

文章编号: 1001-0920(2012)09-1313-07

一种改进的多目标粒子群优化算法及其应用

冯琳, 毛志忠, 袁平

(东北大学信息科学与工程学院, 沈阳 110819)

摘要: 针对多目标粒子群优化算法在求解约束优化问题时存在难以兼顾收敛性能和求解质量这一问题, 提出一种基于免疫网络的改进多目标粒子群优化算法. 该算法通过免疫网络互通种群最优信息达到粒子群算法与人工免疫网络算法的协同搜索, 同时给出了速度迁移策略、自适应方差变异策略和基于聚类的免疫网络策略. 最后将所提出的方法应用于求解电弧炉供电优化模型, 达到了减少电量消耗、缩短冶炼时间、延长炉衬使用寿命的目的, 同时表明了该算法的有效性.

关键词: 粒子群算法; 多目标约束优化; 速度迁移; 自适应变异; 聚类免疫网络; 供电策略

中图分类号: TP301.6

文献标志码: A

An improved multi-objective particle swarm optimization algorithm and its application

FENG Lin, MAO Zhi-zhong, YUAN Ping

(School of Information Science and Engineering, Northeastern University, Shenyang 110819, China. Correspondent: FENG Lin, E-mail: fenglin@ise.neu.edu.cn)

Abstract: Considering that the multi-objective particle swarm optimization(MOPSO) algorithm can not give simultaneously attention to convergence performance and solutions quality when it deals with constrained optimization problems, an improved MOPSO algorithm based on immune network(IN-MOPSO) is proposed. In IN-MOPSO, the information of populations exchange through immune network in IN-MOPSO in order to achieve cooperative search of both MOPSO and artificial immune network(AIN) for solution space. Meanwhile, an improved migration method of particle velocity, an improved adaptive variance mutation method and clustering immune network are proposed in order to enhance the function of MOPSO and AIN. The global convergence properties and convergence rate of the improved algorithm are analyzed and described. Finally, the algorithm is applied to optimize the steelmaking process in practice, which reduces the electric energy consumption, shortens smelting time and improves lifetime of the furnace lining. The result shows the effectiveness of the algorithm.

Key words: particle swarm optimization algorithm; multi-objective constrained optimization; velocity migration; adaptive mutation; clustering immune network; power supply strategy

1 引言

电弧炉炼钢过程是一个多变量、非线性的复杂过程, 合理的供电策略不仅能保证操作顺利进行, 而且有助于降低电量消耗、电极损耗、炉衬消耗以及缩短冶炼周期, 带来良好的经济效益^[1]. 电炉炼钢过程中的供电曲线优化问题, 属于典型的有约束最优化问题, 较难求解.

粒子群优化(PSO)算法是一种新兴的群体智能算法, 以其实现容易、信息单向传递、设置参数少以

及高效并行搜索等优点已广泛应用于各类复杂环境中的优化问题的求解^[2]. 但是, 随着优化问题的复杂程度越来越高, 尤其对于带有约束条件的多目标优化问题, 多目标粒子群算法(MOPSO)在求解过程中存在难以兼顾算法的收敛性能和求解质量的问题.

目前, MOPSO算法主要采用非支配排序^[3-4]和 ϵ 非支配排序^[5-7]策略, 并根据拥挤程度对粒子进行筛选. 这两种排序策略在多数情况下可以获得良好的收敛性和多样性, 但对于复杂且带有约束条件的多目标

收稿日期: 2011-08-04; 修回日期: 2011-10-26.

基金项目: 国家自然科学基金项目(61004083).

作者简介: 冯琳(1979-), 女, 讲师, 博士生, 从事复杂工业过程建模与优化等研究; 毛志忠(1961-), 男, 教授, 博士生导师, 从事复杂工业过程控制、建模与优化等研究.

优化问题,存在运算效率不高,后期收敛速度较慢的问题;在提高算法收敛速度和保持种群多样性能方面,主要采用变异操作^[7-9](速度和位置变异),或小生境技术^[10],但仿真表明收敛速度和均匀多样性不能同时得到满足。

基于上述分析,本文针对带约束的多目标优化问题,提出一种基于免疫网络(AIN)^[11]的多目标粒子群优化算法,并改进了速度概率迁移算子、自适应变异算子和聚类网络压缩策略。最后通过求解以降低吨钢电耗、缩短冶炼时间、减小电极消耗和增加炉衬使用寿命为目标的多目标优化模型,表明了所提出的改进优化算法的有效性。

2 多目标约束优化问题描述

以最大化问题为例,多目标约束优化问题可以描述为^[12]

$$\begin{aligned} & \max \\ & \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x})), \\ & \mathbf{x} = (x_1, x_2, \dots, x_n) \subset X \in \Omega \in S \in R^n. \quad (1) \\ & \text{s.t.} \\ & g_j(\mathbf{x}) \leq 0, j = 1, 2, \dots, l; \\ & h_j(\mathbf{x}) = 0, j = l + 1, \dots, p. \end{aligned}$$

其中: \mathbf{x} 为决策向量; Ω 为可行域; S 为决策空间,是 R^n 中的 n 维长方体 ($a(i) \leq x_i \leq b(i), a(i), b(i)$ 为常数, $i = 1, 2, \dots, n$); $\mathbf{f}(\mathbf{x}), g_j(\mathbf{x}), h_j(\mathbf{x})$ 均为 R^n 上的多元函数, $\mathbf{f}(\mathbf{x})$ 为目标函数, $g_j(\mathbf{x}) \leq 0$ 为第 j 个不等式约束条件, $h_j(\mathbf{x}) = 0$ 为第 j 个等式约束条件; l 为不等式约束条件的个数; $p - l$ 为等式约束条件的个数。

定义 1 (Pareto 支配) 向量 $\mathbf{u} = (u_1, u_2, \dots, u_m)$, 支配向量 $\mathbf{v} = (v_1, v_2, \dots, v_m)$, 表示为 $\mathbf{u} \prec \mathbf{v}$, 当且仅当 $\forall i \in \{1, 2, \dots, m\}$ 满足 $f(u_i) \geq f(v_i)$, 并且 $\exists i \in \{1, 2, \dots, m\}$ 使得 $f(u_i) > f(v_i)$ 。

图 1 给出了具有 2 个待优化目标问题的解的支配关系。

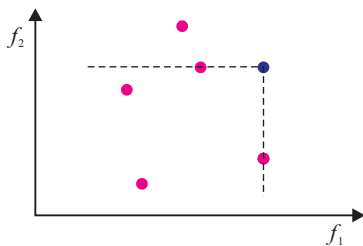


图 1 目标空间中的支配关系

定义 2 (Pareto 最优解集) 对于给定的多目标优化问题 $\mathbf{f}(\mathbf{x})$, Pareto 最优解集 ρ^* 定义为

$$\rho^* = \{\mathbf{x}_u \in X | \neg \exists \mathbf{x}_v \in X, \mathbf{x}_u \prec \mathbf{x}_v\}.$$

Pareto 最优解集中的个体也称为非支配个体。

定义 3 (Pareto 前沿) 对于给定的多目标优化问题 $\mathbf{f}(\mathbf{x})$ 和 Pareto 最优解集 ρ^* , Pareto 前沿 ρf^* 定义为

$$\rho f^* = \{\mathbf{f}(\mathbf{x}_u) | \mathbf{x}_u \in \rho^*\}.$$

图 2 为具有 2 个待优化目标问题的 Pareto 前沿。

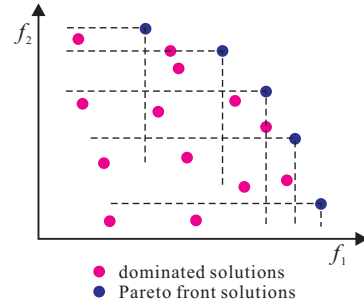


图 2 双目标空间中的非劣前沿

3 基于免疫网络的多目标粒子群算法

3.1 IN-MOPSO 算法思想

本文提出的基于免疫网络的改进多目标粒子群优化算法(IN-MOPSO)的主要思想是: 1) 利用 AIN 的全局搜索能力来增强 MOPSO 的全局开发能力; 2) 通过 AIN 算法与 MOPSO 算法并行协同搜索, 迅速找到问题 Pareto 最优解集, 有效地均衡算法的探索和开发能力。

在算法执行过程中, 网络压缩操作起着协调 IN-MOPSO 算法的全局与局部开发能力的重要作用。为保证算法具有较快的收敛速度并具有全局搜索能力, 本文采用“先将(网络)数据空间划分成有限个单元的网格结构, 再计算各单元中网络节点的相似度, 选取各区间中节点相似度小于某一阈值的节点作为新网络节点”的策略, 在设定网络中所容纳记忆细胞数量上限后, 当新节点数目小于记忆细胞数量上限时随机初始化其他网络节点。

3.2 优化目标转换

将多目标约束优化问题转换为多目标优化问题后, 利用多目标优化技术处理转换后的问题^[13]。针对式(1), 将约束优化问题转换为具有 2 个目标的问题, 第 1 个目标为目标函数 $\mathbf{f}(\mathbf{x})$, 第 2 个目标为个体满足约束条件程度函数 $\varphi(\mathbf{x})$, 即

$$F(\mathbf{x}) = (\mathbf{f}(\mathbf{x}), \varphi(\mathbf{x})). \quad (2)$$

若群体中的个体 \mathbf{x} 满足第 j 个约束条件的程度可以表示为

$$\varphi_{g_j}(\mathbf{x}) = \begin{cases} 1, & g_j(\mathbf{x}) \leq 0; \\ |\delta_j/g_j(\mathbf{x})|, & 0 \leq g_j(\mathbf{x}) \leq \delta_j; \\ 0, & \text{else.} \end{cases} \quad (3a)$$

$$\varphi_{h_j}(\mathbf{x}) = \begin{cases} |\gamma_j/h_j(\mathbf{x})|, & |h_j(\mathbf{x})| \leq \gamma_j; \\ 0, & \text{else.} \end{cases} \quad (3b)$$

其中: δ_j 和 γ_j 的取值根据具体问题具体设置, 但一般

选取范围为 $[0, 1]$. 则

$$\varphi(\mathbf{x}) = \sum_{j=1}^l \varphi_{g_j}(\mathbf{x}) + \sum_{j=l+1}^p \varphi_{h_j}(\mathbf{x}). \quad (4)$$

3.3 个体排序

本文所讨论的问题是最大化问题, 因此设定适应度值越大, 结果越好. 对于多目标优化问题, 本文采用基于 Pareto 的方法并加以改进.

免疫系统中的细胞具有多样性和自我调节能力. 免疫调节时, 具有亲和度大且浓度较低的细胞会受到激励, 而亲和度小或浓度较高的细胞将受到抑制; 因此, 应综合考虑细胞亲和度以及细胞浓度, 利用激励度算子来评价细胞. 细胞激励度可由下式计算:

$$\alpha(x_i^t) = \frac{\|F\|_2}{\|\beta\|_2} = \frac{\|F(x_i^t)\|_2}{\left\| \frac{1}{\text{Pop}} \sum_{i=1}^n F(x_i^t) \right\|_2}. \quad (5)$$

其中: 细胞浓度 $\beta = \frac{1}{\text{Pop}} \sum_{i=1}^n F(x_i^t)$, Pop 为种群规模. 计算出第 t 代种群中的个体细胞激励度值后, 按计算结果由大到小对细胞种群重新排序, 并移除激励度 α 小于 0.4 的细胞.

3.4 改进的 MOPSO 算法

为了提高 MOPSO 的全局寻优能力, 受自适应逃逸微粒群算法^[4]的启发, 提出一种针对 MOPSO 算法的改进策略——速度概率迁移算子, 以提高算法跳出局部极值的能力. 该策略赋予微粒 x_i^t 一个速度迁移阈值 Z , 当微粒速度 $\|v_i(t)\| \geq Z$ 时, 给其一个变异操作, 即

$$v_i^{t+1} = \varepsilon x_{\max}, \lambda_p < p_1. \quad (6)$$

其中: λ_p 为 $[0, 1]$ 之间的随机数, p_1 为迁移变异概率, ε 取值符合标准正态分布函数, 速度迁移阈值 Z 和参数 θ 根据具体问题设定. 因为迁徙操作只对微粒的速度进行变异, 所以迁移后的微粒只忘记了个体最优极值, 而记住了微粒群的全局最优点, 从而有利于微粒摆脱局部极值的吸引, 增强算法的全局寻优能力.

3.5 改进的 AIN 算法

3.5.1 自适应方差变异算子

在实数编码算法中, 变异的原理如下: 将一个较小的扰动加在变异源个体的位置上, 使其稍微偏离原来的位置, 落入源个体邻域中的另一个位置, 实现变异源邻域的搜索.

在上述基本原理的基础上, 提出一种能体现细胞自适应性的变异策略——方差变异算子. 改进的变异算子可描述为

$$T_m(x_i^t) = \begin{cases} x_i^t + (\lambda_m - 0.5) \times E, & \lambda_m < p_2; \\ x_i^t, & \text{else.} \end{cases} \quad (7)$$

其中: $E = \frac{1}{m} \sum_{j=1}^m (f_{\text{avg}} - (f_j(x_i^t)))^2$ 为记忆细胞亲和度

方差, $f_{\text{avg}} = \frac{1}{m} \sum_{j=1}^m (f_j(x_i^t))$ 为记忆细胞当前的平均适

应度值, p_2 为变异概率, λ_m 为 $[0, 1]$ 之间的随机数. 该算子利用记忆细胞亲和度方差来衡量细胞的裂解程度. 细胞裂解程度大时, 亲和度方差值大; 细胞裂解程度小时, 亲和度方差值小. 这样, 随着算法进化过程的执行, 细胞的裂解程度会越来越小, 最后集中在所找到的最优解上, 因此亲和度方差值能够体现记忆细胞的一种自适应性.

3.5.2 聚类网络压缩

在 AIN 算法中, 记忆细胞组成了免疫网络, 并被称为免疫网络的网络节点. 保持免疫网络的动态性能, 主要根据那些具有较高亲和度的记忆细胞、执行细胞间的刺激和抑制来实现. 但是, 目前的免疫网络模型普遍存在过分依赖网络节点的增减以及运算量大、执行速度慢等缺点. 因此, 如何选择网络节点便成为免疫网络压缩的重点.

网络压缩的最基本原则是, 在若干记忆细胞中选取最具有代表性的记忆细胞对免疫网络进行压缩, 以防止网络由于记忆细胞过于相似而使得细胞多样性降低的问题. 在该基本原则的基础上, 将基于网格的数据聚类方法引入网络压缩, 提出一种改进策略——聚类网络压缩.

聚类网络压缩主要针对网络节点进行操作. 设

$$A(t) = \{a_i | a_i = \min\{F(x_i^t)\}, i = 1, 2, \dots, n\},$$

$$C(t) = \{c_i | c_i = \max\{F(x_i^t)\}, i = 1, 2, \dots, n\}.$$

1) 将所有细胞组成的表现型空间 $[a_i, c_i] (i = 1, 2, \dots, n)$ 划分为若干个子区间 $[a_i, b_{i1}), [b_{i1}, b_{i2}), \dots, [b_{i(\text{Pop}-2)}, b_{i(\text{Pop}-1)}), [b_{i(\text{Pop}-1)}, c_i]$ 且 $|b_{i(j+1)} - b_{ij}| = (a_i - c_i)/\text{Pop}$, 则对于落入同一个开区间 $[b_{ij}, b_{i(j+1)})$ 的 q 个细胞, 按下式计算细胞间的相似度:

$$S(x_i^t, x_j^t) = \begin{cases} 1, & F(x_i^t) = F(x_j^t); \\ \frac{1}{1 + |F(x_i^t) - F(x_j^t)|}, & \text{else.} \end{cases} \quad (8)$$

2) 按下式计算细胞的 rank 值:

$$\text{rank} = \begin{cases} 0, & S(x_i^t, x_j^t) = 1 \text{ or } \alpha(x_i^t) \leq 0.4; \\ 1, & \text{else.} \end{cases} \quad (9)$$

3) 根据新的细胞 rank 值排序并移除 rank = 0 的细胞.

4) 随机生成新细胞并避免重复将其加入到免疫网络中, 直到达到预定的免疫网络规模.

聚类网络压缩细胞等级分类如图 3 所示, 其中 P_1 与 P_2 的亲和度相同, 两点的细胞相似度等于 1. 若 P_1

点的激励度值 $\alpha(x_i^t) = 0.6$, P_2 点的激励度值 $\alpha(x_i^t) = 0.2$, 则根据聚类网络压缩方法, 应保留点 P_1 , 移除点 P_2 . 在进化过程中, 细胞的相似度越大, 说明细胞的多样性越小, 如果整个网络中被大量相似细胞所占据, 则很容易陷入局部收敛. 因此, 移除相似度为 1 且 $\alpha(x_i^t) \leq 0.4$ 的那些细胞有利于整个免疫网络的动态性. 采用这种聚类网络压缩方法, 可以避免相似细胞占据当前的免疫网络, 阻止局部收敛, 是维持免疫网络均匀性和多样性的主要动力.

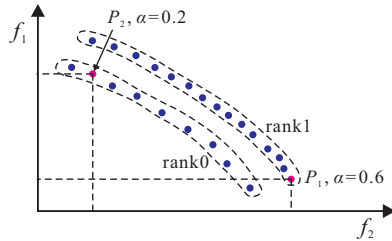


图 3 聚类网络压缩细胞等级分类

3.6 IN-MOPSO 算法流程

Step 1: 随机初始化种群. 确定种群规模 Pop , 迭代次数 D_{max} , 惯性权重 ω , 迁移变异概率 p_1 , 变异概率 p_2 以及参数 $c_1, c_2, \delta_j, \gamma_j, \theta, Z$.

Step 2: 判断是否满足算法终止条件. 如果满足, 则输出结果, 算法运行结束; 如果不满足, 则按顺序执行下一步.

Step 3: 在种群上分别根据式 (6) 和如下公式计算粒子的速度和位置:

$$v_i^{t+1} = \omega v_i^t + c_1 r_1 (P_i^t - x_i^t) + c_2 r_2 (G^t - x_i^t), \quad (10)$$

$$x_i^{t+1} = v_i^{t+1} + x_i^t. \quad (11)$$

同时根据下式和式 (7) 对细胞进行克隆和自适应变异:

$$T_c(x_i^t) = \text{clone}(x_i^t). \quad (12)$$

其中: $\text{clone}(x_i^t)$ 为 m_i 个与 x_i^t 相同的克隆构成的集合, m_i 为细胞克隆个数.

Step 4: 对所得到的种群进行精英选择, 组成细胞网络, 并判断细胞网络中的网络节点是否满足多样性要求. 如果满足, 则执行 Step 2; 如果不满足, 则按顺序执行下一步.

Step 5: 对细胞网络进行聚类网络压缩, 继续执行 Step 2.

3.7 参数设置分析

参数设置的大小对算法的性能有很大影响, 为此通过以下定性分析来确定 IN-MOPSO 算法中的最佳参数.

首先设置算法最大迭代次数 $G_{max} = 1000$. 考虑到惯性权重 ω 以及学习因子 c_1 和 c_2 对算法性能的影响,

兼顾种群多样性和算法收敛速度, 取 $\omega = 0.7298$, $c_1 = c_2 = 1.49618$. 优化目标转换时, 参数取值范围为 $\delta < 0.5, \gamma < 0.05$. 此外, 将精度 $\varepsilon < 10^{-6}$ 作为算法的终止条件. 测试过程中每次仅在特定区间内改变 IN-MOPSO 中的某一参数.

利用 IN-MOPSO 算法分别优化文献 [15] 中的 3 个测试函数 20 次, 对自适应变异概率 p_2 的敏感性分析如表 1 所示. 考虑到变异概率过大会严重影响算法的全局搜索性能, 实验中设置 $p_2 \leq 0.6$. 从表 1 可以看出, 自适应变异概率取 0.2, 0.25 和 0.3 时算法得出的寻优成功率均较好. 但是, 考虑到变异概率设置过大时, 算法已获得的优秀细胞被破坏的概率会变大, 因此自适应变异概率 p_2 取 0.2 是合理的.

表 1 自适应变异概率的敏感性分析

p_2	寻优成功率 / %		
	CONSTR	TNK	REV
0.15	93.1	94.2	93.0
0.2	99.2	99.5	99.3
0.25	98.5	98.2	98.1
0.3	95.5	96.7	95.8
0.4	93.1	93.2	93.0
0.5	93.4	92.7	92.5
0.6	92.5	91.7	91.8

改进 MOPSO 算法的全局搜索能力、参数迁移阈值 Z 和迁移概率 p_1 起着决定性作用, 其值设置得过大或者过小对提高算法性能都是不利的. 这是因为, 若 Z 设置得过大, 则改进 MOPSO 算法近似于基本 MOPSO; 而 Z 设置得过小, 则粒子的迁移起不到缩减搜索范围的作用. 同样, 若迁移概率设置过小, 则微粒不易跳出局部最优点; 而迁移概率设置越大, 则对算法已找到的最优解破坏的概率会越大. 实验表明, $Z = 0.8v_{max}, p_1 = 0.5$ 时, IN-MOPSO 可以得到最好的寻优成功率.

图 4 展示了种群大小与算法结果之间的变化曲线, 其中算法结果取值为种群对应的目标函数范数值.

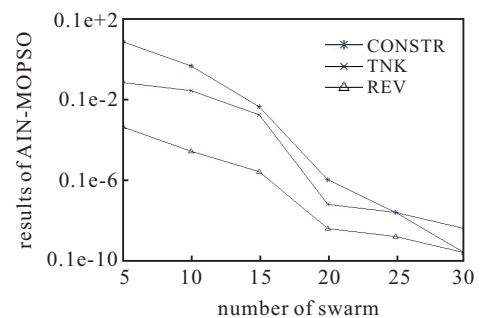


图 4 种群个数与算法范数平均结果之间的关系

由图 4 可以看出, IN-MOPSO 的种群个数 Pop 在 [5, 50] 之间取值时, Pop 值越大, 算法所得到的范数值越好; 当 $Pop \geq 20$ 时, 采用 IN-MOPSO 算法优化的 3

个测试函数均能够满足规定阈值的满意解. 因此, 取 $Pop = 20$ 比较合适.

4 供电曲线多目标优化模型

供电曲线是在一个冶炼炉次中向电炉内供电的功率、电压、电流与时间的关系曲线. 制定合理的交流电弧炉供电曲线的目的是在电弧炉安全稳定运行的前提下, 保证冶炼节奏快、生产成本低. 以一炉钢液为研究对象, 在金属原料成本一定的前提下, 进一步优化电抗投入量、工作电压和工作电流可达到降低成本的目的. 因此, 综合考虑各类因素, 从节省成本的角度出发, 本文确定以降低吨钢电耗、缩短冶炼时间、减小电极消耗、增加炉衬使用寿命为目标进行研究. 这是一个多目标优化问题, 可归结为如下数学模型:

$$\begin{aligned} \min b &= \frac{K P_{\text{arc}} / \eta \cos \varphi + P_{\text{arc}} P_g T_s / M}{\eta (P_{\text{arc}} - P_g)}, \\ \min T &= \frac{K M / \eta \cos \varphi + P_g T_s}{P_{\text{arc}} - P_g}, \\ \min C &= 0.02 \times I^2 \times \frac{K M / \eta \cos \varphi + P_g T_s}{(P_{\text{arc}} - P_g) \eta} + \\ &A \times R \times \left(\frac{K M / \eta \cos \varphi + P_{\text{arc}} T_s}{P_{\text{arc}} - P_g} \right) / M, \\ \max n &= \frac{D}{L} = \frac{D 4 \pi d^2}{k \cos \varphi \int_0^T f(E) P_{\text{arc}} dt}. \end{aligned} \quad (13)$$

其中: b 为吨钢电耗, T 为冶炼时间, C 为电极消耗, n 为炉衬寿命, M 为出钢量, P_{arc} 为电弧电压, η 为电效率, $\cos \varphi$ 为功率因数, T_s 为热停工时间, D 为炉衬厚度, L 为炉衬消耗, d 为电极到炉壁的厚度; $K, M, P_g, T_s, D, d, f(E)$ 由工艺条件决定, 当工艺条件一定时, 可认为是常数.

供电模型的约束条件包括以下几点:

1) 变压器约束. 电炉冶炼中, 变压器输出功率、电弧电流不能超过变压器的允许值(一般电炉变压器允许长期超载 20% 运行), 变压器的电压档位直接决定供电电压, 即

$$S = 3UI < S_e^i, I \leq I_e^i, U \in \{U_1, U_2, \dots, U_{S_t}\}. \quad (14)$$

其中: S_e^i 为变压器第 i 档位容量的 1.2 倍; I_e^i 为变压器第 i 档二次额定电流的 1.2 倍; U_i 为变压器各档位电压; $i = 1, 2, \dots, S_t, S_t$ 为变压器档位数.

2) 电抗器约束

$$x \in \{x^1, x^2, \dots, x^{S_r}\}. \quad (15)$$

其中: $x^j (j = 1, 2, \dots, S_r)$ 为电抗器的各档位感抗; S_r 为电抗器档位数.

3) 电弧稳定燃烧约束. 为使电弧稳定燃烧, 电弧炉的功率因数应小于等于 0.866, 即

$$0.75 \leq \cos \varphi \leq 0.866. \quad (16)$$

4) 耐火材料烧损系数. 电弧燃烧对炉盖、炉衬有

一定烧损. 为了提高炉盖、炉衬耐火材料的使用寿命, 要求电弧的燃烧值 C_{ABI} 满足

$$C_{\text{ABI}} = \frac{P_{\text{arc}} U_{\text{arc}}}{d^2} \leq R_S. \quad (17)$$

考虑整个炼钢过程电弧对炉盖及炉衬的烧损, 耐火材料烧损系数约束在 300 ~ 450 之间, 即

$$R_S \in [300, 450]. \quad (18)$$

5) 文献 [16] 认为运行电抗与短路电流存在以下关系:

$$\frac{X_{\text{op}}}{X_{\text{sc}}} = \frac{\alpha}{(I/I_{\text{sc}})^2} + 1 - \alpha. \quad (19)$$

将优化目标 (13) 及约束条件 (14) ~ (19) 整理, 并考虑运行电抗, 统一各个目标极值, 则可得到如下优化供电模型数学表达式:

$$\begin{aligned} \max & \\ b' &= \frac{3I(\sqrt{U^2 - (IX_{\text{op}})^2} - P_g)(\sqrt{U^2 - (IX_{\text{op}})^2} - Ir)}{[3UIK + 3I(\sqrt{U^2 - (IX_{\text{op}})^2} - Ir)]\sqrt{U^2 - (IX_{\text{op}})^2}}, \\ T' &= \frac{3I(\sqrt{U^2 - (IX_{\text{op}})^2} - Ir)^2 - (\sqrt{U^2 - (IX_{\text{op}})^2} - Ir)P_g}{U + \sqrt{U^2 - (IX_{\text{op}})^2} - Ir}, \\ C' &= \frac{3I(\sqrt{U^2 - (IX_{\text{op}})^2} - Ir)}{(0.02I^2 + Sv)([3UIK + P_g](\sqrt{U^2 - (IX_{\text{op}})^2} - Ir)M)} \rightarrow \\ &\leftarrow \frac{P_g(\sqrt{U^2 - (IX_{\text{op}})^2} - Ir)M}{3I(\sqrt{U^2 - (IX_{\text{op}})^2} - Ir)]\sqrt{U^2 - (IX_{\text{op}})^2}}, \\ n &= \frac{D 4 \pi d^2 U}{3Ik(\sqrt{U^2 - (IX_{\text{op}})^2}) \int_0^T f(E)(\sqrt{U^2 - (IX_{\text{op}})^2} - Ir) dt}. \end{aligned} \quad (20)$$

s.t.

$$IX_{\text{op}} \geq 0.366U,$$

$$I[\sqrt{U^2 - (IX_{\text{op}})^2} - 0.0007I]^2 \leq 2.25R_S,$$

$$X_{\text{op}} = \left[0.00184 + \frac{xU^2}{35000^2} \right] \left(\frac{\alpha I_{\text{sc}}^2}{I^2} + 1 - \alpha \right),$$

$$300 \leq R_S \leq 450,$$

$$0 \leq I \leq I_e^i,$$

$$U \in \{U_1, U_2, \dots, U_{S_t}\},$$

$$x \in \{x^1, x^2, \dots, x^{S_r}\}.$$

其中: I 为工作电流, U 为变压器二次侧电压, X_{op} 为运行电抗, K 为电炉工艺热耗, r 为电路电阻, I_{sc} 为短路电流, α 为运行电抗系数, k 为耐火材料系数, P_g 为炉子的固定热损失.

5 应用实例

以某厂 40 吨 SR 电弧炉为例, 电弧炉的短网电阻

为 0.7 mΩ; 电弧炉短网感抗为 1.84 mΩ; 短路电阻为 0.455 mΩ; 变压器一次侧额定电压为 35 kV, 二次侧额定电压为 570 ~ 350 V, 每档间隔 20 V, 共 12 档, 前 8 档容量恒定为 25 MVA, 后 4 档二次侧额定电流均为 33.565 kA; 附加电抗器分为 5 档, 每档的电抗分别为 11.3 Ω, 10.1 Ω, 8.2 Ω, 6.2 Ω, 0 Ω.

结合炼钢过程中的点弧、穿井、主熔化期和熔化末期 4 个阶段, 利用本文提出的 IN-MOPSO 算法求解上述供电优化问题, 得出相应每个阶段的供电策略及相应的电弧炉工作参数. 优化结果见表 2, 由优化算法得出的供电曲线如图 5 所示.

表 2 电弧炉炼钢供电策略

阶段	变压器档位	电抗器档位	工作电流	烧损系数
点弧	6	1	32.17	321
穿井	2	1	41.68	429
主熔	1	2	46.12	481
末期	5	2	29.31	407

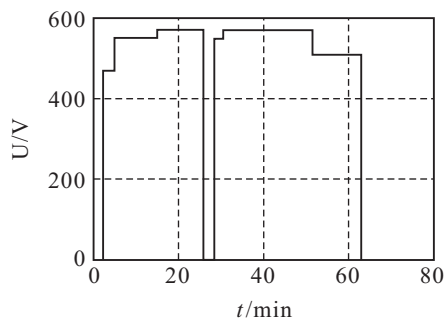


图 5 电弧炉炼钢供电曲线

注 1 图 5 中第 25 ~ 29 min 处为第 2 次加入废钢所需时间.

采用上述供电策略指导电弧炉炼钢过程, 并结合图 5 和表 2 可以得出以下结论:

- 1) 点弧阶段由于电弧不稳定且弧光靠近炉盖, 采用低电压、较小电流方式可以保护炉盖;
- 2) 穿井阶段电弧不稳定, 应采用较高电压、较大电流方式;
- 3) 对于主熔阶段, 由表 2 可以看出此阶段的烧损系数为 481, 远远超出了约束中给定的范围, 这主要是因为主熔阶段电极被废钢铁料包围, 对炉盖和炉衬的烧损可以忽略不计, 因此应采用高电压、大电流方式加快炼钢速度, 以获得最短的冶炼通电时间;
- 4) 末期应采用较低电压、小电流的方式保护整个炉衬.

为测试所提出的改进算法的性能, 本文进行以下比较: 选取电弧炉炼钢过程中的主熔阶段为对象, 利用 NSGA-II 和 MOPSO 算法求解式 (20) 所示的多目标优化问题. 两种算法的初始种群数和最大迭代次数与 IN-MOPSO 相同, 优化结果如表 3 所示.

由表 3 可以看出, 由 IN-MOPSO 算法计算求得的

表 3 主熔阶段对比结果

算法	工作电流	电弧功率	功率因数	烧损系数
IN-MOPSO	46.12	24.76	0.84	481
MOPSO	49.25	25.37	0.79	543
NSGA-II	45.27	23.76	0.81	509

电流处于 MOPSO 算法与 NSGA-II 算法计算得到的电流之间, 这样即可以避免由于电流过大而造成的电弧不稳定、降低电弧功率、增加电耗和烧损系数的问题, 又可以避免由于电流过小而延长冶炼时间等问题. 从这种意义上看, 由本文提出的 IN-MOPSO 算法所得到的结果要优于其他算法.

在实际应用过程中, 采用本文优化的供电策略对该炉原始数据、同一钢种的 410-0172 ~ 410-0191 进行了对比实验. 由实验结果发现, 平均每炉冶炼时间提高了 10.4%, 吨钢电耗减少了 6.7%, 炉衬的使用寿命提高了 18.1%, 同时降低了电极消耗, 延长了使用寿命, 综合经济效益得到了大幅度提高.

6 结 论

本文结合 MOPSO 算法的快速寻优能力及 AIN 算法的全局和局部开发能力, 提出了基于免疫网络的改进多目标粒子群优化算法, 即 IN-MOPSO. IN-MOPSO 作为一种新的混合策略, 在种群上并行运行改进的多目标微粒群算法、改进的自适应变异算子和改进的聚类免疫网络算法, 达到了算法探索与开发能力的有效均衡. 结合电弧炉多目标供电优化模型, 应用 IN-MOPSO 算法进行求解的结果表明, 不但降低了电炉的吨钢消耗, 缩短了冶炼时间, 降低了电极消耗, 而且延长了炉衬使用寿命, 综合经济效益得到了大幅度提高.

参考文献(References)

- [1] 袁平. 电弧炉冶炼过程行进控制方法的研究与应用[D]. 沈阳: 东北大学信息科学与工程学院, 2006.
(Yuan P. Research and application of advanced control technology in the smelting process of EAF[D]. Shenyang: College of Information Science and Engineering, Northeastern University, 2006.)
- [2] 刘波, 王凌, 金以慧, 等. 微粒群优化算法研究进展[J]. 化工自动化及仪表, 2005, 32(3): 1-6.
(Liu B, Wang L, Jin Y H, et al. Advances in particle swarm optimization algorithm[J]. Control and Instruments in Chemical Industry, 2005, 32(3): 1-6.)
- [3] Li X D. A nondominated sorting particle swarm optimizer for multiobjective optimization[Z]. Lecture Notes in Computer Science, 2003, 2723: 37-48.
- [4] Alvarez-benitez J E, Everson R. A MOPSO algorithm based exclusively on Pareto dominance concepts[Z]. Lecture Notes in Computer Science, 2005, 3410: 459-473.

- [5] Laumanns M, Thiele L, Deb K, et al. Combining convergence and diversity in evolutionary multi-objective optimization[J]. *Evolutionary Computation*, 2002, 10(3): 263-282.
- [6] Mostaghim S, Teich J. The role of ε -dominance in multi-objective particle swarm optimization methods[C]. *Proc of IEEE Swarm Intelligence Symposium*. Canberra, 2003: 1764-1771.
- [7] Sierra M R, Coello C A C. Improving PSO-based multiobjective optimization using crowding mutation and ε dominance[C]. *Int Conf on Evolutionary Multi-criterion Optimization*. Guanajuato, 2005: 505-519.
- [8] Coello C A C, Pulido G T, Lechuga M S. Handling multiple objectives with particle swarm optimization [J]. *IEEE Trans on Evolutionary Computation*, 2004, 8(3): 256-279.
- [9] Coello C A C, Lechuga M. MOPSO: A proposal for multiple objective particle swarm optimization[C]. *Proc of IEEE Congress on Evolutionary Computation*. Hawaii, 2002: 1051-1056.
- [10] 贾东立, 张家树. 基于混沌变异的小生境粒子群算法[J]. *控制与决策*, 2007, 22(1): 117-120.
(Jia D L, Zhang J S. Niche particle swarm optimization combined with chaotic mutation[J]. *Control and Decision*, 2007, 22(1): 117-120.)
- [11] 刘丽. 人工免疫网络研究及应用[D]. 无锡: 江南大学物联网工程学院, 2008.
(Liu L. *Research of the Principles and Applications of Artificial Immune Network*[D]. Wuxi: Internet of Things, Jiangnan University, 2008.)
- [12] Xia Y S, Feng G, Wang J. A novel recurrent neural network for solving nonlinear optimization problems with inequality constraints[J]. *IEEE Trans on Neural Networks*, 2008, 19(8): 1340-1353.
- [13] Amirjanov A. The development of a changing range genetic algorithm[J]. *Computer Methods in Mechanics and Engineering*, 2006, 195(19/20/21/22): 2495-2508.
- [14] 赫然, 王永吉, 王青, 等. 一种改进的自适应逃逸微粒群算法及实验分析[J]. *软件学报*, 2005, 16(12): 2036-2044.
(He R, Wang Y J, Wang Q, et al. An improved particle swarm optimization based on self-adaptive escape velocity[J]. *J of Software*, 2005, 16(12): 2036-2044.)
- [15] Deb K, Pratap A, Agarwal S, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II[J]. *IEEE Trans on Evolutionary Computation*, 2002, 6(2): 182-197.
- [16] Chiba K, Ohsaka T, Oyama N. Electrode kinetics of electroactive electropolymerized polymers deposited on graphite electrode surfaces[J]. *J of Electroanalytical Chemistry and Interfacial Electrochemistry*, 1987, 217(2): 239-251.

(上接第1312页)

- [9] Leski J M. TSK-Fuzzy modeling based on ε -insensitive learning[J]. *IEEE Trans on Fuzzy Systems*, 2005, 13(2): 181-193.
- [10] Vapnik V. *Statistical learning theory*[M]. New York: John Wiley, 1998: 181-216.
- [11] Huang G B, Babri H A. Upper bounds on the number of hidden neurons in feedforward networks with arbitrary bounded nonlinear activation functions[J]. *IEEE Trans on Neural Networks*, 1998, 9(1): 224-229.
- [12] Huang G B, Zhu Q Y, Siew C K. Extreme learning machine: A new learning scheme of feed-forward neural networks[C]. *Proc of Int Joint Conf on Neural Networks*. New York: IEEE Press, 2004: 985-990.
- [13] Chung F L, Wang S T, Deng Z H, et al. CATSMLP: Toward a robust and interpretable multilayer perceptron with sigmoid activation functions[J]. *IEEE Trans on Systems, Man, and Cybernetics-part B: Cybernetics*, 2006, 36(6): 1319-1331.
- [14] Lo S P, Lin Y Y. The prediction of wafer surface non-uniformity using FEM and ANFIS in the chemical mechanical polishing process[J]. *J of Materials Processing Technology*, 2005, 168(2): 250-257.