

# 基于莱维萤火虫算法的智能生产线调度问题研究

章菊 李学望

湖北汽车工业学院汽车工程学院 湖北 十堰 442002

汽车动力传动与电子控制湖北省重点实验室 湖北 十堰 442002

**摘要** 针对智能生产线调度过程中易出现饥饿和堵塞等问题,通过分析调度过程,建立调度问题的目标函数和约束条件;然后提出一种基于莱维飞行的新型萤火虫算法,利用莱维分布提高种群的搜索范围和有效性,以最大和最小荧光素作为边界约束优化荧光素迭代公式,提高个体所携带荧光素的合理性;引入立方映射实现对种群的优化,提高种群的搜索能力。算法测试结果表明,莱维萤火虫算法(Levy Glowworm Swarm Optimization, LGSO)相比 GSO(Glowworm Swarm Optimization)、SGSO(Glowworm Swarm Optimization of Scene Understanding)和 CGSO(Chaos Glowworm Swarm Optimization)具有更好的求解精度、收敛性和稳定性。利用 LGSO 算法对典型的 4 种智能生产线的调度问题进行优化求解,并与 GSO 算法和 SGSO 算法进行对比,结果表明:LGSO 算法计算结果的最差值、最优值、结果平均值以及标准偏差基本均优于其他两种算法,特别在复杂路径下,LGSO 算法具有更好的求解精度、收敛速度及稳定性,验证了数学模型的准确性和 LGSO 算法解决调度问题的可行性。

**关键词:** 智能生产线;调度问题;萤火虫优化算法;莱维飞行

**中图分类号** O221

## Research on Intelligent Production Line Scheduling Problem Based on LGSO Algorithm

ZHANG Ju and LI Xue-yun

School of Vehicle Engineering, Hubei University of Automotive Technology, Shiyan, Hubei 442002, China

Key Laboratory of Automotive Power Train and Electronics, Hubei University of Automotive Technology, Shiyan, Hubei 442002, China

**Abstract** Aiming at the problems of starvation and congestion in the scheduling process of intelligent production line, the objective function and constraint conditions of scheduling problem are established by analyzing the scheduling process. Then a new glowworm swarm optimization algorithm based on Levy flight is proposed. Levy distribution is used to improve the search range and effectiveness of the population. The maximum and minimum fluorescein are taken as boundary constraints to optimize the iterative formula of fluorescein, and to improve the rationality of the fluorescein carried by individuals. And the cubic mapping is introduced to optimize the population, so as to improve the comprehensive search ability of the population. The algorithm test results show that LGSO is provided with better solution accuracy, convergence and stability than GSO, SGSO and CGSO. LGSO algorithm is used to solve the scheduling problems of four typical intelligent production lines, and compared with GSO and SGSO. The results show that LGSO is basically better than the other two algorithms in the worst value, the optimal value, the average value and the standard deviation. In the complex path, LGSO has better solution accuracy, convergence speed and stability. Further, the accuracy of the mathematical model and the feasibility of LGSO to solve the scheduling problem are verified by the test.

**Keywords** Intelligent production line, Scheduling problem, Glowworm swarm optimization algorithm, Levy flight

## 1 引言

越来越多的个性化产品定制和激烈的行业竞争推动生产线向智能化、集成化和柔性化发展。柔性生产线效率高且满足并行制造的需求,因此逐渐被制造行业采用。合理的制造过程管理和生产线调度是提高生产效率、减小生产线能耗、提高产品质量的关键手段<sup>[1]</sup>。智能生产线调度是典型的 NP-hard 问题<sup>[2]</sup>,其对于提高资源利用率,降低生产成本,缩短生产周期,提高生产效率具有重要意义。

在智能生产线的调度过程中常出现饥饿和堵塞等问题,

为了解决这一问题,研究者除了通过改进调度方案<sup>[3]</sup>、调度模型研究<sup>[4-7]</sup>和调度策略<sup>[8]</sup>外,还不断将启发式算法、仿生算法、混合算法和智能算法等应用到调度问题中,并进行算法改进研究,例如混合粒子群算法<sup>[9]</sup>、禁忌搜索算法<sup>[10-11]</sup>、多约束启发式算法<sup>[12]</sup>、萤火虫算法<sup>[13]</sup>、遗传算法<sup>[14]</sup>、机器学习算法<sup>[15]</sup>、模糊遗传算法<sup>[16]</sup>,各种算法的应用研究除了完美解决生产线调度问题外,还需要研究如何在多约束下提高了算法的计算精确度和计算效率。

针对智能生产线调度求解问题的复杂性和多约束特点,通过分析生产线的调度过程,确定调度问题的优化目标函数

基金项目:湖北省教育厅科学研究计划项目(B2019083);湖北省重点实验室开放基金项目(ZDK1201905)

This work was supported by the Scientific Research Project of Hubei Provincial Department of Education(B2019083) and Hubei Key Laboratory Open Fund Project(ZDK1201905).

通信作者:章菊(fionanzhang@163.com)

和约束条件,并提出一种基于莱维飞行的新型萤火虫算法,并将其应用于解决调度问题,提高了智能生产线调度求解算法的求解能力和精确度。

## 2 智能生产线调度问题描述

### 2.1 调度过程分析

智能生产线主要包括调度控制中心模块、装载和卸载区域模块、物流运输模块、机器人执行模块以及辅助设备模块。智能生产线开始工作之前,所有物料和待加工工件需在物流中心装载区域内完成准备工作,调度控制中心模块发出指令后,物流运输机器人或运输车将物料或工件运送到各个工位的加工前处理区域;位于加工区域的机器人执行末端按照指令对工件进行加工,然后传递到后处理区域,等待运输系统进行运输;最后由物流机器人将工件从上一工序的后处理区域运输到下一个工序的前处理区域,如果该工件已完成所有工序,则直接运输回到物流中心的卸载区域。

从智能生产线的调度过程可以看出:智能生产线不仅要考虑所有物流机器人的运动路线,还需将运输时间和各个制造单元不同的加工时间相结合,避免工件堆积造成的阻塞状态或者工作站点的空载时间过长导致的饥饿状态。同时还应保证在整个调度过程中不出现干涉、碰撞或者其他机械故障,否则系统会陷入死锁状态。智能生产线的调度过程需遵循以下原则:

- (1)单台机器人单次只能运送单个工件到工作站的处理区域;
- (2)物流中心和工作站的前处理和后处理区域具有足够的物料放置空间;
- (3)当工作站的执行区域完成上一个工件的加工之后,存放在前处理区域的工件自动传送到加工区域;
- (4)忽略工件尺寸对装载及运输所造成的时间差异性。

### 2.2 调度问题的模型建立

由于企业生产产品的工艺工序、原材料以及生产线等方面的差异,不同企业流水线的调度有不一样的要求,但一般情况下都可通过目标函数来衡量所选用的调度方式的优劣性,常见的目标函数主要包括时间最优策略、生产线损耗率最低策略和能量最优策略等,本文采用时间最优策略作为目标进行智能生产线的调度规划。

调度问题的目标函数  $f$  可以表示为:

$$f = \min\{\max[T_k(o_i)]\}, 1 \leq i \leq n, 1 \leq k \leq 5 \quad (1)$$

其中,  $i$  为工件数;  $k$  为工序数;  $T_k(o_i)$  为耗费的时间。

对于目标函数加入以下约束:

(1)首工序时间,即工件  $i$  从物流中心到首工序  $o_{i1}$  所在的工作站:

$$T_1(o_i) \geq \text{mov}(B_1, o_{i1}) \quad (2)$$

其中,  $\text{mov}(B_1, o_{i1})$  表示将工件  $i$  从物流中心搬运至工序所在工作站所耗费时间,  $B_1$  表示物流中心的装载区域。

(2)预处理时间,包括在前处理区域等待和进行前处理的时间:

$$T_2(o_i) \geq \text{pre}(o_{ij}) + \text{wt}_1(o_{ij}) \quad (3)$$

其中,  $\text{pre}(o_{ij})$  表示工件  $i$  进行第  $j$  道工序的预处理时间,  $\text{wt}_1(o_{ij})$  表示在对应工序的前处理区域的等待时间。

(3)加工处理时间,包括工件进行加工,然后传送到后处

理区域进行后处理的时间:

$$T_3(o_{ij}) \geq M(o_{ij}) + \text{add}(o_{ij}) + \text{wt}_2(o_{ij}) \quad (4)$$

其中,  $M(o_{ij})$  表示工件  $i$  进行第  $j$  道工序的加工时间,  $\text{add}(o_{ij})$  为后处理时间,  $\text{wt}_2(o_{ij})$  为在后处理区域的等待时间。

(4)搬运间隙时间,即完成上一搬运任务后,按指令空载到下一搬运点:

$$T_4(o_i) \geq \text{mov}(o_{ij}, o_{ij+1}) \quad (5)$$

(5)末工序时间,即完成所有工序之后,将工件运回物流中心:

$$T_5(o_i) \geq \text{mov}(o_{ij}, B_2) \quad (6)$$

其中,  $B_2$  表示物流中心的卸载区域。

## 3 莱维萤火虫算法

### 3.1 萤火虫优化算法原理

萤火虫算法是仿生优化算法中的一种,经过多次迭代后萤火虫聚集在荧光素最高的萤火虫附近,其基本原理是<sup>[17-18]</sup>:

(1)在求解范围内随机投掷  $N$  个萤火虫个体,赋予每个个体荧光素  $l_i(t)$ ;

(2)设定每个萤火虫的感知半径  $r_d^i (0 < r_d^i < r_s)$ ,将感知半径内荧光素大于本身的个体放入邻域集合  $N_i(t)$ ;

(3)非邻域集合内的个体按照概率  $p_{ij}(t)$  向邻域集合  $N_i(t)$  中所筛选的较优个体移动,并更新  $J(x_i(t))$  所对应的值,重复上述步骤以得到问题最优解。

感知半径更新方程:

$$r_d^i(t+1) = \min\{r_s, \max[0, r_d^i(t) + \beta(n_i - |N_i(t)|)]\} \quad (7)$$

位置更新方程:

$$x_i(t+1) = x_i(t) + s \left( \frac{x_j(t) - x_i(t)}{\|x_j(t) - x_i(t)\|} \right) \quad (8)$$

概率分布方程更新方程:

$$P_{ij}(t) = \frac{l_j(t) - l_i(t)}{\sum_{k \in N_i(t)} l_k(t) - l_i(t)} \quad (9)$$

邻域集合更新方程:

$$N_i(t) = \{j; d_{ij}(t) < r_d^i(t); l_i(t) < l_j(t)\} \quad (10)$$

荧光素值更新方程:

$$l_i(t+1) = (1 - \rho)l_i(t) + \gamma * J[x_i(t+1)] \quad (11)$$

其中,  $r_s$  为个体的感知半径上限;  $\beta$  为半径时变系数;  $n_i$  为萤火虫个体邻域上限;  $N_i(t)$  为个体  $i$  的邻域集合;  $s$  为个体的固定步长;  $\rho$  为荧光素发散系数,  $\rho \in (0, 1)$ ;  $d_{ij}(t)$  为个体  $i$  和个体  $j$  之间的距离;  $\gamma$  为用于调节函数值的系数,用于调整函数  $J[x_i(t+1)]$ 。

通过分析可知,若萤火虫个体  $i$  的邻域中不存在荧光素值更大的个体  $j$ ,则个体  $i$  将停止搜索,导致算法收敛速度减慢,不利于最优解的搜寻。

### 3.2 莱维飞行步长策略

莱维飞行随机游动是服从莱维分布的随机搜索策略,其可以有效增加种群的多样性,扩大搜索范围,更容易跳出局部最优解<sup>[19]</sup>。为了增强萤火虫算法的计算能力,防止算法陷入局部最优解问题,本文引入莱维飞行策略代替原始萤火虫的固定步长对萤火虫算法进行改进。

算法的核心思想是用莱维飞行的概率分布,提高萤火虫搜索步伐的随机性,增强萤火虫种群的多样性,扩大了搜索范围,提高了单只萤火虫的搜索能力,同时解决了传统萤火虫算法易

陷入到局部最优解的问题。莱维飞行化简后的数学描述为：

$$Levy(s, \gamma, \mu) = \begin{cases} \sqrt{\frac{\gamma}{2\pi}} \exp\left[-\frac{\gamma}{2(s-\mu)}\right] \cdot \frac{\gamma}{(s-\mu)^{\frac{3}{2}}}, & 0 < \mu < s \\ 0, & s \leq 0 \end{cases} \quad (12)$$

加入莱维飞行后的萤火虫位置更新公式变为：

$$x_i(t+1) = x_i(t) + \left( \frac{x_j(t) - x_i(t)}{\|x_j(t) - x_i(t)\|} \right) \oplus Levy(\lambda) \quad (13)$$

其中,  $\oplus$  是点对点乘法,  $Levy(\lambda)$  是步径随机化函数, 满足莱维分布：

$$Levy(\lambda) \sim u = t^{-\lambda}, 1 < \lambda \leq 3 \quad (14)$$

因此, 可得到改进后步长的计算公式：

$$s = \frac{\mu}{|v|^{\frac{1}{\beta}}} \quad (15)$$

其中,  $\mu, v$  服从正态分布,  $\mu \sim N(0, \sigma_\mu^2), v \sim N(0, \sigma_v^2)$ 。

$$\sigma_\mu = \left[ \frac{\Gamma(1+\beta) \cdot \sin \frac{\pi\beta}{2}}{\Gamma(\frac{1+\beta}{2}) \cdot \beta \cdot 2^{(\beta-1)/2}} \right] \quad (16)$$

其中,  $\beta$  皆取 1.5, 且在工程应用中为方便计算, 常对该正态分布进行简化<sup>[20]</sup>：

$$\mu \sim N(0, \sigma_\mu^2) = N(0, 0.6966^2)$$

$$v \sim N(0, \sigma_v^2) = N(0, 1)$$

### 3.3 种群优化

荧光素的有效性是引起萤火虫算法陷入局部最优问题的重要原因, 传统萤火虫算法通过将  $\gamma$  作为用于调节函数值的常量, 通过权重分配对荧光素进行更新：

$$l_i(t+1) = (1-\rho)l_i(t) + e^{\frac{l_{\min}}{l_{\max}}} * J[x_i(t+1)] \quad (17)$$

其中,  $l_{\min}$  和  $l_{\max}$  分别为  $t$  时刻个体  $i$  搜索范围内的最小和最大亮度。利用最大和最小亮度代替  $\gamma$ , 减少偶然性误差对种群质量的影响。

此外, 引入立方映射对种群进行优化, 其表达式为：

$$\begin{cases} y(n+1) = 4y(n)^3 - 3y(n) \\ -1 \leq y(n) \leq 1 \end{cases} \quad (18)$$

然后对种群个体分别进行混沌映射：

$$x_{id} = L_d + (1 + y_{id}) \cdot \frac{U_d - L_d}{2} \quad (19)$$

其中,  $U_d$  和  $L_d$  表示种群中个体运动的上下限,  $y_{id}$  表示个体在  $d$  维空间中的映射位置,  $x_{id}$  表示个体进行混沌优化之前的位置。

## 4 算法性能测试

为验证 LGSO 算法的有效性, 选择同类型智能算法进行对比, 选取文献[21]中的基准测试函数分布进行对比测试。初始化各算法的参数, 如表 1 所列。

表 1 LGSO 与 SGSO、GSO 算法的初始化参数

Table 1 Initialization parameters of LGSO, SGSO and GSO

parameters	$\gamma$	$N$	$O_i$	$n_t$	$t_{\max}$	$\beta$	$l_0$	$s_0$	$d_0$	$d_{\max}$	$n$	$k$
value	0.6	100	5	5	500	0.08	5	0.05	5	10	4	10

目标函数为：

$$f_1 = (|x_1| - 5)^2 + (|x_2| + 5)^2, x_i \in [-10, 10], i = 1, 2$$

计算获得的最优解为 25。

$$f_2 = \left( x_2 - \frac{5.1}{4\pi^2} x_1^2 + \frac{5x_1}{\pi} - 6 \right)^2 + 10 \left( 1 - \frac{1}{8\pi} \right) \cos x_1 + 10,$$

$$x_1 \in [-5, 10], x_2 \in [0, 15]$$

计算获得的最优解为 0.397898。

$$f_3 = 0.5 + \frac{\sin^2 \sqrt{x_1^2 + x_2^2} - 0.5}{[1.0 + 0.001(x_1^2 + x_2^2)]^2}, x_1 \in [-5, 10],$$

$$x_2 \in [0, 15]$$

计算获得的最优解为 0。

$$f_4 = 100(x_2 - x_1)^2 + [6.4(x_2 - 0.5)^2 - x_1 - 0.6]^2,$$

$$x_i \in [-5, 5], i = 1, 2$$

计算获得的最优解为 0。

为了保证实验结果的可行度和一般性, 分别通过 LGSO 算法和 SGSO 算法、CGSO 算法、GSO 算法对上述测试函数进行 100 次运行, 并提取出结果中的各项值, SGSO 算法的种群分布如图 1 所示, 收敛结果对比如图 2 所示, LGSO GSO, SGSO, CGSO 计算结果对比如表 1 所列。

由图 1 可以看出, LGSO 算法中萤火虫种群在最优解的正下方聚集度更高, 证明莱维飞行对萤火虫种群优化的有效性, 可以避免局部最优或者萤火虫迷路等问题的出现。

由图 2 可以看出, LGSO 算法的收敛速度明显快于 GSO 算法, 且最终收敛值更接近于测试函数的最优解, 说明改进算法的计算准确度得到提高。

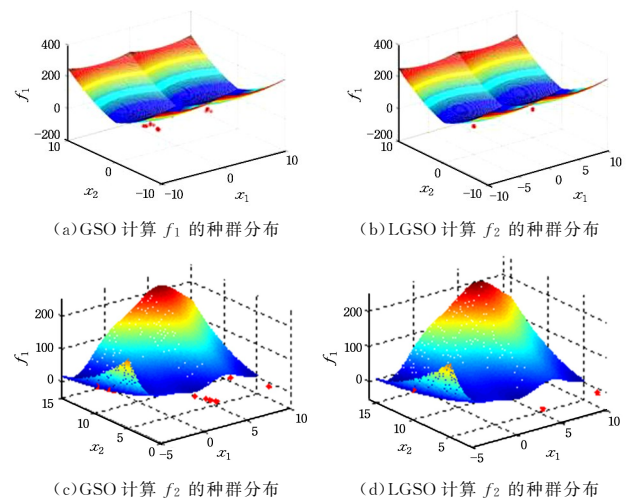


图 1 种群分布对比

Fig. 1 Population distribution comparison map

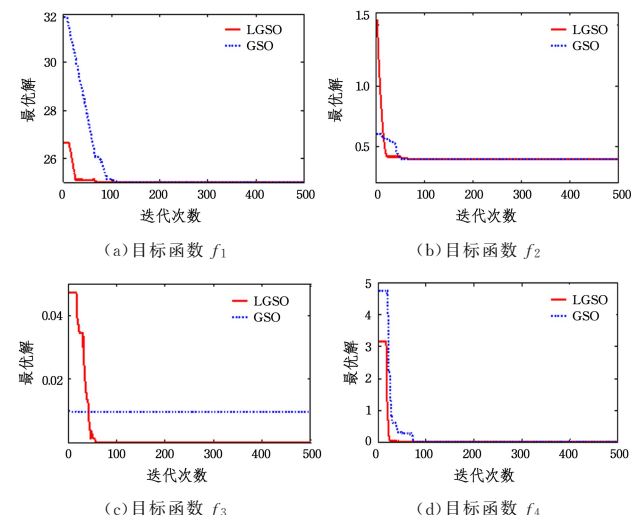


图 2 LGSO 和 GSO 计算结果对比

Fig. 2 Comparison of LGSO and GSO calculation results

表2 LGSO与GSO,SGSO,CGSO的计算结果

Table 2 Comparison of LGSO,GSO,SGSO and CGSO calculation results

目标函数	算法	最差值	最优值	平均值	标准偏差
$f_1$	GSO	28.7196526580	25.0056549371	25.9533696084	1.0061696844
	CGSO	25.0927584310	25.0000124222	25.0114703090	0.0229016010
	SGSO	25.0001283894	25.0000045123	25.0000341724	0.0000333571
	LGSO	25.0000961354	25.0000044645	25.0000131654	0.0000114984
$f_2$	GSO	0.4264597311	0.3660518946	0.3969567561	0.0121966388
	CGSO	0.3984582020	0.3975618095	0.3978958301	0.0001739578
	SGSO	0.3978873972	0.3978935573	0.3978897476	$1.685610 \times 10^{-3}$
	LGSO	0.3974864978	0.3978964657	0.3974987972	$1.453677 \times 10^{-3}$
$f_3$	GSO	$9.719468 \times 10^{-3}$	$7.323000 \times 10^{-7}$	$6.468320 \times 10^{-3}$	$3.945487 \times 10^{-3}$
	CGSO	$9.715969 \times 10^{-3}$	$1.042800 \times 10^{-6}$	$1.509961 \times 10^{-3}$	$2.534505 \times 10^{-3}$
	SGSO	$9.612232 \times 10^{-3}$	$2.024430 \times 10^{-5}$	$1.287434 \times 10^{-3}$	$1.934834 \times 10^{-3}$
	LGSO	$7.685457 \times 10^{-3}$	$3.053379 \times 10^{-6}$	$6.924497 \times 10^{-5}$	$2.497196 \times 10^{-4}$
$f_4$	GSO	$3.274988 \times 10^{-1}$	$8.202069 \times 10^{-6}$	$6.464649 \times 10^{-3}$	$9.416946 \times 10^{-3}$
	CGSO	$5.299788 \times 10^{-2}$	$4.500000 \times 10^{-9}$	$9.298425 \times 10^{-3}$	$1.294901 \times 10^{-2}$
	SGSO	$5.271268 \times 10^{-5}$	$9.358547 \times 10^{-7}$	$4.702712 \times 10^{-5}$	$2.436587 \times 10^{-3}$
	LGSO	$6.464973 \times 10^{-5}$	$1.238558 \times 10^{-6}$	$4.834116 \times 10^{-5}$	$4.523526 \times 10^{-3}$

通过分析表1可知,利用LGSO算法所求的目标函数解的平均值均优于GSO,SGSO,CGSO3种算法,说明LGSO算法具有相对更好的求解精度。最优计算结果对比中,利用LGSO算法计算获得的 $f_3$ 的最优解没有GSO的计算精确度高,但其最差值的精度比GSO高,且计算结果的标准偏差值更小。相比GSO,SGSO,CGSO3种算法,LGSO算法计算得到的结果的标准偏差值均低于其他3种算法,说明改进的算法具有更好的稳定性。

## 5 LGSO算法在调度问题中的应用

为验证LGSO算法对调度问题的有效性,对智能生产线的4种分布进行优化分析,如图3所示,包括两台物流运输模块、4个执行模块(M1—M4)和装载及卸载站(L/U)。分别使用GSO,SGSO和LGSO3种算法对目标函数进行100次最优求解。实验结果如表3所列。

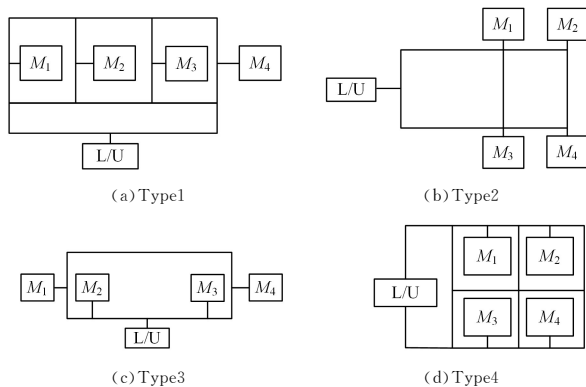


图3 布置类型

Fig. 3 Configuration type

通过分析表3中的实验结果可以发现,对于4种经典分布,分布2最省时,分布4最耗时。

LGSO算法计算得到的最差值、最优值、结果平均值以及标准偏差总体上基本小于其他两种算法。在分布方案4中,相比GSO,SGSO算法,LGSO算法计算得到的最差值相比GSO,SGSO算法分别减小了17.0%,0.29%;最优值相比GSO,SGSO算法分别减小了0.95%,0.2%;平均值相比GSO,SGSO算法分别减小了1.6%,0.35%,说明在复杂的路径下LGSO算法计算结果更优。LGSO算法计算结果的标准

差相比GSO,SGSO算法大幅降低,说明基于LGSO算法的单次结果具有较高的可信度和稳定性。在分布方案2中,相比GSO,SGSO算法,LGSO算法计算得到的最差值相比GSO,SGSO算法分别减小了11.23%,0.27%;最优值相比GSO,SGSO算法分别减小了0.79%,0.46%;平均值相比GSO,SGSO算法分别减小了1.07%,0.47%,标准偏差相比GSO,SGSO算法大幅减小,说明在简单路径下LGSO算法相对GSO算法的计算结果更加精确且稳定可靠,但相对于SGSO算法提升幅度较小。

表3 GSO算法与SGSO的计算结果

Table 3 Comparison of GSO and SGSO calculation results

布置类型	算法	最差值	最优值	平均值	标准偏差
分布1	GSO	108.5674	99.1654	99.8467	2.0469
	SGSO	100.8967	98.5976	98.9667	0.7467
	LGSO	99.1976	98.4672	98.8461	0.6496
分布2	GSO	98.6345	88.7641	89.1467	0.8794
	SGSO	88.9167	88.4674	88.6149	0.4976
	LGSO	88.6746	88.0647	88.2034	0.1647
分布3	GSO	109.4160	95.9634	96.4676	1.0349
	SGSO	96.1674	95.9015	96.0246	0.3974
	LGSO	96.1676	95.8464	95.9846	0.0976
分布4	GSO	128.1644	110.1682	110.9133	1.7467
	SGSO	110.1456	109.3548	109.5497	0.3648
	LGSO	109.8263	109.1349	109.1649	0.0679

**结束语** (1)通过分析智能生产线的调度过程建立了调度过程的数学模型,确定了以时间最优策略为目标函数,并设定了约束条件。

(2)提出了一种基于莱维萤火虫算法,采用莱维飞行提高萤火虫移动步长,改善种群的有效分布质量。引入最大和最小荧光素作为边界约束以优化荧光素迭代公式,提高个体所携带荧光素的合理性,引入立方映射实现对种群的进一步优化,提高种群的搜索能力。通过对测试函数和经典分布进行仿真测试可知:改进的LGSO相比GSO,SGSO和CGSO3种算法呈现出了更好的求解精度、收敛速度以及稳定性。

(3)将LGSO用于解决所提出的调度问题,并与GSO算法和SGSO算法进行对比,在复杂分布方案中,相比GSO和SGSO算法,LGSO算法具有更好的计算能力,计算结果可信度更高,稳定性更好。简单路径下LGSO算法相对于GSO算法的计算结果更加精确且稳定可靠,但相对于SG-

SO 算法提升幅度较小。

## 参 考 文 献

- [1] LIU Y Z, LI M G, DU H. Robust  $H-\infty$  control of NCS with delay and packet dropout[J]. Control and Decision, 2014, 29(3): 517-522.
- [2] AHMADI E, ZANDIEH M, FARROKH M, et al. A multi objective optimization approach for flexible job shop scheduling problem under random machine breakdown by evolutionary algorithms[J]. Computers & Operations Research, 2016, 73: 56-66.
- [3] FANG D J, HUANG Z Y. Method of parallel optimization of facility layout and production scheduling for reconfigurable hybrid flow production line[J]. Logistics Technology, 2019, 38(7): 117-122.
- [4] EBRAHIMPOUR V, NAJJARBASHI A, SHEIKHALISHAHI M. Multi-objective modeling for preventive maintenance scheduling in a multiple production line[J]. Journal of Intelligent Manufacturing, 2013, 26(1): 1-12.
- [5] FLETCHER M, DEEN S M. Task rescheduling in multi-agent manufacturing[C]// Tenth International Workshop on Database and Expert Systems Applications. IEEE Computer Society, 1999.
- [6] LU Z Q, HU X M, ZHU H W. Dynamic scheduling method for aircraft moving assembly line under uncertain supply of material [J]. Journal of Tongji University (Natural Science), 2019, 47(5): 723-730, 738.
- [7] MAO Y N, TANG Q H, ZHANG L P, et al. Mathematic model for automated mixed production line scheduling problem with multi-capacity machines[J]. Computer Integrated Manufacturing Systems, 2019, 25(7): 1717-1728.
- [8] LI H, ZHANG Y, YE Q H, et al. Research on flexible production line scheduling with lot streaming and setup times [J]. Industrial Engineering and Management, 2020, 25(3): 179-187.
- [9] JIA Z H, CHEN H P, SUN Y H. Hybrid particle swarm optimization for flexible job-shop scheduling [J]. Journal of System Simulation, 2007(20): 4743-4747.
- [10] ŠUCHA B L, ZDENĚK P H. Solving the Resource Constrained Project Scheduling Problem using the parallel Tabu Search designed for the CUDA platform[M]. Academic Press, Inc. 2015.
- [11] THOMAS P R, SALHI S. A Tabu Search Approach for the Resource Constrained Project Scheduling Problem[J]. Journal of Heuristics, 1998, 4(2): 123-139.
- [12] ZDENĚK H, SUCHA P. Time symmetry of resource constrained project scheduling with general temporal constraints and take-give resources[J]. Annals of Operations Research, 2016, 248: 209-217.
- [13] ZENG B, LI M F, ZHANG Y, et al. Research on assembly sequence planning based on firefly algorithm[J]. Journal of Mechanical Engineering, 2013, 49(11): 177-184.
- [14] HUANG Q, ZHOU Q H, ZHANG Q, et al. Layout optimization of dip dyeing workshop based on system layout planning-genetic algorithm[J]. Journal of Textile Research, 2020, 41(3): 84-90.
- [15] WU Q D, MA Y M, LI L, et al. Data-driven dynamic scheduling method for semiconductor production line[J]. Control Theory & Applications, 2015, 32(9): 1233- 1239.
- [16] ZHU G Y, XU W J. Multi-objective flexible job shop scheduling method for machine tool component production line considering energy consumption and quality [J]. Control and Decision, 2019, 34(2): 252-260.
- [17] LUO T H, LIANG S, HE Z Y, et al. Path planning of robot based on glowworm swarm optimization algorithm of scene understanding[J]. Journal of Computer Applications, 2017, 37(12): 3608-3613.
- [18] HAO X Y, HE X S, XUE J J. A cuckoo initialization of firefly algorithm and its engineering application[J]. Computer Technology and Development, 2018, 28(12): 167-170, 175.
- [19] NIU H F, SONG W P, NING A P. Hybrid search algorithm on particle swarm optimization and levy flight [J]. Journal of Taiyuan University of Science and Technology, 2016, 37(1): 6-11.
- [20] WANG X W, YAN Y X, GU X S. Welding robot path planning based on Levy-PSO[J]. Control and Decision, 2017, 32(2): 373-377.
- [21] YU S H, SU S B. Research and application of chaotic glowworm swarm optimization algorithm[J]. Journal of Frontiers of Computer Science & Technology, 2014, 8(3): 352-358.



**ZHANG Ju**, born in 1988, master, lecturer. Her main research interests include mechanical dynamics and intelligent control.