

Pareto最大最小蚂蚁算法及其在热轧批量计划优化中的应用

贾树晋¹, 朱俊^{2,3}, 杜斌^{1,3}, 岳恒⁴

(1. 上海交通大学自动化系, 系统控制与信息处理教育部重点实验室, 上海 200240;

2. 东北大学信息科学与工程学院, 辽宁沈阳 110004; 3. 宝钢研究院自动化所, 上海 201900;

4. 东北大学自动化研究中心, 辽宁沈阳 110004)

摘要: 针对双目标旅行商问题提出了基于Pareto概念的最大最小蚂蚁算法(P-MMAS). 通过重新设计状态转移策略、信息素更新策略及局部搜索策略, 同时引入基于自适应网格的多样性保持策略与信息素平滑机制, 使算法能够快速搜索到在目标空间上均匀分布的近似Pareto前端. 通过在6个标准测试函数上的实验及在热轧批量计划优化中的应用, 表明P-MMAS具有良好的优化性能及实用性.

关键词: 蚁群算法; 双目标旅行商问题; 多目标优化; 组合优化; 热轧批量计划

中图分类号: TP18 **文献标识码:** A

Pareto max-min ant system algorithm and its application to hot rolling batch planning problem

JIA Shu-jin¹, ZHU Jun^{2,3}, DU Bin^{1,3}, YUE Heng⁴

(1. Department of Automation, Shanghai Jiao Tong University, and Key Laboratory of System Control and Information Processing, Ministry of Education, Shanghai 200240, China;

2. School of Information Science & Engineering, Northeastern University, Shenyang Liaoning 110004, China;

3. Research Institute of Automation, Academy of BaoSteel, Shanghai 201900, China;

4. Research Center of Automation, Northeastern University, Shenyang Liaoning 110004, China)

Abstract: A Pareto concept-based max-min ant system algorithm for the bi-objective traveling salesman problem is proposed. By modifying the state transition rule, pheromone updating rule and the local search rule; as well as employing an adaptive-grid based diversity maintenance approach and the pheromone trail smoothing mechanism, we find the approximate Pareto front which is uniformly distributed on the objective space. Simulation on 6 benchmark functions and application to a hot rolling batch planning problem indicate that the proposed algorithm has desirable performance and practicability.

Key words: ant colony algorithm; bi-objective traveling salesman problem; multi-objective optimization; combinatorial optimization; hot rolling batch planning

1 引言(Introduction)

多目标优化中不存在唯一的最优解, 而是存在多个相互之间无法比较优劣的Pareto最优解. 这些解构成的集合称为Pareto最优解集, 其对应的目标向量组成的曲线(或曲面)称为Pareto前端. 以往在求解多目标优化问题时往往采用加权聚合的方法将其转化为单目标优化问题, 存在的问题是目标权重一般不易确定, 同时该方法无法获得非凸的Pareto前端^[1]. 为了克服上述传统优化技术的缺点, 近年来一些学者提出了多目标智能优化算法^[2-6], 为求解多目标优化问题提供了新的思路.

多目标旅行商问题(traveling salesman problem, TSP)是经典TSP的扩展和延伸, 属于典型的多目标组合优化问题. 工程决策中的大量问题可归结为多

目标TSP, 如生产计划、物流调度等. 蚁群算法^[7-10]作为一种求解经典TSP高效算法, 已有不少文献^[3-4]将其推广为多目标蚁群算法. 这些算法大都基于蚂蚁算法(ant system, AS)^[7]和蚁群算法(ant colony system, ACS)^[8]. 文献^[11-12]对多目标蚁群算法进行了分类与综述, 同时文献^[12]针对双目标TSP对已有多目标蚁群算法进行了比较研究, 认为MACS^[3]与P-ACO^[4]的性能较好, 但由于缺少局部搜索机制, 与多目标遗传局部搜索算法(multi-objective genetic local search, MOGLS)^[2]相比仍有较大的差距.

由于最大最小蚂蚁算法(max-min ant system, MMAS)^[9]与ACS并称为最成功的蚁群算法, 是解决经典TSP最好的算法之一^[10]. 受此启发, 本文在MMAS算法的基础上, 通过重新设计状态转移策

略、信息素更新策略及局部搜索策略,同时引入基于自适应网络的多样性保持策略与信息素平滑机制,提出了一种新的多目标蚁群算法——Pareto最大最小蚂蚁算法(Pareto max-min ant system, P-MMAS). 仿真实验及在热轧批量计划优化中的应用验证了该算法的有效性与实用性.

2 多目标TSP问题(Multi-objective TSP)

经典TSP是组合优化领域的一个著名难题,已被证明属于NP-hard问题,目前尚未找到多项式算法^[13]. 多目标TSP是经典TSP的扩展问题,可一般化描述为:已知 n 个城市,每两个城市间有 M 个指标值,目的是找到一条通过所有城市各一次且最后回到出发城市的路线,使得该路线上的 M 个总指标值都要尽可能的小. 可知多目标TSP也属于NP-hard问题.

当 $M = 2$ 时,多目标TSP转化为双目标TSP. 作为多目标TSP的典型代表,双目标TSP得到了广泛的研究^[2,12]. 本文针对双目标TSP,提出了一种新的多目标蚁群算法——P-MMAS算法.

3 Pareto最大最小蚂蚁算法(Pareto max-min ant system)

3.1 MMAS简介(Introduction to MMAS)

MMAS是AS的一个变种,与AS的主要区别为:

- 1) MMAS仅对一个最优蚂蚁的轨迹进行信息素更新,提高了优化效率.
- 2) 信息素浓度被限制在 $[\tau_{\min}, \tau_{\max}]$ 内,防止算法早熟收敛.
- 3) 信息素初始化为 τ_{\max} ,有利于算法初期大范围的“探索”.
- 4) 当算法早熟收敛时,信息素被重新初始化以跳出收敛态. 算法细节请参考文献[9–10].

3.2 P-MMAS算法步骤(P-MMAS procedure)

P-MMAS算法是一种全新的多目标蚁群算法,其算法步骤如下:

Step 1 信息素浓度初始化为最大信息素浓度,并将 m 只蚂蚁随机地放在 n 个城市.

Step 2 通过状态转移策略构造每只蚂蚁的可行路径.

Step 3 待所有蚂蚁完成路径构造后,使用局部搜索算法对可行解进行局部搜索.

Step 4 将非支配解更新到外部档案中,并删除外部档案中的被支配解,之后使用自适应网格方法保持外部档案中非支配解的多样性.

Step 5 对每次迭代中成功更新到外部档案中的非支配解进行信息素更新,并根据最大最小信息素浓度对其进行限幅.

Step 6 若算法出现早熟收敛,使用信息素平滑机制(pheromone trail smoothing, PTS)处理蚁群信息素矩阵,以便继续搜索新解.

Step 7 若算法满足终止条件,输出外部档案中的非支配解及对应的近似Pareto前端并停止算法;否则转Step 2.

P-MMAS算法伪码如下:

Initialization(); /*初始化*/

While(termination condition not met) do

ConstructSolutions(); /*路径构造*/

LocalSearch(); /*局部搜索*/

UpdateNondominatedSet(); /*更新外部档案*/

UpdateTrails(); /*更新信息素*/

PTS(); /*信息素平滑*/

End

3.3 状态转移策略(State transition rule)

P-MMAS属于多信息素矩阵单启发式信息的多目标蚁群算法类型^[12],每个目标对应一个信息素矩阵,并采用加权聚合方式确定启发式信息.

在每次迭代中, m 只蚂蚁根据状态转移策略构造各自的路径. 本文选用基于伪随机比例规则(pseudo-random-proportional rule)^[8]的状态转移策略来确定下一个城市:

$$s = \begin{cases} \arg \max_{j \in \Omega} ([\tau_{ij}^1]^{\lambda\alpha} \cdot [\tau_{ij}^2]^{(1-\lambda)\alpha} \cdot \eta_{ij}^\beta), & q \leq q_0, \\ S, & \text{其他,} \end{cases} \quad (1)$$

其中: α, β 是表征信息素与启发式信息相对重要性的参数; q 为在 $[0,1]$ 区间上服从均匀分布的随机数; q_0 是一个确定的参数($0 < q_0 < 1$),一般取 $q_0 = 0.98$; Ω 表示城市 i 的可行邻接城市集; τ_{ij}^1, τ_{ij}^2 分别表示两个信息素矩阵中城市 i 与 j 间的信息素浓度; η_{ij} 表示城市 i 与 j 间加权聚合后的启发式信息,即

$$\eta_{ij} = \lambda\eta_{ij}^1 + (1 - \lambda)\eta_{ij}^2, \quad (2)$$

$\eta_{ij}^k = 1/d_{ij}^k$ 表示第 k 个目标的启发式信息, $k = 1, 2$; λ 表示第1个目标的权重.

为了使每只蚂蚁搜索不同的目标空间,第 l 只蚂蚁对应的 λ 可表示为

$$\lambda^l = \frac{l-1}{m-1}, \quad l \in \{1, \dots, m\}, \quad (3)$$

考虑极限情况,当 $\lambda = 1$ 时,表示仅考虑第1个目标,当 $\lambda = 0$ 时,则仅考虑第2个目标; S 表示按照下述概率分布确定的城市:

$$P(j) = \begin{cases} \frac{[\tau_{ij}^1]^{\lambda\alpha} \cdot [\tau_{ij}^2]^{(1-\lambda)\alpha} \cdot \eta_{ij}^\beta}{\sum_{u \in \Omega} [\tau_{iu}^1]^{\lambda\alpha} \cdot [\tau_{iu}^2]^{(1-\lambda)\alpha} \cdot \eta_{iu}^\beta}, & j \in \Omega, \\ 0, & \text{其他.} \end{cases} \quad (4)$$

根据状态转移策略构造路径的算法伪码如下:

```

procedure ConstructSolutions(···)
{
  For  $i=1$  to  $m$  do /* $m$ 只蚂蚁*/
     $C = \{1, \dots, n\}$ ; /*路径可选城市集合*/
    For  $j=1$  to  $n$  do /* $n$ 个城市*/
      /*状态转移策略*/
      choose city  $s \in C$  with equations (1)–(4);
      /*将该城市加入禁忌表*/
       $C \leftarrow C - \{s\}$ ;
    End For
  End For
}

```

3.4 局部搜索策略(Local search rule)

对于经典TSP来说,局部搜索是为了在蚁群算法提供的基解邻域内搜索使目标下降的解. TSP典型的局部搜索算法有2-opt, 3-opt, L-K算法^[9]等. 2-opt的算法复杂度最小,但精度也最差; L-K算法的精度最好,但算法复杂度最高; 3-opt则居于两者之间.

不同于单目标TSP,多目标TSP局部搜索的目的是为了寻找更好的非支配解或与基解互不支配的解. 本文针对多目标局部搜索的特性,提出了一种新颖的局部搜索策略:以P-MMAS每次迭代中产生的解作为基解,然后对每个基解进行局部搜索后产生4个新解,前两个分别为针对第1,2个目标进行局部搜索后得到的解,第3个为使两个目标值同时下降得到的解,最后1个是由下式定义的效用函数作为目标进行局部搜索后得到的解:

$$u = \lambda^l f_1 + (1 - \lambda^l) f_2. \quad (5)$$

综合考虑搜索效率与精度,局部搜索算法可选择2-opt或reduced 3-opt^[9],算法伪码如下:

```

procedure LocalSearch(···)
{
  solution_1=LS(baseSolution,obj_1);
  solution_2=LS(baseSolution,obj_2);
  solution_3=LS(baseSolution,obj_1,obj_2);
  solution_4=LS(baseSolution,obj_1,obj_2, $\lambda$ );
}

```

3.5 多样性保持策略(Diversity maintenance)

在多目标优化中,一个特殊的问题是要考虑所得非支配解的多样性,即在目标空间上形成的近似Pareto前端是否均匀分布,这里强调的“均匀分布”,不仅指前端上各点的间距均匀,还指前端覆盖的目标空间范围要足够广泛. 本文选用基于自适应网格的多样性保持策略^[5],主要原因是它的算法复杂度要低于传统的小生境方法. 基于多样性保持策

略的非支配解更新伪码如下:

```

procedure UpdateNondominatedSet(···)
{
  /*将当前蚁群中的非支配解更新到外部档案*/
  UpdateToArchive();
  /*多样性保持策略-自适应网格*/
  /* $N$ 为外部档案规定大小*/
  While numOfArchive >  $N$  do
    GridLocations(); /*确定非支配解网格位置*/
    MostCrowdedGrid(); /*确定最拥挤网格*/
    DeleteSolution(); /*删除最拥挤网格中的一个解*/
  End While
}

```

3.6 信息素更新策略(Pheromone updating rule)

在单目标TSP中,MMAS仅对每次迭代中的一个最优解进行信息素更新. 双目标TSP不存在唯一的最优解,在每次迭代中,P-MMAS可以产生多个进入外部档案的非支配解,相当于单目标TSP中每次迭代的最优解. 为了在这些较优解的邻域内进行搜索以得到更优解,基于蚁群算法正反馈原理,P-MMAS对这些非支配解进行信息素更新. 更新公式如下:

$$\tau_{ij}^k(t+1) = [(1 - \rho)\tau_{ij}^k(t) + \Delta\tau_{ij}^k(t+1)]_{\tau_{\min}^k}^{\tau_{\max}^k}, \quad (6)$$

其中: $k=1,2$ 对应两个信息素矩阵; ρ 表示信息素挥发系数; $\tau_{\max}^k, \tau_{\min}^k$ 分别为信息素浓度的上下限,算子 $[x]_a^b$ 定义为

$$[x]_a^b = \begin{cases} a, & x < a, \\ b, & x > b, \\ x, & \text{其他.} \end{cases}$$

信息素变化量表示为

$$\Delta\tau_{ij}^k(t+1) = \begin{cases} \frac{1}{f_k(p^{\text{update}})}, & (i,j) \in p^{\text{update}}, \\ 0, & \text{其他,} \end{cases} \quad (7)$$

其中: p^{update} 表示每次迭代中成功更新到外部档案的解; $f_k(p^{\text{update}})$ 表示该解对应的第 k 个目标值.

最大最小信息素浓度表达式如下:

$$\tau_{\max}^k = \frac{1}{\rho} \cdot \frac{1}{f_k(p_{\text{global-opt}}^k)}, \quad (8)$$

$$\tau_{\min}^k = \frac{\tau_{\max}^k}{\text{maxFactor}},$$

其中: $p_{\text{global-opt}}^k$ 表示第 k 个目标的迄今最优解, $f_k(p_{\text{global-opt}}^k)$ 则相应地表示第 k 个目标的迄今最优值. 文献^[10]指出,当状态转移策略选用伪随机比例规则时,应选择较窄的信息素浓度区间. 比较,本文取 $\text{maxFactor} = 5$.

3.7 信息素平滑机制(Pheromone trail smoothing)

随着P-MMAS的运行,外部档案中存储的非支配解逐渐增加,新产生的解越来越难得到更新,随着信息素的挥发,信息素浓度逐渐下降到 τ_{\min} .此时状态转移策略中起主要作用的变成启发式信息,接近于贪婪搜索,导致在随后的搜索中,这些贪婪解被反复选择,限制了算法的搜索空间,使算法早熟收敛.

为了抑制早熟收敛,P-MMAS中加入了PTS机制.当算法早熟收敛时(通过average branching factor^[10]来判断),对两个信息素矩阵按下式进行处理,有助于算法跳出收敛态,扩大搜索空间:

$$\tau_{ij}^{k*}(t) = \tau_{ij}^k(t) + \delta(\tau_{\max}^k(t) - \tau_{ij}^k(t)), 0 \leq \delta \leq 1, (9)$$

$\tau_{ij}^{k*}(t)$ 表示 t 时刻第 k 个信息素矩阵经PTS作用后城市 i 与 j 间的信息素浓度.当 $\delta = 1$ 时表示重新初始化信息素浓度;当 $\delta = 0$ 时表示禁用PTS机制.本文取 $\delta = 1$.

4 实验研究(Experiment research)

4.1 测试环境(Test environment)

选用双目标TSP标准测试库中的6个算例(Kroab50, Krocd50, Kroab100, Kroad100, Krobc100, Krocd100)^[2]来比较P-MMAS与其他3种多目标优化算法(MOGLS^[2], MACS^[3], P-ACO^[4])的性能. MOGLS参数设置同文献[2]. P-MMAS, MACS与P-ACO的参数经敏感性分析后设置如下:

$$\begin{aligned} \text{AntNum} &= 30, \alpha = 1, \beta = 2, \\ \rho &= 0.1, \text{maxFactor} = 5. \end{aligned}$$

算法使用C++语言编写,统一运行50s作为算法终止条件.

4.2 性能度量(Performance metric)

本文采用定性与定量相结合的方法来度量算法性能.定性方法是指通过图形化的近似Pareto前端来直观地比较.每种算法经10次独立运行产生的所有非支配解合并成一个非支配解集,然后去除该解集中所有被支配解,从而形成该算法的近似Pareto前端.定量方法是指使用 M_1^* , M_2^* , M_3^* , C ^[6,12]4种定量指标来度量算法性能.其中: M_1^* 用来度量与真实Pareto前端的平均距离, M_2^* 用来度量解的分布性, M_3^* 用来度量解覆盖的目标空间范围,上述3个指标均为一元指标, C 指标为二元指标,用来度量两个算法产生的非支配解集的支配关系.

由于算例的真实Pareto前端未知,在计算 M_1^* 指标时,使用伪Pareto前端^[12]来替代,产生方法如下:针对某一算例,待比较的算法经10次独立运行产生的所有非支配解合并成一个非支配解集,然后去除该解集中的所有被支配解,从而形成该算例的伪Pareto前端. M_2^* 指标中的 σ 取10000和20000,分别对

应50和100个城市^[12].

4.3 性能比较(Performance comparison)

图1为各种算法对应6个算例的近似Pareto前端,从中可以得到以下结论:① P-ACO产生的非支配解仅能覆盖目标空间的中间部分,无法扩展到目标空间的两端;MACS的非支配解能够较好地覆盖整个目标空间,但其近似Pareto前端中间部分的解性能较差,被P-MMAS, MOGLS, P-ACO的解支配.② P-MMAS与MOGLS的性能最为接近,但仔细观察可以发现,P-MMAS的解几乎处处支配MOGLS的解,且P-MMAS的近似Pareto前端覆盖的目标空间范围更广,分布更为均匀.③ 对于算例Kroad100与Krocd100,MACS覆盖的目标空间虽然比P-MMAS广,但MACS的边界点均被P-MMAS的解支配.

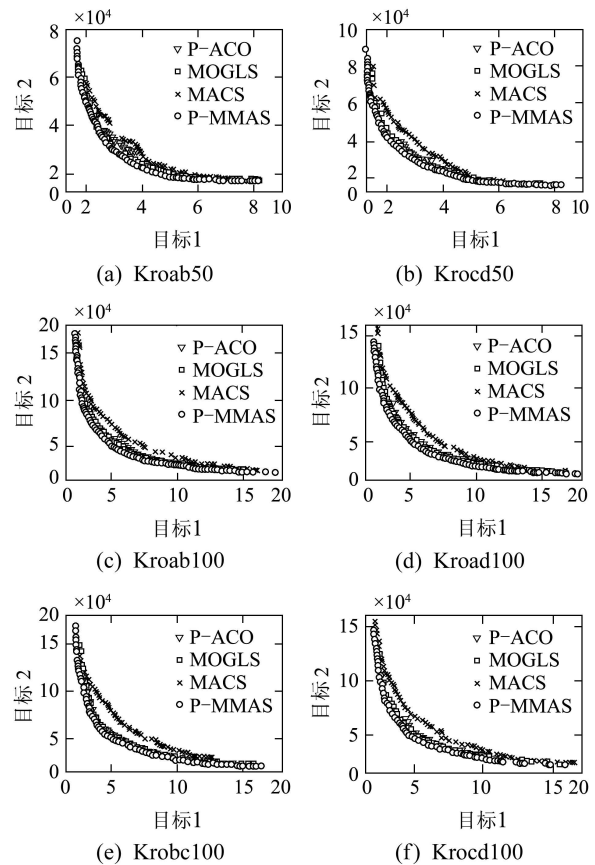


图1 4种算法产生的近似Pareto前端

Fig. 1 Approximate Pareto fronts returned by four algorithms

表1列出了4种算法经10次独立运行后得到的 M_1^* , M_2^* , M_3^* 指标均值,其中最优化结果用粗体标出.由此可知:① 对于 M_1^* 指标,P-MMAS的性能最好,MOGLS次之,P-ACO优于MACS位居第3.② 对于 M_2^* 与 M_3^* 指标,P-MMAS的性能最好,而MACS优于MOGLS位居第2,P-ACO的性能最差.③ 对于算例Krocd100,MACS的 M_3^* 指标虽优于P-MMAS,但由图1可以看出,MACS的这些边界点均被P-MMAS的解支配.

表 1 P-MMAS, MOGLS, MACS与P-ACO(分别用 1, 2, 3, 4表示)对应的 M_1^* , M_2^* , M_3^* 指标均值
Table 1 Mean values of M_1^* , M_2^* , M_3^* metrics corresponding to P-MMAS, MOGLS, MACS and P-ACO

	M_1^*				M_2^*				M_3^*			
	1	2	3	4	1	2	3	4	1	2	3	4
Kroab50	252.15	1389.6	3238.8	3074.4	109.07	32.66	51.70	1.25	86250	78905	76354	12700
Krocd50	283.48	1287.2	4190.9	2399.2	124.44	38.02	55.41	2.49	96542	80924	83178	16520
Kroab100	969.85	3154.3	8348.1	4384.9	95.43	44.30	59.90	0	207140	161290	197300	14504
Kroad100	698.34	2745.5	9008.6	4039.5	99.27	44.38	64.79	0.07	195010	175070	187650	18445
Krobc100	931.74	2732.2	8973.1	4080.5	93.31	48.25	65.67	0.03	199130	163270	190390	18112
Krocd100	999.09	2956.5	8343.8	4208.2	91.51	43.43	63.98	0	189420	163080	193430	16130

图2为几种算法的 C 指标盒形图, 横坐标1-6对应算例Kroab50, Krobc50, Kroab100, Kroad100, Krobc100, Krocd100. 结合图1可以看出P-MMAS产生的非支配解几乎完全支配着其余算法的解,

仅个别解被MOGLS支配; MOGLS的 C 指标仅次于P-MMAS, 支配MACS与P-ACO中大多数的解; 由于P-ACO能在目标空间中间部分产生性能良好的非支配解, 所以其 C 指标优于MACS.

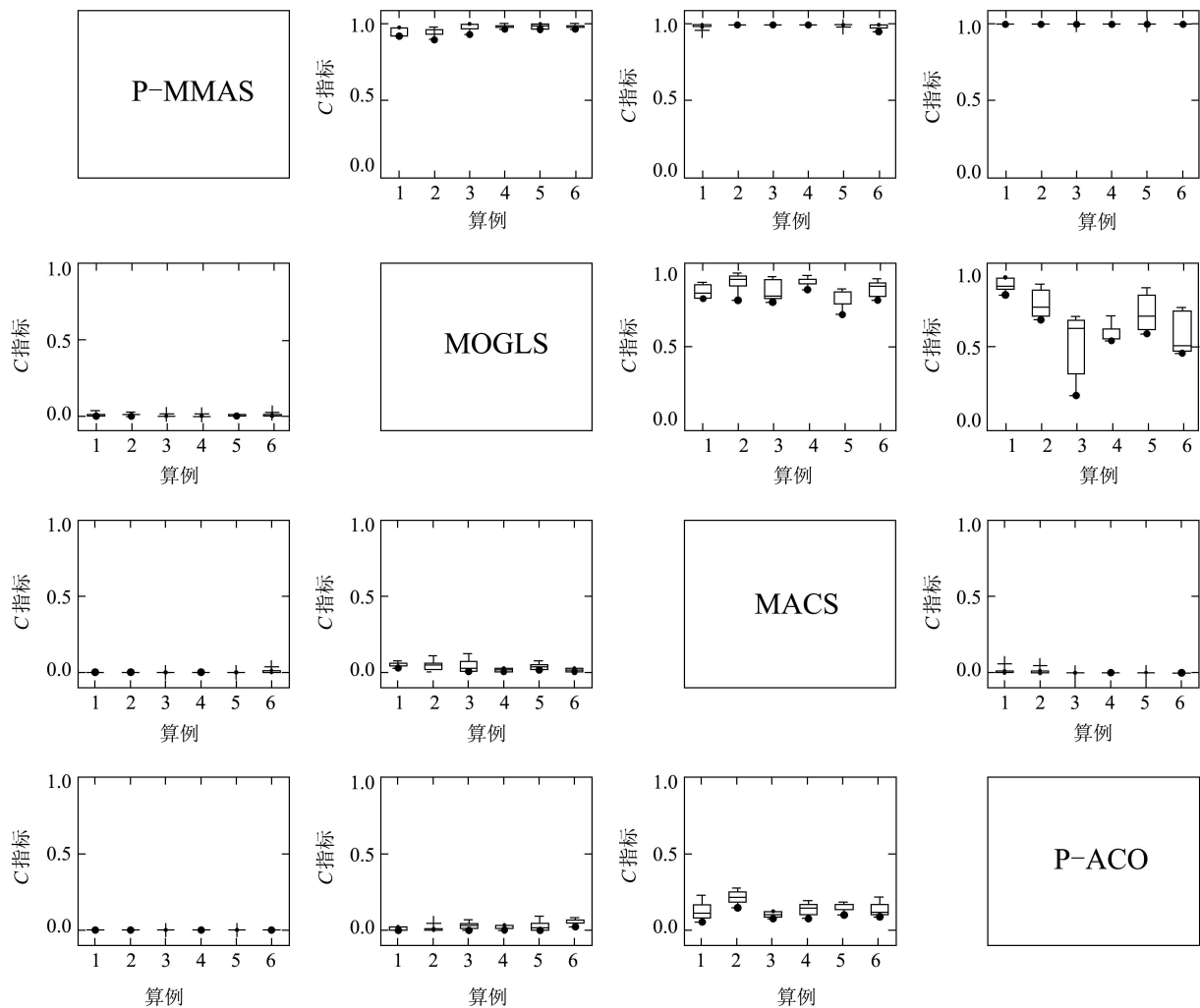


图 2 C 指标盒形图

Fig. 2 Box plots based on C metric

综合以上定性与定量分析, 可以看出对于双目标TSP来说, P-MMAS算法的优化性能是最好的; 而MOGLS由于使用了局部搜索和距离保留交叉

算子, 使得其综合性能仅次于P-MMAS; 另两种多目标蚁群算法MACS与P-ACO性能各有所长, MACS解的分布性能较好, 而P-ACO的解更接近

Pareto最优解.

5 热轧批量计划优化应用(Application to hot rolling batch planning problem)

5.1 问题描述(Problem description)

编制热轧批量计划是指从预选池的 N 块板坯中编制出 M 个满足轧制规范的轧制单元. 一个完整的轧制单元由烫辊材与主体材两部分组成, 单元中的板坯编制在宽度上呈“乌龟壳”结构. 烫辊材主要用于加热轧辊, 一般数量较少, 容易确定. 主体材是轧制单元的主要组成部分, 编制计划时通常要满足以下规程: ① 板坯在硬度和厚度上要求变化平稳. ② 带钢宽度总体上呈递减变化, 且同宽轧制长度不能超过某一上限. ③ 整个轧制单元总轧制长度应小于轧辊的最大轧制长度, 且应尽量不小于最小轧制长度, 以减少换辊次数. 热轧批量计划问题可归结为多旅行商问题^[14]或车辆路径问题^[15], 本文在此基础上, 建立了带能力约束的双目标TSP模型, 并通过启发式规则与P-MMAS进行求解, 同时给出了确定最终方案的决策方法.

5.2 数学模型(Mathematical model)

假设有 N 块板坯需要编制 M 个轧制单元, 这 N 块板坯可以看成 N 个城市, M 个轧制单元可以看成是 M 个旅行商. 引进 M 个虚拟节点, 其编号为 $N+1, \dots, N+M$, 那么热轧批量计划可建成多旅行商模型. 借助单目标TSP模型的表达, 等价于一个旅行商访问 $N+M$ 个城市. 模型中用到的变量和参数定义如下^[14]:

对于 $i, j \in \{1, \dots, N\}$,

$$x_{ij} = \begin{cases} 1, & \text{如果板坯}j\text{直接在板坯}i\text{之后生产,} \\ 0, & \text{否则.} \end{cases}$$

对于 $i \in \{N+1, \dots, N+M\}, j \in \{1, \dots, N\}$,

$$x_{ij} = \begin{cases} 1, & \text{如果板坯}j\text{是第}i-N\text{个计划的第1个,} \\ 0, & \text{否则.} \end{cases}$$

对于 $i \in \{1, \dots, N\}, j \in \{N+1, \dots, N+M\}$,

$$x_{ij} = \begin{cases} 1, & \text{如果板坯}i\text{是第}j-N\text{个计划的最后1个,} \\ 0, & \text{否则.} \end{cases}$$

以往模型使用相邻板坯间的宽度、厚度和硬度的跳跃惩罚之和作为板坯间的“距离”, 从而将其转化为单目标TSP模型^[14], 考虑到轧制计划对宽度跳跃有较高要求(需满足“乌龟壳”形状), 本文将宽度跳跃惩罚单独提取出来, 将其转化为双目标TSP模型, 其意义在于: ① 从Pareto前端中可以反映出两个目标的“折中”关系, 如损失多

少宽度指标, 才能提升要求的厚度与硬度指标, 决策者可以根据Pareto前端进行合理决策. ② 将以往的“先验决策”转化为“后验决策”, 为决策者提供了更多选择(以往单目标模型将板坯宽度、厚度和硬度跳跃惩罚的加权系数固定为1, 其最优解仅是Pareto最优解集中的一个解). 另外本文考虑轧制单元的能力约束, 使模型更加符合轧制规范. 令 d_{ij}^1 表示从板坯 i 转换到板坯 j 时宽度跳跃的惩罚, d_{ij}^2 表示从板坯 i 转换到板坯 j 时厚度与硬度跳跃的惩罚, 可分别定义为:

$$d_{ij}^1 = P_{ij}^W, i, j \in \{1, \dots, N\}, i \neq j,$$

$$d_{ij}^2 = P_{ij}^G + P_{ij}^H, i, j \in \{1, \dots, N\}, i \neq j,$$

$$d_{ij}^k = 0, i \in \{1, \dots, N\}, j \in \{N+1, \dots, N+M\},$$

$$d_{ij}^k = 0, i \in \{N+1, \dots, N+M\}, j \in \{1, \dots, N\},$$

$$d_{ij}^k = \infty, i, j \in \{N+1, \dots, N+M\}, i \neq j,$$

$$d_{ii}^k = \infty, i \in \{1, \dots, N+M\},$$

其中: $k = 1, 2$, $P_{ij}^W, P_{ij}^G, P_{ij}^H$ 分别为宽度、厚度和硬度的跳跃惩罚.

考虑轧制单元的能力约束, 设 $S_k \subseteq \{1, \dots, N\}$ 为第 k 个轧制单元包含的板坯集合; l_i 为板坯 i 的轧制长度; L 为轧制单元的最小轧制长度, U 为最大轧制长度; W_{ik} 为第 k 个轧制单元中与板坯 i 连续同宽的板坯集合; L_w 为每个轧制单元中同宽轧制长度上限.

在上述变量定义下, 令 $n = N+M$, 热轧轧制批量计划数学模型可表达为:

$$\begin{aligned} \min f = & \left(\sum_{i=1}^n \sum_{j=1}^n x_{ij} d_{ij}^1, \sum_{i=1}^n \sum_{j=1}^n x_{ij} d_{ij}^2 \right), \\ \text{s.t.} & \begin{cases} \sum_{j=1}^n x_{ij} = 1, i \in \{1, \dots, n\}, \\ \sum_{i=1}^n x_{ij} = 1, j \in \{1, \dots, n\}, \\ \sum_{i \in V} \sum_{j \in V} x_{ij} \leq |V| - 1, V \subset \{1, \dots, n\}, \\ x_{ij} \in \{0, 1\}, i, j \in \{1, \dots, n\}, \\ L \leq \sum_{i \in S_k} l_i \leq U, k \in \{1, \dots, M\}, \\ \sum_{j \in W_{ik}} l_j \leq L_w, i \in S_k, k \in \{1, \dots, M\}. \end{cases} \end{aligned} \quad (10)$$

5.3 优化算法(Optimization algorithm)

由式(10)可知, 热轧批量计划已转化带能力约束的双目标TSP问题. 本文在P-MMAS中采用如下启发式规则处理能力约束: 从虚拟板坯 $N+1$ 出发, 在预选池中选择最大宽度的板坯作为该轧制单元主体材的第1块板坯, 然后根据状态转移策略

选择下一块板坯, 如果该板坯满足同宽连续轧制约束, 将该板坯加入轧制单元, 否则选择不同宽度的板坯. 当该轧制单元中板坯总长度达到 L 时, 依概率 $P = (\sum_{i=1}^K l_i - L)/(U - L)$ 选择下一块虚拟板坯(K 为当前轧制单元中的板坯数); 当长度超过 U 时, 将最后一块板坯释放回预选池, 并以概率1选择下一块虚拟板坯, 开始下一轧制单元的构造. 重复以上过程, 直到 M 个轧制单元全部完成构造, 这样就形成一条蚂蚁路径.

由于P-MMAS产生的是多个近似Pareto最优解, 需要从中选择一个最终方案, 即后验决策. 本文采用“约束过滤”决策方法, 使用轧制规范中其它约束来逐个检验近似Pareto最优解, 违反约束次数最少的解作为最终方案. 如有若干个解满足要求, 则选择使宽度跳跃惩罚最小的方案作为最终轧制计划. 具体来说, 本文选择的“过滤约束”为: ① 宽度、厚度、硬度不允许同时跳跃. ② 厚度和硬度不能反复跳跃. 这两个约束不易使用启发式规则处理, 将它们作为“过滤约束”确定最终方案, 不仅使启发式规则设计简单, 求解效率提高, 而且使最终方案尽可能满足这两个约束. 应指出的是, 双目标模型使这种处理成为可能, 因为它一次性提供多个Pareto最优解可供验证, 而单目标模型由于仅有一个优化解, 必须在建模时就考虑这两个约束, 增加了算法设计难度.

5.4 实验结果(Experimental results)

为验证上述模型与算法的有效性, 选择某热轧厂132块实际板坯数据进行实验. 模型中的参数为:

$$N = 132, M = 2, L = 60, U = 70, L_w = 15,$$

目标函数中宽度、厚度和硬度跳跃惩罚使用文献[14]的参数, 宽度反跳惩罚取3000. P-MMAS参数同4.1节测试环境, 迭代次数设为200.

图3为热轧批量计划的近似Pareto前端, 图3中标记为五角星的解为根据“约束过滤”方法确定的最终计划(见表2), 可知该计划很好地权衡了宽度、厚度与硬度跳跃. 表2中列出了P-MMAS, MGA^[14]与手工编制的热轧批量计划, 其中目标1, 2分别对应宽度跳跃惩罚、厚度与硬度跳跃惩罚. 从中可知:

1) P-MMAS与MGA编制的计划优于手工计划, 两个目标的总惩罚均比手工计划的小.

2) P-MMAS与MGA编制的计划性能相当, 但由于MGA仅能提供一个优化解, 且在算法设计中需考虑更多的约束, 因此综合性能较P-MMAS差.

3) 第1个轧制单元的手工计划最好, 这是由于计划员编制计划时采用串行策略, 类似于贪婪方法, 先编制的计划较好, 后编制的计划较差. 综合以上结果, 说明了P-MMAS的优越性.

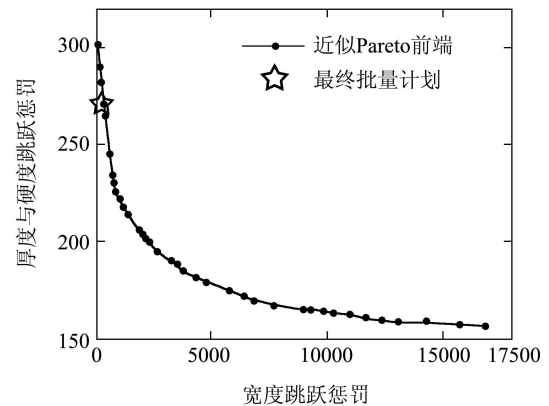


图3 热轧批量计划的近似Pareto前端

Fig. 3 Approximate Pareto front of hot rolling batch plan

表2 3种方法编制的热轧批量计划比较

Table 2 Comparison among hot rolling batch plans optimized by three methods

方法	单元	板坯数	轧制长度	目标1	目标2
P-MMAS	1	66	68.5	160	136
	2	66	67.3	169	135
MGA	1	66	67.8	165	134
	2	66	68.0	167	138
手工	1	66	69.1	153	120
	2	66	66.7	220	172

6 结论(Conclusions)

本文针对双目标TSP提出的多目标蚁群算法—Pareto最大最小蚂蚁算法, 通过状态转移策略、信息素更新策略、局部搜索策略、多样性保持策略与信息素平滑机制的综合作用, 使得该算法能够快速搜索到分布均匀的近似Pareto前端. 定性与定量的性能比较以及在热轧批量计划优化中的应用验证了该算法的有效性与实用性. 作为一种全新的多目标蚁群算法, P-MMAS算法需要进一步研究的课题包括对算法的改进以解决高维多目标TSP, 以及在其它多目标组合优化问题中的应用等.

参考文献(References):

- [1] DAS I, DENNIS J. A closer look at drawbacks of minimizing weighted sums of objectives for Pareto set generation in multicriteria optimization problems[J]. *Structural Optimization*, 1997, 14(1): 63 - 69.

- [2] JASZKIEWICZ A. Genetic local search for multi-objective combinatorial optimization[J]. *European Journal of Operational Research*, 2002, 137(1): 50 – 71.
- [3] BARÁN B, SCHAERER M. A multiobjective ant colony system for vehicle routing problem with time windows[C] // *The 21st IASTED International Conference Applied Informatics*. Calgary: ACTA Press, 2003: 97 – 102.
- [4] DOERNER K F, GUTJAHR W J, HARTL R F, et al. Pareto ant colony optimization with ILP preprocessing in multiobjective project portfolio selection[J]. *European Journal of Operational Research*, 2006, 171(3): 830 – 841.
- [5] KNOWLES J D, CORNE D W. Approximating the nondominated front using the Pareto archived evolution strategy[J]. *Evolutionary Computation*, 2000, 8(2): 149 – 172.
- [6] ZITZLER E, THIELE L. Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach[J]. *IEEE Transactions on Evolutionary Computation*, 1999, 3(4): 257 – 271.
- [7] DORIGO M, MANIEZZO V, COLORNI A. Ant system: optimization by a colony of cooperating agents[J]. *IEEE Transactions on System, Man and Cybernetics, Part B*, 1996, 26(1): 29 – 41.
- [8] DORIGO M, GAMBARDELLA L M. Ant colony system: a cooperative learning approach to the travelling salesman problem[J]. *IEEE Transactions on Evolutionary Computation*, 1997, 1(1): 53 – 66.
- [9] STÜTZLE T, HOOS H H. Max-min ant system[J]. *Future Generation Computer System*, 2000, 16(8): 889 – 914.
- [10] DORIGO M, STÜTZLE T. *Ant Colony Optimization*[M]. Cambridge, American: MIT Press, 2004.
- [11] ANGUS D, WOODWARD C. Multiple objective ant colony optimization[J]. *Swarm Intelligence*, 2009, 3(1): 69 – 85.
- [12] GARCÍA-MARTÍNEZ C, CORDÓN O, HERRERA F. A taxonomy and an empirical analysis of multiple objective ant colony optimization algorithms for bi-criteria TSP[J]. *European Journal of Operational Research*, 2007, 180(1): 116 – 148.
- [13] 吴婷, 陈玉旺, 汪焯. 基于极值动力学的自组织优化算法求解 TSP 问题[J]. *控制理论与应用*, 2010, 27(6): 715 – 720. (WU Ting, CHEN Yuwang, WANG Ye. Self-organized optimization algorithm with extremal dynamics for the traveling salesman problem[J]. *Control Theory & Applications*, 2010, 27(6): 715 – 720.)
- [14] TANG L X, LIU J Y, RONG A Y, et al. A multiple traveling salesman problem model for hot rolling scheduling in Shanghai Baoshan Iron & Steel Complex[J]. *European Journal of Operational Research*, 2000, 124(2): 267 – 282.
- [15] 刘士新, 宋健海, 周山长. 热轧带钢轧制批量计划优化模型及算法[J]. *控制理论与应用*, 2007, 24(2): 243 – 248. (LIU Shixin, SONG Jianhai, ZHOU Shanchang. Model and algorithm for solving hot strip rolling batch planning problems[J]. *Control Theory & Applications*, 2007, 24(2): 243 – 248.)

作者简介:

贾树晋 (1982—), 男, 博士研究生, 主要研究方向为智能优化算法、生产计划优化, E-mail: jiashujin_1@163.com;

朱俊 (1982—), 男, 博士研究生, 主要研究方向为离散事件仿真、生产计划优化, E-mail: yjworm@126.com;

杜斌 (1957—), 男, 教授级高级工程师, 博士生导师, 主要研究方向为生产计划优化、过程控制, E-mail: dubin@baosteel.com;

岳恒 (1968—), 男, 教授, 主要研究方向为复杂工业过程建模与优化, E-mail: yueheng@mail.neu.edu.cn.