

基于改进遗传算法的船体分段车间零部件配送优化研究

施 炜¹, 邓啸尘², 周 宏²

(1. 中远海运重工有限公司 经营总部, 上海 200135; 2. 江苏科技大学 船舶与海洋工程学院, 江苏 镇江 212003)

摘 要: 船体分段制造车间的物资配送问题是典型的车辆路径问题。现阶段利用传统遗传算法规划配送路线已经不能满足日益繁重的物流任务。为满足车间对物资配送准时化的要求, 需要对传统遗传算法进行改进, 添加时间窗对物资配送时间进行约束。利用改进遗传算法, 可以更好地提高物资配送效率。

关键词: 分段车间; 车辆路径; 遗传算法; 时间窗

中图分类号: U671.4

文献标志码: A

文章编号: 1671-9891(2020)01-0039-08

0 引言

近年来, 随着“精益生产”这一造船理念在船舶建造领域的兴起, 对船舶生产物流进行优化逐渐成为实现精益造船的关键。船舶分段制造占据了船舶建造周期中较长时间且包含大量物流作业, 分段生产的物流作业将对船舶建造成本产生极大的影响。对于大型船舶建造企业, 在多船型项目并行时, 船舶分段生产属于“小批量、多品种”生产, 需要按照托盘对零部件进行集配与运输, 而在我国船舶制造过程中零部件的配送尚未能得到足够的重视。^[1]若对船舶分段车间的零部件配送方案进行优化, 将会有效改善船舶分段车间的物流状况, 从而提高分段车间的物流效率, 缩短分段制造周期, 降低船舶制造成本。

1 船体分段车间物资配送优化模型

1.1 船体分段车间物资配送问题描述

通过对分段制造车间的物流情况进行实地调研分析, 发现各工位对物资的储存能力极为有限, 这就需要在适当的时间段内进行物资配送, 以防超出工位的储存能力。针对船体分段制造车间的具体情况, 本研究采用硬时间窗的约束形式, 即当制造分段所需部件的配送时间不在规定的时间段内, 工位将无法接收此次配送的部件, 将船体分段制造车间的零部件配送问题归结为带有硬时间窗约束的车辆路径问题 (VRPTW)。^[2]

VRPTW 问题的定义如下: 设分段制造车间有 n 个工位需要进行部件配送, 由 1 个配送中心负责通过 m 辆载重一定的运输车辆对车间进行物资配送。

已知: 船体分段制造车间内各工位的位置、对部件的需求量以及可接收部件的时间区间。设 UT_i 为配送过程中在工位 i 处对零部件进行装卸作业所消耗的时间, $[ET_i, LT_i]$ 为物流计划所规定的零部件必须送达的时间区间, 其中 ET_i 与 LT_i 指零部件送达工位 i 的最早与最晚两个时间节点。运输车辆将零部件实际送达指定工位 i 的时间用 RT_i 表示, 车辆从工位 i 移动到工位 j 所消耗的时间用 t_{ij} 表示。各项参数应该满足以下关系:

$$RT_0 = 0 \quad ET_i \leq RT_i \leq LT_i \quad (1)$$

约束条件:

- 已知车辆的运载能力、配送起始位置、各工位可接收部件的时间窗;
- 已知车间内各工位对应的物资需求量, 分段车间所需零部件都由配送中心进行供给;

收稿日期: 2020-02-14

作者简介: 施炜 (1973—), 男, 上海人, 中远海运重工有限公司经营总部工程师, 硕士。

- c. 运输道路畅通无阻塞;
- d. 每个工位需要被运输一次;
- e. 每条运输路线上的车辆,在完成零部件配送任务后都必须返回起点;
- f. 运输车辆的型号相同,运输过程中不可超重。

1.2 船体分段车间物资配送数学模型

分段制造车间工位位置已知,车间所含工位与各工位之间的距离可用无向图 $G(V, E)$ 表示,其中 V 是所有工位的集合, $V_0 \in V$ 为配送中心, E 为各工位之间距离的集合 $e_{ij} \in E$, 表示工位 i, j 之间的距离。配送中心需要对分段车间内 n 个工位进行零部件的派送, g_i 为分段车间内工位 i 对零部件的需求量, UT_i 为运输车辆在工位 i 处消耗的时间;运输车辆可抵达工位 i 的时间区间为 $[ET_i, LT_i]$, 车辆从工位 i 行至工位 j 所消耗的时间表示为 $t_{ij} = e_{ij}/v$;参与本次零部件配送任务的运输车辆共有 K 辆,车辆最大载重为 $q (q > g_i, i = 1, 2, \dots, n)$, 车辆行驶速度为 v ;在满足物资准时送达的情况下,运输车辆不得超重运行。现需要选择最短运输路径进行零部件配送。

通过上述数学模型可知,运输车辆将零部件送达工位 j 的时间 RT_j 可表示为:

$$RT_j = RT_i + UT_i + t_{ij} \quad (2)$$

建立车辆调度函数,定义 x_{ijk}, y_{ki} :

$$x_{ijk} = \begin{cases} 1 & \text{车辆 } k \text{ 从工位 } i \text{ 行驶到工位 } j \\ 0 & \text{其他} \end{cases} \quad (3)$$

$$x_{ki} = \begin{cases} 1 & \text{工位 } i \text{ 由车辆 } k \text{ 配送} \\ 0 & \text{其他} \end{cases} \quad (4)$$

目标函数:

$$Z(i, j, k) = \sum_{k=1}^K \sum_{j=0}^n \sum_{i=0}^n x_{ijk} e_{ij} \quad (5)$$

约束条件为:

$$\sum_{i=0}^n g_i y_{ki} \leq q \quad k \in \{1, 2, \dots, K\} \quad (6)$$

$$ET_i \leq RT_i \leq LT_i \quad i = 1, 2, \dots, n \quad (7)$$

$$\sum_{k=1}^K y_{ki} = 1 \quad i = 1, 2, \dots, n \quad (8)$$

$$\sum_{i=0}^n x_{ijk} = y_{kj} \quad j = 1, 2, \dots, n \quad k \in \{1, 2, \dots, K\} \quad (9)$$

$$\sum_{j=0}^n x_{ijk} = y_{ki} \quad i = 1, 2, \dots, n \quad k \in \{1, 2, \dots, K\} \quad (10)$$

$$X = (x_{ik}) \in S \quad (11)$$

$$\sum_{k=1}^K \sum_{j=0}^n x_{ijk} \leq K \quad i = 0 \quad (12)$$

目标函数(5)表示运输车辆进行零部件派送任务所要行驶的总路程;

约束条件(6)表示每次配送任务中车辆所载零部件的重量不可以超过车辆最大载重;

约束条件(7)表示运输车辆将送达零部件的时间区间不得超过 $[ET_i, LT_i]$;

约束条件(8)表示每一个工位由一辆车进行配送;

约束条件(9)表示运输车辆 k 只能对指定路线上的工位进行配送;

约束条件(10)表示运输车辆 k 完成任务后只能从当前工位离开;

约束条件(11)表示运输车辆 k 所行使的路线是完整的;

约束条件(12)表示从配送中心出发进行零部件配送任务的车辆最多只能有 K 辆。

2 船体分段车间物资配送优化算法设计

2.1 染色体编码与解码

在利用遗传算法解决实际问题时, 首先需要解决编码问题。为了减少无效解的产生, 研究使用整数编码的方式。^[3]

分段车间有 i 个工位需要进行零部件的配送, 配送中心安排 k 台运输车辆进行配送作业, 由此可将染色体定义为 $(0, i_{11}, i_{12}, \dots, i_{1a}, 0, i_{21}, i_{22}, \dots, i_{2b}, 0, \dots, 0, i_{k1}, \dots, i_{kc}, 0)$, 染色体长度为 $i + k + 1$ 。其中 0 的实际意义为配送中心, 即配送起点与终点。不难发现, “0, $i_{11}, i_{12}, \dots, i_{1a}, 0$ ” “0, $i_{21}, i_{22}, \dots, i_{2b}, 0$ ” “0, $i_{k1}, \dots, i_{kc}, 0$ ” 即表示第 1 辆车、第 2 辆车以至于第 k 辆车进行零部件配送的行驶路线。以编码 (052301694081070) 为例, 此编码表示当前有 3 台运输车辆对 10 个工位进行物资配送。这 3 台运输车辆的行驶路线如下所示:

路线 1: 0 → 工位 5 → 工位 2 → 工位 3 → 0;

路线 2: 0 → 工位 1 → 工位 6 → 工位 9 → 工位 4 → 0;

路线 3: 0 → 工位 8 → 工位 10 → 工位 7 → 0;

为避免算法在后续的迭代进化过程中产生过多无用解, 在本项研究中去除 0, 得到新的子路线的编码为 (5231694817), 简化后的编码结构需要进行解码操作。本研究采取的解码方式与构造路线的方式相似。先选择一条初始线路, 该线路上的工位用染色体中的基因编码表示, 若此线路出现车辆超重的情况, 或是运输车辆超出允许配送时间, 则需重新规划运输线路。以此规则不断进化, 直到所有工位得到合理优化布置。

2.2 适应度函数

在 VRPTW 问题中, 目标函数的数值越小表示在满足各项约束条件的情况下, 车辆在此路线上的总行驶路程最短。^[4]相反, 在遗传算法中, 染色体个体的适应度越大, 则表示该个体的性能越好。分段车间部件配送优化的目的, 是通过硬时间窗的约束, 在实现准时化的同时, 使配送距离最短。因此在满足约束条件的基础上, 将目标函数转化为适应度函数, 表达式如下:

$$f_i = 1/(Z_i + P_i) \quad (13)$$

其中: Z_i 为染色体个体 i 所对应的目标函数值, 表示运输车辆在该条配送路线下行驶的路程; P_i 为惩罚函数, 惩罚成本表达式如下:

$$P_i(RT_i) = \begin{cases} 0 & ET_i \leq RT_i \leq LT_i \\ M & \text{其他} \end{cases} \quad (14)$$

在带有硬时间窗约束的分段车间零部件配送问题中, 工位所需零部件必须在规定时间段内送达指定工位。此处 M 的初始数值定为 5 000。

2.3 选择算子设计

设置种群规模为 M , 父代种群规模为 Z , 子代种群初始状态设定为 $X = \{\}$, f_i 表示单个个体的适应值, 具体操作步骤如下所示:

(1) 根据适应度值的大小, 所有个体按照从大到小的顺序进行排列, 排列后代种群为 $Z' = \{a_1, a_2, \dots, a_i, \dots, a_M\}$, 其中 $f(a_{i-1}) > f(a_i) > f(a_{i+1})$, 另外求得适应度最大的个体 k , 即 $f(a_{\max}) = \max f(a_i)$;

(2) 对种群中所有染色体的适应值进行求和 $\sum_{i=1}^M f(a_i)$;

(3) 求得各染色体 i 被选中的概率;

$$P_{ai} = f(a_i) / \sum_{i=1}^M f(a_i) \quad (15)$$

(4) 求得各染色体的累计概率;

$$Q_i = \sum_{i=1}^M P_{ai} \quad (16)$$

(5) 转动 M 轮盘:

① 产生 M 个位于区间 $[0, 1]$ 的随机数 r ; 如果 $r < Q_i$, 选择染色体 a_i ; 否则选择染色体 a_i ($2 \leq i \leq M$), 使得 $Q_{i-1} \leq r \leq Q_i$;

② 设 ξ_i 为区域 i 内所有的随机数, 对各区域内 ξ 值进行统计 $\xi_1, \xi_2, \dots, \xi_M$;

③取得最大的 ξ 值 $\xi_j = \max\{\xi_1, \xi_2, \dots, \xi_M\}$, 区间内 j 对应的染色体 a_j , 将其作为本轮转动选择的个体;

$$\textcircled{4} \begin{cases} M_i = a_j & \text{其他} \\ M_i = a_{\min \min(i_1, i_2, \dots, i_j)} & \text{若存在多个相同的 } \xi \text{ 值区间 } \xi_{i1}, \xi_{i2}, \dots, \xi_{il} (l \leq M) \end{cases}; \quad (17)$$

$$\textcircled{5} \text{将 } M_i \text{ 并入 } X, \text{ 即 } \begin{cases} X(0) = \phi \\ X(t) = X(t-1)M \end{cases}; \quad (18)$$

⑥若通过选择得到的个体数达到对种群数量的要求, 转步骤⑦, 否则转①;

⑦找到子代种群 X 中适应度最低的个体 $m: f(b_m) = \min(f(b_1), f(b_2), \dots, f(b_m))$;

(6) 个体 k 代替个体 m ;

(7) 将新选出的个体全部进行储存, 并返回。

2.4 交叉与遗传概率

设交叉概率为 P_c , 变异概率为 P_m , 本研究采用自适应方式建立 P_c 与 P_m 表达式:

$$P_c = \begin{cases} P_{c \max} - \frac{(P_{c \max} - P_{c \min})(f' - f_{\text{avg}})}{f_{\max} - f_{\text{avg}}} & f' \geq f_{\text{avg}} \\ P_{c \max} & f' \leq f_{\text{avg}} \end{cases} \quad (19)$$

$$P_m = \begin{cases} P_{m \max} - \frac{(P_{m \max} - P_{m \min})(f'' - f_{\text{avg}})}{f_{\max} - f_{\text{avg}}} & f'' \geq f_{\text{avg}} \\ P_{m \max} & f'' \leq f_{\text{avg}} \end{cases} \quad (20)$$

式中: f_{\max} 为群体中出现的最大适应值; f_{avg} 为整个群体的平均适应值; f' 为进行基因序列交叉操作两个染色体中较大适应值; f'' 为染色体进行变异操作后得到的适应度值; $P_{c \max}$ 、 $P_{c \min}$ 为交叉概率设定的最大、最小值; $P_{m \max}$ 、 $P_{m \min}$ 为变异概率设定的最大、最小值。

2.5 交叉算子设计

分段车间零部件配送问题使用整数排列的方式进行染色体编码, 故本研究采用部分匹配操作(PMX)进行算子的交叉设计。^[5]类 PMX 操作过程如下:

设一对父代染色体 P_1 、 P_2 , 在 P_1 与 P_2 中用“|”各随机选择一段基因序列进行交叉:

P_1 : 814|567|32;

P_2 : 685|231|47

将 P_2 中“|231|”置于 P_1 前, 将 P_1 中“|567|”置于 P_2 前, 由此得到新个体 P_1' 、 P_2' 。

P_1' : 231|81456732|

P_2' : 567|68523147|

将 P_1' 与 P_2' 中与染色体序列前端交配区域相同的基因删除, 得到新个体 P_1'' 与 P_2'' 。

P_1'' : 23184567

P_2'' : 56782314

2.6 变异算子设计

变异算子的作用体现在, 扩大遗传算法的搜索范围, 同时防止出现局部最优的情况。采用到位变异算子进行操作: 选取一段染色体序列中的两个变异点, 用“|”将变异区域与其他不参与变异的基因隔开, 具体变异操作过程示例如下:

(1) 随机生成染色体 $P_3 = 24137856$, 此染色体包含两个变异点, 如 4 和 8, 即 $P_3 = 2 | 41378 | 56$, 用“|”表示交变异的区域;

(2) 将 | 41378 | 进行反序操作变成 | 87314 |, 再将反序后的基因序列插入原位置, 得到新染色体 $P_3' = 2 | 87314 | 56$ 。

2.7 算法终止规则

为终止遗传算法的进化过程, 需要提前设置算法的终止规则, 即达到某种效果时, 算法停止演化。通常有以下几种方法对算法进行终止:

(1) 预先设定进化代数。当算法的循环演化过程达到预先设定的进化代数时, 进化过程停止, 得到最终结果;

(2) 后续的进化已无法对解进行改进。当进化已经达到预先设定的进化代数, 并且后续进化操作得到的解始终与当前解保持一致, 则此解为最优解, 进化过程停止, 输出最终结果;

(3) 将多种终止规则进行组合。在一个算法中加入多种终止规则, 当进化过程满足某个终止规则时, 停止进化, 输出当前结果。

本研究采用第二种终止规则。即预先设定进化代数 N , 染色体一直进化到 N 代, 期间不停止, 当进化过程达到第 N 代时, 再停止进化, 比较后选择较好的解作为最优解输出。

2.8 算法步骤

算法的具体步骤为:

Step1: 对分段车间各工位进行布置, 并采用整数编码的方式生成合理的染色体序列;

Step2: 设置控制参数;

Step3: 标记进化代数 $g = 0$, 随机生成初始种群 $P(0)$, 初始种群包含 M 个染色体;

Step4: 若满足终止条件, 则停止进化, 并输出当前解; 否则算法继续运行;

Step5: 设置循环变量;

Step6: 计算适应度函数值;

Step7: 如果 $i > N$, 执行 Step8, 否则的话回到 Step6;

Step8: 根据 2.9 节选择算子设计方法, 生成下一代;

Step9: 进行类 PMX 操作并设计变异算子, 对 $P(g)$ 进行重组;

Step10: 标记进化代数 $g = g + 1$;

Step11: 若满足终止条件, 则停止进化, 否则转 Step4。

通过步骤 1 至步骤 11 的迭代进化, 得到最优个体, 并输出最优解。再将染色体进行解码操作, 得到船体分段制造车间物资配送的最优路线。改进遗传算法流程图如图 1 所示。

3 船体分段制造车间物资配送优化实例

3.1 数据及参数确定

某造船厂分段制造车间现有 12 个装配工位 $\{V_1, V_2, \dots, V_{12}\}$ 需要对其进行零部件的配送, 根据分段车间的现场布置图, 可得到个工位间的距离, 如表 1 所示。所有工位的零部件均由配送中心 V_0 进行配送, 配送中心现有 $K = 3$ 辆运输车辆可用于配送作业, 运输车辆型号相同, 载重均为 $q = 6$ t, 根据船厂规定, 车辆的行驶速度均为 $v = 60$ m/min, 运输车辆在工位进行零部件装卸作业时间 $UT_i = 5$ min, 所有运输车辆的最大配送周期为 90 min。各工位对零部件的需求量 g_i 以及硬时间窗如表 2 所示。

完成对必要数据的统计后, 对算法运行所需的参数进行确定。在固定运输车辆的运载能力的前提下, 群体的规模 $N = 50$, 进化代数 $g = 100$, 交叉概率的最大值为 0.8, 交叉概率的最小值为 0.2, 变异概率的最大值为 0.2, 变异概率的最小值为 0.02, 初始惩罚值为 5 000。

3.2 结果分析

必要数据及参数设定完成后, 算法进行迭代进化, 输出的最优解为: $\{6, 12, 9, 5, 11, 3, 7, 4, 10, 2, 8, 1\}$ 。通过解码操作得到配送路径:

路线 1: 6-12-9-5: 配送中心-工位 6-工位 12-工位 9-工位 5-配送中心, 路线 1 总长度为 3 705 m。

路线 2: 11-3-7-4: 配送中心-工位 11-工位 3-工位 7-工位 4-配送中心, 路线 2 总长度为 2 812 m。

路线 3: 10-2-8-1: 配送中心-工位 10-工位 2-工位 8-工位 1-配送中心, 路线 3 总长度为 2 523 m。

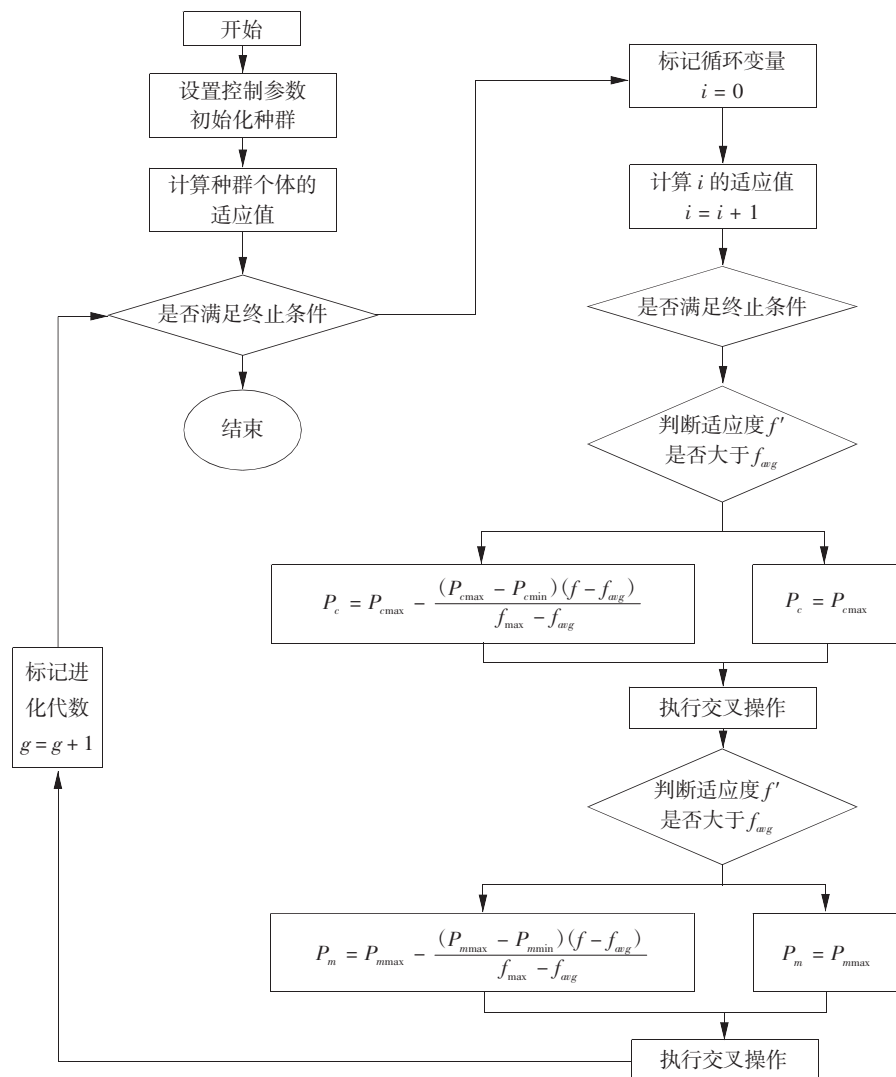


图 1 改进遗传算法流程

表 1 分段制造车间各工位间距离

单位:m

序号	0	1	2	3	4	5	6	7	8	9	10	11	12
0	0	327	446	241	691	551	724	324	241	691	655	630	655
1	327	0	695	536	385	335	569	360	569	1 012	979	918	878
2	446	695	0	216	1 080	724	763	763	342	432	630	241	216
3	241	536	216	0	918	648	763	551	151	482	551	389	432
4	691	385	1 080	918	0	540	763	551	929	1 382	1 303	1 300	1 260
5	551	335	724	648	540	0	241	691	756	1 127	1 188	965	842
6	724	569	763	763	763	241	0	918	889	1 192	1 314	997	821
7	324	360	763	551	551	691	918	0	482	929	778	922	979
8	241	569	342	151	929	756	889	482	0	457	432	446	551
9	691	1 012	432	482	1 382	1 127	1 192	929	457	0	342	241	482
10	655	979	630	551	1 303	1 188	1 314	778	432	342	0	540	763
11	630	918	241	389	1 300	965	997	922	446	241	540	0	241
12	655	878	216	432	1 260	842	821	979	551	482	763	241	0

表 2 工位部件需求量、工位时间窗与装卸时间

序号	物料需求量/100 kg	最早开始时间/min	最晚离开时间/min	装卸时间/min
0	0.0	0.0	90.0	0
1	9.0	24.5	74.5	5
2	23.0	4.5	22.0	5
3	9.0	27.0	64.5	5
4	14.4	22.5	38.0	5
5	12.6	47.5	77.5	5
6	10.8	7.0	27.0	5
7	17.1	5.0	25.0	5
8	12.6	34.5	44.5	5
9	18.0	12.5	27.5	5
10	7.0	17.0	40.0	5
11	17.0	7.5	22.5	5
12	2.0	55.0	82.5	5

对进化结果进行统计:本次对分段车间零部件进行配送,共需 3 辆运输车辆,分 3 条路线进行配送,车辆行驶的总路程为 9 040 m,零部件的配送任务都能按时完成,未出现超载情况。

为了体现改进遗传算法的优越性,针对传统遗传算法,设定初始参数:群体的规模 $N = 50$,进化代数 $g = 100$,交叉概率 0.6,变异概率 0.1。传统遗传算法输出的最优解为{4,8,12,7,3,6,11,9,10,2,5,1},车辆行驶的总路程为 9 178 m。

将改进遗传算法与传统遗传算法进行比较,如图 2 所示。

由对比图可以发现,改进遗传算法在进化到第 62 代时,目标函数值趋于稳定,得到最优解;而传统遗传算法进化到 70 代时才趋于稳定。改进遗传算法的收敛速度更快,要优于传统遗传算法。比较两种算法得到的车辆行驶总路程可以发现,改进遗传算法得到的配送路程更短。

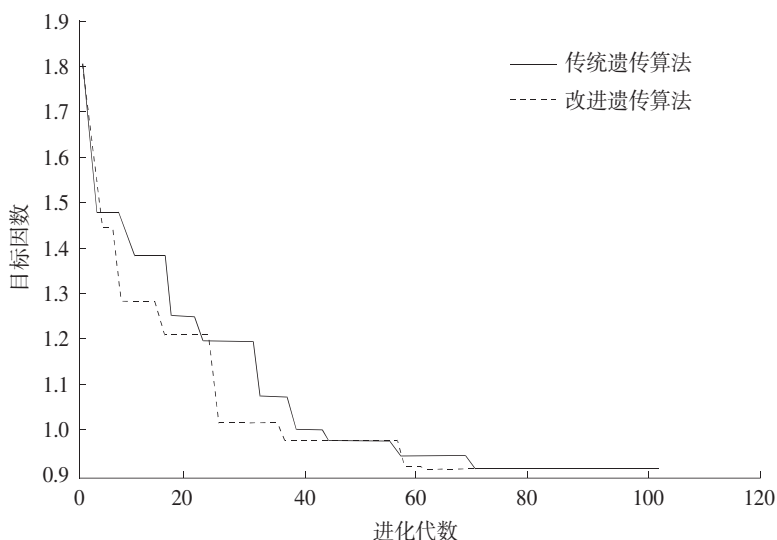


图 2 两种算法进化过程对比

4 结束语

对船舶生产物流进行优化是实现精益造船的关键。在船体分段制造车间零部件配送任务中,对配送车辆进行合理的路径规划,是实现缩短分段制造周期、降低物流成本的重要手段。遗传算法是解决车辆路径问

题的有效方法,通过引入硬时间窗,约束配送时间,对传统遗传算法进行改进。相较于传统遗传算法而言,改进遗传算法有效缩短了车辆的行驶路程,降低了物流成本,节省配送时间,提高分段建造效率,从而缩短造船周期。

参考文献:

- [1]朱志勇,刁洪祥.基于改进遗传算法的车辆路径问题研究[J].湘潭大学自然科学学报,2011(3):115-118.
- [2]王楠.基于实时状态信息的混流装配生产优化与仿真技术研究[D].武汉:华中科技大学,2012.
- [3]魏平,熊伟清.一种改进的实数编码遗传算法[J].计算机应用研究,2004(9):87-88,91.
- [4]黄明,林广智,梁旭等.改进的遗传算法在车辆路径问题中的应用[J].大连交通大学学报,2010(1):95-99.
- [5]雷明达.船舶分段车间物流优化与仿真研究[D].哈尔滨:哈尔滨工程大学,2017.

(责任编辑:张 利)

Research on Optimization of Parts Distribution of Ship Hull Segmented Workshop Based on Improved Genetic Algorithm

SHI Wei¹, DENG Xiao-chen², ZHOU Hong²

(1. Operating Headquarter, COSCO Shipping Heavy Industry Co., Ltd., Shanghai 200135, China;

2. School of Ship and Ocean Engineering, Jiangsu University of Science and Technology, Zhenjiang 212003, China)

Abstract: The problem of material distribution in the hull segmented manufacturing workshop is that of a typical vehicle routing. At current stage, the use of traditional genetic algorithms to plan distribution routes can no longer meet the increasingly heavy logistics tasks. In order to meet the workshop's requirements for punctual material distribution, the traditional genetic algorithm needs to be improved, and a time window is added to restrain material distribution time. The use of improved genetic algorithm can better improve the efficiency of material distribution.

Key words: segmented workshop; vehicle routing; genetic algorithm; time window