

6-25-2020

## A Discrete Krill Herd Algorithm for the No-wait Flow Shop Scheduling Problem

Changping Liu

*1. Faculty of Management Engineering, Huaiyin Institute of Technology, Huaian 223200, China;;2. Department of Industrial Engineering and Engineering Management, National Tsing Hua University, Hsinchu 30013, China;;3. Jiangsu Smart Factory Engineering Research Center, Huaian 223003, China;*

Zhenfu Jian

*2. Department of Industrial Engineering and Engineering Management, National Tsing Hua University, Hsinchu 30013, China;;*

Wenhan Fu

*2. Department of Industrial Engineering and Engineering Management, National Tsing Hua University, Hsinchu 30013, China;;*

Follow this and additional works at: <https://dc-china-simulation.researchcommons.org/journal>



Part of the Artificial Intelligence and Robotics Commons, Computer Engineering Commons, Numerical Analysis and Scientific Computing Commons, Operations Research, Systems Engineering and Industrial Engineering Commons, and the Systems Science Commons

---

This Paper is brought to you for free and open access by Journal of System Simulation. It has been accepted for inclusion in Journal of System Simulation by an authorized editor of Journal of System Simulation.

---

# A Discrete Krill Herd Algorithm for the No-wait Flow Shop Scheduling Problem

## Abstract

**Abstract:** A discrete krill herd algorithm (DKH) is proposed to solve the no-wait flow shop scheduling problem (NWFSP) with the makespan criterion. Based on the characteristics of the optimization mechanism, the notion of “virtual food” in krill herd algorithm and the influenced crossover rules between krill individuals are defined. The concept and calculation method of the individual distance are redefined to measure the neighborhood range to realize the information exchange. The local search strategies of random swap, inverse and insert operator are employed to enhance the local search ability of DKH, and the multiple initialization strategies are redesigned to promote the qualities and diversities of the initial populations. Simulation results on the benchmark instances shows the effectiveness and robustness of the DKH.

## Keywords

discrete krill herd optimization, optimization mechanism, no-wait flow shop scheduling, makespan

## Recommended Citation

Liu Changping, Jian Zhenfu, Fu Wenhan. A Discrete Krill Herd Algorithm for the No-wait Flow Shop Scheduling Problem[J]. Journal of System Simulation, 2020, 32(6): 1051-1059.

## 求解零等待流水线调度问题的离散磷虾群算法

刘长平<sup>1,2,3</sup>, 简祯富<sup>2</sup>, 傅文翰<sup>2</sup>(1. 淮阴工学院 管理工程学院, 江苏 淮安 223200; 2. 清华大学 工业工程与工程管理系, 台湾 新竹 30013;  
3. 江苏省智能工厂工程研究中心, 江苏 淮安 223003)

**摘要:** 针对最小化最大完工时间零等待流水车间调度问题, 设计了一种离散磷虾群优化算法进行求解。根据优化问题特性, 定义了算法中虚拟食物的计算方法, 规定了虚拟食物影响下磷虾个体之间的交叉规则; 提出了个体间的距离计算方式, 从而可以界定邻域范围实现个体信息交流; 进一步采用交换、逆序、插入等策略来提高算法的局部搜索能力, 同时借助多种初始化策略来提高初始种群的质量和多样性。通过典型算例的仿真测试和对比, 验证了所设计算法的有效性和鲁棒性。

**关键词:** 离散磷虾群优化算法; 优化机理; 零等待流水车间调度; 最小化完工时间

中图分类号: TP301.6; N945 文献标识码: A 文章编号: 1004-731X (2020) 06-1051-09  
DOI: 10.16182/j.issn1004731x.joss.18-0762

## A Discrete Krill Herd Algorithm for the No-wait Flow Shop Scheduling Problem

Liu Changping<sup>1,2,3</sup>, Jian Zhenfu<sup>2</sup>, Fu Wenhan<sup>2</sup>(1. Faculty of Management Engineering, Huaiyin Institute of Technology, Huaian 223200, China;  
(2. Department of Industrial Engineering and Engineering Management, National Tsing Hua University, Hsinchu 30013, China;  
3. Jiangsu Smart Factory Engineering Research Center, Huaian 223003, China)

**Abstract:** A discrete krill herd algorithm (DKH) is proposed to solve the no-wait flow shop scheduling problem (NWFSP) with the makespan criterion. Based on the characteristics of the optimization mechanism, the notion of “virtual food” in krill herd algorithm and the influenced crossover rules between krill individuals are defined. The concept and calculation method of the individual distance are redefined to measure the neighborhood range to realize the information exchange. The local search strategies of random swap, inverse and insert operator are employed to enhance the local search ability of DKH, and the multiple initialization strategies are redesigned to promote the qualities and diversities of the initial populations. Simulation results on the benchmark instances shows the effectiveness and robustness of the DKH.

**Keywords:** discrete krill herd optimization; optimization mechanism; no-wait flow shop scheduling; makespan

## 引言

在制药、化工、冶炼、轧钢、半导体测试等生

产过程中, 由于工艺、设备等因素的限制, 工件一旦开始加工便不允许中断, 直至加工结束, 具有这种工艺约束的生产过程被称作零等待流水车间调度问题(No-Wait Flow-Shop Problem, NWFSP)。三机及以上的 NWFSP 属于强 NP 难问题<sup>[1]</sup>, 对 NWFSP 的优化一直是学术界和工程界致力于研究的典型调度问题。

由于 NWFSP 具有 NP 难特性, 目前尚无具有



收稿日期: 2018-11-16 修回日期: 2019-05-18;  
基金项目: 国家自然科学基金(71761027), 教育部人文社会科学规划基金(16YJA630032);  
作者简介: 刘长平(1974-), 男, 河南洛阳, 博士, 副教授, 研究方向为智能优化算法; 简祯富(1966-), 男, 台湾新竹, 博士, 教授, 博导, 研究方向为智能制造, 数据挖掘。

<http://www.china-simulation.com>

• 1051 •

多项式计算复杂度的全局优化算法,对于较大规模的该类问题,通常采用启发式算法或群智能算法来求解。如文献[2]提出了 2 种迭代启发式算法,通过块插入以及成对交换进行多次迭代来提高求解精度;文献[3]在 NEH 启发式算法基础上,按照工件在机器上加工时间的标准差进行非增排序,提出了标准差启发式算法用于求解最小化总流水时间的 NWFSP,测试表明效果优于 NEH 算法;文献[4]基于插入-分段的邻域结构,通过分析基本操作的目标增量性质,提出了一种复合启发式算法。启发式算法原理简单、求解速度快,适用于实时性要求较高的场合,但求解精度不高、性能不够稳定。受自然规律和生物群体智能行为的启发,提出了一些新颖的群智能算法,这类算法能够在可接受时间内以一定概率获得较高质量的解,在生产调度领域得到了广泛应用。如文献[5]将粒子群算法与文化基因算法相结合用于求解总流程时间最小的 NWFSP;文献[6]提出了一种离散和声搜索算法,结合关键路径和机器初始分配规则用于求解多目标柔性车间调度问题;文献[7]提出一种人工蜂群算法,嵌入局部搜索策略求解具有零等待约束的车间调度问题。群智能算法计算复杂度通常随问题规模的增加而大幅增加,为提高计算效率有学者将其与启发式算法相结合,如文献[8]将量子进化机制嵌入萤火虫算法中,并与启发式邻域搜索方法相结合,用于求解最小化总完工时间的 NWFSP;文献[9]首先利用 PF-NEH 启发式算法构造优势种群,再以离散果蝇算法机制来优化 NWFSP,也取得了不错的效果。

磷虾群优化算法<sup>[10]</sup>(Krill Herd Algorithm, KH)是通过模拟磷虾群体觅食过程等行为演变而来的一种群智能优化算法,目前多用在函数优化领域<sup>[11-13]</sup>,组合优化领域的应用还较少见诸文献。本文在分析磷虾群算法优化机理的基础上,针对 NWFSP 特性,提出了一种离散磷虾群算法,采用基于工件序列的编码方式和新的个体距离计算方法来扩展传统磷虾群优化算法的使用范畴,测试结

果验证了所提算法求解 NWFSP 的可行性和优越性,是求解该类问题的一种有效方法。

## 1 零等待流水车间调度问题模型

最小化最大完工时间的零等待流水车间调度问题记为  $F_m | nwt | C_{\max}$ ,问题可描述为:有工件集  $\{j_i | i=1,2,\dots,n\}$  和机器集  $\{k_l | l=1,2,\dots,m\}$ ,用  $t_{j_i,k}$  表示工件  $j_i$  在机器  $k$  的加工时间,  $C(j_i,k)$  表示在机器  $k$  的完工时间,加工过程中满足以下条件:

- (1) 每台机器同时只能加工一个工件;
- (2) 一个工件不能同时在不同的机器上加工;
- (3) 工序准备时间与顺序无关,且包含在加工时间内;
- (4) 工件在每台机器上加工顺序相同且确定;
- (5) 工件不允许停留在机器间的缓冲区内,允许机器在工件未到达时闲置。

假设工件在机器上的加工顺序为  $1\sim m$ ,  $\pi$  表示所有工件的某种加工排序,则  $n$  个工件在  $m$  台机器上的完工时间可以通过式(1)~(4)计算。

$$C(j_1,k)=\sum_{r=1}^k t_{j_1,r} \quad k=1,2,\dots,m \quad (1)$$

$$C(j_i,k)=C(j_{i-1},1)+\sum_{r=1}^k t_{j_i,r}+\Delta(j_i) \quad i=2,\dots,n; k=1,2,\dots,m \quad (2)$$

$$\Delta(j_i)=\max_{2\leq i\leq n} \left\{ \max_{2\leq k\leq m} [C(j_{i-1},k)-C(j_{i-1},1)-\sum_{r=1}^{k-1} t_{j_i,r}], 0 \right\} \quad (3)$$

$$C_{\max}(\pi)=C(j_n,m) \quad (4)$$

式(1)为工件  $j_1$  在机器  $k$  的完工时间;式(2)为工件  $j_i(i>1)$  在机器  $k$  的完工时间;式(3)为由于不允许工件在机器间的缓冲区停留,工件  $j_i$  必要时需要延迟的开始加工时间;式(4)中  $C_{\max}(\pi)$  表示  $n$  个工件的最大完工时间,也是优化目标。

## 2 磷虾群算法优化机理

为了在海洋环境中生存,磷虾以群体形式进行活动,觅食过程中磷虾个体受其他个体和可能的食物位置的影响来调整自身位置并逐渐接近食物,同

时保持群体密度来降低被捕食的几率。磷虾群算法是基于对磷虾生活习性的模拟演变来的一种群智能优化算法, 公式(5)~(10)体现了对磷虾群生活习性的模拟, 构成算法的主体。

算法在解空间内随机初始化种群来确定磷虾个体的初始位置, 位置和问题目标函数相关联, 通过个体适应度来表示优劣。个体位置更新如式(5)所示:

$$\mathbf{x}_i^{t+1} = \mathbf{x}_i^t + \Delta t \times \frac{d\mathbf{x}_i}{dt} \quad (5)$$

式中:  $\mathbf{x}_i^{t+1}$ ,  $\mathbf{x}_i^t$  分别为个体  $i$  在  $t+1$  和  $t$  时刻的位置;  $\Delta t$  为  $t$  时刻速度向量缩放因子, 根据公式(6)计算。

$$\Delta t = C_t \times \sum_{j=1}^{NV} (UB_j - LB_j) \quad (6)$$

式中:  $C_t$  为时间常数, 取值为[0,2];  $UB_j$ ,  $LB_j$  为变量第  $j$  维搜索空间的上下界;  $NV$  为变量维数。

$d\mathbf{x}_i/dt$  为磷虾个体的移动速度, 如式(7)所示。

$$\frac{d\mathbf{x}_i}{dt} = \mathbf{N}_i + \mathbf{F}_i + \mathbf{D}_i \quad (7)$$

式(7)由诱导运动向量  $\mathbf{N}_i$ , 觅食运动向量  $\mathbf{F}_i$  和随机扩散向量  $\mathbf{D}_i$  三部分组成, 分别由式(8)~(10)来计算。

$$\mathbf{N}_i^{t+1} = \mathbf{N}_{\max} (\partial_i^{local} + \partial_i^{gbest}) + \omega_n \mathbf{N}_i^t \quad (8)$$

$$\mathbf{F}_i^{t+1} = V_f (\beta_i^{food} + \beta_i^{pbest}) + \omega_f \mathbf{F}_i^t \quad (9)$$

$$\mathbf{D}_i^{t+1} = \mathbf{D}_{\max} (1 - t/t_{\max}) \delta \quad (10)$$

式(8)中:  $\mathbf{N}_i^{t+1}$ ,  $\mathbf{N}_i^t$  为第  $t+1$  次、 $t$  次搜索时的诱导运动向量;  $\partial_i^{local}$ ,  $\partial_i^{gbest}$  分别为受邻域内其他个体影响和当前最佳个体影响计算出的运动向量;  $\mathbf{N}_{\max}$  为最大诱导速度因子, 一般取值为 0.01;  $\omega_n$  为诱导运动惯性权重, 取 [0, 1] 的常数。

式(9)用来计算虚拟食物对磷虾个体运动的影响, 式中  $\mathbf{F}_i^{t+1}$ ,  $\mathbf{F}_i^t$  为第  $t+1$  次、 $t$  次搜索时的觅食运动向量;  $\beta_i^{food}$ ,  $\beta_i^{pbest}$  分别为根据当前估计的食物位置和个体历史最佳觅食位置计算出的吸引度值;  $V_f$  为觅食速度因子,  $\omega_f$  为觅食运动惯性权重, 取值为 [0, 1] 区间的常数。

式(10)描述磷虾个体的随机运动, 式中  $\mathbf{D}_i^{t+1}$  表

示第  $t+1$  次搜索时磷虾个体  $i$  的随机扩散向量;  $\beta_i^{food}$ ,  $\beta_i^{pbest}$  分别为根据当前估计的食物位置和个体历史最佳觅食位置计算出的吸引度值;  $\mathbf{D}_{\max}$  为磷虾个体最大扩散速度因子;  $\delta$  为随机扩散方向向量, 取 [-1, 1] 的随机数;  $t_{\max}$  为最大搜索次数。

在优化过程中磷虾群算法通过  $\partial_i^{gbest}$ ,  $\beta_i^{pbest}$  因子实现与群体最优和个体历史最优信息的交流, 通过  $\partial_i^{local}$  因子实现局域信息交流, 通过  $\beta_i^{food}$  因子实现群体信息交流, 从而指导个体有目的地向食物移动, 达到优化目的。为了进一步提高算法的性能, 还可以在基本算法结构上增加个体交叉和变异环节, 具体策略可见文献[14]。

### 3 求解零等待流水车间调度问题的离散磷虾群优化算法

传统磷虾群优化算法是针对函数优化领域设计的, 个体位置采用连续值矢量编码, 在连续空间完成位置更新操作, 适用于连续论域中的优化问题。而 NWFSP 属于组合优化问题, 需要针对问题的离散性质, 设计相应的编码方式和位置更新方法, 基于磷虾群优化算法的机理, 设计了一种离散磷虾群优化算法来求解 NWFSP, 测试结果显示出了所提算法的优势。

#### 3.1 个体编码方式

在离散磷虾群算法中, 磷虾个体直接用基于工序序列的整数编码方式表示, 个体  $i$  的位置矢量  $\{\mathbf{X}_i | i=1, 2, \dots, n\}$  表示工件  $\{\pi_j | j=1, 2, \dots, m\}$  的一种加工排序, 从而建立起个体位置与加工序列的映射。这种编码方式简单易行, 而且产生的解均为可行解。例如, 如果个体位置矢量为  $\mathbf{X}_i = [4, 2, 3, 8, 1, 7, 5, 6]$ , 映射为工件的加工排序则为  $\pi_i = [j_4, j_2, j_3, j_8, j_1, j_7, j_5, j_6]$ , 从而可计算磷虾个体位置所对应调度解的适应度。

#### 3.2 个体距离计算方式

函数优化问题的磷虾群算法中个体之间距离采用欧氏空间距离来计算, 不适合用在离散磷虾群

优化算法中。根据要优化的问题特性,在离散磷虾群优化算法中个体位置矢量采用工件加工序列来表示,不同的位置矢量代表不同的加工排序。因此,在离散磷虾群优化算法中以工件加工排序的差异程度来表征“个体距离”,定义如下:

$$d_{ij} = \frac{\sum_{s=1}^n |x_{j,s} - x_{i,s}|}{N} \quad (11)$$

式中:  $x_{i,s}$ ,  $x_{j,s}$  为磷虾个体  $i$  和个体  $j$  第  $s$  维的编码;  $n$  为工件数;  $N$  为分子项取值上限,按照式(12)计算:

$$N = \begin{cases} (n^2 - 1)/2 & n \in \text{奇数} \\ n^2/2 & n \in \text{偶数} \end{cases} \quad (12)$$

从式(11)可知  $d_{ij} \in [0,1]$ ,“个体距离”的大小反映出加工顺序的差异程度,“距离”越远代表个体之间差异越大。

### 3.3 位置更新策略

个体位置更新是磷虾群优化算法中的核心环节,受虚拟食物位置公式(9)、邻域内个体公式(8)、随机扩散运动公式(10)共 3 部分的综合影响,根据 NWFSP 的特点,针对这 3 部分来设计离散磷虾群优化算法。

#### 1) 虚拟食物的影响

定义磷虾群体的平均适应度作为虚拟食物,通过将个体适应度与虚拟食物比较来进行位置更新,更新策略如式(13)所示:

$$x_i^{t+1} = \begin{cases} LOX(x_i^t \otimes x_{food}^t) & \text{if } f(i) < f(food) \\ PMX(x_i^t \otimes x^{gbest}) & \text{elseif } f(i) > f(food) \\ MU(x_i^t) & \text{else} \end{cases} \quad (13)$$

式中:  $f(i)$ ,  $f(food)$  分别为磷虾个体  $i$  和虚拟食物的适应度;  $t$  为当前搜索次数。更新过程为:在第  $t+1$  次搜索过程中,比较第  $i$  个个体和虚拟食物的适应度,如果  $f(i) < f(food)$ ,将个体位置  $x_i^t$  与  $x_{food}^t$  进行线性次序交叉操作(LOX)组成新个体位置  $x_i^{t+1}$ ; 否则,将个体位置  $x_i^t$  与当前种群最优个体  $x^{gbest}$  进行部分映射交叉操作(PMX)以组成新位置  $x_i^{t+1}$ ; 若

均不满足,则对当前个体  $x_i^t$  进行变异操作(MU)来产生新的位置  $x_i^{t+1}$ 。其中,线性次序交叉操作能够尽量保留个体间工序的相对位置和相对工序前端的绝对位置,部分映射交叉操作能够在一定程度上满足模式定理使最佳模式得以最大可能保留。

式(13)体现的位置更新思路是:在寻优过程中如果磷虾个体  $i$  的适应度劣于虚拟食物的适应度,则与虚拟食物进行交叉来获取虚拟食物的优良基因片段;如果个体  $i$  的适应度优于虚拟食物,则与当前群体最优进行交叉获取群体最优的优良基因片段;如果上述条件均不满足,说明当前磷虾个体  $i$  已经处于群体较好位置,则对个体  $i$  进行变异操作,防止陷入局部极值。在位置更新过程中,需要对新产生的编码进行检测,对不可行编码进行修复。

#### 2) 邻域内群体对个体的影响

根据 3.2 节磷虾个体间距离定义来确定邻域范围,通过与邻域内个体适应度的比较来进行位置更新,更新策略如式(14)所示:

$$x_i^{t+1} = \begin{cases} x_i^t \otimes x_{j \in ngb}^t & \text{if } f(x_i^t \otimes x_{j \in ngb}^t) > f(x_i^t) \\ x_i^t & \text{else} \end{cases} \quad (14)$$

式中:  $ngb$  为邻域范围;  $x_j^t$  为第  $t$  次搜索中邻域范围内个体  $j$  的空间位置。

#### 更新过程为:

在  $t+1$  轮搜索过程中,根据磷虾个体之间的距离公式(11)确定邻域范围,选择邻域内个体,个体  $x_i^t$  分别与邻域内被选择个体  $x_j^t$  进行交叉操作,选择其中最优秀个体作为新个体  $x_i^{t+1}$ ;如果邻域范围内没有符合条件的磷虾个体,个体  $x_i^t$  则与群体最优  $x^{gbest}$  进行交叉操作,形成新的个体。在实际操作中,为缩短搜索时间,可以选择符合条件的部分邻域个体(如 30%)进行交叉操作。其中,交叉操作分别从次序交叉、单位置次序交叉方式中随机选择,并且采用精英保留策略。

#### 3) 随机扩散运动的影响

每次寻优后,选取一定比例性能较差的个体重

新初始化来增加群体多样性, 使种群保持持续进化能力。

这样, 离散磷虾群优化算法的位置更新公式可以用式(15)表示:

$$\mathbf{x}_i^{t+1} = F(\mathbf{x}_i^t) \oplus N(\mathbf{x}_i^t) \oplus D(\mathbf{x}_i^t) \quad (15)$$

### 3.4 局部搜索策略

零等待流水线调度问题属于 NP-Hard 问题, 在搜索过程中算法容易陷入局部极值、过早收敛, 基于邻域结构的局部搜索策略可以有效改善算法的寻优能力, 在离散磷虾群优化算法中采用了随机交换、两两交换、逆序和插入等局部搜索策略来提高算法的局部搜索能力, 避免算法早熟。具体策略如下:

#### 1) 随机交换策略

令  $\Pi$  表示工件集,

$$\pi = \{j_i | i = 1, 2, \dots, l \dots m, \dots, n\} \in \Pi$$

表示所有工件的一个排序, 随机交换策略即随机选择  $\{j_l, j_m | l, m = 1, 2, \dots, n; l \neq m\} \in \pi$ , 交换  $j_l, j_m$ , 形成新的

$$\pi' = \{j_1, \dots, j_{l-1}, j_m, j_{l+1}, \dots, j_{m-1}, j_l, j_{m+1}, \dots, j_n\} \in \Pi。$$

#### 2) 两两交换策略

令  $\Pi$  表示工件集,

$$\pi = \{j_i | i = 1, \dots, l \dots m, \dots, n\} \in \Pi$$

表示所有工件的一个排序, 两两交换策略即对于任意的  $\pi$ , 将相邻工件  $j_m, j_{m+1}$  两两交换, 形成新的

$$\pi' = \{j_1, \dots, j_{m+1}, j_m, \dots, j_n\} \in \Pi。$$

#### 3) 逆序策略

令  $\Pi$  表示工件集,

$$\pi = \{j_i | i = 1, \dots, l \dots m, \dots, n\} \in \Pi$$

表示所有工件的一个排序, 逆序策略即随机选择  $\{j_l, j_m | l, m = 1, 2, \dots, n; l < m \text{ 且 } |m - l| > 2\} \in \pi$ , 将  $j_l$  与  $j_m$  间的工件逆序, 形成新的

$$\pi' = \{j_1, \dots, j_{l-1}, j_m, j_{m-1} \dots j_{l+1}, j_l, j_{m+1}, \dots, j_n\} \in \Pi。$$

#### 4) 插入策略

令  $\Pi$  表示工件集

$$\pi = \{j_i | i = 1, \dots, l \dots m, \dots, n\} \in \Pi$$

表示所有工件的一个排序, 插入策略即随机选择  $\{j_l, j_m | l, m = 1, 2, \dots, n; l \neq m \text{ 且 } |m - l| > 2\} \in \pi$ , 将  $j_l$  置于  $j_m$  前, 形成新的

$$\pi' = \{j_1, \dots, j_{l-1}, j_{l+1} \dots j_{m-1}, j_l, j_m, \dots, j_n\} \in \Pi。$$

综合考虑运算时间和搜索精度的平衡, 在每次迭代寻优过程中随机选择一种局部搜索策略, 对群体最优个体进行局部搜索, 若最优个体得以进化, 则更新最优个体。

### 3.5 初始化策略

#### 1) 基于构造方法的个体初始化

初始种群的优劣对于群智能优化算法有较大影响, 初始种群质量高可以加快算法的收敛速度, 在相对短的运算时间内找到较高质量的解。因此, 本文采用构造启发式方法来构造部分初始解, 然后用磷虾群算法的优化机制进行寻优。常用的构造启发式算法有 NEH<sup>[15]</sup>、CDS<sup>[16]</sup>、RAJ<sup>[17]</sup>等以及在此基础上改进的启发式算法, 区别在于采用了不同的启发式规则, 本文采用从 NEH、CDS、RAJ 三种启发式算法中随机选择一种来构造一个初始解, 其余初始解仍采用随机方法生成, 这样有利于在较短时间内找到更优解。

#### 2) 重新初始化策略

群智能优化算法在寻优过程中, 种群易受局部极值的吸引, 多样性不断减少, 从而早熟收敛。本文采取重新初始化策略来减少这种现象, 在种群进化过程中选取性能较劣的部分个体重新初始化来保持种群的多样性, 使得种群具备持续进化的能力。

### 3.6 离散磷虾群优化算法流程

综上所述, 求解 NWFSP 的离散磷虾群优化算法步骤如下:

step 1: 初始化算法基本参数: 种群数量  $m$ , 邻域内交叉个体比例  $n\%$ , 重新初始化个体比例  $a\%$ , 最大搜索次数  $MaxT$  或搜索精度  $\varepsilon$ ;

step 2: 按照 2.1 节编码方式初始化磷虾种群位置  $x_i(i=1,2, \dots, m)$ , 其中一个初始解随机采用 NEH、CDS、RAJ 算法构造, 其余随机化生成;

step 3: 计算磷虾个体适应度, 找出最佳个体;

step 4: 计算虚拟食物的适应度, 找出虚拟食物位置; 根据式(13)更新磷虾个体位置;

step 5: 计算邻域范围, 确定邻域内交叉个体; 根据式(14)更新磷虾个体位置;

step 6: 对磷虾群体进行评估, 找出当前最优个体及所处位置;

step 7: 按照 3.4 节局部搜索策略对当前最优个体进行优化; 按照 3.5 节初始化策略对  $a\%$  适应度最差的个体进行重新初始化;

step 8: 当满足搜索精度或达到最大搜索次数则转入 step 9, 否则转 step 4, 进行下一次搜索;

step 9: 输出最优值和对应的调度排序。

算法流程如图 1 所示。

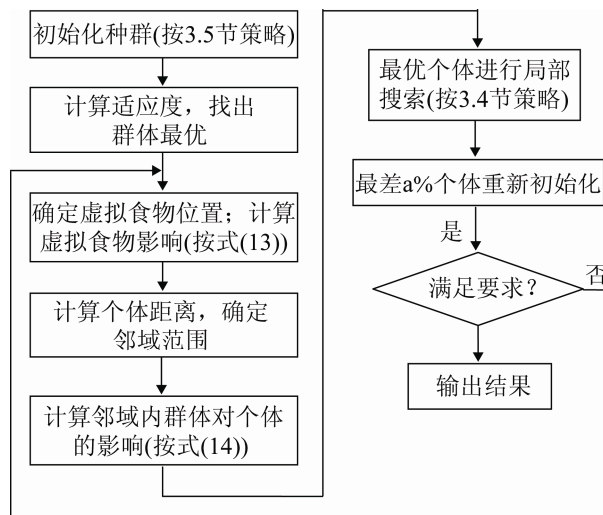


图 1 离散磷虾群优化算法流程

Fig. 1 Procedure of discrete krill herd algorithm

## 4 仿真测试及分析

为测试离散磷虾群优化算法性能, 本文选择了小规模 8 个实例 Car 类问题<sup>[18]</sup>和较大规模 21 个实例 Rec 类问题<sup>[19]</sup>及 2 个实例 Hel 问题<sup>[20]</sup>作为测试用例, 并与遗传退火算法(GASA)<sup>[21]</sup>、变邻域粒子群算法(PSOVNS)<sup>[22]</sup>和离散粒子群算法(DPSO)<sup>[23]</sup>进行对

比。仿真平台为 Windows 10 操作系统, 处理器主频 3.6 GHz, 内存 16 GB, MATLAB R2017b 软件。

### 4.1 参数设置

离散磷虾群优化算法中的参数值根据实验获取的经验值来确定, 磷虾群数量  $m=50$ , 邻域内交叉个体比例为 40%, 重新初始化个体比例为 5%, 最大搜索次数  $\max T=100$ , 算法独立运行 10 次。遗传退火算法参数: 采用精英保留策略, 染色体数  $m=40$ , 退火速率  $\lambda=0.9$ , 初始温度  $t_0=(C_{\text{worst}}-C_{\text{best}})/\ln(0.1)$ , 最大搜索次数  $\max T=30$ 。变邻域粒子群算法参数: 群体数  $m=30$ ; 采用线性减少的惯性权重:  $w_{\min}=0.4$ ,  $w_{\max}=0.9$ ; 学习因子:  $C_1=C_2=2.0$ ; 个体位置取值范围 $[0,4]$ , 速度取值范围 $[-4.0,4.0]$ ; 最大搜索次数  $\max T=50$ 。离散粒子群算法参数: 群体数  $m$ =工件数, 交叉概率  $c_1=0.2$ ,  $c_2=0.8$ ; 变异概率  $w=0.2$ ; 最大搜索次数  $\max T=1000$ 。

### 4.2 测试结果及分析

仿真测试结果如表 1 所示, 表 1 中 DKH 表示离散磷虾群算法, GASA 代表文献[21]所设计的遗传退火算法, PSOVNS 代表文献[22]所设计的变邻域粒子群算法, DPSO 代表文献[23]中的离散粒子群算法, P 列代表测试问题类型,  $n, m$  表示工件和机器数量,  $C^*$  表示该问题的最优值。由于 Rec 类问题目前还没有确定的最优值, 选择用 RAJ 算法的求解值作为 Rec 类和 Hel 问题的比较基准, 分别用三种算法的寻优值与  $C^*$  的相对误差进行对比。相对误差  $= (C_{\max} - C^*) / C^* \times 100\%$ ,  $C_{\max}$  表示寻优值。BRE 表示最优值相对误差, 用来衡量算法的精度; ARE 表示 10 次独立测试求得最优值的平均相对误差, 用来衡量算法的鲁棒性; WRE 表示最差值的相对误差。

从测试结果可以看出, 对于 Car 类测试问题, PSOVNS 在 Car4 实例上未找到最优值, 其余 3 种算法均找到了最优值; 对于 Car 类问题所有实例, DKH 算法的平均相对误差和最差相对误差指标均



优于其他 3 种算法且优势比较明显。对于 Rec 类测试问题, 从 BRE 指标来看, 除在 Rec39 实例上 DKH 算法的精度略差于 DPSO(在 Rec11、Rec15 实例上两者相同), 在其余实例上 DKH 算法的寻优精度均优于 GASA、PSOVNS、DPSO, 也远优于 RAJ 启发式算法; 从 ARE 和 WRE 指标来看, 对于所有实例, DKH 算法均优于 GASA 和 PSOVNS 且优势明显; 相对于 DPSO 算法, 除少数几个实例

外, DKH 算法在 ARE 和 WRE 指标上优于 DPSO, 说明 DKH 算法的鲁棒性优于对比算法。其次, DKH 算法的 BRE 和 ARE 指标数据差值较小, 说明 DKH 算法的稳定性也较好。综合来看, DKH 算法在寻优精度、稳定性、鲁棒性方面优于对比算法, 验证了 DKH 算法求解零等待流水线调度问题的可行性和有效性。

表 1 4 种算法仿真测试结果  
Tab. 1 Results of simulation test for four algorithms

P	n,m	C*	DKH			GASA			PSOVNS			DPSO		
			BRE	ARE	WRE	BRE	ARE	WRE	BRE	ARE	WRE	BRE	ARE	WRE
Car1	11,5	8 142	0	0	0	0	0.22	0.37	0	1.22	3.87	0	0.22	0.85
Car2	13,4	8 242	0	0.12	0.17	0	0.19	0.35	0	0.67	3.00	0	0.30	0.65
Car3	12,5	8 866	0	0.02	0.25	0	0.08	0.13	0	0.33	1.17	0	0.14	0.63
Car4	14,4	9 195	0	0.24	1.20	0	1.19	1.47	0.07	1.74	4.18	0	1.83	4.77
Car5	10,6	9 159	0	0	0	0	0.10	0.23	0	0.04	0.55	0	0	0
Car6	8,9	9 690	0	0	0	0	0	0	0	0	0	0	0	0
Car7	7,7	7 705	0	0	0	0	0	0.04	0	0	0	0	0	0
Car8	8,8	9 372	0	0	0	0	0	0.01	0	0	0	0	0	0
Rec01	20,5	1 590	-4.03	-3.89	-3.52	-3.96	-2.81	-2.09	-3.52	-2.62	-1.76	-3.58	-3.44	-2.89
Rec03	20,5	1 457	-6.59	-5.57	-4.60	-4.46	-2.94	-2.35	-5.63	-3.26	-0.34	-4.80	-4.64	-4.32
Rec05	20,5	1 637	-7.51	-6.84	-6.29	-6.90	-6.12	-5.63	-6.96	-6.26	-5.13	-7.15	-6.17	-5.62
Rec07	20,10	2 119	-3.63	-2.94	-2.08	-3.45	-2.35	0.74	-3.40	-1.88	0.05	-3.59	-3.20	-2.69
Rec09	20,10	2 141	-4.62	-4.18	-3.64	-4.48	-2.69	-1.94	-4.58	-2.26	-0.28	-4.11	-3.12	-0.93
Rec11	20,10	1 946	-3.34	-2.81	-2.42	-3.34	-0.81	0	-2.98	-0.75	1.03	-3.34	-2.53	-1.54
Rec13	20,15	2 709	-6.05	-5.36	-4.02	-5.65	-3.62	-2.67	-5.28	-3.05	-0.55	-5.65	-5.27	-4.25
Rec15	20,15	2 691	-6.02	-5.42	-4.98	-6.02	-4.75	-3.99	-5.69	-3.85	-1.82	-6.02	-5.54	-5.05
Rec17	20,15	2 740	-5.58	-5.53	-5.47	-5.47	-4.36	-3.82	-4.96	-3.81	-2.37	-5.51	-5.27	-5.04
Rec19	30,10	3 157	-8.55	-8.08	-7.60	-5.45	-4.39	-3.75	-8.39	-6.43	-4.78	-8.20	-7.79	-7.32
Rec21	30,10	3 015	-5.67	-5.02	-4.31	-2.22	-1.24	-0.47	-5.24	-3.46	-2.26	-4.88	-4.38	-4.01
Rec23	30,10	3 030	-10.83	-8.84	-6.93	-6.70	-4.50	-3.62	-8.61	-6.89	-4.65	-8.91	-8.78	-8.12
Rec25	30,15	3 835	-6.21	-5.59	-4.64	-2.69	-1.46	-0.74	-4.95	-2.64	-0.78	-5.27	-4.60	-4.02
Rec27	30,15	3 655	-5.64	-4.71	-3.78	-2.60	-0.81	-0.21	-3.99	-2.22	-1.12	-4.49	-3.53	-3.31
Rec29	30,15	3 583	-7.01	-4.67	-3.38	-3.99	-0.65	0.32	-5.08	-2.65	-0.25	-6.67	-6.32	-5.97
Rec31	50,10	4 631	-5.38	-3.67	-2.50	2.72	5.44	6.24	-2.96	-1.95	-0.45	-4.90	-4.29	-3.63
Rec33	50,10	4 770	-4.61	-4.05	-3.27	4.78	5.98	6.70	-3.00	-1.99	-0.75	-4.23	-3.12	-2.05
Rec35	50,10	4 718	-5.19	-4.54	-3.88	3.67	6.13	7.18	-3.52	-1.04	0.61	-4.49	-3.77	-3.18
Rec37	75,20	8 979	-8.23	-7.16	-4.44	5.89	7.43	8.14	-5.26	-4.52	-3.70	-8.16	-7.48	-6.86
Rec39	75,20	9 158	-4.83	-4.23	-3.72	8.80	10.50	11.12	-2.78	-1.88	-1.05	-5.57	-5.05	-4.58
Rec41	75,20	9 344	-7.06	-5.77	-4.98	6.79	9.03	9.95	-4.48	-3.38	-2.68	-6.62	-6.10	-5.57
Hel 1	100,10	780	-6.28	-5.59	-5.13	12.44	15.06	15.96	7.05	8.70	9.74	-6.15	-5.59	-5.26
Hel 2	20,10	189	-5.29	-5.08	-4.76	-4.76	-3.44	-2.49	-2.65	0.35	2.65	-4.23	-2.25	-1.06
合计			-138.2	-119.2	-98.7	-27.1	14.4	35.2	-96.8	-53.7	-7.9	-126.5	-109.7	-90.4

## 5 结论

磷虾群优化算法是基于对磷虾生存环境及觅食行为的模拟演变来的一种群体智能算法, 算法主要由食物位置、虾群密度、随机扩散运动 3 部分组成。本文在分析磷虾群优化算法机理的基础上, 针对求解问题特性, 提出了一种离散磷虾群优化算法, 首先定义了虚拟食物的计算方式和虚拟食物影响下磷虾个体之间的交叉规则, 以此来获取全局信息; 其次, 重新界定了磷虾个体之间的距离计算方法, 根据距离确定邻域范围实现邻域内个体之间的交流, 使个体可以有效获得局部信息; 进一步采用随机交换、逆序、插入等策略提升算法的局部搜索能力, 并借助多样化的初始化策略来提高初始种群的质量和多样性。通过典型实例的仿真测试和对比, 验证了所提算法在求解零等待流水车间调度问题的有效性和优越性, 接下来要进行的研究工作是在保持算法精度和稳定性前提下如何进一步提高算法计算效率, 更好地适应复杂应用环境。

## 参考文献:

- [1] Röck H. The Three-machine No-wait Flow Shop is NP-complete[J]. *Journal of the Association for Computing Machinery* (S0004-5411), 1984, 31(2): 336-345.
- [2] 齐学梅. 无等待流水调度问题迭代启发式算法[J]. *安徽师范大学学报(自然科学版)*, 2009, 32(1): 4-8.  
Qi Xuemei. Iterative Heuristic Algorithms for No-Wait Flowshop Scheduling Problems[J]. *Journal of Anhui Normal University (Natural Science)*, 2009, 32(1): 4-8.
- [3] 孙强强, 高开周, 董斌. 一种求解无等待流水车间调度优化的启发式算法[J]. *计算机应用研究*, 2012, 29(6): 2072-2073, 2101.  
Sun Qiangqiang, Gao Kaizhou, Dong Bin. Heuristic for No-wait Flow Shop Scheduling Optimization[J]. *Application Research of Computers*, 2012, 29(6): 2072-2073, 2101.
- [4] 李亚志, 朱夏. 基于插入-分段的无等待流水作业调度复合启发式算法[J]. *东南大学学报(自然科学版)*, 2013, 43(3): 483-488.  
Li Yazhi, Zhu Xia. Insertion-segment Based Composite Heuristic Algorithm for No-wait Flow Shop Scheduling Problems[J]. *Journal of Southeast University (Natural Science Edition)*, 2013, 43(3): 483-488.
- [5] Rani D S, Subrahmanyam N, Sydulu M. A Hybrid Particle Swarm Optimization Algorithm for a No-wait Flow Shop Scheduling Problem with the Total Flow Time[J]. *International Journal of Advanced Manufacturing Technology* (S0268-3768), 2014, 70(5/8): 1181-1188.
- [6] Gao K Z, Suganthan P N, Pan Q K. Discrete Harmony Search Algorithm for Flexible Job Shop Scheduling Problem with Multiple Objectives[J]. *Journal of Intelligent Manufacturing* (S0956-5515), 2016, 27(2): 363-374.
- [7] Sundar S, Suganthan P N, Jin C T, et al. A Hybrid Artificial Bee Colony Algorithm for the Job-shop Scheduling Problem with No-wait Constraint[J]. *Soft Computing* (S1432-7643), 2017, 21(5): 1193-1202.
- [8] 齐学梅, 王宏涛, 杨洁, 等. 量子萤火虫算法及在无等待流水调度上的应用[J]. *信息与控制*, 2017, 45(2): 211-217.  
Qi Xuemei, Wang Hongtao, Yang Jie, et al. Quantum Glowworm Swarm Algorithm and Its Application to No-wait Flowshop Scheduling[J]. *Information and Control*, 2017, 45(2): 211-217.
- [9] 张其亮, 俞祚明. 基于优势种群的离散果蝇优化算法求解无等待流水车间调度问题[J]. *计算机集成制造系统*, 2017, 23(3): 609-615.  
Zhang Qiliang, Yu Zuoming. Discrete Fruit Fly Optimization Algorithm Based on Dominant Population for Solving No-wait Flow Shop Scheduling Problem[J]. *Computer Integrated Manufacturing Systems*, 2017, 23(3): 609-615.
- [10] Gandomi A H, Alavi A H. Krill Herd: A New Bio-inspired Optimization Algorithm[J]. *Communications in Nonlinear Science and Numerical Simulation* (S1007-5704), 2012, 17(12): 4831-4845.
- [11] Wang G, Guo L, Wang H, et al. Incorporating Mutation Scheme into Krill Herd Algorithm for Global Numerical Optimization[J]. *Neural Computing and Applications* (S0941-0643), 2014, 24(3/4): 853-871.
- [12] Wang G G, Gandomi A H, Alavi A H. Stud Krill Herd Algorithm[J]. *Neurocomputing* (S0925-2312), 2014, 128(5): 363-370.
- [13] Wang G G, Gandomi A H, Alavi A H, et al. A Hybrid Method Based on Krill Herd and Quantum-behaved Particle Swarm Optimization[J]. *Neural Computing &*

- Applications (S0941-0643), 2016, 27(4): 989-1006.
- [14] Bolaji A L, Al-Betar M A, Awadallah M A, et al. A Comprehensive Review: Krill Herd Algorithm (KH) and its Applications[J]. Applied Soft Computing (S1568-4946), 2016, 49: 437-446.
- [15] Nawaz M, Ensore J, Ham I. A Heuristic Algorithm for the M-machine, N-job Flow-shop Sequencing Problem[J]. Omega(S0305-0483), 1983, 11(1): 91-95.
- [16] Campbell H G, Dudek R A, Smith M L. A Heuristic Algorithm for the n Job, m Machine Sequencing Problem[J]. Management Science (S0025-1909), 1970, 16(10): 630-637.
- [17] Rajendran C. A No-wait Flowshop Scheduling Heuristic to Minimize Makespan[J]. Journal of the Operational Research Society (S0160-5682), 1994, 45(4): 472-478.
- [18] Carlier J. Scheduling with Disjunctive Constraints[J]. Operations Research (S0030-364X), 1978, 12(4): 333-350.
- [19] Reeves C R. A Genetic Algorithm for Flowshop Sequencing[J]. Computers and Operations Research (S0305-0548), 1995, 22(1): 5-13.
- [20] Heller J. Some Numerical Experiments for an M×J Flow Shop and its Decision-theoretical Aspects[J]. Operations Research (S0030-364X), 1960, 8(2): 178-184.
- [21] Schuster C J, Framinan J M. Approximative Procedures for No-wait Job Shop Scheduling[J]. Operations Research Letters (S0167-6377), 2003, 31(4): 308-318.
- [22] Tasgetiren M F, Sevkli M, Liang Y C, et al. Particle Swarm Optimization Algorithm for Permutation Flowshop Sequencing Problem[J]. Lecture Notes in Computer Science (S0302-9743), 2004, 3172(14): 382-389.
- [23] Pan Q K, Fatih Tasgetiren M, Liang Y C. A Discrete Particle Swarm Optimization Algorithm for the No-wait Flowshop Scheduling Problem[J]. Computers and Operations Research (S0305-0548), 2008, 35(9): 2807-2839.