

1-18-2021

Improved Wolf Pack Algorithm for Distribution of Molds on Molds Table

Zhonghua Han

*1. Faculty of Information and Control Engineering, Shenyang Jianzhu University, Shenyang 110168, China;
;2. Department of Digital Factory, Shenyang Institute of Automation, the Chinese Academy of Sciences (CAS), Shenyang 110016, China; ;*

Yuehan Liu

*1. Faculty of Information and Control Engineering, Shenyang Jianzhu University, Shenyang 110168, China;
;*

Li Man

3. Yatai Group Shenyang Modern Building Industry Co., Ltd., Shenyang 110136, China; ;

Liangliang Sun

*1. Faculty of Information and Control Engineering, Shenyang Jianzhu University, Shenyang 110168, China;
;*

See next page for additional authors

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.

Improved Wolf Pack Algorithm for Distribution of Molds on Molds Table

Abstract

Abstract: In order to solve the problem of the distribution of molds on the mold table in the production process of concrete prefabricated components, a method combining Bottom-Left Algorithm (BLA) and Improved Wolf Pack Algorithm (IWPA) is proposed. Improved Wolf Pack Algorithm is for optimizing the order of upper line and the mold used for each member is sequentially assigned to the mold table according to the upper line. BLA is used to determine the direction and position of the molds on the molds table. The WPA safari behavior is improved by introducing Levi's flying ideas and the Hamming distance-based wolf pack dynamic update mechanism is used to improve the Wolf Pack Algorithm. The wolf pack diversity is enriched and the search ability of WPA in the solution space is enhanced. An example test is carried out to verify the effectiveness of the method.

Keywords

wolf pack algorithm, concrete prefabricated component, distribution of molds on the mold table, levy flights, hamming distance

Authors

Zhonghua Han, Yuehan Liu, Li Man, Liangliang Sun, and Hongzhi Zheng

Recommended Citation

Han Zhonghua, Liu Yuehan, Li Man, Sun Liangliang, Zheng Hongzhi. Improved Wolf Pack Algorithm for Distribution of Molds on Molds Table[J]. Journal of System Simulation, 2021, 33(1): 127-140.

改进狼群算法求解模具在模台上组合分配问题

韩忠华^{1,2}, 刘约翰¹, 李曼³, 孙亮亮¹, 郑鸿志⁴(1. 沈阳建筑大学 信息与控制工程学院, 辽宁 沈阳 110168; 2. 中国科学院沈阳自动化研究所 数字工厂研究室, 辽宁 沈阳 110016;
3. 亚泰集团 沈阳现代建筑工业有限公司, 辽宁 沈阳 110136; 4. 北京理工大学 机械与车辆学院, 北京 100081)

摘要: 为解决混凝土预制构件生产中存在的模具在模台上组合分配问题, 提出一种BL(Bottom-Left)定位算法与改进狼群算法(Improved Wolf Pack Algorithm, IWPA)结合的方法。改进狼群算法用于优化构件的上线序, 根据上线序依次将每个构件所使用的模具分配到模台上, 采用BL定位算法确定模具在模台上摆放方向和位置, 通过引入莱维飞行思想改进WPA的游猎行为和采用基于汉明距离的狼群动态更新机制来改进狼群算法, 丰富狼群多样性, 增强WPA在解空间的搜索能力, 最后通过实例测试, 验证了该方法对于解决该问题的有效性。

关键词: 狼群算法; 混凝土预制构件; 模具模台分配; 莱维飞行; 汉明距离

中图分类号: TP18; TH186; TU741.2 文献标志码: A 文章编号: 1004-731X(2021)01-0127-14
DOI: 10.16182/j.issn1004731x.joss.19-0262

Improved Wolf Pack Algorithm for Distribution of Molds on Molds Table

Han Zhonghua^{1,2}, Liu Yuehan¹, Li Man³, Sun Liangliang¹, Zheng Hongzhi⁴

(1. Faculty of Information and Control Engineering, Shenyang Jianzhu University, Shenyang 110168, China; 2. Department of Digital Factory, Shenyang Institute of Automation, the Chinese Academy of Sciences (CAS), Shenyang 110016, China; 3. Yatai Group Shenyang Modern Building Industry Co., Ltd., Shenyang 110136, China; 4. School of Mechanical Engineering, Beijing Institute of Technology, Beijing 100081, China)

Abstract: In order to solve the problem of the distribution of molds on the mold table in the production process of concrete prefabricated components, a method combining Bottom-Left Algorithm (BLA) and Improved Wolf Pack Algorithm (IWPA) is proposed. Improved Wolf Pack Algorithm is for optimizing the order of upper line and the mold used for each member is sequentially assigned to the mold table according to the upper line. BLA is used to determine the direction and position of the molds on the molds table. The WPA safari behavior is improved by introducing Levi's flying ideas and the Hamming distance-based wolf pack dynamic update mechanism is used to improve the Wolf Pack Algorithm. The wolf pack diversity is enriched and the search ability of WPA in the solution space is enhanced. An example test is carried out to verify the effectiveness of the method.

Keywords: wolf pack algorithm; concrete prefabricated component; distribution of molds on the mold table; levy flights; hamming distance

引言

随着经济与社会的发展, 人们对居住环境的要求也日益提升, 装配式建筑^[1]较传统的施工方式具

有对周围施工环境影响小, 施工速度快等优点, 装配式建筑的建造^[2-4]过程中所使用的预制构件需在企业中提前生产, 当投产预制构件上线时, 其所使

收稿日期: 2019-06-27 修回日期: 2019-07-30

基金项目: 国家自然科学基金(61873174), 辽宁省重点研发计划(2018106008), 辽宁省高等学校基本科研项目(LJZ2017015), 沈阳市科技计划双百工程(Z18-5-015)

作者简介: 韩忠华(1977-), 男, 博士, 教授, 硕导, 研究方向为生产运作管理、企业自动化系统集成技术、车间排产与生产调度算法工程应用。E-mail: xiaozhonghua1977@163.com

用的模具在支模工艺段被固定在模台上, 模台作为转运工具承载着模具完成预制构件生产的全过程, 每个模台尺寸相同, 而模具尺寸不同。在模具分配到模台的过程中, 模具在模台上会有多种组合分配结果, 如果不能合理将模具分配到模台上, 会导致模台空余的面积内无法再摆放其他模具, 模台的台面得不到充分利用, 模台的利用率降低, 完成生产同样数量的预制构件需要使用模台的数量更多。由于每天投产的预制件所需的模台数远大于企业中提供的模台数, 生产线上的模台会循环重复使用, 而又由于预制件生产线具有流水车间的特征, 重复使用模台会极大的延长生产任务的整体完工时间, 导致企业生产效率降低, 严重的制约了企业产能的发挥。因此, 研究混凝土预制构件生产线上模具在模台上的分配问题, 探索提高模台利用率, 减少单个模台使用次数的方法, 对提高企业生产效率, 降低企业的生产成本, 挖掘现有资源潜力具有重要的意义。

预制构件生产线与传统制造企业生产线既有很多相似的特征, 也有自己独特的工艺要求和生产运作规律, 越来越多的学者开始重视对预制构件生产相关问题的研究。谢思聪等^[5]提出了一种基于多层编码遗传算法对两阶段装配式建筑预制构件生产调度进行优化, 在分析预制构件生产工艺的基础上提出预制构件厂生产参数的量化途径, 采用两阶段优化模式提升优化性能, 提高预制件生产效率。朱敏涛^[6]提出一种新型双向可扩展的组合式预制构件数字化生产线。该生产线具有生产设施和装备可扩展布置、构件生产数字化和机械化程度高、设备可灵活组合等特点。当前学者针对混凝土预制构件生产线, 多从优化工艺流程和调整生产资源的角度进行研究, 对于在混凝土预制构件生产线中支模工艺段模具在模台上组合分配问题研究较少。

模台作为转运工具承载着模具完成预制构件生产的全过程, 依据预制构件的上线序模具依次被摆放到模台上, 同时, 模具的摆放过程还受到其尺寸、摆放位置和摆放方向的影响, 采取不同的模具

在模台上摆放的方法也会对摆放结果产生影响, 因此, 会产生多种模具在模台上组合分配结果。模具在模台上组合分配的过程中, 模具在模台上的摆放方法也对于提高模台的利用率具有至关重要的作用, 但目前国内对该问题研究较少, 王中原等^[7]针对板类预制构件生产中的模具在模台上排布优化问题, 采用低水平线算法建立了模具排布的模型, 对模具进行分层排布, 本文采用 BL(Bottom-Left)定位算法解决模具在模台上的定位摆放问题, 根据某预制构件生产企业实际生产过程可知, 对于预制构件这类大体积和大质量的在制品, 标准模台上最多只能放下 3 个模具, 按上线序新分配到模台上的模具与模台上其他已摆放的模具不发生干涉的条件下, 将模具尽可能的往下往左移动摆放, 直到受到其他模具或者模台边界的阻碍不能再移动为止, 通过被摆放在模台上的第一个模具的摆放角度和位置来依次确定该模台上后续分配的模具的摆放情况。BL 算法具有排样策略简单、计算速度快、易于在实际生产中应用等特点, 因此, 可以更合理高效的实现将模具向模台上摆放。

由于实际企业单日投产预制构件数量较多, 需使用多个模台承载模具, 因此, 多个模具往多个模台上进行分配的过程中会出现“组合爆炸”的情况, 因此, 在实际的工程应用中需要一种能够快速寻优的算法来确定投产预制构件的最优上线序, 狼群算法相比其他群体智能进化算法具有收敛速度快、寻优精度较高等优点, 得到了学者关注, 马龙等^[8]针对离散空间优化问题, 提出了求解离散优化问题的元胞量子狼群演化算法, 为了实现狼群中个体狼与猎物距离的精确描述和量子旋转角的选取调整, 提出二进制编码方式和元胞自动机中的演化规则, 取得较好的寻优能力。黄海松等^[9]提出一种改进的狼群算法求解旅行商问题, 通过在初始化阶段引入位置-次序编码的方法, 同时在迭代过程中引入二次搜索来提高算法求解速度与精度, 以实现在达到最大迭代次数前出现最优解, 相比标准狼群算法求解精度更高, 收敛速度更快。

通过对相关文献的分析可知,目前少有学者结合预制构件生产线的相关特点,对于在预制构件生产过程中模具在模台上摆放进行研究,也未对生产效率密切相关的模具在模台上组合分配问题进行进一步的探索。为更好的解决该问题,提出一种 BL 定位算法与改进狼群算法结合的方法进行求解。依据预制构件的上线序确定与其对应模具的摆放顺序,再通过 BL 定位算法来控制模具在模台上定位与摆放过程。同时,采用狼群算法优化预制构件的上线序,并针对标准狼群算法易陷入局部极值的问题,引入莱维飞行和基于汉明距离种群动态更新机制,分别增强个体探狼寻优性能和丰富狼群的多样性,改善并提升狼群算法的寻优性能,从而获取能够更有效利用模台台面面积的模具组合分配结果,实现提高模台利用率,减少模台使用量的目标。

1 预制构件数学规划模型

1.1 模型参数

W : 模台的长度;

H : 模台的宽度;

m : 模台使用个数;

n : 待生产加工的预制构件的个数;

M_i : 第 i 个模具, $i \in \{1, 2, \dots, n\}$;

B_j : 第 j 个模台, $j \in \{1, 2, \dots, m\}$;

w_i : 第 i 个模具的长度, $i \in \{1, 2, \dots, n\}$;

h_i : 第 i 个模具的宽度, $i \in \{1, 2, \dots, n\}$;

R : 模台的空间利用率;

S_i : 第 i 个模具 M_i 的面积, $i \in \{1, 2, \dots, n\}$;

P_{ij} : 模具 M_i 摆放在模台 B_j 上时, 模具 M_i 定位点所在的位置;

x_{ij} : 模具 M_i 摆放在模台 B_j 上时定位点 P_{ij} 的横坐标;

y_{ij} : 模具 M_i 摆放在模台 B_j 上时定位点 P_{ij} 的纵坐标;

D_i : 模具 M_i 的摆放方向, 当 M_i 不旋转 90° 摆放取值 0, 当 M_i 旋转 90° 摆放取值 1。

1.2 模具在模台上组合分配问题描述

预制构件生产车间是半自动化的流水车间,其生产工艺流程主要包括清理作业、喷油作业、边模安装、钢筋安装、预埋件安装、浇筑振捣、拉毛、码垛、蒸养、拆模、吊装、成品,其中清理作业、喷油作业、边模安装在支模工艺段执行。预制构件的工艺流程中所有工序均在一个固定尺寸的模台上,模台随着不同的加工过程在流水线上进行移动,实现工艺流程中不同工序的加工任务。

模具在模台上的组合分配问题可以看作是一类矩形件排样问题,是指在矩形的平台上摆放不同规格的矩形模具,在一定的约束条件下,令矩形平台的利用率最高,该类问题需要获取预制构件的上线序以确定模台上将要分配的模具,以及模具在模台上的定位与摆放,可以描述为设有未知数 m 个规格尺寸完全相同的模台,模台的宽长为 W ,宽度为 H ,生产任务要求生产 n 个不同的预制构件,每一个预制构件对应一个独立的模具,调整模具对应预制构件的加工先后顺序和模具在模台上的摆放位置以及摆放方向,实现模具在模台上的组合分配,模具在模台上组合分配问题的结果要确定模具对应预制构件的加工先后顺序、模具与模台组合结果、模具在模台上摆放位置与方向以及模台的使用数量和模台的利用率。

依据预制构件的上线序依次将对应使用的模具 $\{M_1, M_2, \dots, M_n\}$ 分配到模台上,将实际的模台平面模拟为具有二维坐标的矩形平面,其中平面的 x 轴设定为固定长度 W ,表示为模台的长度,平面的 y 轴设定为固定长度 H ,表示为模台的宽度,设二维平面的左下角为坐标原点 $(0, 0)$,则二维平面的左上角坐标为 $(H, 0)$,右下角坐标为 $(0, W)$,右上角坐标为 (W, H) 。待摆放的矩形模具 M_i 模拟为二维矩形,且其尺寸参数已知,模具在模台上所需预留的固定螺栓空洞尺寸忽略不计,模具在模台上摆放以模具的左下角为定位点,即模具 M_i 左下角在模台上的位置 P_{ij} 坐标为 (x_{ij}, y_{ij}) ,模具左上角位置坐标为 $(x_{ij}, y_{ij}+h_i)$,模具的右下角的位置坐标为 $(x_{ij}+w_i,$

$y_{i,j}$), 若模具需要旋转 90° 进行摆放, 则在计算矩形模具顶点时的长度 w_i 和宽度 h_i 互换, 即模具左上角位置坐标为 $(x_{i,j}, y_{i,j}+w_i)$, 模具的右下角的位置坐标为 $(x_{i,j}+h_i, y_{i,j})$, 如图 1 所示。用变量 D_i 表示模具 M_i 摆放到模台上时是否需要旋转, 当 $D_i=0$ 时模具不需要旋转 90° 摆放, 当 $D_i=1$ 时模具需要旋转 90° 摆放, 则可用式(1)进行表示:

$$\begin{cases} x'_{i,j} = x_{i,j} + (1 - D_i)w_i + D_i h_i \\ y'_{i,j} = y_{i,j} + D_i w_i + (1 - D_i)h_i \end{cases} \quad (1)$$

利用矩形模具左上角顶点 $(x_{i,j}, y_{i,j})$ 即定位点、长度 w_i 、宽度 h_i 以及旋转参数 D_i , 可以确定模具 M_i 在模台 B_j 上摆放的位置和方向。

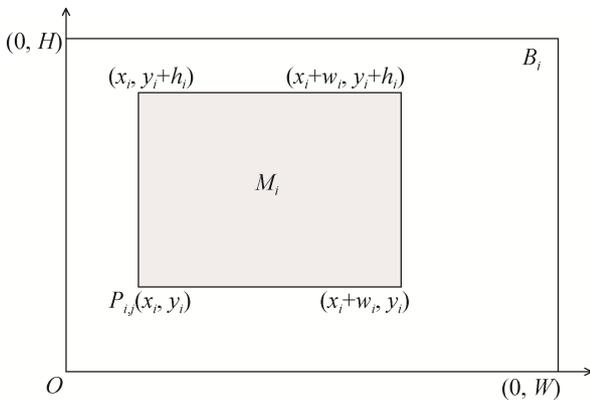


图 1 模具在模台上摆放模型图

Fig. 1 Schematic diagram of mold placed on table

1.3 约束条件

模具在模台上组合分配问题需要满足的约束条件:

(1) 在同一模台上布置的所有模具面积总和小于等于单个模台的面积。

$$\sum_{i=1}^{n'_j} S_i \leq WH \quad (2)$$

式中 n'_j 为在模台 B_i 上摆放模具的数量。

(2) 使用的模台面积总和大于等于所有摆放模具的面积总和。

$$\sum_{i=1}^n S_i \leq mWH \quad (3)$$

(3) 预制构件对应矩形模具的 4 个顶点均位于模台内部。

$$\begin{cases} 0 \leq x_i \leq H \\ 0 \leq y_i \leq W \\ 0 \leq x_i + w_i \leq H \\ 0 \leq y_i + h_i \leq W \end{cases} \quad (4)$$

除了以上约束条件, 还需要满足模具摆放需要正交放置, 不可倾斜摆放, 任意模具摆放不能出现重叠。

1.4 基于 BL 定位算法的模具在模台上摆放方法

模具在模台上组合分配可以看作是一类矩形件排样问题, 目前解决这类问题多采用 BL 定位算法, 该算法是以最左和最下为原则对矩形模具进行摆放, 模具在模台上首先向下摆放, 若在模台区域的范围内不能够继续摆放则向左摆放, 若模台无法放入模具则更换下一个模台进行摆放。具体摆放过程如图 2 所示。

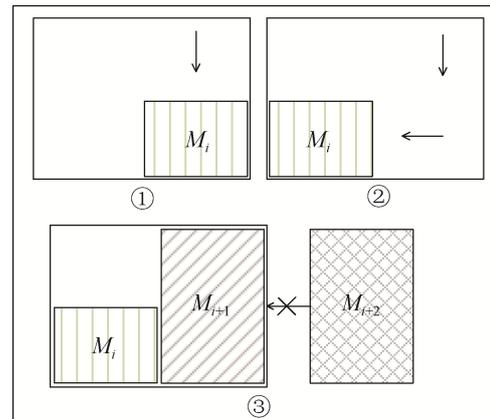


图 2 BL 定位算法模具在模台上摆放

Fig. 2 Placing molds on die table with BLA

将实际模台台面模拟为二维坐标图像, 模具摆放在模台上的过程中依据预制构件投产上线序进行模具在模台上摆放, 当摆放至第 j 个模台时, 假设还有预制构件对应模具 $\{M_i, M_{i+1}, \dots, M_n\}$ 需要摆放到模台上, 首先摆放模具 M_i , 将模具从模台的右上位位置向下摆放, 直至模具达到模台空间边缘或无法继续向下摆放(如图 2 中①所示), 然后将模具向左摆放至模台空间边缘或者无法继续向左摆放(如图 2 中②所示), 同样的方法摆放后续模具

M_{i+1} , 当模台无法继续放置下一个模具 M_{i+2} 时, 更换下一个模台进行摆放。

2 基于改进狼群算法的预制构件上线序优化研究

投产预制构件对应使用模具在模台上组合分配的过程中, 对于一个确定的预制构件上线序, 依次将预制构件对应的模具分配到模台上时, 受到模具尺寸、模具摆放的位置和方向的影响, 会产生多组分配结果, 当预制构件的上线序改变时, 依次选取的预制构件所对应使用的模具进行分配又会产生更多种组合分配结果, 因此, 多个模具往多个模台上进行分配会出现“组合爆炸”的情况, 为能够高效的求解这类具有很高复杂性的问题, 在确定模台摆放方法后, 需要一种能够快速寻优的算法对预制构件的上线序进行优化, 狼群算法相比其他群体智能进化算法具有收敛速度快、寻优精度较高等优点, 因此选用狼群算法来优化预制构件的上线序。

2.1 狼群算法基本思想及关键性问题

狼群算法^[10-13](Wolf Pack Algorithm, WPA)通过模仿自然界中群狼相互协作, 捕食猎物处理优化问题。狼群依照捕食猎物的能力将分为 3 个种类: 头狼、探狼和猛狼, 狼群中每只狼依照其职能划分, 各司其职相互协作共同捕食猎物。在 WPA 算法中, 为在搜索空间中获捕猎物, 选取具有最佳适应度值的人工狼作为头狼。然后在除头狼外的狼群中再选取最佳的 q 匹人工狼作为探狼, 探狼自主进行寻优搜索, 当搜寻到具有更优适应度值的位置时, 如果发现该探狼所在的新位置比当前头狼的位置具有更优的适应度值, 则该探狼代替原头狼成为新的头狼, 此过程称为探狼游猎行为。头狼发起嚎叫, 召集猛狼迅速向头狼靠拢, 猛狼在移动过程中搜寻猎物, 如果发现猛狼的新位置比头狼的位置具有更优的适应度值, 则该猛狼成为新的头狼, 此过程称为猛狼围猎行为。最后将适应度值最差的 R 匹狼淘汰, 同时随机产生 R 匹狼进行补充, 完成种群更新。

WPA 算法通过迭代进化, 不断地寻找出具有最优适应度值个体的位置, 即获得优化问题的解。

(1) 构建初始狼群位置

设在 D 维的搜索空间中, 由 N 匹狼组成一个人工狼群, 狼群中第 k 匹狼的位置表示为:

$$X_k = (x_{k,1}, x_{k,2}, \dots, x_{k,d}, \dots, x_{k,D}),$$

$$k \in \{1, 2, \dots, N\}, d \in \{1, 2, \dots, D\}$$
 (5)

每匹人工狼的初始位置可通过式(6)随机生成:

$$X_{kd} = x_{\min} + \text{rand}(x_{\max} - x_{\min}),$$

$$k \in \{1, 2, \dots, N\}, d \in \{1, 2, \dots, D\}$$
 (6)

式中: rand 为在区间(0, 1)内均匀分布的一个随机数; x_{\max} 和 x_{\min} 为搜索空间的上下限。

(2) 游猎行为

选取适应度值最优的 q 匹人工狼作为探狼, 这 q 匹探狼在附近产生 h 个方向, 并沿着各方向进行搜索, 假设探狼所在的位置为 $Y_k = (y_{k,1}, y_{k,2}, \dots, y_{k,D})$ 。若探狼 k 的当前位置为 Y_k , Y_k 周围产生一个新位置 Y_l , 如果新位置 Y_l 的适应度比 Y_k 优, 则探狼奔向位置 Y_l , 之后继续游猎寻找猎物, 直到搜索的次数大于设定的最大搜索次数 maxdh , 游猎行为结束。

围绕着探狼 k 的位置 Y_k 所产生的共 h 个点中, 第 l 个点的 d 维位置坐标 $y_{l,d}$ 为:

$$y_{l,d} = x_{k,d} + \text{rand} \times \text{stepA}$$
 (7)

式中: rand 为一个均匀分布的随机数, 区间为[-1, 1]; $x_{k,d}$ 为在竞选狼中第 k 匹的 d 维的空间坐标; l 的取值范围为[1, h]; stepA 为探狼的搜索步长。

当 q 匹探狼完成其全部的搜索行为后, 对比它们所在位置 Y_k 对应的适应度, 选取位于计算适应度最优的位置的人工狼作为头狼。

(3) 召唤行为

所竞选出得头狼对其他人工狼进行召唤, 其他 $N-1$ 匹人工狼向头狼位置 P_L 进行移动, 同时部分人工狼在移动过程中可能会发现“猎物”的位置并未出现在头狼位置 P_L 的方向, 而导致其向远离头狼位置进行移动, 因此狼群中第 k 匹狼更新后的位置 Z_k 的第 d 维位置 $z_{k,d}$ 坐标表示为:

$$z_{k,d} = x_{k,d} + rand \times stepB \times (x_{L,d} - x_{k,d}) \quad (8)$$

式中: $stepB$ 为狼群向头狼位置移动的步长; $x_{L,d}$ 为头狼位置的第 d 维坐标。

经过位置更新后第 k 只狼的位置坐标为 $Z_k(z_{k,1}, \dots, z_{k,d}, \dots, z_{k,D})$, 如果更新后位置坐标 Z_k 所计算的适应度值优于当前的位置坐标 X_k 所计算的适应度值, 则该匹人工狼奔向 Z_k 位置, 反之, 则停留在当前位置 X_k 不移动。

(4) 围猎行为

头狼召唤狼群后对猎物进行围猎, 除头狼外其他 $N-1$ 匹人工狼在头狼位置 X_L 附近围猎猎物。

生成一个 $[0, 1]$ 的随机数 r , 当随机数 r 的值大于预设阈值 φ , 则第 k 匹人工狼进行移动, 对猎物进行包围, 否则不进行移动, 其更新后的位置表示为 X_k^{t+1} :

$$X_k^{t+1} = \begin{cases} X_k^{t+1} & r < \theta \\ X_L + r \times C & r \geq \theta \end{cases} \quad (9)$$

式中: C 为狼群的包围步长; X_L 为头狼的位置; X_k^t 为第 t 代第 X_k 只狼的当前位置。

由于采用式(9)更新狼群的位置后可能使狼群的位置超出搜索的空间, 为避免越界情况的发生, 采用归一法处理狼群更新的位置。当算法迭代进化, 所求解会逐渐逼近理论最优, 若令包围步长 C 随着迭代数的增加而减小, 可提高算法找到更优解的概率。

$$C(t) = C_{\min} \times (x_{\max} - x_{\min}) \times \exp\left(\frac{\ln(C_{\min} / C_{\max}) \times t}{\max t}\right) \quad (10)$$

式中: C_{\max} 和 C_{\min} 分别为最大最小包围步长; $\max t$ 为最大的迭代次数; t 为当前迭代次数。

2.2 WPA 算法的改进

在标准狼群算法 (wolf pack algorithm, WPA) 的基础上, 提出一种新的基于莱维飞行^[14-16]的动态狼群算法(Dynamic wolf pack algorithm Based on Levy Flight, LDWPA), 该算法包含了 2 处改进:

(1) 基于莱维飞行的游猎行为

通过引入莱维飞行思想扩大搜索范围, 当在搜索范围内陷入局部极值时, 扩大搜索领域, 提高陷入局部极值时的搜索能力。莱维飞行是一种典型的随机步长的运动模式, 其服从莱维分布, 即短距离运动与少量长距离的运动相结合。基于莱维飞行游猎行为的竞选狼位置更新式为:

$$X_k^{t+1} = X_k^t + a \cdot levy(u, v), k = \{1, 2, \dots, n\} \quad (11)$$

式中: X_k^t 为竞选狼 X_k 第 t 代位置, 为点对点乘法; a 为步长因子; $levy(u, v)$ 为采用 Mantegna 算法模拟莱维飞行的随机步长。该算法由 Mantegna R N 于 1994 年提出^[17], 其公式为:

$$levy(\lambda) = u \left/ |v| \frac{1}{\beta} \right. \quad (12)$$

式中: β 取值为常数 1.5; u, v 服从标准正态分布, $u \sim N(0, \sigma_u^2)$, $v \sim N(0, \sigma_v^2)$, 其中 σ_u 和 σ_v 分别为:

$$\sigma_u = \left\{ \frac{\Gamma(1 + \beta) \sin(\frac{\pi\beta}{2})^{1/\beta}}{\Gamma[1 + \beta] \beta \cdot 2^{(1-\beta)/2}} \right\} \quad (13)$$

$$\sigma_v = 1 \quad (14)$$

采用莱维飞行的思想改进算法的搜索步长, 使得种群个体有一定机率奔出原本的小概率探索区域, 扩大搜索范围, 丰富狼群位置的多样性, 提高了算法的快速性和有效性。因此, 莱维飞行能增加种群多样性和扩大搜索范围, 采用莱维飞行的智能优化算法更容易跳出局部最优点, 可以有效增强算法的寻优能力。

(2) 基于汉明距离的种群动态更新机制

WPA 算法在经过一定次数的迭代进化后, 易陷入局部极值, 出现进化停滞的现象: 随着进化代数的增加, 狼群个体之间的位置坐标相似片断增多, 甚至部分个体的位置坐标可能完全相同, 个体间位置信息交流过程意义降低。因此, 引入一种动态更新种群方法, 当最优值停滞进化次数超过阈值时, 通过汉明距离^[18]判断个体间相似度, 淘汰与最优个体相似度高的个体, 同时产生相似度低、差异性大的新个体, 从而丰富个体多样性, 增强跳出局部极值的能力, 保持算法持续进化活力。

汉明距离常用于描述 2 个码字间差异性, 即 2

个字符串 $x=(x_1, x_2, \dots, x_D)$ 和 $y=(y_1, y_2, \dots, y_D)$ 所对应位置的不同字符个数:

$$D(x, y) = \sum_{d=1}^D x_d \oplus y_d \quad (15)$$

式中: $D(x, y)$ 为任意 2 个码字之间对应位置上码元取值不同的位的数目之和, 准确地反映 2 个码字之间的差异, 进而达到客观判断提供码字间相似性的目的; \oplus 为异或运算; $x_d \in \{0, 1\}$; $y_d \in \{0, 1\}$ 。

狼群中个体之间相似度为 S , 其表示 2 匹人工狼相同位置坐标片段占总坐标长度的比例。将计算出的相似度 S 与定义的阈值 St 进行对比, 从头狼开始将个体按照适应度值进行排序, 计算头狼与狼群中其他人工狼之间位置的相似度 S , 若 S 小于阈值 St 表明 2 个个体不相似, 否则 2 个个体为相似个体。

$$s_{b,c} = \begin{cases} 0 & [a_b] \neq [a_c] \\ 1 & [a_b] = [a_c] \end{cases}$$

假设变量 $s_{b,c}$ 表示狼群中 2 个不同的个体 $\{X_b, X_c\}$ 映射的上线序其所对应位置坐标片段 $[a_b]$, $[a_c]$ 表示的上线序是否相同, 如果相同, 则 $s_{b,c}=1$, 否则 $s_{b,c}=0$ 。

$$N_S = \sum_{d=1}^D s_{b,c} \quad (16)$$

式中: N_S 为狼群中 2 个个体具有相同位置坐标的个数。

$$S = N_S / N_D \quad (17)$$

式中: S 为狼群中两匹人工狼位置坐标的相似度, 即狼群中 2 匹人工狼有相同位置坐标的个数 N_S 与狼群中人工狼的位置坐标长度 N_D 的比值, 若相似度 S 大于相似度阈值 St , 则说明 2 匹人工狼的位置坐标相似。

当寻优过程进行到一定迭代次数后, 寻找与最优人工狼相似度较高的人工狼, 保留其中相似度低的人工狼, 淘汰相似度高的人工狼, 产生与头狼相似度低的人工狼补充狼群, 进而使狼群能够在解空间中保持良好的分布性, 增加狼群算法的全局搜索能力。具体过程为:

step 1: 计算人工狼与头狼之间的相似度 S , 将大于相似度阈值 St 的个体组成临时更新子狼群 $InPop$, 小于等于相似度阈值 St 的人工狼则组成临时保留子狼群 $olPop$, 临时更新子狼群 $InPop$ 中个体数为 $stnp$, 临时保留子狼群 $olPop$ 个体数为 $NP-stnp$;

step 2: 根据符合进化趋势人工狼保留比例参数 K_r 提取临时更新子狼群 $ItPop$ 中优秀的人工狼数为 $K_r \times stnp$, 淘汰临时更新子狼群 $ItPop$ 中剩余人工狼, 后续步骤用于产生新的人工狼用于补充临时更新子狼群 $ItPop$ 淘汰的人工狼, 为统计在临时更新子狼群 $ItPop$ 中新增人工狼的数量, 设变量 tnp , 并令 $tnp=0$;

step 3: 在寻优空间中随机位置产生一只新的人工狼, 判断该人工狼与头狼的相似度, 如果两者间相似度大于设定阈值 St , 则淘汰掉新人工狼; 如果两者间相似度小于等于相似度阈值 St , 则保留该人工狼, 并将该人工狼加入到临时更新子狼群 $ItPop$, 令 $tnp=tnp+1$;

step 4: 判断新增人工狼数量 tnp 是否达到淘汰临时更新子狼群 $ItPop$ 个体数 $stnp$, 即 $tnp+K_r \times stnp=stnp$, 如果达到, 则继续向下执行; 如果未达到, 则转到 step 3 执行;

step 5: 将临时保留子狼群 $olPop$ 与临时更新子狼群 $ItPop$ 结合成新的狼群 $newPop$ 。

因狼群进化过程初期在解空间中具有较好的分布性, 为提高算法时间性能, 当进化过程进行到一定代数, 才启动基于汉明距离的种群动态更新方法。

2.3 LDWPA 实施步骤

LDWPA 算法流程图如图 3 所示。

step1: 初始化狼群算法的参数, 设置狼群拥有人工狼的个数为 Np 、算法的最大迭代次数 gen_{max} 、竞争首领狼的个数 q 、搜索方向个数 h 、最大搜索次数 SC_{max} 、搜索步长因子 a 、移动步长为 $stepb$, 最差狼的个数 bw , 并生成初始种群, 对每匹狼的位置进行初始化;

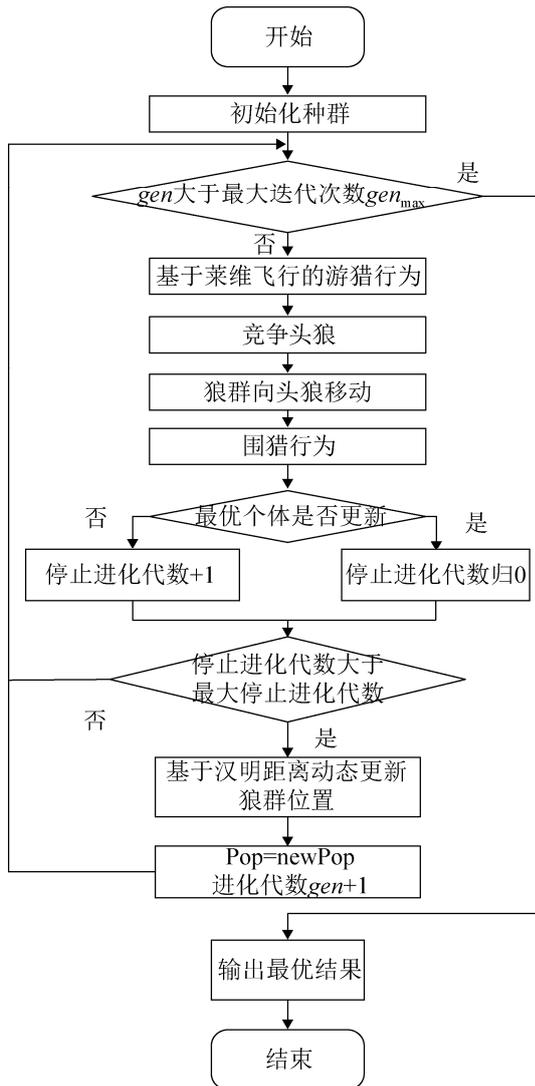


图 3 LDWPA 算法流程图
Fig. 3 Flow chart of LDWPA

step 2: 选取最优的 q 匹人工狼作为探狼, 并采用基于莱维飞行的游猎行为进行搜索, 通过游猎搜索出更优的位置, 计算探狼所在位置的适应度值, 根据计算出的适应度竞选出头狼;

step 3: 确定作为头狼后, 其他的 $Np-1$ 匹人工狼向着头狼的位置进行移动, 按式(8)对其位置进行更新;

step 4: 猛狼联合探狼对猎物进行围攻, 头狼位置视为猎物所在位置;

step 5: 依据狼群的分配原则对狼群位置进行更新, 去除狼群中位置对应的适应度最差的 bw 匹人工狼, 并产生新的 bw 匹人工狼补充狼群;

step 6: 判断是否满足停止进化的条件, 如果满足停止进化条件, 则退出优化, 否则继续执行进化;

step 7: 若当前代的狼群中没有更新出新最优位置的人工狼, 记录已经停止进化的代数;

step 8: 判断是否达到执行检测狼群中个体相似度运算的条件, 若 $gen < startgen$, 则不执行检测相似度运算, 转到 step 2; 否则, 执行 step 9;

step 9: 通过基于汉明距离更新种群中与头狼相似度高的个体, 得新狼群 $newPop$, 将其作为新一代进化的狼群 Pop , 并 $gen=gen+1$, 然后转到 step 2 执行。

3 仿真实验

为进一步研究本文提出的方法, 即 BL 定位算法控制模具在模台上的摆放, 改进的狼群算法优化投产预制构件的上线序, 2 种方法结合对该类问题进行求解的方法, 通过使用某混凝土预制构件生产企业的生产实例对该方法进行仿真实验, 分别从优化结果、算法优化过程和多次优化结果的稳定性方面, 多角度的分析评估该方法求解模具在模台上组合分配问题的效果。

3.1 构造仿真数据

某建筑预制构件生产企业的生产线中, 模台的尺寸为 $600\text{ cm} \times 800\text{ cm}$, 需生产一批不同类型预制构件, 共有 28 种类型, 共计生产 100 个预制构件, 其尺寸及数量如表 1 所示。

表 1 预制构件生产清单

Tab. 1 Production list of prefabricated components			
类型编号	长度/cm	宽度/cm	加工数量/(个)
1	300	400	3
2	600	400	4
3	550	380	2
4	300	320	1
5	600	750	2
6	560	400	4
7	290	360	2
8	260	480	2
9	580	420	1
10	510	450	1

续表

类型编号	长度/cm	宽度/cm	加工数量/(个)
11	570	390	2
12	390	360	1
13	460	550	1
14	480	350	1
15	510	390	1
16	580	400	1
17	250	360	1
18	285	370	6
19	250	250	10
20	550	250	5
21	300	350	4
22	275	330	7
23	455	390	6
24	300	350	5
25	500	600	3
26	265	385	9
27	240	350	7
28	300	350	8

3.2 仿真方案设计及评价指标构建

(1) 仿真方案设计

构建仿真方案的信息如表 2 所示。

表 2 4 组仿真方案信息

Tab. 2 4 sets of simulation scheme information

仿真方案	优化算法
方案 1	贪心算法(Greedy Algorithm)
方案 2	粒子群算法(PSO)优化上线序
方案 3	标准狼群算法(WPA)优化上线序
方案 4	改进的狼群算法(LDWPA)优化上线序

由于模具在模台上组合分配问题是一种类背包问题, 方案 1 采用贪心算法求解该类问题, 将预制构件所使用模具的面积作为单位价值量, 根据贪心原则, 即优先选择单位价值最大的模具进行组合, 再通过 BL 定位算法将组合结果确定的模具分配到模台上, 方案 2, 3, 4 采用群体进化算法作为全局优化算法优化预制构件的上线序, 再通过 BL 定位算法按上线序依次将模具摆放在模台上, 其中方案 2 中采用粒子群算法优化预制构件的上线序, 方案 3 中采用标准狼群算法来优化预制构件的上线序, 方案 4 中采用基于莱维飞行的动态狼群算法

优化预制构件的上线序。对 4 种方案的仿真结果进行对比, 分析各个方案求解模具在模台上组合分配问题优化效果,

(2) 构建评价指标

1) 模台利用率

$$R = \frac{\sum_{i=1}^n S_i}{W \times H \times m} \quad (18)$$

式中: $\sum_{i=1}^n S_i$ 为摆放 n 个模具所占用模台台面面积的总和; m 为使用模台的个数。

2) 平均模台利用率

进行 E 次仿真实验后得出的平均模台利用率见式(19):

$$\bar{R} = \sum_{e=1}^E R_e / E \quad (19)$$

式中: E 为最大试验次数; R_e 为一次实验中获得的模台利用率。

3) 最低模台利用率:

进行 E 次仿真实验过程中模台利用率的最小值如式(20)所示:

$$R_{\min} = \min\{R_1, R_2, \dots, R_E\} \quad (20)$$

3.3 优化仿真结果分析

为分析全局优化算法在组合分配问题中的应用效果, 选取仿真数据中预制构件类型编号 1~17 的数据, 分别为采用随机上线序和采用改进的 LDWPA 算法优化预制构件上线序的模具模台组合分配结果简图, 从图 4 可以看出, 由于没有加入全局优化算法, 会出现多个模台上只摆放一个模具的情况, 模台台面没有被充分利用, 共需使用 18 个模台来承载模具完成 30 个预制构件的加工任务。如图 5 所示, 通过采用 LDWPA 算法作为全局优化算法来优化预制构件上线序, 有 3 个模台承载 3 个模具, 模台台面空间得到了更充分的利用, 承载 30 个模具所使用模台的数量降低到 16 个, 减少了模台的使用数量。说明采用全局优化算法优化上线序对于提高模台利用率和减少模台使用数量是有效的。

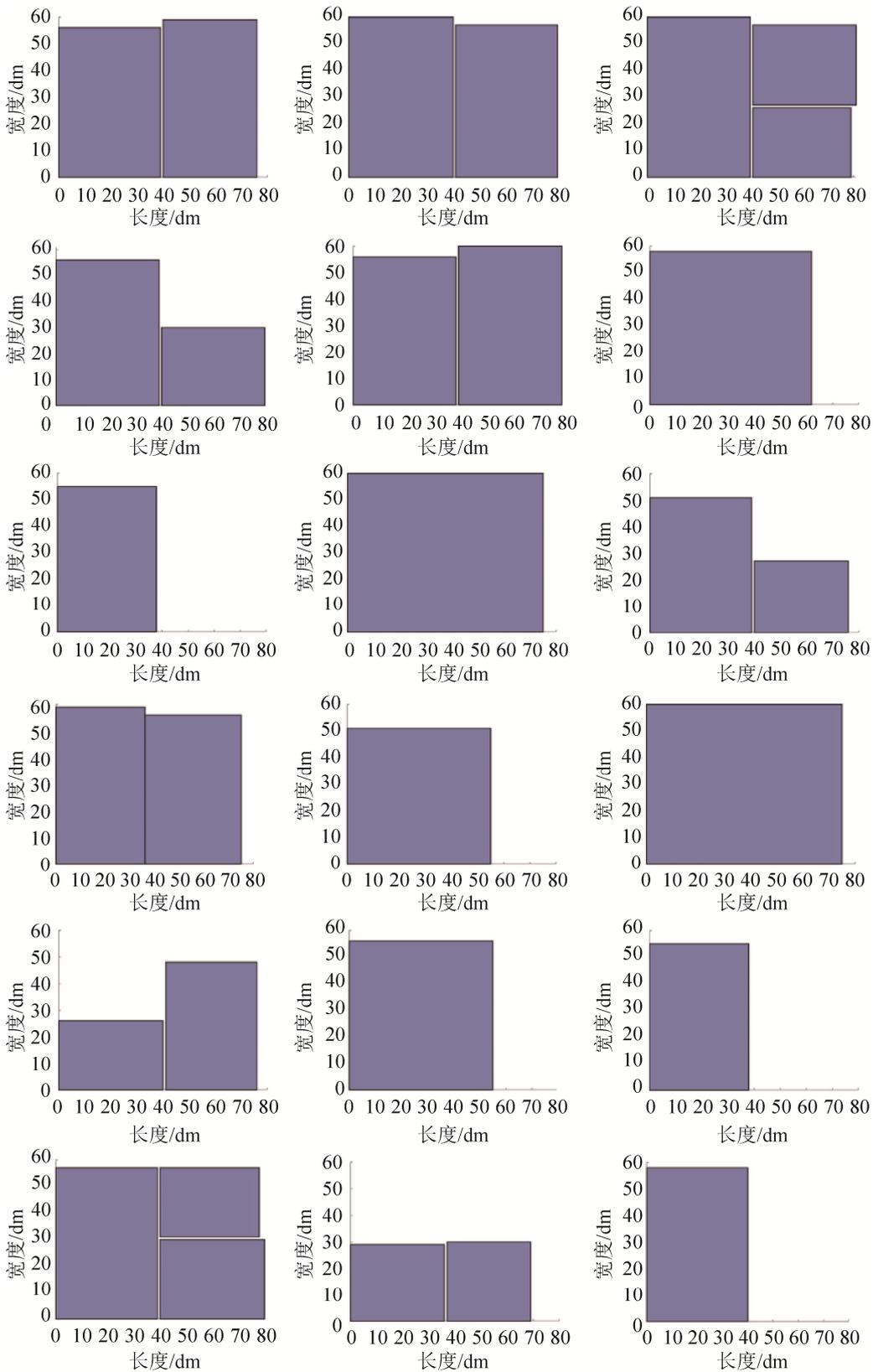


图 4 无优化算法模具模台组合分配结果简图

Fig. 4 Distribution result sketch without optimization algorithm

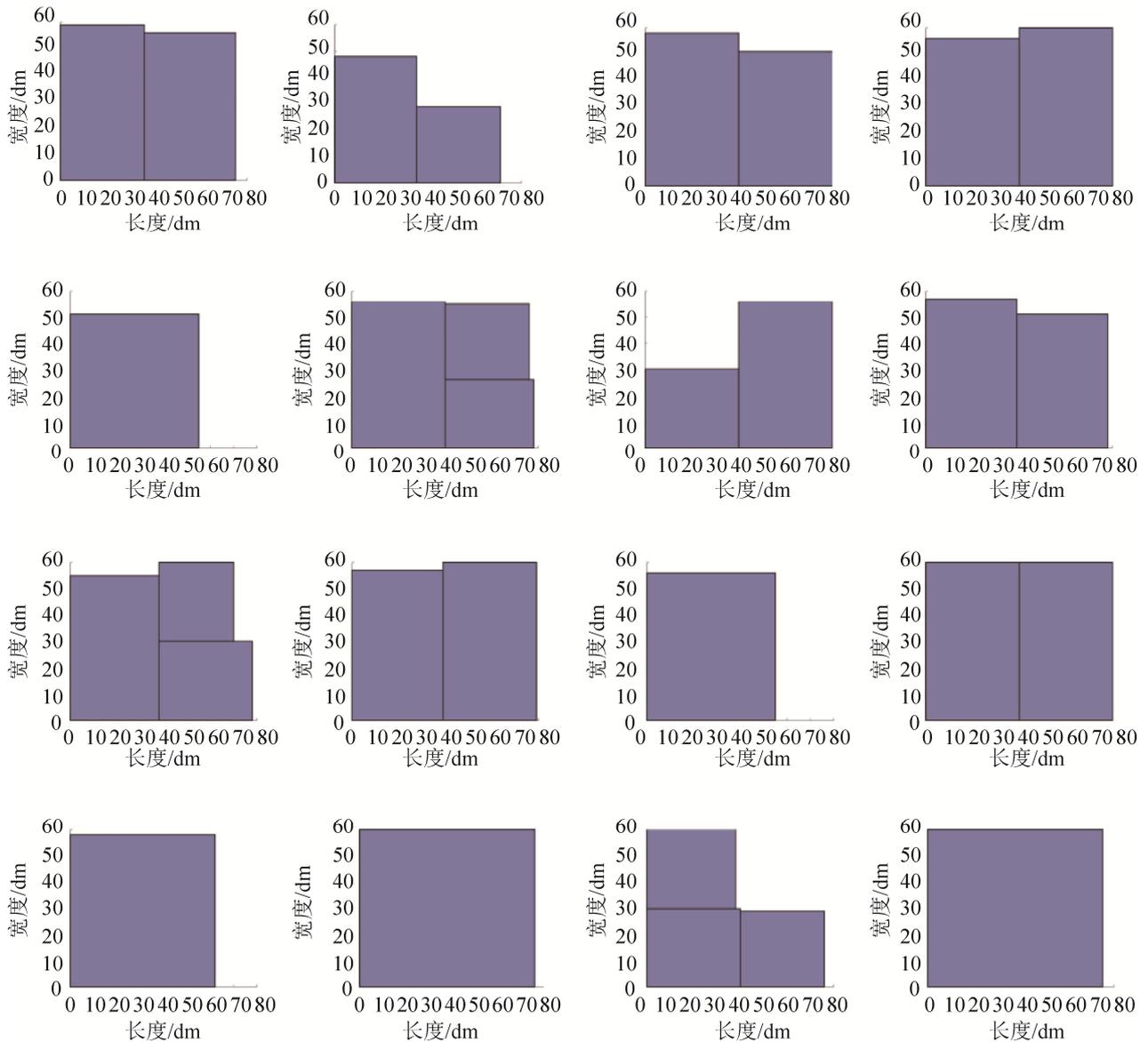


图 5 采用 LDWPA 算法的全局优化算法模具模台组合分配结果简图
Fig. 5 Distribution result sketch with LDWPA as global optimization algorithm

通过分析优化算法求解模具在模台上组合分配问题的优化过程, 来分析加入莱维飞行思想改进 WPA 的游猎行为和采用基于汉明距离的狼群动态更新机制的改进方法对提升 WPA 算法的优化性能的影响和作用。方案 2, 3, 4 中使用的 3 种群体进化算法作为全局优化算法优化投产预制构件的上线序, 实验数据采用预制构件类型编号为 1~28, 共 100 个预制构件, 选取模台利用率 R 作为优化目标, 绘制优化过程的曲线图, 如图 6 所示。

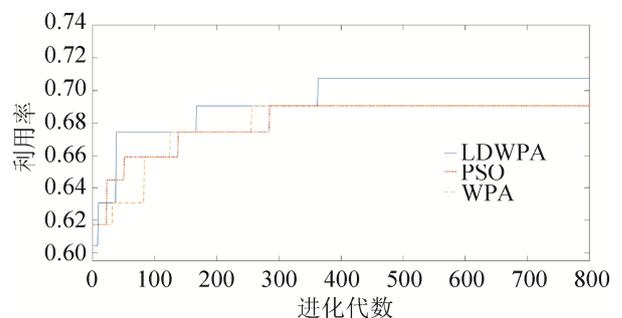


图 6 PSO, WPA 与 LDWPA 求解模具在模台上组合分配问题进化曲线图

Fig. 6 PSO, WPA and LDWPA solve molds on table distribution optimization problem evolution curve

在不同数据规模下, 方案 2, 3, 4 进行 10 次测试和方案 1 的仿真结果, 如表 3 所示。在不同数据规模下方案 1 的模台利用率 R 均低于方案 2, 3, 4 的平均模台利用率 \bar{R} , 说明采用群体进化算法优化上线序能够获得更好的模具在模台上组合分配结果。进一步分析 3 组采用群体进化算法的仿真方案的结果, 可知方案 3 与方案 2 在不同的数据规模下平均模台利用率 \bar{R} 和最低模台利用率 R_{\min} 的值接近, 说明 2 种算法在解决该类问题能够取得相似

的优化效果。方案 4 在投产预制构件数量不同的情况下, 平均模台利用率 \bar{R} 和最低模台利用率 R_{\min} 均比方案 2 和方案 3 高, 当投产预制构件的数量为 10 时, 模台平均利用率 \bar{R} 达到 0.82, 当投产预制构件的数量为 100 时, 模台平均利用率 \bar{R} 是 0.69, 是 4 组方案中的最优结果, 说明 LDWPA 算法应用在求解模具在模台上组合分配问题时能取得更好的优化效果。

表 3 4 组方案在不同数据规模下优化指标
Tab. 3 Optimization index results of 4 groups of schemes under different data scales

投产预制构件类型编号	投产预制构件数量 n	方案 1		方案 2		方案 3		方案 4	
		模台利用率 R	模台平均利用率 \bar{R}	模台最低利用率 R_{\min}	模台平均利用率 \bar{R}	模台最低利用率 R_{\min}	模台平均利用率 \bar{R}	模台最低利用率 R_{\min}	
1~4	10	0.76	0.79	0.68	0.79	0.68	0.82	0.76	
1~8	20	0.78	0.79	0.72	0.8	0.72	0.84	0.78	
1~17	30	0.74	0.81	0.72	0.82	0.74	0.84	0.79	
1~28	100	0.63	0.66	0.63	0.67	0.62	0.69	0.66	

注: 方案 1 为贪心算法; 方案 2 为采用 PSO 算法优化预制构件上线序; 方案 3 为采用 WPA 算法优化预制构件上线序; 方案 4 为用 LDWPA 算法优化预制构件上线序

为进一步分析 LDWPA 算法求解模具在模台上组合分配问题的优化效果, 在投产预制构件数量 $n=100$, 4 组方案分别进行 10 次实例测试, 其结果的分布图如图 7 所示。

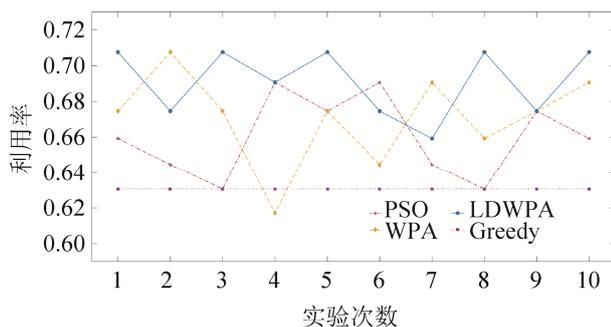


图 7 4 组方案 10 次实例测试对比

Fig. 7 Comparison of 10 case test results in 4 groups

方案 1 得到的是一个确定的结果, 方案 2 没有模台利用率超过 0.7 的结果, 最优结果同最差结果相差 6%, 测试结果的波动性较大。方案 3 有 6 次模台利用率 R 超过 0.66, 比方案 2 多 2 次, 但在第 4 次测试得到的模台利用率 R 为 0.61, 是 4 组方案

全部测试结果中的最低值, 说明方案 3 的测试结果波动性也较大。方案 4 的模台利用率 R 波动幅度最小, 最优结果与最差结果相差 4%, 利用率 R 均高于 0.66, 有 5 次超过 0.7, 且模台的平均利用率 \bar{R} 在 4 组方案中最高, 达到 0.69, 说明 LDWPA 算法应用在求解模具在模台上组合分配问题时, 优化结果具有很好的稳定性。

4 结论

本文以混凝土预制构件生产过程中模具在模台上组合分配问题为研究对象, 采用 BL 定位算法控制模具在模台上的摆放方法, 同时提出一种改进的狼群算法优化投产预制构件的上线序, 两种方法结合对该类问题进行求解, 引入莱维飞行改进狼群算法的游猎行为, 增强个体的搜索性能和进化活力, 引入基于汉明距离种群动态更新机制, 增强整个群体在解空间的分布性, 通过对狼群算法进行改进, 进一步提高狼群算法整体的优化性能, 建立 4

组仿真方案对比分析,并分别测试方法在不同数据规模下的求解效果,测试结果表明,BL 定位算法控制模具在模台上摆放与改进狼群算法优化投产预制构件上线顺序相结合的方法,能够有效的解决混凝土预制构件生产过程中模具在模台上组合分配问题。

参考文献:

- [1] 李永敢. 装配式建筑预制混凝土构件自动化生产线设计配置技术[J]. 施工技术, 2018, 47(4): 47-51.
Li Yonggan. Design Configuration Technology of Automatic Production Line for Prefabricated Concrete Member of Precast Building [J]. Construction Technology, 2018, 47(4): 47-51.
- [2] 张晓薇. 装配式混凝土小型构件自动化生产施工技术[J]. 建筑安全, 2019, 34(2): 41-44.
Ma Xiaowei. Construction Technology for Automation Production of Small Assembled Concrete Components [J]. Construction Safety, 2019, 34(2): 41-44.
- [3] 廖礼平. 绿色装配式建筑发展现状及策略[J]. 企业经济, 2019, 38(12): 139-146.
Liao Liping. Development status and strategy of green prefabricated building[J]. Enterprise Economy, 2019, 38(12): 139-146.
- [4] 李上志, 金伟. 建筑预制构件的工业化生产[J]. 四川建筑, 2018, 38(3): 210-211.
Li Shangzhi, Jin Wei. Industrial Production of Building Prefabricated Components[J]. Sichuan Architecture, 2018, 38(3): 210-211.
- [5] 谢思聪, 陈小波. 基于多层编码遗传算法的两阶段装配式建筑预制构件生产调度优化[J]. 工程管理学报, 2018, 32(1): 18-22.
Xie Sicong, Chen Xiaobo. Two Phase Job-shop Scheduling Optimization for Prefabricated Construction Projects Based on Multilayer Coding Genetic Algorithm[J]. Journal of Engineering Management, 2018, 32(1): 18-22.
- [6] 朱敏涛. 可扩展组合式预制构件数字化生产线的研究[J]. 建筑施工, 2018, 40(6): 1026-1028.
Zhu Mintao. Research on Digitalized Production Line of Extendable Modular Precast Components[J]. Building Construction, 2018, 40(6): 1026-1028.
- [7] 王中原, 陈小波. 基于遗传算法的板类预制构件生产中的模具排布优化研究[J]. 工程管理学报, 2019, 33(1): 45-49.
Wang Zhongyuan, Chen Xiaobo. An Optimization Analysis of Prefabricated Component Production Mold Arrangement Based on Genetic Algorithm[J]. Journal of Engineering Management, 2019, 33(1): 45-49.
- [8] 马龙, 卢才武, 顾清华. 求解离散优化问题的元胞量子狼群演化算法[J]. 智能系统学报, 2018, 13(5): 716-727.
Ma Long, Lu Caiwu, Gu Qinghua. Cellular and Quantum-behaved Wolf Pack Evolutionary Algorithm for Solving Discrete Optimization Problems[J]. CAAI Transactions on Intelligent Systems, 2018, 13(5): 716-727.
- [9] 黄海松, 任竹鹏, 魏建安. 改进狼群算法求解旅行商问题[J]. 计算机应用研究, 2019, 36(12): 3644-3646.
Huang Haisong, Ren Zhupeng, Wei Jianan. Improved Wolf Group Algorithm for Solving Traveling Salesman Problem[J]. Application Research of Computers, 2019, 36(12): 3644-3646.
- [10] 吴虎胜, 张凤鸣, 吴庐山. 一种新的群体智能算法—狼群算法[J]. 系统工程与电子技术, 2013, 35(11): 2430-2438.
Wu Husheng, Zhang Fengming, Wu Lushan. New Swarm Intelligence Algorithm-Wolf Pack Algorithm[J]. Systems Engineering and Electronics, 2013, 35(11): 2430-2438.
- [11] 吴虎胜, 张凤鸣, 李浩, 等. 求解 TSP 问题的离散狼群算法[J]. 控制与决策, 2015, 30(10): 1861-1867.
Wu Husheng, Zhang Fengming, Li Hao, et al. Discrete Wolf Pack Algorithm for Traveling Salesman Problem[J]. Control and Decision, 2015, 30(10): 1861-1867.
- [12] Hui X B, Guo Q, Wu P P, et al. An Improved Wolf Pack Algorithm[J]. Control and Decision (S1001-0920), 2017, 32(7): 1163-1172.
- [13] Chen X Y, Tang C J, Wang J, et al. Improved Wolf Pack Algorithm Based on Differential Evolution Elite Set [J]. IEICE Transactions on Information and Systems (S1745-1361), 2018, 101(7): 1946-1949.
- [14] 王学武, 严益鑫, 顾幸生. 基于莱维飞行粒子群算法的焊接机器人路径规划[J]. 控制与决策, 2017, 32(2): 373-377.
Wang Xuewu, Yan Yixin, Gu Xingsheng. Welding Robot Path Planning Based on Levy-PSO[J]. Control and Decision, 2017, 32(2): 373-377.
- [15] 刘晓龙, 宁芊, 赵成萍. 基于莱维飞行的鸟群优化算法[J]. 计算机测量与控制, 2016, 24(12): 194-197.
Liu Xiaolong, Ning Qian, Zhao Chengping. Bird Swarm Algorithm Based on Levy Flight[J]. Computer

- Measurement & Control, 2016, 24(12): 194-197.
- [16] Ehsan V. A Cuckoo Search Algorithm by Levy Flights for Solving Reliability Redundancy Allocation Problems [J]. Engineering Optimization (S1029-0273), 2013, 45(11): 1273-1286.
- [17] Mantegna R N. Fast, Accurate Algorithm for Numerical Simulation of Lévy Stable Stochastic Processes[J]. Phys Rev E Stat Phys Plasmas Fluids Relat Interdiscip Topics (S1063-651X), 1994, 49(5): 4677-4683.
- [18] 丛培强, 李梁, 陈亚茹. 基于汉明距离与免疫思想的粒子群算法[J]. 重庆理工大学学报(自然科学), 2019, 33(4): 122-127.
Cong Peiqiang, Li Liang, Chen Yaru. Particle Swarm Optimization Algorithm Based on Optimization Hamming Distance and Immune Thought[J]. Journal of Chongqing University of Technology (Natural Science), 2019, 33(4): 122-127.