

基于改进布谷鸟搜索算法的 TFT-LCD 制造调度方法^①



刘庭宇, 叶春明

(上海理工大学 管理学院, 上海 200093)

通讯作者: 刘庭宇, E-mail: 553828539@qq.com

摘要: 针对基于改进布谷鸟搜索算法的 TFT-LCD 制造 cell 阶段绿色调度问题, 建立了以最小化最大完工时间和碳排放总量为目标的数学模型. 采用基于机器选择、转速选择和工序选择的三段式编码, 应用在步长因子前加入动态系数的改进布谷鸟搜索算法, 结合双元锦标赛和动态淘汰制来构建 Pareto 最优解集. 通过对某车间实际生产数据进行仿真, 验证了模型和算法的有效性, 仿真结果表明, 改进布谷鸟搜索算法在保障最大完工时间的前提下, 可以有效地减少碳排放量.

关键词: TFT-LCD 制造 cell 阶段; 机器转速; 碳排放; 改进布谷鸟搜索算法; Pareto 最优解集

引用格式: 刘庭宇, 叶春明. 基于改进布谷鸟搜索算法的 TFT-LCD 制造调度方法. 计算机系统应用, 2020, 29(3): 47-54. <http://www.c-s-a.org.cn/1003-3254/7211.html>

TFT-LCD Manufacturing Scheduling Method Based on Improved Cuckoo Search Algorithm

LIU Ting-Yu, YE Chun-Ming

(Business School, University of Shanghai for Science and Technology, Shanghai 200093, China)

Abstract: Considering the green scheduling problem of TFT-LCD manufacturing cell stage based on improved cuckoo search algorithm, a mathematical model was established aiming at minimization of the maximum completion time and total carbon emissions. By using three-stage coding based on machine selection, speed selection and process selection and using an improved cuckoo search algorithm with dynamic coefficients before step size factor, the Pareto optimal solution set is constructed by combining the dual championship and the dynamic elimination system. The validity of the model and algorithm is verified by simulating the actual production data of a workshop. The simulation results show that the improved cuckoo search algorithm can effectively reduce carbon emissions while guaranteeing the maximum completion time.

Key words: TFT-LCD manufacturing cell stage; machine speed; carbon emissions; improved cuckoo search algorithm; Pareto optimal solution set

由于资源消耗和环境影响, 可持续性成为工业界的一个重要课题. 其中, 能源消耗和碳排放是制造业在绿色条件下运作的两个主要问题. Garetti 等^[1]指出制造业占世界能源消耗的 33%, 碳排放总量的 38% 以上.

随着阴极射线管 (CRT) 被平板显示器取代, 薄膜晶体管液晶显示器 (TFT-LCD) 面板在监控设备、液晶电视、移动电话和平板个人电脑等领域广泛应用, 其需求迅速增长. 与其他平板显示类产品^[2]相比, 10 英寸

① 基金项目: 国家自然科学基金 (71840003); 上海理工大学科技发展基金 (2018KJFZ043)

Foundation item: National Natural Science Foundation of China (71840003); Science and Technology Development Fund of Shanghai University for Science and Technology (2018KJFZ043)

收稿时间: 2019-05-29; 修改时间: 2019-06-24; 采用时间: 2019-07-01; csa 在线出版时间: 2020-02-28

TFT-LCD 及以上产品的年销售总量占全球的 50% 以上。TFT-LCD 产品在生产过程中消耗大量的能源, 电力约占 75%, 其中, 机器运转所消耗的电力占 45~50%^[3]。因此, 如何通过生产调度, 有效降低 TFT-LCD 产品生产过程中的能源消耗和碳排放量, 达到节能减排显得尤为重要。

在 TFT-LCD 生产调度研究领域, 阵列工艺 (array) 阶段包含大量可重入制造工艺, 面板工艺 (cell) 阶段包含大量并行机工艺, 模块工艺 (module) 阶段可以看作柔性作业车间问题。在 array 阶段, Choi 等^[4]提出了一种具有决策树的实时调度机制, 它消除了通过仿真运行选择调度规则所需的计算负担。Hong 等^[5]提出一种两相解码遗传算法, 以最大限度地利用光刻阶段, 提高了机器利用率和生产系统的能力。在 cell 阶段, Lin 等^[6]在考虑批量释放时间的同时也考虑了调度规则的影响, 并提出一种基于批量释放时间的启发式算法和基于队列时间最大不匹配的摩擦机调度规则, 提高了 cell 阶段的生产效率。Wu 等^[7]建立了一种 DBR 定制模型, 并提出了一种利用转鼓缓冲绳 (DBR) 系统对 cell 阶段进行调度控制的方法, 提高了 cell 阶段的生产效率。徐峰等^[8]以最小化最大完工时间和交货期最短为目标函数, 使用精英保留和贪婪解码相结合的遗传算法进行求解, 缩短了 cell 阶段的生产时间。吴思思等^[9]以最小化最大完工时间、机器等待时间和交货期为目标, 使用布谷鸟算法进行求解, 并首次在 cell 阶段引入学习退化效应, 分析了学习退化效应对调度结果的影响。在 module 阶段, Chou 等^[10]提出一种多目标混合遗传算法来解决 TFT-LCD 模块装配的调度问题, 该问题是一种柔性作业车间调度问题: 力求在满足客户需求的同时, 最大限度地减少制造时间和机器加工准备时间。

绿色调度通过对资源的合理分配和优化工件排序, 以达到增效、节能、减排、降耗的目的, 提高经济效益的同时实现制造过程的绿色化^[11]。在绿色调度研究领域, 针对并行机调度问题, Cataldo 等^[12]在计算总能耗时适当考虑不同路径移动待加工零件所需的能耗, 并采用模型计算最优控制行为, 以限制总能耗, 使整体产量最大化。Wang 等^[13]提出一种两阶段启发式求解并行机调度问题的方法, 该方法的目标是使最大完工时间最小。Ji 等^[14]提出了一种新的粒子群优化算法, 在最大完工时间在不超过一定水平的情况下, 使资源消耗最小化。Li 等^[15]提出了一种基于 LPT 规则的启发式求解方法, 在机器总成本不超过给定阈值的约束下, 使最

大完工时间最小化。针对流水作业车间调度问题, Ding 等^[16]提出了一种多目标 NEH 算法和一种改进的多目标迭代贪心算法来解决置换流车间低碳调度问题, 旨在降低加工过程的能耗, 从而减少碳排放。Zhai 等^[17]利用时间序列模型对部分可再生能源的电价进行逐时更新, 将电价反馈到调度模型中, 优化能源成本。Huang 等^[18]通过优化维修流程, 提高了能源效率, 减少了两阶段多处理器流水车间调度问题中的最大完工时间。Lu 等^[19]提出了一种混合多目标回溯搜索算法来提高置换流车间调度问题的能源效率, 该算法主要考虑了准备阶段和加工阶段的能源消耗。

综上所述, 近些年随着人们对环境的重视, 绿色调度的研究越来越多, 而在 TFT-LCD 制造 cell 阶段调度问题上, 大多数研究只考虑了经济效益, 并未考虑对环境的影响。因此本文研究以最大完工时间和碳排放量为目标的 TFT-LCD 制造 cell 阶段绿色调度问题, 在考虑机器选择和工序的基础上, 增加了机器转速的选择, 不同的机器转速会影响加工时间的长短和碳排放量的多少。本文提出一种改进布谷鸟搜索算法, 通过在步长因子加入动态系数, 提高算法的搜索效率和解的质量, 并构建 Pareto 最优解集。

1 问题描述

1.1 TFT-LCD 装配过程描述

薄膜晶体管液晶显示器 (TFT-LCD) 的制造工艺由阵列工艺 (array)、面板工艺 (cell) 和模块工艺 (module) 等 3 个基本工艺阶段组成。除了材料成分外, TFT 的阵列工艺与半导体晶圆的阵列工艺非常相似。阵列工艺的主要原料是玻璃基板, 经过清洗、涂布、曝光、显影、蚀刻等 5-7 次加工。面板工艺包括许多组装步骤来组装 TFT 玻璃基板和彩色滤光膜。模块工艺是 TFT-LCD 制造工艺的最后一个阶段, 通过面板工艺生产的 TFT-LCD 面板与所有其他必要部件组装在一起, 完成最终的 TFT-LCD 产品。

TFT-LCD 制造中的 cell 阶段是将彩色滤光膜 (CF) 和 TFT 玻璃基板两个部件组装在一起的组装过程。通常彩色滤光膜是从外部供应商购买的, 而 TFT 玻璃基板是由自己的阵列工厂生产的。在整个组装过程中, TFT 玻璃基板和彩色滤光膜分别经过配向膜印刷, 摩擦, 密封胶/隔离子涂布, 粘接, 切割, 填充液晶, 偏振片粘附和检查等 11 道工序。cell 阶段可以看作非等效并行机的混合流水车间问题。cell 阶段制造工艺流程如图 1 所示。

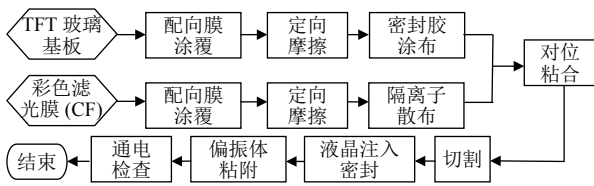


图1 TFT-LCD制造cell阶段制造工艺流程

1.2 TFT-LCD制造cell阶段绿色调度模型建立

本文涉及的数学符号及含义如下:

n : 工件类型数 ($i = 1, 2, \dots, n$);

N : 工件批数, 包括零件 A, B 及装配后工件 C;

J_r/J_r^* : 第 r /第 r^* 批零件或工件 ($r = 1, 2, \dots, N$), 其中, r 代表零件 A, r^* 代表零件 B;

O_j : 工件第 j 道工序 ($j = 1, 2, \dots, a, a+1, \dots, b, b+1, \dots, c$), 其中零件 A 批加工工序为 $j = 1, 2, \dots, a$, 零件 B 批加工工序为 $j = a+1, \dots, b$, 装配后工序为 $j = b+1, \dots, c$;

m : 当前工序下的第 m 台机器 ($m = 1, 2, \dots, M$);

v_l : 每台机器的转速 ($V = v_1, v_2, \dots, v_l, \dots, v_d$), 共 d 种转速;

t_{rjm} : 第 r 批工件第 j 道工序在第 m 台机器上的开始加工时间;

P_{rjm} : 第 r 批工件第 j 道工序在第 m 台机器上的基础加工时间;

P_{rjml} : 第 r 批工件第 j 道工序在第 m 台机器上以转速 l 加工的时间;

D_{rjm} : 第 r 批工件第 j 道工序在第 m 台机器上的结束加工时间;

W_{jml} : 第 j 道工序第 m 台机器上以转速 l 加工的单位能耗;

l_{jm} : 第 j 道工序第 m 台机器上的单位空载能耗;

S_{ij} : 工序 O_j 不同类型工件之间转换的准备时间;

X_{rjml} : 第 r 批工件第 j 道工序在第 m 台机器以转速 l 加工为 1, 否则为 0;

$Y_{jrr'm}$: 在第 j 道工序, J_r 先于 J_r^* 在第 m 台机器上加工为 1, 否则为 0.

基于机器转速的 cell 阶段绿色调度在满足约束条件的前提下, 不仅要考虑机器的选择和工件的排序, 也要考虑机器转速的选择. 即在约束条件下对每一批工件, 确定加工工序, 选择合适的机器和转速, 确保最终优化目标达到满意解. 模型中涉及的假设如下:

(1) 某一时刻, 每台机器只能加工一批工件 (在装

配工序中, 零件 A 和零件 B 合成一批工件 C), 并且加工开始后, 机器转速保持不变.

(2) 某一时刻, 一批工件只能在一台机器上加工, 并且加工开始后, 加工过程不能被中断.

(3) 所有机器在整个调度期间都是可用的 (没有机器故障).

(4) 零件 A、零件 B 和装配后工件 C 的加工路径事先确定.

(5) 每道工序的加工时间由机器转速和工件类型共同决定.

因此, 本文建立基于机器转速的 TFT-LCD 制造 cell 阶段绿色调度模型, 优化目标为最大完工时间 T_{\max} 和总碳排放量 C_{total} .

目标函数:

$$T_{\max} = \max(t_{rjm} + P_{rjml}X_{rjml}) \quad (1)$$

$$C_{\text{total}} = \varepsilon \sum_{j=1}^{O_j} \sum_{m=1}^M \sum_{r=1}^N \sum_{l=1}^d (P_{rjml}X_{rjml}W_{jml} + (D_{jm} - X_{rjml}P_{rjml} - t_{jm})l_{jm}) \quad (2)$$

约束条件:

$$\sum_{j=1}^{O_j} \sum_{m=1}^M \sum_{l=1}^d X_{rjml} \leq 1 \quad (3)$$

$$\sum_{r=1}^N X_{rjml} \leq 1 \quad (4)$$

$$Z_1 X_{rjml} \geq t_{rjm} \quad (5)$$

$$t_{rj'm'} \geq t_{rjm} + P_{rjml}X_{rjml} \quad (6)$$

$$T_{\max} = t_{rjm} + P_{rjml}X_{rjml} \quad (7)$$

$$t_{rjm} + P_{rjml}X_{rjml} + S_{ij} - t_{r'jm} + Z_2(Y_{jrr'm} - 1) \leq 0 \quad (8)$$

$$(Y_{jrr'm} + Y_{jr'rm}) + Z_2(X_{rjml} + X_{r'jml} - 2) \leq 1 \quad (9)$$

$$(Y_{jrr'm} + Y_{jr'rm}) - Z_2(X_{rjml} + X_{r'jml}) \leq 0 \quad (10)$$

$$(Y_{jrr'm} + Y_{jr'rm}) - Z_2(X_{rjml} - X_{r'jml} + 1) \leq 0 \quad (11)$$

$$X_{rjml} = X_{r^*jml} \quad (j = b+1) \quad (12)$$

$$t_{rjm} = t_{r^*jm} \quad (j = b+1) \quad (13)$$

式中, Z_1 和 Z_2 为足够大的正数. 式 (1) 中, T_{\max} 代表最大完工时间; 式 (2) 中, C_{total} 代表碳排放总量, ε 为能耗

与碳排放量之间的转换系数,通常取 0.7559^[20];式(3)表示在某一时刻,一批工件只能在一台机器上加工;式(4)表示在某一时刻,一台机器只能加工一批工件;式(5)表示任何一批工件在某一机器加工时,都有相应开始加工时间;式(6)表示在加工过程中,工件要严格按照事先确定的加工路径进行加工;式(7)表示任何一批工件的结束加工时间不能超过最大完工时间;式(8)表示同一台机器前后两批工件的开始加工时间约束;式(9)~式(11)表示同一台机器加工不同工件批之间关系约束;式(12)、式(13)表示装配工序时,零件 A 和零件 B 同时在同一机器上加工。

2 改进布谷鸟搜索算法

2.1 标准布谷鸟搜索算法描述

2.1.1 布谷鸟繁殖行为

由于布谷鸟的寄生特性,布谷鸟依赖于一些宿主种类并在宿主巢中下蛋。宿主鸟会把布谷鸟的蛋误认成自己的蛋,孵出布谷鸟的幼鸟。然而,并不是所有的布谷鸟蛋都不会被宿主鸟发现,一旦发现,宿主鸟会扔掉布谷鸟蛋或选择别处重新筑巢。因此,布谷鸟会选择一些与自身孵化方式相似的宿主鸟。

2.1.2 莱维飞行

除了布谷鸟的寄生特征外, Yang 等^[21]提出的布谷鸟搜索算法(Cuckoo Search Algorithm, CSA)依赖于布谷鸟的觅食性质。布谷鸟的觅食模式受到一个重要因素的控制,这个因素被称为莱维飞行,莱维飞行是一种步长遵循重尾概率分布的结构化随机游走。很多动物和昆虫的觅食路径也遵循以莱维飞行为特征的重尾概率分布的结构化随机游走^[22]。Yang 等^[23]证明了用基于莱维飞行的结构化随机游走代替纯随机游走的布谷鸟搜索算法比许多现有的元启发式算法(如粒子群算法、蜂群算法和差分进化算法)更有效。

2.1.3 布谷鸟搜索过程

在布谷鸟搜索算法中,巢中的蛋是一种解决方案,布谷鸟蛋代表一种新的解决方案。为新一代选择的鸟蛋质量较好的鸟巢,对应的是鸟蛋产生新布谷鸟的能力。布谷鸟搜索算法是基于以下假设构建的:

- (1) 每只布谷鸟选择一个宿主巢,然后随机下蛋。
- (2) 为下一代选择最好的鸟巢。

(3) 巢的数量保持不变,宿主鸟可以以一定的概率 $pa \in [0, 1]$ 扔掉外来的蛋。

利用莱维飞行的布谷鸟第 w 个鸟巢第 $p+1$ 代的位置为:

$$X_w^{p+1} = X_w^p + \alpha Levy(\lambda) \quad (14)$$

式(14)中, X_w^p 是当前解, $\alpha > 0$ 表示步长因子,大小取决于问题的规模。该值是常量,它具有固定的、有限的时间复杂度。布谷鸟随机选择宿主巢下蛋,采用基于莱维飞行的随机游走。莱维飞行符合以下概率分布:

$$Levy(\lambda) \sim u = p^{-\lambda} \in (1, 3] \quad (15)$$

上述莱维分布服从重尾概率分布的阶跃幂律。方差和均值都是无穷大。步长将搜索域划分为局部搜索空间和全局搜索空间,为全局搜索提供了机会,而不是停留在局部最小值上。由于布谷鸟搜索算法采用的是在莱维飞行中进行全局随机游走,因此能够更快的在搜索空间内达到最优。

2.2 改进布谷鸟搜索算法描述

在标准布谷鸟搜索算法中,步长因子为固定的值。步长因子的大小会直接影响求解的速度和效率,当步长因子过大时,虽具备较好的全局搜索能力,收敛速度也快,但易错过全局最优解或在全局最优解附近震荡;当步长因子过小时,收敛速度变慢,也易陷入局部最优解中,该缺点限制了 CSA 算法的快速收敛性,并有可能导致无法获得高质量的解。因此,为了更好实现全局快速搜索和局部精确搜索,朝着全局最优解快速而稳定的搜索,本文提出一种改进布谷鸟搜索算法(Improved Cuckoo Search Algorithm, ICSA),在步长因子 α 前加入动态系数 β ,如式(16):

$$\beta = \omega(T_max - Times) + \beta_0 \quad (16)$$

式(16)中, T_max 表示最大迭代次数, $Times$ 表示当前迭代次数, $\omega > 0$ 表示变化率,控制减小的幅度, $\beta_0 > 0$ 表示最小动态系数,防止步长因子缩减到 0, ω 和 β_0 的大小取决于问题的规模。可以看出,动态系数 β 随着迭代次数的增加而线性递减,一方面,在搜索前期,有利于对全局进行快速搜索,寻找优质解所在区域;在搜索后期,有利于加深对优质解周边区域精确搜索,找到优质解;另一方面,动态系数能够有效的提高算法的搜索能力,平衡全局搜索和局部搜索的关系,通过迭代前期较大的动态系数,扩大搜索空间,加强全局搜索的能力,迭代后期较小的动态系数,使得算法的局部搜索能力不断加强。因此,改进布谷鸟搜索算法的更新位置公式

为式 (17):

$$X_w^{p+1} = X_w^p + \beta \alpha \text{Levy}(\lambda) \quad (17)$$

相比于其他改进方式, 本文提出的改进方式参数少, 实现方式简单, 求解速度快.

2.3 构建 Pareto 最优解集

与单目标优化问题不同的是, 多目标优化问题目标之间可能存在矛盾, 这时可以构建 Pareto 最优解集来解决. Pareto 最优解集是指: 在解集 K 中, 对于解 x 、 y , 若解 x 的所有目标函数值不劣于解 y , 且解 x 至少存在一个目标函数值优于解 y , 则称 x 支配 y , 记作 $x < y$, 若解集 K 中不存在任一解的所有目标函数值不劣于解 x , 且存在一个目标函数值优于解 x , 则称解 x 为 Pareto 最优解集中的一个解.

本文结合二元锦标赛和动态淘汰制 2 种方法构建 Pareto 最优解集. 实现过程为: 选择解集 K 中的一个解 x , 将其与剩余解进行比较, 若存在解被 x 支配, 则把该解从解集 K 中删除, 若存在解支配 x , 则直接把解 x 从解集 K 中删除, 若剩余解皆不支配 x , 则把解 x 加入 Pareto 最优解集中, 并把解 x 从解集 K 中删除, 重复这一过程, 直至解集 K 为空集. 若存在上代 Pareto 最优解集, 合并两代 Pareto 最优解集, 继续比较. 若 Pareto 最优解集规模大于规定值, 则使用聚焦距离进行筛选.

2.4 改进布谷鸟搜索算法步骤描述

改进布谷鸟搜索算法遵循以莱维飞行为特征的重尾概率分布的结构化随机游走. 莱维飞行为全局搜索提供了机会, 而改进布谷鸟搜索算法在搜索前期, 有利于对全局进行快速搜索, 寻找优质解所在区域; 在搜索后期, 有利于加深对优质解周边区域精确搜索, 找到优质解. 在结合二元锦标赛、动态淘汰制和聚焦距离筛选 3 种方法之后, 改进布谷鸟搜索算法步骤如下:

(1) 参数初始化. 设置种群规模: z ; 最大迭代次数: T_max ; 被宿主发现的概率: pa ; Pareto 解的个数: Z ; 步长因子: α ; 变化率: ω ; 最小动态系数: β_0 .

(2) 种群初始化. 随机选取 z 个鸟巢位置.

(3) 计算鸟巢位置所对应的两个目标函数值, 构建初始 Pareto 最优解集.

(4) 根据式 (17) 得到下一代鸟巢位置, 并计算其目标函数值.

(5) 每个新鸟巢随机产生一个 R , 如果 R 大于被宿主发现的概率 pa , 返回步骤 (4), 否则, 进入步骤 (6).

(6) 比较两代鸟巢目标函数值, 构建新 Pareto 最优解集, 如果 Pareto 最优解个数大于 Z , 那么使用聚焦距离进行筛选.

(7) 若达到最大迭代次数 T_max , 输出最终 Pareto 最优解集, 否则, 回到步骤 (4).

3 实验与结果分析

3.1 TFT-LCD 制造 cell 阶段问题编码

TFT-LCD 制造 cell 阶段绿色调度需要同时考虑当前工件工序机器选择、机器转速选择和工件在机器上加工顺序, 为此, 本文设计一种基于机器选择、转速选择和工序选择相结合的三段式编码方式, 如图 2 所示.



图 2 三段式编码方式

图 2 中, 在基于机器编码中, 下行数字表示工件编号, 上行数字表示当前工件加工机器编号, 表示的机器依次是 $\{m_3, m_5, m_1, m_4, m_2, m_3, m_4, m_1\}$; 在基于转速编码中, 下行数字表示机器标号, 上行数字表示当前机器转速编号, 表示的转速依次是 $\{v_3, v_2, v_5, v_3, v_5, v_1, v_4, v_2\}$; 在基于工序编码中, 数字表示工件编号, 出现频次表示当前工序数, 表示的工序依次是 $\{O_{21}, O_{31}, O_{11}, O_{41}, O_{12}, O_{32}, O_{22}, O_{42}\}$. 第一段编码确定当前工件工序的加工机器, 第二段编码确定当前加工机器的转速, 第三段编码确定工件工序的加工顺序, 三段编码相结合确保所求解可行.

3.2 测试实例和对比算法

Ding 等^[16]提出了一种关于加工时间和能耗之间的假设: 对于确定的工序和机器, 随着转速的变大, 加工时间减少, 能耗增加, 即对于 $\forall v_l > v_g, l, g \in \{1, 2, \dots, d\}$ 满足 $P_{rjml} < P_{rjmg}, P_{rjml} \times W_{jml} > P_{rjmg} \times W_{jmg}$. 测试实例选取某车间的实际生产数据^[8], 产品类型及批量数如表 1

所示, 工序机器数如表 2 所示, 不同工件工序在机器上的加工时间如表 3 所示, 不同类型工件在各工序之间转换的准备时间如表 4 所示. 需要增加机器转速和能耗信息, $v = \{1.00, 1.30, 1.55, 1.80, 2.00\}$, $W_{jml} = 4 \times v_l^2$, $l_{jm} = 1$, 显然加工时间和能耗之间的关系满足上述假设.

表 1 产品类型及批量数

产品类型	批量数
类型 1-TFT	3
类型 1-CF	3
类型 2-TFT	4
类型 2-CF	4
类型 3-TFT	2
类型 2-CF	2

表 2 工序机器数

工序	1	2	3	4	5	6	7	8	9	10	11
机器数	3	2	3	3	2	2	2	3	3	1	2

表 3 不同工序加工时间 (单位: 分钟)

类型	1	2	3	4	5	6	7	8	9	10	11
1	12	12	12	12	12	12	12	45	60	15	40
2	15	15	15	15	15	15	15	47	65	18	40
3	20	20	18	20	20	20	15	50	80	20	45

表 4 不同工序、不同类型工件之间转换的准备时间 (单位: 分钟)

类型	1	2	3	4	5	6	7	8	9	10	11
1	15	30	18	15	30	15	25	15	16	15	0
2	15	30	18	15	30	15	25	15	20	15	0
3	20	35	20	20	35	15	30	20	22	15	0

本文选择标准布谷鸟搜索算法 (CSA) 和 Deb 等^[24] 提出的带精英策略的快速非支配排序遗传算法 (NSGA-II) 作为对比算法. 由于原算法没有对机器转速选择进行编码, 对于 CSA, 采取与本文一致的编码方式; 对于 NSGA-II, 增加机器转速的选择、交叉和变异部分. 两种算法构建 Pareto 最优解集的方法与本文一致.

3.3 结果分析

求解 cell 阶段绿色调度问题算法的运行环境为: 操作系统为 Windows 10, 处理器为 Intel(R) Core(TM) i5-3210M, 主频为 2.50 GHz, 内存为 6 GB, 编程环境为 Matlab R2018a.

基于大量测试得到, 当变化率 ω 为 0.02, 最小动态系数 β_0 为 0.5 时, 算法性能最佳, 因此 ICSA 的 6 个参数, 设计如下: 种群规模 z 为 50, 最大迭代次数 T_{max} 为 100, 被宿主发现的概率 pa 为 0.25, 步长因子 α 为

0.1, 变化率 ω 为 0.02, 最小动态系数 β_0 为 0.5; CSA 具有 4 个参数, 设计如下: 种群规模 z 为 50, 最大迭代次数 T_{max} 为 100, 被宿主发现的概率 pa 为 0.25, 步长因子 α 为 0.1; NSGA-II 具有 5 个参数, 设计如下: 种群规模为 50, 最大迭代次数为 100, 选择概率为 0.9, 交叉率为 0.8, 变异率为 0.1. 3 种算法 Pareto 解的个数 Z 都为 10.

每种算法各运行 30 次, 每次运行结果的最好解 (取整) 如表 5 所示.

表 5 3 种算法最好解的结果对比

次数	ICSA		CSA		NSGA-II	
	T_{max}	C_{total}	T_{max}	C_{total}	T_{max}	C_{total}
1	489	10 006	470	10 354	532	10 942
2	467	10 249	490	10 273	533	11 134
3	470	10 070	496	10 341	539	11 120
4	497	9944	484	10 340	588	10 874
5	481	10 218	503	10 419	516	11 370
6	483	10 362	499	10 453	556	11 169
7	470	10 105	486	10 397	484	11 022
8	486	9969	502	10 286	542	11 149
9	503	10 145	489	10 428	548	11 234
10	469	10 218	501	10 277	525	10 767
11	467	10 255	487	10 422	540	10 984
12	490	10 118	501	10 564	583	11 110
13	466	10 235	491	10 310	567	10 984
14	466	10 334	479	10 600	536	11 098
15	481	9998	510	10 312	503	11 282
16	488	10 162	500	10 202	581	10 945
17	494	10 162	491	10 381	469	11 431
18	484	10 258	502	10 450	566	11 298
19	509	10 116	463	10 236	554	11 248
20	479	10 258	486	10 349	556	11 330
21	448	10 299	495	10 495	558	11 380
22	438	10 345	490	10 412	563	11 268
23	444	10 391	505	10 434	527	11 290
24	479	10 071	503	10 137	487	11 246
25	487	10 169	502	10 260	496	11 378
26	468	10 254	503	10 470	519	11 202
27	441	10 493	489	10 349	488	11 121
28	486	10 245	491	10 312	540	11 018
29	462	10 343	462	10 485	525	11 417
30	515	10 267	450	10 401	521	11 228

一方面, 总共 30 次运行结果中, ICSA 在 23 次运行中得到了优于 CSA 和 NSGA-II 的最好解, 剩余 7 次运行结果也只是单个目标值劣于 CSA 或 NSGA-II, 没有两个目标值都劣于 CSA 或 NSGA-II 的情况发生. 另一方面, 表 5 中有 4 次最大完工时间小于 450, 3 次碳排放总量小于 10 000, 均由 ICSA 运行得到, 说明 ICSA 能够得到更好的解.

图 3 是独立运行一次, 不同算法求得的 Pareto 解

分布对比图. 图中, 叉号、五角、圆圈分别代表 ICESA、CSA、NSGA-II 求得的解. 可以看出, ICESA 所求解最好, CSA 次之、NSGA-II 最差. ICESA 的 10 个解支配 CSA 的 8 个解和 NSGA-II 的 10 个解, 即在最大完工时间相同的情况下, 碳排放总量更小. 因此, ICESA 的表现要优于 CSA 和 NSGA-II. 另外, 最大完工时间越短, 碳排放量越高, 因此在完工时间允许的情况下, 选择较低的转速, 可以有效的减少碳排放量.

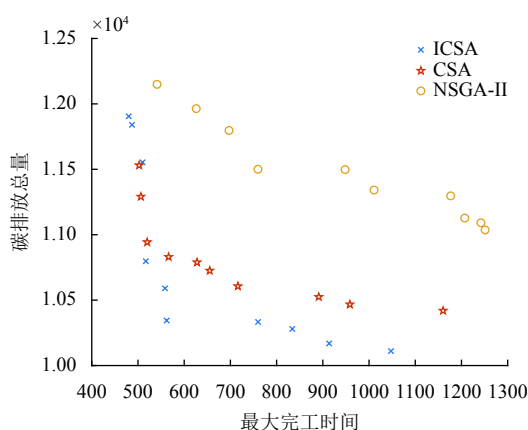


图3 3种算法 Pareto 解分布对比

综上所述, 结合表5和图3可以得到, 在求解基于机器转速的 TFT-LCD 制造 cell 阶段绿色调度问题上, ICESA 的求解质量要高于 CSA 和 NSGA-II.

4 结论与展望

本文研究基于改进布谷鸟搜索算法的 TFT-LCD 制造 cell 阶段绿色调度问题, 建立以最小化最大完工时间和碳排放总量为目标的数学模型. 对基于机器选择、转速选择和工序选择相结合的三段式编码后, 应用一种改进布谷鸟搜索算法进行求解, 该算法在步长因子前加入动态系数, 使其在搜索前期, 快速对全局进行搜索, 在搜索后期, 加深对局部进行搜索, 提高了解的质量. 并对某车间的实际生产数据进行仿真, 实验结果表明, 相比于 CSA 和 NSGA-II, 在最大完工时间相同的情况下, ICESA 可以有效的减少碳排放量. 关于 TFT-LCD 多目标绿色调度的研究才刚刚开始, 作者会继续研究关于 TFT-LCD 制造 array 阶段和 module 阶段绿色调度问题, 同时作者将深入研究 ICESA 和其他智能算法及改进策略, 为后续研究的问题提供算法支撑.

参考文献

- Garetti M, Taisch M. Sustainable manufacturing: Trends and research challenges. *Production Planning & Control*, 2012, 23(2-3): 83-104.
- Lee ZY, Pai CC. Operation analysis and performance assessment for TFT-LCD manufacturers using improved DEA. *Expert Systems with Applications*, 2011, 38(4): 4014-4024. [doi: 10.1016/j.eswa.2010.09.063]
- 程星华, 李强. TFT-LCD 工厂能源消费结构及用能特点分析. *液晶与显示*, 2014, 29(5): 856-863.
- Choi HS, Kim JS, Lee DH. Real-time scheduling for reentrant hybrid flow shops: A decision tree based mechanism and its application to a TFT-LCD line. *Expert Systems with Applications*, 2011, 38(4): 3514-3521. [doi: 10.1016/j.eswa.2010.08.139]
- Hong TY, Chien CF, Wang HK, et al. A two-phase decoding genetic algorithm for TFT-LCD array photolithography stage scheduling problem with constrained waiting time. *Computers & Industrial Engineering*, 2018, 125: 200-211.
- Lin JT, Wang FK, Peng CC. Lot release times and dispatching rule for a TFT-LCD cell process. *Robotics and Computer-Integrated Manufacturing*, 2008, 24(2): 228-238. [doi: 10.1016/j.rcim.2006.10.006]
- Wu HH, Chen CP, Tsai CH, et al. Simulation and scheduling implementation study of TFT-LCD cell plants using Drum-Buffer-Rope system. *Expert Systems with Applications*, 2010, 37(12): 8127-8133. [doi: 10.1016/j.eswa.2010.05.075]
- 徐峰, 步丰林. 改进遗传算法求解混合流水装配作业调度问题. *微型电脑应用*, 2013, 29(9): 58-61. [doi: 10.3969/j.issn.1007-757X.2013.09.017]
- 吴思思, 叶春明, 李瑞婷. 具有学习退化效应的 TFT-LCD 面板成盒多目标调度问题研究. *上海理工大学学报*, 2018, 40(3): 238-248.
- Chou CW, Chien CF, Gen M. A multiobjective hybrid genetic algorithm for TFT-LCD module assembly scheduling. *IEEE Transactions on Automation Science and Engineering*, 2014, 11(3): 692-705. [doi: 10.1109/TASE.2014.2316193]
- Gahm C, Denz F, Dirr M, et al. Energy-efficient scheduling in manufacturing companies: A review and research framework. *European Journal of Operational Research*, 2016, 248(3): 744-757. [doi: 10.1016/j.ejor.2015.07.017]
- Cataldo A, Perizzato A, Scattolini R. Production scheduling of parallel machines with model predictive control. *Control Engineering Practice*, 2015, 42: 28-40. [doi: 10.1016/j.conengprac.2015.05.007]

- 13 Wang YC, Wang MJ, Lin SC. Selection of cutting conditions for power constrained parallel machine scheduling. *Robotics and Computer-Integrated Manufacturing*, 2017, 43: 105–110. [doi: [10.1016/j.rcim.2015.10.010](https://doi.org/10.1016/j.rcim.2015.10.010)]
- 14 Ji M, Wang JY, Lee WC. Minimizing resource consumption on uniform parallel machines with a bound on makespan. *Computers & Operations Research*, 2013, 40(12): 2970–2974.
- 15 Li K, Zhang X, Leung JYT, *et al.* Parallel machine scheduling problems in green manufacturing industry. *Journal of Manufacturing Systems*, 2016, 38: 98–106. [doi: [10.1016/j.jmsy.2015.11.006](https://doi.org/10.1016/j.jmsy.2015.11.006)]
- 16 Ding JY, Song SJ, Wu C. Carbon-efficient scheduling of flow shops by multi-objective optimization. *European Journal of Operational Research*, 2016, 248(3): 758–771. [doi: [10.1016/j.ejor.2015.05.019](https://doi.org/10.1016/j.ejor.2015.05.019)]
- 17 Zhai YX, Biel K, Zhao F, *et al.* Dynamic scheduling of a flow shop with on-site wind generation for energy cost reduction under real time electricity pricing. *CIRP Annals*, 2017, 66(1): 41–44. [doi: [10.1016/j.cirp.2017.04.099](https://doi.org/10.1016/j.cirp.2017.04.099)]
- 18 Huang RH, Yu SC. Two-stage multiprocessor flow shop scheduling with deteriorating maintenance in cleaner production. *Journal of Cleaner Production*, 2016, 135: 276–283. [doi: [10.1016/j.jclepro.2016.06.109](https://doi.org/10.1016/j.jclepro.2016.06.109)]
- 19 Lu C, Gao L, Li XY, *et al.* Energy-efficient permutation flow shop scheduling problem using a hybrid multi-objective backtracking search algorithm. *Journal of Cleaner Production*, 2017, 144: 228–238. [doi: [10.1016/j.jclepro.2017.01.011](https://doi.org/10.1016/j.jclepro.2017.01.011)]
- 20 赵敏, 张卫国, 余立中. 上海市能源消费碳排放分析. *环境科学研究*, 2009, 22(8): 984–989.
- 21 Yang XS, Deb S. Cuckoo search via Lévy flights. *Proceedings of World Congress on Nature & Biologically Inspired Computing*. Coimbatore, India. 2009. 210–214.
- 22 Brown CT, Liebovitch LS, Glendon R. Levy flights in Dobe Ju/'hoansi foraging patterns. *Human Ecology*, 2007, 35(1): 129–138. [doi: [10.1007/s10745-006-9083-4](https://doi.org/10.1007/s10745-006-9083-4)]
- 23 Yang XS, Deb S. Cuckoo search: Recent advances and applications. *Neural Computing and Applications*, 2014, 24(1): 169–174. [doi: [10.1007/s00521-013-1367-1](https://doi.org/10.1007/s00521-013-1367-1)]
- 24 Deb K, Pratap A, Agarwal S, *et al.* A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 2002, 6(2): 182–197. [doi: [10.1109/4235.996017](https://doi.org/10.1109/4235.996017)]