

博士学位论文

表面组装过程中生产效率提升的关键 优化方法研究

THE KEY OPTIMIZATION METHOD FOR PRODUCTIVITY IMPROVEMENT IN SURFACE ASSEMBLY PROCESS

卢光宇

哈尔滨工业大学

2025 年 4 月

国内图书分类号: TP273
国际图书分类号: 681.5

学校代码: 10213
密级: 公开

博士学位论文

表面组装过程中生产效率提升的关键 优化方法研究

博士研究生: 卢光宇

导师: 高会军 教授

申请学位: 工学博士

学科: 控制科学与工程

所在单位: 航天学院

答辩日期: 2025 年 4 月

授予学位单位: 哈尔滨工业大学

Classified Index: TP273

U.D.C: 681.5

Dissertation for the Doctoral Degree

THE KEY OPTIMIZATION METHOD FOR PRODUCTIVITY IMPROVEMENT IN SURFACE ASSEMBLY PROCESS

Candidate:	Guangyu Lu
Supervisor:	Prof. Huijun Gao
Academic Degree Applied for:	Doctor of Engineering
Specialty:	Control Science and Engineering
Affiliation:	School of Aerospace
Date of Defence:	April, 2025
Degree-Conferring-Institution:	Harbin Institute of Technology

摘 要

表面组装过程优化是电子制造业赋能提效的关键技术，其中涉及的电路板元件组装过程的调度和规划，是综合了仓库选址、任务分配与路径规划的高维组合优化问题，具有多变量、多约束及高耦合性的复杂特性。本文所研究的装配有并列式贴片头的贴片机，因具备组装多种类型元件的能力而得到了广泛的应用。然而，现有研究并未充分考虑组装任务多样性、生产配置差异性及贴片头并列分布结构，导致其适用场景受限、组装效率有待提升。本文根据优化任务主要环节之间的关联性，将其分解为贴片头任务分配、贴装过程路径规划以及组装线负载平衡三个子问题，并分阶段地构建数学规划模型和设计具有普适性的高效表面组装过程优化方法。

首先，本文对组装过程的问题特性和任务分解方法进行了研究，分析了作业完整性、拾取过程一致性和工具适配性等约束条件，提出了基于关键子目标分解的贴片头任务分配模型，其子目标涵盖贴片头的拾贴元件动作、更换吸嘴动作以及组装过程往复运动等环节；在贴片头任务的约束下，进一步构建了多周期的贴装过程路径规划模型。此外，依据关键子目标加权效率评估、生产线中的可用工具和拾贴优先级等约束条件，本文建立了组装生产线的负载平衡模型。分阶段模型可用于求解小规模子问题的最优解，并为解决大规模问题的启发式算法设计提供了参考基准。

在任务分配阶段，本文分别从数学规划法和启发式算法的角度开展了研究，旨在通过优化关键效率指标提升贴片机拾取过程的效率，保证了组装过程整体解的高质量。通过构建基于周期任务集成的整数线性规划模型，实现了解空间的系统搜索。本文提出了递归启发式算法确定模型初始解，并通过限制决策变量的规模和缩小可行域的范围，提升模型的求解效率。同时，一种带有路径预判的启发式策略用于从解集池中筛选解，降低其与后续子问题之间的耦合性。为解决更大规模的问题，提出了基于分层启发式前瞻扫描的贴片头任务分配算法，拆解原问题为供料器-槽位分配和贴片头-元件分配问题，并设计了具有相似搜索架构的分层解决方案。本文从构造吸嘴分配模式出发，设计了最大化拾取过程效率的供料器分配启发式、以及对应的贴片头元件分配算法，综合了拾取可行性、工具可用性以及前瞻性收益等多个方面，平衡了拾取过程的长期与短期收益，适配了不同的生产配置。

在路径规划阶段,本文研究贴装过程的路径规划启发式方法,确保了不同贴片头任务分配约束下解的高质量,提升了贴装过程的效率。本文提出了适用于多轴作业的动态规划路径规划算法,保证了单个周期内解的最优性;为确定贴片头分配的贴装点,提出了多源贪心的动态导向集束搜索策略,将贪心准则应用于多作业区域并行搜索的过程中,增强了解的多样性。动态导向的集束搜索机制通过维持一组非最优的候选解和调整搜索方向,拓宽了搜索范围、提升了解的质量。进一步地,本文提出了基于自适应大邻域搜索的聚合路径重构算法,依据已有解的贴装点分布特征,自适应地调整路径结构,解决了贪心式搜索仅关注当前周期解局部最优的问题,用于在线迭代改进路径规划解。

在负载平衡阶段,本文构建了基于多特征融合集成组装时间估计器的超启发式元件分配算法,通过平衡多台贴片机的工作负载,提高了生产线的组装效率。其中,以集成学习为框架的估计器用于评估解的质量,融合了组装任务的基本参数以及启发式预估的子目标等多维特征,实现了组装时间的快速准确评估。贴片机-元件分配算法由数据驱动和目标驱动的底层算子构成,结合了组装优先级、可用工具等约束。算法在搜索过程中融合多种群策略以增加解的多样性、减少估计器误差对解的影响。此外,本文还结合聚合聚类的分组贴装点分配策略实现多机协同优化。

基于实际生产实例,本文分析了模型的参数灵敏度和复杂度,并在表面组装设备平台上围绕优化目标、运行效率等多项指标进行对比验证,证明了提出算法的有效性。

关键词: 表面组装过程优化; 贴片头任务分配; 多轴路径规划; 组装线负载平衡; 启发式优化; 数学规划法

Abstract

Surface assembly optimization is a key technique for empowering and improving efficiency in the electronics manufacturing industry, which involves the production scheduling and planning in the component assembly process of printed circuit boards. It is a high-dimensional combinatorial optimization problem integrating warehouse locating, task allocation, and path planning with the complex characteristics of multi-variable, multi-constraint, and high coupling. The surface mounter with linear-aligned heads is widely employed due to its ability to assemble different types of components. However, the existing studies have not fully considered complex factors such as the diversity of assembly tasks, the difference of production configuration, and the linear-aligned structure of the placement heads, resulting in limited application scenarios and efficiency to be improved. This thesis decomposes the optimization problems into three sub-ones, based on the correlation of the assembly process, namely, head task assignment, path planning of the placement process, and load balancing of the assembly line. The mathematical programming model and an efficient surface assembly process optimization method with general applicability is constructed in stages.

First, we study the characteristics of the assembly process problem, task decomposition methods, and the constraints of the model, such as work completeness, pickup process, and tool consistency constraints. A head task assignment model is proposed based on the decomposition of the key sub-objectives, which include the pick-and-place (PAP) operations, nozzle changing, and reciprocating movement of the placement heads. Moreover, this thesis further builds a multi-cycle path planning model for the placement process under the restrictions of head task assignment. This thesis develops a load balancing model for the assembly line using the weighted value of the key sub-objective as the evaluation metric and combining the machine constraints, assembly priority constraints, etc. The multi-stages models can get the optimal solution of small-scale sub-problems, which provides a reference for the design of heuristic algorithms for larger-scale problems.

In the task assignment stage, this thesis carries out research from the perspectives of mathematical programming and heuristic algorithms, which improve the pick-up process efficiency of the surface mounter by optimizing the key efficiency indicators of the assem-

bly process, so as to ensure the high quality of the overall solution. This thesis proposes an integer linear programming model based on the cyclic task integration to search the solution space systematically. A recursive heuristic algorithm is proposed to determine the initial solution and to improve the efficiency of the model solution by restricting the value range of the decision variables and narrowing the feasible domain. Meanwhile, a heuristic strategy with path estimation is applied to filter solutions from the pool to reduce the coupling with subsequent sub-problems. To solve larger-scale problems, this thesis further proposes a hierarchical heuristic algorithm based on look-forward scanning for head task assignment, which decomposes the problem into feeder-slot and head-component assignment sub-problems. To this end, two hierarchical solutions with a similar search architecture are designed for these sub-problems. This thesis designs a feeder allocation heuristic to maximize the efficiency of the pickup process from the construction of a nozzle allocation pattern, and a corresponding head-component allocation algorithm, which combines the feasibility of the pickup, the tool availability, and the prospective benefits, and balances the long- and short-term benefits of the pickup process, to achieve application in different production configurations.

In the path planning stage, this thesis investigates a placement path planning heuristic algorithm, which ensures the quality of solutions under different head task allocation constraints and thus improves the efficiency of the placement process of the surface mounter. A dynamic programming-based PAP cycle path planning for multi-axis assembly process algorithm is proposed, which ensures the optimality of the results in each cycle. This thesis proposes a multi-point greedy beam search path planning algorithm with dynamic orientation to determine the allocation of placement points. The strategy effectively enhances the diversity of solutions by searching multiple work regions in parallel and applies the greedy criterion to the placement point allocation for heads. The beam search mechanism broadens the search scope by maintaining a set of non-optimal candidate solutions and adjusting the search direction. Furthermore, an aggregation path reconstruction algorithm based on adaptive large neighborhood search is proposed, which adaptively adjusts the path structure according to the distributed characteristics of the existing solution. It solves the problem that greedy search only focuses on the local optimal solution of the current cycle and realizes the purpose of online improvement of result iteratively.

In the load balancing stage, this thesis proposes a hyper-heuristic component allocation algorithm based on a multifeature fusion ensemble assembly time estimator, which

can improve the assembly efficiency by balancing the workload of surface mounters. The assembly time estimator is framed by ensemble learning that integrates the basic parameters of the assembly process as well as the sub-objectives by the heuristic estimation to quickly and accurately evaluate the assembly time, etc. The component allocation algorithm for surface mounters is composed of data- and target-driven low-level operators under constraints such as assembly priority and available tools. In the search process, the algorithm combines multipopulation strategies to increase the diversity of solutions and reduce the influence of estimator errors on the results. In addition, this thesis also combines the aggregated grouping strategy for the placement points to achieve multiple machine co-optimization.

Based on the real-life production examples, this thesis analyses the parameter sensitivity and complexity of the model and conducts comparative experiments of the algorithm centered on the objectives, solving efficiency, etc., with a real surface assembly equipment as the platform, proving the effectiveness of the proposed algorithm.

Keywords: Surface assembly process optimization, placement head task allocation, multi-axis path planning, assembly line load balancing, heuristic optimization, mathematical programming

物理量名称及符号表

表 1 索引和集合的符号说明

符号	描述
$i \in I$	元件索引 i , 元件索引集合 $I = \{1, 2, \dots\}$
$j \in J$	吸嘴索引 j , 吸嘴索引集合 $J = \{1, 2, \dots\}$
$s \in S$	供料器基座槽位索引 s , 供料器基座槽位索引集合 $S = \{1, 2, \dots\}$
$h \in H$	贴片头索引 h , 贴片头索引集合 $H = \{1, 2, \dots\}$
$k \in K$	拾贴周期/周期组索引, 拾贴周期/周期组索引集合 $K = \{1, 2, \dots\}$
$m \in M$	贴片机索引 m , 贴片机索引集合 $M = \{1, 2, \dots\}$
$p \in P$	贴装点索引 p , 贴装点索引集合 $P = \{1, 2, \dots\}$
$a \in A$	移动弧索引 a , 弧索引集合 $A = \{(h, h') \mid h \neq h', h \in H, h' \in H\}$
$q \in Q$	组装顺序优先级索引 q , $q = (i, i') \in Q, i, i' \in I$, 表示元件 i 的拾贴任务需在元件 i' 开始前完成

表 2 决策变量的符号说明

符号	描述
g_{km}	$= 1$ 时表示贴片机 m 在拾贴周期 k 至少组装一个贴装点, 否则为 0
n_{khm}	$= 1$ 时表示贴片机 m 的贴片头 h 在拾贴周期/周期组 k 和 $k+1$ 之间发生吸嘴更换, 否则为 0
f_{ism}	$= 1$ 时表示贴片机 m 的槽位 s 分配的元件为 i , 否则为 0
e_{skm}	$= 1$ 时表示贴片机 m 在拾贴周期/周期组 k 拾取元件时, 悬臂的最左侧贴片头所对齐槽位为 s , 否则为 0
x_{ikhm}	$= 1$ 时表示贴片机 m 在拾贴周期/周期组 k 用贴片头 h 拾取元件 i , 否则为 0
y_{skhm}	$= 1$ 时表示贴片机 m 在拾贴周期/周期组 k 用贴片头 h 从槽位 s 拾取元件, 否则为 0
z_{jkhm}	$= 1$ 时表示贴片机 m 在拾贴周期/周期组 k 用贴片头 h 安装吸嘴 j , 否则为 0
$w_{pp'kam}$	$= 1$ 时表示贴片机 m 在拾贴周期 k 贴装点 p 后沿着弧 a 去贴装点 p' , 否则为 0
u_{pkhm}	$= 1$ 时表示贴片机 m 在拾贴周期 k 用贴片头 h 组装的首个贴装点为点 p , 否则为 0
v_{pkhm}	$= 1$ 时表示贴片机 m 在拾贴周期 k 用贴片头 h 组装的最后贴装点为点 p , 否则为 0
r_{im}	$= 1$ 时表示元件 i 被分配到贴片机 m , 否则为 0
d_{km}	表示贴片机 m 在拾贴周期 k 拾取元件过程中, 悬臂移动所经过的槽位数
b_{km}	表示贴片机 m 的拾贴周期组 k 中的拾贴周期数
c_{ikhm}	表示贴片机 m 在拾贴周期组 k 用贴片头 h 拾取的元件 i 的贴装点数

表 3 常数参数的符号说明

类别	符号	描述
时间参数	T^{CY}	表示贴片头往返于供料器基座和组装电路板之间的平均用时
	T^{NZ}	表示贴片头完成一次吸嘴更换的平均用时
	T^{PU}	表示贴片头完成一次拾取动作的平均用时
	T^{PL}	表示贴片头完成一次贴装动作的平均用时
	T^{PM}	表示贴片头移动过一个槽位的平均用时
位置参数	X_p, Y_p	表示贴装点 p 的 X 坐标和 Y 坐标
	$X^{\text{F1}}, Y^{\text{F1}}$	表示最左侧供料器槽位的 X 坐标和 Y 坐标
	F_{km}^{BW}	表示贴片机 m 在拾贴周期 k 的第一个取料槽位。
	F_{km}^{FW}	表示贴片机 m 在拾贴周期 k 的最后一个取料槽位
距离参数	D_{pkhm}^{FW}	表示贴片机 m 完成从供料器基座拾取元件后贴片头 h 到拾贴周期 k 的第一个贴装点 p 的移动距离
	$D_{pp'a}^{\text{PL}}$	表示贴片头在组装过程中沿着弧 a 从点 p 移动到点 p' 的移动距离
	D_{pkhm}^{BW}	表示贴片机 m 的贴片头 h 从拾贴周期 k 的最后一个贴装点 p 到供料器基座拾取元件的移动距离
	ρ	表示相邻贴片头的间隔距离
匹配参数	μ_{ij}	$= 1$ 时, 表示吸嘴 j 可完成元件 i 的拾贴任务, 否则为 0
	ξ_{im}	$= 1$ 时, 表示元件 i 可以被分配给贴片机 m , 否则为 0
	η_{ip}	$= 1$ 时, 表示贴装点 p 对应的元件为 i , 否则为 0
其他参数	ψ_i	表示元件 i 的贴装点数
	ϕ_i	表示元件 i 的可用供料器数量
	ζ_j	表示吸嘴 j 的可用数量
	τ	表示相邻贴片头和相邻供料器槽位之间的间隔距离之比
	N	表示一个足够大的数

目 录

摘 要	I
Abstract	III
物理量名称及符号表	VI
第 1 章 绪论	1
1.1 研究背景及意义	1
1.2 表面组装系统概述	3
1.3 表面组装过程优化的研究现状	6
1.3.1 贴片机的组装过程优化	6
1.3.2 生产线的负载平衡优化	15
1.4 现有研究存在的问题	19
1.5 本文主要研究内容及安排	22
第 2 章 表面组装过程优化的数学规划模型	26
2.1 引言	26
2.2 系统的描述与分析	27
2.2.1 系统构成	27
2.2.2 表面组装过程优化	29
2.3 多阶段模型的构建	30
2.3.1 分析与假设	30
2.3.2 贴片头的任务分配模型	32
2.3.3 贴装过程的路径规划模型	36
2.3.4 生产线的负载平衡模型	39
2.4 模型的验证与分析	40
2.4.1 实例演示	40
2.4.2 灵敏度分析	44
2.4.3 复杂度分析	46
2.5 本章小结	47
第 3 章 表面组装过程中贴片头的任务分配优化算法	48
3.1 引言	48
3.2 问题分析	49

3.3 基于周期任务集成模型的贴片头任务分配算法	49
3.3.1 周期任务集成模型	50
3.3.2 初始可行解的构建	51
3.3.3 模型的可解性增强策略	53
3.3.4 最优解的选择准则	55
3.4 基于分层前瞻扫描启发式的贴片头任务分配策略	56
3.4.1 算法概述	56
3.4.2 吸嘴的分配模式启发式算法	57
3.4.3 供料器的槽位分配启发式算法	58
3.4.4 贴片头的元件分配启发式算法	62
3.5 实验设计	65
3.5.1 基于模型的贴片头任务分配的对比实验	66
3.5.2 基于启发式的贴片头任务分配的对比实验	69
3.6 本章小结	75
第4章 表面组装过程中贴装过程的路径规划算法	76
4.1 引言	76
4.2 问题分析	77
4.3 贴装过程的路径规划方法	78
4.3.1 算法概述	78
4.3.2 基于动态规划的拾贴周期路径规划算法	79
4.3.3 基于多源贪心的动态导向集束搜索路径规划算法	80
4.3.4 基于自适应大邻域搜索的聚合路径重构算法	83
4.4 实验设计	86
4.4.1 基于启发式的贴装过程路径规划的对比实验	87
4.4.2 基于邻域搜索的贴装过程路径规划的对比实验	89
4.4.3 运行效率分析	93
4.5 本章小结	94
第5章 表面组装过程中生产线的负载平衡算法	95
5.1 引言	95
5.2 问题分析	96
5.3 算法框架	97
5.4 基于超启发式的表面组装生产线元件分配算法	98
5.4.1 底层启发式算子	98

5.4.2 可重复元件分配策略	102
5.4.3 超启发式优化算法	103
5.5 基于多特征融合的集成表面组装时间估计器	105
5.5.1 数据的生成与准备	105
5.5.2 组装过程的特征提取	105
5.5.3 集成学习估计器的结构设计	107
5.6 实验设计	108
5.6.1 组装时间估计器的对比实验	109
5.6.2 生产线元件分配算法的对比实验	110
5.6.3 运行效率分析	114
5.7 本章小结	114
结 论	116
参考文献	119
攻读博士学位期间发表的论文及其他成果	130
学位论文评阅人、答辩委员会名单及答辩决议	132
哈尔滨工业大学学位论文原创性声明和使用权限	133
致 谢	134
个人简历	136

Contents

Abstract (In Chinese)	I
Abstract (In English)	III
List of physical quantity and symbol	VI
Chapter 1 Introduction	1
1.1 Background and significance of research	1
1.2 Overview of surface assembly system	3
1.3 Research review on surface assembly process optimization	6
1.3.1 Assembly process optimization of surface mounters	6
1.3.2 Workload balancing optimization of assembly lines	15
1.4 Problems of current research	19
1.5 Major research contents and organization of this thesis	22
Chapter 2 Mathematical programming models for surface assembly process optimization	26
2.1 Introduction	26
2.2 Description and analyses of the system	27
2.2.1 System configuration	27
2.2.2 Surface assembly process optimization	29
2.3 Multi-stage modelling	30
2.3.1 Analyses and assumptions	30
2.3.2 Task assignment model for assembly-heads	32
2.3.3 Path planning model for placement process	36
2.3.4 Workload balancing model for assembly line	39
2.4 Validation and analyses of the model	40
2.4.1 Example demonstration	40
2.4.2 Sensitivity analyses	44
2.4.3 Complexity analyses	46
2.5 Summary of this chapter	47
Chapter 3 Optimization algorithm for head task assignment algorithm in surface assembly process	48

3.1	Introduction.....	48
3.2	Problem analysis	49
3.3	Head task assignment algorithm based on the cyclic task integration model....	49
3.3.1	Cyclic task integration model	50
3.3.2	Construction of the initial feasible solution	51
3.3.3	Solvability enhancement strategies for the model	53
3.3.4	Selection criteria for the optimal solutions	55
3.4	Hierarchical heuristic look-forward scanning-based head task assignment	56
3.4.1	Overview of the algorithm.....	56
3.4.2	Nozzle pattern assignment heuristic algorithm	57
3.4.3	Feeder-slot assignment heuristic algorithm	58
3.4.4	Head-component assignment heuristic algorithm	62
3.5	Experimental design.....	65
3.5.1	Comparative experiments on the model-based head task assignment	66
3.5.2	Comparative experiments on the heuristic-based head task assignment ...	69
3.6	Summary of this chapter	75
Chapter 4 Path planning algorithm for the placement process in the sur-		
	face assembly process	76
4.1	Introduction.....	76
4.2	Problem analysis	77
4.3	Path planning algorithms for placement process	78
4.3.1	Overview of the algorithm.....	78
4.3.2	Dynamic programming-based path planning algorithm of pick- and-place cycles.....	79
4.3.3	Path planning algorithm based on the multi-point greedy beam search with dynamic orientation	80
4.3.4	Aggregated route relink algorithm based on adaptive large neigh- borhood search	83
4.4	Experimental design.....	86
4.4.1	Comparative experiments on heuristic-based path planning for placement process	87
4.4.2	Comparative experiments on the neighborhood search-based path planning for placement process	89

4.4.3	Analysis of solving efficiency	93
4.5	Summary of this chapter	94
Chapter 5 Workload balancing algorithm for the assembly line in surface		
	assembly process	95
5.1	Introduction.....	95
5.2	Problem analysis	96
5.3	Algorithmic framework	97
5.4	Hyper-heuristic-based component allocation algorithm for surface as- sembly line.....	98
5.4.1	Low-level heuristic operators.....	98
5.4.2	Duplicated component allocation strategy	102
5.4.3	Hyper-heuristic optimization algorithm	103
5.5	Surface assembly time ensemble estimator based on multifeature fusion.....	105
5.5.1	Data generation and preparation	105
5.5.2	Feature extraction of assembly process	105
5.5.3	Structure design of the ensemble learning estimator.....	107
5.6	Experimental design.....	108
5.6.1	Comparative experiments on the assembly time estimators.....	109
5.6.2	Comparative experiments on the line component allocation algorithms .	110
5.6.3	Analysis of solving efficiency	114
5.7	Summary of this chapter	114
Conclusions		116
References		119
Papers published in the period of Ph.D. education		130
List of Dissertation Reviewers and Defense Committee and Defense Resolution.		132
Statement of copyright and Letter of authorization		133
Acknowledgements		134
Resume		136

第1章 绪论

1.1 研究背景及意义

制造业是国民经济发展的支柱产业，对经济增长起着重要的推进作用。我国的“十四五”规划指出，要深入实施增强制造业核心竞争力和技术改造的专项计划，鼓励企业应用先进适用技术、加强设备更新和新产品规模化应用，培育先进制造业集群，推动集成电路、高端数控机床等产业的创新发展，推动制造业产品“增品种、提品质、创品牌”^[1]。“中国制造2025”进一步指出，要加快发展智能制造装备和产品，组织研发具有深度感知、智慧决策、自动执行功能的智能制造装备以及智能化生产线，提升国家制造业创新能力，实现制造强国的战略任务。大力发展先进智能制造，促进新质生产力的形成和现代化产业体系的构建，确保我国产业体系的完整性、先进性和安全性，是实现中国式现代化的必由之路^[2-3]。

电子制造是专注于研究、制造和生产各类电子设备和电子元件的产业，涵盖了电子产品从设计、制造到测试的全过程，作为现代制造业的重要组成部分，其在诸多领域如消费电子、通信设备、汽车电子等有着广泛的应用，具有市场规模大、技术进步快、产业关联度强等特点。电子制造业的发展对整个产业链的升级与变革起到了重要的推动作用。现代电子制造技术包含电路设计、元件选型、表面组装和生产质量管控等多个方面。其中，表面组装是将电子元件安装在印制电路板（Printed Circuit Board, PCB）焊盘上的板级组装技术，是现代电子组装技术的核心。近年来，随着电子消费市场需求的持续增长，表面组装等制造自动化技术也在快速地发展进步与更新迭代。表面组装过程优化通过调整生产工序、改进组装流程以最大化利用生产制造设备的性能，在生产效率提升方面起着关键作用，推动着生产流程向高效化转型，是现代电子制造业核心竞争力的重要体现。

传统的表面组装生产线构成如图1-1所示，主要包括上板机、丝网印刷机、贴片机、回流焊炉、光学检测机以及下板机。作为生产线的前端设备，上板机内部装载有堆叠放置的电路板，通过逐级提升的方式将其传入生产线。丝网印刷机用于将锡膏印制到电路板上的焊接位置（即焊盘），贴片机则将电子元件准确地放置到涂有锡膏的电路板上。回流焊炉将贴装好的电路板进行加热，使

锡膏融化并与电子元件的引脚连接。冷却凝固后的锡膏在电子元件和焊盘间形成稳固的连接。光学检测机作为可选设备，用于检测表面组装质量。最后，下板机负责接收完成组装的电路板。



图 1-1 表面组装生产线的构成

Fig.1-1 Composition of a surface assembly line

贴片机作为表面组装生产线中最关键、最重要的设备，其主要任务是完成 PCB 上元件的表面组装工作。贴片机的研制与表面组装元件的发展密不可分。图1-2展示了表面组装相关技术的发展历程。自上世纪七十年代后期，日本的大型电子企业集团率先开始自动贴片机的研制工作，并逐渐将贴片机由内部专用设备改为标准化的商用设备，开始批量应用到电子设备加工生产中。经过四十余年的迅速发展，贴片机逐步向着精密、高速、模块化的方向发展，以松下、三洋、韩华等为代表的企业研发的贴片机占据了国内外主要市场。贴片机的生产制造能力是反映电子制造业发展水平的重要指标之一，表面组装生产线的生产效率更是整个电子产业链发展中的核心竞争力。我国作为制造业大国，电子制造产业起步相对较晚而发展迅速，目前已经成为世界上贴片机应用最广的国家之一，对贴片机的需求也亟待从依赖国外进口走向国内自主研发。

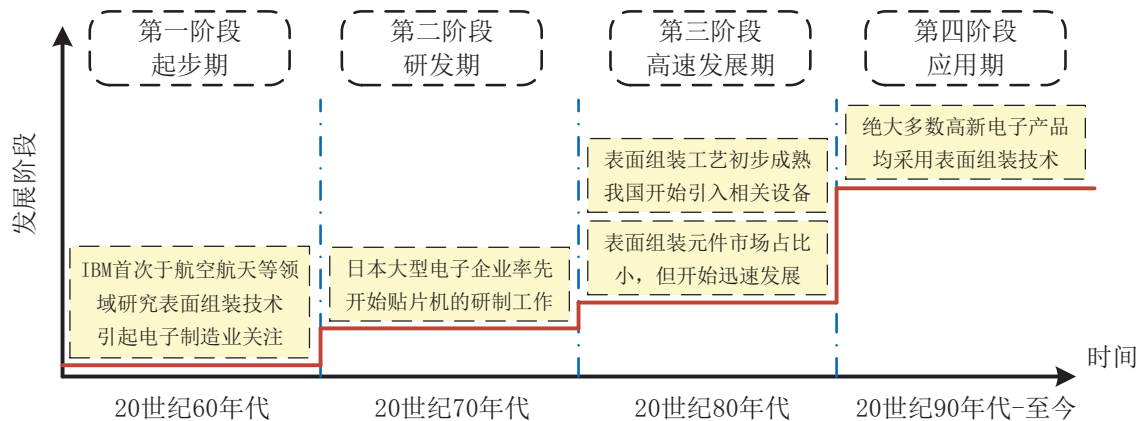


图 1-2 表面组装技术的发展历程

Fig.1-2 The development history of surface mount technology

贴片机作为光机电一体化设备，其研制过程是涉及机械设计技术、运动控制技术、组装过程优化技术、软件工程技术和机器视觉图像处理技术等多学科

交叉的系统性工程。其中，表面组装过程优化技术极大影响着贴片机的组装效率。表面组装过程优化问题是一类特殊的复杂优化问题，启发式、数学规划和机器学习等算法常用于解决此类工业组合优化问题^[4]。作为一个高维度、多变量且高度耦合的非确定性问题，该问题具有较高的时间复杂度，导致现有的优化算法难以在可接受时间内找到全局最优解。同时，表面组装过程还面临着来自内外部多种因素的制约，限制了算法的适用范围。为了提升组装过程效率，制造商常采用多台贴片机协同作业的方式来完成 PCB 的组装任务，由此带来了生产线负载平衡问题。组装生产线的优化研究的是平衡不同贴片机之间的工作负载、最小化贴片机最长工作时间的策略。表面组装过程优化，作为工业实际需求驱动的课题，在系统建模与算法设计时，需在满足约束条件下优化生产效率指标，以确保优化结果的高效性、可行性与可部署性。

1.2 表面组装系统概述

本文研究的是装配有并列式贴片头的贴片机（以下简称并列式贴片机），其布局如图1-3所示，包括一个固定的 PCB 平台，两个固定的供料基座，以及带多个贴片头的移动悬臂，装载有元件的供料器安装在前后两个基座上。带式、管式、托盘式的供料器用于供给各类封装元件。并列式贴片机适用于中小批量生产，也可将多台贴片机组合用于大批量生产。悬臂在电路板和供料基座之间移动，其装配有多个贴片头用于拾取和贴装元件，贴片头上配备的飞行相机用于元件检测；对于大尺寸的元件，悬臂必须移动到固定相机上进行检测。元件组装所用的吸嘴类型和其封装的尺寸相关，自动吸嘴更换器（Auto Nozzle Changer, ANC）中装载着不同类型的吸嘴，以适应贴片机对多种元件封装的装配要求。并列式设计结构的贴片头其间隔为槽位间隔的整数倍，使多个贴片头可以同时从不同槽上的供料器中拾取元件。相较于装配有转塔式贴片头的贴片机（以下简称转塔式贴片机），并列式贴片机的机械结构简单，可靠性强，多头同时取料装配效率更高，可实现高速、高精度表面组装，适用的元件封装类型更加多样，因此也被称为多功能贴片机，其在结构上主要由以下部分组成：

1) 传动与驱动机构: 贴片机的基本构成为一个高精密的运动控制系统。传动系统负责将电路板传送至贴装区域，伺服定位系统支撑着贴片头，保证其精确定位和精准移动。贴片机的主体包括两条相互平行的定臂和一条动臂，由三个直线导轨控制悬臂在水平面内沿 X 轴和 Y 轴方向运动。直线导轨由电机驱动，两个定臂对应的电机为 Y 轴方向电机，带动动臂沿 Y 轴方向运动，动臂对

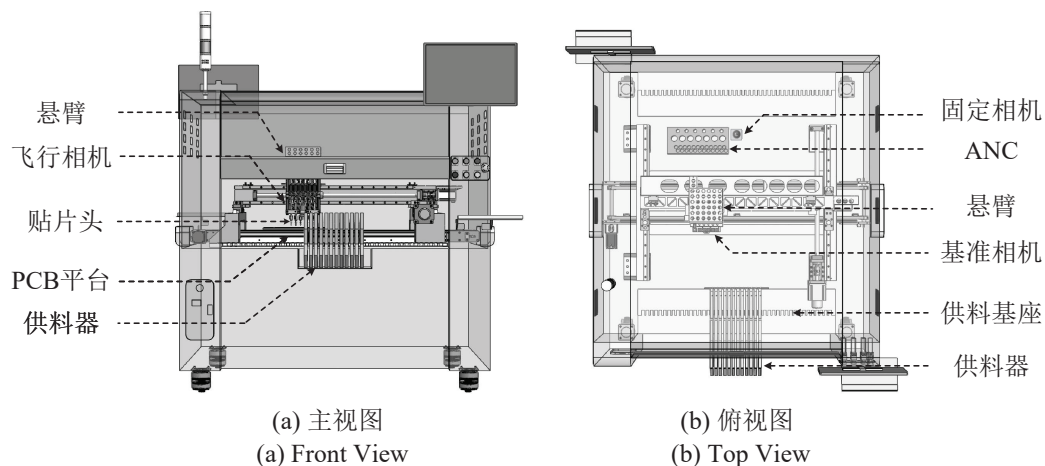


图 1-3 贴片机组成结构图

Fig.1-3 Composition and structure of surface mounter

应的电机为 X 轴方向电机，带动悬臂实现 X 轴方向运动。悬臂上安装有多个并列排放的贴片头，贴片头上的吸杆均可以在电机的驱动下实现 Z 轴方向运动，完成对元件的拾取和贴装，另有 R 轴电机带动吸杆绕轴转动，实现对拾取元件的角度偏移进行补偿。

2) 元件供应与贴装机构: 元件由供料器供应，不同封装的元件所使用的供料器类型、占用的槽位数也不相同。供料器通过机械化接口固定在供料器基座上，该接口被称为供料器槽。元件的拾取由贴片头上的吸杆完成，供应则由供料器提供。吸杆内部中空并在底部装有吸嘴，贴片头与真空泵连通，真空泵抽气在吸杆内部建立真空以吸取元件、关闭真空以贴装元件。贴片头为完成多种类型元件的组装，需移动至 ANC 进行吸嘴的安装与更换。

3) 视觉检测系统: 贴片机配备视觉系统实现高精度组装，其由一个基准相机、一个固定相机和多个飞行相机构成。基准相机用以检测 PCB 上的基准点，对电路板位置进行校准补偿。飞行相机直接安装在悬臂上，用于校准贴片头拾取元件的位置偏移和角度偏差，其可在贴片头移动过程中同步完成检测，从而节省识别时间、提高组装效率，但飞行相机的识别范围和识别精度相对较低。固定相机安装在贴片机内部，具有视野大、识别精度高的特点，适用于细间距和大型尺寸的元件。各类相机配合使用，可用于对各种尺寸的基准点和元件的检测识别，实现贴片机的高速高精度元件组装。

4) 计算机控制系统: 计算机控制系统记载着大量的贴片机数据参数，用于识别和处理设备及各传感器的信号，并根据预定程序控制贴片机的组装动作，控制系统内部的调度程序保证贴片机的高效运转、状态的实时监控和异常的快速处理等。

以上系统共同协作，使得贴片机能够高效、精准地完成电子元件的组装任务。表面组装过程的流程如图1-4所示，具体地，其主要由以下步骤组成：

- 1) **PCB 装载**：传送装置将 PCB 载入到预定位置，以基准相机校准其位置。
- 2) **工具匹配**：检查贴片头吸嘴和拾贴元件的一致性，不匹配则进行更换。
- 3) **元件拾取**：贴片头移动到供料器基座，从供料器中拾取元件。
- 4) **视觉检测**：检测已拾取元件是否存在缺陷，若是，执行步骤 5；否则，校准元件的中心偏移后执行步骤 6。
- 5) **元件废弃**：贴片头废弃有缺陷的元件在废料盒中，执行步骤 7。
- 6) **元件贴装**：贴片头搬运检测通过的元件并将放置在 PCB 对应焊盘上。
- 7) **循环**：重复执行以上步骤 2–6，直至所有焊盘均组装完成对应元件后，传出 PCB。

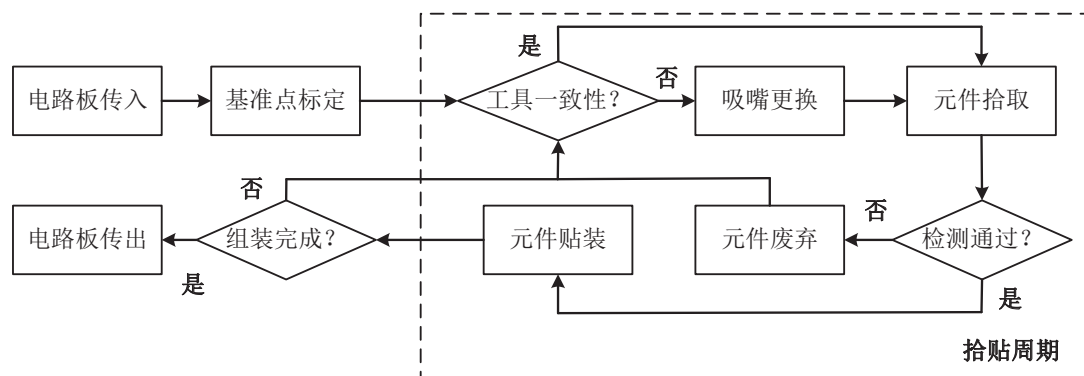


图 1-4 表面组装过程流程图

Fig. 1-4 Flowchart of surface assembly process

贴片机的表面组装过程由一系列复杂的工艺流程组成，本文将拾取、检测、贴装的一个周期性过程称为拾贴周期，将具有相同贴片头拾取槽位的连续拾贴周期称为周期组。主流贴片机的元件废弃率普遍低于 0.1%，对整体组装效率的影响较小，故元件废弃过程通常不在优化的环节之内。贴片头拾取的槽位决定了当前周期拾贴的元件类型和需要安装的吸嘴类型。在组装过程中，贴片头的移动路径和组装动作共同决定了组装过程的效率。组装动作包括贴片头的元件拾取、吸嘴更换和元件贴装动作，元件贴装动作仅和贴装点数相关，而元件拾取和吸嘴更换动作则取决于对组装过程优化的结果。

表面组装过程优化涉及多个问题的决策优化。对于贴片机的组装过程，元件拾取和贴装动作、贴片头的吸嘴更换动作以及移动路径均会对组装时间产生影响。其中，供料器安装的基座槽位决定了拾取过程的移动路径和能达到的最大拾取效率；贴片头装配的吸嘴类型决定了可拾取的元件类型，其需要根据组

装任务进行吸嘴更换以提高拾取效率；贴片头的拾贴元件类型和供料器安装槽位共同决定了拾取次数，同时限制了各周期可贴装的焊盘位置，进而影响了贴装过程的路径规划。多变量和高度耦合的决策过程增大了问题求解的难度。

表面组装生产线负载平衡问题，在贴片机组装过程优化的基础上，增加了贴片机之间元件分配的决策优化。表面组装生产线与传统生产线在元件分配层面既具相同性，又有差异性。相同之处在于，二者均涉及将元件分配至不同机器进行组装加工的决策，且允许多台机器协同装配同一类型元件，以提升整体的生产效率。同时，二者均需考虑装配的优先级要求，并可能因元件封装等限制而影响其可分配的机器。差异性则主要体现在，组装生产线的负载均衡优化受到可用工具资源限制的影响大，不同类型元件因受限于供料器等专用工具，其可分配至贴片机的数量上限受到影响。生产线元件分配的结果决定了贴片机的组装任务，更为关键的是，在评估贴片机的组装效率时，组装生产线的复杂性更为突出。贴片机的组装时间还受到贴片机类型、可用工具、优化算法等多重因素的影响。由此可见，表面组装生产线的优化是一个多维度、高度集成的决策过程，必须综合考虑生产线各环节的协同工作，以实现整体效率的最大化。

1.3 表面组装过程优化的研究现状

1.3.1 贴片机的组装过程优化

作为一类组合优化问题，数学规划和启发式优化是两类常用于解决贴片机的组装过程优化的方法。其中，数学规划是用定量的数学表达式构建决策变量之间的目标和约束函数的方法，可以实现系统的解空间搜索，确保了优化结果的可靠性和准确性，而其在处理复杂优化问题时，运算效率受数据规模的影响大；启发式优化则基于经验和直觉设计搜索规则快速寻找问题的较优解，具有适应性强、求解效率高和易于实现等特点，算法的性能取决于搜索策略的设计，复杂解空间增加了设计有效搜索规则的难度，同时可能会降低解的质量。混合优化将复杂优化问题进行拆分，运用不同的方法分阶段地解决问题，旨在同时提高问题的求解效率和解的质量。图1-5展示了用于解决表面组装过程优化问题的主要方法分类。

1.3.1.1 数学规划法

在早期的表面组装过程优化研究中，贴片机类型呈现出多样化的特点，相关研究以最小化悬臂的移动路径为目标，构建了不同类型的简化数学模型。Ho等^[5]研究了单头顺序贴片机的建模问题，通过构建供料器位置分配和贴装顺序

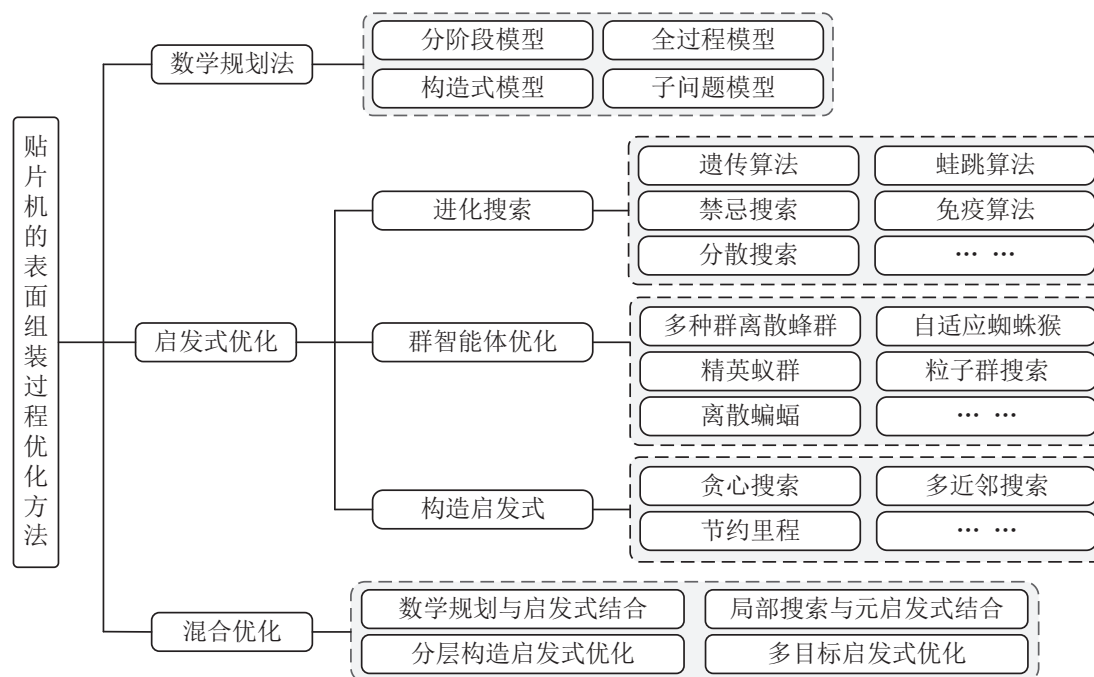


图 1-5 表面组装过程的主要优化方法分类

Fig.1-5 Classification of primary optimization methods for surface assembly process

规划互为已知约束的双向模型，将两个模型整合为整数线性模型，获得了该类贴片机组装过程优化的最优解。Altinkerner 等^[6] 围绕转塔式贴片机的建模问题开展研究，其将表面组装问题归结为供料器分配和拾贴顺序决策问题，运用拉格朗日松弛定理分解数学模型到两个子问题，将组装路径优化问题转换为车辆路径规划问题（Vehicle Routing Problem, VRP），提出 ε 近似解的模型求解方法，将所得解与最优解的误差控制在了等效 VRP 的误差范围内。Alkaya 等^[7] 研究了供料器基座和电路板支撑平台可移动的转塔式贴片机的数学模型，通过将移动路径规划和供料器位置分配视为顺序相关的旅行商和二次分配问题，建立了非线性整数规划模型。Luo 等^[8] 研究了 LED 贴片机的贴装路径优化问题，将贴片位置的拾贴顺序问题视为具有城市批次的旅行商问题，构建了贴片头移动路径最小化的混合整数规划模型。

为应对日趋复杂的贴装任务，相关研究围绕并列式贴片机展开研究，其表面组装过程模型的构建更为困难，因而部分研究采取分解策略来进行建模。Luo 等^[9-10] 将问题分解并设计了两阶段混合整数线性规划模型：在第一阶段解决吸嘴-贴片头分配问题，在第二阶段通过优化供料器-槽位分配和元件贴装顺序最小化移动路径，适用于小规模数据的求解。Ashayeri 等^[11] 提出了针对元件-贴片机分配和拾贴顺序规划的模型分解方法，从装配时间最小化和换线时间最小化两个角度，对组装过程的供料器分配和路径规划问题进行建模。Hsu 等^[12] 讨论

了多机头贴片机吸嘴更换、拾取动作、移动路径等因素对组装效率的影响,以最小化组装时间为目标,构建了可同时解决元件顺序分配、供料器分配和吸嘴分配问题的混合整数线性规划模型。此外,也有部分研究关注于组装过程子问题模型的构建。**Raduly-Baka**等分别研究了吸嘴分配^[13]和供料器模组分配^[14]问题,前者讨论了有无成本限制的吸嘴选择问题,并以近似线性规划模型实现了在多项式时间内获得最优的吸嘴分配的解;后者则研究了多类型电路板组装时,最小化准备供料器的换线时间优化问题,将其建模为工具分区模块分配模型。**Sun**等^[15]研究了贴装过程路径规划的整数规划模型,提出了具有高效剪枝策略的分支定价算法,运用了动态规划解决列生成中的子问题,使其在处理小规模数据时能高效地获得问题的最优解。

数学规划法在解决复杂组合优化问题时有较高的计算复杂度,因而部分研究在构建模型的过程中应用了一系列技术降低问题的规模,进而提升模型的可解性。**Torabi**等^[16]将表面组装过程优化问题视为多目标优化问题,提出最小化周期和最大化吸嘴-元件适配度两个目标,建立最大最小化整数规划模型,并设计增广 ϵ 方法实现了对中小规模问题的求解。**Ashayeri**等^[17]引入了批处理的方法,将单个贴片头组装的一批同类贴装点的元件类型作为决策变量,建立了决策变量数更少、模型复杂度更低的混合整数规划模型,能有效平衡贴片头工作量、减少吸嘴更换和提升悬臂移动速度,适用于元件类型有限而贴装点较多的数据优化。**Guo**等^[18]将并列式贴片机的组装过程优化等价于多车厢车辆路径规划问题(**Multi-Compartment Vehicle Routing Problem, MCVRP**),通过将问题拆解为拾取组合和贴装排序问题,从提升拾取效率的角度出发,以不同类型元件构成同步拾取组,建立了相应的混合整数规划模型。

综上所述,现有研究对表面组装过程模型的模型构建展开了详细的分析,并取得了一定的进展,相关分类如表1-1所示。全过程的模型主要围绕简单类型贴片机的模型展开,而并列式贴片机的数学模型通常采用分阶段构建的方式,并有部分研究专门对其中的子问题进行建模。不同角度的构造式建模方式降低了模型的复杂度,但尚未完全解决模型求解效率受数据规模限制大的局限性。

表 1-1 主流研究中数学规划模型的构建方法

Table 1-1 Construction method of mathematical programming model in mainstream research

类别	相关研究
全过程模型	双向约束整数线性规划 ^[5] 等效转换 ^[6,8] 非线性整数规划 ^[7]
分阶段模型	两阶段：吸嘴分配-拾贴顺序 ^[9-10] ，供料器分配-拾贴路径规划 ^[11] 三阶段：元件顺序-供料器分配-吸嘴分配 ^[12]
子问题模型	吸嘴分配 ^[13] ，供料器模组分配 ^[14] ，贴装过程路径规划 ^[15]
构造式模型	多目标最大-最小整数规划 ^[16] 元件组批处理整数规划 ^[17] 基于同步拾取组的 MCVRP 模型 ^[18]

1.3.1.2 启发式优化方法

多种元启发式优化技术已被成功运用在表面组装过程的优化领域，遗传算法作为其中最具代表性的优化方法，其编码方式是相关元启发式优化方法设计的基础，主要可分为以下几类：

1) 子过程的一维顺序编码

表面组装的子过程优化主要可分为供料器-槽位分配和元件贴装顺序优化两部分。在供料器-槽位分配编码中，主要通过迭代不同的供料器排列顺序，确定元件的拾贴顺序。Sun 等^[19]提出了贪心的供料器基座扫描策略，以最大化同步取料的贴片头数作为评价编码结果的指标；Xing 等^[20]提出了改进的单亲遗传算法，根据个体适应度大小，合理划分个体在子代出现的次数，减少种群迭代次数。在元件贴装顺序编码中，Hardas 等^[21]以贴装路径和运算效率为比较指标，通过实验分析路径、顺序、邻接的基因表示形式，以及部分映射、有序交叉、交替边搜索算子对优化结果的影响；Peng 等^[22]以贴片头移动距离最小为目标，通过分析不同取料方式下的贴装路径差异，优化了贴片头的取贴顺序和取料方向。

2) 全过程的一维顺序编码

全过程的顺序编码是对表面组装过程中多个子问题的同时编码，具有并行搜索的优势。Kulak 等^[23]提出了供料器分配和贴装顺序相关的局部关联染色体编码方式，运用分布密度导向构建法对贴装点顺序进行分组聚类，有效地限制了问题规模、提升了搜索效率。Gyorfi 等^[24]在供料器-贴装顺序编码的基础上，提出一种支持多贴片头的供料器拾取和元件贴装顺序相互独立的关联链接评估方法，研究在迭代过程中平衡父链接样本空间和搜索多样性的方法。Wu 等^[25]设计了元件拾取序列、贴装序列、供料器排列序列的三链染色体，在同时优化贴

装顺序和供料器分配槽位基础上, 结合贪心解决元件拾取顺序问题。Wu 等^[26]提出适用于贴装顺序、拾取顺序、贴片头元件以及供料槽元件分配相结合的个体编码与解码方式, 将最小准则原则运用在个体的筛选过程中, 维持搜索过程的多样性。

3) 全过程的二维实数编码

二维实数的编码同时对分配顺序和分配类型进行编码, 具有收敛速度快、输出结果稳定等优势。在对元件贴装顺序和供料器安装槽位同时优化的相关研究中, Du 等^[27]将问题拆分为元件分组、元件组拾取和元件组贴装子问题, 提出一种二维实数编码的混合遗传算法, 设计了元件分组启发式算法; Yin 等^[28]结合元胞自动机增强算法全局搜索能力, 提出二维分段式编码方式。此外, Li 等^[29]从贴片头的元件分配出发, 通过同类元件的贴装点组的拆分和重组, 按元件组在贴片头上的分配顺序进行编码, 以平衡拾贴周期、吸嘴更换、同步拾取等性能指标为目标, 提出一种基于元胞分裂的贴片头元件分配算法, 并设计启发式算法用于供料器的槽位分配。

基于上述的编码方式, 在表面组装过程优化问题中, 相关的元启发式算法也得到了广泛的研究和应用, 主要可分为以下几类:

1) 进化搜索优化方法

进化算法受自然选择和生物进化机制启发, 通过不同的基因表达方式代表优化问题的解, 适用于解决包括组装过程优化在内的复杂问题。Tsuchiya 等^[30]为解决贴片头同步拾取优化问题, 提出一种启发式多起点的局部搜索算法, 对随机生成的供料器分配结果增加解的可行性、贴片头的工作效率和同步拾取可行性等判断, 并采用拾取模式匹配确定了取料顺序。Zhu 等^[31]提出改进的混合蛙跳算法解决组装过程中的拾贴顺序优化问题, 建立了针对整数编码的跃迁步进、二元步进以及适用于离散数据的跃迁运算符, 结合个体记忆化搜索模式, 运用方差分析法确定了算法的最优参数, 使其收敛速度更快、精度更高。Luo 等^[32]研究将禁忌搜索运用到贴装路径规划问题解决中, 采用双禁忌表避免迂回搜索, 根据问题特性设计了插入移动的参考解局部搜索策略, 有利于算法跳出局部最优解。Liu 等^[33]提出分散搜索法分别解决表面组装过程中的元件-供料器、供料器-槽位和拾贴元件顺序子问题, 通过最小化拾贴周期、构建同步拾取和优化元件拾贴顺序来减少组装时间。Xu 等^[34]在免疫算法中运用了供料器槽位分配-贴片顺序编码方式, 通过改进免疫算子以及调整参数, 对多头贴片机的贴装过程进行优化求解, 独特性免疫网络使得搜索群体呈现多样化的特点, 有利于算法的全局优化。

2) 群体智能优化方法

群体智能通过模拟社会性生物的群体行为进行优化,具有鲁棒性强、搜索速度快的优点,被广泛应用到表面组装过程优化中。**Castellani**等^[35]将定制的蜂群算法用于解决供料器槽位分配和贴装顺序问题,提出了五种新的搜索算子,使算法不依赖于初始解且有较好的收敛性。**Hsu**等^[36]提出了改进的多种群离散蜂群算法,同时优化元件取贴顺序和供料器分配槽位,在全局探索-开发策略的基础上,运用多蜂群以及自适应、离散移动步长等新特性,有助于算法找到问题的较优解。**Wang**等^[37]提出基于自适应蜘蛛猴算法优化表面组装过程,通过分层次的算法结构与表面组装过程结合,实现了在全局领导者环节解决供料器槽位分配问题,并在局部领导者环节解决贴装顺序问题,提升了组装过程效率。**Fu**等^[38]研究了转塔式贴片机的优化问题,将表面组装过程拆分为拾贴周期内和周期间规划的子问题,在改进的蚁群算法中引入精英蚂蚁,并采用全局和局部相结合的信息更新策略,具有全局搜索能力、较好的求解精度和较快的收敛速度。基于此,**Yu**等^[39]提出了基于拾贴周期内部路径优化和周期间路径的蚁群算法,对双悬臂双旋转头路径规划进行研究,克服传统算法中约束条件较多的局限性。**Hsu**等研究了运用粒子群算法^[40]和改进的混合蛙跳算法^[41]同时解决供料器槽位分配和元件拾贴顺序规划问题,其在粒子群算法中提出了离散问题向连续空间转化的编码机制,并引入了自适应跳跃、自适应变体等智能算子,通过迭代减少记忆体数量,实现了探索到开发的过渡,有助于算法搜索高质量解。**Griffiths**等^[42]研究了新型离散混合蝙蝠启发算法,重新定义了搜索过程的位置和速度等参数,用于解决六轴机械臂组装元件规划时的供料器槽位分配和拾贴顺序规划问题。**Lu**等^[43]研究了复合式双悬臂转塔式贴片机的优化过程,根据种群的交流结构和信息传递机制,将元胞自动机理论融入蝙蝠算法中,引入多交叉算子自适应的策略提升鲁棒性,设置精英归档提高迭代速度和种群位置多样性,提出迭代互换操作来执行局部搜索,以提高算法的收敛精度。

3) 多目标启发式优化方法

多目标优化也被应用到具有多个效率相关子目标的表面组装过程优化问题中,且常与元启发式算法结合解决问题^[44]。**Torabi**等^[16]将表面组装过程的优化目标拆分为最小化瓶颈贴片头负载和最大化元件-吸嘴匹配度两部分,以**Taguchi**方法调整了算法参数,研究了优化贴装过程的粒子群算法。**Zhu**等^[45]将表面组装优化任务视为最小化元件贴装移动路径、供料器拾取移动路径和吸嘴更换的多目标优化问题,通过最佳参考点和比较点之间的关系,提出了一种基于**Fréchet**距离曲线匹配的组装过程多目标优化遗传算法。**Lu**等^[46]基于灰熵关

联分析建立表面组装中拾取顺序、贴装顺序和吸嘴更换顺序子目标相关的灰熵关联度函数，提出的多目标差分算法保证至少两个子目标优于传统遗传算法，且解集分布更均匀。Gao 等^[47]运用分层序列法，分析影响组装效率的子目标之间的耦合关系，根据优化子目标将问题拆分为吸嘴分配、元件分配、供料器分配和拾贴顺序规划问题，提出了分层次启发式的表面组装过程优化方法。

4) 其他启发式优化方法

构造启发式等优化方法也被应用到了表面组装过程子问题的优化中。吸嘴分配限制了贴片头可分配的元件类型，影响了表面组装过程的拾取次数和吸嘴更换次数。Knuutila 等^[48]研究了拾取次数最小化的吸嘴分配问题，在给定吸嘴拾贴顺序的前提下，贪心的吸嘴分配策略在单次拾取中尽可能多地拾取元件以提升效率。Raduly-Baka 等^[13]研究了贴片头的吸嘴类型的选取问题，讨论了在可用贴片头数量限制下的吸嘴分配问题，而后提出了在增加吸嘴预算限制下的最优吸嘴分配的高效的贪心搜索算法。在此基础上，Knuutila 等^[49]研究了以最小化拾贴次数为优化目标的吸嘴配置问题，分别运用遗传算法和粒子群算法启发式方法对吸嘴库配置进行了优化，并在求解小规模问题实例时取得了接近最优的结果。Pyötiälä 等^[50]研究了以多近邻搜索解决供料器位置分配问题，并同随机分配、最小生成树、基于频次的平衡算法进行比较，提出了基于随机/周期空闲优先启发式、随机/周期下一空闲启发式规则的算法对拾取贴装路径进行优化。Han 等^[51]研究了一种供料器分配算法，该算法以基座最左侧槽位为起点，贪心地将距离当前贴装点最近的元件分配到槽位，并不断更新基准点进行连续分配，通过自定义距离函数进行全局优化。Chen 等^[52]研究了供料器分配已知的元件拾贴顺序优化，通过混合蛙跳-蚁群算法对贴装顺序进行路径寻优，提出了适用于贴装过程的分段启发函数、信息素分段编码和更新等改进方法，有效地提升了算法的求解精度和全局搜索能力。Li 等^[53]将贴装路径问题和供料器分配问题被转化为启发式排序优化问题，以索引序列对的能量矩阵优化拾贴周期序列，提出了基于 Hopfield 神经网络的路径相关序列优化方法。Wang 等^[54]运用发明问题解决（Theory of Inventive Problem Solving, TIPS）理论，研究了表面组装中时间延迟、元件供料、拾取和贴装过程之间的协同优化问题，提出一种基于蚁群优化算法的仿真精简装配路径策略，减少装配时间的同时保持组装过程的高精度和低能耗。

1.3.1.3 混合优化

混合优化算法是指将多种不同的优化方法和策略进行组合，用来解决复杂优化问题的算法。其中，将进化算法常与构造启发算法相结合的混合优化方法，

可以明显提高优化的效果,适用于多模态、高维、带约束和多目标的优化问题,在工业界有着广泛的应用^[55]。

1) 两阶段混合优化

两阶段优化法通常从拾贴-贴装的角度对表面组装过程优化问题进行拆分,并以不同的策略分别求解。**Grunow**等^[56]通过改进最小生成树并结合贪心法分配供料器至槽位中,将贴装顺序问题视为VRP通过改进Clarke-Wright节约里程法进行求解,并结合了局部搜索规则改进供料器分配和拾贴顺序规划结果。**Park**等^[57]采用动态规划法分别解决供料器分配和贴装顺序规划问题,通过启发式算法从贴片头元件分配结果中构造供料器组合,确定了供料器的安装槽位和拾贴路径,以2-opt交换拾贴周期期间的已分配贴装点,从而改进解的质量。**Chen**等^[58]研究了供料器分配和贴装顺序规划子问题的独立性和关联性,采用分治法在两个问题中分别运用改进的蛙跳算法和禁忌搜索算法,通过二者迭代优化改进搜索解的质量。**Gao**等^[59]提出了一种结合局部搜索和整数规划模型的混合优化算法,通过引入自定义距离权重矩阵并采用贪心法确定了拾贴顺序的初始解,研究了凸包调整策略优化拾贴顺序和整数规划模型求解供料器分配问题。此外,部分研究在两阶段优化中将元件对应多类吸嘴的情形作为优化的环节进行了讨论。**Huang**等^[60]提出了最小化吸嘴-元件适配度和吸嘴更换次数的整数规划模型解决吸嘴分配问题,结合了混合蛙跳算法同时对贴装顺序、供料器分配、拾取顺序进行优化。**Hsu**等^[12]提出了改进的萤火虫算法与动态规划结合的优化算法,采用秩序值技术进行整数编码和自适应搜索,同时优化了供料器安装槽位和拾贴顺序,最后运用动态规划解决了元件-吸嘴适配度最小化的吸嘴分配问题。

2) 多阶段混合优化

更多的研究致力于将表面组装过程的优化问题进一步拆分,分阶段地解决一系列更小的子问题。其中代表性的工作为**Lee**等^[61-62]提出的分层启发式优化算法,其通过平衡各贴片头的工作负载并从中提取供料器组合,以启发式动态规划的方法确定供料器组的最优安装槽位,运用启发式的吸嘴对排序算法和元件组构造算法确定贴片头分配任务,最后采用最近邻算法求解贴装过程路径规划问题,给出了一套完整的解决方案。类似地,多阶段的混合优化算法被广泛应用于并列式贴片机的组装过程优化。**Xing**等^[63]在平衡贴片头负载的基础上,运用启发式的选择策略为贴片头分配吸嘴,减少拾取过程中的吸嘴更换和元件拾取次数,在贴装过程中则以最近邻算法确定周期间和周期内的元件贴装顺序。**Luo**等^[10]将优化问题拆分为吸嘴-贴片头、元件-贴片头、供料器-槽位、拾取顺

序和贴装顺序五个相互关联的子问题,研究了最小化周期的构造启发式算法确定贴片头分配的吸嘴类型,将带有两个算子的进化算法和多邻域禁忌搜索结合,同时解决元件分配、供料器分配和拾贴顺序等问题。Guo 等^[18,64]将组装优化过程拆分为吸嘴分配、元件拾取组生成、元件拾取组拆分-合并和贴装顺序规划四个子问题,研究了单一类型元件对应多类吸嘴的编码方式,以贪心启发式分配吸嘴至贴片头,并为元件的同步拾取构造了拾取组,进而提出了基于拾取组拆分-重组的改进混合遗传算法和贪心启发式的路径规划算法。

部分研究在同步拾取研究的基础上,进一步考虑了实际应用中约束条件对组装过程的限制。Lin 等^[65-66]将组装过程分解为吸嘴-ANC 分配、贴片头-吸嘴分配和元件拾贴顺序规划问题,其在 ANC 分配阶段采用比例分配法确定了最优的吸嘴-ANC 分配结果。基于给定的 ANC 配置,在拾贴拾取、元件高度等约束下,Lin 等分别提出了基于人工蜂群^[65]和混合优化^[66]的方法解决其余子问题:前者以改进人工蜂群算法先同时解决吸嘴分配和拾取顺序问题,再解决贴装顺序问题;后者则以轮盘赌解决吸嘴分配和拾取顺序问题,再运用最近邻与 2-opt 结合优化贴装顺序。更进一步地,Lin 等^[67]研究了群搜索优化在组装过程优化中的应用,提出了改进群搜索与随机整数规划结合的方法,通过融入混沌局部搜索策略,消除了种群相似个体中的不良解,增强了搜索效率与解的多样性,优化了元件的拾取顺序。

迭代式的多阶段混合优化在表面组装过程优化中也得到了应用。Li 等^[68]将组装过程拆分为吸嘴分配、供料器分配和拾贴顺序规划问题,采用启发式确定吸嘴分配和供料器分配的初始解,以自适应最近邻搜索优化拾贴顺序,并结合受限的禁忌搜索算法搜索供料器分配的邻域解,迭代改进其他子问题的解。根据贴装点的分布特性,Li 等^[69]提出了均值 Chebyshev 联动定向搜索算法,以自适应聚类启发式的贴装点分组策略优化吸嘴和供料器分配,通过受限的禁忌搜索迭代改进供料器分配和拾贴顺序。

多阶段混合优化同样被运用到了双悬臂贴片机的组装过程优化中。He 等^[70]在吸嘴准备、供料器分配和拾贴顺序分解的基础上增加了负载平衡和周期规划的研究,其在双臂吸嘴准备阶段运用分段交换算法,基于此,提出了分层受限的平衡算法解决元件分配问题,并以爬山算法优化供料器槽位分配和拾贴顺序,在悬臂周期规划阶段则采用受限的邻域搜索算法。为解决双悬臂的吸嘴准备问题,He 等^[71]提出了基于不确定参数搜索启发式的线性模型分支定界优化方法,研究了具有单臂吸嘴配置、供料器配置和拾贴顺序编码的自适应聚类遗传算法,用于规划双臂拾贴周期和元件分配结果,缩小了问题的搜索空间、提高了算法

的搜索效率。

1.3.2 生产线的负载平衡优化

表面组装过程优化是表面组装生产线负载平衡的基础，组装线优化可分为单模态、混合模态和多模态生产线模式^[72]。其中，单模态和混合模态生产分别是在固定配置下组装单一类型和多种类型电路板的生产模式，多模态生产则是具有生产线配置切换的多种类型电路板的生产模式。平衡不同机器之间的工作负载可以有效缩短组装线上贴片机的最长处理时间，提升整线装配效率。生产线任务分配包含元件分配和组装顺序优化两个子问题，前者指元件的贴片机分配优化，后者指贴片机的组装过程规划。图1-6演示了表面组装过程优化的研究路线。

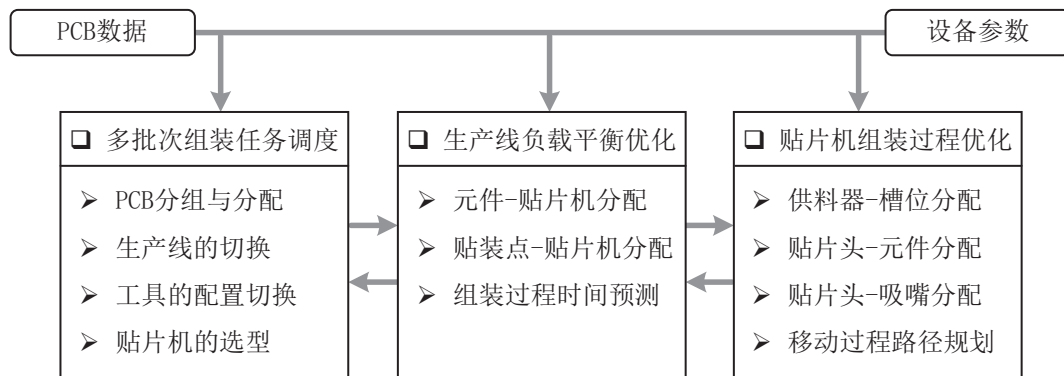


图 1-6 表面组装过程优化问题的研究路线

Fig. 1-6 Research route of surface assembly process optimization problem

1.3.2.1 生产线负载平衡

1) 数学规划法

对于单贴片头和转塔式贴片机组成的生产线，相关研究^[73-76]假定贴装点数和组装用时为线性关系，建立了组装生产线整数规划模型，实现了在可接受的时间内最优地解决元件分配问题。**Kodek**等^[73]提出了最小-最大近似整数规划模型，以产线配置用时和不同元件的拾贴用时之和作为模型的优化目标，提出了一种高效的分支定界搜索策略，使其适用于变量数中等规模的问题求解，并给出大规模数据的近似最优解和理论下界值。基于此，**Emet**等^[74]对模型做了进一步改进，根据贴片机的作业过程，增加了供料器模块、拾贴优先级和元件可重复分配等约束条件。**Rong**等^[75]研究了模组贴片机的贴片头配置和组装生产线的负载平衡问题，建立了非线性数学模型，研究了模型的线性化与松弛化方法以加快搜索速度，适用于中小规模问题的求解。**Jin**等^[76]研究了高速贴片机和多功能贴片机组成的生产线的优化，在满足异型元件分配给多功能贴片机、

细间距元件分配给高速贴片机的约束条件下,以悬臂的移动路径为工作负载的评价指标,分配元件到各模组贴片机的工作台,使工作负载和吸嘴更换次数最小。

2) 启发式优化方法

元启发式算法常用于表面组装生产线的优化,以解决贴片机共同组装单一类型电路板的优化问题。**Kulak** 等^[77] 将组装线优化问题拆分为贴片头元件分配、供料器槽位分配、拾贴顺序优化三个子问题,提出了基于遗传算法的两阶段生产线负载均衡算法,在第一阶段平衡贴片机的的工作负载并提供多个高质量的候选解,在第二阶段则对候选解进一步执行贴片机的组装过程优化算法,确定供料器的安装槽位和元件的拾贴顺序。**Wu** 等^[78] 将元启发元件分配算法和线性回归时间估计模型相结合,建立了贴装点数、贴装点间距、元件类型数等数据参数和组装时间的关系模型,以此为适应度函数并将元件分配作为遗传算法的染色体编码方式,利用贴装点分布的聚类特性,获得高质量的解决方案。**Guo** 等^[79] 研究了适用于同类型元件可分配多贴片机的混合遗传算法,以贴装点数、拾贴周期数和吸嘴更换数线性拟合组装时间,迭代式地解决元件分配、吸嘴组分配和贴片头分配三个子问题,将元件分配转换为多元件的弱分组问题进行编码,设计了选择式和非选择式的初始化和交叉策略,用贪心启发式解决吸嘴组分配和贴片头分配问题。**Liu** 等^[80] 研究了分散搜索法解决生产线负荷平衡,以统计数据评估各类元件的贴装时间,将元件分配的贴片机进行编码构建初始解集,运用 2-opt 改良初始解、修复不可行解,从可行解中选择若干最好解用于参考解集的合并和更新,保持了搜索过程中解的多样性,避免过早收敛以提升生产线的组装效率。**Mumtaz** 等^[81-82] 将提出离散化的混合蜘蛛猴算法用于同时解决组装线中的元件分配问题和单机拾贴顺序问题,其中,元件分配用于平衡贴片机的的工作负载,算法设计中融入了新的分类食物来源和基因操作算子,较传统智能优化算法提升了求解效率和解的质量。**Chen** 等^[83] 在生产线的元件分配和贴片机的拾贴顺序基础上增加了周期性维护的考虑,以同时优化最大完工时间、总能量消耗和总维护时间,并将改进的蜘蛛猴算法作为基本框架,设计启发式算子生成初始解,根据 Pareto 熵变化的演化状态平衡搜索过程中探索 and 开发,提出了基于存档的 Q 学习策略自适应调整参数,实现了优化子目标的 Pareto 最优。**Yan** 等^[84] 提出了结合自适应遗传算法与改进蚁群的混合优化算法,利用其全局搜索与快速收敛的优势,在过程中动态调整交叉和变异算子,以解决组装线负载均衡问题。

构造启发式也被应用于解决表面组装生产线优化问题。**Tóth** 等^[85] 研究了

模块化贴片机的配置和生产线的负载平衡问题，将组线问题拆分为模块-贴片头分配、贴片头-吸嘴分配和元件-供料器分配三个子问题，设计了定制启发式解决各子问题，研究了暴力搜索、随机搜索、贪心搜索和启发式算法在求解组装线优化问题中的实际效果。He 等^[86]提出了确定性分层启发式分阶段地解决组装线生产优化中的子问题，在吸嘴分配阶段以最小化周期为目标按比例将同类型元件分配给贴片机，在元件分配阶段则以最小化供料器数为目标平衡贴片机负载，以贪心算法迭代解决供料器安装槽位和拾贴顺序规划问题。Choudhury 等^[87]研究了双悬臂贴片机组装线的优化问题，通过将决策过程拆分供料器槽位分配、元件拾取组合、元件贴装分配和拾贴顺序规划四个相关的子问题，采用列表处理启发式分配元件的供料器和贴片头，将拾贴顺序规划问题转换为非对称旅行商问题，在考虑贴片头的吸嘴更换前提下，提出一种贪心启发式规划近似最优的拾贴路径。

3) 其他优化方法

部分研究将随机优化、模糊优化、多目标优化等新理论运用到表面组装生产线及生产车间的优化中。Lin 等^[88]研究了表面组装生产线的混合模态优化，将组装生产线优化问题转换为元件分组问题，为应对组装过程的订单变更和机器故障等不确定性因素，以最小化组装全部电路板的用时为目标，建立了随机混合整数规划，平衡解的最优性和环境的不确定性。Hu 等^[89]将组装生产线优化转换为分配问题和传输问题模型，提出了基于非线性隶属度函数的模糊目标规划模型，分析了模糊、不精确和不确定的环境中目标或约束条件可能出现偏差，同时优化组装线生产成本、故障次数和操作时间等多个目标，通过调整区间目标值的上下界给出最优的折中方案。Zhang 等^[90]研究了不确定需求下提升生产线效率和鲁棒性的优化方法，设计了基于强化学习的多目标进化算法，通过基于解决方案优先级的编码方式和新的任务-人工-序列的解码方式，提出了适用于鲁棒性处理和空闲时间减少处理的交叉和突变算子，运用基于 Q 学习和时间概率的自适应策略协调迭代过程中的算子，从而有效地获得 Pareto 解决方案集。Wang 等^[91]研究了拆装生产线的负载平衡问题，以最大限度地提高工人的效率、增加利润、降低能耗和平衡工人的负荷为目标，开发了一种离散多目标人工蜂群算法，综合考虑了优先级等约束和拆装过程的特性，获得了 Pareto 最优的解决方案。Zhang 等^[92]提出了一种基于直觉模糊集相似度的电路板组装车间的调度优化算法，通过聚合作业完成时间、总延迟成本、投资仓储成本和电力消耗等多个优化子目标映射至直觉模糊相似集，集合适应度函数分配策略和最优觅食算法，设计了改进的表面组装多目标进化算法。

1.3.2.2 多层次生产规划

多品种小批量的表面组装生产线负载平衡是典型的多模态组装线优化问题,相关研究主要围绕生产线的配置切换和负载平衡展开。**Jin** 等^[93]建立了包含切换时间在内的最小化组装时间的表面组装生产线调度优化模型,提出针对电路板组装特有的切换后效性的主动禁忌搜索算法,研究了组间搜索和组内搜索的分层排序优化策略,并与理论下界值比较说明了算法的有效性。**Liu** 等^[94]研究了两条生产线的多品种小批量生产线优化问题,以最小化生产线的最大流程时间为目标,根据相同品种子集的相对相似系数进行分组,采用主动禁忌搜索算法对电路板进行排序,并利用预置即需工具(Keep Tool Needed Soonest, KTNS)策略切换供料器的配置,实现柔性生产线的负载均衡优化。**Guo** 等^[95]研究了喷嘴切换和供料车在组线生产线优化中带来的瓶颈问题,设计了混合遗传算法优化多品种小批量订单的总完工时间,以遗传算法解决电路板的分组问题,提出改进的解决席位公平分配问题的增量算法进行组内分配,并采用改进的 KTNS 算法进行元件排序使切换次数最小。

表面组装线优化也可作为多层次生产规划的一部分,包括电路板-生产线分配、元件-贴片机分配和贴片机拾贴顺序优化等多个环节。**Tóth** 等^[72]研究了固定贴片机配置下多种不同批次类型电路板的整线优化问题,提出了一种迭代式的两阶段优化算法,将进化算法与整数规划结合,运用遗传算法解决生产线的重配置问题,并以整数规划模型优化整线负载平衡。更进一步,**Tóth** 等^[96]研究了整线贴装多种电路板的多模态负载均衡问题,提出了基于图论的电路板拆分组启发式算法,将多种电路板组装问题拆分为独立的生产问题、并配置模块的公共子配置,有效地缩减了问题的规模。**Mumtaz** 等^[97]将多模态组线优化问题转换为生产线-电路板分配、元件-贴片机分配和贴片机元件拾贴顺序三个子问题,在预定交期和不同的产线容量条件下,提出了混合蜘蛛猴算法来同时解决多层次规划问题。**Koskinen** 等^[98]研究了分层启发式解决作业分区、选择、分组、负载平衡和调度问题,以最小化多周期生产计划中工作的总延误时间为目标,提出了包括生产线任务分配初始化和生产计划步骤改进的两阶段启发式,用以优化多条组装线下的多种类型电路板的滚动水平生产计划。**Pan** 等^[99]研究了分布式并行的生产规划问题,以最小化总能耗和总延时为目标,提出了基于知识的双种群优化算法,通过分析问题特性,在种群初始化中使用了两种基于特定问题知识的启发式算法和一种随机启发式算法,运用非有序排序遗传算法和差分进化算法在种群上并行执行,以基于知识的局部搜索算子迭代,提升了机器的利用率。**Chen** 等^[100]分层设计分别对生产线配置、贴片机分配和元件分

配进行优化,提出了进化算法和数学规划结合的数学启发式算法,在进化算法迭代过程中以数学模型评价个体的性能指标,确定了生产线中贴装机类型的最优配置。

1.3.2.3 组装过程时间估计

组装时间估计是表面组装生产线中优化元件分配、实现贴片机工作负载均衡的重要依据。组装过程的时间受贴片机类型、优化算法及生产数据特性的影响。相关研究通常采用(线性)拟合的方法进行时间估计。**Wu**等^[78]以元件类型数、贴装点数、覆盖全部贴装点的最小区域面积及其乘积项的平方根为拟合项的线性回归估计组装时间。**Vainio**等^[101]分别研究了基于制造商参数和回归模型的组装时间估计方法,前者采用最近邻启发式快速优化组装过程,以给定参数模拟组装过程估计组装时间,后者则以元件类型、吸嘴类型及其数量等参数构成特征集,在训练阶段通过贪心的特征选择算法,通过正则化最小二乘回归对组装时间快速估计。进一步地,**Vainio**等^[102]研究了不同类型贴片机的非线性拟合方法,以元件类型数、贴装点数、电路板尺寸、元件形状及其周围的最小矩形面积等为训练特征,提出了基于多层感知器网络的组装时间估计器,采用交叉验证和 **Bayesian** 正则化等技术避免模型的过拟合,提高准确度。监督学习具有较强的非线性拟合能力,在时间估计中有更高的准确度。**Li**等^[103]提出了基于共生生物搜索的支持向量回归组装时间估计方法,分析了装配过程中反映电路板和组装线生产时间不确定性的特征,通过特征选择、正则化编码、数据缩放和异常值去除等技术,结合集成学习等技术对组装时间进行估计。**Chen**等^[104]将学习算法与启发式算法相结合,以随机森林算法预处理数据的特征,结合了电路板特征、操作员习惯和程序设置等,以差分进化算法为基础,提出了基于决策树的改进轻量级梯度提升机算法,用于估计表面组装的生产准备时间。

1.4 现有研究存在的问题

表面组装过程优化任务,作为一类典型的非确定性多项式(Non-deterministic Polynomial, NP)难组合优化问题^[105],直接对其整体进行求解的算法搜索效率低下,且容易陷入局部最优解。现有的主流研究将贴片机的表面组装过程优化分解为供料器槽位分配问题和元件贴装顺序问题。前者可以被视为二次分配问题,后者则可被转换为路径规划问题。对于更高层次的表面组装生产线优化问题,已有研究对负载平衡算法展开了初步的讨论与分析。然而,由于该问题解空间庞大,加之准确获取贴片机的组装时间困难,制约了生产线效率的进一步

提升。目前关于表面组装过程优化方法的研究尚存在诸多问题，主要有：

1) 组装过程数学模型的目标和约束不完备、或不具备可解性

相关研究已经探讨了表面组装过程数学模型的构建策略。然而，目前已有研究中构建的模型普遍存在优化目标不全面、约束条件不完备或模型自身非线性不可解等问题。影响并列式贴片机组装效率的因素较多，在相关研究中，文献[5-6]在构建模型时，仅将贴片头的移动路径长度作为评价指标，却忽视了拾取动作、吸嘴更换动作等关键因素对组装效率产生的影响。文献[16-18]分别从拾贴周期规划、吸嘴更换策略以及同步拾取机制等角度对模型构建进行了讨论。其中，文献[16]以周期数和悬臂移动速度为优化目标；文献[17]进一步优化了周期数、悬臂移动速度和吸嘴更换次数；文献[18]通过构建同时拾取组，规避了吸嘴更换动作。上述研究中对组装效率影响因素考虑仍不全面，导致大幅降低了模型解的组装效率。文献[6, 12, 29, 47]虽然为表面组装全过程模型建立了相对完备的约束条件和优化目标，适用于描述复杂问题和验证解的可行性，但模型本身的非线性特性却使其无法直接求解，也无法根据模型的解来定量分析启发式算法的效果。文献[10, 15]讨论了贴装过程的路径规划建模，但由于模型复杂度过高，其适用范围仅限于极小规模的数据，在工程中难以部署和应用。此外，部分研究仅针对特定应用场景[8]或其中的子问题[10, 13-15]进行了模型构建，缺少了从整体上对组装过程效率指标进行优化的综合考量。

2) 贴片机组装过程优化解的生产效率不高、适用场景有限

表面组装过程的优化任务，因其多模态、高维度和复杂约束的特性，促使当前主流研究普遍倾向于采用启发式作为算法设计的主体框架。文献[18, 26, 29, 31, 33, 37, 40-41, 60]均是在元启发式框架下，对表面组装过程优化进行了研究。元启发式算法在处理大规模组装优化任务时，其收敛速度可能会因问题规模的增大而减缓，进而影响了优化解的质量。算法中的随机搜索策略虽然有助于拓宽搜索范围、增强算法的适应性，但同时也带来了优化结果的不确定性，易使结果陷入局部最优解，从而降低了组装效率。在并列式贴片机组装过程的优化研究中，当前研究通常关注于优化目标和约束条件的某个或某些方面，而对于整个组装过程中关键优化目标的综合分析并不全面。特别是在决定组装效率的关键因素——同步元件拾取方面，仅有文献[18-19, 29-30, 47, 64-65]对此进行了研究。同步拾取动作受到贴片头上所安装吸嘴类型的限制，为实现同步拾取而更换吸嘴，或为满足元件与吸嘴类型的兼容而进行多次拾取，均会直接导致组装效率的降低。上述研究中，文献[18, 65]未对影响效率的供料器分配槽位予以讨论；文献[19]采用的贪心同步拾取策略会增加吸嘴更换的次数；文献[29]以最

小化拾贴周期数为主要优化目标,在处理包含较多类型元件的优化问题时,会导致频繁的吸嘴更换,进而降低整体效率;文献[18, 64]在设计同时拾取组时禁止了吸嘴更换动作,限制了组装效率解的质量;文献[30]在构造同步拾取供料器分配时,未充分考虑元件对应的吸嘴类型,导致实际组装效率下降。相关研究中,文献[47]在算法设计过程中较为全面地讨论了约束条件和优化目标,但其对贴片头上分配的元件、拾取的槽位以及吸嘴类型排列顺序的研究仍显不足,限制了算法效率的进一步提升。

在表面组装过程优化搜索解的过程中,机械约束与工具约束等限制了解的可行性。其中,机械约束包括各贴片头存在物理上无法到达的拾取槽位,为解决此问题,文献[18]与文献[19]分别采用了构造同步拾取组与贪心扫描算法的策略;而在文献[30, 47]等关于供料器位置分配的研究中,若贴片头无法达到所分配的槽位拾取元件,则必须调整其拾取的槽位或所用贴片头,均会降低组装效率。在工具约束层面,文献[62]假定各类元件仅有1个供料器,且供料器仅能由指定贴片头拾取元件,限制了组装效率的提升。此外,不同型号的供料器被设计用于供应不同尺寸的元件,文献[18, 62]中供料器仅占用一个槽位的设定,无法满足大尺寸元件的拾贴作业需求。文献[29, 47]中提出的算法需要对供料器的安装槽位重新进行分配,在小批量多品种、且供料器更换频率较低的生产场景下,增加了生产准备的时间成本。对于可用吸嘴工具的约束,文献[18, 64, 79, 106]的研究中不进行吸嘴更换的设定,使其在处理吸嘴类型较多的组装数据优化时,可能导致所得解的不合理甚至无解的情况出现。文献[17, 60]将元件与吸嘴的匹配问题转化为代价函数进行优化,由于在实际组装过程中,不合理的贴片头吸嘴配置对拾取稳定性和贴装准确性均会产生不利影响,因此其所得解的无法保证可部署到实际的生产环境中。

3) 生产线的组装用时评估不准确、负载分配不平衡导致组装效率低

为提高电路板组装流程的效率,制造商通常以多台贴片机构成的表面组装生产线来执行组装作业,此过程中涉及到生产线的负载平衡问题。表面组装生产线的负载平衡优化需要综合考虑供料器配置、吸嘴分配、以及元件分配等多个子问题,以提供更全面的组装生产线优化解决方案。多层次的决策规划和子问题之间的相互耦合等多方面因素的共同作用导致现有优化算法对组装效率的提升有限。表面组装生产线优化研究中,元件分配作为核心任务,其负载平衡算法已成为主流研究的重点。文献[73-75]研究了数学规划法解决中小规模数据的元件分配策略,文献[77, 79, 81, 81, 83, 85-86]则研究了基于启发式的元件分配算法,以满足更大规模数据的优化需求。组装生产线优化还需解决同类元

件在多台贴片机间的分配问题以及可用工具的约束条件。文献[77]等限制各类元件仅能分配至单一贴片机，未能充分利用元件的可重复分配性以提升组装效率；文献[74-75, 79, 81, 86]研究了将同类元件分配至不同贴片机进行组装的优化策略，但仅确定了各类型元件在不同机器上的分配数量，未涉及吸嘴、供料器等组装工具的分配问题，也未明确贴装点与贴片机的分配关系。解决方案不完整将导致组装线的效率过度依赖于贴片机的优化效果，可能造成贴片机间工作负载分配的不均衡。文献[81, 83]运用元启发式算法同时对组装线的元件分配和贴片机的拾贴顺序进行优化，较大的复杂解空间使得算法难以在可接受的时间内获得高质量的解。负载均衡优化相关研究主要围绕转塔式贴片机展开，直接推广到并列式贴片机构成的生产线中将会导致组装效率的下降。

为降低因引入贴片机优化所导致的组装线负载均衡问题复杂程度和求解时间的增加，文献[78, 80, 101-104]等围绕表面组装时间估计开展研究。其中，文献[78, 80]将贴装点数和组装时间建立关联，其估计结果与实际组装时间之间存在较大的偏差；文献[101]研究了基于线性回归的时间估计模型，同样无法准确估计并列式贴片机复杂组装过程的用时；文献[102-104]研究了基于机器学习的非线性组装时间估计方法，此类研究较线性拟合在准确度上有了显著的提升。对于时间估计模型，特征的选择是决定估计准确度的关键因素，有效的特征能够减少估计模型的训练时间和计算成本，提高模型的泛化能力。总体而言，当前关于组装时间的估计主要集中于转塔式贴片机，而对于以平衡工作负载为目标的并列式贴片机生产线优化问题，现有研究在分配过程和组装时间估计时未能全面考虑影响组装时间的多种因素，进而引发了负载分配的不平衡，降低了组装线的整体效率。

1.5 本文主要研究内容及安排

本文致力于解决表面组装过程中生产调度优化问题，包括贴片机的组装过程优化和组装生产线的负载均衡优化。针对现有研究中模型建立不完备、组装优化算法适用场景受限、组装过程生产效率低等诸多问题展开研究，旨在提高优化算法的可应用性、可部署性和可拓展性，提升表面组装生产过程的效率。对于复杂的表面组装过程优化，全局直接优化方法易受问题的高维计算复杂性制约，并可能陷入局部最优。本文采用分层优化策略，基于从局部到整体的分阶段降维框架：聚焦核心贴片头任务分配问题的优先求解，再逐层优化贴装过程路径规划和生产线负载均衡，通过降维有效压缩解空间维度，以提升求解效

率、逐步逼近最优解。论文的基本组成结构如图1-7所示，各章节的组织安排如下：

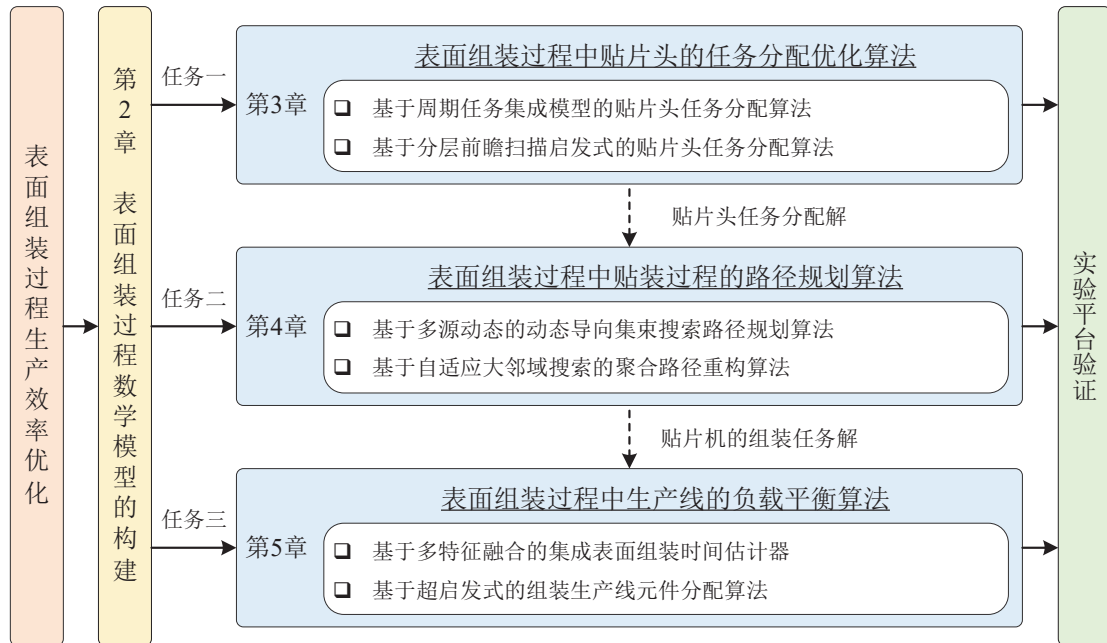


图 1-7 论文主要研究内容结构框图

Fig. 1-7 Structural block diagram of main research contents of this dissertation

第2章是全文研究的基础性工作，通过分析影响表面组装过程效率的主要因素，对组装过程的约束条件和优化目标进行了研究。贴片机的组装过程优化问题可以被拆解为拾取过程和贴装过程的规划问题，表面组装生产线是以此为基的进一步研究。结合实际应用场景，对问题进行了适当简化并提出假设条件，将组装过程优化问题拆分为贴片头任务分配、贴装过程路径规划和组装生产线负载平衡问题，并以此作为本文研究的主线任务。以（混合）整数线性规划为基础分阶段地建立了准确反映研究对象特性的数学模型，提出了适用于拾取过程的基于关键目标分解的贴片头任务分配模型、适用于贴装过程的多拾贴周期路径规划模型、以及适用于表面组装生产线的负载平衡模型。数学模型可用于验证表面组装过程解的可行性，定量分析解的质量，为优化算法的设计提供参考。最后，给出了具体的实例进行分析说明，分析了模型参数的灵敏度和复杂度。

第3章从数学规划和启发式优化算法的角度，研究了表面组装过程中的贴片头任务分配的优化方法，将问题进行进一步拆分为供料器-槽位和贴片头-元件分配优化两部分，通过保证组装过程关键性能指标的高质量，提升了表面组装过程的整体效率。两类算法适用于不同规模的数据，数学规划法适用于小规

模解空间的系统化搜索,启发式优化则适用于大规模解空间的高效搜索。在基于数学规划的算法设计中,为降低贴片头任务分配模型的复杂度、提高模型的求解效率,提出了基于周期任务集成的模型增强策略,根据问题特征设计了模型的初始解构造算法,提出了一种带有路径预判的模型解的评估策略,用于降低其与后续路径规划问题之间耦合性;在基于启发式的算法设计中,进一步提出了基于前瞻扫描启发式的贴片头任务分配算法,研究了基于长短期收益评估的元件分配策略,围绕吸嘴分配模式、供料器槽位分配和贴片头元件分配,分阶段地设计启发式算法,结合实际应用中的生产配置,对算法的实现细节进行了详细的讨论。最后,对比分析了不同模型设计的收敛效果,并将所提出算法和主流研究在组装性能指标等方面进行了比较,通过在不同应用场景下的对比验证,证明了所提算法结果的高效性、可行性和可扩展性。

第4章在第3章研究的基础上,分析了贴装过程的问题特性和分阶段优化方法,研究了基于给定的贴片头任务分配约束下贴装过程路径规划算法,将其拆分为周期内和周期间的优化算法。其中,前者是对周期内有限贴装点的移动路径进行规划,后者则是对贴装点分配的拾贴周期和贴片头进行优化。在周期内规划阶段,结合贴装过程多轴协同作业的特性,建立了拾贴周期路径规划的状态转移方程,提出了基于动态规划的拾贴周期最优路径规划算法;在周期间规划阶段,以贪心式贴装点分配为基础,结合多源贪心和动态导向的集束搜索算法,提出了周期间的贴装点分配算法。周期内和周期间相结合的路径规划算法保证了解的高质量,为进一步优化贴装过程的移动路径,提出了自适应大邻域搜索的聚合路径重构算法,根据贴装点的分布和拾贴周期的移动路径,对已有路径进行破坏和修复,克服了贪心式路径规划导致的拾贴周期间路径移动长度不均衡的问题,实现了对解的持续迭代优化。最后,将提出的路径规划方法和主流研究在相同及不同贴片头任务分配约束下进行了对比,对路径规划算法的关键组成部分的作用进行了分析验证,比较了不同初始解对自适应大邻域搜索结果的影响,证明了提出算法结果的高质量。

第5章在第3-4章研究的基础上,将贴片机组装过程的优化进一步拓展至表面组装生产线的负载平衡优化领域,旨在通过整合多台贴片机提升表面组装流程的整体效率。为优化由并列式贴片机构成的表面组装生产线,探讨了运用多特征融合集成组装时间估计器的超启发式负载平衡方法,设计了基于数据驱动与目标驱动的底层启发式算子,并提出了不同算子快速比较贴片机之间相关性能指标的方案。此外,为了以可重复分配元件提升组装效率,提出了元件分组搜索策略与聚合聚类元件贴装点分配算法。在评估解的质量方面,研究了组

装时间估计模型的训练数据的生成流程和处理方法、特征提取算法以及设计架构；在此基础上，提出了基于集成学习的组装时间估算器，该估算器以神经网络为基模型，融合了组装数据的基本参数、估算子目标等多维度特征进行编码，实现了对组装时间的准确估计。最后，将所提出的组装时间估计器与其他方法进行了对比，验证了其在准确度方面的提升；围绕解的组装效率、结果稳定性以及算法运行效率等指标，将提出的超启发式负载平衡算法同主流研究进行对比验证，证明了提出算法的高效性、优化结果的高质量和稳定性。

第2章 表面组装过程优化的数学规划模型

2.1 引言

建立准确反映研究对象特性的数学模型是分析和求解问题的首要环节。表面组装过程优化作为一类组合优化问题，数学规划法是求解此类问题的主要方法之一。以整数规划为代表的方法通过系统地搜索解空间、排除不可行解和次优解、逐步逼近最优解，在充足的求解时间下理论上能准确地获得问题的最优解。然而，研究对象模型的构建必须充分考虑影响优化目标的主要因素和约束条件，以确保模型的实用性和有效性。对于涉及多个优化变量和复杂决策过程的数学模型，其构建过程面临着约束条件非线性、过程环节间强耦合、优化目标影响因素复杂等诸多挑战，大幅增加了模型构建的难度；即使成功建立了问题的模型，也可能因过高的复杂度而难以将其直接应用到实际的生产环境中。

拆分和简化问题模型是建模过程中的常用策略，用于提升模型处理复杂数据的求解效率。模型的拆分与构建应遵循从整体到个体、从主要目标到次要目标的原则。合理地将表面组装过程中的优化问题进行拆分，聚焦于分解问题的主要优化目标，关注子模型之间的耦合关系与约束条件，可以显著降低问题求解的难度。此外，对模型做出合理假设能有效凸显主要问题、弱化次要问题、省略非关键环节，有助于模型的进一步简化。在数学规划领域，现有的通用求解方法对于线性模型展现出了高效的求解能力，因而有必要在建模过程中线性化调整模型，以确保其可解性。可解的数学模型不仅可以验证解的可行性、定量分析解的质量，还可为启发式算法的设计提供参考基准，从而为算法的设计和改进提供依据。

表面组装过程的优化旨在调度和规划组装过程中拾取、贴装等工序，其决定了电路板的组装生产效率，同时受到多重外部与内部因素的制约，加大了问题求解的难度。此类问题的模型构建过程需要侧重于影响装配效率的关键因素，以降低模型的复杂度、提高模型的求解效率。本章将研究单一品种电路板的组装过程建模问题，分析该问题的特点，确定表面组装过程的优化目标及约束条件，从贴片机的组装过程优化的角度出发，对模型进行拆解与分析，进而扩展至整个表面组装生产线负载平衡模型的建立。

2.2 系统的描述与分析

2.2.1 系统构成

本文研究的装配并列式贴片头的贴片机平台系统构成如图2-1所示，平台中主要设备的厂商、型号等信息如表2-1所示。实验平台为H型龙门结构多自由度运动平台，X轴为单轴直驱运动模组，Y轴为两套平行安装的单轴直驱运动模组，X轴同步装配有一个悬臂及多个独立贴片头。系统采用轻量化悬臂设计，结合双伺服驱动系统，支持多轴同步运动，可实现复杂轨迹的快速响应，并确保高速运动下的稳定性。内置的吸嘴更换装置可根据元器件类型自动切换吸嘴。PCB传送与定位系统可自动调整宽度，适配不同尺寸基板。双视觉协同识别系统在贴片头作业过程中实时识别元件，通过补偿机械误差，确保贴装位置与预设位置的一致性。供料系统提供120个标准供料槽，配备的电动供料器具备吸料位置自动整列功能，减少了供料卡顿。操作系统与界面基于Windows 10平台，提供直观编程界面，支持在线程序编辑与优化。

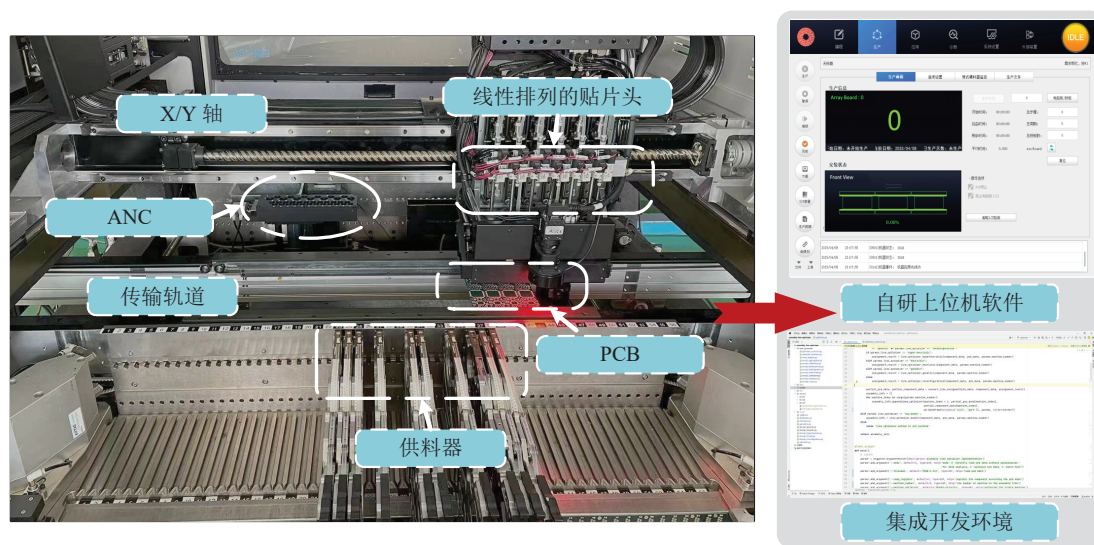


图 2-1 贴片机平台

Fig. 2-1 Platform of surface mounter

典型的表面组装生产线如图2-2所示，上板机、印刷机、贴片机和回流焊为其基本组成部分。在电子制造工艺流程中，各类设备的产能特性存在显著差异。丝网印刷机在焊膏涂覆环节展现出显著的速度优势，而回流焊接设备采用连续进板设计，其工艺节拍独立于前序工序，通常不会形成产能制约。AOI检测系统具备多工位并行检测能力，其图像处理耗时可忽略不计。与之对比的是表面组装设备，该工序需完成数百个异形元件的精密拾取与定位作业，其贴装周期

直接决定了整线生产节拍。在生产线优化实践中，设备综合效率管理的核心在于突破关键瓶颈——由于表面组装工序的固有特性，其设备产能往往构成整线产能的核心制约因素，这使得表面的效率优化成为提升整体装配效率的关键突破点。

表 2-1 表面组装设备实验平台配置
Table 2-1 Platform configuration of surface assembly equipment

系统组成	设备型号	设备厂商
工控机	IPC-610-L	研华科技
X 轴驱动器	MCDLN35SE	Panasonic（松下）
Y 轴驱动器	MDDL55SE	Panasonic（松下）
Z 轴驱动器	MMDKT2C09E	Panasonic（松下）
R 轴驱动器	ZC-SR2	鸣志
X 轴电机	MHMF082L1U2M	Panasonic（松下）
Y 轴电机	MSMF102L1G6M	Panasonic（松下）
Z 轴电机	MNMA2ACF1A	Panasonic（松下）
R 轴电机	AM11HS3007-02	鸣志

本文相关的实验均采用 Python 3.11 在集成开发环境 PyCharm 2022.1.4 软件中实现，相关的数学模型则采用优化器 Gurobi 11.0^[107] 进行求解，算法均在配置为 Intel Core i5-14600KF 主频 3.5GHz CPU 的计算机上实现。程序优化结果包含供料装置安装槽位、ANC 库配置、元件的拾取和贴装顺序规划等，通过标准化文本格式文件导入到实验平台中。实验平台内置生产过程控制程序和统计程序：控制程序中的同步拾取和多轴运动等动作按最优条件执行，并可根据程序步骤实现贴片头自动更换吸嘴、在供料基座上拾取元件和前往预定贴装点放置元件等组装过程运动指令；统计程序可准确记录电路板传送时间、贴装时间等



图 2-2 表面组装生产线的系统构成

Fig. 2-2 System configuration of surface assembly line

相关生产数据，其中贴装时间为开始准备生产到贴装完最后一个元件的用时，包含了贴装第一个元件前的贴片头吸嘴更换等过程，是后续对比实验所用的数据之一。

2.2.2 表面组装过程优化

表面组装过程的优化可划分为贴片机组装过程优化与生产线负载平衡优化两个方面。前者用于解决贴片机在组装过程中悬臂移动用时与组装动作用时总和的最小化问题；而后者则用于平衡生产线上贴片机的负载，减少其中作为生产效率提升瓶颈的贴片机的组装用时。贴片机组装过程优化的结果用以评估生产线负载分配的平衡性，而组装线负载平衡的结果又是贴片机组装过程优化的输入。尽管贴片机组装过程优化可被视为生产线负载平衡优化的一个组成部分，但由于其模型的构建本身已有较高的复杂度，若直接在生产线负载平衡优化中对多个决策过程进行建模，将大幅度增加模型复杂度，导致模型求解变得异常困难、甚至不可解。在实施优化策略时，需要综合考虑两者的特点与关系，采取适当的策略与方法进行分别优化与处理。

贴片机的组装过程优化可被视为仓库选址、任务分配和路径规划相结合的综合组合优化问题。在电路板组装过程中，贴片头需从装载有电子元件的供料器中拾取元件，供料器在基座上的安装槽位可以等效为仓库选址问题。任务分配问题需要确定贴片头在各个拾取周期内取料的槽位，槽位的分配直接影响元件拾取的次数，进而影响了组装效率。同时，不同元件类型所需的吸嘴类型决定了贴片头的吸嘴更换的次数，影响了悬臂移动过程的用时。路径规划用于优化悬臂上贴片头贴装元件的路径，同时确保元件贴装位置与贴片头拾取元件类型相匹配。不同子问题之间紧密耦合、相互影响，共同决定了表面组装过程的整体性能。

表面组装生产线负载平衡指的是解决元件在不同贴片机的分配问题。分配给不同贴片机的元件类型、元件的贴装点数都会影响贴片机的组装效率。在元件分配过程中，供料器的可用数量会限制不同类型元件可分配的贴片机数量。通常情况下，元件分配的组合数量庞大，即使是小规模数据也需要大量的计算资源，且随着问题规模的扩大，搜索过程的运算量也会迅速增加。因此，提高对复杂解空间的搜索效率对组装线负载平衡优化至关重要。在确定装配时间时，贴片机组装过程用时已难以直接获得，将贴片机组装优化问题直接扩展到生产线负载平衡问题因复杂度较高更不具备可行性。在生产线元件分配过程中，通过提取组装效率相关的关键因素进行建模和优化，以及提取特征估计组装过程

的用时，有利于降低问题求解的难度，实现对解空间的快速迭代搜索，进而获得高质量的解。

2.3 多阶段模型的构建

2.3.1 分析与假设

贴片机组装过程优化中的主要决策子问题、优化子目标和约束条件之间的关系如图2-3所示。贴片机的组装过程优化可以被拆解为拾取过程和贴装过程的优化问题。前者解决了供料器的安装槽位和贴片头拾取元件类型的问题，而后者则决定了贴片头的贴装顺序问题。这两个子问题之间存在着递进关系，即在确定贴片头的组装元件类型基础上，进一步分配对应的贴装点。分层递进求解可以逐步缩减搜索空间的大小，降低问题的计算复杂度。表面组装过程的各环节存在相关性，供料器的安装槽位影响贴片头的拾取位置和分配的元件类型，决定了拾取过程的移动路径；贴片头分配的元件类型影响着贴装过程的移动路径，同步拾取元件所必要的吸嘴更换带来悬臂往返 ANC 的移动过程。为了提升表面组装效率，同时保证贴片头的吸嘴类型、元件类型、贴装点和拾取槽位的一致性，可能会产生贴片头的吸嘴更换和拾取元件过程的移动动作。供料器的安装槽位和贴片头的元件分配共同决定了组装过程中悬臂的整体移动路径。

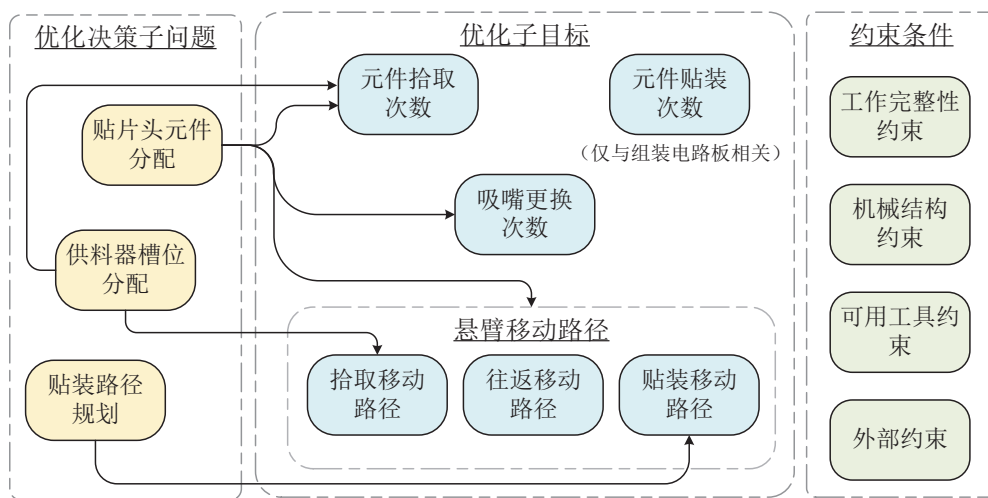


图 2-3 表面组装优化中子目标、共同约束条件和子问题之间的关系

Fig. 2-3 The relationship among sub-objectives, constraints and sub-problems.

表面组装过程的目标是最小化组装时间，该指标与组装过程多项子目标相关，子目标可分为组装动作指标和移动路径指标。组装动作指标包括贴片头的元件拾取次数、吸嘴更换次数和元件贴装次数，其中元件贴装次数仅和组装数据相关，而元件拾取次数和吸嘴更换次数则取决于优化结果，同样是影响并列

式贴片机整体组装效率的重要因素^[47]。悬臂的移动路径又可进一步分为拾取移动路径、往返移动路径和贴装移动路径，其中，往返移动路径指悬臂在电路板和供料器基座之间的往复运动。在移动指标中，拾取移动指标和元件拾取次数关联度较大，贴装过程的移动指标对整体组装效率影响较小，往返移动路径则同拾贴周期数直接相关。

在表面组装过程中，多个优化子目标均会影响整体组装过程效率，且不同性能指标彼此之间存在关联性又相互冲突。对于影响组装效率的主要因素——元件拾取次数和吸嘴更换次数，构造同步拾取可能会带来贴片头冗余的吸嘴更换动作，减少吸嘴更换次数又可能会导致拾取路径和拾取次数的增加，优化算法须在二者之间权衡。贴片头的移动路径规划之间也存在相关性。为了保证多个贴片头从预定槽位同步完成拾取动作，其需要根据实际情况调整吸嘴类型以满足拾取要求，过多的吸嘴更换则需要悬臂频繁往返于 ANC，增加了移动路径的长度。为了减少悬臂在电路板和供料器基座之间的移动路径，贴片头任务分配需要保证各拾贴周期尽可能多地拾取元件，其将会间接导致拾取移动路径增加，同时可能减少同步拾取的贴片头数、增加吸嘴更换，进而降低拾取过程的效率。

表面组装过程优化算法不仅要实现对不同元件类型及贴装点数的 PCB 数据的组装任务优化，同时要考虑实际生产过程中的约束条件。相关约束可被分为四类：

1) 工作完整性约束：指所有元件必须准确地装配在相应的 PCB 焊盘上，是表面组装任务的基本要求；

2) 机械结构约束：指贴片机组成构件之间的相互作用和限制关系，如各贴片头均无法到达供料器槽位，供料器宽度不同导致的安插槽位位置干涉等；

3) 可用工具约束：指吸嘴、供料器的数量和一致性约束，数量约束为相关可用工具数量的有限性，一致性约束为吸嘴类型需要和元件类型匹配，以保证拾取过程的顺利进行。不合适的元件-吸嘴类型组合将导致抓取失败、废料率高等问题，会进一步降低组装效率；

4) 外部约束：指现场操作人员根据实际情况额外增加的限制，如调整部分供料器安装的槽位、禁用贴片头或槽位、更改供料器可用数目、指定贴片头所用吸嘴类型等，此类约束通常是根据机器状况所进行的适当调整，是保证组装过程顺利进行的必要条件。

表面组装生产线是对贴片机优化问题的进一步研究，其通过优化组装时间最长的贴片机实现整条生产线性能的提升。组装线的负载分配是指不同类型元

件及其贴装点组装任务的调度。对表面组装全过程进行建模在实现上具有较高的复杂度, 前述已讨论了影响贴片机组装效率的主要因素, 而通过拾贴周期数、吸嘴更换数和元件拾贴次数近似化贴片机的生产效率具有较高的准确度, 可以实现组装线上贴片机负载平衡。影响组装线调度的两种主要限制是工具限制和机器限制, 工具限制指的是供料器、吸嘴等可用组装工具数量的有限性, 而机器限制指的是不同类型的专用贴片机在生产线上协同工作时, 必须由特定贴片机组装指定类型的元件。此外, 在表面组装过程中, 小尺寸或高度较低的元件通常被优先贴装, 以保证后续组装过程中不会出现元件相撞的问题。

结合实际应用条件, 适当简化问题有助于降低问题求解的难度, 同时保证了模型解的可行性。本文对表面组装过程优化问题做出如下假设:

- 1) X 轴和 Y 轴电机的运动控制相互独立, 悬臂移动用时仅和两个方向上的 Chebyshev 移动距离相关。
- 2) 相邻贴片头之间与相邻供料器槽位之间的间隔距离比为整数。
- 3) 只有特定类型的吸嘴才能拾取对应类型的元件, 元件类型和吸嘴类型的匹配性是已知的。
- 4) ANC 的配置是预先确定的, 即各类型吸嘴的可用数量是已知的; 更换吸嘴过程中, 悬臂在不同吸嘴孔间的运动时间被忽略不计。
- 5) 所有待贴装元件均由带式供料器供给, 其他类型供料器的拾取位置不在本文研究之列。

基于此, 本文研究了表面组装过程分阶段的(混合)整数规划模型, 模型可按贴片机和组装生产线分别进行建模。其中, 贴片机模型可被进一步拆分为贴片头任务分配模型和贴装过程路径规划模型, 前者以组装过程的关键效率指标为优化目标, 旨在解决组装过程中贴片头拾取的槽位、组装的元件类型和装配的吸嘴类型, 后者则基于已有的贴片头任务, 对元件贴装过程路径进行规划。进一步地, 生产线负载平衡优化是基于关键指标拟合的组装效率而构建的最小-最大组装线贴片机组装用时的模型。

2.3.2 贴片头的任务分配模型

贴片头任务分配模型仅考虑单台贴片机的分配结果, 即贴片机索引集 $M = \{1\}$, 其优化目标 (2-1) 为最小化拾取过程相关的主要性能指标的加权和, 包括拾贴周期数、吸嘴更换数、拾取次数和拾取移动距离四项, 其中的拾取移动距离由贴片头拾取元件过程经过的槽位数表示。

$$\min T^{\text{CY}} \cdot \sum_{k \in K} g_{k1} + T^{\text{NZ}} \cdot \sum_{k \in K} \sum_{h \in H} n_{kh1} + T^{\text{PU}} \cdot \sum_{s \in S} \sum_{k \in K} e_{sk1} + T^{\text{PM}} \cdot \sum_{k \in K} d_{k1} \quad (2-1)$$

约束 (2-2) 用于确保贴片头的元件分配结果和拾贴周期规划结果的一致性, 由于 g_{km} 是 0-1 变量, 其同时限制了贴片头在各拾贴周期至多被分配一类元件。

$$\sum_{i \in I} x_{ikhm} \leq g_{km} \quad \forall k \in K, h \in H, m \in M \quad (2-2)$$

约束 (2-3) 用于确定贴片头分配的吸嘴类型。贴片头上的吸嘴装配与其拾取元件之间不存在必然的关联性, 即贴片头装配吸嘴是拾取元件的前提, 但并不决定拾取元件动作的执行与否。约束 (2-4) 用于计算贴片头 h 在拾贴周期 k 和 $k+1$ 之间是否发生吸嘴更换, 约束 (2-5) 则用于计算当前电路板的最后一个拾贴周期到下一个电路板首个拾贴周期之间是否发生吸嘴更换, 以提升连续拾贴作业的组装效率。

$$x_{ikhm} \leq \sum_{j \in J} \mu_{ij} \cdot z_{jkhm} \quad \forall i \in I, k \in K, h \in H, m \in M \quad (2-3)$$

$$n_{khm} = \sum_{j \in J} (|z_{jkhm} - z_{j(k+1)hm}|) \quad \forall k \in K \setminus \{|K|\}, h \in H, m \in M \quad (2-4)$$

$$n_{|K|hm} = \sum_{j \in J} (|z_{j|K|hm} - z_{j1hm}|) \quad \forall h \in H, m \in M \quad (2-5)$$

约束 (2-4) 可以被线性化为 (2-6)^[17], 通过引入两个非负整数项 $n_{j'k h m}^+$ 和 $n_{j'k h m}^-$, 有

$$\begin{cases} n_{k h m} = \frac{1}{2} \sum_{j' \in J} (n_{j'k h m}^+ + n_{j'k h m}^-), \\ z_{jk h m} - z_{j(k+1) h m} = n_{j'k h m}^+ - n_{j'k h m}^-, \quad \forall j \in J, k \in K \setminus \{|K|\}, h \in H, m \in M \\ n_{j'k h m}^+ \in \mathbb{N}_0, n_{j'k h m}^- \in \mathbb{N}_0 \end{cases} \quad (2-6)$$

其中 \mathbb{N}_0 为非负整数集, 同理, 约束 (2-5) 可以被进一步线性化处理。

并列式贴片机的拾取过程如图2-4所示, 为计算同步拾取次数, 本章将各贴片头拾取元件的槽位转换为最左侧的贴片头所对齐的槽位, 并称之为等效槽位。约束 (2-7) 用以计算各个拾贴周期中槽位 s 是否为等效槽位, 拾贴周期 k 内的等效槽位数即为该周期的元件 (同步) 拾取次数。

$$e_{skm} \leq \sum_{h \in H_s} y_{[s+(h-1) \cdot \tau]k h m} \leq N \cdot e_{skm} \quad \forall s \in S, k \in K, m \in M \quad (2-7)$$

其中, 贴片头集合 H_s 是 H 的子集, 表示能从槽位 s 中拾取元件的贴片头索引集合, 其计算方式为

$$H_s = \left\{ \max\left(1, |H| - \left\lfloor \frac{|S| - s}{\tau} \right\rfloor\right), \dots, \min\left(|H|, \left\lceil \frac{s}{\tau} \right\rceil\right) \right\}$$

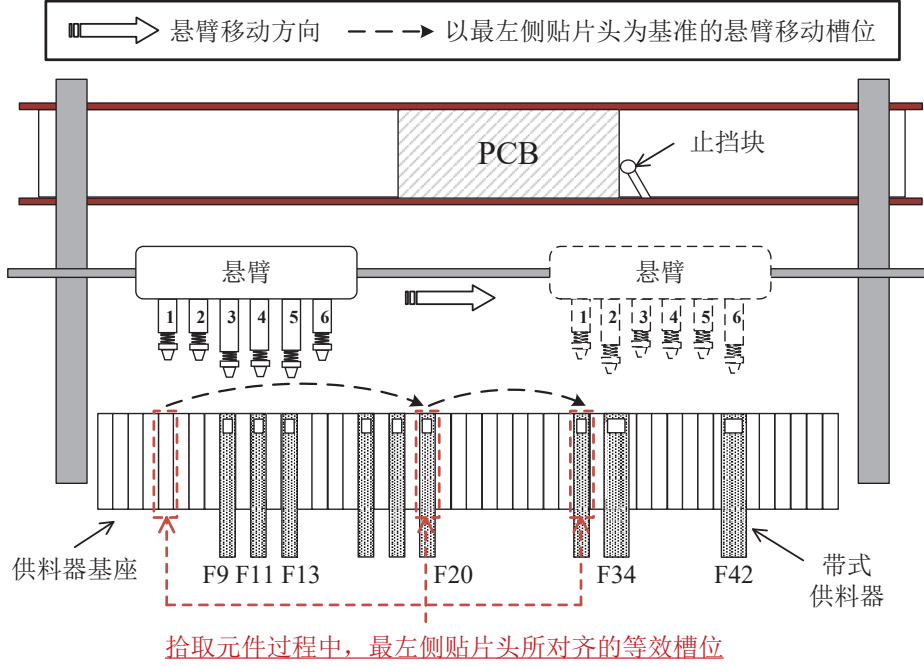


图 2-4 拾取过程示意图

Fig.2-4 Schematic diagram of the pick-up process

约束 (2-8) 根据各拾贴周期中等效槽位的位置，计算贴片头拾取过程中悬臂移动经过的槽位数。

$$d_{km} \geq s \cdot e_{skm} - s' \cdot e_{s'km} + N \cdot (e_{skm} + e_{s'km} - 2) \quad \forall k \in K, s \in S, s' \in S, m \in M \quad (2-8)$$

约束 (2-9) 用以确定贴片头拾取的槽位、分配的元件类型以及元件供料器所分配的槽位之间结果的一致性。

$$f_{ism} \leq \sum_{k \in K} \sum_{h \in H} x_{ikhm} \cdot y_{skhm} \leq N \cdot f_{ism} \quad \forall i \in I, s \in S, m \in M \quad (2-9)$$

其中的非线性项为两个 0-1 变量的乘积，可以用 0-1 中间变量 $\gamma_{iskhm} = x_{ikhm} \cdot y_{skhm}$ 线性化处理，即

$$\begin{cases} \gamma_{iskhm} \leq x_{ikhm}, \\ \gamma_{iskhm} \leq y_{skhm}, \\ \gamma_{iskhm} \geq x_{ikhm} + y_{skhm} - 1, \end{cases} \quad \forall i \in I, s \in S, k \in K, h \in H, m \in M \quad (2-10)$$

约束 (2-11) 为工作完整性约束，即各类型元件的所有贴装点均被分配到给定拾贴周期的贴片头上。

$$\sum_{k \in K} \sum_{h \in H} \sum_{m \in M} x_{ikhm} = \psi_i \quad \forall i \in I \quad (2-11)$$

约束 (2-12)–(2-13) 分别为机械结构中的贴片头左限位和右限位约束，即各贴片头存在物理上无法到达的取料槽位。

$$\sum_{s \in S} [s - (h - 1) \cdot \tau] \cdot y_{skhm} \geq \sum_{s \in S} y_{skhm} \quad \forall k \in K, h \in H, m \in M \quad (2-12)$$

$$\sum_{s \in S} [s + (|H| - h) \cdot \tau] \cdot y_{skhm} \leq |S| \quad \forall k \in K, h \in H, m \in M \quad (2-13)$$

组装过程可用工具的有限性体现在约束 (2-14)–(2-16)。约束 (2-14) 确保各供料器槽位至多分配一类元件，约束 (2-15) 和约束 (2-16) 则分别表示有限的可用吸嘴和可用供料器。

$$\sum_{i \in I} f_{ism} \leq 1 \quad \forall s \in S, m \in M \quad (2-14)$$

$$\sum_{h \in H} z_{jkhm} \leq \zeta_j \quad \forall j \in J, k \in K, m \in M \quad (2-15)$$

$$\sum_{s \in S} f_{ism} \leq \phi_i \quad \forall i \in I, m \in M \quad (2-16)$$

贴片头分配的元件类型决定了其拾取的供料器槽位，约束 (2-17) 确定了拾取槽位与元件分配结果之间的关系。

$$\sum_{s \in S} y_{skhm} = \sum_{i \in I} x_{ikhm} \quad \forall k \in K, h \in H, m \in M \quad (2-17)$$

约束 (2-18)–(2-21) 确定了决策变量的取值范围。

$$g_{km} \in \{0, 1\}, d_{km} \in \mathbb{N}_0, n_{khm} \in \{0, 1\} \quad \forall k \in K, h \in H, m \in M \quad (2-18)$$

$$e_{skm} \in \{0, 1\}, f_{ism} \in \{0, 1\} \quad \forall k \in K, s \in S, i \in I, m \in M \quad (2-19)$$

$$\gamma_{iskhm} \in \{0, 1\}, x_{ikhm} \in \{0, 1\} \quad \forall i \in I, s \in S, k \in K, h \in H, m \in M \quad (2-20)$$

$$y_{skhm} \in \{0, 1\}, z_{jkhm} \in \{0, 1\} \quad \forall s \in S, j \in J, k \in K, h \in H, m \in M \quad (2-21)$$

优化目标 (2-1) 和约束条件 (2-2)–(2-5), (2-7)–(2-9), (2-11)–(2-21) 共同构成了贴片头的任务分配模型。

2.3.3 贴装过程的路径规划模型

贴装过程路径规划问题可以被视为受贴片头任务分配结果约束的路径规划问题，模型中的参数根据2.3.2节中贴片头分配的元件类型和拾取的供料器槽位计算得到，并沿用了任务规划模型的索引、集合、参数和决策变量等符号。为降低模型的规模，本小节将贴装过程中连续作业的贴片头构成的有序对称为移动弧，移动弧对应贴装过程中的一段移动路径。以 $a = (h, h') \in A$ 为例， $w_{pp'ka} = 1$ 表示悬臂上的贴片头 h 完成点 p 的贴装后，用贴片头 h' 完成点 p' 的贴装，其中， h 被称为先序作业头， h' 被称为后序作业头。 A_h , A_h^{FRM} 和 A_h^{TO} 是 A 的子集， A_h 表示有序对中有贴片头 h 的弧集， A_h^{FRM} 表示贴片头 h 作为先序作业头的弧集合， A_h^{TO} 表示贴片头 h 作为后序作业头的弧集合。图2-5展示了移动弧 $a = (2, 3)$ 的贴装过程，其中先序贴片头为头2，后序贴片头为头3，弧 a 同时是集合 A_2 、 A_3 、 A_2^{FRM} 和 A_3^{TO} 中的元素。

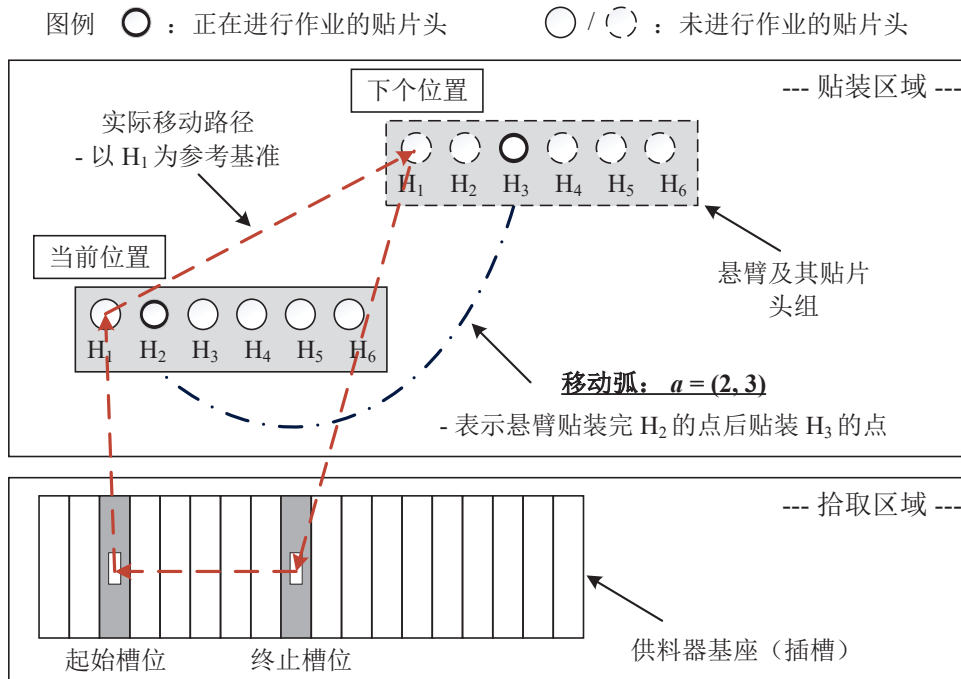


图 2-5 移动弧的演示图

Fig.2-5 Demonstration of a moving arc

表面组装过程中悬臂的移动路径长度也是影响组装效率的性能指标之一，贴片头任务分配模型已解决拾取过程移动路径，而悬臂从供料器基座到电路板间往返路径和悬臂在电路板上贴装过程中的移动路径则由贴装过程路径规划模型确定。同样地，贴片机索引 $M = \{1\}$ ，贴装过程的路径规划目标函数为 (2-22)。

$$\min \sum_{k \in K} \left\{ \sum_{p \in P} \sum_{h \in H} D_{pkhm}^{FW} \cdot u_{pkhm} + \sum_{p \in P} \sum_{p' \in P} \sum_{a \in A} D_{pp'a}^{PL} \cdot w_{pp'kam} + \sum_{p \in P} \sum_{h \in H} D_{pkhm}^{BW} \cdot v_{pkhm} \right\} \quad (2-22)$$

优化目标 (2-22) 表示贴片头的移动路径长度，共由三部分组成：从供料器基座前往电路板的移动路径长度，在电路板上进行贴装作业的移动路径长度以及从电路板返回供料器基座的移动路径长度，其中拾贴周期 k 中贴片头拾取元件的首个槽位 F_{km}^{BW} 和最后一个槽位 F_{km}^{FW} 是由贴片头的任务分配结果确定的，即

$$F_{km}^{BW} = \max_{h \in H} \{s \cdot y_{skhm} - (h-1) \cdot \tau \mid y_{skhm} > 0, s \in S\} \quad \forall k \in K, m \in M \quad (2-23)$$

$$F_{km}^{FW} = \min_{h \in H} \{s \cdot y_{skhm} - (h-1) \cdot \tau \mid y_{skhm} > 0, s \in S\} \quad \forall k \in K, m \in M \quad (2-24)$$

据此可确定贴片头拾取完最后一个元件前往电路板上的贴装点 p 的移动路径长度，以及贴装完电路板上的贴装点 p 返回下个拾贴周期首个取料位置的路径长度，即

$$\begin{cases} D_{pkhm}^{BW} = \max \left\{ \left| X^{F1} + F_{km}^{BW} \cdot \frac{\rho}{\tau} - X_p + (h-1) \cdot \rho \right|, \right. \\ \left. |Y^{F1} - Y_p| \right\} \quad \forall p \in P, k \in K, \\ D_{pkhm}^{FW} = \max \left\{ \left| X^{F1} + F_{km}^{FW} \cdot \frac{\rho}{\tau} - X_p + (h-1) \cdot \rho \right|, \right. \\ \left. |Y^{F1} - Y_p| \right\} \quad h \in H, m \in M \end{cases} \quad (2-25)$$

贴片头在电路板上进行贴装元件的移动路径长度可以根据其作业的先后顺序及对应的贴装点坐标确定，即

$$D_{pp'a}^{PL} = \max_{a=(h,h')} \{|X_p - X_{p'} - (h-h') \cdot \rho|, |Y_p - Y_{p'}|\} \quad \forall a \in A, p \in P, p' \in P \quad (2-26)$$

约束 (2-27) 为贴片头任务分配一致性约束，即贴装过程中贴片头贴装的点对应的元件类型和贴片头任务分配的元件类型相同。

$$\sum_{p' \in P} \left(\sum_{a \in A_h^{FRM}} w_{pp'kam} + \sum_{a \in A_h^{TO}} w_{p'pkam} \right) + u_{pkhm} + v_{pkhm} \leq 2 \cdot \sum_{i \in I} \eta_{ip} \cdot x_{ikhm} \quad (2-27)$$

$$\forall p \in P, k \in K, h \in H, m \in M$$

约束 (2-28) 为拾取过程移动路径的确定性约束，即各贴片头在从供料基座前往电路板、在电路板上贴装以及从电路板返回供料基座的过程中，至多经过一个贴装点，对应贴在贴装点上为点的出度和入度之和不大于 2。

$$\sum_{p \in P} \sum_{p' \in P} \sum_{a \in A_h} w_{pp'kam} + \sum_{p \in P} (u_{pkhm} + v_{pkhm}) \leq 2 \quad \forall k \in K, h \in H, m \in M \quad (2-28)$$

约束 (2-29)–(2-31) 为贴装任务的连续性约束，即对于同一个贴装点，表示贴装先后顺序的弧对应的贴片头是一致的，以保证贴装过程移动路径的连续性。

$$\sum_{p' \in P} \sum_{a \in A_h^{\text{TO}}} w_{p'pkam} + u_{pkhm} = \sum_{p' \in P} \sum_{a \in A_h^{\text{FRM}}} w_{pp'kam} + v_{pkhm} \quad (2-29)$$

$$\forall k \in K, h \in H, p \in P, m \in M$$

$$u_{pkhm} \leq \sum_{p' \in P} \sum_{a \in A_h^{\text{FRM}}} w_{pp'kam} + v_{pkhm} \quad \forall k \in K, h \in H, p \in P, m \in M \quad (2-30)$$

$$v_{pkhm} \leq \sum_{p' \in P} \sum_{a \in A_h^{\text{TO}}} w_{p'pkam} + u_{pkhm} \quad \forall k \in K, h \in H, p \in P, m \in M \quad (2-31)$$

约束 (2-32) 和 (2-33) 表示悬臂从供料器基座到电路板，以及从电路板回到供料器基座的移动路径在各个拾贴周期都是唯一的，其中的唯一性包括贴片头和贴装点的唯一性。

$$\sum_{p \in P} \sum_{h \in H} u_{pkhm} = 1 \quad \forall k \in K, m \in M \quad (2-32)$$

$$\sum_{p \in P} \sum_{h \in H} v_{pkhm} = 1 \quad \forall k \in K, m \in M \quad (2-33)$$

约束 (2-34) 和 (2-35) 分别表示贴片头前往和离开各个贴装点的移动路径是唯一的。

$$\sum_{k \in K} \left(\sum_{h \in H} u_{pkhm} + \sum_{p' \in P} \sum_{a \in A} w_{pp'kam} \right) = 1 \quad \forall p \in P, m \in M \quad (2-34)$$

$$\sum_{k \in K} \left(\sum_{h \in H} v_{pkhm} + \sum_{p' \in P} \sum_{a \in A} w_{p'pkam} \right) = 1 \quad \forall p \in P, m \in M \quad (2-35)$$

约束 (2-36)–(2-39) 基于 Dantzig-Fulkerson-Johnson 的形式，通过引入非负连续变量 $\hat{w}_{pp'm}$ ， \hat{u}_{pm} 和 \hat{v}_{pm} 消除各拾贴周期的闭合子回路。

$$\hat{v}_{pm} + \sum_{p' \in P} \hat{w}_{pp'm} - \hat{u}_{pm} - \sum_{p' \in P} \hat{w}_{p'pm} = 1 \quad \forall p \in P, m \in M \quad (2-36)$$

$$\hat{w}_{pp'm} \leq \sum_{k \in K} \sum_{a \in A} (|P| - |K| + 1) \cdot w_{pp'kam} \quad \forall p \in P, p' \in P, m \in M \quad (2-37)$$

$$\hat{u}_{pm} \leq \sum_{k \in K} \sum_{h \in H} (|P| - |K| + 1) \cdot u_{pkhm} \quad \forall p \in P, m \in M \quad (2-38)$$

$$\hat{v}_{pm} \leq \sum_{k \in K} \sum_{h \in H} (|P| - |K| + 1) \cdot v_{pkhm} \quad \forall p \in P, m \in M \quad (2-39)$$

约束 (2-40)–(2-42) 确定了决策变量的取值范围。

$$w_{pp'kam} \in \{0, 1\} \quad p \in P, p' \in P, k \in K, a \in A, m \in M \quad (2-40)$$

$$u_{pkhm} \in \{0, 1\}, v_{pkhm} \in \{0, 1\} \quad p \in P, k \in K, h \in A, m \in M \quad (2-41)$$

$$\hat{w}_{pp'm} \geq 0, \hat{u}_{pm} \geq 0, \hat{v}_{pm} \geq 0 \quad p \in P, p' \in P, m \in M \quad (2-42)$$

优化目标 (2-22) 和约束条件 (2-27)–(2-42) 构成了贴装过程的路径规划模型。

2.3.4 生产线的负载平衡模型

组装生产线的负载平衡模型是对多台贴片机组装任务分配的优化，其中，组装时间最长的贴片机是生产效率提升的瓶颈。对整个生产线的完整建模将导致模型复杂度过高进而不可解。本小节以贴片头任务分配模型为基础，进一步提出了生产线负载平衡模型，该模型以加权关键子目标评估单台贴片机的组装时间，贴片机的索引集合 M 扩展至整个生产线中，贴片机索引 m 按组装的先后顺序递增编号，式 (2-43) 为模型的优化目标，表示最小化组装用时最长的贴片机。

$$\min \max_{m \in M} \left(T^{\text{PL}} \cdot \sum_{i \in I} \sum_{k \in K} \sum_{h \in H} x_{ikhm} + T^{\text{CY}} \cdot \sum_{k \in K} g_{km} + T^{\text{NZ}} \cdot \sum_{k \in K} \sum_{h \in H} n_{khm} \right. \\ \left. + T^{\text{PU}} \cdot \sum_{s \in S} \sum_{k \in K} e_{skm} + T^{\text{PM}} \cdot \sum_{k \in K} d_{km} \right) \quad (2-43)$$

优化目标相关约束 (2-43) 在贴片头任务分配目标 (2-1) 的基础上增加了分配的贴装点项，即 x_{ikhm} 。相关约束中，可用工具约束 (2-15)–(2-16) 则被扩展为生产线的可用工具限制，其中，可用供料器和可用吸嘴被分别调整为约束 (2-44) 和约束 (2-45)。

$$\sum_{s \in S} \sum_{m \in M} f_{ism} \leq \phi_i \quad \forall i \in I \quad (2-44)$$

$$\sum_{m \in M} \max_{k \in K} \sum_{h \in H} z_{jkhm} \leq \zeta_j \quad \forall j \in J \quad (2-45)$$

在生产线相关的约束中, 约束 (2-46) 用于确保贴片头的元件分配结果和生产线负载均衡结果的一致性。

$$r_{im} \leq \sum_{k \in K} \sum_{h \in H} x_{ikhm} \leq N \cdot r_{im} \quad \forall i \in I, m \in M \quad (2-46)$$

表面组装生产线中新增的约束关系包括元件可分配贴片机约束和元件拾贴优先级约束。其中, 约束 (2-47) 用于约束不同贴片机可组装的元件类型。

$$r_{im} \leq \xi_{im} \quad \forall i \in I, m \in M \quad (2-47)$$

在元件组装优先级约束中, 具有拾贴优先级约束的元件分别被记为先序元件和后序元件, 后序元件在开始拾贴作业前, 需要确保所有先序元件的拾贴作业已完成, 约束 (2-48) 用于约束不同贴片机之间的元件拾贴先后顺序。

$$m - N \cdot (1 - r_{im}) \leq m' + N \cdot (1 - r_{i'm'}) \quad \forall q = (i, i') \in Q, m' \in M, m \in M \quad (2-48)$$

在同一台贴片机进行元件拾取贴装时, 同样存在优先级的约束, 约束 (2-49) 用以表明拾贴周期之间的优先级约束, 即先序元件组装完成的拾贴周期需不晚于后序元件的组装开始的拾贴周期。

$$\begin{aligned} \max_{k \in K, h \in H} k \cdot x_{ikhm} + N \cdot (r_{im} + r_{i'm} - 2) \leq \min_{k \in K, h \in H} \{k \cdot x_{i'khm} + \\ N \cdot (1 - x_{i'khm})\} \quad \forall q = (i, i') \in Q, m \in M \end{aligned} \quad (2-49)$$

与生产线负载平衡相关的决策变量取值范围为约束 (2-50)。

$$r_{im} \in \{0, 1\} \quad \forall i \in I, m \in M \quad (2-50)$$

优化目标 (2-43) 和约束条件 (2-2)–(2-5), (2-7)–(2-9), (2-12)–(2-13), (2-17)–(2-21), (2-44)–(2-50) 构成了表面组装生产线的负载平衡模型。

2.4 模型的验证与分析

2.4.1 实例演示

为说明本章提出模型的可行性, 本节以一个实际电路板为例对提出的各个模型进行求解。电路板实例如图2-6所示, 其中的贴片式元件由贴片机进行组装。示例电路板长度为 150mm, 宽度为 30mm, 共由 3 种吸嘴、8 种元件和 28 个贴装点构成。取原点为电路板的左下角, 坐标系为第一象限直角坐标系, 表2-2列出了电路板上的贴装点坐标和元件类型参数, 元件 i 记为 CP_i , 吸嘴类型 j 记为 NZ_j 。本节以装配有 6 个贴片头、贴片头间隔为供料器槽位间隔 2 倍的并列

式贴片机为例，分析贴片头的任务分配模型和贴装过程的路径规划模型；同时以三台贴片头数为6的并列式贴片机组成的表面组装生产线为例，进一步分析生产线负载平衡模型。由于数据的元件种类和贴装点数较少，本节假定可用供料器槽位数为25，同表面组装过程优化相关的子目标的模型系数根据实际组装用时线性拟合的得到，有 $T^{CY} = 0.326$ ， $T^{NZ} = 0.870$ ， $T^{PU} = 0.159$ ， $T^{PL} = 0.041$ ， $T^{PM} = 0.030$ 。

表 2-2 用于验证模型的 PCB 数据参数
Table 2-2 PCB data parameters for model validation

点 p	(X_p, Y_p) (mm)	元件 i^1	点 p	(X_p, Y_p) (mm)	元件 i	点 p	(X_p, Y_p) (mm)	元件 i
1	(43.7, 9.2)	CP ₁	11	(68.2, 27.0)	CP ₂	21	(125.3, 13.0)	CP ₄
2	(49.2, 9.2)	CP ₁	12	(44.0, 43.5)	CP ₂	22	(105.8, 40.1)	CP ₄
3	(54.2, 4.5)	CP ₁	13	(65.0, 35.2)	CP ₂	23	(111.8, 40.1)	CP ₅
4	(61.0, 4.5)	CP ₁	14	(65.0, 43.5)	CP ₂	24	(122.8, 40.1)	CP ₅
5	(34.8, 32.0)	CP ₁	15	(93.8, 9.5)	CP ₂	25	(125.0, 21.7)	CP ₆
6	(56.7, 24.5)	CP ₁	16	(99.2, 9.5)	CP ₂	26	(117.3, 40.1)	CP ₇
7	(43.0, 35.2)	CP ₁	17	(97.6, 21.8)	CP ₃	27	(111.4, 22.0)	CP ₇
8	(53.7, 35.2)	CP ₁	18	(100.3, 40.1)	CP ₃	28	(111.0, 13.0)	CP ₈
9	(68.2, 19.2)	CP ₁	19	(125.3, 4.2)	CP ₃			
10	(43.0, 43.5)	CP ₁	20	(128.5, 40.1)	CP ₃			

¹ 数据中共有3组不同的吸嘴类型NZ₁–NZ₃，其中，CP₁的吸嘴类型为NZ₁，CP₂–CP₃的吸嘴类型为NZ₂，CP₄–CP₈的吸嘴类型为NZ₃

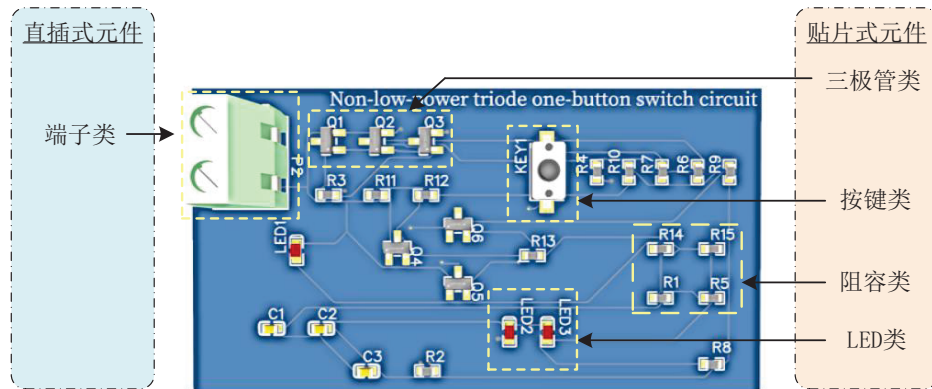


图 2-6 电路板示意图

Fig.2-6 Schematic of printed circuit board

2.4.1.1 贴片头的任务分配实例

表2-3列出了贴片头在各拾贴周期分配的元件类型，元件分配类型决定了拾取槽位和装配的吸嘴类型，表2-4给出了元件的供料器分配槽位。贴片头任务分配结果的拾贴周期数为5。以元件1为例，供料器分配结果为 $f_{1,19} = 1$ ，即元件1被分配到槽位19中。以拾贴周期1为例，贴片头1–6拾取的元件类型分别为元

件 5、元件 2、元件 3、元件 1、元件 1 和元件 7，对应的拾取槽位分别为 11、15、17、19、19 和 23，即 $y_{11,1,1,1} = 1$ ， $y_{15,1,2,1} = 1$ ， $y_{17,1,3,1} = 1$ ， $y_{19,1,4,1} = 1$ ， $y_{19,1,5,1} = 1$ 和 $y_{23,1,6,1} = 1$ 。将拾贴周期 1 各贴片头拾取槽位转换为最左侧贴片头对齐的等效槽位，贴片头 1、5 的等效槽位 \hat{s} 均为 11，贴片头 2、3、4、6 的等效槽位 \hat{s} 均为 13，即 $e_{11,1,1} = 1$ 、 $e_{13,1,1} = 1$ ，该拾贴周期共计发生 2 次同步拾取，取料过程中等效槽位的跨度为 2。依此类推，拾贴周期 2、3、4、5 的拾取次数分别为 2、2、3、2，共计 11 次；对应的拾取移动槽位分别是 2、2、4、2，共计 12 个槽位。吸嘴分配结果与元件分配结果一致，贴片头 1 和贴片头 6 装配有吸嘴 NZ_3 ，贴片头 2 和头 3 装配有吸嘴 NZ_2 ，贴片头 4 和头 5 装配有吸嘴 NZ_1 。拾贴周期 3 中贴片头 1 和拾贴周期 5 中贴片头 6 未拾贴元件，其装配吸嘴保持不变，整个组装过程的吸嘴更换次数为 0。

2.4.1.2 贴装过程的路径规划实例

贴装过程的路径规划讨论贴片机的悬臂移动过程。在表 2-2 给出的电路板数据和表 2-3 的贴片头任务分配结果的约束下，贴装过程路径规划模型的解如表 2-5 所示。以拾贴周期 1 为例，悬臂上从供料器基座拾取本周期最后一个需要贴装的元件后，前往点 24 用贴片头 1 贴装元件，即 $u_{24,1,1,1} = 1$ ；随后依次用贴片头 3 放置元件至点 13，即 $w_{24,13,1,(1,3),1} = 1$ ；用贴片头 2 贴装元件至点 20，即 $w_{13,20,1,(3,2),1} = 1$ ；用贴片头 6 贴装元件至点 5，即 $w_{20,5,1,(2,6),1} = 1$ ；用贴片头 4 贴装元件至点 6，即 $w_{5,6,1,(6,4),1} = 1$ ；用贴片头 5 贴装元件至点 27，即 $w_{6,27,1,(4,5),1} = 1$ ；最后，悬臂完成当前拾贴周期的贴装任务后，出发返回供料器基座拾取下一个周期的元件，即 $v_{27,1,5} = 1$ 。拾贴周期 1 的最后一次拾取动作由贴片头 2 和 6 在槽位 17、25 完成，对应的等效槽位为 15；下一周期的首个拾取动作由贴片头 1 和 6 在槽位 11、17 完成，对应的等效槽位为 11。各拾贴周期拾取的元件类型和贴装点的元件类型相同。图 2-7 以最左侧贴片头的位置为基准，展示了拾贴周期 1 的悬臂组装过程移动路径图。

表 2-3 贴片头任务分配模型的解
Table 2-3 Solution of head task assignment model

周期	H ₁	H ₂	H ₃	H ₄	H ₅	H ₆
1	CP ₅	CP ₂	CP ₃	CP ₁	CP ₁	CP ₇
2	CP ₅	CP ₂	CP ₃	CP ₁	CP ₁	CP ₇
3		CP ₂	CP ₃	CP ₁	CP ₁	CP ₈
4	CP ₄	CP ₂	CP ₂	CP ₁	CP ₁	CP ₆
5	CP ₄	CP ₂	CP ₃	CP ₁	CP ₁	

表 2-4 贴片头任务分配模型中元件的供料器安装槽位

Table 2-4 Feeder slots for components of placement head task allocation model

元件	CP ₁	CP ₂	CP ₃	CP ₄	CP ₅	CP ₆	CP ₇	CP ₈
槽位	19	15	17	13	11	25	23	21

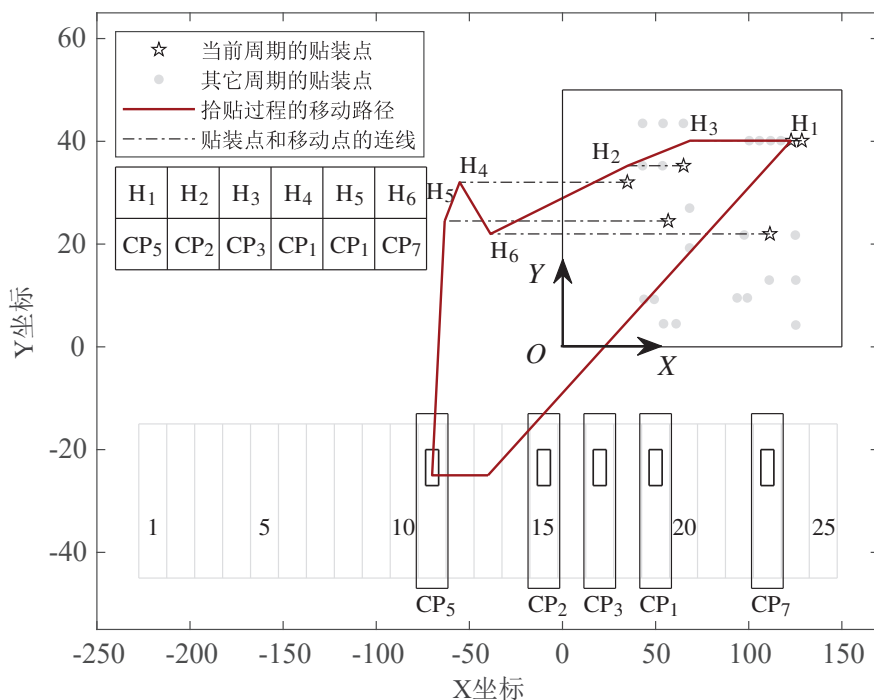


图 2-7 第一个拾贴周期的悬臂移动路径

Fig.2-7 Movement path of the gantry in the 1st PAP cycle

表 2-5 贴装过程路径规划模型的解

Table 2-5 Solution of path planning model in placement process

周期	起始	贴装点访问顺序及贴片头作业顺序	返回
1	$\hat{s} = 13$	$p = 24 \rightarrow p = 13 \rightarrow p = 20 \rightarrow p = 5 \rightarrow p = 6 \rightarrow p = 27$ $h = 1 \rightarrow h = 3 \rightarrow h = 2 \rightarrow h = 6 \rightarrow h = 4 \rightarrow h = 5$	$\hat{s} = 11$
2	$\hat{s} = 13$	$p = 23 \rightarrow p = 14 \rightarrow p = 17 \rightarrow p = 7 \rightarrow p = 9 \rightarrow p = 26$ $h = 1 \rightarrow h = 3 \rightarrow h = 2 \rightarrow h = 6 \rightarrow h = 4 \rightarrow h = 5$	$\hat{s} = 11$
3	$\hat{s} = 13$	$p = 12 \rightarrow p = 18 \rightarrow p = 1 \rightarrow p = 4 \rightarrow p = 28$ $h = 3 \rightarrow h = 2 \rightarrow h = 6 \rightarrow h = 4 \rightarrow h = 5$	$\hat{s} = 11$
4	$\hat{s} = 15$	$p = 21 \rightarrow p = 16 \rightarrow p = 15 \rightarrow p = 10 \rightarrow p = 8 \rightarrow p = 25$ $h = 1 \rightarrow h = 2 \rightarrow h = 3 \rightarrow h = 6 \rightarrow h = 4 \rightarrow h = 5$	$\hat{s} = 11$
5	$\hat{s} = 13$	$p = 22 \rightarrow p = 11 \rightarrow p = 19 \rightarrow p = 2 \rightarrow p = 3$ $h = 1 \rightarrow h = 3 \rightarrow h = 2 \rightarrow h = 4 \rightarrow h = 5$	$\hat{s} = 11$

2.4.1.3 生产线的负载平衡实例

表面组装生产线负载平衡是不同元件分配给各贴片机的决策过程优化，同一类元件可能存在多个可用供料器，本小节假定有较多贴装点的元件 1 和元件

2 分别有 3 个和 2 个可用供料器，其余元件的可用供料器数为 1。表2-6列出了表面组装生产线模型的分配结果。以元件 1 为例，其分给贴片机 1、贴片机 2 和贴片机 3 的点数分别为 8、0 和 2，即 $\sum_{k \in K} \sum_{h \in H} x_{1,k,h,1} = 8$ ， $\sum_{k \in K} \sum_{h \in H} x_{1,k,h,2} = 0$ ， $\sum_{k \in K} \sum_{h \in H} x_{1,k,h,3} = 2$ ，各个类型元件分配的贴片机数不超过其可用供料器数的上限。表2-7列出了各贴片机的优化子目标和加权目标值，贴片机的组装效率由加权性能指标的结果表示。由表可知，对于并列式贴片机，贴装点数不再是影响组装效率的唯一主要因素，拾取效率同样会对整体效率产生较大的影响。

表 2-6 表面组装生产线的负载平衡后不同贴片机分到的各类元件贴装点数
Table 2-6 Number of placement points assigned to each surface mounter after load balancing of the surface assembly line

元件	CP ₁	CP ₂	CP ₃	CP ₄	CP ₅	CP ₆	CP ₇	CP ₈
贴片机 1	8	0	0	0	0	0	0	0
贴片机 2	0	4	4	0	0	0	0	0
贴片机 3	2	2	0	2	2	1	2	1
总数	10	6	4	2	2	1	2	1

表 2-7 表面组装生产线的负载平衡模型的主要性能指标
Table 2-7 Main performance indexes of load balancing models for a surface assembly line

贴片机	子目标					目标值
	周期数	拾取数	吸嘴更换数	拾取移动槽位	拾贴点数	
1	2	4	0	4	8	1.692
2	2	4	0	4	8	1.692
3	2	3	0	2	12	1.571

2.4.2 灵敏度分析

灵敏度分析旨在探究模型参数发生变动后，模型的最优解及其变化趋势的调整情况，进而评估模型对于参数变化的敏感性与稳定性。在2.4.1节的模型实例演示中，分阶段模型的主要目标权重系数已通过线性拟合的方法获得。本节将在此基础上，进一步分析模型优化过程中关键子目标参数灵敏度。在分阶段模型框架下，通过目标权重拟合所得的目标参数可能因组装数据的差异而出现参数值的变化，相比之下，依据贴装点间距离确定的目标参数模型结果则为确定值。因此，本节将探讨模型目标拟合权重系数（即 T^{CY} ， T^{NZ} ， T^{PU} ， T^{PL} 和 T^{PM} ）值的变动对贴片头任务分配模型及生产线负载平衡模型的最优解的影响。表2-8列出了用于灵敏度分析的 PCB 数据组 1，并以贴片头数为 6 的并列式贴片机及由 3 台此类贴片机组成的组装生产线为对象，进行了模型的灵敏度分析。对于整数规划模型，原有模型中目标系数值和子目标的优化结果均会对参数的

灵敏度分析产生影响，本节将参数变化范围的结果保留三位小数。

表 2-8 模型灵敏度分析的数据参数

Table 2-8 Data parameters for model sensitivity analysis

PCB	1-1	1-2	1-3	1-4	1-5
吸嘴类型数	1	2	2	5	4
元件类型数	1	2	2	3	8
可用供料器数	4	5	5	7	8
贴装点数	28	20	28	24	20

表2-9列出了贴片头任务分配模型中，目标参数变化对模型最优解的影响。对于表面组装任务，贴片头中相关的关键子目标是决定组装效率的主要因素，本文认为子目标值发生改变即最优解发生改变，并会对组装效率产生影响，表2-9列出了各组数据主要目标值不变时参数的容许变化范围。在主要目标中，周期项和元件拾取项存在着关联性，为了构成同步拾取，元件会被分配在不同的贴片头上，间接地降低了组装过程的周期数，从而使得周期项有相对大的可变化范围。移动项因目标值较小、对组装效率贡献较低，因而容许变化范围较大。拾取项和吸嘴更换项是决定组装效率的关键因素，其变化范围相对较小，反映了其对参数变化更加敏感。可以看出，尽管参数的具体数值可能会因数据差异而有所调整，但这些调整有较大的容许变化范围使得模型的最优解保持不变，对组装效率不会产生影响，反映了组装过程模型的稳定性和鲁棒性。

表 2-9 贴片头任务分配模型的参数灵敏度分析

Table 2-9 Parameter sensitivity analysis of head task assignment model

PCB	目标项的参数							
	周期项: T^{CY}		吸嘴更换项: T^{NZ}		拾取项: T^{PU}		移动项: T^{PM}	
1-1	0.279	~ N	—	—	0.000	~ 0.206	0.000	~ 0.041
1-2	0.030	~ N	—	—	0.000	~ N	0.000	~ 0.326
1-3	0.120	~ N	0.018	~ N	0.060	~ 13.799	0.000	~ 0.079
1-4	0.030	~ N	0.016	~ N	0.000	~ 10.410	0.000	~ 0.326
1-5	0.060	~ N	0.015	~ N	0.030	~ N	0.000	~ 0.158

同理，表2-10对生产线的负载平衡模型优化目标参数灵敏度进行了分析，选取了制约组装效率设备的主要子目标值作为评估最优解变化的基准。在贴片头任务分配策略的基础上，进一步引入了贴装项的灵敏度分析。由于生产线中周期项与贴装项之间存在关联性，平衡这两个子目标的过程呈现出相关性，从而使其有相对较大的允许变化范围。此外，拾取项的表现受到所分配元件类型及贴装点数的影响。在模型经过平衡优化后，拾取效率通常能达到较高水平，使得拾取项的参数变化范围相对较大。各项参数变化的影响和贴片头任务分配相

同。对于生产线负载平衡模型，对小规模数据的负载均衡通常不会有吸嘴更换的动作，因而其参数变化不会对最优解产生影响。负载平衡模型其余主要目标参数值的可变化范围较大，其优化结果对目标参数的变化同样不敏感。

表 2-10 生产线负载平衡模型的参数灵敏度分析

Table 2-10 Parameter sensitivity analysis of assembly line balancing model

PCB	目标项的权重参数												
	周期项: T^{CY}				拾取项: T^{PU}			移动项: T^{PM}			贴装项: T^{PL}		
1-1	0.031	~	N		0.157	~	1.980	0.157	~	0.283	0.000	~	0.326
1-2	0.034	~	N		0.064	~	0.305	0.064	~	0.305	0.000	~	1.185
1-3	0.030	~	N		0.000	~	N	0.000	~	0.348	0.000	~	0.437
1-4	0.031	~	N		0.000	~	0.305	0.000	~	0.374	0.000	~	0.326
1-5	0.000	~	N		0.000	~	N	0.000	~	0.286	0.000	~	N

2.4.3 复杂度分析

模型的复杂度分析用于评估精确解法在解决实际问题时的难易程度，判断给定规模的问题需要的求解时间和内存空间。贴片头任务分配模型的决策变量和约束数分别如式 (2-51) 和式 (2-52) 所示，其中供料器槽位数 $|S|$ 、贴片头数 $|H|$ 和贴片机数 $|M|$ 为与生产数据无关的常数，拾贴周期数 $|K|$ 与贴装的点数 $|P|$ 正相关。据此可知，贴片头任务分配模型变量和约束的规模均为 $O((|I| + |J|) \cdot |P| + |P| + |I|)$ 。对于复杂的 PCB 组装任务，其元件类型数和贴装点数同步增加将显著增加模型的规模。

$$\left\{ (|S| \cdot |I| + |S| + |I| + |J|) \cdot |K| \cdot |H| + 2 \cdot (|S| + 1) \cdot |K| + 2 \cdot |S| \cdot |I| \right\} \cdot |M| \quad (2-51)$$

$$\left\{ (4 + |I| \cdot |J| + |S| \cdot |I|) \cdot |K| \cdot |H| + (|S|^2 + |S| + |J|) \cdot |K| + (|S| + 2) \cdot |I| + |S| \right\} \cdot |M| \quad (2-52)$$

贴装过程路径规划模型是在已知贴片头任务分配解的基础上构建的，其决策变量和约束数同贴装点数直接相关，分别如式 (2-53) 和 (2-54) 所示。模型中的移动弧数 $|A|$ 为与贴片头数 $|H|$ 相关的常数，其变量和约束的规模均为 $O(|P|^3 + |P|^2 + |P|)$ ，贴装点数的增加会使得模型的规模迅速扩大。

$$|P|^2 \cdot |K| \cdot |A| + 2 \cdot |P| \cdot |K| \cdot |H| + |P|^2 + 2 \cdot |P| \quad (2-53)$$

$$4 \cdot |P| \cdot |K| \cdot |H| + (2 + |H|) \cdot |K| + |P|^2 + 5 \cdot |P| \quad (2-54)$$

生产线负载均衡模型是贴片头任务分配模型的进一步拓展，其决策变量数在式 (2-51) 的基础上增加了 $|I|$ 项，约束数则为式 (2-52) 和式 (2-55) 之和，对应

的规模分别为 $O((|I| + |J|) \cdot |P| + |I| + |J|)$ 和 $O((|I| + |J|) \cdot |P| + |I|)$ ，负载均衡的分配方式并未降低模型的复杂度。

$$(|S| \cdot |I| + |S| + 2 \cdot |I|) \cdot |M| + |I| + |J| + (|M|^2 + |M|) \cdot |Q| \quad (2-55)$$

上述分析说明了本文所研究的表面组装过程优化问题的复杂性，多阶段的数学规划模型求解方法受问题规模的影响较大。在处理大规模数据时，模型计算时间的急剧增长成为优化的瓶颈，因而需要采用启发式或近似算法进行替代，通过适度放宽最优性标准来获取可接受的计算效率。传统的（混合）整数规划模型的规模主要取决于变量和约束的数量、问题结构、计算资源等因素，在常规的计算资源配置下，现有算法在可接受时间范围内能有效处理的问题规模上限为决策变量数量级小于 10^4 、约束数量级小于 5×10^4 。在 PCB 组装的应用场景中，模型复杂度主要取决于元件类型数和贴装点数两个维度，而本文所求解的大规模 PCB 数据通常元件类型数与贴装点数之积约为 2×10^3 及其以上，远超出常规求解器的处理能力范围，需要进一步改进模型或设计启发式算法解决此类优化问题。

2.5 本章小结

本章阐述了贴片机及其生产线的表面组装过程优化问题，分析了影响组装过程效率的主要约束与性能指标。在建模策略上，本章以影响组装效率的关键因素为优化目标，依据工具匹配性、资源有限性及机械结构等约束条件，构建了贴片头任务分配的整数线性规划模型；根据贴片头任务分配的结果，确定了贴装过程路径规划模型的参数，将其视为中心位置已知的带约束的多拾贴周期路径规划问题，以最小化悬臂移动路径为目标，构建了混合整数线性规划模型。此外，本章以加权子目标评估贴片机的组装效率，将工具有限性等约束扩展至生产线，并结合可分配贴片机与组装优先级等限制，建立了负载平衡的最小-最大整数线性规划模型。最后，本章通过引入电路板实例说明了模型的可解性和可行性，并对模型的参数灵敏度和复杂度进行了分析。

第3章 表面组装过程中贴片头的任务分配优化算法

3.1 引言

贴片头是表面组装过程中拾取和贴装元件的核心部件，其任务分配主要研究的是贴片头拾取的元件类型、槽位以及安装的吸嘴类型的决策优化，是表面组装过程优化的重要子问题。绝大多数与组装效率相关的关键因素以及组装过程的约束条件均和贴片头任务分配的结果直接相关，制约了解决方案的有效性和可行性。贴片头拾取元件类型和元件的供料器安装槽位之间存在关联性，其任务分配的结果决定了贴装过程路径规划可行域的范围。将贴片头任务分配问题作为分解后的首个子问题，不仅可以降低问题求解的难度，还可以通过优化主要性能指标提升整体解的质量，对表面组装过程效率的提升有着重要意义。

基于模型的优化方法兼具搜索全面性和结果可靠性，将其应用于问题的拆分环节是解决复杂优化问题的常用策略，结合数学规划和启发式算法优势的混合优化方法可快速求解问题并获得高质量的解^[108]。直接建模的求解方式在数据的适用范围上有较大的局限性，在实际应用中，相关研究常根据问题的结构特征采取近似化的建模策略。在模型改进方面，启发式的模型缩减和增强式的剪枝策略有助于提升模型的求解效率^[109]，通过寻找一系列可替代的最优解，评估其具体执行操作所带来的收益，有助于进一步提升解的质量^[110]。本章基于第2章的贴片头任务分配模型，提出了基于周期任务集成的整数线性规划模型，以获得更大规模问题的、关键效率指标拟合的近似最优解。

数学规划法的求解效率受问题规模的限制，而构造启发式算法作为基于直观或经验设计的搜索算法，具有高效解决问题的能力。在实际应用中，启发式算法难点在于设计高效的搜索规则，经过精心设计的规则可获得高质量的解。贴片头任务分配问题的约束条件和决策变量较多，对大规模问题的解进行搜索在运算量上将带来维度灾难，因此需要从中提取主要子问题进行优化。层次分析法按目标优先级及重要性拆解问题，常被运用于解决复杂优化问题。为了优化大规模数据的组装过程，本章进一步提出了基于分层启发式前瞻扫描的贴片头任务分配算法，在搜索过程中前瞻性地调整分配结果，通过效率相关的影响因素综合评估解的质量，保证了搜索过程的高效性、解的高质量和方法的可拓展性。

3.2 问题分析

贴片头任务分配是与表面组装拾取过程密切相关的问题,通过模拟并列式贴片头的拾取过程可得到任务分配的可行解。最直接的问题求解方式是贪心地从基座上不同槽位同步拾取元件,以减少拾取的总次数,相关研究在文献[19]中已有过讨论。此类求解方式能够适应多样化的供料器槽位布局,同时也存在着局限性,主要有:

1) 供料器分配布局研究不充分:当前针对供料器配置的研究相对较少,而供料器安装槽位决定了能达成同步拾取的贴片头数的上限,供料器分配与贴片头拾取问题在结构上具有相似性,若将两者完全独立讨论,将会降低解决方案的整体质量。

2) 效率影响因素考虑不全面:在解决多类型元件组装过程优化问题时,吸嘴更换是拾取过程中的必要环节。然而,频繁且耗时的吸嘴更换虽然能使更多的贴片头同步拾取,但无法带来效率上的提升;在供料器安装槽位和贴片头拾取槽位的决策中,元件的吸嘴类型均应是重要的参考因素。

3) 贪心搜索策略设计短视性:采用贪心算法进行搜索时,往往缺乏对整体布局位置的充分考量,对单一槽位可能有多种组合方式构成同步拾取的研究不足,在搜索过程中存在短视性,容易陷入局部最优解。

对于表面组装的拾取过程,贴片头通常倾向于从供料器槽中重复执行能使同步拾取贴片头数最大化的拾取动作。本文将多个贴片头同步拾取元件记为一次拾取,最大化同步拾取的贴片头数是并列式贴片机提升组装效率的重要手段。供料器分配过程与贴片头拾取过程是具有一定的相似性的。供料器通常安插在若干个连续的贴片头对齐槽位以实现同步拾取,且其安装槽位接近贴装区域以减少贴片头的移动路径长度。此外,由于元件尺寸各异,不同类型的供料器所占据的槽位数可能也各不相同,安插槽位之间的位置干涉也是供料器槽位分配中需要避免的问题。同一个拾贴周期中,贴片头从不构成同步拾取的槽位中取料可减少拾贴周期数,但此过程可能会带来较长的拾取移动距离。在贴片头拾取元件时,既要衡量贪心地最大化当前同步拾取所能带来的效率提升,又要评估从当前槽位拾取元件后,其对后续拾取过程以及整体组装效率的影响。

3.3 基于周期任务集成模型的贴片头任务分配算法

直接构建的贴片头任务分配模型面临求解效率低的问题,本节将讨论提升模型求解效率的方法。在元件拾取过程中,供料器内装载有同类元件,且允许贴

片头多次重复地从中拾取元件。这表明在构成同步拾取任务的供料器组合中,各供料器对应的贴装点数量较多时,贴片头可通过重复相同拾取动作来提升拾取效率。受这一过程的启发,本节将连续且执行相同任务的周期组视为决策的基本单元,通过重组和整合拾贴周期,提出了一种基于周期任务集成模型(Cyclic Task Integration Model, CTIM)的算法,旨在解决贴片头任务分配问题。图3-1展示了本章提出模型的构建、增强、求解和输出环节中运用的主要策略。

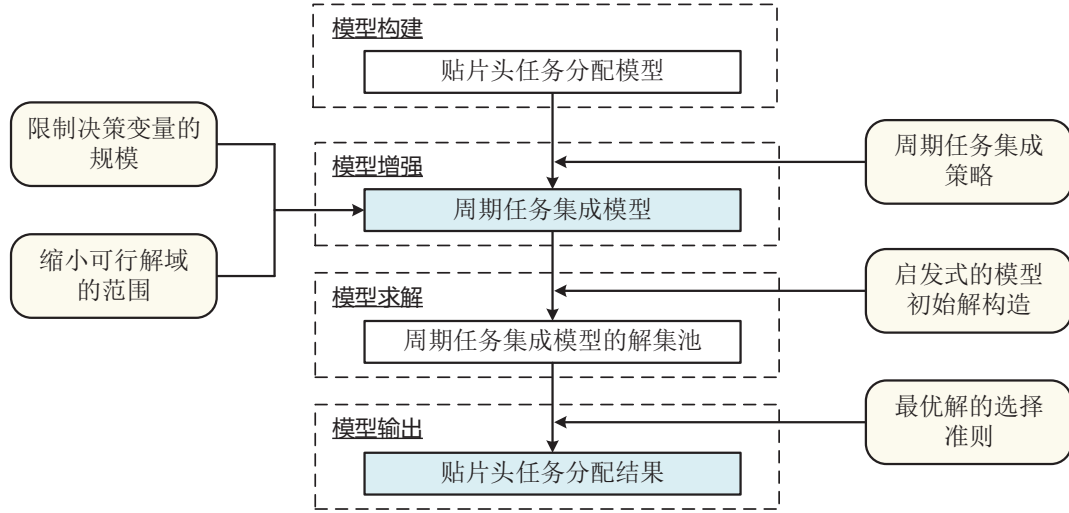


图 3-1 周期任务集成模型的算法框图

Fig. 3-1 Algorithm block diagram of the cyclic task integration model

3.3.1 周期任务集成模型

在既有的贴片头任务分配模型基础上, CTIM 将具有一致拾贴任务的连续拾贴周期作为周期组进行决策优化。原模型中贴片头的拾取移动路径项 d_{km} 对整体组装效率影响较小, 为了降低模型复杂度, 本章在构建模型的过程中不计与之相关的优化目标和约束条件。CTIM 可被视为贴片头任务分配的增强模型, 其周期组内拾取的元件类型和槽位、安装的吸嘴类型均与原有模型的优化结果保持一致。其与原模型的主要区别在于, 拾贴周期索引 k 及其集合 K 在 CTIM 中被视为周期组索引及其集合。原模型中用于决定拾贴周期 k 是否被分配贴装点的决策变量 g_{km} , 在 CTIM 中被调整为记录任务集中拾贴周期数的决策变量为 b_{km} 。同时, 与拾取次数相关的项 e_{skm} 则被调整为 $b_{km} \cdot e_{skm}$, 表示贴片机 m 在周期组 k 中贴片头拾取元件时, s 为等效槽位的总拾贴周期数。同样的, 贴片机索引集 $M = \{1\}$ 。CTIM 的优化目标如式 (3-1) 所示, 该目标由拾贴周期数、吸嘴更换次数以及元件拾取次数三项构成。

$$\min T^{\text{CY}} \cdot \sum_{k \in K} b_{k1} + T^{\text{NZ}} \cdot \sum_{k \in K} \sum_{h \in H} n_{kh1} + T^{\text{PU}} \cdot \sum_{s \in S} \sum_{k \in K} b_{k1} \cdot e_{sk1} \quad (3-1)$$

优化目标 (3-1) 中的非线性项为 0-1 变量和非负整数变量之积, 可以用引入整数中间变量 $\lambda_{sk} = b_{k1} \cdot e_{sk1}$ 来代替, 式 (3-2) 用大 M 法对此项做了线性化处理。

$$\begin{cases} \lambda_{sk} \leq N \cdot e_{sk1}, \\ \lambda_{sk} \leq b_{k1}, \\ \lambda_{sk} \geq b_{k1} - N \cdot (1 - e_{sk1}), \end{cases} \quad \forall s \in S, k \in K \quad (3-2)$$

贴片头任务分配模型中和拾贴周期决策变量 g_{km} 相关的变量均需做出调整。其中, 元件分配和拾贴周期数规划的一致性约束由式 (2-2) 调整为式 (3-3)。

$$x_{ikhm} \leq 1 \quad \forall i \in I, k \in K, h \in H, m \in M \quad (3-3)$$

同理, 工作完整性约束由式 (2-11) 调整为式 (3-4), 用以确保所有周期组中元件 i 的贴装点数总和等于其在 PCB 上的贴装点数。

$$\sum_{k \in K} \sum_{h \in H} \sum_{m \in M} b_{km} \cdot x_{ikhm} = \psi_i \quad \forall i \in I \quad (3-4)$$

式 (3-4) 的非线性项可用中间变量 $c_{ikhm} = b_{km} \cdot x_{ikhm}$ 表示, 其线性化过程与目标 (3-2) 中非线性项的处理方法一致。

新增的决策变量的取值范围如式 (3-5) 所示, 其中 \mathbb{N}_0 表示非负整数集。

$$b_{kh} \in \mathbb{N}_0, c_{ikhm} \in \mathbb{N}_0 \quad \forall i \in I, k \in K, h \in H, m \in M \quad (3-5)$$

优化目标 (3-1) 和约束条件 (2-4)–(2-5), (2-7), (2-9), (2-14)–(2-21), (3-3)–(3-5) 构成了基于周期任务集成的贴片头任务分配模型, 该模型同时是整数线性规划模型。

3.3.2 初始可行解的构建

构造初始可行解是求解整数规划模型中的重要步骤。分支定界法作为解决整数线性规划模型的常用解法, 其通过线性松弛模型的最优解构造一个整数可行解, 进而递归地搜索模型的全局最优解。初始解的选择决定了搜索过程的起始基准点, 快速确定模型的初始解可以避免因问题特性不明确而导致模型的迭代搜索时间过长。对于不同类型的优化问题, 采用不同的初始解选择策略可能会产生显著的差异, 一个高质量的初始解可以降低模型求解过程中的迭代次数。此外, 在构建问题 CTIM 的过程中, 周期组数的设定是决定模型复杂度的主要因素。过多的周期组数设置不仅无法提升解的质量, 反而会延长求解时间; 过

少的周期组数设置则会导致模型无解。因此，依据数据的特征确定周期组数在建模过程中较为重要。本节将围绕上述问题，研究启发式的模型初始化方法，同时确定模型的初始解和超参数周期组集合 K 的值。

模型的初始化启发式共由两部分组成，分别确定了吸嘴装配的贴片头数和贴片头上的元件分配结果，如算法3-1所示。在确定吸嘴的贴片头数时，优化条件为在不改变贴片头上装配吸嘴的约束下最小化拾贴周期数，即在各吸嘴至少分配一个贴片头的基础上，将剩余贴片头优先分给已分配贴片头平均点数最少的吸嘴。吸嘴类型 j 分配的贴片头数 π_j 将作为后续算法设计的基础。在确定贴片头分配的元件类型时，基于给定吸嘴贴片头数，模型初始化算法从首个周期组开始，通过向周期组索引集合添加新的索引，递归搜索可行的初始解 C 和 \mathcal{W} 。其中， C_{kh} 为周期组 k 中贴片头 h 拾贴的元件类型， \mathcal{W}_k 为周期组 k 内的拾贴周期数。算法3-1构造的模型解的结构简单，可直接确定周期组索引集合 $K = \mathcal{K}$ ，其输出的结果可依据式 (3-6) 确定模型的初始周期组数 $b_{k1}^{(0)}$ 和拾贴周期元件分配结果 $x_{ikh1}^{(0)}$ ，其余决策变量可根据约束条件通过试凑的方式快速确定。

$$b_{k1}^{(0)} = \mathcal{W}_k, \quad \sum_{i \in I} i \cdot x_{ikh1}^{(0)} = C_{kh} \quad \forall k \in K, h \in H \quad (3-6)$$

算法 3-1 周期任务集成模型的启发式初始化算法

Algo.3-1 Heuristic initialization algorithms for cyclic task integration model

输入：元件的贴装点数 ψ ，元件和吸嘴匹配性 μ
输出：贴片头的元件分配类型 C ，周期组的拾贴周期数 \mathcal{W} ，周期组索引集合 \mathcal{K}

- 1 令周期组索引集合 $\mathcal{K} \leftarrow \{1\}$ 和吸嘴装配的贴片头数 $\pi_j \leftarrow 1, \forall j \in J$;
- 2 **while** $\sum_{j \in J} \pi_j \neq |H|$ **do**
 - /* 按照吸嘴贴片头平均分配点数最大的原则选择吸嘴 */
 - 3 $j \leftarrow \operatorname{argmax}_{j' \in J} \{j' \mid \sum_{i \in I} \mu_{ij'} \cdot \psi_i / \pi_{j'}\}$;
 - 4 $\pi_j \leftarrow \pi_j + 1$;
- 5 **end**
 - /* 递归分配各周期组贴片头的元件，直至找到可行解 */
- 6 **while true do**
 - 7 令 C 为大小为 $|\mathcal{K}| \times |H|$ 的列表，令 \mathcal{W} 为大小为 $|\mathcal{K}| \times 1$ 的列表；
 - 8 **if** $\operatorname{recursive}(\max_{i \in I} \psi_i, \psi, 1, \pi, \mathcal{K}, C, \mathcal{W}) = \text{success}$ **then**
 - 9 | **break**; // 找到可行解，退出搜索
 - 10 **end**
 - 11 $\mathcal{K} \leftarrow \mathcal{K} \cup \{|\mathcal{K}| + 1\}$; // 当前周期组无解，增加周期组数
- 12 **end**

初始化过程中递归分配贴装点函数 *recursive* 的实现如算法3-2所示，其分配过程是按照周期组索引，以贴装点非递减的顺序递归向贴片头分配贴装点。同时，算法优先分配贴装点多于拾取周期分配组数且点数最少的元件，以增加搜索解的多样性。递归过程的返回值可能有三种情况。其中 *success* 表示搜索

算法 3-2 递归分配函数 *recursive* 的实现Algo.3-2 Implementation of the *recursive* assignment function

输入：周期组数 ℓ ，元件贴装点数 ψ ，周期组索引 k ，周期组索引集合 \mathcal{K} ，吸嘴分配贴片头数 π ，贴片头已分配的元件 \mathcal{C} ，周期组已分配的拾贴周期数 \mathcal{W}

输出：贴片头分配的元件类型 \mathcal{C} ，周期组分配的拾贴周期数 \mathcal{W}

```

1 function recursive( $\ell, \psi, k, \pi, \mathcal{K}, \mathcal{C}, \mathcal{W}$ )
2   if  $k > |\mathcal{K}|$  and  $\sum_{i \in I} \psi_i = 0$  then
3     return success; // 已找到可行解
4   else if  $\ell \leq 0$  and  $k = 1$  then
5     return fail; // 当前周期组数无解，需要增加周期组数重新搜索
6   else if  $\ell \leq 0$  or  $k > |\mathcal{K}|$  then
7     return backtrack; // 当前分配周期组数不可行
8   end
9   令  $\psi' \leftarrow \psi$ ,  $\pi' \leftarrow \pi$ ,  $\mathcal{W}_k \leftarrow \ell$ ，初始化  $\mathcal{C}_k$  为空列表;
10  for  $j \in J$  do
11    while  $\pi'_j > 0$  do
12      // 选择吸嘴类型为  $j$  的贴装点数最少且大于周期组数的元件
13       $i' \leftarrow \operatorname{argmin} \{ \psi_i \mid \mu_{ij} \cdot \psi_i \geq \ell, i \in I \}$ ;
14      // 更新贴片头的分配元件，元件的点数和已遍历的头数
15       $\mathcal{C}_k \leftarrow [\mathcal{C}_k, i']$ ,  $\psi_{i'} \leftarrow \psi_{i'} - \ell$ ,  $\pi'_j \leftarrow \pi'_j - 1$ ;
16    end
17  end
18  /* 递归寻找可行解 */
19   $res \leftarrow recursive(\max_{i \in I} \psi_i, \psi, k + 1, \pi, \mathcal{K}, \mathcal{C}, \mathcal{W})$ ;
20  if  $res = success$  then
21    return success;
22  else if  $res = backtrack$  then
23    return recursive( $\ell - 1, \psi', k, \pi, \mathcal{K}, \mathcal{C}, \mathcal{W}$ );
24  end
25 end

```

成功，指已找到模型的初始可行解；*fail* 表示搜索失败，指对于给定的模型，当前的周期组集合 \mathcal{K} 不可行，需要增大周期组的个数重新进行搜索；*backtrack* 表示回溯搜索，指分配周期组 k 时，当前分配的周期组数 ℓ 无法获得可行解，需要执行回溯尝试分配更少的周期组数 $\ell - 1$ 。

3.3.3 模型的可解性增强策略

周期任务集成策略降低了模型的规模，使其复杂度主要依赖于数据的元件类型数，而降低了与贴装点数之间的关联性。为了进一步提高模型的求解效率，本节将研究两类模型的可解性增强策略：一是通过限定决策变量的规模来压缩决策空间；二是通过引入额外的约束条件缩小可行解域的范围。

1) 限制决策变量的规模

供料器基座上较多的槽位数大幅增加了模型的决策变量数。在实际的表面组装过程中，为提高拾取过程的效率，供料器通常被紧密排列在供料器基座的

有限区域内，因此，距离贴装区域较远的插槽可被忽略不计。对于贴片头和供料器插槽间隔比为 τ 的贴片机，相邻供料器之间的间隔槽数通常为 $\tau - 1$ 。在元件类型数有限的情况下，假定 $\tau = 1$ 不会降低模型解的质量，反而有助于简化求解过程。在模型求解过程中，本节以供料器总数为拾取槽位的区间跨度，以最左侧拾取槽位为基准，根据贴片头的元件分配结果确定参考槽位号，具体为：

步骤 I：确定不同类型元件 i 所分配的贴片头与对应的拾贴周期数的加权平均值 \bar{h}_i 。

$$\bar{h}_i \leftarrow \left(\sum_{k \in K} \sum_{h \in H} x_{ikh1} \cdot h \cdot b_{k1} \right) / \sum_{k \in K} b_{k1} \quad \forall i \in I. \quad (3-7)$$

步骤 II：以最左侧贴片头为基准，将所有贴装点的 X 坐标转换为其贴装坐标，取其平均值记为 \bar{X} 。

$$\bar{X} \leftarrow \sum_{p \in P} \left\{ X_p - \left(\sum_{i \in I} \eta_{ip} \cdot \bar{h}_i - 1 \right) \cdot \rho \right\} / |P|. \quad (3-8)$$

步骤 III：记拾取过程中贴片头在供料器基座在各周期组内移动经过的平均槽位数为 $\Delta \bar{S}$ 。

$$\Delta \bar{S} \leftarrow \sum_{k \in K} \mathcal{R} \left(\left\{ s \cdot y_{skh1} - (h-1) \cdot \frac{\rho}{\tau} \mid y_{skh1} > 0, s \in S, h \in H \right\} \right) / \sum_{k \in K} b_{k1} \quad (3-9)$$

其中， $\mathcal{R}(\cdot)$ 表示集合中最大值和最小值的差。

步骤 IV：根据贴片头拾取范围（贴片头移动的槽位数）和贴片头的平均贴装位置，确定参考槽位号 s^{REF} 。

$$s^{\text{REF}} \leftarrow \left\lfloor \frac{\bar{X} - X^{\text{F1}}}{\rho} \cdot \tau + \frac{\Delta \bar{S} + 1}{2} \right\rfloor + 1 \quad (3-10)$$

据此可知元件类型 i 的拾取槽位为 $s^{\text{REF}} + \tau \cdot \sum_{s \in S} s \cdot f_{is1}$ 。

2) 缩小可行解域的范围

表面组装过程优化问题的解域较大，根据问题特性缩小解域的范围将有效提升模型的求解效率。本节通过添加约束条件来缩减模型的求解空间，从而进一步提高模型的求解效率。需要说明，约束条件 (3-11)–(3-14) 并非模型求解的必要条件，而是用于进一步缩小可行域范围，在搜索过程中去除重复解和质量较差的解。

约束 (3-11) 用于确保拾贴周期数多的周期组具有更高的优先级。对于连续作业的表面组装过程，调整周期组的先后顺序并不会影响解的质量，优先分配拾贴周期数多的周期组可以消除分配过程中的重复解。

$$b_{km} \geq b_{(k+1)m} \quad \forall k \in K \setminus \{|K|\}, m \in M \quad (3-11)$$

约束 (3-12) 界定了拾贴周期总数的取值范围。在理想的贴片头配置中, 不考虑吸嘴更换以及元件拾取效率的前提下, 可使贴片头的拾贴周期数最小化, 即尽可能均匀地将贴装点分配到贴片头上。相对而言, 在最坏的情形下, 所有贴片头在组装过程中均不更换吸嘴, 仅对分得同类型吸嘴的贴片头进行贴装点均分。

$$\lceil \sum_{i \in I} \frac{\psi_i}{|H|} \rceil \leq \sum_{k \in K} b_{km} \leq \sum_{k \in K} \mathcal{W}_k \quad \forall m \in M \quad (3-12)$$

约束 (3-13) 假定所有贴片头在组装全过程中均装配有吸嘴, 即使其不拾取和贴装元件, 用以避免在计算过程中重复判断周期组之间贴片头是否存在吸嘴更换。

$$\sum_{j \in J} z_{jkhm} = 1 \quad \forall k \in K, h \in H, m \in M \quad (3-13)$$

约束 (3-14) 进一步限定了拾贴周期规划和贴片头元件分配之间的关系, 增强了组装过程中工作完整性约束对贴片头元件分配的限制。

$$N \cdot \sum_{i \in I} \sum_{h \in H} x_{ikhm} \geq b_{km} \quad \forall k \in K, m \in M \quad (3-14)$$

3.3.4 最优解的选择准则

基于 CTIM 的贴片头任务分配的优化目标由关键效率子目标构成, 并未包含拾贴过程的悬臂移动路径项, 而对其同时进行建模会极大地增加模型的复杂度。模型的最优解通常是不唯一的, 不同的贴片头任务分配结果会影响悬臂移动路径项的优化结果。解集池用于存储模型求解过程中找到的多个最优解或近似最优解。本节通过综合考虑拾取和贴装过程中的移动路径长度, 实现对贴片头分配结果的快速评估, 进而从解集池中筛选出作为后续贴装过程路径规划约束的解。CTIM 的解直接决定了拾取过程的移动路径长度 \tilde{D}^{PU} , 即

$$\tilde{D}^{\text{PU}} = \frac{\rho}{\tau} \cdot \sum_{k \in K} b_{k1} \cdot \mathcal{R}(\{y_{skh1} \cdot (s - h \cdot \tau) \mid y_{skh1} \neq 0, s \in S, h \in H\}) \quad (3-15)$$

对于贴装过程的路径长度 \tilde{D}^{PL} , 算法3-3提出了一种快速的启发式评估方法。算法的基本原理是按元件类型和贴装点的位置对其进行分组, 估计各周期组内贴片头的贴装中心位置, 以中心位置代替实际贴装位置计算各周期组的移动路径长度。贴装点分组的中心点可以近似为实际贴装路径规划中可选择的潜在贴

装点, 预估贴装路径长度和实际规划路径长度有较强的关联性。在模型的解集池中, 取 $\tilde{D}^{PU} + \tilde{D}^{PL}$ 值最小的解作为模型的最终结果。

算法 3-3 贴片头部任务分配模型贴装路径评估启发式

Algo.3-3 Placement path evaluation heuristic for head task assignment model

输入: 贴片头元件分配的解 \mathcal{C} , 周期组的拾贴周期数 \mathcal{W} , 元件-贴装点一致性 η , 贴装点 p 的 X 坐标 X_p

输出: 贴片头的预估贴装路径 \tilde{D}^{PL}

- 1 将贴装点索引元件类型分至不同的有序集合 $P_i \leftarrow \{p \mid \eta_{ip} = 1, \forall p \in P\}$, $\forall i \in I$, 按 X 坐标递增的顺序对集合内元素进行排序;
- 2 记有序集合 $\mathcal{V}_k \leftarrow \emptyset$, $\forall k \in K$, 表示周期组内贴片头的移动位置中心点 X 坐标, 初始化 $\tilde{D}^{PL} \leftarrow 0$;
/* 按序遍历各周期组及贴片头 */
- 3 **for** $k \in K$ **do**
- 4 **for** $h \in H$ **do**
- 5 记 $i \leftarrow \mathcal{C}_{kh}$ 为当前拾贴周期 k 贴片头 h 拾贴的元件类型, 取 P_i 中前 \mathcal{W}_k 个点记为 $p_1, p_2, \dots, p_{\mathcal{W}_k}$;
- 6 向 \mathcal{V}_k 中添加中心 X 坐标 $\sum_{t=1}^{\mathcal{W}_k} [X_{p_t} - (h-1) \cdot \tau] / \mathcal{W}_k$, 移除集合 P_i 中前 \mathcal{W}_k 个元素;
- 7 **end**
- 8 $\tilde{D}^{PL} \leftarrow \tilde{D}^{PL} + (\max_{k \in K} \mathcal{V}_k - \min_{k \in K} \mathcal{V}_k) \cdot \mathcal{W}_k$; // 评估的贴装移动路径长度
- 9 **end**

3.4 基于分层前瞻扫描启发式的贴片头任务分配策略

3.4.1 算法概述

CTIM 的方法在求解中小规模数据时能保证解的质量, 但其仍存在求解效率受元件类型数限制的问题, 对元件类型更复杂的数据优化能力有限。本章将继续研究适用性更强的贴片头任务分配算法, 提出基于分层前瞻扫描启发式 (Hierarchical Look-Forward Scanning Heuristic, HLFSSH) 的贴片头任务分配算法。高效的分层框架及具有前瞻式搜索规则的启发式方法被用于解决贴片头任务分配问题。HLFSSH 通过模拟拾取过程确定贴片头任务分配的结果, 将贴片头任务拆分为三个互相关联的分层次优化问题: 吸嘴分配模式、供料器-槽位分配和贴片头-元件分配。吸嘴分配模式不以为贴片头直接分配吸嘴为目标, 而是提供一种分配过程的范式, 用于约束供料器-槽位分配和贴片头-元件分配过程, 二者共同决定了贴片头拾取过程的效率。图3-2展示了 HLFSSH 的基本流程, 基于启发式的供料器-槽位分配和贴片头-元件分配算法具有相似的搜索方式, 二者均以贴片头装配的吸嘴类型为参考, 均通过扫描供料器基座的方式, 分配供料器至插槽和分配贴片头拾贴的元件类型, 同时在分配过程中融入不同的评估函数。基于 HLFSSH 的贴片头任务分配算法在搜索效率、解的质量和可扩展性方面具有

优势，能够满足多样化的组装任务需求。

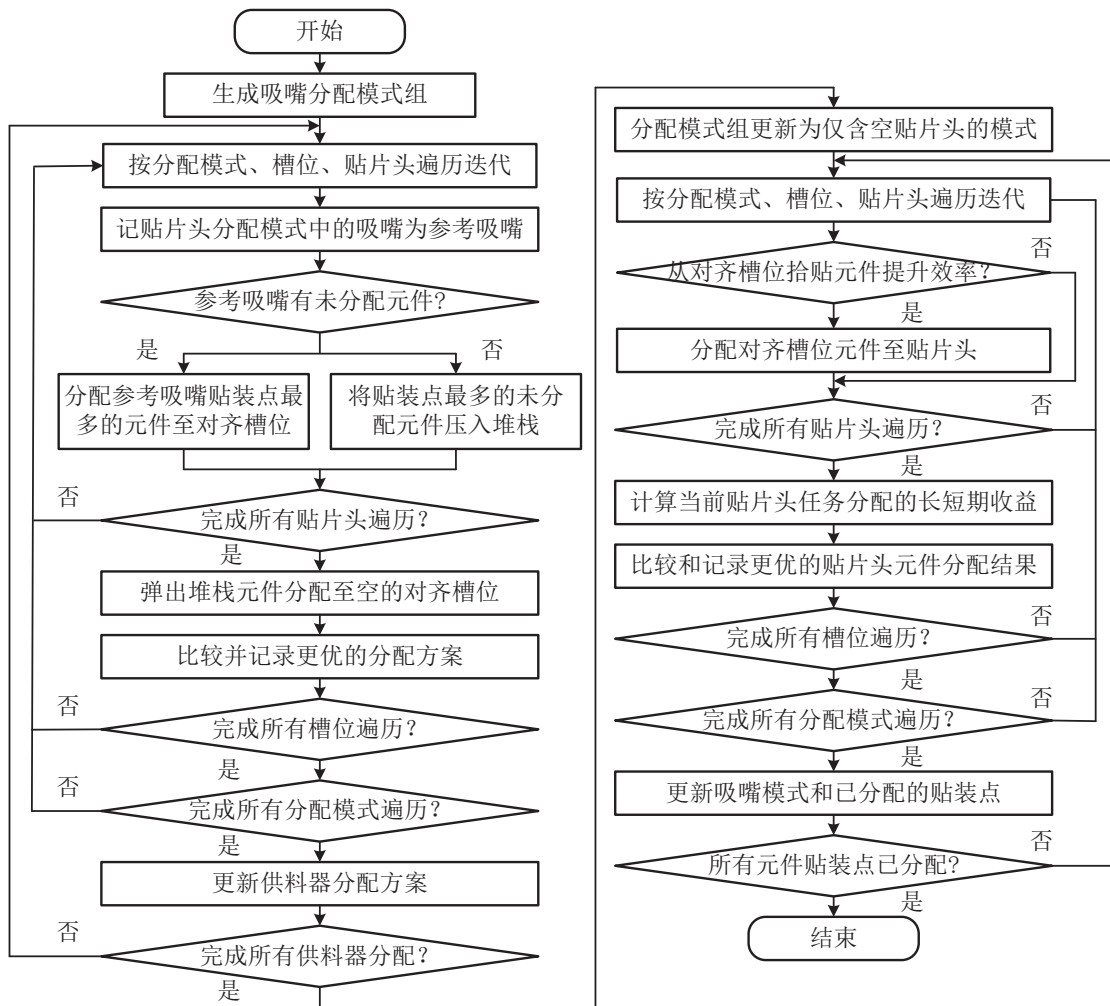


图 3-2 分层前瞻扫描启发式算法流程图

Fig.3-2 Flowchart of the hierarchical heuristic forward-looking scanning algorithm

3.4.2 喷嘴的分配模式启发式算法

在贴片头拾取元件过程中，喷嘴的类型与供料器所安装的槽位共同决定了贴片头所能达成的同步拾取数。为了保证在同步拾取效率提升的同时，尽量减少因喷嘴更换所导致的效率损失，供料器的槽位分配需充分考虑喷嘴更换的实际情况。因此，在供料器分配之前，有必要明确贴片头上所安装的喷嘴类型。本节讨论了一种喷嘴匹配模式的枚举策略，将贴片头上所安装的喷嘴类型组合称为喷嘴组，喷嘴组中各类喷嘴的数量是确定的。理论上，通过穷举喷嘴组的全排列，可以枚举出所有可能的喷嘴分配模式。然而，全排列的搜索方式会造成重复的供料器-槽位和贴片头-供料器分配结果，大幅增加算法的求解时间。本节提出的喷嘴模式启发式采用两种不同的分配策略确定喷嘴分配模式，分别是

从单侧向对侧排列吸嘴，以及从中心向两侧排列吸嘴。前者将同类型的吸嘴在贴片头上连续排列，该分配策略能使贴装点数量较多的元件构成同步拾取，分配的结果中移除了吸嘴组类型完全相同的对称项。后者则是从中间的贴片头开始分配吸嘴，按照吸嘴 j 对应贴片头数 π_j 非递增的顺序向两侧交替地排布吸嘴。两种不同的分配方式考虑了不同的拾取情况，为后续的供料器分配和元件分配提供了依据。吸嘴分配模式的生成流程如算法3-4所示。

算法 3-4 吸嘴分配模式启发式算法

Algo.3-4 Nozzle mode assignment heuristic algorithm

输入：吸嘴分配贴片头数 π ，元件贴装点数量 ψ ，元件-吸嘴匹配性 μ
输出：吸嘴模式集合 m

- 1 按照索引 $j \in J$ 对应的 $\sum_{i \in I} \mu_{ij} \cdot \psi_i$ 递减的顺序确定吸嘴索引列表 J^{LT} ， J^{LT} 中的元素由 π_j 个吸嘴索引构成；
 /* 遍历吸嘴索引列表 J^{LT} 的全排列 */
- 2 **for** J' **in** permutation (J^{LT}) **do**
 /* 由单侧向对侧排列吸嘴 */
- 3 初始化 $n \leftarrow []$ ，按序遍历 π_j 次 J' 中的 j ，令 $n \leftarrow [n, j]$ ；
- 4 将 n 添加到吸嘴模式集 m ；
- 5 **end**
- 6 移除吸嘴分配模式中的对称项；
 /* 由中心向两侧排列吸嘴 */
- 7 令贴片头索引列表 $H_L \leftarrow \left\lfloor \frac{|H|}{2} + (-1)^h \cdot \left(\lceil \frac{h}{2} \rceil - \frac{1}{2}\right) + \frac{[(|H|+1) \bmod 2]}{2} \right\rfloor$ ， $h \in H$ ；
- 8 令 n 为大小 $1 \times |H|$ 的空列表，按序遍历 π_j 次 J^{LT} 中的 j ，同时按序遍历 H_L 中的 h ，令 $n_h \leftarrow j$ ，将 n 添加到吸嘴模式集 m ；

3.4.3 供料器的槽位分配启发式算法

3.4.3.1 基本框架

吸嘴模式可以被视为贴片头安装的吸嘴类型，其并不直接作为决策的子问题，供料器-槽位分配则是贴片头-元件分配等子问题的前置问题，决定了组装过程贴片头的元件拾取次数和拾取过程的移动路径，合理的安装配置将显著提高取料效率。本节所述的供料器-槽位分配过程用于确定供料器中装载元件的拾取槽位。算法3-5中描述了供料器-槽位分配启发式的流程，在此框架下，吸嘴模式作为一项约束，限定了扫描过程中贴片头对齐的槽位中所能分配供料器对应的元件类型。在贴片头进行基座扫描与供料器分配的过程中，其分配策略为最大化每轮分配的贴装点数量，同时增加贴片头安装的吸嘴类型与槽位中供料器的吸嘴类型的一致性要求。在各轮次的基座扫描过程中，算法保留了贴片头对齐槽位中已有的供料器，同时将未分配的供料器分配到其中的空槽中。贴片头对齐槽位所能安装的供料器，受到贴片头上安装的吸嘴类型限制。采用吸嘴模式引导供料器分配过程，能够减少后续取料过程中的吸嘴更换次数，从而提

算法 3-5 供料器-槽位分配启发式算法

Algo.3-5 Feeder-slot assignment heuristic algorithm

输入：元件的贴装点数 ψ ，吸嘴的贴片头数 π ，元件和吸嘴的匹配性 μ ，吸嘴模式 \mathcal{N}

输出：供料器槽分配的元件 \mathcal{F}^{CP} 和分配的贴装点数 \mathcal{F}^{PT}

- 1 令 \mathcal{F}^{CP} 和 \mathcal{H}^{CP} 为大小为 $1 \times |S|$ 和 $1 \times |H|$ 的空列表 // 分别表示供料器槽位和贴片头对齐槽位的元件类型
- 2 令 \mathcal{F}^{PT} 和 \mathcal{H}^{PT} 为大小为 $1 \times |S|$ 和 $1 \times |H|$ 的元素全为 0 的列表 // 分别表示供料器槽位和贴片头对齐槽位的贴装点数
- /* 分配元件至供料器槽位，直至全部贴装点完成分配 */
- 3 **while** $\sum_{i \in I} \psi_i \neq 0$ **do**
- 4 令 $V^* \leftarrow 0$; // 表示已分配最优解的值
- 5 **for** 遍历所有槽位从 $s = 1$ 到 $s = |S| - (|H| - 1) \cdot \tau$ **do**
- 6 /* 记录贴片头对齐槽位的元件类型和贴装点数，备份元件索引集合 */
- 7 **foreach** $h \in H$ **do** $\mathcal{H}_h^{\text{CP}} \leftarrow \mathcal{F}_{s+(h-1) \cdot \tau}^{\text{CP}}$, $\mathcal{H}_h^{\text{PT}} \leftarrow \mathcal{F}_{s+(h-1) \cdot \tau}^{\text{PT}}$;
- 8 令 $\hat{I} \leftarrow I$; // 未分配元件的索引集合
- 9 /* 根据贴片头的吸嘴，为当前供料器槽位分配元件 */
- 10 **for** $j \leftarrow \mathcal{N}_h$, $h \in \{h' \mid \mathcal{H}_{h'}^{\text{CP}} > 0, h' \in H\}$ **do**
- 11 **if** $\psi_i = 0, \forall i \in \{i' \mid \mu_{i'j} \neq 0, i' \in \hat{I}\}$ **then**
- 12 令 $i \leftarrow \operatorname{argmax}_{i' \in \hat{I}} \{\psi_{i'}\}$, 将元件 i 压入堆栈;
- 13 **else**
- 14 令 $i \leftarrow \operatorname{argmax}_{i' \in \hat{I}} \{\psi_{i'} \mid j \cdot \mu_{i'j} > 0\}$, $\mathcal{H}_h^{\text{CP}} \leftarrow i$, $\mathcal{H}_h^{\text{PT}} \leftarrow \psi_i$;
- 15 **end**
- 16 $\hat{I} \leftarrow \hat{I} \setminus \{i\}$; // 将已分配元件从未分配元件索引集中移除
- 17 **end**
- 18 弹出堆栈中元件 i 分配给头 $h \in \{h' \mid \mathcal{H}_{h'}^{\text{PT}} = 0, h' \in H\}$, 令 $\mathcal{H}_h^{\text{CP}} \leftarrow i$, $\mathcal{H}_h^{\text{PT}} \leftarrow \psi_i$;
- 19 /* 比较并记录分配的最优值 */
- 20 **if** $\sum_{h \in H} \left(\mathcal{H}_h^{\text{PT}} + \left| n_h - \sum_{j \in J} j \cdot \mu_{\mathcal{H}_h^{\text{CP}} j} \right| \right) > V^*$ **then**
- 21 $V^* \leftarrow \sum_{h \in H} \left(\mathcal{H}_h^{\text{PT}} + \left| n_h - \sum_{j \in J} j \cdot \mu_{\mathcal{H}_h^{\text{CP}} j} \right| \right)$, $s^* \leftarrow s$, $\tilde{\mathcal{H}}^{\text{CP}} \leftarrow \mathcal{H}^{\text{CP}}$
- 22 **end**
- 23 **end**
- 24 $l = \min \left\{ \mathcal{F}_{s^*+(h-1) \cdot \tau}^{\text{PT}} \mid \mathcal{F}_{s^*+(h-1) \cdot \tau}^{\text{PT}} \neq 0, h \in H \right\}$; // 计算最大同步拾取次数
- 25 **for** $s \leftarrow s^* + (h - 1) \cdot \tau$, $h \in H$ **do**
- 26 /* 更新供料器槽的元件类型和贴装点数 */
- 27 $\mathcal{F}_s^{\text{CP}} \leftarrow \tilde{\mathcal{H}}_h^{\text{CP}}$, $\mathcal{F}_s^{\text{PT}} \leftarrow \mathcal{F}_s^{\text{PT}} - l$;
- 28 $n_h \leftarrow j$, $\psi_i \leftarrow \psi_i - l$, 其中, $i = \tilde{\mathcal{H}}_h^{\text{CP}}$, $j = \sum_{j' \in J} j' \cdot \mu_{ij'}$;
- 29 **end**
- 30 **end**

升整体组装效率。在实现过程中，算法优先分配同贴片头吸嘴类型一致且贴装点数最多元件的供料器，若该类吸嘴全部元件均已分配完成，则跳过当前贴片头，并将贴装点最多的元件暂存于堆栈中。待所有贴片头完成初步分配后，算法再将堆栈中元件的供料器依次分配至贴片头所对齐的空槽位中，以此确保供料器分配的槽位相对集中，降低拾取过程的移动路径。

3.4.3.2 实现细节

基于给定的算法框架，为增强提出算法的适用范围，解决多种组装过程约束条件下的供料器-槽位分配问题，本小节从以下几个角度对提出的算法做补充说明：

1) 供料器的预先安装

预先安装供料器是表面组装中的常见配置，旨在缩减换线时间，提升生产效率。为实现快速换线，现场操作人员通常会将频繁使用的供料器预先安装在固定的槽位上。对于小批量订单生产，若因数据调整而对供料器的槽位重新分配将会增加人力成本。在供料器启发式分配算法中， \mathcal{F}^{CP} 和 \mathcal{F}^{PT} 用于记录槽位基座上的元件类型和点数，其不仅适用于记录算法执行过程中已分配的供料器，还可进一步拓展至预分配的场景，为组装过程调度提供更为灵活的解决方案。在供料器分配过程中，贴片头会扫描基座上的供料器，以最大化利用全部贴片头对齐槽位中已分配的供料器进行同步取料，提升整体的组装效率。

2) 供料器的分配基准

在供料器启发式分配算法设计中，针对每一轮贴片头的扫描基座过程，在满足吸嘴分配模式约束下，最大化贴装点分配数量的方案并不唯一。因此，本小节引入了一项加权评估指标，以量化供料器安装于不同槽位时，贴片头拾取元件对其移动路径产生的影响。具体地，对于元件类型 i ，定义其所有贴装点平均坐标所对齐槽位为 \bar{S}_i ，计算方式为

$$\bar{S}_i \leftarrow \left(\sum_{p \in P} \eta_{ip} \cdot X_p / \sum_{p \in P} \eta_{ip} - X^{\text{Fl}} \right) \cdot \tau / \rho + 1 \quad \forall i \in I \quad (3-16)$$

当供料器分配过程遍历至槽位 s 时，将在该槽位的评估指标中附加一个权重较小的惩罚项 $-\sum_{h \in H} \left| s - \bar{S}_{\mathcal{H}_h^{\text{CP}}} \right|$ 。此策略可以减少贴片头完成拾取任务后移动向贴装区域的路径长度。

3) 供料器的可用数量

各类元件所对应的可用供料器资源有限，其也是决定拾取过程效率的重要因素之一。在电路板组装的准备环节中，具有较多贴装点的元件通常会配备更多的供料器以增加拾取效率。增加供料器的可用数量能提升对应元件的拾取效率，但分配过多同类元件的供料器会导致其占据最优的拾取槽位，进而增加拾取其他类型元件的移动距离。为单一元件拾取效率的最大化而过多分配供料器资源，反而可能导致整体拾取效率的下降。因此，在可用供料器数量上限的约束下，确定各类型元件分配供料器数的组合，也是提升组装效率的手段之一。本

算法 3-6 最小化拾取次数的可用供料器分配启发式

Algo.3-6 Available feeder assignment heuristic for minimizing the number of pickups

输入：元件的贴装点数 ψ ，元件的可用供料器数 ϕ
输出：元件的分配供料器数 $\hat{\phi}$

- 1 令元件分配的供料器数 $\hat{\phi}_i = 1, \forall i \in I$ ，当前遍历的已分配供料器数 $\tilde{\phi} \leftarrow \hat{\phi}$ ，最少可达成拾取次数 $\mathcal{J} \leftarrow \infty$;
- 2 **while** $\sum_{i \in I} \tilde{\phi}_i < \sum_{i \in I} \phi_i$ **do**
- 3 按 $\psi_i / \tilde{\phi}_i$ 值递减的顺序对元件索引 $i \in I$ 排序;
- 4 令 $c \leftarrow 0, g \leftarrow 1, p \leftarrow 0$; // c 为已遍历的元件组数, g 为当前遍历元件组索引, p 为各拾贴周期拾取次数总和
- 5 **for** $i \in I$ **do**
- 6 $c \leftarrow c + \tilde{\phi}_i$;
- 7 **while** $g \leq c$ **do**
- 8 $p \leftarrow p + \psi_i / \tilde{\phi}_i, g \leftarrow g + |H|$; // 记录元件分组后各拾贴周期的元件拾取次数
- 9 **end**
- 10 **end**
- 11 **if** $p \leq \mathcal{J}$ **then**
- 12 $\mathcal{J} \leftarrow p, \hat{\phi} \leftarrow \tilde{\phi}$; // 记录最少的拾取次数和对应的可用供料器分配方案
- 13 **end**
- 14 **for** $i \in I$ **do**
- 15 **if** $\tilde{\phi}_i < \phi_i$ **then**
- 16 $\tilde{\phi}_i \leftarrow \tilde{\phi}_i + 1$; // 增加未达到最大可用供料器上限的, 分组平均贴装点最多元件的可用供料器数
- 17 **break**;
- 18 **end**
- 19 **end**
- 20 **end**

小节提出一种最小化拾取次数的可用供料器分配启发式算法，具体实施流程如算法3-6所示。算法依据元件的可用供料器数上限及其贴装点数，采用逐轮递增的方式，调整元件的分配供料器实际数目。在此过程中，相同类型的元件被均分至不同的组，且均分后的元件被视为不同的类型。在随后执行贪心式的分组分配过程中，算法以所能达成的最少拾取次数作为评价指标，确定了最优的供料器数分配数量组合。

4) 供料器的类型差异

元件尺寸的多样性导致了其在供料器中所占槽位数的差异，为解决由此产生的占位冲突问题，本小节引入了额外的变量，记录供料器的占位信息。在执行供料器分配前，已分配供料器所占用的槽位会被预先记录。在供料器分配过程中，算法根据贴片头对齐槽位的元件分配方式采用不同的策略。对于分配同吸嘴类型一致的贴装点数最多的元件的情形，算法会增加额外的占位冲突约束，以确保分配的有效性；而对于采用堆栈弹出方式分配元件的情形，若某元件无法满足占位约束条件，则不会对其进行分配，并将从元件堆栈中移除，等待

后续的基座扫描重新进行分配。该策略在分配过程中增加了供料器占位冲突管理，保证了供料器槽位的准确分配和拾取效率的提升。

3.4.4 贴片头的元件分配启发式算法

3.4.4.1 基本框架

供料器-槽位分配和贴片头-元件分配在算法设计层面上的搜索机制具有相似性，均是基于启发式的基座扫描策略。前者解决的是供料器中元件的安装槽位问题，而后者解决的则是贴片头在不同拾贴周期拾取的元件类型和对应的拾取槽位的问题。启发式的扫描策略通过整合多个贴片头的拾取元件动作，有效地优化了同步拾取性能指标，缩减了整体拾取过程的用时。与供料器分配流程相对比，算法在基座扫描过程中逐一对单个贴片头进行元件分配，贴片头上分配的元件类型及其对齐槽位中的供料器，应满足如下条件：

- 1) 拾取可行性：贴片头对齐的槽位中存在尚未被拾取的贴装点；
- 2) 可用工具约束：当前拾贴周期贴片头装配的吸嘴总数不超过对应类型吸嘴的可用数量；
- 3) 前瞻收益最优：当前对齐槽位进行元件拾取的动作，其预期能达成的（同步）拾取收益不低于其他对齐槽位预期的（同步）拾取收益；
- 4) 拾取效率改进：执行拾取动作所带来的效率收益，需超过因更换吸嘴所产生的效率损失。

算法 3-7 介绍了贴片头元件分配启发式算法的实现过程，该过程采用基于扫描的拾取策略，旨在最大化同步参与元件拾取的贴片头数量，并尽量减少吸嘴更换的成本。扫描策略确定了各贴片头拾取的元件类型及其对应周期组内的拾贴周期数，有效规避了机械限位对拾取过程的制约，确保了所得解的可行性。元件分配的启发式方法具备前瞻性，即在为单个贴片头分配元件时，不仅计算了当前分配对性能指标的直接影响，还评估其对后续拾取过程效率的长远影响。扫描过程对拾取效率的提升主要体现在合并多个贴片头的拾取动作上，本文将因合并动作而节省的拾取时间与因吸嘴更换产生的额外用时之差称为拾取效率目标改进值。元件分配策略不仅以最大化拾取效率改进值为目标，还增加了长短期拾取收益项，以实现对其拾取效率的综合评估。在分配过程中，吸嘴模式用于确定贴片头从对齐槽位拾取元件所需的吸嘴更换次数。单次扫描的最优解确定后，贴片头所拾取的元件类型、周期组数及其装配的吸嘴信息被记录。其中，吸嘴的类型将作为新的吸嘴模式，应用于后续的扫描轮次。在记录贴片头装配的吸嘴类型时，算法会合并处理相同的结果，以避免因相同模式重复扫描

带来的运算效率的下降。

算法 3-7 贴片头-元件分配启发式算法

Algo.3-7 Head-component assignment heuristic algorithm

输入：供料器分配元件类型 \mathcal{F}^{CP} ，元件贴装点数 ψ ，元件-吸嘴匹配性 μ
输出：元件分配解 \mathcal{C} 和周期组索引集 \mathcal{K}

```

1  初始化一个大小为  $1 \times |H|$  的空列表  $n$  为初始吸嘴模式，添加  $n$  至吸嘴模式集  $m$ ；
2  while  $\sum_{i \in I} \psi_i \neq 0$  // 仍有未分配到贴片头的点
3  do
4      令  $V^* \leftarrow 0$  作为分配最优解的值，初始化  $\mathcal{H}^{\text{CP}}$ ， $\mathcal{H}^{\text{NZ}}$  和  $\mathcal{H}^{\text{PT}}$  为大小  $1 \times |H|$  的
        空列表 // 分别表示贴片头分配的元件类型，吸嘴类型和贴装点数
        /* 逐层遍历所有吸嘴模式、所有槽位和所有贴片头 */
5      for  $n \in m$  do
6          for  $s \leftarrow 1$  to  $s \leftarrow |S| - (|H| - 1) \tau$  do
7              for  $h \in H$  do
8                  确定贴片头的对齐槽位及其元件类型  $s' \leftarrow s + (h - 1) \cdot \tau$ ， $i \leftarrow \mathcal{F}_{s'}^{\text{CP}}$ ；
                  /* 计算目标函数值，并同基准值  $V^{\text{BASE}}$  进行比较 */
9                   $V \leftarrow T^{\text{PU}} \cdot \min \{ \mathcal{H}_{h'}^{\text{PT}} \mid \mathcal{H}_{h'}^{\text{PT}} > 0, h' \in H \} \cup \{ \psi_i \}$ 
                   $- T^{\text{NZ}} \cdot \sum_{h' \in H} |n_{h'} - \sum_{j \in J} j \cdot \mu_{ij}|$ ；
10                 if  $\psi_i > 0$  and  $V > V^{\text{BASE}}$  then
11                      $\mathcal{H}_h^{\text{CP}} \leftarrow i$ ， $\mathcal{H}_h^{\text{PT}} \leftarrow \psi_i$ ， $\mathcal{H}_h^{\text{NZ}} \leftarrow \sum_{j \in J} j \cdot \mu_{ij}$ ， $\mathcal{H}_h^{\text{FD}} \leftarrow s + (h - 1) \cdot \tau$ ；
12                 end
13             end
14             用算法 3-8 计算短期目标  $V^{\text{SO}}$  和长期目标  $V^{\text{LO}}$ ，记  $\omega$  为长短期目标权重
                系数，满足  $0 \leq \omega \leq 1$ ；
15             if  $\omega \cdot V^{\text{LO}} + (1 - \omega) \cdot V^{\text{SO}} > V^*$  then
16                 记录贴片头分配的最优值及解  $V^* \leftarrow \omega \cdot V^{\text{LO}} + (1 - \omega) \cdot V^{\text{SO}}$ ， $s^* \leftarrow s$ ；
17                  $\tilde{\mathcal{H}}^{\text{PT}} \leftarrow \mathcal{H}^{\text{PT}}$ ， $\tilde{\mathcal{H}}^{\text{CP}} \leftarrow \mathcal{H}^{\text{CP}}$ ， $\tilde{\mathcal{H}}^{\text{NZ}} \leftarrow \mathcal{H}^{\text{NZ}}$ ， $\tilde{\mathcal{H}}^{\text{FD}} \leftarrow \mathcal{H}^{\text{FD}}$ ；
18             end
19         end
20     end
21     令  $\ell \leftarrow \min \{ \tilde{\mathcal{H}}_h^{\text{PT}} \mid \tilde{\mathcal{H}}_h^{\text{PT}} > 0, h \in H \}$ ；
22     foreach  $h \in H$  do  $s \leftarrow s^* + (h - 1) \cdot \tau$ ， $i \leftarrow \mathcal{F}_s^{\text{CP}}$ ， $\psi_i \leftarrow \psi_i - \ell$ ；
23     if  $\sum_{h \in H} \tilde{\mathcal{H}}_h^{\text{PT}} > 0$  or  $\sum_{s \in S} \mathcal{F}_s^{\text{PT}} = 0$  then
24         添加  $\tilde{\mathcal{H}}^{\text{CP}}$  至元件分配解  $\mathcal{C}$ ，添加  $\ell$  至周期组索引集  $\mathcal{K}$ ，不重复地添加
             $\tilde{\mathcal{H}}^{\text{NZ}}$  至吸嘴模式集  $m$ ；
25     end
26 end

```

在扫描过程中，各贴片头对齐槽位中所包含的元件贴装点存在差异，长短期目标旨在反映不同拾取策略下元件拾取次数的变化。其中，短期拾取目标指的是在拾取过程中，同时从所有贴片头的对齐槽位中拾取元件，直至至少有一个槽位内贴装点全部被拾取完。对于在此轮拾取中未被完全取空的槽位，可以在后续的扫描环节中与其他槽位构成同步拾取；而长期拾取目标则采取一种贪心策略，即从对齐的槽位中拾取全部贴装点，其拾取次数等于所有对齐槽位中元件贴装数的最大值。在短期目标中，重点在于合并所有贴片头的拾取动作，以

实现整体上的拾取效率的提升。相比之下，长期目标则更侧重于当前对齐槽位所能达成的同步取料，而不考虑和其他槽位构成的同步拾取。短期目标可以被视为对当前贴片头-元件分配结果的即时反馈，而长期目标则是对整个拾取过程效率的全局性评估。算法3-8介绍了长短期拾取子目标的计算方法，其中的目标值是指贴片头所合并的拾取次数。通过长短期拾取子目标的加权和，可以实现对当前的贴片头-元件分配结果的综合性评估。

算法 3-8 长短期拾取子目标计算函数

Algo.3-8 Long- and short-term sub-objective of pick-up calculation functions

 输入：贴片头分配的元件类型 \mathcal{H}^{CP} ，贴装点数 \mathcal{H}^{PT} ，吸嘴类型 \mathcal{H}^{NZ} ，吸嘴模式 n

 输出：短期目标 V^{SO} 和长期目标 V^{LO}

```

1 记  $\bar{h}$  为贴装点非空的贴片头数， $n$  为吸嘴更换数，有
    $\bar{h} = |H| - |\{h \mid \mathcal{H}_h^{\text{PT}} > 0, h \in H\}| - 1$ ， $n = \sum_{h \in H} |n_h - \mathcal{H}_h^{\text{NZ}}|$ ；
2 令  $V^{\text{SO}} \leftarrow T^{\text{PU}} \cdot \bar{h} - T^{\text{NZ}} \cdot n$ ， $V^{\text{LO}} \leftarrow -T^{\text{NZ}} \cdot n$ ；
   /* 贴片头上仍有分配的贴装点 */
3 while  $\sum_{h \in H} \mathcal{H}_h^{\text{PT}} \neq 0$  do
4     令最小拾贴周期数  $\ell \leftarrow \min \{\mathcal{H}_h^{\text{PT}} \mid \mathcal{H}_h^{\text{PT}} > 0, h \in H\}$ ；
5     令  $V^{\text{LO}} \leftarrow V^{\text{LO}} + T^{\text{PU}} \cdot \bar{h} \cdot \ell$ ， $\mathcal{H}_h^{\text{PT}} \leftarrow \mathcal{H}_h^{\text{PT}} - \ell$ ， $\forall h \in \{h' \mid \mathcal{H}_{h'}^{\text{PT}} > 0, h' \in H\}$ ；
6     令  $\bar{h} = |H| - |\{h \mid \mathcal{H}_h^{\text{PT}} > 0, h \in H\}| - 1$ ；
7 end
```

3.4.4.2 实现细节

为增强贴片头-元件分配启发式的适用范围，提升算法的运算效率和贴片头的拾取过程的效率，本小节从以下几个角度对算法做补充说明：

1) 同一周期内同类元件的拾取

同一拾贴周期内，不同贴片头能从各自的对齐槽位中拾取相同类型的元件。算法3-7讨论了元件仅由单一供料器提供的情形。对于元件有多个供料器可用时，在进行贴片头-元件分配之前，一种预扫描策略用于处理同类元件的拾取分配，该策略通过确定对齐槽位中各类型元件对应的贴片头数，将贴装点进行均分。均分后的点数代表槽位内供料器的实际贴装点数，该值仅用于计算后续的目标函数值以及贴片头的拾取点数。

2) 贴片头可拾取元件类型的限制

为满足不同封装元件的组装需求，贴片头装配有多种类型的吸嘴。机械结构的特殊设计可能导致某些特定类型的吸嘴只能安装在指定的贴片头上，从而限制了其所能拾取的元件类型。为解决这一问题，算法在扫描分配元件过程中可限定贴片头可拾取元件的集合。此外，当贴片头无法正常拾贴元件时，扫描算法也可通过将可分配元件集合置空，以实现禁用贴片头的功能。

3) 连续作业的吸嘴更换指标

在吸嘴更换过程中，仅考虑吸嘴模式的匹配是不充分的。一个更为准确的计算方式是考虑连续作业带来的吸嘴更换，即当前分配的元件组装所需的吸嘴更换与下一周期的吸嘴更换。记 n 为当前匹配的吸嘴模式、 \tilde{n} 为下一组的吸嘴模型，则目标函数 V 被调整为

$$V \leftarrow T^{\text{PU}} \cdot \min \{ \mathcal{H}_h^{\text{PT}} \mid \mathcal{H}_h^{\text{PT}} > 0, h \in H \} \cup \{ \psi_i \} - T^{\text{NZ}} \cdot \left(\sum_{h \in H} \left| n_h - \sum_{j \in J} j \cdot \mu_{ij} \right| + \sum_{h \in H} \left| \tilde{n}_h - \sum_{j \in J} j \cdot \mu_{ij} \right| \right) / 2 \quad (3-17)$$

4) 拾取效率目标改进值的基准

在为贴片头分配元件时，算法的基本原则是同步拾取的收益大于吸嘴更换的成本，即 $V^{\text{BASE}} = 0$ 。在吸嘴类型较多的组装任务中，更换吸嘴是必要的组装动作，而部分元件贴装点数较少可能导致拾取效率目标改进无法产生正收益，进而使搜索过程陷入死循环。因此，若在一轮搜索内算法未找到新解，则需动态下调 V^{BASE} 的值，以确保搜索过程能够继续进行。

5) 拾取过程的贴片头移动距离

贴片头拾取过程中的移动路径同样会影响组装效率，在槽位相近的供料器中拾取元件可减少拾取过程的移动路径。因此，本小节在目标函数的评估 V 中引入了一个拾取移动的加权项。贴片头的拾取移动距离 \bar{D}^{PU} 由供料器的槽位和当前遍历的槽位 s 共同决定，即

$$\bar{D}^{\text{PU}} \leftarrow \mathcal{R} \left(\{s\} \cup \{ \mathcal{H}_h^{\text{FD}} - (h-1) \cdot \tau \mid \mathcal{H}_h^{\text{FD}} > 0, h \in H \} \right) \quad (3-18)$$

式中， $\mathcal{R}(\cdot)$ 表示集合中最大值和最小值的差。

3.5 实验设计

为说明本章提出算法的实际效果，本节选取了四种主流的表面组装过程优化方法进行对比实验验证，其中包括 Ashayeri 等^[17] 提出的聚合混合整数规划法 (Aggregated Mixed Integer Programming, AMIP)、Guo 等^[18] 提出的基于同步拾取组的混合遗传算法 (Hybrid Genetic Algorithm, HGA)、Li 等^[29] 提出的元胞分裂遗传算法 (Cell Division Genetic Algorithm, CDGA) 以及一个内置于工业软件中的元件贴装优化器 (Component Placer Optimizer, CPO)。对比实验的选取兼顾了数学规划法和进化算法等多种方法。其中，HGA 和 CDGA 是基于进化算法的组装过程优化的代表方法，AMIP 则是一种结合元件批处理技术的数学规划方法，

能在可接受的时间内优化中等规模的数据。本章提出的 CTIM 和 AMIP 模型由于变量和约束的规模差异较大,因而终止搜索时间分别设定为 1 小时和 100 秒。实验所用的数据均来源于某生产车间的表面组装生产线,同表面组装过程相关的优化子目标权重与第 2 章相同。

3.5.1 基于模型的贴片头任务分配的对比实验

基于模型的贴片头任务分配过程适用于处理元件类型有限的中小规模数据。表3-1中列出了 10 组用于比较 PCB 数据组 2 的基本参数,所有元件的供料器类型相同,算法通过吸嘴类型、元件类型、不同元件的可用供料器数和贴装点数确定一个贴片头任务分配的结果,部分贴装点数较多的元件配备有多个供料器以提高组装效率。基于模型的优化算法在搜索过程中保留多个最优解,搜索池的大小设置为 30,解池的容差为 10^{-4} ,辅助项 N 的大小等于数据的贴装点数。对于中等规模的数据,完全求解模型是一个相当耗时的过程,CTIM 的搜索终止条件在原有设定上增加了模型最优解的未改变时间 \mathcal{T}^{UC} 满足式 (3-19)。

$$\mathcal{T}^{\text{UC}} > 0.9 \cdot \left(1 - \frac{\odot^{\text{BND}}}{\odot^{\text{BST}}}\right) \quad (3-19)$$

其中 \odot^{BST} 和 \odot^{BND} 分别为模型求解过程中的最好下界和最优已知可行目标值。

本小节首先比较了基于 CTIM 的贴片头任务分配算法和其他主流研究算法拾贴周期数、同步拾取数和吸嘴更换数的加权值,如表3-2所示。比较结果中,工业软件内置的 CPO、基于数学规划的 AMIP 以及基于进化算法的 HGA 和 CDGA 对应解的加权目标值分别为 \odot^{CPO} , \odot^{AMIP} , \odot^{HGA} 和 \odot^{CDGA} , CTIM 解的加权目标值记为 \odot^{CTIM} 。由表可知,本章提出的基于 CTIM 的算法和同样考虑了拾取效率的 CPO、HGA 和 CDGA 在组装效率上明显优于仅优化了拾贴周期和吸嘴更换的 AMIP; HGA 和 CDGA 在迭代搜索过程中有随机成分,结果输出上具备不确定性,整体的性能指标上也差于 CTIM 和 CPO。整体上而言,基于 CTIM 的算法在加权组装效率目标方面优于其他主流研究,相较于 CPO、HGA、AGM 和 CDGA 在加权组装优化目标上分别提升了 30.71%、18.13%、35.99% 和 28.21%。

表 3-1 中小规模 PCB 数据的基本参数

Table 3-1 Basic parameters of small- and medium-scale PCB data

PCB	2-1	2-2	2-3	2-4	2-5	2-6	2-7	2-8	2-9	2-10
吸嘴类型数	1	2	2	3	2	3	3	4	4	4
元件类型数	3	4	5	5	6	7	8	9	10	12
供料器数	5	4	6	6	8	8	8	9	11	15
贴装点数	288	352	306	384	375	232	194	148	196	160

表 3-2 周期任务集成模型与其他主流算法的目标值比较

Table 3-2 Comparison of the objectives value between cyclic task integration model and other mainstream algorithms

PCB	CTIM	CPO		HGA		AMIP		CDGA	
	$\mathcal{O}^{\text{CTIM}}$	\mathcal{O}^{CPO}	$\mathcal{G}(\%)$	\mathcal{O}^{HGA}	$\mathcal{G}(\%)$	$\mathcal{O}^{\text{AMIP}}$	$\mathcal{G}(\%)$	$\mathcal{O}^{\text{CDGA}}$	$\mathcal{G}(\%)$
2-1	30.912	35.682	15.43	43.824	41.77	38.544	24.69	39.659	28.30
2-2	42.504	51.793	21.85	46.178	8.64	42.663	0.37	49.448	16.34
2-3	29.052	42.948	47.83	31.008	6.73	48.609	67.32	45.766	57.53
2-4	38.736	40.880	5.53	44.556	15.02	46.368	19.70	54.419	40.49
2-5	39.913	48.786	22.23	59.749	49.70	62.745	57.20	47.530	19.08
2-6	25.824	47.250	82.97	33.584	30.05	43.632	68.96	37.064	43.53
2-7	25.710	41.964	63.22	25.188	-2.03	37.248	44.88	29.429	14.47
2-8	18.748	23.076	23.09	21.610	15.27	22.564	20.35	22.866	21.97
2-9	25.029	26.664	6.53	26.174	4.57	32.160	28.49	32.227	28.76
2-10	19.368	22.925	18.37	21.602	11.53	24.774	27.91	21.626	11.66
AVG			30.71		18.13		35.99		28.21

周期任务集成的建模策略提升了模型的可解性，实现了对问题整个解空间更全面的搜索。

降低模型复杂度和构造模型初始解是提升模型求解效率的有效途径，本小节将未进行模型增强（Non-Enhanced, NE）和未进行初始化（Non-Initialized, NI）的 CTIM 同原模型进行了比较。在对比收敛过程时，实验所用的模型求解器仅能以秒为单位输出求解过程的中间结果，因而图3-3中同一秒内的模型迭代收敛值为均匀间距插值的结果，与实际模型计算时间存在一定的偏差。图3-3中的实线、点划线和双划线分别对应原模型、未进行增强模型和未进行初始化模型的迭代过程，虚线则表示模型搜索初始解的截断部分。对于贴片头任务分配模型，找到模型的初始解和完成模型的求解同样困难，PCB 数据中 2-3、2-5、2-7 和 2-8 用了整个求解过程接近或超过半数的时间去构造模型的初始解。模型的初始解虽然在理论上不改变模型的最优解，但其也是影响搜索过程收敛速度的重要因素。对于 PCB 数据 2-6，求解器构造的初始解比启发式初始解更快地收敛至最终解，而对于 PCB 数据 2-8，求解器构造的初始解收敛速度则相对较慢，导致最终解的质量仍无法收敛到原模型的解。除此之外，其他数据的启发式构造初始解均能使模型更快地收敛至最终解，提升了模型的求解效率。在构造模型初始解时，本章提出的启发式构造算法仅对模型的主要决策变量进行了初始化，因而模型在不使用增强策略时确定的初始解质量相对较差。模型增强有助于快速迭代收敛，PCB 数据中 2-6、2-7、2-8 和 2-9 的最终迭代值仍与原模型的解存在明显偏差，且未增强的模型收敛速度更慢、迭代用时更长。

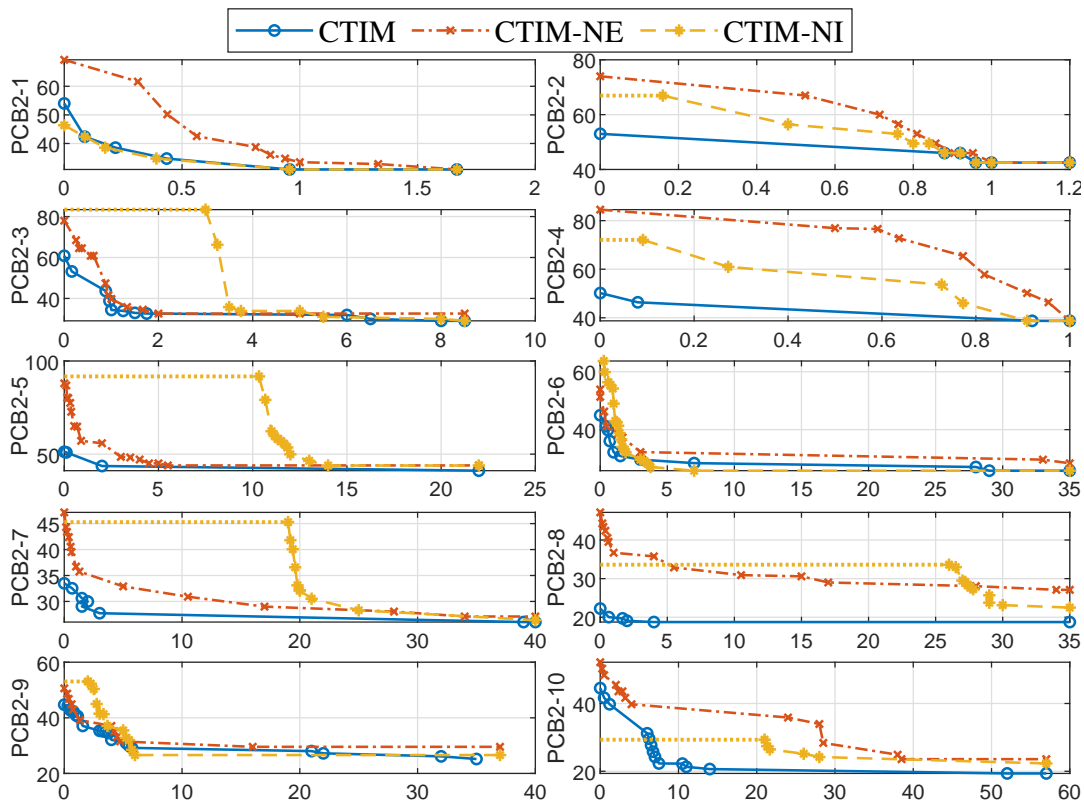


图 3-3 不同模型设定条件下的目标值迭代收敛曲线

Fig.3-3 Iterative convergence curves of objective values under different model settings

最后，本小节对比分析了多种优化策略在解决小规模 PCB 数据时的运算效率，如表3-3所示。表中增加了单拾贴周期模型（Single PAP Model，SPAPM）项，其周期组内的拾贴周期数设定为 1，用以说明周期任务分组策略的必要性。SPAPM 的迭代过程直至满足以下任一条件终止：模型求解时长超过 1 小时，或模型的解与 CTIM 解的一致。在比较求解时间时，由于工业软件中 CPO 的核心

表 3-3 周期任务集成模型与其他主流算法的求解时间比较

Table 3-3 Comparison of solving time between cyclic task integration model and other mainstream algorithms

PCB	SPAPM (s)	CTIM (s)	HGA (s)	AMIP (s)	CDGA (s)
2-1	16.16	5.46	49.84	0.09	5.18
2-2	109.83	6.95	38.36	0.11	4.05
2-3	142.37	36.13	30.85	0.27	2.92
2-4	205.47	12.01	36.25	0.28	4.67
2-5	1044.94	90.47	61.51	0.95	2.82
2-6	—	81.71	41.31	0.58	2.13
2-7	766.49	113.90	38.90	1.33	2.56
2-8	—	60.75	21.13	1.84	2.51
2-9	—	92.72	28.61	1.04	3.21
2-10	—	99.52	28.68	1.01	3.10

算法未公开，加之不同实现方式间存在显著的效率差异，且集成的优化器包含数据输入、过程优化、结果输出等多个环节，难以准确获得优化求解时间，故 CPO 未纳入本次对比分析之中。相较于其他方法，AMIP 求解策略因在建模时简化了影响组装效率的多重因素考量，模型结构相对简单，从而有最高的求解效率。CDGA 求解策略在处理简单数据时，得益于其高效的元胞分裂机制，同样保持了较高的求解速度。然而，HGA 由于需同时考虑拾取组的组合优化，其求解效率不及 AMIP 和 CDGA，尽管如此，其求解效率受问题规模增长的影响相较于 CTIM 而言较小。

周期任务分组作为一种有效提升搜索效率的手段，相较于 SPAPM，CTIM 具备更快的收敛速度，显著缩短了求解时间。对于 PCB 数据 2-6 和 2-8，SPAPM 模型的收敛速度显著放缓，且其解与 CTIM 的解之间存在持续偏差值；而对于 PCB 数据 2-9 和 2-10，由于问题规模的扩大导致模型复杂度急剧增加，求解器甚至无法找到 SPAPM 模型的初始可行解，更无法对模型进行迭代搜索。此外，随着问题规模的逐步扩大，本章所提出的 CTIM 在求解时间呈现上升趋势，说明其不再适用于解决大规模问题。

3.5.2 基于启发式的贴片头任务分配的对比实验

基于启发式的贴片头任务分配算法在参数设定方面与 CTIM 求解方法一致。启发式的优化方法无法在理论上保证解的最优性，为定量分析启发式算法解的质量，本小节首先围绕主要子目标的加权值及运算时间等指标在 PCB 数据组 1 上进行比较，对比 HLFISH 算法和基于模型的 CTIM 算法在中小规模数据优化时的表现，算法的主要性能指标加权值分别记为 \odot^{HLFISH} 和 \odot^{CTIM} ，优化运算时间分别记为 $\mathcal{E}^{\text{HLFISH}}$ 和 $\mathcal{E}^{\text{CTIM}}$ ，结果如表3-4所示。HLFISH 优化过程是以增加拾贴周期数为代价提高同步拾取次数，较好地兼顾了不同子目标之间的优化结果。根据子目标加权后所得的目标函数值可知，本章所提出的 HLFISH 算法与 CTIM 相比，平均差距为 5.04%，最大差距为 9.85%，这表明本章设计的启发式算法能达到或逼近模型求解的同等效果。基于启发式的 HLFISH 算法同时大幅缩短了运算时间，相较于 CTIM，HLFISH 的问题求解时间平均缩短了 97.41%。随着模型规模的增大，基于模型的 CTIM 运算时间大幅增加，而 HLFISH 优化方法运算时间的增幅较小，这一特性使得该方法能够进一步拓展至更大规模的数据优化问题中，说明了启发式算法的实用性和可扩展性。

为验证基于启发式的贴片头任务分配算法在大规模数据上的表现，10 组具有不同特性的 PCB 数据组 3 用于进一步比较 HLFISH 算法的实际效果，其参数

表 3-4 基于启发式的贴片头任务分配结果和模型的解比较
Table 3-4 Comparison of solution of model-based and heuristic-based algorithm for head task assignment

PCB	目标值			运算时间		
	$\mathcal{O}^{\text{CTIM}}$	$\mathcal{O}^{\text{HLFSH}}$	$\mathcal{G}(\%)$	$\mathcal{E}^{\text{CTIM}}(\text{s})$	$\mathcal{E}^{\text{HLFSH}}(\text{s})$	$\mathcal{G}(\%)$
2-1	30.912	33.456	8.23	5.46	0.07	98.75
2-2	42.504	46.270	8.86	6.95	0.37	94.71
2-3	29.052	30.324	4.38	36.13	0.27	99.24
2-4	38.736	39.372	1.64	12.01	0.73	93.92
2-5	39.913	40.962	2.63	90.47	0.55	99.39
2-6	25.824	28.368	9.85	81.71	0.84	98.97
2-7	25.710	26.118	1.59	113.90	1.13	99.01
2-8	18.748	20.179	7.63	60.75	2.18	96.41
2-9	25.029	25.800	3.08	92.72	3.06	96.69
2-10	19.368	19.845	2.46	99.52	2.96	97.03
AVG			5.04			97.41

如表3-5所示。本小节选取的 PCB 数据具有更多的元件类型和贴装点数，供料器的可用数量根据元件贴装点数和元件类型确定，且所有元件的供料器类型相同。PCB 数据的复杂性主要体现在两个方面：一是贴装点数较多，二是元件类型更多样。为优化更大规模数据的组装过程，基于 CTIM 的算法通过整合周期任务的策略，有效地解决了在有限元件类型条件下，贴装点数量较多的组装过程中的优化问题。相比之下，启发式的搜索策略则将重点放在了优化供料器分配槽位、平衡贴片头拾取过程的子目标指标上。启发式方法通过灵活调整搜索策略，在面对复杂元件类型和大规模 PCB 数据时，依然保持较高的搜索效率和优化性能。

表 3-5 大规模 PCB 数据的基本参数
Table 3-5 Basic parameters of large-scale PCB data

PCB	3-1	3-2	3-3	3-4	3-5	3-6	3-7	3-8	3-9	3-10
吸嘴类型数	1	1	2	3	3	3	4	3	4	4
元件类型数	7	18	13	40	16	20	24	42	29	40
可用供料器数	9	18	18	50	16	28	26	46	31	42
贴装点数	564	176	264	546	114	150	236	384	165	204

表3-6进一步对比了 HLFSH 算法与其他主流优化策略在加权目标值方面的表现。具体而言，HLFSH 算法与工业软件所用的 CPO 在平衡关键性能指标方面展现出显著优势，其加权组装效率相对更高。相较于 CPO，HLFSH 在处理复杂数据时能够取得更优的结果。此外，针对提升拾取效率的策略，HGA 和 CDGA 开展了相关研究，对于元件类型较多的数据，在同样以进化算法为基础的算法框架下，HGA 在整体组装效率上有出更优的表现。CDGA 在处理元件类型数相

对较少的 PCB 数据 3-1、3-3 和 3-5 时，子目标加权值较优，而随着元件类型数的增加，CDGA 优化所得的解的加权目标值与 HLFSSH 的差距也呈现增大的趋势。基于元启发式的 CDGA 以最小化周期数为主要目标，忽视了元件类型增多所导致的频繁吸嘴更换，而基于数学规划的 AMIP 在构建优化目标时未能充分优化拾取过程，导致这两种优化方法相较其他算法在解决大规模数据优化时组装效率最低。

表 3-6 分层前瞻扫描启发式与其他主流算法的加权目标值比较

Table 3-6 Comparison of weighted objectives value of hierarchical look-forward scanning heuristic with other mainstream algorithms

PCB	HLFSH	CPO		HGA		AMIP		CDGA	
	\odot_{HLFSH}	\odot_{CPO}	ζ (%)	\odot_{HGA}	ζ (%)	\odot_{AMIP}	ζ (%)	\odot_{CDGA}	ζ (%)
3-1	55.130	54.812	-0.58	71.748	30.14	99.968	81.33	75.826	37.54
3-2	18.048	18.048	0.00	20.640	14.36	29.496	63.43	26.542	47.07
3-3	26.484	41.284	55.88	34.792	31.37	38.727	46.23	26.760	1.04
3-4	50.058	53.657	7.19	70.013	39.86	90.444	80.68	91.000	81.79
3-5	13.230	15.565	17.65	14.502	9.61	17.841	34.85	16.074	21.50
3-6	17.245	18.642	8.10	23.637	37.07	19.312	11.99	31.308	81.55
3-7	23.081	25.901	12.22	26.174	13.40	41.525	79.91	46.634	102.04
3-8	40.477	44.694	10.42	51.906	28.24	70.369	73.85	108.037	166.91
3-9	19.368	23.569	21.69	21.133	9.11	30.498	57.47	52.933	178.19
3-10	24.663	26.488	7.40	26.110	5.87	35.793	45.13	68.607	178.18
AVG			14.00		21.90		57.49		89.12

图3-4展示了不同优化算法在处理大规模数据时主要性能指标的对比柱状图。在表面组装过程中，吸嘴更换动作是一项相对耗时的任务，因此，HLFSH、CPO 及 HGA 等算法在搜索过程中均采用尽可能减少吸嘴更换动作的策略。具体而言，HLFSH 通过构建更多的拾贴周期来规避吸嘴更换，同时确保了更少的拾取元件次数；与之相对，CPO 则以少量吸嘴更换为代价平衡了周期规划与拾取效率，实现了较优的性能。HGA 在优化中禁止吸嘴更换以提升同步拾取能力，然而，在面对元件类型数较多的情况时，其对同步拾取组的合并处理能力受限，导致整体拾取效率仍低于 HLFSSH 和 CPO。另一方面，CDGA 在元胞组合阶段引导供料器布局的分配，对拾取效率产生了一定程度的提升，但这一提升是以增加吸嘴更换次数为代价的，特别是在元件类型数较多时吸嘴更换次数会显著增加，因此，其组装效率也未能超越 HLFSSH 和 CPO。AMIP 因未考虑同步取料的设计，在对比方法中组装效率较低。此外，受限于数据规模，在处理 PCB 数据 3-8、3-9 和 3-10 时，AMIP 在限定时间内无法求解出模型的最优解，也降低了模型解的质量。

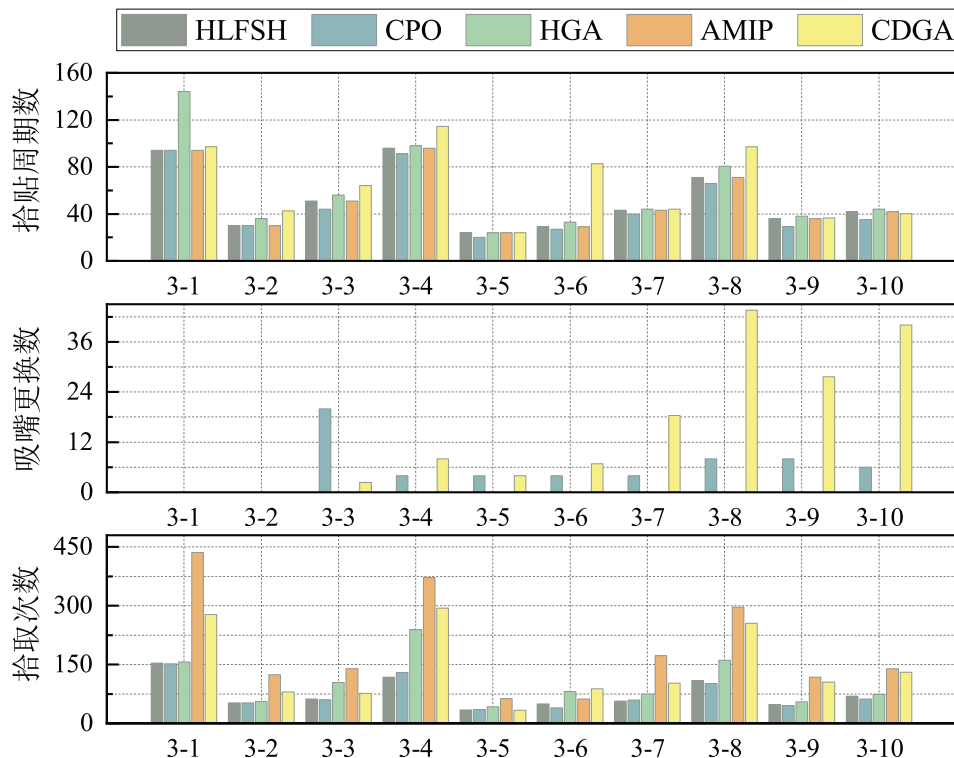


图 3-4 分层前瞻扫描启发式与其他主流算法的子目标柱状图对比

Fig.3-4 Bar chart of sub-objectives comparison of hierarchical look-forward scanning heuristic with other mainstream algorithms

相较于主流研究方法，本章提出的 **HLFSH** 算法和工业软件内置的 **CPO** 均适用于不同的生产配置设定，包括预先安装供料器、多类型供料器槽位分配等情形。三种不同的测试用例用于比较供料器安装槽位约束下优化算法的实际效果。其中，用例 1 根据 Guo 等^[18]提出的构建同步拾取组的方法进行供料器槽位分配，该供料器分布位置更接近于现场操作人员按照经验所安装的槽位；用例 2 和用例 3 则用于交叉验证 **HLFSH** 和 **CPO** 优化所得的供料器槽位分配作为约束时，两种优化算法在对方分配结果约束下的优化效果。需要指出，用例 1 的供料器分配方式并不能充分利用全部可用供料器，**HLFSH** 和 **CPO** 均需要根据已有供料器安装布局，对未分配的多余供料器进行分配，以提高拾取效率。基于不同的供料器排布用例的优化目标值比较结果如表 3-7 所示。由表可知，在加权性能指标上，不同的供料器布局会对最终优化结果搜索方向产生影响，**HLFSH** 的整体表现优于 **CPO**。此外，本章提出的 **HLFSH** 将供料器分配和贴片头元件拾取分别讨论，因而采用其优化所得的供料器安装槽位重新优化时，优化结果将保持不变，这是 **CPO** 所不具备的。

图 3-5 比较了 **CPO** 和 **HLFSH** 算法在不同供料器配置下主要子目标的值，其中，“C”和“H”分别代表 **CPO** 和 **HLFSH**。在供料器布局 1 的设定下，**HLFSH** 和

表 3-7 分层前瞻启发式和工业求解器在不同供料器布局下优化加权目标值的比较
Table 3-7 Comparison of weighted objective values between hierarchical look-forward scanning heuristic and industrial solver with different feeder arrangements

PCB	布局 1			布局 2			布局 3		
	\odot_{CPO}	\odot_{HLFSH}	$\mathcal{G}(\%)$	\odot_{CPO}	\odot_{HLFSH}	$\mathcal{G}(\%)$	\odot_{CPO}	\odot_{HLFSH}	$\mathcal{G}(\%)$
3-1	54.812	57.356	-4.44	54.812	54.812	0.00	58.628	55.130	6.35
3-2	18.684	18.684	0.00	18.048	18.366	-1.73	18.048	18.048	0.00
3-3	39.412	26.484	48.81	40.966	38.673	5.93	34.96	26.484	32.00
3-4	56.042	52.300	7.15	60.224	58.25	3.39	57.155	50.058	14.18
3-5	16.201	13.548	19.58	18.594	22.624	-17.81	15.247	13.230	15.25
3-6	19.437	17.245	12.71	29.593	21.202	39.58	20.081	17.245	16.45
3-7	28.922	23.240	24.45	25.901	25.901	0.00	26.696	23.081	15.66
3-8	47.413	41.272	14.88	53.11	48.217	10.15	47.715	40.477	17.88
3-9	29.937	19.527	53.31	23.728	24.564	-3.40	30.899	19.368	59.54
3-10	32.503	24.186	34.39	26.488	32.108	-17.50	28.237	24.663	14.49
AVG			21.09			1.86			19.18

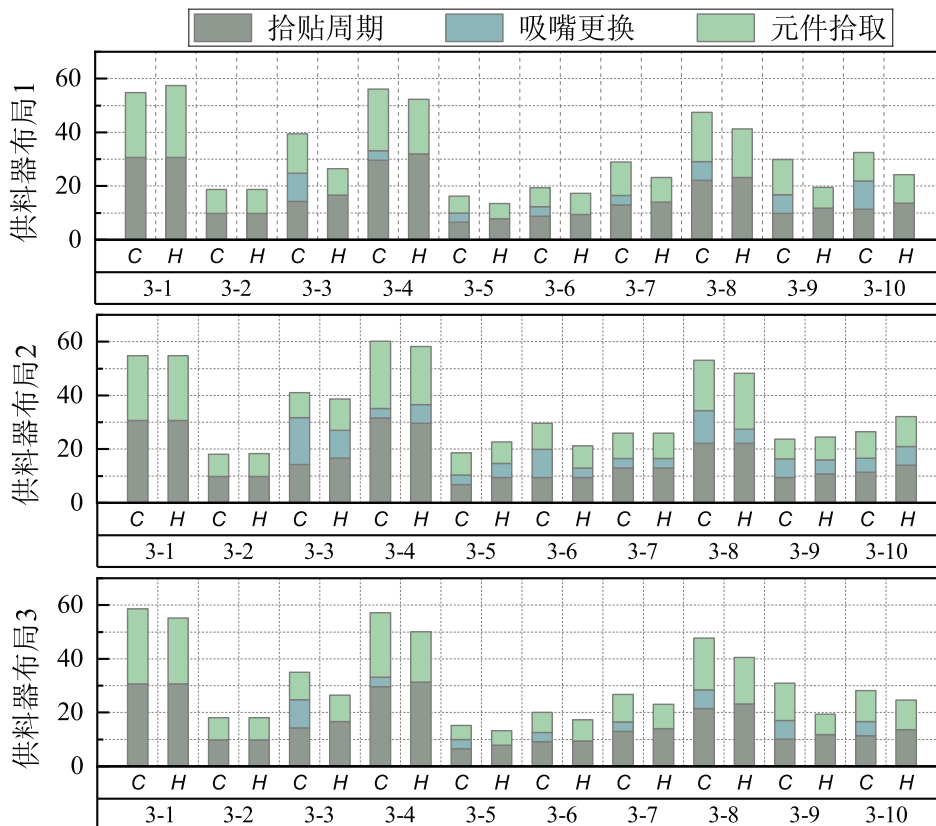


图 3-5 分层前瞻扫描启发式与工业求解器在不同供料器布局下的子目标柱状图对比
Fig.3-5 Bar chart of sub-objectives comparison between hierarchical look-forward scanning heuristic and industrial solver with different feeder arrangement

CPO 有较为接近的周期规划结果，而 HLFSSH 能以较少的吸嘴更换取得和 CPO 相接近的拾取效率，因而整体指标更优。在工业软件供料器布局 2 的设定下，CPO 能充分利用其分配的槽位达到更高的拾取效率，而 HLFSSH 为降低吸嘴更换，拾取效率也有所下降，HLFSSH 和 CPO 有着相近的表现。在本章提出的供料器布局 3 的设定下，CPO 在吸嘴更换和拾取效率方面均不占优，相较于 CPO，HLFSSH 以较小的拾贴周期数增加为代价提升了整体组装效率。

针对算法在大规模数据上的运行效率，本小节将不同优化算法的求解时间进行了对比分析，如表3-8所示。据此可知，HLFSSH 的优势在于能快速搜索大规模数据的高质量解，在所有对比方法中，HLFSSH 的搜索用时最短且随着问题规模的扩张增幅较小。基于进化策略的 HGA 与 CDGA 在搜索时间上则相对较长，两者相比较而言，CDGA 有更高的运算效率。基于数学规划的 AMIP 由于模型结构简单，在处理部分数据时，其运算时间接近 HLFSSH。然而，随着数据规模的持续扩大，AMIP 出现了在预定时间内无法求得模型最优解的问题。

表 3-8 分层前瞻扫描启发式与其他主流算法的求解时间比较

Table 3-8 Comparison of solving time of hierarchical look-forward scanning heuristic with other mainstream algorithms

PCB	HLFSSH (s)	HGA (s)	AMIP (s)	CDGA (s)
3-1	0.09	51.50	0.18	325.67
3-2	0.14	39.57	0.15	6.88
3-3	0.74	20.37	10.67	5.11
3-4	2.39	85.68	25.69	27.94
3-5	1.46	23.13	3.33	4.46
3-6	3.49	40.65	4.00	6.01
3-7	2.70	42.84	6.42	11.43
3-8	7.53	129.18	100.00	20.43
3-9	8.77	54.24	100.00	12.57
3-10	11.64	78.72	100.00	21.63

HLFSSH 和 CPO 均可应用于不同类型供料器的优化。表3-9列出了 PCB 数据组 4，共由 5 组数据构成，元件对应供料器具有不同占位数（槽位数）。其中，类型 1 供料器占据 1 个槽位，类型 2 和类型 3 供料器均占据 2 个槽位但具有不同的占位宽度，类型 4 供料器占据 4 个槽位，可用供料器数多于元件类型数。表3-10比较了 HLFSSH 和 CPO 在解决不同类型供料器贴片头任务分配时主要子目标和加权目标的值。整体上而言，HLFSSH 在拾取次数上更具优势，同时减少了非必要的吸嘴更换，HLFSSH 相较于 CPO 在加权子目标的平均差距了 20.03%。

表 3-9 具有不同类型供料器的 PCB 数据的基本参数
Table 3-9 Basic parameters of PCB data with different types of feeders

PCB	吸嘴 类型数	元件 类型数	贴装点数	可用供料器数			
				类型 1	类型 2	类型 3	类型 4
4-1	2	16	78	10	4	2	1
4-2	3	10	588	9	4	1	1
4-3	3	14	126	11	1	1	3
4-4	3	26	97	13	7	4	0
4-5	4	23	110	14	7	2	0

表 3-10 启发式前瞻扫描和工业优化器选用多种类型、不同宽度供料器的加权目标值比较
Table 3-10 Comparison of weighted objective values between hierarchical look-forward scanning heuristic and industrial solver with feeders of multiple types and width

PCB	周期		吸嘴更换		拾取次数		目标值		
	\mathcal{Q}_{CPO}	\mathcal{Q}_{HLFSH}	\mathcal{Q}_{CPO}	\mathcal{Q}_{HLFSH}	\mathcal{Q}_{CPO}	\mathcal{Q}_{HLFSH}	\mathcal{Q}_{CPO}	\mathcal{Q}_{HLFSH}	$\mathcal{G}(\%)$
4-1	15	22	4	0	31	31	13.299	12.101	9.90
4-2	98	110	24	4	168	189	79.540	69.391	14.63
4-3	24	24	0	0	50	36	15.774	13.548	16.43
4-4	27	30	4	0	63	48	22.299	17.412	28.07
4-5	19	21	4	0	52	43	17.942	13.683	31.13
AVG									20.03

3.6 本章小结

贴片头任务分配的结果决定了表面组装过程中效率相关的关键性能指标,本章分别从数学规划和启发式的角度研究了问题的求解方法。为降低贴片头任务分配模型的复杂度,本章提出了基于周期任务集成的增强模型,研究了模型的初始解生成方法,从限制决策变量规模和压缩可行解域的角度降低了模型的复杂度,提出了一种带有路径预判的模型解评估策略,用于降低模型和路径规划问题耦合性。为优化更大规模的数据,本章进一步提出了基于分层前瞻扫描启发式的贴片头任务分配算法,从构造的吸嘴分配模式出发,在供料器-槽位分配过程中,以平衡元件拾取和吸嘴更换为目标进行槽位分配;在贴片头-元件分配过程中,通过引入一系列约束条件、性能指标计算方式和前瞻性效率评估策略完成分配过程。最后,本章将提出的算法同主流研究进行了对比验证,结果表明,周期任务模型的改进策略提升了可解性,保证了解的近似最优性;启发式算法在中等规模数据上表现与模型解法相近但更高效,在大规模数据上解的质量和运算效率等指标优于主流研究。

第4章 表面组装过程中贴装过程的路径规划算法

4.1 引言

贴装过程的路径规划是研究悬臂贴装元件过程中移动路径最小化的算法,其结果同样是影响组装过程效率的因素之一,可被视为一类带约束的有容量限制的车辆路径规划问题(Capacitated Vehicle Routing Problem, CVRP)的变体。贴装过程的路径规划是在贴片头任务分配基础上的进一步决策规划,二者共同决定了完整的贴片机组装过程。路径规划的复杂性体现在既有贴片头元件分配结果的限制,又有多轴线性排列的贴片头偏移对移动位置的影响。较大的可行域使找到问题高质量解的难度增大,以贪心为代表的启发式优化算法搜索方式单一,但其获得的可行解基本能满足生产要求,因而相关研究较少且不够深入。对于大批量的表面组装任务,即使悬臂的贴装移动用时在整体组装过程中占比不高,但其累积产生的经济效益仍不可被忽视。

启发式的路径规划算法通常以最近邻贪心搜索方法为基础,适用于解决大规模的路径规划问题。贪心算法从随机选取的贴装点开始进行搜索,对于贴装路径规划任务,不同拾贴周期首个贴装点的选取会对当前及整体解的质量产生较大影响。为此,本章提出一种基于多源点启发式的贴片头路径规划算法,通过不同的起始搜索点和贴片头分配迭代顺序,增加搜索解的多样性。集束搜索作为贪心的扩展算法,其在搜索过程中采用广度优先的方式建立搜索树,并在搜索树的各层时按启发代价对结点进行排序,剪去代价值较低的结点,以被保留的多个相对较优的结点为基础在下一层次继续扩展,有助于算法跳出搜索过程的局部最优解,进而提升构造的路径规划解的质量。

贪心的搜索方式有一定的短视性,而邻域搜索可根据各拾贴周期内贴装点的分布特性,对已有解进行再规划。现有研究中,相关的元启发式算法已被广泛应用于解决此类路径规划问题中,且取得了较好的成效^[111-112]。作为 CVRP 的变体问题,聚类算法通过将访问点分组,并与邻域搜索算法相结合,将复杂的路径规划问题分解为易于处理的子问题,可以降低问题的复杂度、提高求解效率。为此,本章研究了自适应大邻域搜索在拾贴周期路径再规划中的应用,通过重新分配各拾贴周期的贴装点,平衡拾贴周期之间的移动路径长度,降低了贴装过程悬臂移动路径的总长度。

4.2 问题分析

贴装过程作为多轴协同运动的过程，其特殊性在于连续两点贴装作业的用时不仅取决于贴装点之间的距离，同时与 $X/Y/R$ 轴协同工作有关。在贴装过程中，各贴装位置之间的 Chebyshev 距离决定了移动过程的用时；而贴装过程的用时不仅取决于选取的贴装点位置，也与同轴贴片头连续作业所完成的 R 轴旋转作用时有关。如图4-1所示，在贴片头机械设计中，其紧凑的结构使每两个相邻的贴片头共用一个 R 轴电机，并通过传动齿轮带动贴片头转动。在表面贴装过程中，相机识别拾取元件的角度偏差后，贴片头可预先旋转至待贴装角度，因而部分贴片头 R 轴旋转用时并不影响整体组装用时。然而，同轴旋转动作的存在也使得部分贴片头需等待上一个点贴装完成后，才可以旋转到预定的贴装角度。多轴电机独立工作的最长用时决定了组装过程的移动用时。

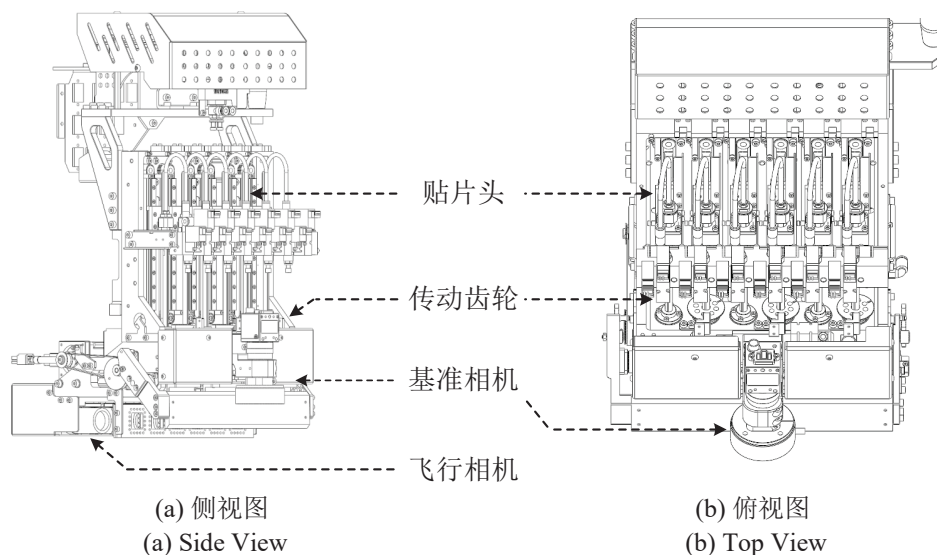


图 4-1 贴片头的结构示意图

Fig. 4-1 Schematic diagram of the structure of heads

贴装过程的路径规划较 CVRP 有更多的限制条件。贴片头任务分配的结果优化了影响组装效率的主要指标，同时限制了拾贴周期可访问路径可行解的范围。对于给定的拾贴周期，各贴片头的拾取元件类型已确定，限制了其可访问的贴装点集。路径规划的容量受限体现在各拾贴周期分配有元件的贴片头数是确定的。贴装点选择与其所对应的贴片头对路径规划也有影响，受线性排列结构的影响，不同贴片头吸取同类型元件时，其贴装同一贴装点会存在位置偏差。贴装过程路径规划可被视为是贴装点分配与贴装顺序规划相结合的组合问题。拾贴周期是路径规划的最小单位，算法通过确定单个拾贴周期内各贴片头访问的点及访问顺序，实现最小化悬臂移动路径长度的优化目标。贴装过程路

径规划涉及三层决策问题，分别是：拾贴周期贴装点分配，贴装点的贴片头分配和贴片头作业的先后顺序规划，分层决策降低了问题的求解难度。

贴装过程路径规划的三层决策问题存在一定的耦合关系，在算法设计中，本章以贴装点分配到的拾贴周期和贴片头为第一阶段优化问题，以各拾贴周期的路径规划为第二阶段优化问题。为使移动路径长度最小化，贪心的搜索过程仅用于确定各拾贴周期贴片头访问的贴装点，单一周期内相对密集的贴装点分布有助于降低整体移动路径的长度；动态规划法可最优地解决各拾贴周期的路径规划问题。基于聚类的自适应邻域搜索规划法用于进一步改进解的质量，在贴片头任务分配的约束下，从全局角度对拾贴周期路径规划较差的解进行再规划，能有效地克服贪心算法的局限性。

4.3 贴装过程的路径规划方法

4.3.1 算法概述

本节分别介绍了基于启发式和邻域搜索的贴装过程路径规划算法，启发式算法用于构造路径规划的高质量可行解，邻域搜索则用于在线改进解的质量，算法的构成如图4-2所示。在启发式算法设计阶段，本节提出了周期内的多源贪心动态导向集束搜索的贴装点分配算法，以及周期内的动态规划路径规划算法；在邻域搜索算法设计阶段，则进一步提出了基于自适应大邻域搜索的聚合路径重构算法。在周期内完成贴装点再分配后，动态规划用于规划拾贴周期内的最优路径。

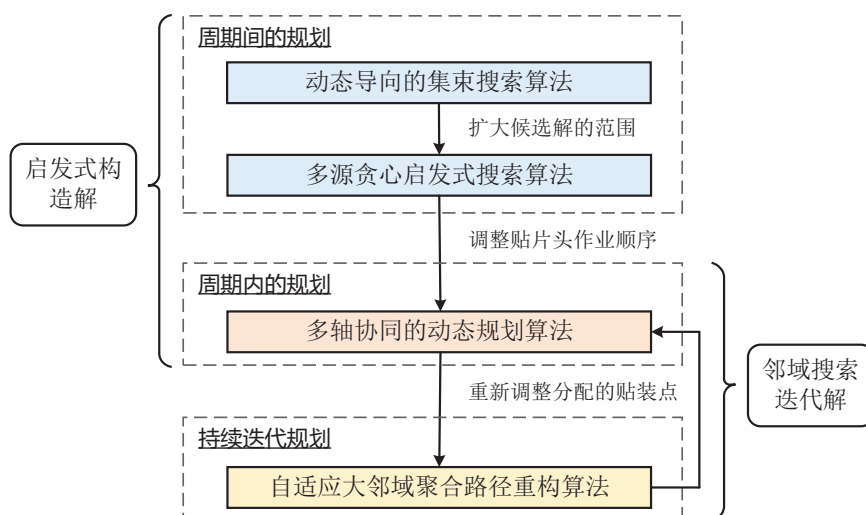


图 4-2 贴装过程的路径规划算法结构图

Fig. 4-2 Framework of path planning algorithm for placement process

4.3.2 基于动态规划的拾贴周期路径规划算法

拾贴周期内规划路径点数等于作业的贴片头数，对于数量有限的访问点，动态规划法可用于求解贴片头移动位置已知的路径规划最优解。记贴片头 m 的拾贴周期 k 贴装过程的起点位置坐标为 (X_Θ, Y_Θ) ，即元件拾取的终止位置，其满足：

$$X_\Theta = X^{F1} + F_{km}^{FW} \cdot \frac{\rho}{\tau}, \quad Y_\Theta = Y^{F1}, \quad \mathcal{H} = \{h' \mid \mathcal{P}_{kh'} > 0, h' \in H\} \quad (4-1)$$

式中， \mathcal{H} 为当前拾贴周期工作的贴片头索引集。

如前所述，多轴独立工作共同决定了贴装过程的移动用时，为计算悬臂在连续贴装点作业的用时，记当前访问贴装点 p 的坐标为 (X_p, Y_p) ，旋转角度为 R_p ，贴片头为 h ；下一访问贴装点 \hat{p} 的坐标为 $(X_{\hat{p}}, Y_{\hat{p}})$ ，旋转角度为 $R_{\hat{p}}$ ，贴片头为 \hat{h} 。记 $\delta(p, \hat{p}, h, \hat{h})$ 表示贴片头 h 完成元件 p 的贴装后，用贴片头 \hat{h} 贴装点 \hat{p} 的移动用时。如果 p 或 \hat{p} 为拾取点，即 $p = \Theta$ 或 $\hat{p} = \Theta$ ，则当前移动路径为拾取位置到贴装位置之间的往返路径之一，有

$$\delta(\Theta, p, \bullet, h) = \delta(p, \Theta, h, \bullet) = \max \left(\frac{|X_p - X_\Theta - (h-1) \cdot \rho|}{V^X}, \frac{|Y_p - Y_\Theta|}{V^Y} \right) \quad (4-2)$$

否则，对于非同轴运动贴装过程的移动路径，即 $[h/2] \neq [\hat{h}/2]$ ，有

$$\delta(p, \hat{p}, h, \hat{h}) = \max \left(\frac{|X_p - X_{\hat{p}} - (h - \hat{h}) \cdot \rho|}{V^X}, \frac{|Y_p - Y_{\hat{p}}|}{V^Y} \right) \quad (4-3)$$

否则，对于同轴运动贴装过程的移动路径，将进一步考虑 R 轴转动的用时，有

$$\delta(p, \hat{p}, h, \hat{h}) = \max \left(\frac{|X_p - X_{\hat{p}} - (h - \hat{h}) \cdot \rho|}{V^X}, \frac{|Y_p - Y_{\hat{p}}|}{V^Y}, \frac{|R_p - R_{\hat{p}}|}{V^R} \right) \quad (4-4)$$

其中 V^X , V^Y 和 V^R 分别对应 X 、 Y 、 R 三轴运动的平均速度。记 \mathcal{P}_{kh} 为拾贴周期 k 的贴片头 h 拾贴元件的贴装点索引，令贴片头 0 对应拾取位置，即 $\mathcal{P}_{k0} = \Theta$ 。拾贴周期动态规划的访问点集合由贴片头及拾取起始位置构成，贴片头与贴装点存在对应关系，由此构成了作业贴片头集合 $\mathcal{H} \cup \{0\}$ 。

基于给定的访问点集合和移动过程用时函数，可构造动态规划的状态转移方程，如式 (4-5)–(4-6) 所示。 $\mathcal{F}(p, \mathcal{H})$ 表示从点 p 出发，访问集合 \mathcal{H} 中贴片头对应的贴装点有且仅有一次，最后返回点 p 的最短路径长度。

$$\mathcal{F}(\Theta, \{0\}) = 0 \quad (4-5)$$

$$\mathcal{F}(\mathcal{P}_{kh}, \mathcal{H}^{\text{SUB}} + \{h\}) = \min_{\hat{h} \in \mathcal{H}^{\text{SUB}}} \left\{ \mathcal{F}(\mathcal{P}_{k\hat{h}}, \mathcal{H}^{\text{SUB}}) + \delta(\mathcal{P}_{kh}, \mathcal{P}_{k\hat{h}}, h, \hat{h}) \right\} \quad (4-6)$$

$$\mathcal{H}^{\text{SUB}} \subseteq \mathcal{H} \cup \{0\}, h \in \mathcal{H} \cup \{0\}, h \notin \mathcal{H}^{\text{SUB}}$$

式中, \mathcal{H}^{SUB} 为 \mathcal{H} 的子集。对于小规模类旅行商问题, 状态压缩法常用于基于动态规划的路径规划问题求解中, 通过求解状态转移方程可得, 拾贴周期内的最小移动用时为

$$\min_{\mathcal{P}_{kh} \in \mathcal{H}} \{ \mathcal{F}(h, \mathcal{H} \setminus \{h\}) + \delta(\Theta, \mathcal{P}_{kh}, h, \bullet) \} \quad (4-7)$$

在求解过程中记录不同状态对应的访问贴装点的先后顺序, 即可确定拾贴周期路径规划的最优访问顺序。

4.3.3 基于多源贪心的动态导向集束搜索路径规划算法

4.3.3.1 多源贪心启发式

动态规划用于解决拾贴周期内的路径规划, 本节将介绍拾贴周期分配贴装点的相关算法。在为贴片头分配贴装点时, 朴素的贪心算法是选取一个贴片头分配任一贴装点, 按照最近邻的分配原则对其它贴片头进行贴装点分配。多源贪心搜索通过选择不同的搜索方向调整起始搜索点和贴片头的分配顺序, 以满足电路板不同贴装点分布特性的实际要求。根据待贴装电路板确定有效贴装区域, 可确定其左边界和右边界的 X 坐标为 X^{LB} 和 X^{RB} , 搜索步长为 $(X^{\text{RB}} - X^{\text{LB}}) / (2 \cdot |H|)$ 。算法选取了 3 个不同的搜索方向完成优化, 分别是自左向右搜索, 自右向左搜索和自中心向两侧搜索。根据不同的搜索方向, 可生成起始搜索点顺序列表 $\hat{\mathcal{S}}$ 和贴片头访问顺序列表 $\hat{\mathcal{H}}$, 其中 $\hat{\mathcal{H}}$ 表示贴片头分配贴装点的先后顺序。搜索方向和起始点、贴片头的分配先后顺序的对应关系如表 4-1 所示。

表 4-1 不同搜索方向的起始搜索点和贴片头分配顺序

Table 4-1 Starting point and head assignment sequence of different search directions

方向	起始点和贴片头分配顺序
自左向右	$\hat{\mathcal{S}} = \{X^{\text{LB}} + (h-1) \cdot \rho \mid h \in H\}$ $\hat{\mathcal{H}} = H$
自右向左	$\hat{\mathcal{S}} = \{X^{\text{RB}} - (h-1) \cdot \rho \mid h \in H\}$ $\hat{\mathcal{H}} = \{ H + 1 - h \mid h \in H\}$
中心向两侧	$\hat{\mathcal{S}} = \{(3 \cdot X^{\text{LB}} + X^{\text{RB}}) / 4 + (h-1) \cdot \rho / 2 \mid h \in H\}$ $\hat{\mathcal{H}} = \left\{ \left\lfloor \frac{ H }{2} \right\rfloor + (-1)^{h+1} \cdot \left(\left\lceil \frac{h}{2} \right\rceil - \frac{1}{2} \right) + \frac{[(H +1) \bmod 2]}{2} \mid h \in H \right\}$

在进行贴装点分配时, 首先置空贴片头上的贴装点, 即令 $\mathcal{P}_{kh} = 0, \forall k \in K$,

$h \in H$, 表示各拾贴周期所有贴片头均未分配贴装点; 然后遍历不同的搜索方向, 记对应搜索起点为 $\Psi \in \hat{\delta}$, 贴片头分配顺序列表为 $\hat{\mathcal{H}}$ 。对于拾贴周期 k , 当遍历到 $\hat{\mathcal{H}}$ 中的贴片头 h 时, 其可访问的贴装点集合记为 P' 。若 h 为列表 $\hat{\mathcal{H}}$ 首个贴片头, 则找出水平方向上距离搜索起始点最近的点,

$$P' \leftarrow \{p \mid \eta_{ip} = 1, i = C_{kh}, p \in P\} \quad (4-8)$$

$$p \leftarrow \operatorname{argmin}_{p' \in P'} |X_{p'} - (h-1) \cdot \rho - \Psi| \quad (4-9)$$

否则, 按移动过程单向访问的顺序, 计算分配贴装点 p 后拾贴周期的移动路径长度, 记集合 \mathcal{X}_p 和 \mathcal{Y}_p 为当前拾贴周期已分配的贴装点和在进行点 p 贴装作业时, 悬臂所在位置的 X 坐标和 Y 坐标构成的有序集合, 具体为

$$\mathcal{X}_p \leftarrow \{X_{p'} - (h' - 1) \cdot \rho \mid p' = \mathcal{P}_{kh'}, p' \neq 0, h' \in H\} \cup \{X_p - (h - 1)\} \quad (4-10)$$

$$\mathcal{Y}_p = \{Y_{p'} \mid p' = \mathcal{P}_{kh'}, p' \neq 0, h' \in H\} \cup \{Y_p\} \quad (4-11)$$

记 t 为 \mathcal{X} 中 X 坐标第 t 小的索引, 有

$$p \leftarrow \operatorname{argmin}_{p' \in P'} \sum_{t=1}^{t < |\mathcal{X}_{p'}| - 1} \max(|\mathcal{X}_{p't} - \mathcal{X}_{p'(t+1)}|, |\mathcal{Y}_{p't} - \mathcal{Y}_{p'(t+1)}|) \quad (4-12)$$

根据最近邻或快排规划确定的贴装点 p 为贴片头分配吸嘴, 即令 $\mathcal{P}_{kh} = p$ 。

4.3.3.2 动态导向的集束搜索

集束搜索是一类启发式的图搜索算法, 在邻域中具有较强的搜索能力。集束搜索通过保留预定数量的最佳局部解最为候选, 在每一步深度扩展时进行进一步的更新和筛选, 通常能找到问题得近似最优解。搜索过程中保留的结点数被称为集束宽度, 影响着搜索的效率和最终解的质量。当集束宽度为无穷大时, 集束搜索等同于广度优先搜索; 而当集束搜索宽度为 1 时, 则其退化为贪心算法。贴装过程作为贴片头任务分配约束下的路径规划问题, 其对贴片头的贴装点分配可视为搜索树的结点。本节将集束搜索运用于此过程中, 进一步完善多源贪心的路径规划算法, 通过引入动态导向的搜索策略, 在扩展集束搜索宽度的同时, 以最佳解引导集束搜索的方向, 提升解的质量。

算法4-1将多源贪心启发式与动态导向集束搜索算法相结合, 用于路径规划可行解的生成。其中多源贪心算法构成了算法的基本框架, 集束搜索则体现在单个贴片头同时分配多个较优的贴装点, 扩大搜索范围。贴片头数量的有限性为集束搜索创造了条件, 一方面, 集束搜索的剪枝过程可在拾贴周期元件分配

完成后进行，避免了单一贴片头分配完成后过早剪枝造成的搜索的短视性，较低的复杂度可以进一步扩充集束宽度；另一方面，采用动态规划确定各周期的拾贴顺序，在贴装点数量有限的情况下效率很高，拾贴周期分配结束后，动态规划法会比快排规划策略更为准确地计算贴装过程时间，有利于保留高质量的解。去重过程则用于分配中消除集束搜索到的相同解，提升搜索效率。

算法 4-1 多源贪心动态导向集束启发式路径规划算法

Algo.4-1 Multi-point greedy beam search algorithm with dynamic orientation

输入：贴片头元件分配解 C ，周期组索引集 \mathcal{K} ，贴装点坐标 (X_p, Y_p) ， $p \in P$
 输出：贴片头分配贴装点 \mathcal{D} 和贴装先后顺序 \mathcal{Q}

```

1  令集束索引集  $B \leftarrow \{1, 2, \dots, \beta\}$ ， $|B|$  为集束宽度，令导向贴片头分配点  $\mathcal{D}^{\text{REF}}$  为空；
2  for 遍历不同的集束搜索宽度  $\beta \in B$  do
3      令  $\mathcal{D}(b)$  为大小  $|\mathcal{K}| \times |H|$  值均为 0 的矩阵，表示贴片头分配的贴装点，对应
        地， $\mathcal{Q}(b)$  为贴片头贴装顺序， $\mathcal{T}(b)$  为贴装过程用时， $b \in \{1, 2, \dots, \beta + 1\}$ ；
4      令  $P(b) \leftarrow P$  表示不同集束未分配的点集， $b \in \{1, 2, \dots, \beta + 1\}$ ；
5      for 遍历各拾贴周期  $k \in \mathcal{K}$  do
6          for 遍历不同的搜索方向 do
7              /* 根据搜索方向，遍历不同的搜索起始点和贴片头分配顺序 */
8              for  $\Psi \in \hat{\mathcal{S}}$ ， $h \in \hat{\mathcal{H}}$  do
9                  if  $\sum_{h' \in H} \mathcal{D}(1)_{kh'} = 0$  then
10                     按最近邻规则，以式 (4-8)–(4-9) 确定贴片头  $h$  分配的  $p$ ，令
11                      $\mathcal{D}(b)_{kh} \leftarrow p$ ， $b \in \{1, 2, \dots, \beta\}$ ；
12                 else
13                     按快排规划算法，以式 (4-10)–(4-12) 分配移动路径最小的前  $\beta$ 
14                     个贴装点至贴片头  $h$ ，令  $\mathcal{D}(b)_{kh} \leftarrow p_b$ ， $b \in \{1, 2, \dots, \beta\}$ ；
15                 end
16                 从  $P(b)$  中移除已分配的贴装点， $b \in \{1, 2, \dots, \beta\}$ ；
17                 if  $\mathcal{D}^{\text{REF}}$  非空 then
18                     根据参考点分配贴片头贴装点，令  $\mathcal{D}(\beta + 1)_{kh} \leftarrow \mathcal{D}_{kh}^{\text{REF}}$  作为候
19                     选解；
20                 end
21             end
22         end
23     end
24     运用动态规划从候选解中筛选出前  $\beta$  个总移动用时最小的贴装点分配解
25      $\mathcal{D}(b)$ ，对应的贴片头贴装顺序为  $\mathcal{Q}(b)$ ，对应的贴装过程用时为  $\mathcal{T}(b)$ ；
26     更新集束未分配点集  $P(b)$ ， $b \in \{1, 2, \dots, \beta\}$ ；
27 end
28 以当前最优解作为新的参考点  $\mathcal{D}^{\text{REF}} \leftarrow \mathcal{D}(\arg\min_{b \in \{1, \dots, \beta + 1\}} \mathcal{T}(b))$ ；
29 end
    
```

集束搜索通过扩充集束宽度增加了搜索解的多样性，动态导向则以集束宽度增加前的已知最优解作为参考，在此基础上进行扩展搜索。图4-3演示了动态导向的集束搜索过程。在为贴片头分配贴装点时，搜索树在迭代过程中对已知最优解进行了扩展，同时在剪枝过程中保留了当前最佳节点。集束搜索宽度的增加会对搜索树的搜索方向产生影响，贪心式地截断搜索树的结点可能会降低

解的质量, 本节在集束搜索中引入了最佳解作为导向点, 引导搜索过程扩展结点, 同时不会改变宽度增加可能引入的效率降低。可以说, 动态导向的搜索过程避免了集束搜索的盲目性, 有助于保留高质量的拾贴周期规划解, 进而提升整体解的质量。

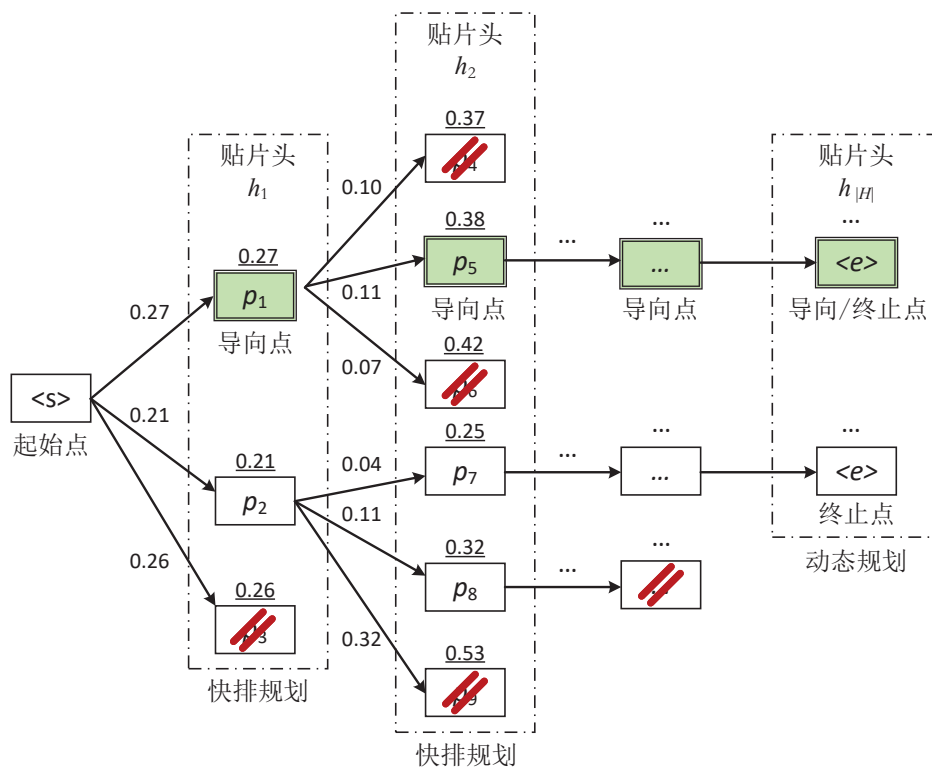


图 4-3 具有导向点的贴装过程集束搜索过程示意图

Fig. 4-3 Schematic diagram of beam search process with guide points of placement process

4.3.4 基于自适应大邻域搜索的聚合路径重构算法

以贪心为基础的动态导向集束搜索确定了高质量的路径规划可行解, 基于此, 本节将进一步讨论路径的改进方案。为克服贪心式搜索的局限性, 路径重构技术用于破坏和修复不同拾贴周期的已分配贴装点。重构过程是基于固定的供料器和 ANC 配置的优化, 其同样受到贴片头任务分配结果的约束。本节提出的自适应大邻域聚合路径重构算法可作为贴装过程的在线优化方案, 用于持续提升组装效率。

在邻域搜索过程中, 路径重构本质上也是不断破坏和修复周期内拾贴路径的过程, 与之对应的破坏和修复算子通常成对出现。传统的邻域搜索算法中, 单一的邻域搜索策略仅能满足小部分解空间搜索的要求, 容易陷入局部最优解。即使引入局部跳出策略, 由于搜索空间的约束性和广泛性, 算法找到全局最优解的概率仍然较小。变邻域搜索法通过引入多种邻域算子, 大幅提升了搜索范围,

自适应大邻域搜索则在此基础上,根据邻域解的质量调整不同邻域搜索算子的执行概率,运用启发式的信息指导搜索过程,算子间的相互竞争自适应生成当前解的邻域结构。

算法4-2给出了自适应大邻域聚合路径重构算法的整体框架。算法以大邻域搜索为基础,通过使用试探法交替地破坏和修复解来探索复杂邻域,改进初始解的质量。聚合过程以拾贴周期的移动路径和贴片头中心偏移量为基础,破坏算法从拾贴周期中移除已分配的贴装点,修复算法则在满足贴片头任务分配约束下,对移除的贴装点执行再分配的操作。自适应算法在搜索过程中,融合了Metropolis抽样稳定接受准则,通过调整破坏和修复算子的权重,实现搜索初期的大范围搜索,并最终迭代收敛至稳定解。

算法 4-2 自适应大邻域搜索的聚合路径重构算法

Algo.4-2 Adaptive large neighborhood search and aggregated route relink algorithm

输入: 贴片头元件分配解 \mathcal{C} , 贴片头贴装点分配解 \mathcal{P}
输出: 贴片头贴装点分配解 \mathcal{P}^* 和贴装顺序解 \mathcal{Q}^*

- 1 以式 (4-13)–(4-15) 和 \mathcal{P} 计算拾贴周期中心 (\bar{X}, \bar{Y}) , 移动距离 \mathcal{D} 和中心偏移 ϱ ;
- 2 令 $\mathcal{P}^* \leftarrow \mathcal{P}$, $\mathcal{D}^* \leftarrow \mathcal{D}$, $\bar{X}^* \leftarrow \bar{X}$, $\bar{Y}^* \leftarrow \bar{Y}$, $\mathcal{P}' \leftarrow \mathcal{P}$, $\tilde{\mathcal{D}}' \leftarrow \mathcal{D}$, $\bar{X}' \leftarrow \bar{X}$, $\bar{Y}' \leftarrow \bar{Y}$;
- 3 **for** 遍历最多迭代次数 **do**
- 4 **while** 未达到迭代终止条件 **do**
- 5 令被移除贴装点集 $P^{RM} \leftarrow \emptyset$, 令 $\mathcal{P} \leftarrow \mathcal{P}'$, $\mathcal{D} \leftarrow \mathcal{D}$, $\bar{X} \leftarrow \bar{X}'$, $\bar{Y} \leftarrow \bar{Y}'$;
- 6 依概率 κ^- 从集合 Ω^- 选择破坏算子 l , 并根据 \mathcal{D} 和 ϱ , 选择若干拾贴周期-贴片头对 (k, h) , 构成集合 Ξ ;
- 7 令 $P^{RM} \leftarrow P^{RM} \cup \{\mathcal{P}_{kh}\}$, $\mathcal{P}_{kh} \leftarrow 0$, $\forall (k, h) \in \Xi$, 以式 (4-16)–(4-17) 更新 (\bar{X}, \bar{Y}) ;
- 8 依概率 κ^+ 从 Ω^+ 选择修复算子 l ;
- 9 **for** 遍历被移除贴装点 $p \in P^{RM}$ **do**
- 10 以式 (4-10)–(4-11) 确定 $(k, h) \in \Xi$ 时 p 对应的 \mathcal{X}_p 和 \mathcal{Y}_p , 记 t 为 \mathcal{X}_p 中值第 t 大的索引;
- 11 根据算子 l 和 $\sum_{t=1}^{|\mathcal{X}_p|-1} \max(|\mathcal{X}_{pt} - \mathcal{X}_{p(t+1)}|, |\mathcal{Y}_{pt} - \mathcal{Y}_{p(t+1)}|)$ 的值, 选择 k 和 h , 满足 $\eta_{ip} = 1$, 其中 $i = C_{kh}$, 令 $\mathcal{P}_{kh} \leftarrow p$, $\Xi \leftarrow \Xi \setminus \{(k, h)\}$;
- 12 **end**
- 13 以式 (4-15) 更新 ϱ , 以式 (4-18)–(4-19) 更新 (\bar{X}, \bar{Y}) ;
- 14 **if** 满足式 (4-20) 的 Metropolis 接受准则 **then**
- 15 令 $\mathcal{P}' \leftarrow \mathcal{P}$, $\mathcal{D}' \leftarrow \mathcal{D}$, $\bar{X}' \leftarrow \bar{X}$, $\bar{Y}' \leftarrow \bar{Y}$;
- 16 **end**
- 17 **if** $\sum_{k \in K} \mathcal{D}_k < \sum_{k \in K} \mathcal{D}_k^*$ **then**
- 18 令 $\mathcal{P}^* \leftarrow \mathcal{P}$, $\mathcal{D}^* \leftarrow \mathcal{D}$, $\bar{X}^* \leftarrow \bar{X}$, $\bar{Y}^* \leftarrow \bar{Y}$, 以动态规划确定 \mathcal{Q}^* ;
- 19 **end**
- 20 根据 \mathcal{D} 以及 \mathcal{P} 是否被接受, 以式 (4-21)–(4-22) 更新 κ^- 和 κ^+ , 以式 (4-23) 更新 ϑ ;
- 21 **end**
- 22 **end**

路径重构过程以拾贴周期为基本单位，路径的破坏以拾贴周期 k 的移动路径 \mathcal{D}_k 为依据，修复则以贴装点到拾贴周期 k 的中心位置 (\bar{X}_k, \bar{Y}_k) 为准，具体为

$$\bar{X}_k \leftarrow \sum_{h \in H} \frac{X_{\mathcal{P}_{kh}} - (h-1) \cdot \rho}{|H|}, \quad \bar{Y}_k \leftarrow \sum_{h \in H} \frac{Y_{\mathcal{P}_{kh}}}{|H|} \quad \forall k \in K \quad (4-13)$$

$$\mathcal{D}_k \leftarrow \sum_{q=(h_1, h_2) \in \mathcal{Q}_k} \max \left(\left| X_{\mathcal{P}_{kh_1}} - X_{\mathcal{P}_{kh_2}} - (h_1 - h_2) \cdot \rho \right|, \left| Y_{\mathcal{P}_{kh_1}} - Y_{\mathcal{P}_{kh_2}} \right| \right) \quad \forall k \in K \quad (4-14)$$

选择破坏的拾贴周期时，共有三种不同的方案：随机选取，加权选取和贪心选取。加权和贪心的选取方式根据拾贴周期移动路径 \mathcal{D} 的长度进行选择破坏。贴装点的选取与拾贴的贴片头及其位置相关，根据中心位置偏移量 ϱ 确定，即

$$\varrho_{kh} \leftarrow \begin{cases} \infty & \mathcal{P}_{kh} = -1 \\ \max \left\{ \bar{X}_k - X_p + (h-1) \cdot \rho, \bar{Y}_k - Y_p \right\} & p = \mathcal{P}_{kh} \neq -1 \end{cases} \quad \forall k \in K, h \in H \quad (4-15)$$

被选取的贴装点同对应的贴片头 h 相关，即 $p = \mathcal{P}_{kh}$ ，未分配元件的贴片头不在选取之列。同理，选择被破坏路径的拾贴周期和对应的贴装点也有三种不同的方案——依据 ϱ 随机选取、加权选取和最远选取，不同的拾贴周期和贴片头的选取方案构成了9种不同的破坏-修复算子对。破坏过程选取一定比例的贴装点，增加邻域搜索的多样性。拾贴周期移除贴装点 p 后，其对应的拾贴周期 k 平均坐标点也随之调整，即

$$\bar{X}_k \leftarrow \left(1 + \frac{1}{\sum_{h' \in H} (\mathcal{P}_{kh'} \neq 0) - 1} \right) \cdot \bar{X}_k - \frac{X_p - (h-1) \cdot \rho}{\sum_{h' \in H} (\mathcal{P}_{kh'} \neq 0) - 1} \quad (4-16)$$

$$\bar{Y}_k \leftarrow \left(1 + \frac{1}{\sum_{h' \in H} (\mathcal{P}_{kh'} \neq 0) - 1} \right) \cdot \bar{Y}_k - \frac{Y_p}{\sum_{h' \in H} (\mathcal{P}_{kh'} \neq 0) - 1} \quad (4-17)$$

对应地，移除拾贴周期 k 的贴片头 h 上的贴装点 p ，令 $\mathcal{P}_{kh} \leftarrow 0$ ，将贴装点加入未分配点集。修复算子与破坏算子执行相反的操作，修复算子将未分配集中的贴装点分配至空的贴片头上。修复算子本质上是发生在同类型元件贴装点之间的交换操作，其中交换的范围根据随机移除元件类型和贴装点数确定。贴片头分配新的贴装点 p 后，更新对应的拾贴周期 k 平均坐标。

$$\bar{X}_k \leftarrow \left(1 - \frac{1}{\sum_{h' \in H} (\mathcal{P}_{kh'} \neq 0) + 1} \right) \cdot \bar{X}_k + \frac{X_p - (h-1) \cdot \rho}{\sum_{h' \in H} (\mathcal{P}_{kh'} \neq 0) + 1} \quad (4-18)$$

$$\bar{Y}_k \leftarrow \left(1 - \frac{1}{\sum_{h' \in H} (\mathcal{P}_{kh'} \neq 0) + 1} \right) \cdot \bar{Y}_k + \frac{Y_p}{\sum_{h' \in H} (\mathcal{P}_{kh'} \neq 0) + 1} \quad (4-19)$$

修复算子操作完成后，拾贴周期 k 对应的移动路径长度 \mathcal{D}_k 和贴片头拾贴先后顺序 \mathcal{Q}_k 由4.3.2节提出的动态规划法重新计算获得。

Metropolis 抽样稳定准则在用于决定系统是否接受新状态，从而影响随机搜索的方向和收敛速度，其以一定概率 χ 接受使整体优化目标变差的解，随着搜索过程的深入，接受差解的概率逐步降低，重复搜索过程，直到满足终止准则，得到改进的路径规划问题的较优解。记当前迭代解的拾贴周期 k 的移动路径为 \mathcal{D}_k ，完成修复-破坏后新解的拾贴周期 k 的移动路径长度为 $\tilde{\mathcal{D}}_k$ ，有

$$\chi = \begin{cases} 1 & \sum_{k \in K} \tilde{\mathcal{D}}_k < \sum_{k \in K} \mathcal{D}_k \\ e^{\left(-\sum_{k \in K} \frac{\tilde{\mathcal{D}}_k - \mathcal{D}_k}{T}\right)} & \sum_{k \in K} \tilde{\mathcal{D}}_k \geq \sum_{k \in K} \mathcal{D}_k \end{cases} \quad (4-20)$$

记破坏算子和修复算子集合分别为 Ω^- 和 Ω^+ ，搜索过程中，破坏算子和修复算子根据其对解产生的实际效果自适应地调整其动态权重 κ^- 和 κ^+ ，破坏算子和修复算子的更新过程如式 (4-21) 和式 (4-22) 所示。

$$\kappa_l^- \leftarrow \alpha \kappa_l^- + (1 - \alpha) \cdot \Gamma \quad \forall l \in \Omega^- \quad (4-21)$$

$$\kappa_l^+ \leftarrow \alpha \kappa_l^+ + (1 - \alpha) \cdot \Gamma \quad \forall l \in \Omega^+ \quad (4-22)$$

其中 α 表示算子权重的更新系数， Γ 的取值与算子的执行效果相关， $\Gamma = \Gamma_1$ 时表示改善最优解， $\Gamma = \Gamma_2$ 时表示改善当前解， $\Gamma = \Gamma_3$ 时表示接受当前解， $\Gamma = \Gamma_4$ 时表示拒绝当前解，不同情况的取值互斥且满足 $\Gamma_1 > \Gamma_2 > \Gamma_3 > \Gamma_4$ 。第 l 个破坏算子被调用的概率 ϑ_l 根据权重决定，即

$$\vartheta_l = \begin{cases} \kappa_l^- / \sum_{l'=1}^{|\Omega^-|} \kappa_{l'}^- & l \in \Omega^- \\ \kappa_l^+ / \sum_{l'=1}^{|\Omega^+|} \kappa_{l'}^+ & l \in \Omega^+ \end{cases} \quad (4-23)$$

自适应聚合路径重构的过程作为一种启发式搜索算法，融合了贴装点布局的结构特征，动态调整路径破坏和重连的算子权重，根据算子的历史表现和搜索次数选择下次迭代的算子，通过算子间的相互竞争生成复杂的邻域结构，从而加速了更优解决方案的搜索过程。

4.4 实验设计

本章提出的多源贪心的动态导向集束搜索 (Multi-Point Greedy Beam Search with Dynamic Orientation, MPGBS-DO) 算法和自适应大邻域聚合路径重构 (Adaptive Large Neighborhood Aggregated Route Relink, ALNARR) 算法分别用于生成和

改进路径规划的解。为说明提出算法的实际效果，本章选取了四组不同的贴装过程路径优化算法，分别为同类型贴片机生产制造商发布的工业软件内置的元件贴装顺序优化器（Component Place Sequence Optimizer, CPSO），Ashayeri 等^[17]提出的贪心分级贴装启发式（Greedy Level Placing Heuristic, GLPH）算法，Guo 等^[18]提出的贴装聚类排序贪心启发式（Placement Cluster and Sequence Greedy Heuristic, PCSGH）算法和 Gao 等^[47]提出的贴装分配排序启发式（Place Allocation and Sequence Heuristic, PASH）算法。

实验程序运行和参数配置同第2章相同。本章所研究的以及用于对比的算法均是在贴片头任务分配约束下进行的路径规划算法，如无特殊说明，实验所用的贴片头任务分配结果均选用第3章提出的算法获得。对比实验所用的10组PCB数据组5参数如表4-2所示，各类型元件的可用供料器数均为1。其中，第一组数据为IPC-9850，该数据是由表面贴装设备标准集成设备附属委员会提出的，用以描述组装机性能的标准测试数据。在实际验证中，各组电路板数据的组装用时取最近三次运行的稳定结果，以其平均值作为比较所用的指标。

表 4-2 PCB 数据的基本参数
Table 4-2 Basic parameters of PCB data

PCB	5-1	5-2	5-3	5-4	5-5	5-6	5-7	5-8	5-9	5-10
吸嘴类型数	1	1	1	2	3	3	3	3	4	3
元件类型数	1	7	18	4	5	6	16	20	10	24
贴装点数	400	564	176	352	384	336	114	150	196	236

4.4.1 基于启发式的贴装过程路径规划的对比实验

MPGBS-DO 算法是一类基于规则的启发式路径规划，用于比较的集束宽度设定为4。贴片头的任务分配结果作为贴装过程的约束条件之一，限制了贴装过程解优化目标的边界值，确定了贴片头的拾取过程的路径，而并未对其贴装路径及基座-电路板间的往返路径进行优化。本章提出的路径规划算法同时对贴装过程和往返过程进行了优化，表4-3比较了相同贴片头任务分配下不同启发式路径规划算法结果，CPSO 因未开源实现方式不在比较之列。实验结果表明，本章提出的 MPGBS-DO 路径规划的贴装用时 $\mathcal{T}^{\text{BASE}}$ 相较于 GLPH 和 PCSGH 的贴装过程用时 $\mathcal{T}^{\text{GLPH}}$ 和 $\mathcal{T}^{\text{PCSGH}}$ 分别缩短了 14.41% 和 14.97%，和同为考虑了贴片头线性排列的 PASH 的结果 $\mathcal{T}^{\text{PASH}}$ 在贴装过程用时上缩短了 2.77%。

表 4-3 相同贴片头任务下 MPGBS-DO 算法和其他算法的贴装过程用时比较
Table 4-3 Comparison of placement process time between MPGBS-DO algorithm and other algorithm in the same PAP head task

PCB	MPGBS-DO	GLPH		PCSGH		PASH	
	$\mathcal{T}^{\text{BASE}}$ (s)	$\mathcal{T}^{\text{GLPH}}$ (s)	\mathcal{Q} (%)	$\mathcal{T}^{\text{PCSGH}}$ (s)	\mathcal{Q} (%)	$\mathcal{T}^{\text{PASH}}$ (s)	\mathcal{Q} (%)
5-1	40.01	61.42	34.86	59.35	32.59	42.32	5.46
5-2	69.93	86.05	18.73	92.71	24.47	73.16	4.41
5-3	28.29	33.30	15.07	30.92	8.52	28.99	4.41
5-4	50.31	59.26	15.10	61.72	18.48	50.56	0.50
5-5	58.26	66.17	11.95	70.04	16.81	60.74	4.07
5-6	51.04	57.35	11.00	60.07	15.03	51.59	1.06
5-7	17.59	18.98	7.32	19.53	9.93	17.88	1.64
5-8	24.38	29.33	16.89	27.08	9.99	24.79	1.64
5-9	33.93	36.31	6.60	37.94	10.62	35.05	3.25
5-10	33.38	35.74	6.60	34.49	3.21	34.48	3.19
AVG			14.41		14.97		2.77

由于贴片头任务分配与贴装路径规划之间存在关联，本节将进一步讨论在不同贴片头任务分配约束条件下，路径规划算法的组装效率对比。此前已分析了影响组装效率的关键性能指标，验证了不同贴片头任务分配算法在子目标之间存在显著差异，故直接对比贴装过程中的移动路径用时缺乏公平性。在评估组装效率的主要指标体系中，拾贴周期数是评价基座-供料器往返移动路径及贴装路径之和对组装效率影响的指标之一。因此，在比较不同贴片头约束下的路径规划算法时，本节选取拾贴周期的平均移动用时作为评价标准，此参数同时也对应贴片头任务分配模型中拾贴周期数的权重系数，且值更为准确。

表4-4以拾贴周期平均贴装过程用时为指标比较了不同的路径规划方法，其中的差距值 \mathcal{Q} 是相较于 MPGBS-DO 得到的结果。由表可知，本章提出的 MPGBS-DO 得到的平均贴装过程用时优于工业软件内置的 CPO-CPSO 的结果 \mathcal{T}^{CC} ，与 CDGA-PASH 的结果 \mathcal{T}^{CP} 有相近的表现。在具有相近子目标的贴片头优化算法中，AMIP-GLPH 在拾贴周期和吸嘴更换上与 MPGBS-DO 对应的贴片头任务分配结果接近，然而，AMIP-GLPH 平均贴装过程移动用时 \mathcal{T}^{AG} 更长，且更多的拾取次数和贴装过程移动用时使 AMIP-GLPH 组装效率较低。HGA-PCSGH 得到的结果 \mathcal{T}^{HP} 则是在基于禁止吸嘴更换和构造同步拾取组的方案下进行的路径规划，较多的拾贴周期数导致其平均拾贴周期的移动用时指标值较优，而悬臂频繁的基座-供料器往返移动会降低整体组装效率。

表 4-4 不同贴片头任务下路径规划算法的拾贴周期平均贴装过程用时的比较
Table 4-4 Comparison of average placement time of PAP cycle of different path planning algorithms with different PAP head task

PCB	CPO-CPSO		AMIP-GLPH		HGA-PCSGH		CDGA-PASH	
	\mathcal{T}^{CC} (ms)	\mathcal{G} (%)	\mathcal{T}^{AG} (ms)	\mathcal{G} (%)	\mathcal{T}^{HP} (ms)	\mathcal{G} (%)	\mathcal{T}^{CP} (ms)	\mathcal{G} (%)
5-1	596.3	-0.14	916.3	34.83	424.1	-40.80	834.1	28.41
5-2	765.9	2.86	996.8	25.37	508.9	-46.20	783.4	5.03
5-3	916.7	-2.85	1230.7	23.39	920.2	-2.47	870.1	-8.36
5-4	788.9	-2.86	901.7	10.01	803.7	-0.97	721.3	-12.50
5-5	715.2	-13.14	915.5	11.61	975.9	17.08	869.5	6.94
5-6	827.7	3.64	968.4	17.64	915.2	12.85	838.3	4.86
5-7	893.5	17.99	836.2	12.37	809.1	9.43	770.8	4.93
5-8	905.6	15.88	1092.0	30.24	518.1	-47.03	820.9	7.20
5-9	902.5	16.49	811.5	7.13	814.1	7.43	519.2	-45.16
5-10	760.4	2.44	901.9	17.75	732.9	-1.21	757.6	2.07
AVG		4.03		19.03		-9.19		-0.66

表4-5进一步分别比较了集束宽度为 1、4、7 和 10 情形下的拾贴周期贴装过程平均用时 \mathcal{T} 以及不同搜索策略对整体组装效率的影响, 当集束搜索过程无动态导向时, 算法可视为多源贪心集束搜索 (Multi-Point Greedy Beam Search, MPGBS), 而当集束宽度为 1 时, 算法则进一步退化为多源贪心搜索 (Multi-Point Greedy Search, MPGS)。由表可知, 集束宽度的增加将缩短拾贴周期的平均贴装过程的移动用时, 无动态导向的集束搜索会盲目搜索低质量的解, 导致集束宽度增加对解的质量改进有限。动态导向的引入通过调整搜索方向进一步提升了解的质量。表4-5中 MPGS 和 MPGBS 对应的 \mathcal{G} 分别表示由贪心搜索调整为集束搜索和在集束搜索中引入动态导向的效率改进值。当集束宽度由 1 增加至 4, 电路板的贴装过程用时平均缩短 1.30%, 动态导向的引入则将贴装过程用时进一步缩短了 1.85%, 二者结合共同构成了有确定性解的多源贪心动态导向集束路径规划算法。

4.4.2 基于邻域搜索的贴装过程路径规划的对比实验

自适应大邻域搜索用于在线持续更新路径迭代过程的解, 表4-6列出了自适应大邻域搜索相关的参数, 其中各轮搜索的最大迭代次数为 2000。根据路径规划解的数值, Metropolis 稳定性准则的初始温度设定为 1, 迭代终止温度为 0.2, 降温系数为 0.8。在每轮搜索中, 30% 的拾贴周期中单个贴片头的路径被破坏和修复。在自适应搜索过程中, 改善已知最优解和当前最优解的奖励分数分别为 3 和 2, 采用概率接受的奖励分数为 0.8, 拒绝当前解奖励分数为 0.2, 修复和破

表 4-5 不同集束宽度和搜索策略对路径规划解的质量影响比较

Table 4-5 Comparison of effectiveness of different beam width and search strategies on the quality of path planning solutions

PCB	MPGBS				MPGS	MPGBS-DO
	$\mathcal{T} : \beta = 1$ (ms)	$\mathcal{T} : \beta = 4$ (ms)	$\mathcal{T} : \beta = 7$ (ms)	$\mathcal{T} : \beta = 10$ (ms)	\mathcal{G} (%)	\mathcal{G} (%)
5-1	625.3	620.7	617.3	617.3	0.74	3.96
5-2	796.3	787.7	787.7	787.7	1.09	5.88
5-3	1000.0	990.1	978.7	978.6	1.00	5.01
5-4	819.9	811.3	803.2	803.2	1.06	-0.02
5-5	821.0	804.3	797.9	797.9	2.07	-0.60
5-6	815.1	807.7	806.7	803.3	0.92	1.27
5-7	768.0	746.1	746.1	746.1	2.93	1.82
5-8	768.0	763.2	763.2	762.3	0.63	0.18
5-9	760.4	746.5	746.5	746.4	1.86	-0.94
5-10	762.3	756.6	754.6	750.7	0.75	1.99
AVG					1.30	1.85

坏算子的权重更新系数为 0.5。所有算子的调用概率均根据其权重系数确定。

表 4-6 自适应大邻域聚合路径重构搜索参数

Table 4-6 Parameters of adaptive large neighborhood aggregation route relink search

参数	设定值	参数	设定值
最大迭代次数	2000	改善最优解分数	3
迭代最终温度	1	改善当前解分数	2
迭代终止温度	0.2	接受当前解分数	0.8
降温系数	0.8	拒绝当前解分数	0.2
拾贴周期破坏比率	0.3	算子权重更新系数	0.5

为说明自适应大邻域搜索的实际效果，本节比较了随机生成（Random Generation, RG）与启发式规则构造（Heuristic Rule Construction, HRC）分别作为初始解在自适应搜索过程中的迭代效果，其中，后者采用本章提出的 MPGBS-DO 路径规划算法，对应的拾贴周期平均贴装过程用时分别被记为 \mathcal{T}^{RG} 和 \mathcal{T}^{HRC} 。迭代搜索过程共重复 5 次取平均值，结果如表 4-7 所示。由表可知，本章所提出的大邻域搜索算法相较于规则构造的初始解平均可缩短贴装过程时间 1.85%，即使是随机构造生成的初始解，邻域搜索算法仍可将其迭代收敛至与已知最优解平均误差 4.50%，最大误差 11.04% 以内。自适应大邻域搜索给构造启发式的解带来的提升相对较少，主要是由于其已产生质量较高的解，可供邻域搜索解提升范围有限。即使是较小的提升在较多拾贴周期中也有一定的累积。此外，邻域搜索虽然也能提升随机生成解的质量，但由于初始解的不确定性，导致算法在早期阶段无法有效地缩小搜索范围，从而影响收敛速度、进而降低解的质量，构造解则有效地规避了这一问题。

表 4-7 随机生成和启发式规则构造初始解在迭代过程中的优化结果
Table 4-7 Optimization result of initial solutions of random generation and heuristic rule construction in the iterative process

PCB	HRC + ALNARR		RG + ALNARR		MPGBS-DO
	\mathcal{T}^{HRC} (ms)	\mathcal{Q} (%)	\mathcal{T}^{RG} (ms)	\mathcal{Q} (%)	\mathcal{T} (ms)
5-1	590.8	1.06	656.1	11.04	597.1
5-2	742.8	0.16	801.7	7.93	744.0
5-3	916.3	2.90	941.6	2.76	942.8
5-4	803.3	1.01	847.2	5.46	811.5
5-5	785.8	2.98	821.3	4.51	809.2
5-6	783.3	1.82	808.4	3.20	797.6
5-7	721.1	1.62	758.4	5.18	732.8
5-8	748.2	1.82	763.6	2.06	761.8
5-9	724.8	3.98	735.4	1.46	753.6
5-10	733.5	1.13	743.7	1.39	741.9
AVG		1.85		4.50	

图4-4进一步比较了 10 组数据的 ALNARR 搜索过程的收敛曲线。贴片头的任务分配限制了初始解的生成，对于单类元件有较多贴装点的情形，其随机初始解效果远差于构造初始解，进而降低最终解的质量。本章提出的 ANLARR 具有快速收敛的能力，在有限次迭代内，随机初始解能快速接近构造初始解的值。构造启发式已经获得了质量较高的初始解，自适应搜索过程仍能在此基础上持续优化，获得更优解，且解的质量始终优于随机生成解的迭代结果。

基于表4-4，表4-8进一步比较了各类贴片头任务-贴装过程路径规划相结合优化得到的解在贴片机平台上的实际组装效率，其中组装时间被转换为了标准的单位芯片/每小时（Chip Per Hour，CPH）。本文提出的贴片头任务分配结果与 ALNARR 算法相结合，在标准速度测试数据 IPC-9850 上组装效率指标优于工业优化求解器的组装效率 ε^{CC} ，在整体组装效率上相较其平均提升 7.72%。随着问题规划的扩大，电路板贴装元件类型更加复杂，其在组装效率上提升的优势也更加显著。AMIP-GLPH 和 HGA-PCSGH 分别作为数学规划法和（元）启发式算法的代表，二者的组装效率分别记为 ε^{AG} 和 ε^{HP} 。然而，前者未考虑贴片头的同步拾取，后者则未平衡好贴片头的吸嘴更换与同步拾取，使得其整体组装效率不高，本文提出的方法相较其分别提升 39.25% 和 23.00%。CDGA-PASH 相对全面的考虑了组装过程，其组装效率记为 ε^{CP} ，该算法的输出结果存在不确定性，在处理复杂元件数据时的优化结果并不理想，本文提出的方法相较于其平均提升 23.16%。图4-5比较了不同优化方法的平均组装效率分布，由图可知，本文提出的算法和工业求解器的结果输出比较稳定，且平均组装效率更高。基于进化

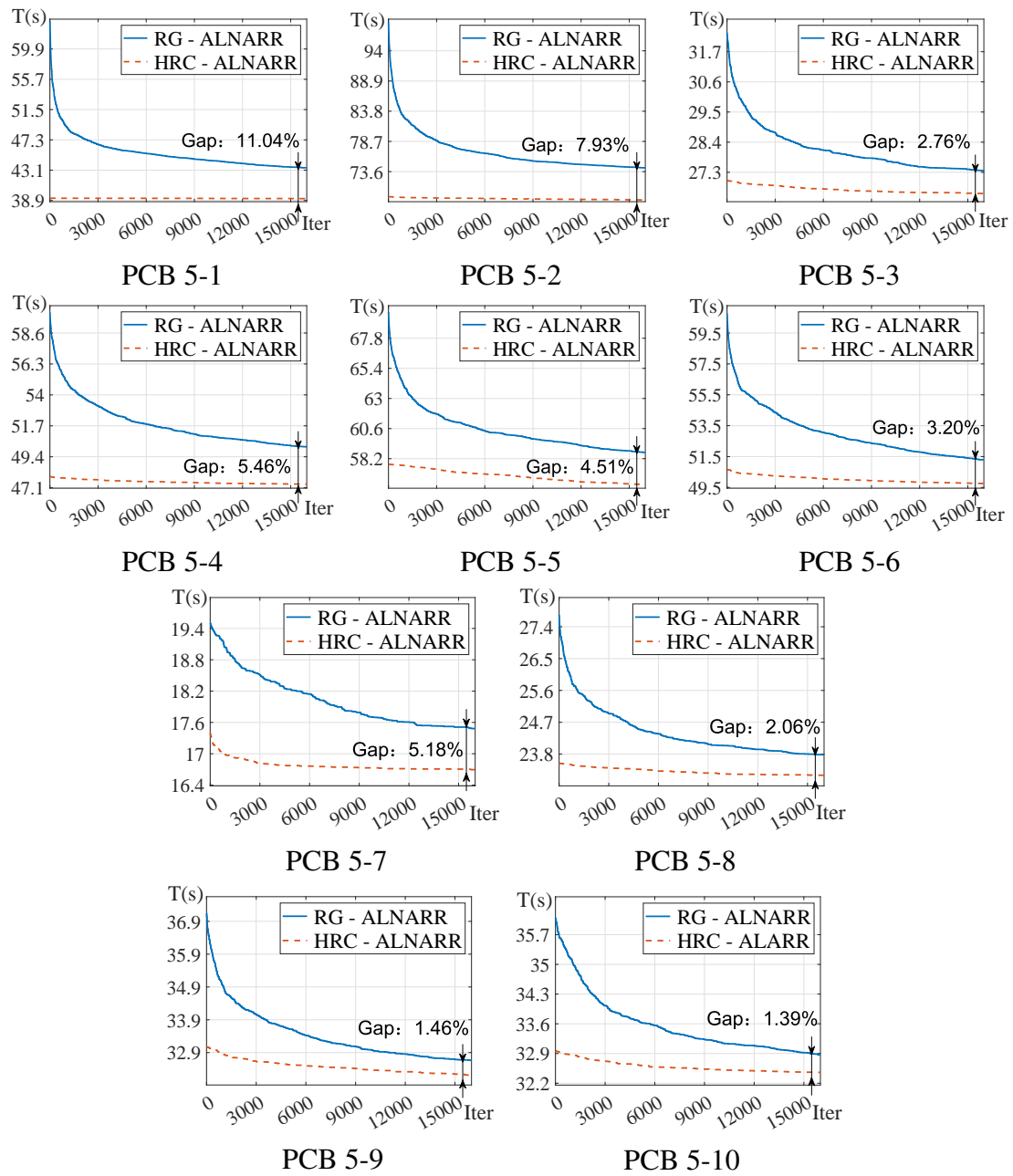


图 4-4 自适应大邻域聚合路径重构随机初始值和构造初始值收敛曲线图

Fig. 4-4 Convergence curves of adaptive large neighborhood aggregated route relink with random and constructive initial values

算法的优化方法结果输出相对不稳定，而基于数学规划的优化方法对其中影响组装效率的主要因素拾取数考虑不足，在所有的优化方法中组装效率最低。从整体上来说，贴片头任务分配优化的目标在组装效率中仍为主要因素，贴装过程路径在其中的影响占比相对较小，本文提出的方法整体效率上优于工业软件和主流研究的优化算法。

表 4-8 不同贴片头元件分配和贴装过程路径规划算法组合的实际组装效率比较
Table 4-8 Comparison of the actual assembly efficiency of different combinations of head task assignment and placement path planning algorithms

PCB	ALNARR	CPO-CPSO		AMIP-GLPH		HGA-PCSGH		CDGA-PASH	
	$\mathcal{E}^{\text{ALNARR}}$ (CPH)	\mathcal{E}^{CC} (CPH)	$\mathcal{Q}(\%)$	\mathcal{E}^{AG} (CPH)	$\mathcal{Q}(\%)$	\mathcal{E}^{HP} (CPH)	$\mathcal{Q}(\%)$	\mathcal{E}^{CP} (CPH)	$\mathcal{Q}(\%)$
5-1	12627	12604	0.19	10681	18.22	7005	80.26	11733	7.62
5-2	11306	11255	0.45	6991	61.72	7035	60.71	10673	5.93
5-3	16023	16003	0.12	11460	39.81	14958	7.12	12462	28.57
5-4	14527	14648	-0.82	12527	15.97	13582	6.96	13115	10.77
5-5	15023	11716	28.23	11742	27.94	14054	6.90	12917	16.30
5-6	15631	15799	-1.06	9741	60.46	15512	0.77	15034	3.97
5-7	15263	13022	17.21	9932	53.67	12346	23.62	11372	34.21
5-8	13839	11627	19.02	8843	56.50	10457	32.34	7556	83.15
5-9	13811	13059	5.76	11613	18.22	13379	3.23	12764	8.20
5-10	13065	12087	8.09	—	—	12087	8.09	9830	32.90
AVG			7.72		39.25		23.00		23.16

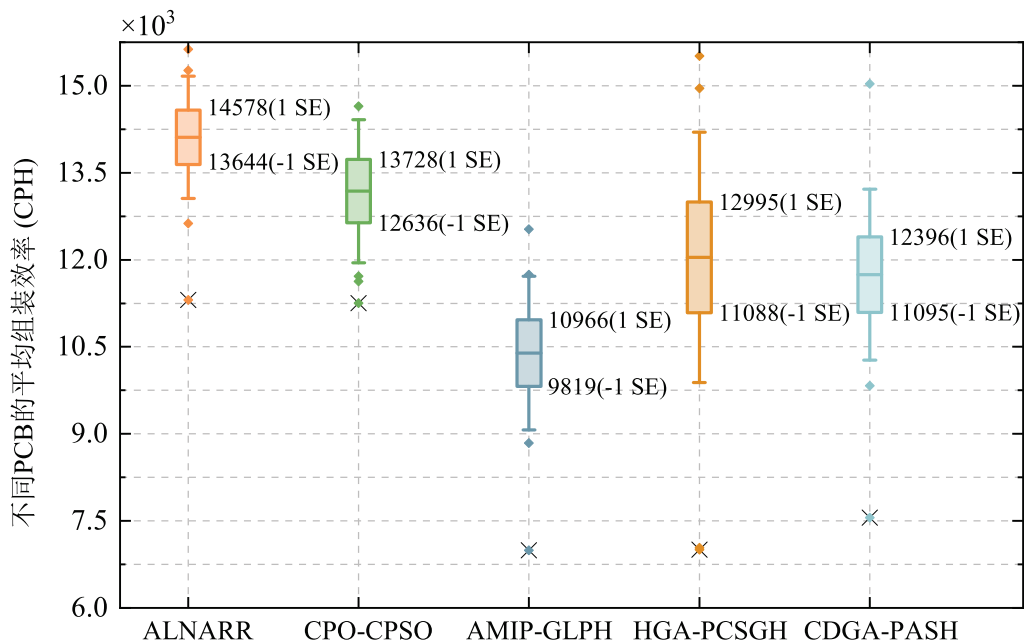


图 4-5 不同优化方法的平均表面组装效率分布比较

Fig. 4-5 Comparison of average surface assembly efficiency distribution of different optimization methods

4.4.3 运行效率分析

最后, 本节比较了基于启发式规则和基于邻域搜索的贴装过程路径规划搜索用时, 如表4-9所示。相较于已有的路径规划方法, 本章提出的 MPGBS-DO 搜索范围更广、用时更长且获得的解的质量更高, 其与 ALNARR 算法的运算时间

仍在可接受范围内。主流研究中的贴装过程路径规划方法虽然搜索速度快，但是解的质量偏低，对贴装过程的考虑不全面。集束搜索过程用时和集束宽度之间存在着正相关关系，其增幅主要取决于不同元件对应的贴装点数。对于前 2 个数据 PCB5-1 和 5-2，其各类元件对应的贴装点数较多，贴片头可分配贴装点的候选解较多，因而随着集束宽度的增加，其搜索用时也随之迅速增加，限制了集束搜索宽度的进一步扩大。在实际应用中，根据不同类型元件的贴装点数，调整单一贴片头的贴装点分配策略或将同类元件进行分组，将有助于缩短搜索用时。自适应大邻域搜索整体用时较长，可作为在线运行算法在表面组装过程中实时改进优化，进一步提高组装效率。

表 4-9 基于启发式规则和基于邻域搜索的贴装过程路径规划求解时间

Table 4-9 Solving time of path planning for placement process based on heuristic rules and neighborhood search

PCB	MPGBS-DO				ALNARR (s)
	$\beta = 1$ (s)	$\beta = 2$ (s)	$\beta = 3$ (s)	$\beta = 4$ (s)	
5-1	5.88	23.28	46.52	75.40	57.46
5-2	7.66	30.52	60.85	98.90	77.00
5-3	0.50	1.96	4.02	6.47	28.96
5-4	2.06	8.20	16.25	26.22	51.17
5-5	2.26	8.99	17.82	28.79	58.85
5-6	1.44	5.89	11.71	18.89	52.18
5-7	0.25	0.82	1.65	2.62	40.97
5-8	0.41	1.55	3.05	4.96	26.98
5-9	0.51	2.12	4.16	6.70	37.23
5-10	0.61	2.38	4.76	7.65	26.17

4.5 本章小结

本章研究了适用于表面组装的贴装过程路径规划的启发式算法，结合多轴运动的相关性和移动位置的可变性，提出了基于动态规划的周期内路径规划算法，保证了单个周期路径规划解的最优性。为确定各贴片头分配的贴装点，本章提出了基于多源贪心的动态导向集束路径规划算法，算法以贪心搜索为基础，通过不同的起始点搜索多种解，融合了集束策略并在搜索过程中动态调整搜索方向。为克服贪心式搜索的局限性，本章进一步提出了基于自适应大邻域搜索的聚合路径重构算法，以 Metropolis 接受准则跳出局部最优解，自适应地调整路径的破坏和修复算子权重，通过重新分配贴片头的贴装点改进解的质量。实验结果表明，多源贪心和动态导向的集束搜索有助于增加搜索解的多样性，进而获得了高质量的路径规划解；自适应大邻域路径重构则有效改进了拾贴周期贴装点分配不均的问题，实现了从全局角度对解的持续优化改进。

第5章 表面组装过程中生产线的负载平衡算法

5.1 引言

在第3–4章中,本文已对并列式贴片机的组装过程优化算法开展了深入的研究。为进一步提升组装效率,制造商通常会将多个贴片机级联成生产线,而此过程涉及到生产线的负载平衡问题。负载平衡算法研究的是如何平衡多台贴片机之间的生产任务,通过调整元件在生产线上分配的贴片机,提升生产线中瓶颈设备的组装效率。负载平衡的结果决定了贴片机的生产任务,而贴片机的组装过程效率又用于评估负载平衡分配的结果,两个问题之间的耦合性增加了问题求解的难度。解空间的庞大性以及评估解质量的复杂性是表面组装生产线负载平衡优化任务的主要难点,本章将从组装元件的分配策略以及组装时间的估计两个角度开展研究,提出表面组装生产线的负载平衡算法。

超启发式算法是一类解决复杂优化问题的通用算法,其通过高层次启发式来管理和操作一系列底层启发式算子,实现对大规模组合优化问题的寻优^[113],在复杂问题求解中展现出强大的适应性、有效性和可扩展性。算法由高层启发式和底层启发式构成。其中,高层启发式不依赖于具体问题,直接管理和操纵底层启发式算法,通常具有通用的求解结构和较高的求解效率;底层启发式算子的设计可根据问题特性,融合领域知识和专家经验等,构造多样化的搜索算子,以进一步提升算法的寻优能力。超启发式算法通过决策底层算子的执行顺序,实现了对问题分阶段求解,并可根据解决方案的评估价值,调整底层算子操作的执行顺序。在表面组装过程中,贴装点分配构成了负载平衡的基本决策环节,超启发式算法通过确定不同阶段分配给生产线贴片机的贴装点元件类型和数量,得到了不同的负载分配解决方案。

高效率的迭代搜索是算法求解复杂优化问题的关键,而执行贴片机的组装过程优化用时较长,在有限的时间内不足以全面评估所有负载平衡分配的结果来获取准确组装时间。以机器学习为基础的时间估计器在各类产能预测问题中被广泛应用且具有较高的准确度,特别是在处理复杂、非线性以及多变量交互影响的估计问题上,其优势尤为明显。估计器通过学习历史数据的模式,自动调整模型参数,具备一定的泛化能力,并可以实现实时或接近实时的时间估计,适用于表面组装过程时间估计问题。

5.2 问题分析

生产线的平衡优化通过对各工序的作业内容、时间和效率的分析,改进或调整其中瓶颈工序的作业顺序,以实现机器工作负载的平衡,在过去已有较为充分的研究^[114]。表面组装生产线负载平衡问题可被视为一类特殊的生产线平衡优化问题,电路板上元件组装可被视为不同的生产工序。优化算法将元件分配至不同的贴片机完成装配过程,同类型的元件可被分配到多台贴片机以提高装配效率。表面组装生产线的负载平衡优化更为复杂之处在于装配时间的计算。生产线的效率取决于各贴片机的组装过程效率,以及贴片机类型、可用工具、待组装元件的类型和贴装点数,而贴片机的组装过程效率则与组装任务的元件类型、贴装点数和组装过程的调度优化算法相关。为了实现较高的求解效率,在组装生产线的平衡问题中,解的质量评估无法直接依赖于贴片机表面组装过程优化算法的执行结果,而是需要通过快速的预测算法进行评估。可以认为,组装生产线平衡问题本质上是两个子问题的组合——元件分配算法和组装时间估算器的设计。图5-1展示了组装生产优化中相关的输入/输出和主要决策问题。

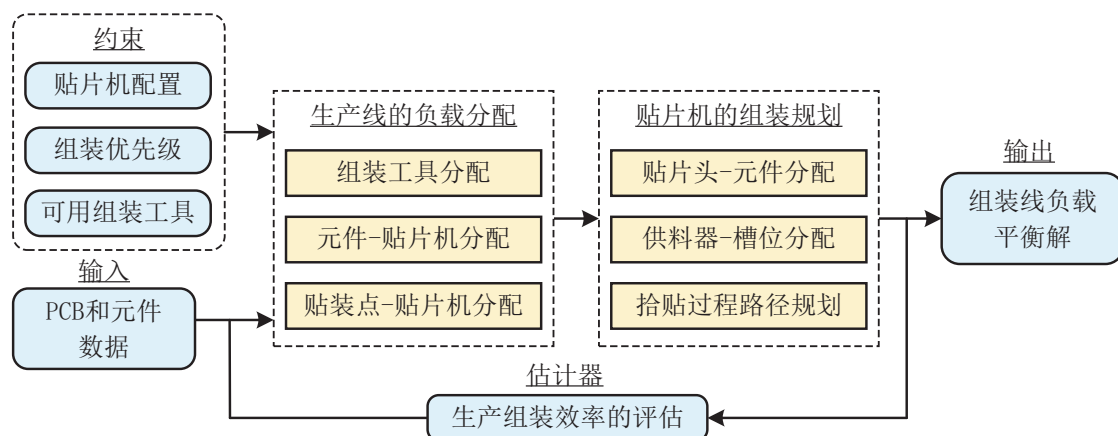


图 5-1 表面组装生产线负载平衡优化问题框架

Fig. 5-1 Framework of load balancing optimization problem for surface assembly lines

表面组装生产线通常由印刷机,贴片机,回流焊和光学检测机等多种设备组成,其中,贴片机的组装时间最长,决定了整条生产线的效率。本文在第2章中已讨论了影响表面组装过程装配效率的因素,包括贴片机的拾贴周期数、元件拾取次数、吸嘴更换次数和贴装点数等组装效率指标。贴片机分配的元件的类型、贴装点数以及对应的吸嘴类型都会对组装效率指标产生较大的影响。元件分配算法用于确定贴片机上各类型元件的贴装点数,对于元件贴装点可重复分配的情形,贴装点的布局会对贴装移动路径产生影响,不同贴装点的分配结

果同样会对生产线的组装效率产生影响。此外，表面组装生产线优化还涉及一系列约束，包括可用工具约束、贴片机类型约束和组装优先级约束。可用工具约束指的是供料器、吸嘴和其他可用组装工具的数量有限，直接限制了特定类型元件可分配贴片机的数量上限，而贴片机类型约束则适用于不同类型贴片机的生产线上协作组装的情形，必须由特定机器完成某一类元件的装配。受限于不同类型的元件高度、尺寸各不相同，表面组装生产线通常也存在优先级约束。

表面组装生产负载平衡问题中，元件不同的组合分配方案数量庞大，设计高效率的搜索方法寻找高质量的解对生产线优化至关重要。而且随着问题规模的扩大，其求解的运算量也会迅速增加，即使是小规模的数据也需要大量的计算资源。启发式搜索算法已在组合优化问题中展现出较强的全局搜索能力，基于问题特征的搜索规则与超启发式算法结合，可以有效提升解的质量。表面组装过程优化通常需要较长的时间才能获取准确时间，即使是求解第3章中具有关键效率的贴片头任务分配算法，大量分配结果的组装过程运算的时间也较长，不适用于具有较大解空间的表面组装生产线优化问题中。在估算装配时间时，时间估计器必须考虑到并列式贴片机的特殊机械结构对组装效率的影响，分析影响组装效率的关键因素，设计提取数据特征，以提高组装时间的估计准确度。贴片机组装数据的元件类型、贴装点数量均会对拾取效率、吸嘴更换等指标产生影响，传统的基于贴装点数量的线性拟合方法已不再适用。非线性的神经网络时间估计器已能解决特征项较多的拟合问题，通过进一步融合多个具有组装特性的特征参数，能实现快速、准确的表面组装时间估计。

5.3 算法框架

以多特征融合集成估计器为基础的超启发式表面组装生产线负载平衡算法流程如图5-2所示，负载平衡以进化算法为基本框架，底层启发式和组装时间估计器则是算法的主要组成部分。元件分组策略用于将同类型元件的贴装点划分为多个元件组，元件组是进行贴片机负载分配的最小单位，元件组分配先后顺序在不同的种群中随机生成以增加搜索的多样性。群体初始化用于确定底层启发式算子的组合方式和执行顺序，群体中个体的基因对应一个底层启发式算子，算子则根据不同的策略确定元件组分配的贴片机。更进一步，底层启发式算子可被分为数据驱动与目标驱动启发式，前者根据数据的基本参数进行分配，后者则基于组装过程子目标的比较结果进行分配。组装时间的估计基于 Bagging 集成学习器，其通过融合组装过程的多个特性准确估计组装时间，用于计算种

群迭代个体的适应度。在迭代过程中，下一代的个体依据适应度进行加权随机选择，进化过程则以个体基因的截断交叉和突变完成。超启发式元件组分配确定了各类型元件分配的贴装点数量，自适应聚类分组用以确定贴装点分配的贴片机。多个群体的最佳解决方案作为候选解，进行表面组装过程优化确定准确装配时间，以降低时间估计器误差对最终结果产生的影响。

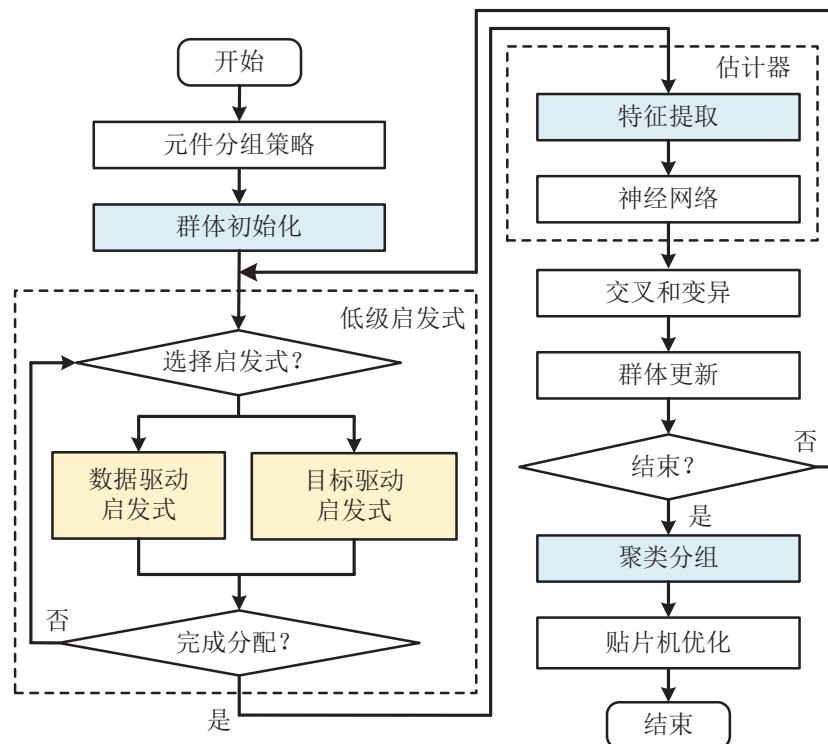


图 5-2 基于多特征融合集成组装时间估算器的超启发式优化算法流程

Fig. 5-2 Flowchart of hyper heuristic optimization algorithm based on multifeature fusion ensemble assembly time estimator

5.4 基于超启发式的表面组装生产线元件分配算法

5.4.1 底层启发式算子

底层启发式是超启发式算法的基本组成，启发式算子根据当前待分配元件的吸嘴类型、贴装点数量等参数确定其分配的贴片机。其中，待分配元件类型和数量为预先设定的。本文将底层分配启发式进一步划分为数据驱动和目标驱动启发式。特别说明的是，底层启发式算子设计旨在平衡各贴片机间的工作负载，其在评估分配指标时仅对各贴片机的某项指标的相对大小进行比较，而对评估的绝对准确性不做要求。两类启发式算子的设计如下所示：

5.4.1.1 数据驱动的底层启发式

基于数据驱动的启发式与贴装点、元件类型和吸嘴类型相关，其分别将元

件分配到已分配贴装点数最少、元件类型数最少、吸嘴类型数最少、元件类型数和吸嘴类型数最小之比的贴片机上。在数据驱动的启发式算子中，分配贴装点数与贴装次数直接相关，是直接决定组装效率的因素之一。将元件组分配给元件类型少的贴片机，则有助于构成更多的同步拾取；将元件组分配给吸嘴类型少的贴片机，可用于规避吸嘴更换动作，提升整体的组装效率；元件类型数-吸嘴类型数之比则为平衡同步拾取与吸嘴更换的相互影响，单一类型吸嘴对应的元件过多会限制其同步拾取的构造，增加贴片头任务分配的可行解数，间接导致搜索空间变大，获得高质量解的难度也随之增大。

5.4.1.2 目标驱动的底层启发式

目标驱动的底层启发式与表面组装过程装配效率影响因素相关，通过分析和比较关键子目标的值作为元件分配的基础。元件组分配的贴片机取决于子目标的相对大小，而无需确定子目标的具体值。启发式计算的子目标值旨在平衡各贴片机之间的工作负载。目标驱动的子目标包括拾贴周期数、吸嘴更换数和元件拾取次数，其与各类吸嘴分配的拾贴头数相关。基于第3章提出的算法3-1，可确定各类型吸嘴对应的贴片头数，记已分配给贴片机 m 的吸嘴类型 j 的对应的贴片头数为 π_{jm} 。

目标驱动的底层启发式的实现如下：

1) 最少周期启发式：将元件分配到拾贴周期数最少的贴片机 \hat{m} 上，即：

$$\hat{m} = \arg \min_{m \in M} \max_{j \in J} \left(\sum_{i \in I} \sum_{k \in K} \sum_{h \in H} (\mu_{ij} \cdot x_{ikhm}) / \pi_{jm} \right) \quad (5-1)$$

组装任务的吸嘴类型单一时，最少周期启发式同时是最少贴装点分配启发式。在处理组装任务的有多种吸嘴类型时，算子在计算拾贴周期数时，禁用了贴片头的吸嘴更换动作，以各类型吸嘴分配的贴装点数均分贴片头得到的最大周期数作为评定标准，能平衡不同类型吸嘴分配点数，进而降低拾贴周期和吸嘴更换数。

2) 最少吸嘴更换启发式：将元件分配到吸嘴更换数最小的贴片机 \hat{m} 上，

$$\hat{m} = \arg \min_{m \in M} \sigma \left(\left\{ \overbrace{\sum_{i \in I} \sum_{k \in K} \sum_{h \in H} (\mu_{ij} \cdot x_{ikhm}) / \pi_{jm}}^{\text{共 } \pi_{jm} \text{ 项}}, \dots \mid j \in J \right\} \right) \quad (5-2)$$

其中 $\sigma(\cdot)$ 表示集合的均方差。吸嘴更换和元件拾取次数存在关联，对吸嘴更换的判定依据主要为不同类型吸嘴分得的贴装点数。算子以各类型吸嘴分得的贴片头数，对其已分配的贴装点均分分组，并根据分组后组间贴装点的均方差值

评估吸嘴更换可能性，均方差越大、对应的贴片机更有可能发生吸嘴更换。

3) 最少拾取启发式：将元件分配给拾取元件次数最小的贴片机。不同于前两种目标驱动启发式，贴片机的拾取次数直接计算比较困难。算法5-1提出了一种分级贪心启发式的同步拾取次数估算的方法。算法根据各类吸嘴分配的贴片头数量，按贴装点数量递减的顺序将元件分配到吸嘴类型一致的贴片头。确定当前分配元件 i 对应的吸嘴 j ，记吸嘴 j 已分配的贴片头数为 \mathcal{A}_j ，当其分配满所有可用贴片头后，进入下一周期 c 进行分配，令吸嘴当前分配周期索引 \mathcal{L}_j 增加 1，同时记录各个周期分配的贴装点数的最大值 \mathcal{K}_c 。算法结果中（同步）拾取次数等于各周期内分配到贴片头的元件对应的最大贴装点数量之和。

算法 5-1 分级贪心启发式同步拾取计算算法

Algo.5-1 Level greedy heuristic simultaneous pick-up calculation algorithm

输入：吸嘴对应的贴片头数 π ，元件对应的贴装点数量 ψ 和贴片机索引 m
输出：拾取次数

- 1 令 \mathcal{L} 为大小 $1 \times |J|$ 的向量， \mathcal{A} 为大小 $1 \times |J|$ 的向量， \mathcal{K} 为 $1 \times \sum_{i \in I} \psi_i$ 全为 0 的向量；
- 2 按贴装点数量 ψ_i 降序排列元件索引 $i \in I$ ；
- 3 **for** $i \in I$ **do**
- 4 $j \leftarrow \sum_{j' \in J} \mu_{ij'} \cdot j'$ ；// 分配元件对应的吸嘴类型
- 5 **if** $\mathcal{A}_j \bmod \pi_{jm} = 0$ **then**
- 6 $\mathcal{L}_j \leftarrow \mathcal{L}_j + 1$ ；// 当前周期吸嘴的贴片头分配已满，开始分配下一周期
- 7 **end**
- 8 /* 更新当前分配周期的最大分配点数量和已分配贴片头数 */
- 8 $c \leftarrow \mathcal{L}_j$, $\mathcal{K}_c \leftarrow \max(\mathcal{K}_c, \psi_i)$, $\mathcal{A}_j \leftarrow \mathcal{A}_j + 1$ ；
- 9 **end**
- 10 输出元件拾取次数 $\sum_{c=1}^{c=\sum_{i \in I} \psi_i} \mathcal{K}_c$

上述的底层启发式规则用于确定元件组分配的贴片机，在分配过程中，底层启发式在评估不同贴片机的指标时如果出现相同的值，算子将先按贴片机已分配的贴装点数量最少、再按贴片机组装先后顺序的规则进行元件组分配。

5.4.1.3 组装任务的元件分配过程

如前所述，元件组是算法进行分配的最小单位，将同类型元件划分为多个元件组有助于贴片机之间的协作生产，平衡贴片机间的工作负载，进而提升组装效率。然而，由于生产线在拾贴元件的过程中同时受到可用贴片机、组装优先级和可用工具等条件的限制，直接按上述规则进行分配将导致负载平衡的解不可行。本节从元件分配优先级处理和可用组装工具限制两个角度对底层启发式算子进行完善。

1) 元件分配优先级的处理

不同于传统的生产线，表面组装生产线不存在复杂的优先级约束关系。组

装过程中的优先级约束通常指元件尺寸小、高度低或精度要求低的元件的组装顺序需要先于其它元件，以保证元件组装位置的准确性。在实际生产中，不同类的元件通常被划为若干优先级组，按分组的优先级顺序进行组装作业。分配过程对元件组装优先级处理主要分为两个阶段。在第一阶段中，算法5-2根据生产线的贴片机数、元件的贴装点数等参数，按贴装点在贴片机间进行均分的设定分组，并在此基础上调整允许偏差界限，确定了优先级约束下不同元件的可分配贴片机索引集合 \tilde{M}^{PR} 。按组装优先级限制元件可分配的贴片机索引，有助于平衡贴片机之间的负载，提升优先级约束下的搜索效率。

算法 5-2 优先级约束下的元件可分配贴片机构造算法

Algo.5-2 Assignable surface mounters constructive algorithm for component under priority constraint

输入：组装优先级分组 \mathcal{G} ，元件的贴装点数 ψ ，贴片机负载分配允许偏差 ς
输出：元件可分配贴片机索引集合 \tilde{M}^{PR}

- 1 按组装先后顺序对元件优先级分组 \mathcal{G}_q 进行排序， $q \in Q$ ，按贴装点数非递减的顺序对 \mathcal{G}_q 组内元件 i 进行排序， $i \in \mathcal{G}_q$ ；
- 2 令元件优先级约束的可分配贴片机索引集合 $\tilde{M}^{\text{PR}} \leftarrow \emptyset$ ，记录已遍历元件的贴装点数 $C \leftarrow 0$ ；
- 3 **for** 遍历元件优先级组内的元件 $i \in \mathcal{G}_q$ ， $q \in Q$ **do**
- 4 **for** 遍历组装线的贴片机 $m \in M$ **do**
- 5 **if** $m \cdot \sum_{i' \in I} \psi_{i'} / |M| - \varsigma \leq C$ **and** $C \leq m \cdot \sum_{i' \in I} \psi_{i'} / |M| + \varsigma$ **then**
- 6 /* 已遍历的贴装点数在贴片机的贴装点均分的允许偏差范围内 */
- 6 $\tilde{M}^{\text{PR}} \leftarrow \tilde{M}^{\text{PR}} \cup \{m\}$ ， $C \leftarrow C + \psi_i$ ；
- 7 **end**
- 8 **end**
- 9 **end**

在第二阶段中，分配算子需依据待分配的元件组和既定的优先级约束关系，对元件组的负载平衡分配结果进行调整。算法5-2确定了优先级约束下的元件可分配贴片机索引集，但其无法完全保证解的可行性。由于不同优先级的元件可分配至相同的贴片机完成组装作业，因此元件的可分配贴片机索引集 \tilde{M}^{PR} 间会有同一贴片机。为满足组装过程的优先级约束，根据各贴片机所分配元件的最高与最低优先级，本节引入了链式置换规则来优化元件组合的分配过程。

在具体实现过程中，对于当前待分配的元件，算法依据其分配目标贴片机进行比对：若前置贴片机的最低优先级高于当前元件的优先级，则将当前元件与前置贴片机的元件进行交换；类似地，若后置贴片机的最高优先级低于当前元件的优先级，则与后置贴片机的元件进行互换。重复上述操作，直至生产线上所有已分配的元件均满足优先级约束关系。在选择被置换的元件时，算法根据待分配元件的吸嘴类型和贴装点数，按照如下规则进行：若被置换出的贴片机已分配元件中存在同类型吸嘴的元件，则优先选择与当前待分配元件贴装点数

最接近的元件进行交换；若不存在同类型吸嘴的元件，则从吸嘴对应已分配元件类型最多的类别中，选择与待分配元件组贴装点数最接近的元件进行置换。此规则旨在最大限度地减少因元件置换而引起的贴片机间组装效率的变化，从而保持生产线上贴片机之间的负载平衡状态。

2) 可用组装工具的限制

可用贴片机约束限制了元件可分配的贴片机集合，底层启发式算子需在可用贴片机集合中完成分配。可用贴片机集合受到元件已分配贴片机和可用工具数的限制，元件供料器可用数等限制了元件可分配的贴片机数，记 \tilde{r}_{im} 表示贴片机 m 是否已分配元件 i 的组装任务，记元件 i 的可用供料器数为 ϕ_i ，则分配过程中可用组装工具限制的可分配贴片机索引集 \tilde{M}^{TL} 计算如式 (5-3) 所示，其在元件负载平衡分配的过程中被动态调整。

$$\tilde{M}_i^{\text{TL}} = \begin{cases} M & \sum_{m \in M} \tilde{r}_{im} < \phi_i \\ \{m \mid \tilde{r}_{im} > 0, m \in M\} & \sum_{m \in M} \tilde{r}_{im} \geq \phi_i \end{cases} \quad (5-3)$$

组装分配优先级和可用组装工具确定的可分配贴片机索引集共同确定了元件组可分配的贴片机。

5.4.2 可重复元件分配策略

元件配备多个可用供料器用以提升组装效率，引出元件可重复分配的问题。当同类型元件的物料有多个可用供料器时，元件可分配至多个贴片机协同完成组装过程。元件分配的供料器数依据第 3 章中提出的算法 3-6 确定。元件可重复分配策略包括进行元件分配前的分组策略和完成元件分配后的贴装点分配策略，具体为：

1) 可重复元件分组策略

贴装点分组是辅助决策贴片机分配的元件贴装点数的主要方式，分组后的元件组是超启发式分配的基本单位。贴装点的分组数并不仅取决于可用供料器数，而且与贴装点数有关。过少的分组数会压缩搜索解的搜索空间，导致算法无法搜索到高质量的解，无法平衡不同的贴片机之间的工作负载；过多的分组数则会导致搜索效率下降。记元件 i 可用的供料器数为 ϕ_i ，对应的分组基准为 $\hat{\phi}_i$ ，分组基准的计算方式如式 (5-4)，其中 ε 为分组基准系数。贴装点数超过该分组基准的元件将被分配到一个元件组，各元件组的贴装点数均不超过分组基准，元件组是超启发式进行元件分配的最小单位。

$$\hat{\phi}_i = \max \left(\varepsilon \cdot \phi_{i'} \cdot \psi_i / \sum_{i' \in I} \psi_{i'}, \phi_i \right) \quad \forall i \in I \quad (5-4)$$

2) 可重复元件贴装点的分配策略

超启发式算法仅确定贴片机分配到的各类型元件的贴装点数，而贴装点的布局会对组装效率产生较大的影响，不同的贴装点分配方案同样限制着生产线组装效率的提升。元件可重复分配策略研究的另一方面就是对贴装点-贴片机的分配结果进行优化。包括本文在内的多数主流研究均将贴片机的表面组装过程划分为拾取过程和贴装过程两部分，而拾取过程优化通常与贴装点的分布位置关联度较低。算法5-3提出了一种基于聚合聚类启发式的可重复元件贴装点分配策略，其充分考虑了贴装点的位置分布特性和贴片头线性分布的特性，将位置集中的贴装点分配给同一台贴片机，能有效缩短表面组装移动路径长度，从而进一步提升贴片机的组装效率。

可重复元件贴装点的分配策略的输入为贴片机 m 在周期组 k 中贴片头 h 拾贴的元件类型为 $C_{k h m}$ ，以及对应的拾贴周期数 $W_{k m}$ 。据此，可确定各贴片机组装作业过程中，不同类型元件的贴装偏移量 $\bar{h}_{i m}$ ，其是元件组贴装点分配的基准。对于仅有单一供料器提供的元件的贴装点，其被分配至同一台贴片机，结合组装该类型元件贴片机的贴装偏移量 \bar{h} ，可确定各贴片机的初始中心点，而尚未分配任何贴装点的贴片机的初始中心点将被随机设定。算法分配的基本单位由贴片机进一步具体到贴片头，根据贴片头任务分配结果，可确定其拾贴元件 i 对应的贴装点数 $\Lambda_{i h m}$ 。随后，元件的贴装点按最近邻原则，在负载平衡结果的约束下，分配到距离中心点最近的贴片机-贴片头索引对 (\hat{m}, \hat{h}) ，并在分配过程中动态调整中心点。算法不断重复上述过程，直至所有中心点的位置趋于稳定。

5.4.3 超启发式优化算法

在基于进化算法的超启发式架构中，个体的编码直接映射至一个底层的启发式操作算子。元件组作为同类型元件分配的基本单位，其分配顺序影响着负载平衡的结果。为适应复杂的分配需求，个体的基因长度随机生成。底层启发式算子与元件组相对应，当个体基因长度小于元件组数时，元件分配的过程将循环访问个体基因。个体的基因长度以及底层启发式算法的基因组合均在初始化阶段随机生成，以确保搜索的全面性和随机性。在进化流程中，算法依据适应度进行加权随机选择，确定参与交叉和变异的个体。对于交叉操作，相关个体的基因将分别随机选择一个插入段交换基因段；对于变异操作，则在个体基

算法 5-3 可重复元件贴装点分配的聚合聚类算法

Algo.5-3 Aggregated clustering algorithm for placement points of duplicated component

输入：元件的可用供料器数 ϕ ，贴装点的位置 (X_p, Y_p) ， $p \in P$ ，元件-贴片机的分配结果 r ，贴片头元件分配 C 和周期分配结果 \mathcal{W}

输出：机器分配的贴装点 \mathcal{P}

- 1 令贴片机 m 已分配的贴装点集 $\mathcal{P}_m \leftarrow \emptyset$ ，贴片机 m 的贴片头 h 已分配元件 i 的贴装点 $\Lambda_{ihm} \leftarrow 0$ ，元件贴装的贴片头偏移 $\bar{h}_{im} \leftarrow 0$ ， $i \in I$ ， $h \in H$ ， $m \in M$ ；
- 2 **for** $m \in M$ **do**
- 3 **for** $i \in \{i' \mid r_{i'm} = 1, \phi_{i'} = 1, i' \in I\}$ **do**
- 4 $\mathcal{P}_m \leftarrow \mathcal{P}_m \cup \{p \mid \eta_{ip} = 1, p \in P\}$; // 确定各贴片机的固定分配点
- 5 **end**
- 6 **for** $k \in K$ **do**
- 7 **for** $h \in H$ **do**
- 8 记 $i = C_{k h m}$ ，有 $\Lambda_{ikm} \leftarrow \Lambda_{ikm} + \mathcal{W}_{km}$ ，
- 9 $\bar{h}_{im} \leftarrow (1 - \mathcal{W}_{km} / \sum_{h \in H} \Lambda_{ihm}) \cdot \bar{h}_{im} + \mathcal{W}_{km} \cdot (h - 1) / \sum_{h \in H} \Lambda_{ihm}$ ；
- 10 **end**
- 11 **end**
- 12 $\mathcal{X}_m \leftarrow \sum_{p \in \mathcal{P}_m} (X_p - \sum_{i \in I} \eta_{ip} \cdot \bar{h}_{im}) / |\mathcal{P}_m|$ ， $\mathcal{Y}_m \leftarrow \sum_{p \in \mathcal{P}_m} Y_p / |\mathcal{P}_m|$; // 设置贴片机 m 的分配中心点
- 13 **end**
- 14 **while true do**
- 15 $\bar{\mathcal{X}} \leftarrow \mathcal{X}$ ， $\bar{\mathcal{Y}} \leftarrow \mathcal{Y}$ ， $\bar{\Lambda} \leftarrow \Lambda$ ， $\bar{\mathcal{P}} \leftarrow \mathcal{P}$ ；
- 16 **for** $i \in \{i' \mid \theta_{i'} > 1, i' \in I\}$ **do**
- 17 **for** $p \in \{p' \mid \eta_{ip'} = 1, p' \in P\}$ **do**
- 18 /* 计算待分配贴装点的贴片机和贴片头索引 */
- 18 $(\hat{m}, \hat{h}) \leftarrow \arg \min_{m \in M, h \in H} \left\{ (\bar{\mathcal{X}}_m - X_p + (h - 1) \cdot \rho)^2 + (\bar{\mathcal{Y}}_m - Y_p)^2 \mid \bar{\Lambda}_{ihm} > 0 \right\}$ ， $\bar{\mathcal{P}}_{\hat{m}} \leftarrow \bar{\mathcal{P}}_{\hat{m}} \cup \{p\}$ ， $\bar{\Lambda}_{ihm} \leftarrow \bar{\Lambda}_{ihm} - 1$ ， $\hat{h} \leftarrow \hat{h} - 1$ ；
- 18 /* 更新已分配的贴装点参数和贴片机的分配中心位置 */
- 19 $\bar{\mathcal{X}}_{\hat{m}} \leftarrow \bar{\mathcal{X}}_{\hat{m}} + (X_p - \bar{\mathcal{X}}_{\hat{m}} - \hat{h} \cdot \rho) / |\bar{\mathcal{P}}_{\hat{m}}|$ ， $\bar{\mathcal{Y}}_{\hat{m}} \leftarrow \bar{\mathcal{Y}}_{\hat{m}} + (Y_p - \bar{\mathcal{Y}}_{\hat{m}}) / |\bar{\mathcal{P}}_{\hat{m}}|$ ；
- 20 **end**
- 21 **end**
- 22 **if** $\sum_{m \in M} (|\bar{\mathcal{X}}_m - \mathcal{X}_m| + |\bar{\mathcal{Y}}_m - \mathcal{Y}_m|) \leq 10^{-3}$ **then**
- 23 /* 贴片机的分配中心位置未发生变化，退出搜索过程 */
- 23 **break**；
- 24 **end**
- 25 **end**

因中随机选择一个插入点，插入一段随机生成的基因。二者在完成操作后，算法均会对基因长度超出元件组数的部分截断处理，以确保后续迭代过程中不会因为无效的基因段导致搜索的盲目性。

底层启发式操作算子与元件分配顺序直接相关，同时对启发式算子执行顺序和元件分配顺序编码，会增加编码的复杂性，导致算法难以找到算子和元件分配间的规律，进而增加了寻找高质量解的搜索时间。因此，本章并未将元件分配顺序作为基因编码的组成部分。对超启发式底层算子执行顺序的优化能将多

数元件分配顺序优化到一个相对高质量的解。在算法设计时，超启发式采用多种群的优化方案，既可以实现并行搜索，又避免了单一元件组分配顺序在有限次迭代内无法获得高质量的解。不同种群迭代的最佳解决方案在估算的组装时间通常较接近，而应用多种群的方案可进一步筛选高质量的解，通过在少数候选解上执行表面组装过程优化算法获得更准确的装配时间。在具体实施时，算法将优先在估计的组装时间较长的贴片机上运行组装过程优化算法，以进一步提升运行效率。

5.5 基于多特征融合的集成表面组装时间估计器

5.5.1 数据的生成与准备

组装数据用于组装时间估计器模型的训练过程。估计器通过处理和分析数据，学习其特征并建立预测模型。生产数据的数量、质量和多样性直接影响着时间估计器预测的性能和效果，更多的训练数据意味着网络能学习到更为复杂的特性，从而提升神经网络的泛化性能。高质量的数据通常噪声少，结果准确，有助于网络更准确地学习。对于表面组装过程问题，直接获取大量的实际生产数据较为困难，且生产数据的组装时间不可避免的受到噪声的影响。在实际应用中，本章以贴片机软件内置的组装过程模拟器，计算生产数据的组装过程用时，以获得更为准确的预测模型。

在数据生成方面，用以训练的数据兼顾吸嘴类型单一到多样，不同元件的贴装点数相近到不等，贴装点分布位置随机分布的特点进行生成，避免贴装点布局对预测结果产生偏差，进而影响整体的预测效果。在训练模型之前，有必要检测并清除可能影响机器学习模型的异常值，偏离中心位置的点会增加训练结果的不确定性。本章运用四分位距的规则（Interquartile Range Rule, IRR）^[103] 检测和剔除异常值，其中装配时间 \mathcal{T} 大于 $\mathcal{T}_3 + 1.5 \cdot (\mathcal{T}_3 - \mathcal{T}_1)$ 或小于 $\mathcal{T}_1 - 1.5 \cdot (\mathcal{T}_3 - \mathcal{T}_1)$ 的数据点会被移除， \mathcal{T}_1 和 \mathcal{T}_3 分别代表第一四分位数和第三四分位数。

5.5.2 组装过程的特征提取

特征选择是预测拟合问题的关键步骤，其对于提高模型的准确性、可解释性和计算效率都至关重要。特征提取涉及从生产数据中选择对于预测组装时间最有用或最具代表性的特征，其与表面组装过程优化算法相关。组装过程的复杂性使某些生产过程特性难以被揭示，因而本章提出一种基于集成学习框架的多特征数据融合时间估计器，将数据的基本参数和估计的性能指标参数等同时

作为网络输入数据。其中，启发式算法用于评估性能指标，提高拟合精度。用于训练和测试的数据特征主要包含以下内容：

- 贴装点数、元件类型数和吸嘴类型数
- 贴装区域的长度、宽度和面积
- 预估的拾贴周期数、拾取次数、吸嘴更换次数等子目标
- 各类型元件对应的吸嘴类型编码和贴装点数

算法 5-4 吸嘴更换次数评估启发式算法

Algo.5-4 Heuristic algorithm for estimating the number of nozzle changes

输入：吸嘴的贴片头数 π ，元件的贴装点数 ψ
 输出：吸嘴更换次数 ι^*

```

1  令  $\mathcal{H}^{\text{PT}}$  为  $1 \times |H|$  的全 0 向量,  $\mathcal{H}^{\text{NZ}}$  为  $1 \times |H|$  的向量; // 分别表示贴片头上的贴
    装点数和吸嘴类型数
2  令  $V \leftarrow 0$ ,  $V^* \leftarrow \infty$ ,  $\iota^* \leftarrow 0$ ; // 分别表示评估指标、最优评估指标和对应的吸嘴
    更换次数
3  while  $V \leq V^*$  do
4      令吸嘴组  $\mathcal{Z}_j$  为大小  $1 \times \pi_j$  的元素值为  $\sum_{i \in I} \psi_i \cdot \mu_{ij} / \pi_j$  的向量,  $\forall j \in J$ ;
5      for  $n \in \mathcal{Z}_j, j \in J$  do
6           $h \leftarrow \arg \min_{h' \in H} \{\mathcal{H}_h^{\text{PT}}\}$ ,  $\mathcal{H}_h^{\text{NZ}} \leftarrow j$ ,  $\mathcal{H}_h^{\text{PT}} \leftarrow \mathcal{H}_h^{\text{PT}} + n$  // 分配吸嘴组至各贴
            片头
7      end
8       $V \leftarrow \max_{h \in H} \mathcal{H}_h^{\text{PT}}$  // 设置拾贴周期数
9      while true do
10         /* 平衡已分配贴装点数最多的贴装和最少的贴片头 */
11          $h_1 \leftarrow \arg \max_{h \in H} \mathcal{H}_h^{\text{PT}}$ ,  $h_2 \leftarrow \arg \min_{h \in H} \mathcal{H}_h^{\text{PT}}$ ;
12         if  $\mathcal{H}_{h_1}^{\text{NZ}} = \mathcal{H}_{h_2}^{\text{NZ}}$  then
13             break;
14         end
15         /* 比较周期减少与吸嘴更换的加权值, 确定最优方案 */
16          $\mathcal{H}_1 \leftarrow \{h \mid \mathcal{H}_h^{\text{NZ}} = \mathcal{H}_{h_1}^{\text{NZ}}, h \in H\}$ ,  $\mathcal{H}_2 \leftarrow \{h \mid \mathcal{H}_h^{\text{NZ}} = \mathcal{H}_{h_2}^{\text{NZ}}, h \in H\}$ ;
17         if  $T^{\text{CY}} \cdot (\mathcal{H}_{h_1}^{\text{PT}} - \mathcal{H}_{h_2}^{\text{PT}}) > T^{\text{NC}} \cdot (|\mathcal{H}_2| - |\mathcal{H}_1|)$  then
18             break;
19         end
20         /* 计算吸嘴更换数和吸嘴更换后的分配结果 */
21          $\iota \leftarrow |\mathcal{H}_2| - |\mathcal{H}_1|$ ,  $V \leftarrow V - T^{\text{CY}} / T^{\text{NC}} \cdot (\mathcal{H}_{h_1}^{\text{PT}} - \mathcal{H}_{h_2}^{\text{PT}}) + \iota$ ,  $\hat{\mathcal{H}}^{\text{PT}} \leftarrow \mathcal{H}^{\text{PT}}$ ;
22         for  $h \in \mathcal{H}_1 \cup \mathcal{H}_2$  do
23              $\mathcal{H}_h^{\text{PT}} \leftarrow \sum_{h' \in \mathcal{H}_1 \cup \mathcal{H}_2} \hat{\mathcal{H}}_{h'}^{\text{PT}} / (|\mathcal{H}_1| + |\mathcal{H}_2|)$ ,  $\mathcal{H}_h^{\text{NZ}} \leftarrow \mathcal{H}_{h_1}^{\text{NZ}}$ ;
24         end
25     end
26 end
    
```

生产数据的基本参数类型包含贴装点数、元件类型数、吸嘴类型数和电路

板尺寸等与组装信息密切相关的数据，同时包含组装过程优化的子目标等性能参数。与底层启发式为了满足任务负载均衡的设计不同，用于拟合特征更侧重于估计结果的准确性。目标项在编码过程中主要用于反映组装过程优化算法的性能，在目标数据特征中，拾贴周期数可参照式(5-1)，根据贴装点数和吸嘴分配的贴片头数确定，应用算法5-1可进一步确定评估的同步拾取次数，算法5-4则提出了一种吸嘴更换计算启发式。

吸嘴更换计算启发式首先将具有相同吸嘴类型的元件的贴装点均分形成不同的吸嘴组，其中吸嘴 j 对应的吸嘴组集合记为 \mathcal{Z}_j 。吸嘴组按照贴片头工作平衡的原则进行分配，算法先从空吸嘴头开始，依次分配给已分配贴装点数量最少的贴片头。初次分配完成后，算法平衡贴片头上的贴装点，并评估均分后整体指标的变化。如果均分后减少的拾贴周期带来的效率提升大于增加吸嘴更换带来的效率损失，那么此次均分是有效的，重复均分过程直至出现无效的操作。随后，增加平均贴装点数最多的吸嘴的分组数，重复上述操作过程，并记录搜索过程中加权值最优对应的吸嘴更换次数，直到总体效率没有增加为止。

5.5.3 集成学习估计器的结构设计

集成学习通过结合多个学习器的预测结果提供了一种提高预测性能的学习范式，可以降低模型的泛化误差。在机器学习算法中，全连接神经网络以其简单的结构、高效的运算速度、强大的特征提取能力等优势，在数据拟合和预测等领域中发挥着重要作用。本章中，融合多数据特征的 **Bagging** 集成学习器用于估计表面组装时间；集成学习共由 5 个基模型并行训练，其中的基模型采

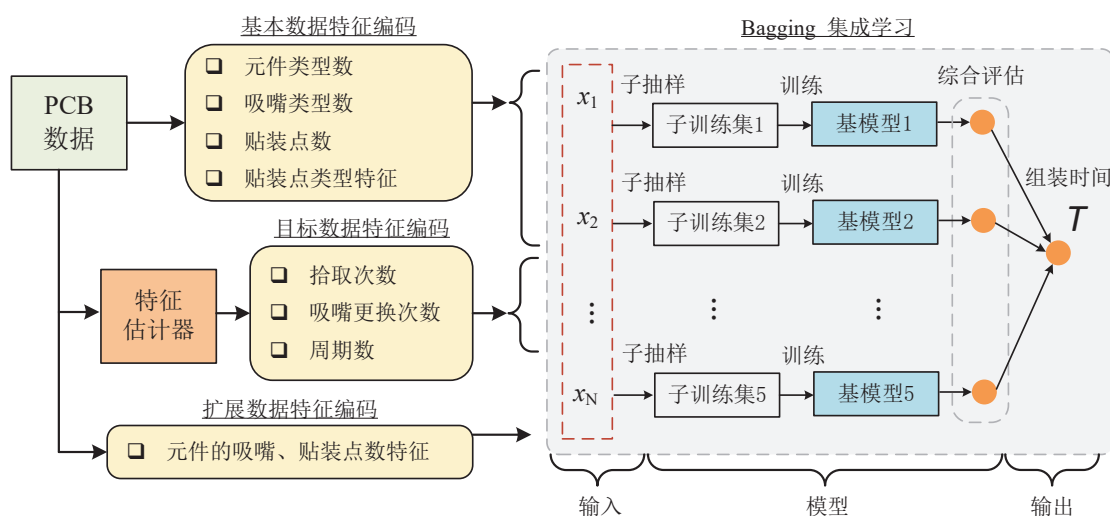


图 5-3 基于集成学习的多特征融合组装时间估计器结构

Fig. 5-3 Structure of ensemble learning-based multifeature fusion assembly time estimator

用具有双隐藏层的全连接神经网络，并取基模型输出结果的平均值作为最终组装时间预测结果。在估计器的输入数据的编码上，分为基本数据类型，目标数据类型和扩展数据类型三部分。其中，基本数据类型包含元件类型数、吸嘴类型数和贴装点数；目标数据类型则采用前一节估算的同步拾取数、吸嘴更换数等特征；扩展数据类型则包含不同元件的吸嘴类型、元件类型等参数。在扩展数据类型中，各吸嘴、元件类型的编码按贴装点数降序进行编号。图5-3展示了估计器的输入参数和基本结构。估计器的输入数据设定为足够长的编码，以确保不同数据输入的一致性，并用零补充冗余位。最终的输出结果则采用标准化的单位时间贴装点数进行归一化处理。

5.6 实验设计

为说明本章提出的基于多特征融合集成时间估计器的超启发式组装线优化（Hyper-Heuristic Assembly Line Optimization based on Multifeature Fusion Ensemble Estimator, HHALO-MFEE）算法的实际效果，本节将对比分析组装时间估计器的拟合精度，将生产线负载平衡算法的运行结果同模型的性能指标加权最优解进行比较，最后将 HHALO-MFEE 同主流的相关研究进行比较。对比实验中参数设置和程序运行配置同第3—4章相同。表5-1列出了超启发式元件分配和多特征融合集成组装时间估计器的基本参数。10组随机生成的元件分配序列用于迭代搜索元件分配的最优解，种群的个体数为20，种群迭代的总次数为100，交叉和变异发生的概率为60%和10%。超启发式元件分配过程中，元件分组基准的系数 $\varepsilon = 1.5$ 。组装时间估计器的基模型时采用双中间层的全连接神经网络，各层的神经元个数为1000，网络的不同层中使用相同的 Relu 作为激活函数，并运用 Adam 梯度下降算法自适应地调整学习率。

表 5-1 神经网络和超启发式算法的参数

Table 5-1 Parameters of neural networks and hyper-heuristic algorithm

方法	参数	设定值
超启发式	种群大小	10
	个体数	20
	交叉率 变异率	0.6 0.1
	迭代次数	100
神经网络	学习率	10^{-5}
	迭代次数	8000

在实验数据选取上，15组来自实际表面组装生产线的的数据用于评估生产线负载平衡算法的实际效果，其中前5组小规模数据用于模型对比验证，后10组

大规模数据用于算法效果对比,数据的基本参数包括元件类型数、吸嘴类型数、贴装点数和可用供料器数。当可用供料器数多于元件类型数时,元件可重复分配至不同的贴片机完成组装任务。三条不同的表面组装生产线用于比较算法的实际效果,其中 $L1$ 、 $L2$ 和 $L3$ 分别配备有 2、3 和 4 台贴片机。本节仅以各优化算法在一般应用场景下的实际效果作对比分析,不计组装优先级、可用贴片机等约束。由于元启发式算法和部分对比算法结果的随机性,所有比较指标均取 10 次运行的平均值作为最终结果。

5.6.1 组装时间估计器的对比实验

如前所述, MFEE 的训练数据和测试数据均为随机生成,组装任务的拾贴顺序采用第 3–4 章提出的表面组装过程优化算法。通过运行贴片机内置的模拟器,估计组装过程的运动和动作用时,结合电机参数、机械运动特性等获取更为准确的组装时间。表5-2列出了训练数据和测试数据的样本数、异常比、均值等组装时间相关的参数,二者具有相似的分布特性,数据的异常值是通过 IRR 检测和剔除的。贴装点的分布位置随机生成避免点稀疏或集中的分布特征对拟合方法泛化性能的影响。

表 5-2 训练数据和测试数据的参数
Table 5-2 Parameters of training data and testing data

	样本数	异常比	平均值	中位数	最小值	最大值	均方差
训练集	2000	11.25	128.67	130.13	2.71	302.94	71.67
测试集	400	10.75	126.76	127.11	3.80	311.38	72.23

时间估计器的准确性会影响元件分配的搜索方向和最终解的质量。4 种具有代表性的组装时间估计器用于和 MFEE 进行比较。本节所提出的组装时间估计器的准确度记为 E_1 ,为说明提出的特征融合的有效性,具有相同结构的,以贴装点数、元件类型数、吸嘴数、电路板大小等基本参数为输入的神经网络的拟合准确度记为 E_2 。文献 [79] 和文献 [85] 中提出的启发式估计器选取不同的特征对组装时间进行估计,其均采用线性拟合的方式进行估计,本节中以最小二乘法计算线性拟合的系数,组装时间估计的准确度分别记为 E_3 和 E_4 。在最新的研究中,文献 [103] 提出一种基于共生搜索的支持向量回归的集成学习算法,其时间估计准确度记为 E_5 。

表5-3列出了不同组装时间估计器在训练数据和测试数据的平均绝对误差和最大绝对误差。组装时间估计器在测试数据上的表现可用于评价时间估计器的准确性。可以看出,以神经网络为基础的组装时间估计器在准确度上更有优

势，与输入基本参数的方法相比，所提出的 MFEE 方法将测试数据上获得的组装时间的平均绝对误差从 5.16% 降低到 3.43%。基于启发式的线性回归拟合方法在预测效果上差于神经网络估计器，对估计特征的考虑不全面和自身结构的简单导致拟合结果不佳，平均绝对误差分别为 9.41% 和 9.44%。基于共生生物搜索算法的支持向量回归算法尽管已经验证在组装车间生产线上对组装时间的预估结果更为准确，但由于其忽略了单个电路板的组装过程的特性，因此拟合精度最低。需要指出，由于贴装点分布的随机性，本章提出的 MFEE 和用于对比的组装时间估计方法，在预测准确度上均存在误差。

表 5-3 多特征融合集成组装时间估计器和其他估计器的准确度比较

Table 5-3 Comparison of the accuracy of the multifeature fusion ensemble assembly time estimator and other estimators

类别	参数	E_1	E_2	E_3	E_4	E_5
训练集	平均绝对误差 (%)	2.01	5.09	8.75	8.75	45.30
	最大绝对误差 (%)	18.80	21.28	37.61	37.68	214.94
测试集	平均绝对误差 (%)	3.43	5.16	9.41	9.44	45.99
	最大绝对误差 (%)	16.57	18.65	27.65	28.82	183.98

5.6.2 生产线元件分配算法的对比实验

5.6.2.1 同数学规划法比较

数学规划法可以最优地求解小规模的组合优化问题，为说明本章提出的算法与理论最优值的差距，本节将其同第 2 章组装线负载平衡模型进行比较，该模型是通过提取影响表面组装过程效率的关键子目标建立的数学模型，建立模型的结果接近实际组装时间，模型的权重设定与第 2 章相同。在比较过程中，模型和所提出的算法均不计贴装点布局对装配效率的影响。表 5-4 列出了用于比较的小规模电路板 PCB 数据组 6，其元件类型数和对应点数均较少。模型和超启发式算法求解所得的加权性能指标分别记为 \mathcal{O}^M 和 \mathcal{O}^H ，如表 5-5 所示。同模型所得的最优解相比，超启发式负载平衡在 3 条生产线上同最优解的平均差距记为 $\mathcal{G} = (\mathcal{O}^H / \mathcal{O}^M - 1) \cdot 100\%$ ，分别是 7.28%、6.58% 和 3.44%。由此可见，超启发式算法在求解小规模数据时的性能接近模型求解的结果，而其更高的求解效率使得其可以应用于更大规模的组装数据的优化。

5.6.2.2 同其它元件分配算法比较

生产线负载平衡的任务是将元件分配到生产线的贴片机中，以实现其中生产瓶颈的贴片机组装时间最小化的目标。本节将比较提出的 HHALO 和主流研究在实际生产中的表现，表 5-6 列出了用于比较的大规模 PCB 数据组 7 的基本参

表 5-4 小规模 PCB 组装生产线的数据参数
Table 5-4 Small-scale data parameters for PCB assembly lines

PCB	6-1	6-2	6-3	6-4	6-5
元件类型数	4	4	5	5	5
吸嘴类型数	3	3	3	2	2
贴装点数	28	34	34	30	30
可用供料器数	10	6	8	7	5

表 5-5 加权关键性能指标数学模型与超启发式负载平衡算法的比较
Table 5-5 Comparison of weighted key metrics between mathematical model and hyper heuristic workload balancing algorithm

生产线	L1			L2			L3		
	Θ^M	Θ^H	ζ (%)	Θ^M	Θ^H	ζ (%)	Θ^M	Θ^H	ζ (%)
6-1	2.585	2.626	1.59	1.758	1.837	4.49	1.676	1.813	8.17
6-2	3.286	3.672	11.75	2.785	3.122	12.10	2.473	2.514	1.66
6-3	2.719	2.998	10.26	2.218	2.445	10.23	1.947	2.054	5.50
6-4	2.744	3.017	9.95	2.202	2.314	5.09	2.202	2.243	1.86
6-5	2.933	3.017	2.86	2.432	2.456	0.99	2.432	2.432	0.00
平均值			7.28			6.58			3.44

数，用于对比的元件分配算法包括集成选择式分配启发式（Integrated Selective Allocation Heuristic, ISAH）算法，混合重构式产线优化（Hybrid Reconfiguration Assembly Line Optimization, HRALO）算法以及组装线负载平衡求解器（Assembly Line Workload Balancing Solver, ALWBS）。其中，ALWBS 是某贴片机生产制造商于 2022 年发布的内置于生产线管理软件的负载平衡优化器；HRALO 算法以 Tóth 等^[85]提出的混合优化算法为基本框架，进一步结合了 Mumtaz 等^[81-82]提出的 PCB 组装线优化中的混合蜘蛛猴算法的编码规则与搜索方式；Guo 等^[79]提出的 ISAH 则是适用于表面组装生产线的基于进化的改进方法，其通过设计启发式算子搜索解的可行域，为 PCB 组装线优化提供实用有效的解决方案。HHALO、ALWBS 和 ISAH 均提供了元件可重复分配的解决方案，HRALO 则未给出解决方案，因此在对比实验中采用了 5.4.2 节所提出的分组策略。在贴片机组装过程优化方面，ALWBS 通过内置程序优化表面组装过程，其余方法则采用第 3—4 章提出的表面组装过程优化方法。

表 5-6 大规模 PCB 组装生产线的数据参数
Table 5-6 Large-scale data parameters for PCB assembly lines

PCB	7-1	7-2	7-3	7-4	7-5	7-6	7-7	7-8	7-9	7-10
元件类型数	16	29	7	24	45	7	47	40	10	40
吸嘴类型数	3	3	3	3	4	4	4	2	3	4
贴装点数	78	165	192	236	209	320	390	546	720	1510
可用供料器数	19	30	12	28	47	14	54	50	19	40

HHALO 在不同的生产线配置上均取得的有益的效果, 记其组装时间为 $\mathcal{T}^{\text{HHALO}}$, 对比方法 ALWBS、HRALO 和 ISAH 的组装时间分别被记为 $\mathcal{T}^{\text{ALWBS}}$, $\mathcal{T}^{\text{HRALO}}$ 和 $\mathcal{T}^{\text{ISAH}}$, 表5-7列出了上述 4 种算法在不同组装生产线上大规模电路板数据的负载平衡优化所得的装配时间, 相较于此三种算法, 本章所提出的 HHALO 算法适用于不同生产配置的并列式贴片机组装生产线, 在组装效率上分别提高了 7.21%, 8.67% 和 9.47%。随着生产线装配的贴片机数量的增加, 负载分配的可行解空间也随之增加, 而 HHALO 算法在搜索高质量的解的方面也更具有优势。由表可知, HHALO 和同类型的工业软件中的 ALWBS 有着相近的表现, 虽然存在少量数据的 HHALO 的解不如 ALWBS, 但 HHALO 整体上平均组装效率更优。ISAH 和 HRALO 均是基于进化算法的搜索框架, 而其算子的设计相对简单, 对并列式贴片机组装效率的影响因素考虑不足, 加之对搜索到的解的时间评估也不准确, 因而整体组装时间较长。

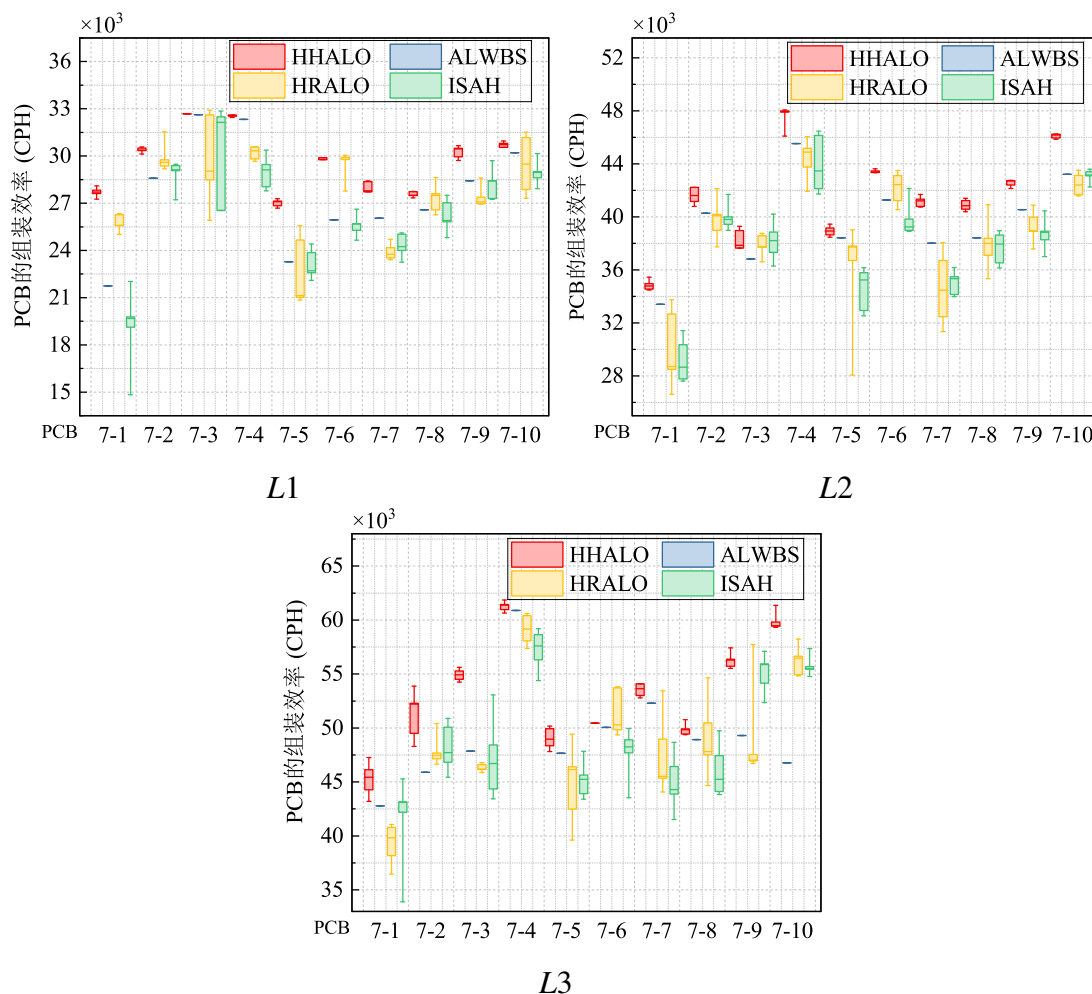


图 5-4 不同方法的装配效率优化结果在三条印刷电路板装配线上的分布比较
Fig. 5-4 Comparison of the distribution of assembly efficiency optimization results of different methods on three PCB assembly lines

此外,图5-4比较了不同优化算法在3条表面组装生产线上优化得到的装配时间的数据分配情况。在包含随机搜索算子的优化算法中,HHALO算法解的分布更为集中,且整体优于其他两种组装优化算法。ISAH和HRALO作为基于进化搜索的优化算法,搜索过程中的随机算子扩大了算法的搜索范围,但也导致解的分布方差较大、输出的结果不稳定。虽然ISAH部分解的质量优于HRALO

表 5-7 超启发式组装线优化器与主流算法的装配时间比较

Table 5-7 Comparison of assembly time between the hyper-heuristic assembly line optimizer and mainstream algorithms

PCB	产线	HHALO	ALWBS		ISAH		HRALO	
		$\mathcal{T}^{\text{HHALO}}$ (s)	$\mathcal{T}^{\text{ALWBS}}$ (s)	\mathcal{G} (%)	$\mathcal{T}^{\text{ISAH}}$ (s)	\mathcal{G} (%)	$\mathcal{T}^{\text{HRALO}}$ (s)	\mathcal{G} (%)
7-1	L1	10.14	12.91	27.36	10.91	7.57	14.97	47.62
	L2	8.06	8.41	4.34	9.42	16.87	8.41	4.34
	L3	6.21	6.56	5.68	7.17	15.39	6.83	10.03
7-2	L1	19.55	20.78	6.27	19.89	1.74	20.61	5.40
	L2	14.28	14.75	3.29	14.93	4.57	14.75	3.29
	L3	11.61	12.95	11.48	12.42	6.95	12.35	6.35
7-3	L1	21.15	21.19	0.17	23.39	10.57	23.18	9.59
	L2	18.06	18.77	3.90	18.26	1.07	18.77	3.90
	L3	12.59	14.44	14.72	14.92	18.54	14.72	16.93
7-4	L1	26.10	26.29	0.71	28.14	7.83	29.37	12.54
	L2	17.85	18.66	4.50	19.17	7.40	18.66	4.50
	L3	13.87	13.96	0.62	14.38	3.66	14.86	7.13
7-5	L1	27.86	32.32	16.03	33.48	20.18	32.57	16.93
	L2	19.33	19.59	1.33	21.29	10.14	19.59	1.33
	L3	15.35	15.79	2.88	16.89	10.05	16.66	8.58
7-6	L1	38.63	44.42	14.97	39.11	1.24	45.05	16.62
	L2	26.53	27.91	5.22	27.34	3.07	27.91	5.22
	L3	22.83	23.02	0.80	22.44	-1.72	24.22	6.08
7-7	L1	50.15	53.91	7.49	58.73	17.12	57.76	15.19
	L2	34.12	36.93	8.25	40.79	19.55	36.93	8.25
	L3	26.23	26.85	2.36	29.72	13.30	31.32	19.40
7-8	L1	71.32	73.96	3.69	72.02	0.97	75.09	5.28
	L2	48.08	51.16	6.40	51.91	7.96	51.16	6.40
	L3	39.42	40.18	1.92	40.28	2.18	42.76	8.47
7-9	L1	85.69	91.18	6.41	94.66	10.47	91.95	7.31
	L2	60.91	63.91	4.93	66.08	8.48	63.91	4.93
	L3	46.07	52.57	14.09	53.05	15.14	47.10	2.23
7-10	L1	176.99	179.94	1.67	185.07	4.56	188.01	6.23
	L2	117.93	125.79	6.66	128.07	8.59	125.79	6.66
	L3	90.78	116.23	28.04	96.73	6.56	97.54	7.44
AVG				7.21		8.67		9.47

的解，但 ISAH 的优化结果不稳定使其平均表现最差。ALWBS 作为工业优化求解方法，其内部采用了确定性的启发式优化方法，解的整体质量优于 ISAH 和 HRALO，但不如本章提出的 HHALO 优化算法。在多数情况下，HHALO 算法单次运行得到的优化结果优于其他方法，即使单次运行存在质量相对较差的解，其与其他方法最优解的差距也相对较小。

5.6.3 运行效率分析

求解效率是衡量算法性能的重要指标之一，特别是对于大规模组合优化问题，高效的求解效率有助于算法的部署推广。表5-8比较了 HHALO，ISAH 和 HRALO 算法的求解时间，由于 ALWBS 未开源具体的实现方式，因此不在比较之列。由表可知，HRALO 具有最高的求解效率，但其选用了相对简单的搜索方式，以较低的时间估计准确度为代价快速搜索。HHALO 和 ISAH 均采用了更为复杂的时间预估方式，且在搜索过程中融入了更为多样的搜索算子，能实现对解空间更充分的搜索，获得解的质量更高，搜索时间在可接受的范围内。相比较而言，本章所提出的 HHALO 比 ISAH 搜索效率更高、解决方案的质量更好。此外，HHALO 在评估候选解时调用表面组装过程优化算法造成了求解时间的增加，通过缩短单台贴片机的搜索时间或提升组装时间估计器的准确度，将有助于进一步提升算法的求解效率。

表 5-8 超启发式表面组装线优化算法与其他主流算法的求解时间比较

Table 5-8 Comparison of solving times between hyper-heuristic surface assembly line optimization algorithms and other mainstream algorithms

PCB	HHALO			HRALO			ISAH		
	L1 (s)	L2 (s)	L3 (s)	L1 (s)	L2 (s)	L3 (s)	L1 (s)	L2 (s)	L3 (s)
7-1	17.28	20.95	24.26	15.84	18.97	21.69	54.13	59.35	62.99
7-2	33.98	31.35	30.71	63.45	63.70	68.54	64.05	68.51	75.21
7-3	13.98	15.62	19.64	26.56	32.01	37.27	50.74	54.99	63.95
7-4	21.51	23.73	26.20	9.31	11.08	12.26	64.05	68.17	76.23
7-5	100.22	81.51	87.45	23.49	28.06	32.57	85.59	90.02	96.65
7-6	21.32	18.32	21.74	49.13	56.61	65.24	63.57	67.34	73.92
7-7	93.22	70.93	68.72	12.80	14.17	16.01	100.31	96.79	104.06
7-8	40.19	42.99	38.08	55.18	59.92	65.67	91.20	95.64	104.69
7-9	29.20	27.52	30.12	40.48	48.99	55.56	89.30	92.85	101.16
7-10	135.98	76.67	76.71	25.48	24.94	24.90	144.55	155.60	171.15

5.7 本章小结

本章提出了基于多特征融合集成组装时间估计器的超启发式表面组装生

产线负载平衡算法,将估计器的高准确度与超启发式的大邻域搜索能力相结合,提供了高质量的解决方案。本章分析了生产线的工作特性,设计了以进化算法为基本框架的超启发式负载平衡算法,通过多种群迭代增加搜索多样性,提出了数据驱动和目标驱动的元件分配底层启发式,并在分配过程中结合了组装优先级、可用组装工具等约束。对于元件可重复分配的情形,元件分组策略用于确定底层算子操作的基本对象,聚合聚类启发式用于确定贴装点分配的贴片机。估计器以集成学习为框架、以神经网络为基模型,在编码中融合了启发式估计的子目标等多维特征,用于估计组装时间。实验结果表明,本章提出的估计器的结果准确度更高,超启发式分配算法在处理小规模数据时与模型获得的最优解的质量接近;与其他主流研究相比,算法能在可接受的求解时间内获得装配效率更高和更为稳定的结果。

结 论

表面组装技术作为现代电子制造业的一项核心技术，用于完成高效的集成化密集电路板表面元件组装任务，推动了相关产业的发展与升级。随着技术创新与产品升级，表面组装元件正向微型化、多引脚、高集成度的方向发展，对组装过程的生产效率也提出了更高的要求，表面组装过程优化技术在其中发挥着重要作用。现有研究已对表面组装过程优化开展了广泛的探讨，分析了其中影响组装效率的主要环节，提升了表面组装过程生产效率。为解决已有研究中模型构建不完备、优化算法适用场景受限、表面组装过程效率有待提升的问题，本文基于既有的研究成果，从数学模型的建立和优化算法的设计两方面，围绕组装过程的关键环节由点及面地研究了贴片机的组装过程优化及其生产线的负载平衡优化，旨在扩宽算法的适用范围，进一步提升贴片机及其生产线的表面组装过程效率。

现将本文的主要研究内容总结如下：

1. 通过提取影响组装效率的关键性能指标，建立了基于关键子目标分解的贴片头任务分配模型；进一步地，提出了带约束的多拾贴周期表面组装过程路径规划模型；通过结合组装生产线的约束条件，以关键性能指标近似组装效率，提出了最小-最大表面组装生产线负载平衡模型。相较于以往的研究，本文提出的数学规划模型充分考虑了实际组装条件，对各环节模型进行了深入讨论，给出了模型的实例说明、参数灵敏度和模型复杂度分析。分阶段的数学模型可用于小规模数据的优化，从而获得组装过程优化问题的近似最优解；

2. 提出了周期任务集成的增强数学规划模型，以基于递归的启发式确定模型的初始解，运用限制决策变量规模和压缩模型可行解域的策略，加快了周期任务集成模型求解的速度；研究了模型的解集池的选择标准，以预先评估不同解的组装过程的移动路径指标，降低了分解子问题之间的耦合性，实现了对加权关键子目标的优化。为提升大规模数据的优化求解效率，本文提出了分层前瞻启发式搜索框架，以吸嘴分配模式的构造为基础，研究了启发式的供料器安装策略，提出了基于多种评价准则和长短期目标收益的贴片头元件分配启发式算法，优化了组装任务中的主要子目标，同时结合了实际应用场景的需求进行了扩展；

3. 提出了结合组装过程多轴作业特性的动态规划算法，保证拾贴周期规划

解的最优性。动态规划可作为周期内的路径规划方法，基于此，本文设计了周期内的贴装点分配方法。在贴片头任务约束下，本文研究了多源贪心算法并融合了动态导向集束搜索策略保留非贪心解，有效提高了路径规划解的质量；提出了基于聚合路径重构启发式的自适应大邻域搜索算法，用于克服贪心搜索解的拾贴周期路径贴装点分配不均衡的局限性，实现了对组装过程贴装路径持续在线优化；

4. 设计了基于数据驱动和目标驱动的底层启发式用以快速搜索高质量的负载平衡解，提出了元件可重复分配策略用以分配同类型元件在不同贴片机的组装任务，研究了组装优先级和可用组装过程对元件分配过程的限制，进一步平衡了贴片机的工作负载。本文基于集成学习构建了组装时间估计器，通过全连接神经网络的基模型和融合子目标等的多维特征编码，提升了估计器的准确度。高准确度的估计器与有较强邻域搜索能力的超启发式算法结合，使其在可接受时间内获得具有更高装配效率和更稳定输出的解。

结合上述分析，现将本论文的主要创新点归纳如下：

1. 分析了表面组装过程的优化目标和约束条件，对复杂组装任务进行分阶段解耦，建立了完备的表面组装过程混合整数线性规划模型，分析了模型优化目标参数灵敏度和复杂度，实现了解的可行性验证和解的质量的定量分析；

2. 针对贴片头的任务分配问题，提出了周期任务集成增强模型和分层前瞻扫描启发式算法，用于解决不同规模的任务分配问题，算法具备全面搜索和高效优化的能力，在提升适用范围的基础上，保证了组装过程关键子目标的高质量；

3. 针对贴装过程的路径规划问题，设计了基于动态规划的周期内和动态导向集束搜索的周期内路径规划算法，研究了基于自适应大邻域搜索的在线迭代路径优化算法，实现了多轴路径规划高质量解的构造和对解的持续迭代优化；

4. 针对组装生产线的负载平衡问题，研究了基于多特征融合集成时间估计的超启发式优化方法用于表面组装生产线负载平衡，构建了数据驱动和目标驱动的启发式算子，实现了多台贴片机不同生产配置下负载任务的平衡，提升了整条生产线的组装效率。

综上所述，本文针对表面组装过程优化问题，对问题的各阶段模型的建立、优化算法的设计等方面开展了全面而深入的研究。所研究的模型在完备性和可解性均有所改进，所提出的启发式算法具有运算速度快、解的质量高、适用范围广等特点，能直接部署到表面组装设备及其生产线，取得了实质性的创新性成果。然而，尽管本文的研究取得了一定的进展，但其在处理具有复杂解空间

的表面组装过程优化问题时，在求解效率和解的质量方面仍有提升空间。未来的研究工作将从以下几个方面展开：

1. 改进表面组装过程数学模型：本文提出的模型仅适用于中小规模数据，且所得解仅为关键指标加权拟合的近似最优解，不同角度的建模策略有望得到深入研究，模型的求解效率仍待改进提升；

2. 完善贴装过程路径规划算法：本文对贪心式路径规划算法做了改进，下一步工作中将优化路径规划搜索策略，利用组装路径规划问题的结构特征生成更高质量的解，并设计更优的局部搜索策略提升组装过程效率；

3. 提升组装时间估计器的准确度：本文将组装时间估计器作为生产线负载均衡中分配结果质量评估的依据，后续研究可通过减少组装估计器的预测误差、融合贴装点的分布特征等，提升表面组装生产线的效率。

参考文献

- [1] 中共中央关于制定国民经济和社会发展第十四个五年规划和二〇三五年远景目标的建议[J]. 共产党员, 2020(11): 12-23.
- [2] 臧冀原, 刘宇飞, 王柏村, 等. 面向 2035 的智能制造技术预见和路线图研究[J]. 机械工程学报, 2022, 58(4): 285-308.
- [3] 王文泽. 以智能制造作为新质生产力支撑引领现代化产业体系建设[J]. 当代经济研究, 2024(2): 105-115.
- [4] SCHLENKRICH M, PARRAGH S N. Solving large scale industrial production scheduling problems with complex constraints: an overview of the state-of-the-art [J]. Procedia Computer Science, 2023, 217: 1028-1037.
- [5] HO W, JI P. An integrated scheduling problem of PCB components on sequential pick-and-place machines: Mathematical models and heuristic solutions[J]. Expert Systems with Applications, 2009, 36(3): 7002-7010.
- [6] ALTINKEMER K, KAZAZ B, KÖKSALAN M, et al. Optimization of printed circuit board manufacturing: Integrated modeling and algorithms[J]. European Journal of Operational Research, 2000, 124(2): 409-421.
- [7] ALKAYA A F, DUMAN E. Combining and solving sequence dependent traveling salesman and quadratic assignment problems in PCB assembly[J]. Discrete Applied Mathematics, 2015, 192(C): 2-16.
- [8] LUO J, LIU J. An MILP model and clustering heuristics for LED assembly optimisation on high-speed hybrid pick-and-place machines[J]. International Journal of Production Research, 2013, 52(4): 1016-1031.
- [9] LUO J, ZHANG X, LIU H. Modelling the operation optimization of multi-head surface mounting machines with over-head gantry in printed circuit board assembly [C]//Proceedings of the 33rd Chinese Control Conference. Nanjing, China, 2014: 7537-7542.
- [10] LUO J, LIU J, HU Y. An MILP model and a hybrid evolutionary algorithm for integrated operation optimisation of multi-head surface mounting machines in PCB assembly[J]. International Journal of Production Research, 2017, 55(1): 145-160.

- [11] ASHAYERI J, SELEN W. A planning and scheduling model for onsertion in printed circuit board assembly[J]. *European Journal of Operational Research*, 2007, 183(2): 909-925.
- [12] HSU H P. Solving the feeder assignment, component sequencing, and nozzle assignment problems for a multi-head gantry SMT machine using improved firefly algorithm and dynamic programming[J]. *Advanced Engineering Informatics*, 2022, 52: Article 101583.
- [13] RADULY-BAKA C, KNUUTILA T, JOHNSSON M, et al. Selecting the nozzle assortment for a gantry-type placement machine[J]. *OR Spectrum*, 2008, 30(3): 493-513.
- [14] RADULY-BAKA C, JOHNSSON M, NEVALAINEN O S. Tool-feeder partitions for module assignment in PCB assembly[J]. *Computers & Operations Research*, 2017, 78: 108-116.
- [15] SUN D S, LEE T E. A branch-and-price algorithm for placement routing for a multi-head beam-type component placement tool[J]. *OR Spectrum*, 2008, 30(3): 515-534.
- [16] TORABI S, HAMED M, ASHAYERI J. A new optimization approach for nozzle selection and component allocation in multi-head beam-type SMD placement machines[J]. *Journal of Manufacturing Systems*, 2013, 32(4): 700-714.
- [17] ASHAYERI J, MA N, SOTIROV R. An aggregated optimization model for multi-head SMD placements[J]. *Computers & Industrial Engineering*, 2011, 60(1): 99-105.
- [18] GUO S, GENG F, TAKAHASHI K, et al. A MCVRP-based model for PCB assembly optimisation on the beam-type placement machine[J]. *International Journal of Production Research*, 2018, 57(18): 5874-5891.
- [19] SUN D S, LEE T E, KIM K H. Component allocation and feeder arrangement for a dual-gantry multi-head surface mounting placement tool[J]. *International Journal of Production Economics*, 2005, 95(2): 245-264.
- [20] 邢星, 柴茜, 贾志淳, 等. 一种高速贴片机喂料器快速分配方法[J]. *渤海大学学报 (自然科学版)*, 2018, 39(1): 70-77.
- [21] HARDAS C, DOOLEN T, JENSEN D. Development of a genetic algorithm for component placement sequence optimization in printed circuit board assembly[J]. *Computers & Industrial Engineering*, 2008, 55(1): 165-182.

- [22] 彭乾伟, 曹乐, 万婷婷, 等. 拱架式贴片机贴装路径优化[J]. 重庆大学学报, 2021, 45(9): 51-60.
- [23] KULAK O, YILMAZ I O, GÜNTHER H O. PCB assembly scheduling for collect-and-place machines using genetic algorithms[J]. International Journal of Production Research, 2007, 45(17): 3949-3969.
- [24] GYORFI J, WU C H. An efficient algorithm for placement sequence and feeder assignment problems with multiple placement-nozzles and independent link evaluation[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2008, 38(2): 437-442.
- [25] 吴忻生, 黄志龙, 周炯, 等. 基于改进混合遗传算法的贴片机贴装路径优化[J]. 制造业自动化, 2014, 36(9): 85-88+103.
- [26] 武洪恩, 王凯, 倪良月. 最小准则遗传算法求解贴片机贴装调度问题[J]. 现代制造工程, 2020(3): 14-20.
- [27] 杜轩, 李登桥, 张屹. 基于混合遗传算法的多头动臂式贴片机贴装过程优化研究[J]. 机床与液压, 2014, 42(1): 40-47.
- [28] 殷旅江, 高亮, 李登桥, 等. 改进元胞遗传算法的转塔式贴片机贴装优化[J]. 华中科技大学学报(自然科学版), 2015, 43(3): 113-117+132.
- [29] LI Z, YU X, QIU J, et al. Cell division genetic algorithm for component allocation optimization in multi-functional placers[J]. IEEE Transactions on Industrial Informatics, 2022, 18(1): 559-570.
- [30] TSUCHIYA T, YAMASHITA A, KANEKO T, et al. Scheduling optimization of component mounting in printed circuit board assembly by prioritizing simultaneous pickup[C]//IEEE/RSJ International Conference on Intelligent Robots and Systems. San Diego, CA, USA, 2007: 2913-2918.
- [31] ZHU G Y, ZHANG W B. An improved shuffled frog-leaping algorithm to optimize component pick-and-place sequencing optimization problem[J]. Expert Systems with Applications, 2014, 41(15): 6818-6829.
- [32] 罗家祥, 罗树浩, 吴忻生. 基于 RLS 的用于贴片机贴装顺序优化的禁忌搜索算法[J]. 华南理工大学学报(自然科学版), 2012, 40(3): 74-80.
- [33] LIU H, YUAN P, LUO J, et al. An optimization algorithm of PCB assembly time for multi-head placement machine[C]//36th Chinese Control Conference (CCC). Dalian, China, 2017: 2876-2881.

- [34] 许继影, 陈仕军. 基于免疫算法的拱架式贴片机贴装过程优化[J]. 现代电子技术, 2017, 40(7): 100-104+108.
- [35] CASTELLANI M, OTRI S, PHAM D T. Printed circuit board assembly time minimisation using a novel bees algorithm[J]. Computers & Industrial Engineering, 2019, 133(C): 186-194.
- [36] HSU H P. Printed circuit board assembly planning for multi-head gantry SMT machine using multi-swarm and discrete firefly algorithm[J]. IEEE Access, 2021, 9: 1642-1654.
- [37] WANG Z, MUMTAZ J, ZHANG L, et al. Application of an improved spider monkey optimization algorithm for component assignment problem in PCB assembly [J]. Procedia CIRP, 2019, 83: 266-271.
- [38] 付永忠, 潘云峰. 基于改进蚁群算法的垂直旋转式贴片机贴装顺序优化[J]. 计算机与现代化, 2017(8): 17-21.
- [39] 余松乐, 肖曙红, 林德育. 双臂式双旋转头贴片机贴装效率优化研究[J]. 计算机应用与软件, 2019, 36(9): 146-150.
- [40] HSU H P. Solving feeder assignment and component sequencing problems for printed circuit board assembly using particle swarm optimization[J]. IEEE Transactions on Automation Science and Engineering, 2017, 14(2): 881-893.
- [41] HSU H P, YANG S W. Optimization of component sequencing and feeder assignment for a chip shooter machine using shuffled frog-leaping algorithm[J]. IEEE Transactions on Automation Science and Engineering, 2020, 17(1): 56-71.
- [42] GRIFFITHS C A, GIANNETTI C, ANDRZEJEWSKI K T, et al. Comparison of a bat and genetic algorithm generated sequence against lead through programming when assembling a PCB using a six-axis robot with multiple motions and speeds [J]. IEEE Transactions on Industrial Informatics, 2022, 18(2): 1102-1110.
- [43] 陆瞳瞳, 余珠玛, 郑小东. 基于元胞蝙蝠算法的复合式贴片机贴装调度优化[J]. 现代制造工程, 2017(9): 22-28+101.
- [44] MENG Z, YILDIZ B S, LI G, et al. Application of state-of-the-art multiobjective metaheuristic algorithms in reliability-based design optimization: a comparative study[J]. Structural and Multidisciplinary Optimization, 2023, 66(191): 1-27.
- [45] ZHU G Y, JU X, ZHANG W B. Multi-objective sequence optimization of PCB component assembly with GA based on the discrete fréchet distance[J]. International Journal of Production Research, 2018, 56(11): 4017-4024.

- [46] 路军营, 朱光宇. 基于灰熵关联分析的表面贴装多目标优化[J]. 计算机集成制造系统, 2013, 19(4): 766-773.
- [47] GAO H, LI Z, YU X, et al. Hierarchical multiobjective heuristic for PCB assembly optimization in a beam-head surface mounter[J]. IEEE Transactions on Cybernetics, 2021, 52(7): 6911-6924.
- [48] KNUUTILA T, PYÖTTIÄLÄ S, NEVALAINEN O S. Minimizing the number of pickups on a multi-head placement machine[J]. Journal of the Operational Research Society, 2007, 58(1): 115-121.
- [49] KNUUTILA T, SUOMI T, EMET S, et al. Organizing the nozzle magazine of a gantry-type PCB assembly machine[J]. International Journal of Advanced Manufacturing Technology, 2013, 68(5): 1189-1202.
- [50] PYÖTTIÄLÄ S, KNUUTILA T, JOHNSON M, et al. Minimizing the assembly cycle time on a revolver gantry machine[J]. Computers & Operations Research, 2013, 40(11): 2611-2624.
- [51] HAN J, SEO Y. Mechanism to minimise the assembly time with feeder assignment for a multi-headed gantry and high-speed SMT machine[J]. International Journal of Production Research, 2017, 55(10): 2930-2949.
- [52] 陈铁梅, 罗家祥, 胡跃明. 基于蚁群-混合蛙跳算法的贴片机贴装顺序优化[J]. 控制理论与应用, 2011, 28(12): 1813-1820.
- [53] LI Z, SUN H, YU X, et al. Heuristic sequencing hopfield neural network for pick-and-place location routing in multi-functional placers[J]. Neurocomputing, 2022, 472(1): 35-44.
- [54] WANG C N, NHIEU N L, VIET T A P. Enhancing efficiency in PCB assembly for the leading global electronics manufacturing services firm: a TRIZ and Ant Colony Optimization approach[J]. International Journal of Advanced Manufacturing Technology, 2024, 133: 5529-5552.
- [55] SIOWIK A, CPAŁKA K. Hybrid approaches to nature-inspired population-based intelligent optimization for industrial applications[J]. IEEE Transactions on Industrial Informatics, 2022, 18(1): 546-558.
- [56] GRUNOW M, GÜNTHER H O, SCHLEUSENER M, et al. Operations planning for collect-and-place machines in PCB assembly[J]. Computers & Industrial Engineering, 2004, 47(4): 409-429.

- [57] PARK T H, KIM N. A dynamic programming approach to PCB assembly optimization for surface mounters[J]. International Journal of Control Automation and Systems, 2007, 5(2): 192-199.
- [58] CHEN T, LUO J, HU Y. Component placement process optimization for multi-head surface mounting machine based on tabu search and improved shuffled frog-leaping algorithm[C]//3rd International Workshop on Intelligent Systems and Applications. Wuhan, China, 2011: 1-4.
- [59] GAO J, ZHU X, LIU A, et al. An iterated hybrid local search algorithm for pick-and-place sequence optimization[J]. Symmetry, 2018, 10(11): 633-649.
- [60] HUANG Y, ZHAO L, LIU P. Applied research of hierarchical multi-objective optimization method in high speed and high precision placement machine[J]. Journal of Physics: Conference Series, 2020, 1605(1): Article 012029.
- [61] LEE S H, PARK T H, LEE B H, et al. A dynamic programming approach to a reel assignment problem of a surface mounting machine in printed circuit board assembly[C]//Proceedings of the IEEE International Conference on Robotics & Automation. Leuven, Belgium, 1998: 227-232.
- [62] LEE S, LEE B, PARK T. A hierarchical method to improve the productivity of a multi-head surface mounting machine[C]//Proceedings of IEEE International Conference on Robotics and Automation. Detroit, MI, USA, 1999: 2110-2115.
- [63] 邢星, 贾志淳, 孙乙铭, 等. 一种高速贴片机在线贴装优化方法研究[J]. 计算机工程与应用, 2017, 53(9): 220-225+252.
- [64] GUO S, TAKAHASHI K, MORIKAWA K. PCB assembly scheduling with alternative nozzle types for one component type[J]. Flexible Services and Manufacturing Journal, 2011, 23(3): 316-345.
- [65] LIN C J, MEI-LING H. Modified artificial bee colony algorithm for scheduling optimization for printed circuit board production[J]. Journal of Manufacturing Systems, 2017, 44(1): 1-11.
- [66] LIN C J, LIN C H. Component placement process optimization for multi-head surface mounting machine using a hybrid algorithm[J]. Sensors and Materials, 2021, 33(10): 3529-3545.
- [67] LIN C J, HUANG M L. Efficient hybrid group search optimizer for assembling printed circuit boards[J]. Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 2019, 33(3): 259-274.

-
- [68] LI D, YOON S W. PCB assembly optimization in a single gantry high-speed rotary-head collect-and-place machine[J]. International Journal of Advanced Manufacturing Technology, 2017: 2819-2834.
- [69] LI D, HE T, YOON S W. Clustering-based heuristic to optimize nozzle and feeder assignments for collect-and-place assembly[J]. IEEE Transactions on Automation Science and Engineering, 2019, 16(2): 755-766.
- [70] HE T, LI D, YOON S W. A multi-phase planning heuristic for a dual-delivery SMT placement machine optimization[J]. Robotics and Computer-Integrated Manufacturing, 2017, 47: 85-94.
- [71] HE T, LI D, YOON S W. An adaptive clustering-based genetic algorithm for the dual-gantry pick-and-place machine optimization[J]. Advanced Engineering Informatics, 2018, 37: 66-78.
- [72] TÓTH A, KNUUTILA T, NEVALAINEN O S. Machine configuration and workload balancing of modular placement machines in multi-product PCB assembly[J]. International Journal of Computer Integrated Manufacturing, 2018, 31(9): 815-830.
- [73] KODEK D M, KRISPER M. Optimal algorithm for minimizing production cycle time of a printed circuit board assembly line[J]. International Journal of Production Research, 2004, 42(23): 5031-5048.
- [74] EMET S, KNUUTILA T, ALHONIEMI E, et al. Workload balancing in printed circuit board assembly[J]. International Journal of Advanced Manufacturing Technology, 2010, 50(9): 1175-1182.
- [75] RONG A, TOTH A, NEVALAINEN O S, et al. Modeling the machine configuration and line-balancing problem of a PCB assembly line with modular placement machines[J]. International Journal of Advanced Manufacturing Technology, 2011, 54(1): 349-360.
- [76] 靳志宏. 印刷电路板组装生产线调度优化问题建模[J]. 中国管理科学, 2008, 16(S1): 122-127.
- [77] KULAK O, YILMAZ I O, GÜNTHER H O. A GA-based solution approach for balancing printed circuit board assembly lines[J]. OR Spectrum, 2007, 30(3): 469-491.
- [78] WU Y, JI P. A solution method for the component allocation problem in printed circuit board assembly[J]. Assembly Automation, 2010, 30(2): 155-163.

- [79] GUO S, TAKAHASHI K, MORIKAWA K, et al. An integrated allocation method for the PCB assembly line balancing problem with nozzle changes[J]. *International Journal of Advanced Manufacturing Technology*, 2012, 62(1): 351-369.
- [80] 刘海明, 袁鹏, 罗家祥, 等. 表面组装生产线的负荷平衡优化算法研究[C]//第三十二届中国控制会议 (B 卷). 西安, 中国, 2013: 945-950.
- [81] MUMTAZ J, GUAN Z, YUE L, et al. Hybrid spider monkey optimisation algorithm for multi-level planning and scheduling problems of assembly lines[J]. *International Journal of Production Research*, 2020, 58(20): 6252-6267.
- [82] ZHONG J, CHEN Y, MUMTAZ J. A multi-objective scheduling optimization method for PCB assembly lines based on the improved spider monkey algorithm [C]//International Conference on Advances in Mechanical Engineering. MDPI, 2022: Article 15.
- [83] CHEN Y, ZHONG J, MUMTAZ J, et al. An improved spider monkey optimization algorithm for multi-objective planning and scheduling problems of PCB assembly line[J]. *Expert Systems with Applications*, 2023, 229: 120600.
- [84] YAN X, ZUO H, HU C, et al. Load optimization scheduling of chip mounter based on hybrid adaptive optimization algorithm[J]. *Complex System Modeling and Simulation*, 2023, 3(1): 1-11.
- [85] TÓTH A, KNUUTILA T, NEVALAINEN O S. Reconfiguring flexible machine modules of a PCB assembly line[J]. *Production Engineering*, 2010, 4(1): 85-94.
- [86] HE T, LI D, YOON S W. A heuristic algorithm to balance workloads of high-speed SMT machines in a PCB assembly line[J]. *Procedia Manufacturing*, 2017, 11: 1790-1797.
- [87] CHOUDHURY N D, WILHELM W E, RAO B, et al. Process planning for circuit card assembly on a series of dual head placement machines[J]. *European Journal of Operational Research*, 2007, 182(2): 626-639.
- [88] LIN W L, TARDIF V. Component partitioning under demand and capacity uncertainty in printed circuit board assembly[J]. *International Journal of Flexible Manufacturing Systems*, 1999, 11: 159-176.
- [89] HU K J. Fuzzy goal programming technique for solving flexible assignment problem in PCB assembly line[J]. *Journal of Information and Optimization Sciences*, 2017, 38(3-4): 423-442.

- [90] ZHANG Z, TANG Q, CHICA M, et al. Reinforcement learning-based multiobjective evolutionary algorithm for mixed-model multimanned assembly line balancing under uncertain demand[J]. *IEEE Transactions on Cybernetics*, 2024, 54(5): 2914-2927.
- [91] WANG K, LI X, GAO L, et al. A discrete artificial bee colony algorithm for multi-objective disassembly line balancing of end-of-life products[J]. *IEEE Transactions on Cybernetics*, 2022, 52(8): 7415-7426.
- [92] ZHANG W B, ZHU G Y. A multiobjective optimization of PCB prototyping assembly with OFA based on the similarity of intuitionistic fuzzy sets[J]. *IEEE Transactions on Fuzzy Systems*, 2021, 29(7): 2054-2061.
- [93] 靳志宏, 陈贞, 李向军, 等. 考虑切换时间的多品种印刷电路板表面贴装生产线调度优化[J]. *控制与决策*, 2011, 26(1): 10-16.
- [94] 刘颖, 靳志宏. 多品种小批量生产环境下表面贴装生产线的平衡优化[J]. *大连海事大学学报*, 2012, 38(2): 87-90.
- [95] 郭姝娟, 张晓庆, 靳志宏. 考虑吸嘴切换的多品种印刷电路板组装优化[J]. *大连海事大学学报*, 2016, 42(4): 125-132.
- [96] TÓTH A, KNUUTILA T, NEVALAINEN O S. A divide and combine method for machine configuration and workload balancing problem in multiple product PCB assembly[J]. *International Journal of Advanced Manufacturing Technology*, 2022, 120: 4079-4095.
- [97] MUMTAZ J, GUAN Z, YUE L, et al. Multi-level planning and scheduling for parallel PCB assembly lines using hybrid spider monkey optimization approach[J]. *IEEE Access*, 2019, 7: 18685-18700.
- [98] KOSKINEN J, RADULY-BAKA C, JOHNSON M, et al. Rolling horizon production scheduling of multi-model PCBs for several assembly lines[J]. *International Journal of Production Research*, 2019, 58(4): 1052-1073.
- [99] PAN Z, LEI D, WANG L. A knowledge-based two-population optimization algorithm for distributed energy-efficient parallel machines scheduling[J]. *IEEE Transactions on Cybernetics*, 2022, 52(6): 5051-5063.
- [100] CHEN T L, CHEN J C, CHEN Y Y, et al. The optimal configuration for various placement machines in PCB assembly lines[J/OL]. *Annals of Operations Research*, 2024. DOI: 10.1007/s10479-024-05828-6.

- [101] VAINIO F, PAHIKKALA T, JOHNSON M, et al. Estimating the production time of a PCB assembly job without solving the optimised machine control[J]. *International Journal of Computer Integrated Manufacturing*, 2015, 28(8): 823-835.
- [102] VAINIO F, MAIER M, KNUUTILA T, et al. Estimating printed circuit board assembly times using neural networks[J]. *International Journal of Production Research*, 2010, 48(8): 2201-2218.
- [103] LI D, CHEN S, CHIONG R, et al. Predicting the printed circuit board cycle time of surface-mount-technology production lines using a symbiotic organism search-based support vector regression ensemble[J]. *International Journal of Production Research*, 2020, 59(23): 7246-7265.
- [104] CHEN S, LI D, GAN X, et al. An optimised light gradient boosting machine model for setup time prediction in electronic production lines[C]//*Proceedings of the Genetic and Evolutionary Computation Conference Companion*. New York, NY, USA, 2024: 647-650.
- [105] JI P, SZE M, LEE W. A genetic algorithm of determining cycle time for printed circuit board assembly lines[J]. *European Journal of Operational Research*, 2001, 128(1): 175-184.
- [106] GUO S, TAKAHASHI K, MORIKAWA K. Nozzle assignment in PCB assembly considering one component type picked up by multiple nozzle types[C]//*9th Asia Pacific Industrial Engineering and Management Society Conference*. Kitakyushu, Japan, 2009: 2773-2781.
- [107] GUROBI OPTIMIZATION L. Gurobi optimizer reference manual[EB/OL]. 2022. "https://www.gurobi.com".
- [108] DUAN S, JIANG S, DAI H, et al. The applications of hybrid approach combining exact method and evolutionary algorithm in combinatorial optimization[J]. *Journal of Computational Design and Engineering*, 2023, 10(3): 934-946.
- [109] MIRHOSSEINI M, FAZLALI M, FALLAH M K, et al. A fast MILP solver for high-level synthesis based on heuristic model reduction and enhanced branch and bound algorithm[J]. *Journal of Supercomputing*, 2023, 79(11): 12042-12073.
- [110] KAKKAD D A, GROSSMANN I E, SPRINGUB B, et al. Iterative MILP algorithm to find alternate solutions in linear programming models[J/OL]. *Optimization and Engineering*, 2024. DOI: 10.1007/s11081-024-09887-3.

- [111] CAI J, ZHU Q, LIN Q. Variable neighborhood search for a new practical dynamic pickup and delivery problem[J]. Swarm and Evolutionary Computation, 2022, 75: Article 101182.
- [112] ACCORSI L, VIGO D. A fast and scalable heuristic for the solution of large-scale capacitated vehicle routing problems[J]. Transportation Science, 2021, 55(4): 832-856.
- [113] DOKEROGLU T, KUCUKYILMAZ T, TALBI E G. Hyper-heuristics: A survey and taxonomy[J]. Computers & Industrial Engineering, 2024, 187: Article 109815.
- [114] BOYSEN N, SCHULZE P, SCHOLL A. Assembly line balancing: What happened in the last fifteen years?[J]. European Journal of Operational Research, 2022, 301(3): 797-814.

攻读博士学位期间发表的论文及其他成果

（一）发表的学术论文

- [1] LU G, LI Z, SUN H, YU X, QIN J, QIU J and GAO H. A Two-Phase PCBA Optimization with ILP Model and Heuristic for a Beam Head Placement Machine[J]. IEEE Transactions on Industrial Informatics, 2024, 20(12):13612-13621. (WOS: 001292793500001, IF: 11.7, 对应本文第三章和第四章)
- [2] LU G, YU X, SUN H, LI Z, QIU, J and GAO H. A Scan-Based Hierarchical Heuristic Optimization Algorithm for PCB Assembly Process[J]. IEEE Transactions on Industrial Informatics. 2024, 20(3): 3609-3618. (WOS: 001077226700001, IF: 11.7, 对应本文第二章、第三章和第四章)
- [3] LU G, GAO H, LI Z, YU X, WANG T, QIU J and RODRÍGUEZ-ANDINA J J. Hyper-Heuristic Optimization Using Multifeature Fusion Estimator for PCB Assembly Lines with Linear-Aligned-Heads Surface Mounters[J]. IEEE Transactions on Cybernetics. DOI:10.1109/TCYB.2025.3556512. (WOS: 001480230000001, IF: 9.4, 对应本文第二章和第五章)

（二）授权的发明专利

- [1] 高会军, 卢光宇, 杨宪强, 王大钊, 刘金泽. 基于动态规划的直排式贴片机贴装头元件分配方法: ZL202211392055.2 [P], 2023-07-28.
- [2] 高会军, 卢光宇, 杨宪强. 一种基于最大二分配的贴片机吸嘴分配方法: ZL202211392089.1 [P], 2023-05-09.
- [3] 高会军, 李政锴, 卢光宇, 邱剑彬, 于兴虎. 一种基于最近插入法的贴片机贴装路径规划方法: ZL202010387811.7 [P], 2021-09-24.
- [4] 高会军, 李政锴, 卢光宇, 邱剑彬, 于兴虎. 一种基于禁忌搜索算法的 LED 贴片机拾贴路径优化方法: ZL202010388771.8 [P], 2021-06-01.
- [5] 高会军, 李政锴, 卢光宇, 邱剑彬, 于兴虎. 一种基于聚类的多功能贴片机贴装路径规划方法: ZL202010387784.3 [P], 2021-05-07.

（三）参与的科研项目

- [1] 精密电子表面焊装全工序调控理论与方法, 国家重点研发计划“高性能制造技术与重大装备”重点专项, 2024YFB3409200, 2024 年 12 月-2027 年 11 月.

- [2] 表面贴装高端智能装备基于视觉的高速高精度运动控制理论与方法，国家自然科学基金重点支持项目（联合）项目，U20A20188，2020 年 1 月–2024 年 12 月.

学位论文评阅人、答辩委员会名单及答辩决议

评阅人 (根据实际 人数填写)		姓名	职称 (是否博导)	工作单位	所在学科
答辩委员会成员	职务	姓名	职称 (是否博导)	工作单位	所在学科
	主席	曾庆双	教授/是	哈尔滨工业大学	控制科学与工程
	委员	高会军	教授/是	哈尔滨工业大学	控制科学与工程
		邱剑彬	教授/是	哈尔滨工业大学	控制科学与工程
		杨学博	教授/是	哈尔滨工业大学	控制科学与工程
		周 彬	教授/是	哈尔滨工业大学	控制科学与工程
		钟伟民	教授/是	华东理工大学	控制科学与工程
		王雅琳	教授/是	中南大学	控制科学与工程
	秘书	郑晓龙	副教授/是	哈尔滨工业大学	控制科学与工程
<p>答辩委员会决议 (对论文的评语及是否建议授予博士学位等):</p> <p>哈尔滨工业大学控制科学与工程学科博士研究生卢光宇所完成的题为“表面组装过程中生产效率提升的关键优化方法研究”的学位论文, 选题正确, 具有重要的理论意义和实用价值。</p> <p>作者详细地归纳和评述了大量的有关文献, 掌握了该领域国内外的研究现状和发展方向。对贴片表面组装过程优化问题开展了系统深入的研究, 在模型构建、算法设计及工程应用验证方面取得了如下创新性成果:</p> <ol style="list-style-type: none"> 1. 建立了完备的表面组装过程混合整数线性规划模型, 分析了模型优化目标参数灵敏度, 实现了复杂组装过程解的可行性验证与质量定量分析。 2. 提出了周期集成任务模型与分层前瞻启发式算法, 以及基于动态规划和导向集束搜索的路径规划算法, 实现了贴装过程任务分配与路径规划的高效协同优化。 3. 提出了超启发式元件分配优化方法, 构建了数据驱动和目标驱动的启发式算子, 实现了多台贴片不同生产配置下负载平衡, 提升了组装效率。 <p>学位论文结构合理, 叙述清晰, 分析全面, 结论正确。答辩中作者正确地回答了委员们提出的问题。答辩委员会认为, 达到了《中华人民共和国学位法》对学位论文的要求, 说明作者具有坚实全面的基础理论和系统深入的专门知识, 具备独立从事科学研究工作的能力。答辩委员会7人投票, 7票赞成、0票弃权、0票反对。根据投票结果, 答辩委员会一致同意卢光宇通过博士学位论文答辩, 同意其毕业, 并建议授予控制科学与工程学科工学博士学位。</p>					

哈尔滨工业大学学位论文原创性声明和使用权限

学位论文原创性声明

本人郑重声明：此处所提交的学位论文《表面组装过程中生产效率提升的关键优化方法研究》，是本人在导师指导下，在哈尔滨工业大学攻读学位期间独立进行研究工作所取得的成果，且学位论文中除已标注引用文献的部分外不包含他人完成或已发表的研究成果。对本学位论文的研究工作做出重要贡献的个人和集体，均已在文中以明确方式注明。

作者签名：卢光宇

日期：2025年4月14日

学位论文使用权限

学位论文是研究生在哈尔滨工业大学攻读学位期间完成的成果，知识产权归属哈尔滨工业大学。学位论文的使用权限如下：

(1) 学校可以采用影印、缩印或其他复制手段保存研究生上交的学位论文，并向国家图书馆报送学位论文；(2) 学校可以将学位论文部分或全部内容编入有关数据库进行检索和提供相应阅览服务；(3) 研究生毕业后发表与此学位论文研究成果相关的学术论文和其他成果时，应征得导师同意，且第一署名单位为哈尔滨工业大学。

保密论文在保密期内遵守有关保密规定，解密后适用于此使用权限规定。本人知悉学位论文的使用权限，并将遵守有关规定。

作者签名：卢光宇

日期：2025年4月14日

导师签名：刘永

日期：2025年4月14日

致 谢

二十余载的求学之路，至今日暂告一段落。书山有路，学海无涯，我以这篇博士学位论文，为我的求学生涯标上一个阶段性的句点，同时也为新的征程拉开序幕。谨以此文，向一路走来给予我支持与帮助的亲人、师长、朋友表达深深的谢意。

首先，衷心感谢我的导师高会军教授。高老师作为团队的主心骨和领路人，带领团队在光机电一体化、智能装备研发领域深耕十余年，得益于老师搭建的高端研发平台和前瞻性的课题规划，我在其中将产学研紧密结合，从工程实践中提炼科学问题，将所学知识运用于解决实际工程问题中，真正把论文写在祖国的大地上。高老师对待科研项目严谨的态度、勤勉不懈的精神，始终是我学习的榜样。

感谢邱剑彬教授，邱老师在我课题研究过程中多次为我指点迷津、答疑解惑，在学术论文和博士学位论文的撰写过程中，邱老师细致耐心地为修改论文，提出宝贵建议，指导我回复审稿人的意见，使我有信心一步步深入课题的研究工作。感谢 Juan J. Rodríguez-Andina 教授，Juanjo 教授对我的学术论文进行了大量细致入微的修改，对我论文写作水平的提升起到很大助益。感谢杨宪强教授，杨老师总是身体力行地参与到项目之中，在项目的立项和开展过程中提供了很多指导，在忙碌之余关心和鼓励我的课题进展，令我深受感动。智能控制与系统研究所的杨学博教授、孙维超教授、王桐教授、潘惠惠教授、于金泳教授、林伟阳教授等也在平时的学习生活中给予了我诸多关心和帮助，在此一并致谢。

感谢李政锴师兄，作为课题组的大师兄，师兄在课题前期做了大量基础性工作，并给予我悉心指导，支持和帮助我开展课题的相关研究。感谢刘志太师兄，师兄从研究生入学起便给予我诸多帮助，在我科研、生活、工作中遇到困难时，总是为我答疑解惑、出谋划策，于我而言既是良师也是益友。感谢陈琦师兄，我们在学习和生活中常常交流探讨，在数年的硕博求学过程中他常常为我排忧解难，宿舍内的深夜长谈更是为我的读博生活增添了许多乐趣。感谢任昕昉师兄，即使是对初来乍到的小师弟，师兄也总是愿倾囊相助，在我项目研究中做那个可靠又热心的“智囊”，为我解决工程难题提供了重要的助力。感谢皮棋棋和曹峻虎师弟，同为课题组早期成员，我们相识最久，交流最频繁，时

常讨论论文的想法和进展，感谢他们在学习、工作和生活上的支持和帮助。感谢优化课题组的尹宝青、毕宇航、高磊、王跃翔、杨利龙等师弟，在相互学习和交流中，我们一起参与项目的研发和课题的研究，很高兴能与他们相遇。此外，还要感谢同门师兄弟陈兆楠、刘欢锂、张忠金、李苑、龚见素、余朝海、贾志祥、韩春刚、刘译彭、张元明等同学，我们在读博期间相互沟通、共同进步，让我感受到了集体的力量和温暖。

感谢宁波智能装备研究院有限公司，研究院立足于智能装备研发、培养了一批批硕博研究生，取得了一系列突出的成果。在学校异地培养政策的支持下，我在研究院学习和工作了四年有余，工程能力和科研能力跟着项目共同进步和成长。公司全体同仁通力合作，共同研制了多款表面组装设备，为智能装备国产化贡献了力量。我也得益于平台的快速发展，为本学位论文相关实验的顺利开展创造了有利条件。

感谢我的女友赵姗，相识近五年，我们无话不谈。在我攻读博士学位期间，她始终不离不弃地陪伴在我身边，即使面对长时间的异地恋，她也始终支持我的学习和工作。无论是我的正面或是负面情绪，她都愿意做我生活中的倾听者和分享者，我很幸运能遇到理解我的另一半。未来的路，我们一同走。

特别感谢我的父母，他们含辛茹苦地抚养我长大，始终默默支持我的求学之路。他们虽不学识渊博，但总是以鼓励的态度支持我继续深造。他们是我生命中最重要的人，是他们让我认识到知识改变命运，也是他们教会我做人做事的道理。感谢他们对我无微不至的关怀和辛勤付出，是他们在我成长的道路上指引方向，使我能在求学和科研的道路上坚定不移地走下去。

本文的研究得到国家自然科学基金项目（项目编号：U20A20188）、国家重点研发计划项目（项目编号：2024YFB3409200）等科研项目的资助，谨此致谢。基金为实验工作的开展提供了关键支持，确保了研究的顺利推进。

感谢各位评阅专家能在百忙之中参与论文的答辩和审阅工作。

生在红旗下，长在春风里。我为自己能生活在这样一个伟大的时代和国家而感到无比幸运和自豪。感谢祖国，感谢中国共产党。未来，我将继续发光发热，用自己的微薄之力为祖国的发展贡献力量！

2025 年 4 月
于哈尔滨工业大学明德楼

个人简历

卢光宇，男，汉族，中共党员，1996年6月出生于山西省太原市。

学习经历：

2015年9月，考入大连海事大学信息科学技术学院自动化专业，2017年9月转入船舶电气工程学院，2019年6月本科毕业并获得工学学士学位。

2019年9月，推荐免试攻读哈尔滨工业大学航天学院智能控制与系统研究所控制科学与工程硕士学位，同年获得硕博连读资格；2020年9月-至今，攻读控制科学与工程博士学位。

学术成果：

- 1、在国际顶级期刊 IEEE Transactions on Industrial Informatics/IEEE Transactions on Cybernetics 上共发表学术论文 3 篇。
- 2、申请并授权国家发明专利 8 项。

参与的科研项目

- 1、2020年-至今，高速高精度贴片机研发，参与贴片机软件开发、表面组装过程算法设计、表面组装生产线调度算法研究等工作，团队研制贴片机初步投入市场应用。
- 2、2021年-2024年，国家自然科学基金重点支持项目（联合）项目“表面贴装高端智能装备基于视觉的高速高精度运动控制理论与方法”，参与表面贴装装备中智能优化算法的研究工作。
- 3、2024年-至今，国家重点研发计划“高性能制造技术与重大装备”重点专项“精密电子表面焊装全工序调控理论与方法”，参与表面焊装生产线全工序调控与优化算法的研究工作。

荣誉奖励：

- 1、2015-2016 学年一等优秀学业奖学金，2016-2017 学年、2017-2018 学年本科生国家奖学金。
- 2、2019 年辽宁省普通高等学校优秀毕业生。
- 3、2022-2023 学年博士生国家奖学金。
- 4、2025 年哈尔滨工业大学优秀毕业生。