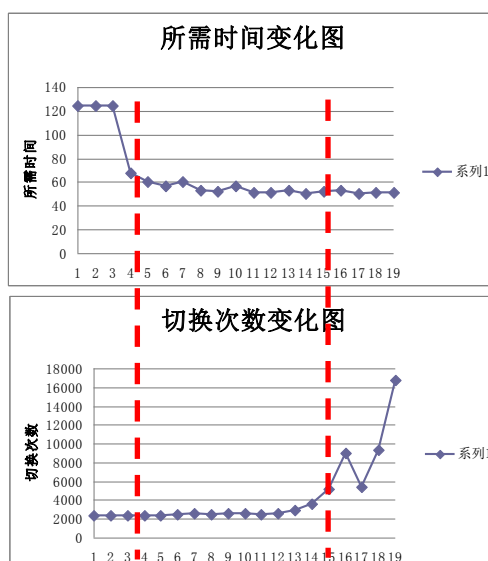


图 4.90 某车型分批策略案例数据

(2) 各车型的特殊零件

在各车型单线生产条件下，分析对最大完工时间这一性能指标影响较显著的零件，将它们称为特殊零件。由于特殊零件的加工成为提升整个车型生产时间的瓶颈，因此我们分别将它们特殊处理，如增大分批数、提升工时压缩比例。经过特殊处理后，系统运行时间与去掉特殊零件后系统的运行时间相当。各车型的特殊零件和特殊处理结果见表4.1。

表 4.1 各车型的特殊零件和特殊处理结果

序号	车型	特殊零件	特殊处理	特殊处理后的时间	去掉特殊零件后的时间
1	A	KD442T.015 履带销	5,10,15,20,30工序分5批	60	60
		KD120T.063 辐板	15,20工序压缩到70% 所有工序分3批		
		KD120T.061 耐磨圈	5,10,20,30工序压缩到70% 所有工序分3批		
2	B	KD442T.015 履带销	5,10,15,20,30工序分5批	92	94
		JX0431.003 右板体	20工序压缩到70%，分3批		
		JX0431.001 左板体	20工序压缩到70%，分3批		
		JX0292.020 压板	40工序压缩到70%，分3批		
3	C	JX0293.020 压板	40工序压缩70%，分3批	30 (15台)	26
		CS0001.93.006 轮毂	工时整体压缩到50% 5,10,50,110工序分3批 15,20工序分5批		
		CS0001.93F2.001-1 铝合金轮毂	工时整体压缩到50% 5,10,15,70,95工序分3批		
		CS0001.93.007-1 衬套	5,25,70,90工序分3批		
4	D	CS0001.93.008 衬套	5,10,15,20工序分3批	38	36
		CS122.81.031 偏心柄	5,10,15,20工序分3批		
5	E	CS122R.08.08 复合框架	10工序压缩到60%	26 (20台)	23
		CS212.40.159 螺纹钢套M14*1.5-6H	5,20工序分3批		
		CS212.40.156 螺纹钢套M10*1-6H	5,20工序分5批		
		CS212.40.128 连接盘	5,10,40,65,105工序压缩到70% 35,80工序压缩90%		
		CS212.64.044 锥齿轮轴2	55工序压缩到70%		

(3) 特殊车型

在各车型单线生产条件下，分析哪些车型的生产可能成为整个生产调度的瓶颈。采用排产软件运行5个车型的零件加工数据，得到各车型的加工时间结果如表4.2所示。从表中分析，车型C、车型D和车型E的运行天数基本相当，而车型A和车型B的运行天数约是其他三个车型的两倍，因此我们认为车型A和车型B是特殊车型，在制定生产计划时需要优先考虑特殊车型的生产任务。

表 4.2 各车型的平均加工时间和特殊车型判断结果

车型	平均天数	是否特殊车型
A	158.7	★
B	106.3	★
C	54.7	
D	51.3	
E	32.5	

4.5.2 混线生产

(1) 双车型混线生产环境下投产拐点

实现了各产品双车型混线多种生产任务与生产环境下，预测不同投产比例下起主导作用的拐点。我们以特殊车型A和B为例，分别对与其他车型投产的比例进行详细测试，结果如表4.3和表4.4所示。

表 4.3 车型 A 与其他车型相混生产时拐点测试

10/20台份A车型与其他车型相混拐点测试								
保障数/ 参考数	10/20	15/30	20/40	30/60	40/80	50/100	60/120	70/140
B	A	B	B	B	B	B	B	B
C	A	A	A	C	C	C	C	C
D	A	A	A	A	A	D	D	D
E	A	A	A	A	A	A	E	E

表 4.4 车型 B 与其他车型相混生产时拐点测试

10/20台份B车型与其他车型相混拐点测试								
保障数/ 参考数	10/20	15/30	20/40	25/50	30/60	40/80	50/100	60/120
A	B	A	A	A	A	A	A	A
C	B	C	C	C	C	C	C	C
D	B	B	B	B	D	D	D	D
E	B	B	B	B	E	E	E	E

(2) 多种车型混线生产环境下最优投产比例策略

实现了各产品单线/二车型混线/三车型混线/四车型混线/五车型混线等多种生产

任务与生产环境下，各车型的最优投产比例策略分析，指导生产趋近某些“优质路线”，避开某些“劣质路线”。如图4.91所示，以双车型混线生产为例，当确定车型A的生产计划后，通过调整同时投放的车型B的生产计划，发现生产台份数存在一个拐点，此时两种车型对完工时间的影响相当，此时确定了混线生产车型A和车型B的合理投放比例。图中第一层表示单线生产车型A所需调度时间，第二层表示车型A与其他车型双混线生产所需的调度时间，以此类推。图中绿箭头表示应趋近的路线，红箭头表示应避免的路线。

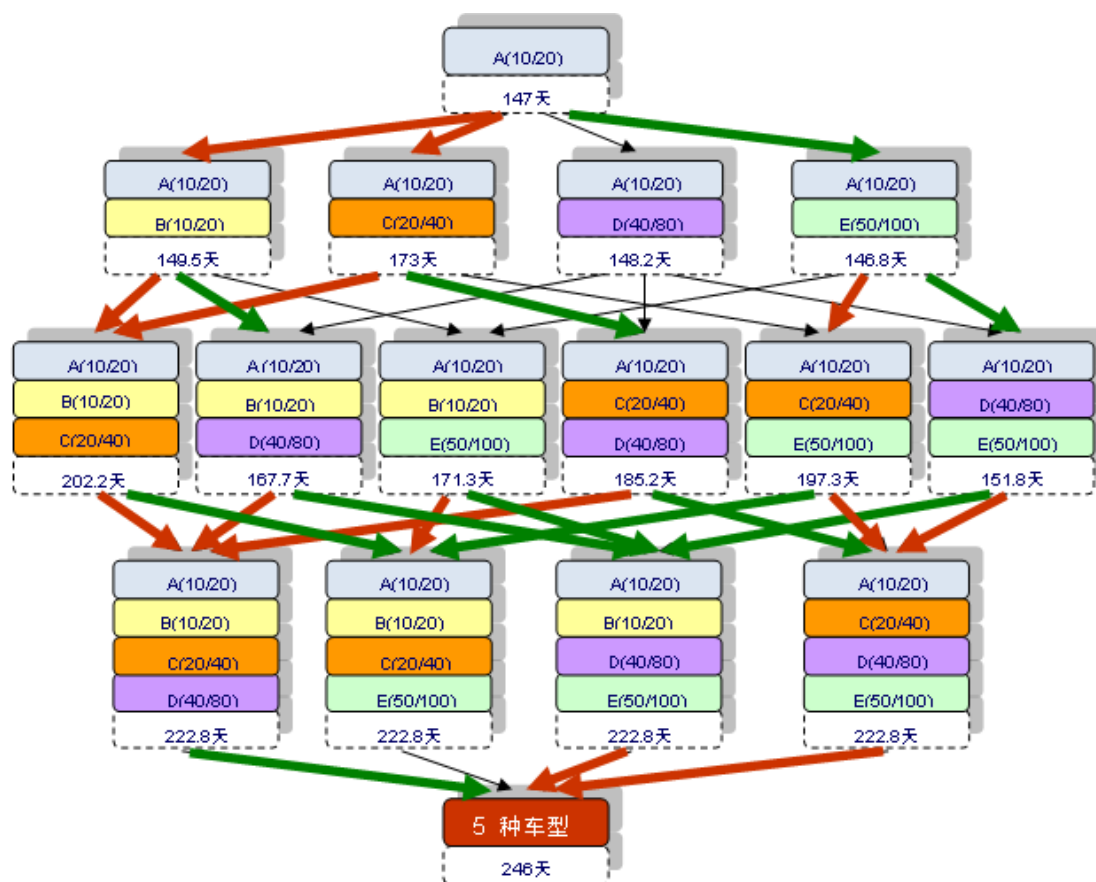


图 4.91 单线及各种混线生产条件下不同生产路径的完工时间

4.6 本章小结

本章主要针对智能优化排产软件，从软件功能、系统结构、模块详细设计和使用说明等角度详细阐述该软件的设计细节。针对兵器一机集团军工产品的复杂生产过程、生产环境和工艺特征，以解决企业生产计划和调度问题为目标，研制开发了智能排产软件。作品目前已在兵器一机集团成功应用，任务完成量增加18%，设备利用率提高了20%，在制品库存下降28%，生产周期缩短35%，全员劳动生产率提高

20%，为企业节约成本总计1000余万元，新增产值总计7000余万元。

排产软件使用方便灵活、性能稳定、安全性强，很好的支持了企业综合传动装置的生产管理决策。

另外，作品今年5月起在北奔集团投入试运行。

结论

作品面向军工装备制造业在生产管理中遇到的实际问题，针对多品种、变批量、产研结合、混线生产的独特生产模式，设计并开发了智能优化排产软件，解决复杂产品生产中的跨车间计划调度问题。作品的核心和难点在于设计了新颖的兼具较高计算效率和较好优化能力的跨车间调度算法，突破了制约相关理论成果投入实际应用的重要瓶颈。作品不仅在理论上进一步扩充和丰富了跨车间计划与调度理论和技术，并且对解决我国军工装备制造业复杂零部件排产决策中的难点问题提供支持。

目前作品已在兵器一机集团成功应用，应用后任务完成量增加 18%，设备利用率提高了 20%，在制品库存下降 28%，生产周期缩短 35%，全员劳动生产率提高 20%，为企业节约成本总计 1000 余万元，新增产值总计 7000 余万元。

另外，作品今年 5 月起已在北奔集团投入试运行。

作品可推广至军用装备、航天航空、海洋工程、轨道交通等行业生产管理中，具有广阔的应用前景。

参考文献

- [1] Garza O, Smunt T L. Countering the negative impact of intercell flow in cellular manufacturing[J]. *Journal of Operations Management*, 1991, 10(1): 92-118.
- [2] Yang W H, Liao C J. Group scheduling on two cells with intercell movement[J]. *Computers & Operations Research*, 1996, 23(10): 997-1006.
- [3] Solimanpur M, Vrat P, Shankar R. A heuristic to minimize makespan of cell scheduling problem[J]. *International journal of production economics*, 2004, 88(3): 231-241.
- [4] Tavakkoli-Moghaddam R, Javadian N, Khorrami A, et al. Design of a scatter search method for a novel multi-criteria group scheduling problem in a cellular manufacturing system[J]. *Expert Systems with Applications*, 2010, 37(3): 2661-2669.
- [5] Gholipour-Kanani Y, Tavakkoli-Moghaddam R, Khorrami A. Solving a multi-criteria group scheduling problem for a cellular manufacturing system by scatter search[J]. *Journal of the Chinese Institute of Industrial Engineers*, 2011, 28(3): 192-205.
- [6] Wang X, Tang J, Gong J. Multi-objective dynamic in cell formation approach based on scatter search. *Journal of Management Sciences in China*. 2009,12 (5):44-52.
- [7] Tang J, Wang X, Kaku I, et al. Optimization of parts scheduling in multiple cells considering intercell move using scatter search approach[J]. *Journal of Intelligent Manufacturing*, 2010, 21(4): 525-537.
- [8] Elmi A, Solimanpur M, Topaloglu S, et al. A simulated annealing algorithm for the job shop cell scheduling problem with intercellular moves and reentrant parts[J]. *Computers & Industrial Engineering*, 2011, 61(1): 171-178.
- [9] Burke E K, Hyde M, Kendall G, et al. A classification of hyper-heuristic approaches[M]//*Handbook of Metaheuristics*. Springer US, 2010: 449-468.
- [10] Burke E K, Hyde M, Kendall G, et al. Hyper-heuristics: A survey of the state of the art[J]. *Journal of the Operational Research Society*, 2010.
- [11] Fukunaga A S. Automated discovery of local search heuristics for satisfiability testing[J]. *Evolutionary Computation*, 2008, 16(1): 31-61.
- [12] Fayad C, Petrovic S. A fuzzy genetic algorithm for real-world job shop scheduling[M]//*Innovations*

in applied artificial intelligence. Springer Berlin Heidelberg, 2005: 524-533.

[13] Ponnambalam S G, Ramkumar V, Jawahar N. A multiobjective genetic algorithm for job shop scheduling[J]. *Production planning & control*, 2001, 12(8): 764-774.

[14] Vázquez-Rodríguez J A, Petrovic S. A new dispatching rule based genetic algorithm for the multi-objective job shop problem[J]. *Journal of Heuristics*, 2010, 16(6): 771-793.

[15] Dorndorf U, Pesch E. Evolution based learning in a job shop scheduling environment[J]. *Computers & Operations Research*, 1995, 22(1): 25-40.

[16] Yang T, Kuo Y, Cho C. A genetic algorithms simulation approach for the multi-attribute combinatorial dispatching decision problem[J]. *European Journal of Operational Research*, 2007, 176(3): 1859-1873.

[17] Park S C, Raman N, Shaw M J. Adaptive scheduling in dynamic flexible manufacturing systems: a dynamic rule selection approach[J]. *Robotics and Automation, IEEE Transactions on*, 1997, 13(4): 486-502.

[18] LaForge R L, Barman S. The performance of simple priority rule combinations in a flow dominant shop[J]. *Production and Inventory Management Journal*, 1989, 30(3): 1-4.

[19] Sarper H, Henry M C. Combinatorial evaluation of six dispatching rules in a dynamic two-machine flow shop[J]. *Omega*, 1996, 24(1): 73-81.

[20] Barman S. Simple priority rule combinations: an approach to improve both flow time and tardiness[J]. *International Journal of Production Research*, 1997, 35(10): 2857-2870.

[21] Ruiz R, Vázquez-Rodríguez J A. The hybrid flow shop scheduling problem[J]. *European Journal of Operational Research*, 2010, 205(1): 1-18.

[22] Zandieh M, Mozaffari E, Gholami M. A robust genetic algorithm for scheduling realistic hybrid flexible flow line problems[J]. *Journal of Intelligent Manufacturing*, 2010, 21(6): 731-743.

[23] Luke S. *Essentials of Metaheuristics*. Lulu (2009)[J]. Available for free at <http://cs.gmu.edu/~sean/book/metaheuristics>, 2011.

[24] Ruiz R, Maroto C. A genetic algorithm for hybrid flowshops with sequence dependent setup times and machine eligibility[J]. *European Journal of Operational Research*, 2006, 169(3): 781-800.

[25] Koza J R. Genetic programming as a means for programming computers by natural selection[J]. *Statistics and Computing*, 1994, 4(2): 87-112.

[26] Montgomery D C. *Design and analysis of experiments*[M]. Wiley, 2008.

作品成果清单

获得国家发明专利授权 1 项：

[1] 一种柔性路径下跨作业单元调度方法[P]. 中国专利授权号：ZL201110276331.4.

申请国家发明专利 2 项：

[1] 跨车间协作模式下车间生产与调度一体化方法 [P]. 中国专利申请号：201110449696.2.

[2] 一种柔性路径下机加与装备交替混合调度方法 [P]. 中国专利申请号：201110316543.0.

发表论文 4 篇：

[1] Dynamic parts scheduling in multiple job shop cells considering intercell moves and flexible routes[J]. Computer & Operations Research, 2013, 40(5):1207-1223.

[2] 一种柔性路径下的跨单元调度方法[J]. 自动化学报. 2012, 38(6): 969-975.

[3] Aggregate scheduling of machining and assembly systems using bidirectional negotiation[C]. In Proceedings of 2012 IET International Conference on Information Science and Control Engineering. Shenzhen:The Institute of Engineering and Technology,2012,488.

[4] 基于双向协商机制的机加与装配混合调度，北京理工大学学报自然科学版，已录用.

致谢

两年前我们到 617 参加社会实践时，没想到今天我们的作品能参加全国挑战杯。当时我们对军工制造行业一无所知，只想看看我国的三代主战坦克是怎么造出来的。经过两年的努力我们完成了这个作品，这真令我们激动和自豪！

首先要感谢 617，特别是四分厂的居玉辉经理、计调室的赵凯师傅、技术室的杨忠林主任和卢忠师傅等。我们对生产管理的理解开始是很粗浅和幼稚的，但他们没有嘲笑我们异想天开的想法，而是给了我们大力的支持，甚至是宽容。今天排产软件在四分厂应用了，大家的努力终于有了回报，希望这个软件能给厂里的工作带来更多的便捷。

我们还要感谢清华大学的吴澄院士。我们和吴院士素不相识，但我们从一接触 CIMS 领域就知道吴院士的大名。这次得知我们的作品能参加全国挑战杯，我们就毛遂自荐的给吴院士发了一封邮件，邀请他做我们作品的推荐人。我们从来也没跟院士对话过，以为邮件发出后会石沉大海。没想到吴院士第二天就回了信。信中对我们的作品表示赞许，并且对我们提出了希望。尽管最后吴院士没有成为我们的推荐人，但通过跟吴院士的沟通，我们感到院士也像我们身边的老师一样，和蔼可亲、平易近人，我们同样受益匪浅。

我们还要感谢东北大学的唐加福教授和北京理工大学的魏一鸣教授，他们都是我们作品相关领域的著名学者，能邀请到他们做我们作品的推荐人我们感到很骄傲。特别是唐教授，在作品的设计过程中给过我们很多指导和建议，在此表示深深感谢！

我们还要感谢肖广雪学姐。小雪学姐一直是我们的一份子，尽管由于去年毕业而没能参加这届挑战杯，但我们的作品里也有她的功劳！遥祝在大洋彼岸求学的她一切顺利！

最后我们要互相感谢一下！通过这个作品我们真正知道了什么是团队。尽管有争论、甚至争吵，但是我们最终都因为共同的目标而选择站在一起。感谢每个人为作品付出的努力！感谢一起挥洒汗水的青春岁月！

附件清单

一、应用报告

(1) 兵器一机应用报告 1 份

(2) 北奔应用报告 1 份

二、查新报告

“基于超启发式的跨车间调度方法”查新报告一份

三、发明专利

(1) 专利授权证书 1 份

(2) 专利申请受理通知书 2 份

四、学术论文

发表学术论文 4 篇

五、推荐信

专家推荐信 2 份