

混合三维EDA求解带二维装载约束的车辆配送与分布式生产集成调度问题

孙蓉洁¹, 钱斌^{1,2†}, 胡蓉^{1,2}, 张大骋¹, 向凤红¹

(1. 昆明理工大学 信息工程与自动化学院, 云南 昆明 650500; 2. 昆明理工大学 云南省人工智能重点实验室, 云南 昆明 650500)

摘要: 针对一类广泛存在的带二维装载约束的车辆配送与分布式生产集成调度问题(VD2LDPISP), 本文建立问题模型, 并提出混合三维分布估计算法(H3DEDA)进行求解. 首先, 结合问题各阶段特性, 采用各阶段成本均衡策略设计新颖的解码规则, 对编码个体进行分阶段解码, 可确定较高质量的解码个体. 其次, 采用三维分布估计算法(3DEDA)学习和积累种群中优质编码个体的块结构及其位置信息, 再通过采样3DEDA中的概率模型生成新的编码个体, 从而提高算法全局搜索发现解空间中优质解区域的能力. 然后, 设计高低分层的超启发式局部搜索(HHLS)来增强算法的局部寻优能力. HHLS的低层问题域包含分别针对编码个体、配送阶段解码子个体和生产阶段解码子个体的共16种有效邻域操作, 其高层策略域采用概率模型学习优质邻域操作排列的结构信息, 进而通过采样该模型来直接控制新邻域操作排列的生成, 有利于对不同优质区域进行深入搜索. 最后, 在不同规模测试问题上的算法比较, 验证了所提H3DEDA的有效性.

关键词: 集成调度; 二维装载; 车辆配送; 分布式生产; 三维分布估计算法

引用格式: 孙蓉洁, 钱斌, 胡蓉, 等. 混合三维EDA求解带二维装载约束的车辆配送与分布式生产集成调度问题. 控制理论与应用, 2023, 40(5): 903–912

DOI: 10.7641/CTA.2022.11195

Hybrid three-dimensional estimation of distribution algorithm for vehicle distribution with two-dimensional loading constraints and distributed production integrated scheduling problem

SUN Rong-jie¹, QIAN Bin^{1,2†}, HU Rong^{1,2}, ZHANG Da-cheng¹, XIANG Feng-hong¹

(1. Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming Yunnan 650500, China;

2. Yunnan Key Laboratory of Artificial Intelligence, Kunming University of Science and Technology, Kunming Yunnan 650500, China)

Abstract: Aiming at a kind of widely existing vehicle distribution with two-dimensional loading constraints and distributed production integrated scheduling problem (VD2LDPISP), this paper establishes the problem model and proposes a hybrid three-dimensional estimation of distribution algorithm (H3DEDA) to solve it. Firstly, combining with the characteristics of each stage of the problem, a novel decoding rule is designed by using the cost balance strategy of each stage. The coding individual is decoded in stages, and the decoding individual with high quality can be determined. Secondly, the three-dimensional estimation of distribution algorithm (3DEDA) is used to learn and accumulate the block structure and location information of high-quality coding individuals in the population, and generates new coding individuals by sampling the probability model in 3DEDA, which can improve the ability of the algorithm to find high-quality solution regions in the solution space globally. Then, the hyper-heuristic local search (HHLS) is designed to enhance the local optimization capability of the algorithm. The HHLS low-level problem domain contains 16 effective neighborhood operations for coding individuals, decoding sub-individuals in distribution and production phase. It is high-level policy domain, by using the probability model learning quality neighborhood operation arrangement of information structure, and then by sampling the model to directly control the new neighborhood operation arrangement, it is conducive to in-depth search of different high-quality areas. Finally, the effectiveness of the proposed H3DEDA is verified by comparison of algorithms on different scale test problems.

Key words: integrated scheduling; two-dimensional loading; vehicle distribution; distributed production; three-dimensional estimation of distribution algorithm

收稿日期: 2021–12–07; 录用日期: 2022–04–30.

†通信作者. E-mail: bin.qian@vip.163.com; Tel.: +86 13312529481.

本文责任编辑: 贾英民.

国家自然科学基金项目(62173169, 61963022), 云南省基础研究重点项目(202201AS070030)资助.

Supported by the National Natural Science Foundation of China (62173169, 61963022) and the Basic Research Key Project of Yunnan Province (202201AS070030).

Citation: SUN Rongjie, QIAN Bin, HU Rong, et al. Hybrid three-dimensional estimation of distribution algorithm for vehicle distribution with two-dimensional loading constraints and distributed production integrated scheduling problem. *Control Theory & Applications*, 2023, 40(5): 903 – 912

1 引言

随着经济全球化的发展,为提高生产效率和资源利用率,多工厂协同生产的方式变得越来越普遍,生产制造活动由以往的集中式转变为分布式^[1].同时,由分布式工厂加工的工件(原材料、半成品等)通常需要在原材料仓库进行装载后,再通过车辆配送至工厂加工.此外,由于易碎、易损工件之间不能相互叠放,故只能对该类工件进行二维的装载^[2].在上述背景下,带二维装载约束的车辆配送与分布式生产集成调度问题(vehicle distribution with two-dimensional loading constraints and distributed production integrated scheduling problem, VD2LDPISP)的重要性日益凸显.

基于分布式生产的应用价值与理论复杂性,其受到了研究者的广泛关注与研究^[3-6].此外,将待加工工件配送至分布式工厂的过程,可归结为带装载约束的车辆路径问题.目前,针对带装载约束的车辆路径问题的研究主要包括如下4类.第1类为单车场、单车型的带装载约束的车辆运输问题^[7-8];第2类为单车场、多车型的带装载约束的车辆运输问题^[9];第3类为多车场、单车型的带装载约束的车辆运输问题^[10-11];第4类为多车场、多车型的带装载约束的车辆运输问题^[12].从上述文献调研可知,目前对带装载约束的车辆路径问题的相关研究都集中在最终产品的装载与配送至客户阶段,而对于将待加工工件装载后配送至各工厂加工这一集成过程尚无相关研究.因此,建立VD2LDPISP问题模型,并设计有效算法进行求解.

分布估计算法(estimated of distribution algorithm, EDA)是一种基于统计学习技术的智能算法^[13-14].近年来基于EDA的算法已经成功被应用于求解各类组合优化问题^[15-17].然而,现多数基于EDA的算法均使用二维概率模型,这使其无法同时学习和积累块结构的序信息与位置信息,而采用三维概率模型能够同时记录两者信息,继而在采样生成新个体时能较准确地确定优质块结构在解中的具体位置,从而可更合理地引导种群进化方向.此外,算法缺乏对局部搜索的合理引导,会使其搜索深度有限.故本文采用基于三维概率模型的EDA为基本框架,设计合理的局部搜索引导策略以实现全局搜索发现的优质解空间实现深入的搜索.

综上,本文针对VD2LDPISP,建立以最小化装载、配送与生产阶段成本的加权和为优化目标的VD2LDPISP模型.设计混合三维分布估计算法(hybrid three-dimensional estimation of distribution algorithm, H3DEDA)进行求解.首先,在算法的编解码阶段,根

据问题结构特点,设计独特的解码规则,对以工件为基本单位的编码个体进行分阶段解码,以得到包含各阶段信息的解码矩阵.然后,H3DEDA全局搜索阶段采用基于三维概率模型的3DEDA学习和积累优质个体信息并采样生成新种群个体,引导算法到达优质解空间.局部搜索阶段设计高低分层的超启发式局部搜索(hyper-heuristic local search, HHLS),同时在生产阶段采用快速评价机制加快算法搜索速度^[13].具体来说,高层采用二维概率模型学习优质邻域结构信息并采样生成新的邻域操作排列,进而在低层将高层确定的每个排列中的邻域操作按阶段依次执行,有利于对不同优质区域进行深入搜索.最后,通过仿真实验和算法比较验证了H3DEDA的有效性.

2 问题建模与分析

2.1 符号定义

本文所涉及的有关数学符号定义如表1所示.

2.2 问题模型

VD2LDPISP可描述为: n 个工件通过 H 辆车由原材料仓库配送至 F 个处于不同地理位置的工厂进行加工后产出最终产品.VD2LDPISP示意图如图1所示,整个过程分为3个阶段:装载阶段、配送阶段和生产阶段.在装载阶段,每辆车均有一个长为 L 、宽为 W 、最大载重为 Q 的矩形装载空间,且设置装载空间的左上角作为原点 $(0, 0)$,水平向右为横坐标,垂直向下为纵坐标,所有工件在不违反车辆长、宽和载重的约束前提下进行装载.在配送阶段,各车辆以恒定的行驶速度 v 同时从原材料仓库出发,按照计划路线将所装载的待加工工件配送到相应工厂后返回原材料仓库.在生产阶段,各工厂的配置相同,均为流水线加工过程,每个工厂包括 m 台机器,工件需按照固定的机器加工顺序进行流水线加工.

装载阶段装入车辆 h 的第 i 个工件 $\pi_h(i)$ 的装载约束条件和装入车辆 h 的任意两个工件 $\pi_h(k)$ 与 $\pi_h(k')$ 的约束条件如下:

$$0 \leq \pi_h^x(i) \leq W - w(\pi_h(i)),$$

$$i = 1, \dots, N_h, h = 1, \dots, H, \quad (1)$$

$$0 \leq \pi_h^y(i) \leq L - l(\pi_h(i)),$$

$$i = 1, 2, \dots, N_h, h = 1, \dots, H, \quad (2)$$

$$\pi_h^x(k') + w(\pi_h(k')) \leq \pi_h^x(k),$$

$$k, k' = 1, 2, \dots, N_h, h = 1, 2, \dots, H, k \neq k', \quad (3)$$

$$\pi_h^y(k') + l(\pi_h(k')) \leq \pi_h^y(k),$$

$$k, k' = 1, 2, \dots, N_h, h = 1, 2, \dots, H, k \neq k', \quad (4)$$

$$\sum_{i=1}^{N_h} q(\pi_h(i)) \leq Q, h = 1, 2, \dots, H. \quad (5)$$

配送阶段车辆 h 到达路径点 $\lambda_h(f)$ 的时间 $T_{\lambda_h(f)}$ 和属于路径点 $\lambda_h(f)$ 的工件的到达时间 $T_{\pi_h(i)}$, 车辆总行驶路径长度 ς , 计算方式如下:

$$Dt_{f,f'}^h = \begin{cases} 1, & \text{车辆}h\text{从}f\text{行驶至}f', \\ 0, & \text{否则,} \end{cases} \quad (6)$$

$$f, f' = 0, \dots, F, h = 1, \dots, H, \quad (6)$$

$$T_{\lambda_h(1)} = 0, h = 1, \dots, H, \quad (7)$$

$$T_{\lambda_h(f)} = T_{\lambda_h(f-1)} + D_{\lambda_h(f-1), \lambda_h(f)} / v, \quad (8)$$

$$h = 1, \dots, H, f = 0, 1, \dots, F,$$

$$T_{\pi^{\lambda_h(f)}(i)} = T_{\lambda_h(f)}, \forall \pi^{\lambda_h(f)}(i) \in \pi_h, \quad (9)$$

$$h = 1, 2, \dots, H, i = 1, \dots, n_{\lambda_h(f)}, f = 1, \dots, F,$$

$$\varsigma = \sum_{h=1}^H \sum_{f=0}^F \sum_{f'=0}^F (Dt_{f,f'}^h \times D_{f,f'}). \quad (10)$$

生产阶段工厂 f 完工时间 C_{\max}^f 计算方式如下:

$$S_{\pi^f(1),1} = T_{\pi^f(1)}, f = 1, \dots, F, \quad (11)$$

$$S_{\pi^f(1),j} = S_{\pi^f(1),j-1} + p_{\pi^f(1),j-1}, \quad (12)$$

$$j = 2, \dots, m, f = 1, \dots, F,$$

$$S_{\pi^f(i),1} = \max(S_{\pi^f(i-1),1} + P_{\pi^f(i-1),1}, T_{\pi^f(i)}), \quad (13)$$

$$i = 2, \dots, n_f, f = 1, \dots, F,$$

$$C_{\pi^f(i),j} = S_{\pi^f(i),j} + P_{\pi^f(i),j}, i = 1, \dots, n_f, \quad (14)$$

$$j = 1, \dots, m, f = 1, \dots, F,$$

$$S_{\pi^f(i),j} = \max(C_{\pi^f(i),j-1}, C_{\pi^f(i-1),j}), \quad (15)$$

$$i = 2, \dots, n_f, j = 2, \dots, m, f = 1, \dots, F,$$

$$C_{\max}^f = \max(C_{i,m}), \quad (16)$$

$$i = 1, \dots, n_f, f = 1, \dots, F,$$

$$C_{\max} = \max(C_{\max}^f), f = 1, 2, \dots, F. \quad (17)$$

VD2LDPISP总成本计算方式如下:

$$TC = \alpha H \times hc + \eta \times \varsigma + tc + \beta C_{\max} \times pc. \quad (18)$$

VD2LDPISP的优化目标为在装载、配送、生产3阶段的集成调度过程中找到一个最优排序 π^* , 使得总成本TC最小, 其如下所示:

$$\pi^* = \arg\{TC\} \rightarrow \min. \quad (19)$$

其中: 式(1)–(2)为工件不可超过车厢边界, 式(3)–(4)为同一车厢中工件之间不可叠放, 式(5)为车辆的载重约束, 式(6)用于标识车辆是否从 f 行驶至 f' , 式(7)用于计算所有车辆到达第1个路径点的时间, 式(8)用

于计算车辆到达路径点的时间, 式(7)–(9)用于计算工件到达工厂的时间, 式(10)用于计算配送阶段车辆行驶路径总长度, 式(11)–(16)用于计算工厂的完工时间, 式(17)用于计算所有工件的最大完工时间, 式(18)用于计算总成本.

表 1 符号表

Table 1 Notations

符号	说明
索引	
i	工件编号, $i = \{1, 2, \dots, n\}$.
j	机器编号, $j = \{1, 2, \dots, m\}$.
f	工厂及原材料仓库编号, $f = \{0, 1, \dots, F\}$, 其中, 点0表示原材料仓库.
h	车辆编号, $h = \{1, 2, \dots, H\}$.
参数	
$l(i)$	工件 i 的长度.
$w(i)$	工件 i 的宽度.
$P_{i,j}$	工件 i 在机器 j 上加工时间.
L	车辆车厢长度.
W	车辆车厢宽度.
Q	车辆最大载重.
v	车辆行驶速度.
hc	单位车辆调用成本.
tc	单位路径成本.
pc	单位时间成本.
$D_{f,f'}$	f 到 f' 的距离.
α	车辆调用成本的权重.
η	路径成本的权重.
β	时间成本的权重.
变量	
π	编码个体, $\pi = \{\pi(1), \dots, \pi(n)\}$.
n_f	工厂 f 中总加工工件数.
H	装载阶段使用车辆总数.
F_h	车辆 h 行驶路径点总数.
N_h	车辆 h 上装载工件总数.
π_h	车辆 h 的工件装载序, $\pi_h = \{\pi_h(1), \dots, \pi_h(N_h)\}$.
$\pi_h^x(i)$	工件 $\pi_h(i)$ 左上角到原点(0, 0)的水平距离.
$\pi_h^y(i)$	工件 $\pi_h(i)$ 左上角到原点(0, 0)的垂直距离.
π^f	工厂 f 的加工工序, $\pi^f = \{\pi^f(1), \dots, \pi^f(n_f)\}$.
$S_{i,j}$	工件 i 在机器 j 上开工时间.
$C_{i,j}$	工件 i 在机器 j 上完工时间.
C_{\max}^f	工厂 f 的完工时间.
λ_h	车辆 h 行驶路径, $\lambda_h = \{\lambda_h(1), \dots, \lambda_h(F_h)\}$ ($\lambda_h(1)$ 与 $\lambda_h(F_h)$)均为0, 即表示原材料仓库.
T_i	工件 i 到达工厂的时间.
$T_{f,f'}^h$	车辆 h 从 f 到 f' 的行驶时间.
$T_{\lambda_h(f)}$	车辆 h 到达第 f 个路径点的时间.
TC	总成本.

2.3 问题特点分析和求解算法设计

根据VD2LDPISP的问题模型及图1可知, VD2LDPISP包含装载阶段、配送阶段和生产阶段, 这3个阶段对应的子问题间存在明显的影响关系. 装载阶段子问题(二维装箱问题)工件装载情况影响配送阶段子问题(单车场多车辆配送问题)中车辆配送工件到达工厂的时间, 进而影响生产阶段子问题(分布式流水车间调度问题)中工件的完工时间. 也就是说, 前一阶段子问题的任意一个解, 均对应一个不同的后一阶段子问题, 即VD2LDPISP的解空间大小为3个子问题解空间大小相乘, 而不是三者的简单相加. 这表明VD2LDPISP的解空间庞大, 且各阶段子问题间存在耦合, 难以分割出独立子问题进行优化.

因此, 为求解这类大规模、多阶段耦合的非确定性多项式(non-deterministic polynomial, NP)难问题, 本文在设计H3DEDA时, 需先设计综合考虑各阶段成本均较小的解码规则, 来确定种群中每个编码个体对应在各子问题上的较好解. 这有利于局部搜索将部分搜索集中在解码子个体对应的较优子区域中进行, 从而可合理限定搜索空间并一定程度上实现各阶段间的局部解耦. 进而, 设计可有效学习和积累优编码个体信息的三维分布估计算法(3DEDA), 用于引导算法全局搜索较快到达解空间中的优质解区域. 为增强算法局部搜索能力, 进一步设计可动态控制邻域操作组合的HHLS, 用于对全局搜索发现且由解码规则确定的优质解区域和部分优质解子区域进行深入搜索. 基于上述设计, 第3节给出H3DEDA的具体细节.

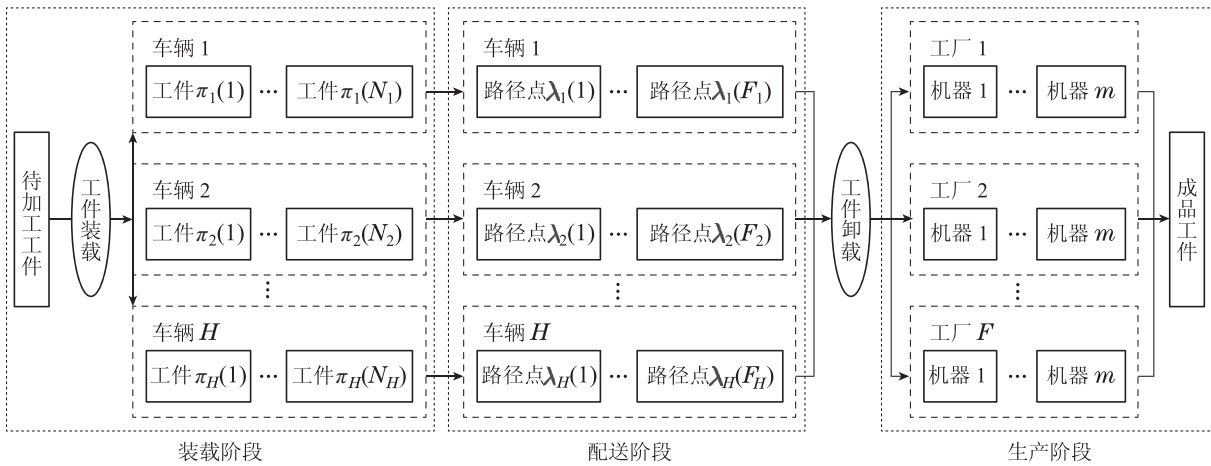


图1 VD2LDPISP示意图
Fig. 1 Diagram of VD2LDPISP

3 H3DEDA

在H3DEDA中, 首先针对编码个体进行评价, 以判断出编码个体的优劣. 然后在全局搜索阶段, 采用3DEDA学习和积累优质编码个体的信息, 进而引导编码个体的更新. 在局部搜索阶段, 设计HHLS分阶段对编码个体以及解码个体进行变邻域局部搜索, 进而发现存在于复杂解空间中的优质解.

3.1 装载策略

在装载阶段, 基于同工厂工件优先装载的规则, 采用工件最宽边越长装载优先级越高的启发式规则和以二叉树的形式递归处理装载空间的方式^[18], 设计了一种二维装载策略, 将每个矩形工件装入矩形装载空间, 可在提高车辆装载率同时有益于其他阶段的决策优化. 装载策略步骤如下:

步骤1 初始化工厂 f 的当前装载标志 $f(\varepsilon) = 0$, $f \in \{1, \dots, F\}$, 工件 $\pi^f(i)$ 的装载标志 $\delta(\pi^f(i)) = 0$, $i \in \{1, \dots, n_f\}$, 当前用于装载的车辆编号 $h = 0$,

当前装载工厂编号 $f = 0$.

步骤2 将分配至工厂 f 的加工工序 π^f 按照工件最宽边进行降序排序.

步骤3 令 $h = h + 1$, $N_h = 0$, $f = 0$; 初始化车辆 h 的装载空间, 即车辆 h 的装载二叉树的根节点 rt , 横坐标 $x(rt) = 0$, 纵坐标 $y(rt) = 0$, 长 $w(rt) = W$, 宽 $l(rt) = L$, 以及使用位标志 $rt(\varpi) = 0$.

步骤4 根据当前装载结果, 对所有工厂 $f(\varepsilon)$ 进行赋值, $f \in \{1, \dots, F\}$. 若 $\sum_{i=1}^{n_f} \delta(\pi^f(i)) = n_f$, 则 $f(\varepsilon) = 2$; 若 $\sum_{i=1}^{n_f} \delta(\pi^f(i)) = 0$, 则 $f(\varepsilon) = 0$; 否则, $f(\varepsilon) = 1$.

步骤5 选择一个 $f(\varepsilon) \neq 2$ 的工厂 f , $f \in \{1, \dots, F\}$, 并对 π^f 进行装载. 若存在工厂 $f(\varepsilon) = 1$, 则转至步骤6; 否则, 选择工厂 f' , $f' \in \{1, \dots, F\}$, 使得 $D_{f,f'} = \min(D_{f,f'})$, $f(\varepsilon) \neq 2$, 则 $f = f'$, 转至步骤6. 若所有工厂 $f(\varepsilon) = 2$, 则表示所有工件装载完毕.

步骤6 若 $\pi^f(i)$ 满足 $\delta(\pi^f(i)) = 0$, $i \in \{1, \dots,$

$n_f\}$, 则将工件 $\pi^f(i)$ 装入车辆 h . 优先选择右侧装载空间, 确定装载空间节点信息 $pr(\varpi) = 0$, 若 $\pi^f(i)$ 满足

$$q(\pi^f(i)) + \sum_{j=1}^{N_h} q(\pi_h(j)) > Q,$$

$$l(\pi^f(i)) + \pi_h^y(\pi^f(i)) > L,$$

$$w(\pi^f(i)) + \pi_h^x(\pi^f(i)) > W$$

或

$$\pi_h^y(\pi^f(i)) + l(\pi^f(i)) > L,$$

则令 $i = i + 1$, 若 $i > n_f$, 则转至步骤3, 否则转至步骤6; 否则置 $\delta(\pi^f(i)) = 1$, 令 $\pi_h(N_h + 1) = \pi^f(i)$, 然后转至步骤7.

步骤7 置 $N_h = N_h + 1, j = N_h$, 更新 $\pi_h(j)$ 装入的 pr 节点信息, 先更新 pr 右子节点即右侧装载空间 pr^R 信息 $l(pr^R) = l(pr), y(pr^R) = y(pr), x(pr^R) = x(pr) + w(\pi_h(j)), w(pr^R) = w(pr) - w(\pi_h(j)), pr^R(\varpi) = 0$; 然后更新 pr 左子节点即下侧装载空间 pr^L 信息 $y(pr^L) = y(pr) - l(\pi_h(j)), l(pr^L) = l(pr) - l(\pi_h(j)), x(pr^L) = x(pr), w(pr^L) = w(pr), pr^L(\varpi) = 0$; 最后更新 pr 信息 $l(pr) = l(\pi_h(j)), w(pr) = w(\pi_h(j)), pr(\varpi) = 1$, 且工件 $\pi_h(j)$ 在车辆 h 的位置信息即为 pr 的位置信息. 若 $\pi_h(j) = \pi^f(n_f)$, 则转至步骤4; 否则, 转至步骤6.

3.2 编码与解码

在H3DEDA中, 编码个体由各工件按一定顺序排列而成, 如图2解码过程示意图中所示的编码个体.

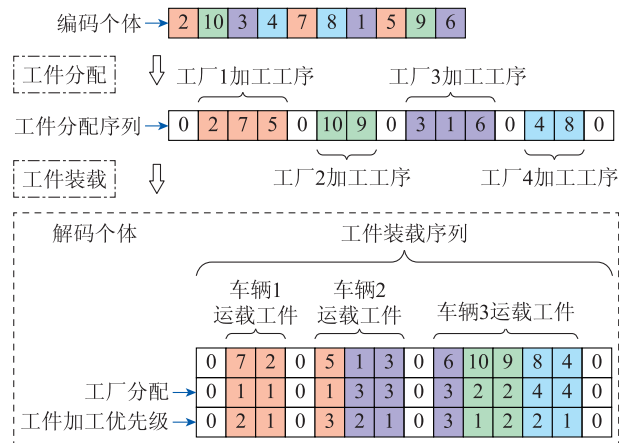


图2 解码过程示意图

Fig. 2 Diagram of decoding process

由于VD2LDPISP各阶段相互影响, 常规解码方式无法直接适配于该问题, 因此, 本文根据问题各阶段特性, 采用各阶段成本均衡策略设计新颖的解码规则, 首先, 该解码方式在生产阶段采用Rule2^[3]将工件分配至对应工厂, 并确定各工厂中工件加工优先级为工件

分配顺序; 然后, 在装载阶段, 依据装载策略对工件进行装载, 从而获得包含各阶段信息的解码矩阵, 即解码个体. 如图2所示, 矩阵第1行由各车辆工件装载序列 π_h 组成, 相邻两车辆间用数字0隔开. 矩阵第2行与第3行分别为工件分配序列与工件加工优先级, 同时, 结合第一行即可获得各工厂加工序列 π^f . 此外, 依据解码矩阵中各车辆内所装载工件先后顺序与工件所属工厂相关信息, 可确定车辆配送路径 λ_h .

3.3 种群初始化

本文采用随机初始化方法, 以保证解的多样性和分散性.

3.4 3DEDA

本文提出了一种基于三维概率模型的分布估计算法对种群中编码个体进行优化. 该算法中的三维概率模型能准确记录由相邻两工件所构成块结构的分布特征, 可更好地保护较优解模式不被破坏, 进而引导种群的更新方向.

3.4.1 块结构

对于编码个体而言, 将整个序列中连续相邻的两个工件视为一个块结构.

3.4.2 三维概率模型

定义 $P(G)$ 为第 G 代的种群, $P_B(G)$ 为 $P(G)$ 中的优质解, $P_B(G) = \{\Phi_B^{G,1}, \Phi_B^{G,2}, \dots, \Phi_B^{G,bps}\}$, ps 为 $P(G)$ 的规模, bps 为 $P_B(G)$ 的规模, 令 θ 为种群优质个体占比, 则 $bps = ps \times \theta$, $\Phi_B^{G,k}$ 为 $P_B(G)$ 中第 k 个优质解, $\pi_B^{G,k}$ 为 $\Phi_B^{G,k}$ 的编码个体, 编码个体长度为 n , $\pi_B^{G,k} = [\pi_B^{G,k}(1) \ \pi_B^{G,k}(2) \ \dots \ \pi_B^{G,k}(n)]$.

1) 统计矩阵.

定义三维统计矩阵 $M_{n \times n \times n}^G$ 用于储存第 G 代优质编码个体的块结构及其分布等信息.

$$I_{n \times n \times n}^{G,k}(x, y, z) = \begin{cases} 1, & \text{若 } y = \pi_B^{G,k}(x) \text{ 且 } z = \pi_B^{G,k}(x + 1), \\ 0, & \text{其他,} \end{cases}$$

$$x = 1, \dots, n-1, \ y, z = 1, \dots, n, \ k = 1, \dots, bps, \quad (20)$$

$$M_{n \times n \times n}^G(x, y, z) = \sum_{k=1}^{bps} I_{n \times n \times n}^{G,k}(x, y, z),$$

$$x = 1, \dots, n-1, \ y, z = 1, \dots, n, \quad (21)$$

其中: 式(20)中 $I_{n \times n \times n}^{G,k}(x, y, z)$ 为指示函数, 用于记录 $P_B(G)$ 中第 k 个编码个体中块结构的位置; 式(21)用于记录所有优质编码个体中工件的顺序和块结构的位置信息, $M_{n \times n \times n}^G(x, y, z)$ 为在 $P_B(G)$ 中块结构 $[y, z]$ 在 x 位置上出现的次数.

2) 概率矩阵.

本文采用基于 $M_{n \times n \times n}^G$ 的三维概率矩阵 $P_{n \times n \times n}^G$

来学习和积累优质编码个体中块结构位置信息, 令 $P_{n \times n \times n}^G(x, y, z)$ 为 $P_{n \times n \times n}^G$ 中的元素.

块结构分布在位置 x 上的概率和为

$$P_S^G(x) = \sum_{y=1}^n \sum_{z=1}^n P_{n \times n \times n}^G(x, y, z),$$

$$x = 1, \dots, n - 1. \quad (22)$$

块结构分布在位置 x 上的次数和为

$$M_S^G(x) = \sum_{y=1}^n \sum_{z=1}^n M_{n \times n \times n}^G(x, y, z),$$

$$x = 1, \dots, n - 1. \quad (23)$$

3) 三维概率模型更新.

三维概率矩阵模型更新过程如下:

步骤 1 按式(24)初始化三维概率矩阵.

$$P_{n \times n \times n}^0(x, y, z) = 1/n^2,$$

$$x = 1, \dots, n - 1, y, z = 1, \dots, n. \quad (24)$$

步骤 2 按式(25)更新第 G 代三维概率矩阵, 其中 ρ 表示三维概率模型的学习率.

$$P_{n \times n \times n}^{G+1}(x, y, z) = (1 - \rho) \times P_{n \times n \times n}^G(x, y, z) +$$

$$\rho \times M_{n \times n \times n}^G(x, y, z) / M_S^G(x),$$

$$x = 1, 2, \dots, n - 1, y, z = 1, \dots, n. \quad (25)$$

步骤 3 $G = G + 1$, 若 $G \leq \max G$, 则转步骤2.

3.5 种群更新

令 $\pi^{G,k} = [\pi(1) \ \pi(2) \ \dots \ \pi(n)]$ 为三维概率模型采样生成的第 k 个种群个体的编码个体, 由于块结构 $[\pi^{G,k}(i - 1) \ \pi^{G,k}(i)]$ 被选中的概率存储在三维概率模型 $P_{n \times n \times n}^{G-1}(i - 1)$ 中, 故 $\pi^{G,k}$ 中 $\pi(i)$ 依据三维概率模型 $P_{n \times n \times n}^{G-1}(i - 1)$ 进行采样. 新种群生成方式如下:

步骤 1 令 $k = 1$.

步骤 2 生成 $\pi^{G,k}$ 的第1个工件. 令 $i = 1$ 按式(26)计算 $S_{\text{init}}^{G-1}(y)$. 生成随机数 $r, r \in [0, \sum_{h=1}^{\text{pos}+1} S_{\text{init}}^{G-1}(h)]$, 若 r 满足 $r \in [0, N_{\text{init}}^{G-1}(h)]$, 则 $s = 1; r \in [\sum_{h=1}^{\text{pos}} S_{\text{init}}^{G-1}(h),$

$\sum_{h=1}^{\text{pos}+1} S_{\text{init}}^{G-1}(h)]$, $\text{pos} \in \{1, \dots, n-1\}$, 则 $s = \text{pos}+1, i = i + 1$.

$$S_{\text{init}}^{G-1}(y) = \sum_{z=1}^n P^{G-1}(1, y, z), y = 1, \dots, n. \quad (26)$$

步骤 3 生成剩余工件. 生成随机数 $r, r \in [0, \sum_{h=1}^n P_{n \times n \times n}^{G-1}(i - 1, \pi^{G,k}(i - 1), h)]$, 若 r 满足 $r \in [0, P_{n \times n \times n}^{G-1}(i - 1, \pi^{G,k}(i - 1), 1)]$, 则 $s = 1$, 若 r 满足 $r \in [\sum_{h=1}^{\text{pos}} P_{n \times n \times n}^{G-1}(i - 1, \pi^{G,k}(i - 1), h), \sum_{h=1}^{\text{pos}+1} P_{n \times n \times n}^{G-1}(i - 1, \pi^{G,k}(i - 1), h)]$, $\text{pos} \in \{1, \dots, n - 1\}$, 则 $s = \text{pos} + 1, i = i + 1$.

步骤 4 若 $i \leq n$, 执行步骤3, 否则输出 $\pi^{G,k}$, 令 $k = k + 1$.

步骤 5 若 $k \leq \text{ps}$, 执行步骤2, 否则输出 $P(G)$.

3.6 HHLS

对于VD2LDPISP这类复杂组合优化问题, 全局最优解在所有邻域结构下均为局部最优解, 而接近全局最优解的优质解往往在多种邻域结构下均为局部最优解. 采用单一邻域操作的迭代搜索容易较早达到并陷入对应邻域结构的局部最小, 导致算法搜索深度不够, 因此本文在HHLS低层设计多种有效的邻域操作, 并通过不同的排列构成包含多种邻域结构信息的高层邻域操作序, 来指导算法对低层个体执行超启发式局部搜索.

本文采用二维概率模型学习和积累各邻域操作间组成的优质邻域结构信息并采样生成新邻域操作序, 以引导低层个体执行分阶段局部搜索. HHLS示意图如图3所示, 其中, 1阶段局部搜索作用于低层编码个体, 2, 3阶段局部搜索分别作用于低层配送阶段解码子个体即车辆行驶路径和生产阶段解码子个体即工厂加工工序, 可实现对全局搜索发现的优质区域进行多种优质邻域结构下的细致搜索, 进而增强获取优质解的能力.

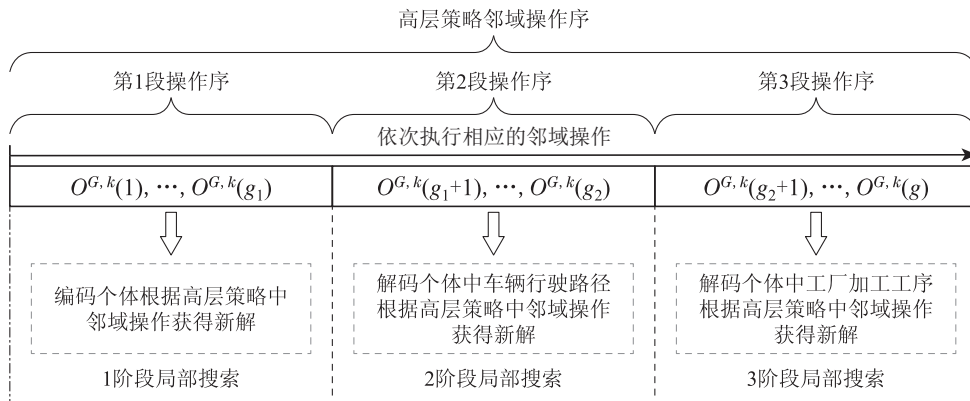


图3 HHLS示意图

Fig. 3 Diagram of HHLS

3.6.1 邻域操作

本文针对编码个体与解码个体共设计了16种邻域操作作为低层邻域操作。

1) LLH₁: 编码个体的工件交换操作. 从编码个体中选择两个工件进行位置交换。

2) LLH₂: 编码个体的工件插入操作. 先随机选择编码个体中的某一工件, 再随机选择一个工件, 然后将前者插入后者的前方。

3) LLH₃: 编码个体的工序片段插入操作. 选取某一工序片段, 将其插入序列剩余任一工件前方。

4) LLH₄: 编码个体的工序学习优秀序列操作. 将工序的一段替换为优质解中同样位置的工序片段。

5) LLH₅: 编码个体的工序片段交换操作. 从编码个体中随机选取两个工序片段进行位置互换。

6) LLH₆: 编码个体的工序片段顺序逆序操作. 从编码个体中随机选取工序片段, 进行逆转。

7) LLH₇: 车辆行驶路径交换操作. 从车辆行驶路径中随机选择两个路径点进行位置交换。

8) LLH₈: 车辆行驶路径插入操作. 先随机选择行驶路径中的某一路径点, 再在行驶路径中随机选择一个路径点, 然后将前者插入后者的前方。

9) LLH₉: 车辆行驶路径相邻路径点插入操作. 从行驶路径中选择一个路径点, 将所选路径点分别插入其左右相邻位置。

10) LLH₁₀: 车辆行驶路径片段插入操作. 随机选择一段行驶路径, 随机将其插入其余位置。

11) LLH₁₁: 工厂加工序列工件交叉操作. 选择完工时间最大的工厂, 从工厂加工序列中随机选择两个工件进行位置交换。

12) LLH₁₂: 工厂加工序列工件插入操作. 选择完工时间最大的工厂, 从工厂加工序列选择一个工件, 将其随机插入其余工件前方。

13) LLH₁₃: 工厂加工序列相邻工件插入操作. 选择完工时间最大的工厂, 从工厂加工序列选择一个工件, 将其分别插入其左右相邻位置。

14) LLH₁₄: 工厂加工工序片段逆序插入操作. 选择完工时间最大的工厂, 从工厂加工序列随机选择一个工序片段进行逆序, 将其随机插入其余位置。

15) LLH₁₅: 工厂加工工序片段插入操作. 选择完工时间最大的工厂, 从工厂加工序列随机选择一个工序片段, 将其随机插入其余位置。

16) LLH₁₆: 工厂加工工序片段交换操作. 选择完工时间最大的工厂, 从工厂加工序列中随机选择两个工序片段, 并将两者进行位置交换。

3.6.2 二维概率模型

在HD3EDA局部搜索中, 采用二维概率模型 $C_{g \times g}^G$

学习和累计高层策略域中优质邻域操作序信息, 并将两个相邻操作视为操作块. 定义 $O_B^k(G)$ 为 $\Phi_B^{G,k}$ 的高层策略邻域操作序, 序列长度为 g , $O_B^k(G) = \{O_B^{G,k}(1), \dots, O_B^{G,k}(g)\}$. $T_{g \times g}^{G,k}(x, y)$ 用于记录 $O_B^k(G)$ 中操作块 $[x \ y]$ 出现的概率, 计算过程如式(27)所示; $D_{g \times g \times g}^G(x, y)$ 为 $P_B(G)$ 的所有个体中高层策略邻域操作序中操作块 $[x \ y]$ 出现总次数, 计算过程如式(28)所示。

$$T_{g \times g}^G(x, y) = \begin{cases} 1, & \text{若 } x = O_B^{G,k}(s) \text{ 且 } y = O_B^{G,k}(s+1), \\ 0, & \text{其他,} \end{cases} \quad (27)$$

$$s = 1, \dots, g-1, \quad x, y = 1, \dots, g, \quad k = 1, \dots, \text{bps.}$$

$$D_{g \times g}^G(x, y) = \sum_{k=1}^{\text{bps}} T_{g \times g}^{G,k}(x, y), \quad x, y = 1, \dots, g. \quad (28)$$

二维概率模型的初始化如式(29)所示, 即

$$C_{g \times g}^0(x, y) = \frac{1}{g}, \quad x, y = 1, 2, \dots, g. \quad (29)$$

G 代二维概率模型的更新如式(30)所示, 其中 γ 表示二维概率模型的学习率。

$$P_{g \times g}^G(x, y) = \gamma \times P_{g \times g}^{G-1}(x, y) + (1 - \gamma) \times \frac{D_{g \times g \times g}^G(x, y)}{\sum_{y=1}^g D_{g \times g}^{G,k}(x, y)}, \quad (30)$$

$$x, y = 1, \dots, g.$$

3.6.3 高层策略域更新

本文高层策略邻域操作序通过二维概率模型采样生成, 目的是使具有优质邻域结构的操作块 $[x \ y]$ 有更大的概率产生, 最终产生优质的邻域操作序列. 为保证多样性, 采用轮盘赌操作进行采样. 将二维概率模型每一行归一化, 并进行概率累积, 得到归一化矩阵 $N_{g \times g}^G$. 令 $O^{G,k} = [O^{G,k}(1) \ O^{G,k}(2) \ \dots \ O^{G,k}(g)]$ 为 $P(G)$ 中第 k 个个体的高层策略邻域操作序. 新高层策略邻域操作序生成方式如下:

步骤 1 令 $i = 1$.

步骤 2 若 $i \leq g_1$, 则 $\text{op} \in \{1, \dots, g_1\}$; 若 $g_1 < i \leq g_2$, 则 $\text{op} \in \{g_1+1, \dots, g_2\}$; 若 $g_2 < i \leq g$, 则 $\text{op} \in \{g_2+1, \dots, g\}$. 生成随机数 $r, r \in [0, 1]$, 若 $r \leq \sum_{y=1}^{\text{op}} N_{g \times g}^G(O^{G,k}(i-1), y)$, 则 $O^{G,k}(i) = \text{LLH}_{\text{op}}, i = i+1$.

步骤 3 若 $i \leq g$, 执行步骤2, 否则输出 $O^{G,k}$.

3.7 算法流程

步骤 1 初始化种群、二维概率模型及算法参数。

步骤 2 评价初始种群个体, 并初始化三维概率模型。

步骤 3 选择当前种群bps个优质个体更新三维概率模型, 并通过采样生成下一代新种群。

步骤 4 采样产生所有种群个体的对应高层策略邻域操作序.

步骤 5 对种群的所有编码个体执行邻域操作.

步骤 6 对所有编码个体解码生成解码个体.

步骤 7 对种群的所有解码个体执行邻域操作.

步骤 8 评价种群个体.

步骤 9 选择当前种群bps个优质个体的高层策略邻域操作序更新二维概率模型.

步骤 10 判断是否满足终止条件. 若不满足则转到步骤3; 否则, 终止循环.

4 实验设计与分析

4.1 实验设置

由于目前没有合适的VD2LDPISP的标准算例, 本文所有测试算例均在由Naderi等^[3]为解决分布式置换流水车间调度问题 (distributed permutation flow-shop scheduling problem, DPFSP)所提供的测试算例的基础上生成. 本文共有32个测试算例, 其包括大小规模算例各16个, 并按照 $F \times n \times m$ 的形式组合, 并设置小规模算例运行时间为 $25 \times n \times m$ (ms); 大规模算例运行时间为 $100 \times n \times m$ (ms).

所有算法和实验均由 Visual Studio 2017 编程实现, 操作系统为 Window10, CPU为 2.90 GHz, 内存为 16 GB. 以算法独立运行 20 次输出最优结果的平均值(AVG)为性能指标的实验结果如表2-5所示. 需要说明的是, 为了节约空间, 表2-5中仅提供部分算例的相关测试结果, 更多算例下的测试结果以及其余指标的完整实验结果可以在<https://pan.baidu.com/s/1u9u-JsgWCpB3is4ebqQaxQ?pwd=cbu>中下载 (提取码为cbu).

表 2 局部搜索策略小规模算例对比结果

Table 2 Comparison results of partial search strategies on small-scale instances

问题规模	H3DEDA_V1	H3DEDA_V2	H3DEDA
	AVG	AVG	AVG
2 × 20 × 10	602.7	591.1	586.4
2 × 50 × 10	1194.1	1159.1	1148.0
2 × 50 × 20	1505.0	1457.4	1446.0
3 × 20 × 5	432.5	427.4	423.6
3 × 20 × 20	876.0	869.9	864.4
3 × 50 × 10	948.9	928.8	915.4
5 × 50 × 5	562.6	554.7	540.7
5 × 50 × 20	1155.2	1151.1	1120.7
6 × 20 × 20	826.4	827.1	820.0
6 × 50 × 10	757.5	751.9	738.6
6 × 50 × 20	1071.0	1067.0	1043.6

4.2 参数实验

H3DEDA的关键参数为三维概率模型学习率 ρ , 二维概率模型学习率 γ , 优质个体占比 η . 此外, 种群规模ps设置为20. 本节分别针对小规模问题(以 $2 \times 50 \times 20$ 为例)与大规模问题(以 $4 \times 200 \times 10$ 为例)采用实验设计方法(design of experiment, DOE)^[19]进行参数实验分析, 进而确定H3DEDA的参数取值. 经实验分析, 大小规模算例的最佳参数组合均为: $\rho = 0.5, \gamma = 0.5, \eta = 0.3$.

表 3 局部搜索策略大规模算例对比结果

Table 3 Comparison results of partial search strategies on large-scale instances

问题规模	H3DEDA_V1	H3DEDA_V2	H3DEDA
	AVG	AVG	AVG
3 × 100 × 10	1696.1	1660.2	1644.0
3 × 200 × 10	2892.4	2835.0	2822.9
3 × 200 × 20	3280.4	3223.8	3188.2
4 × 100 × 5	1138.3	1124.3	1110.9
4 × 100 × 10	1474.9	1460.8	1428.4
4 × 200 × 20	2865.9	2816.9	2792.7
5 × 100 × 10	1366.1	1353.1	1324.6
5 × 100 × 20	1633.5	1615.8	1590.9
5 × 200 × 20	2894.3	2860.9	2836.1
6 × 100 × 20	1565.9	1554.0	1532.5
6 × 200 × 10	2030.1	2010.2	1989.1

表 4 H3DEDA与3种算法的小规模算例对比结果

Table 4 Comparison results of H3DEDA and three algorithms on small-scale instances

问题规模	H2LFGGA	EBSO	AMA	H3DEDA
	AVG	AVG	AVG	AVG
2 × 20 × 10	623.6	606.2	703.5	586.4
2 × 50 × 10	1263.1	1309.2	1473.8	1148.0
2 × 50 × 20	1577.0	1625.5	1736.3	1446.0
3 × 20 × 5	439.1	433.7	455.7	423.6
3 × 20 × 20	890.6	877.7	992.7	864.4
3 × 50 × 10	989.5	978.2	1135.7	915.4
5 × 50 × 5	582.3	580.1	735.2	540.7
5 × 50 × 20	1178.8	1207.5	1338.5	1120.7
6 × 20 × 20	837.5	814.2	901.9	820.0
6 × 50 × 10	770.6	762.3	893.9	738.6
6 × 50 × 20	1084.3	1083.4	1212.1	1043.6

4.3 局部搜索策略有效性验证

为验证H3DEDA中超启发式局部搜索HHLs的有效性, 将H3DEDA与在局部搜索阶段采用按固定顺序执行邻域操作的H3DEDA_V1和随机选择邻域操作的H3DEDA_V2进行比较. 这3种算法除邻域操作序

生成策略不同外, 其他部分均相同. 由表2-3可知, 在小规模和大规模算例中, H3DEDA在大部分问题上的测试结果均明显优于H3DEDA_V1和H3DEDA_V2, 这验证了H3DEDA中基于二维概率模型的HHLS作为局部搜索策略来动态选择LLH构造高层策略域中邻域操作序的有效性.

表5 H3DEDA与3种算法的大规模算例对比结果
Table 5 Comparison results of H3DEDA and three algorithms on large-scale instances

问题规模	H2LFGGA	EBSO	AMA	H3DEDA
	AVG	AVG	AVG	AVG
$3 \times 100 \times 10$	1785.1	1839.8	2081.3	1644.0
$3 \times 200 \times 10$	3069.5	3186.1	3436.4	2822.9
$3 \times 200 \times 20$	3469.1	3570.4	3777.9	3188.2
$4 \times 100 \times 5$	1177.0	1175.6	1343.4	1110.9
$4 \times 100 \times 10$	1536.0	1578.0	1800.5	1428.4
$4 \times 200 \times 20$	3005.4	3159.9	3398.4	2792.7
$5 \times 100 \times 10$	1418.2	1437.6	1609.3	1324.6
$5 \times 100 \times 20$	1693.3	1730.4	1949.3	1590.9
$5 \times 200 \times 20$	3006.7	3086.3	3473.9	2836.1
$6 \times 100 \times 20$	1599.9	1636.6	1851.7	1532.5
$6 \times 200 \times 10$	2094.3	2206.0	2607.1	1989.1

H3DEDA中HHLS利用二维概率模型合理的学习优质邻域操作序信息进而动态引导H3DEDA搜索方向, 可一定程度上避免由于按固定顺序执行邻域操作所形成的单一的邻域结构和由于随机挑选邻域操作而形成的不合理的邻域结构而造成搜索深度不够的缺陷. 因此, H3DEDA在VD2LDPISP这类解空间十分复杂的问题中, 表现出更为优异的性能.

4.4 本文算法有效性验证

为验证H3DEDA的有效性, 将H3DEDA与自适应模因算法(adaptive memetic algorithm, AMA)^[9], 混合二级模糊引导遗传算法(hybrid 2-level fuzzy guided genetic algorithm, H2LFGGA)^[10]和增强型头脑风暴优化算法(enhanced brain storm optimization algorithm, EBSO)^[11]进行对比. 由表4-5可知, 在小规模和大规模算例中, H3DEDA在绝大部分问题上的测试结果均明显优于对比算法, 表明H3DEDA是求解VD2-LDPISP的有效算法.

H3DEDA采用基于三维概率模型的方式进行种群更新, 有利于保护优良解的结构, 从而可更合理地引导种群进化方向, 以提高算法的搜索效率. 此外, H3DEDA的局部搜索阶段, 利用二维概率模型动态控制低层个体执行包含多种邻域操作组合的分阶段局部搜索, 使得算法可在到达多种邻域结构共同的局部最优解前一直持续进行搜索, 进而能发现存在于复杂

解空间中的优质解. 因此, H3DEDA在上述实验中取得较好结果.

5 结论

本文首次建立带二维装载约束的车辆配送和分布式生产集成调度问题(VD2LDPISP)模型, 并提出一种混合三维分布估计算法(H3DEDA)进行求解. H3DEDA具有如下优点: 1) 设计更为合理的编解码规则对以工件为基本单位的编码个体进行分阶段解码, 可提高解的质量; 2) 采用3DEDA学习与积累优质编码个体的结构信息并采样生成新个体来持续引导种群的更新, 以增强算法发现优质解区域的能力; 3) 设计高低分层的超启发式局部搜索(HHLS)分阶段对编码个体与解码个体进行局部搜索. 具体来说, 在HHLS高层采用二维概率模型持续更新邻域操作序列, 进而动态确定每个低层个体所执行的邻域操作顺序, 可引导算法较快到达解空间中多个优质区域执行深入地搜索. 后续研究将把H3DEDA扩展用于求解模糊带二维装载约束的车辆配送和分布式生产集成调度问题.

参考文献:

- [1] DENG J, WANG L. A competitive memetic algorithm for multi-objective distributed permutation flow shop scheduling problem. *Swarm and Evolutionary Computation*, 2017, 32(5): 121 - 131.
- [2] JI Bin, ZHOU Saiqi, ZHANG Zheng. Branch-and-price approach for solving the vehicle routing problem with two-dimensional loading constraints. *Control Theory & Applications*, 2023, 40(3): 409 - 418. (季彬, 周赛琦, 张政. 分支定价方法求解带二维装箱约束的车辆路径问题. *控制理论与应用*, 2023, 40(3): 409 - 418.)
- [3] NADERI B, RUIZ R. The distributed permutation flowshop scheduling problem. *Computers & Operations Research*, 2010, 37(4): 754 - 768.
- [4] YANG Xiaolin, HU Rong, QIAN Bin, et al. Enhanced estimation of distribution algorithm for low carbon scheduling of distributed flow shop problem. *Control Theory & Applications*, 2019, 36(5): 803 - 815. (杨晓林, 胡蓉, 钱斌, 等. 增强分布估计算法求解低碳分布式流水线调度. *控制理论与应用*, 2019, 36(5): 803 - 815.)
- [5] LUO Q, DENG Q, GONG G, et al. An efficient memetic algorithm for distributed flexible job shop scheduling problem with transfers. *Expert Systems with Applications*, 2020, 160(12): 113721.
- [6] SHAO W, SHAO Z, PI D. Modeling and multi-neighborhood iterated greedy algorithm for distributed hybrid flow shop scheduling problem. *Knowledge-Based Systems*, 2020, 194(4): 105527.
- [7] LEUNG S C H, ZHANG Z, ZHANG D, et al. A meta-heuristic algorithm for heterogeneous fleet vehicle routing problems with two-dimensional loading constraints. *European Journal of Operational Research*, 2013, 225(2): 199 - 210.
- [8] BORTFELDT A, YI J. The split delivery vehicle routing problem with three-dimensional loading constraints. *European Journal of Operational Research*, 2020, 282(2): 545 - 558.
- [9] SABAR N R, BHASKAR A, CHUNG E, et al. An adaptive memetic approach for heterogeneous vehicle routing problems with two-dimensional loading constraint. *Swarm and Evolutionary Computation*, 2020, 58(6): 100730.
- [10] SUN X T, CHUNG S H, CHAN F T S. Integrated scheduling of a multi-product multi-factory manufacturing system with mar-

- itime transport limits. *Transportation Research Part E: Logistics and Transportation Review*, 2015, 79(7): 110 – 127.
- [11] HOU Y, FU Y, GAO K, et al. Modelling and optimization of integrated distributed flow shop scheduling and distribution problems with time windows. *Expert Systems with Applications*, 2022, 187(1): 115827.
- [12] WANG Y, LI Q, GUAN X, et al. Collaborative multi-depot pickup and delivery vehicle routing problem with split loads and time windows. *Knowledge-Based Systems*, 2021, 231(11): 107412.
- [13] ZHANG Z Q, QIAN B, HU R, et al. A matrix-cube-based estimation of distribution algorithm for the distributed assembly permutation flow-shop scheduling problem. *Swarm and Evolutionary Computation*, 2021, 60(2): 100785.
- [14] CEBERIO J, IRUROZKI E, MENDIBURU A, et al. A distance-based ranking model estimation of distribution algorithm for the flowshop scheduling problem. *IEEE Transactions on Evolutionary Computation*, 2013, 18(2): 286 – 300.
- [15] DENG C, HU R, QIAN B, et al. Hybrid estimation of distribution algorithm for solving three-stage multiobjective integrated scheduling problem. *Complexity*, 2021, 2021(2): 5558949.
- [16] SHEN Y D, PENG L W, LI J P. An improved estimation of distribution algorithm for multi-compartment electric vehicle routing problem. *Journal of Systems Engineering and Electronics*, 2021, 32(2): 365 – 379.
- [17] DAI M, ZHANG Z, GIRET A, et al. An enhanced estimation of distribution algorithm for energy-efficient job-shop scheduling problems with transportation constraints. *Sustainability*, 2019, 11(11): 3085.
- [18] WANG Aihu, CHA Jianzhong, WANG Jinmin. A heuristic method of rectangular object layout based on binary tree structure expression. *Journal of Software*, 1996, 7(4): 252 – 257.
(王爱虎, 查建中, 王金敏. 一种基于二叉树结构表达的矩形物体布局的启发式方法. *软件学报*, 1996, 7(4): 252 – 257.)
- [19] MONTGOMERY D C. *Design and Analysis of Experiments*. Hoboken: John Wiley and Sons, 2005.

作者简介:

孙蓉洁 硕士研究生, 目前研究方向为复杂系统智能优化, E-mail: 1014104678@qq.com;

钱 斌 教授, 博士生导师, 目前研究方向为优化调度理论与方法、智能优化方法, E-mail: bin.qian@vip.163.com;

胡 蓉 副教授, 硕士生导师, 目前研究方向为智能优化调度、物流优化, E-mail: ronghu@vip.163.com;

张大骋 讲师, 硕士生导师, 目前研究方向为数据处理、系统随机过程建模和可靠性评估, E-mail: dacheng.zhang@kust.edu.cn;

向凤红 教授, 硕士生导师, 目前研究方向为智能优化方法, E-mail: xiangfh5447@sina.com.