文章编号: 1000−8152(2009)09−1014−05

具有转换函数的均匀差分进化算法及性能分析

赵云涛, 王 京, 宋 勇, 凌 智

(北京科技大学 高效轧制国家工程研究中心, 北京 100083)

摘要: 对于求解复杂优化问题, 差分进化算法存在后期收敛缓慢、易于陷入局部最优等缺点. 为此, 从充分利用求解信息和目标信息角度提出了具有转换函数的均匀差分进化算法. 首先对3个算子进行分布均匀性分析及设计, 使其生成的个体能完全表征解空间特征, 并增强种群多样性. 其次, 为简化优化环境, 利用一种适应度转换函数使得当前局部极小点及相关区域拉伸一定高度而优于当前极小点的函数部分保持数值不变. 最后通过性能指标的定量评价, 结果验证了改进算法在有效性、鲁棒性和效率上的优异性能.

关键词: 差分进化算法; 均匀设计; 适应度转换函数; 函数优化

中图分类号: TP301 **文献标识码**: A

Uniform differential evolution algorithm with transform function and performance analysis

ZHAO Yun-tao, WANG Jing, SONG Yong, LING Zhi

(National Engineer Research Center of Advanced Rolling, University of Science and Technology Beijing, Beijing 100083, China)

Abstract: When differential evolution algorithm is applied in complicated optimization problems, it has the shortages of prematurity and stagnation. By efficiently utilizing the information of objective function and solving problems, a uniform differential evolution algorithm with transform function is proposed in this paper. Firstly, three operators are designed to generate individuals which obey uniform distribution. Individuals can fully represent the solution space. So the diversity of populations and capability of global search will be enhanced. Secondly, a transform function used to simplify the objective function is constructed. It stretches the current local minimum and related regions up to a certain height, while keeps the optimized function unchanged under the local minimum. Thus, the number of local minima will be largely decreased with the progress of iterations. Finally, the improved algorithm is quantitatively evaluated by performance indices. The simulation results show that it has perfect property in efficacy and converges faster, and is more stable.

Key words: differential evolution algorithm; uniform design; transform function; function optimization

1 引言(Introduction)

由Storn R和Price K于1995年提出的差分进化算法(differential evolution, DE)[1]是一种随机的群体演化搜索算法, 其通过个体合作与竞争所产生的群体智能来指导优化过程. 由于其易用性、稳健性和强大的全局寻优能力, 目前已在多个领域取得成功[2~4]. 但在具有多峰, 噪声等复杂的优化环境下, 依然存在后期收敛速度缓慢, 易于陷入局部极值点的缺点. 为避免出现这些问题, 前人在改善算法参数[2], 算子改进[4]和与其他优化算法相结合形成混合算法[5]等方面做了相关改进研究.

本文从充分利用求解信息和目标信息角度即将个体有效映射解空间特征和简化复杂优化环境相结合对算法进行改进. 首先依据文献[6]提出的均匀分布能完全表征解空间特征并有利于增强种群多样性, 对差分进化算子进行均匀性研究. 通过分析, 经变异和选择操作后均可得到符合均匀分布个体, 而对不能产生均匀分布特征的交叉算子则根据数论一致分布点集理论进行改进, 得到具有均匀分布性的改进交叉算子. 其次, 采用转换函数对原函数变换, 使当前局部极小点及劣于此点的区域得以拉伸一定高度并保持被优化函数值在当前局部极小点以下部分的数值不变, 这样迭代中可大量消除局部极小点数量从而有效降低了后续搜索难度. 最终得到可快

收稿日期: 2008−07−03; 收修改稿日期: 2008−12−19.

速求解复杂优化问题的具有转换函数的均匀差分进化算法(简记为AtDE).

最后采用“平均截止代数”和“平均优化性能”两个性能指标对算法量化评估, 仿真表明, 在求解复杂优化问题上AtDE算法收敛性、鲁棒性和寻优效率等方面都得到了显著提高.

2 具有转换函数的均匀差分进化算法(Uniform differential evolution algorithm with transform function)

连续参数优化模型定义如下: $\min f: S \to \mathbb{R}$, 其中, f为适应值函数, $S = \prod\limits_{i=1}^{n}[x_i^D, x_i^U]$为搜索空间也即解空间, 这里$[x_i^D < x_i^U]$, $i = 1, 2, \cdots, n$, 每个变量信息可以用M维向量表示$x_i = (x_{i1}, x_{i2}, \cdots x_{iM})^{\mathrm{T}}$, $\mathbb{R}$为值域空间.

在DE算法中, 将待优化变量可作为算法个体直接处理, 设$x_i^g (i = 1, 2, \cdots, n)$是第$g$代的$n$个个体, 这些个体组成了第$g$代的种群$x^g$.

2.1 算子均匀性研究(Study of uniformity)

算法初始化后, 所有个体在解空间内均符合均匀概率分布, 通过一系列变异(记为M_g)、交叉(记为C_g)、选择(记为S_g)操作最终完成种群进化. 因搜索过程中保持均匀分布的群体能充分表征解空间特征, 有利于增强种群多样性和阻止后期算法搜索停滞. 下面对各算子做具体研究, 不失一般性, 令个体取值范围均映射到$(0, 1)$.

1) 变异算子.

引理 1 设$x_1, \cdots, x_n$是区间$(0, 1)$上独立随机变量, $c_1, \cdots, c_n$是常数, 又设$X = \sum\limits_{i=1}^{n} c_i \cdot x_i$, 若$x_i \sim U(0, 1)$且$c_1$是整数, 则$X \sim U(0, 1)$[7].

定理 1 差分进化算法经变异操作M_g后生成的中间种群服从均匀分布.

证 对于DE算法中某个待变异的目标个体x_i^g, 经变异操作后生成中间个体v_i^g, 记为

$$v_i^g = M_g(x_i^g) = x_{r1}^g + F(x_{r2}^g - x_{r3}^g). \tag{1}$$

其中: $r1, r2, r3$为在种群中随机选择的3个互不相同个体, 且$r1, r2, r3$与目标个体序号i不同, $F \in [0, 2]$是一个实常数因子.

因$x_{r1}^g, x_{r2}^g, x_{r3}^g \sim U(0, 1)$, $c_1 = 1, c_2 = -c_3 = F$. 根据引理1, 经$x_{r1}^g, x_{r2}^g, x_{r3}^g$线性组合得到的中间变量$v_i^g$服从均匀分布, 即$v_i^g \sim U(0, 1)$.

定理1证明经变异操作个体依然服从均匀分布.

2) 交叉算子.

交叉算子用于个体搜索空间的扩展, 其通过原目标个体x_i^g与变异后中间个体v_i^g分量的随机交换得到中间个体u_i^g, 具体计算形式见公式(2):

$$\begin{aligned} & u_i^g = C_g(x_i^g \otimes v_i^g) = \\ & \begin{cases} v_{ij}^g, \text{如果 rand} \leqslant CR \text{ 或 } j = mbr(i); \\ x_{ij}^g, \text{如果 rand} > CR \text{ 或 } j \neq mbr(i), \end{cases} \\ & u_i^g = (u_{i1}^g, u_{i2}^g, \cdots, u_{iM}^g). \end{aligned} \tag{2}$$

式中: $\otimes$代表交叉操作; rand为$[0, 1)$间随机数; $mbr\,(i) \in 1, 2, \cdots, M$; $CR \in [0, 1]$为交叉算子.

这种采用分量与分量交换的随机生成方式, 很难得到均匀分布个体而且搜索范围相当有限, 可能会忽略临域中更好的点. 根据数论网格定义, 采用网格法产生的点集能均匀地分布于搜索空间[8], 它的搜索效果要比上述随机交叉法好, 因此可采用数论网格理论改进交叉算子.

定义 1 命C^M为M维单位立方体, 对于某点集$H_1 = \{(b_{i1}, b_{i2}, \cdots, b_{iM}), i = 1, \cdots, n\}$, $\forall h = (\tilde{b}_1, \tilde{b}_2, \cdots, \tilde{b}_M)$, 满足$0 \leqslant b_{ij} \leqslant \tilde{b}_j$点的个数记为$N(H_1, h)$, $j = 1, 2, \cdots, M$.

$$\phi(n) = \sup_{h \in C^M} \left| \frac{N(H_1, h)}{n} - \mu(h) \right|,$$

其中: $\mu(h) = \tilde{b}_1\ \tilde{b}_2\ \cdots\ \tilde{b}_M$, 称点集$H_1$有偏差$\phi(n)$.若$\phi(n) = O\left(n^{-(1+\varepsilon)}\right)$, $\forall \varepsilon > 0$, 则H_1为一致分布点集[8].

定义 2 形如$H_1 = \{(\tilde{b}_1 \times i, \cdots, \tilde{b}_M \times i), i = 1, 2, \cdots, n\}$的点集, 采用分圆域方法取$\tilde{b}_j = \mathrm{e}^j, 1 \leqslant j \leqslant M$, 则$H_1$满足上述一致分布点集定义[8].

对于DE中两待交叉个体$x_i^g = (x_{i1}^g, x_{i2}^g, \cdots, x_{iM}^g)$, $v_i^g = (v_{i1}^g, v_{i2}^g, \cdots, v_{iM}^g)$, 命$J$为两个体不同分量集合即$J = \left\{j | x_{ij}^g \neq v_{ij}^g, 1 \leqslant j \leqslant M\right\}$[9]. 设集合$H_2 = \left\{(a_1, a_2, \cdots, a_n) | j \in J, a_j = *; j \notin J, a_j = x_{ij}^g\right\}$, 由于$x_i^g$和$v_i^g$交叉后的个体必然位于集合$H_2$, 则可用定义的一致分布点集得到优良个体.

定理 2 采用上述定义的一致分布点集得到的交叉个体符合均匀分布.

证 设x_i^g和v_i^g有r个分量不相等, 将其集合记为$H_3 = \{(b_1, b_2, \cdots, b_r)\}$. 根据定义, 一致分布点集$H_1 = \{(\tilde{b}_1 \times i, \cdots, \tilde{b}_r \times i), i = 1, \cdots, n\}$, 且$\tilde{b}_j = \mathrm{e}^j, 1 \leqslant j \leqslant r$.

因一致分布点集中各点均符合均匀分布, 可任取一点替代H_3. 则经交叉操作后得到个体可取为

$$u_i^g = C_g(x_i^g \otimes v_i^g) = (u_{i1}^g, u_{i2}^g, \cdots, u_{iM}^g),$$

其中: 当$j \notin J$, $u_{ij}^g = a_j$; 当$j \in J$则$\{b_1\}, \cdots, \{b_r\}$被依次填充到u_{ij}^g, $\{b_j\}$表示b_j的小数部分. 因

此经改进交叉算子得到的中间个体符合均匀分布, 定理得证.

3) 选择算子.

DE算法根据贪婪准则选择下一代成员, 见公式(3):

$$x_i^{g+1}=S_g(x_i^g,u_i^g)=\begin{cases}x_i^g, 如果\ f(u_i^g)>f(x_i^g);\\ u_i^g, 其他.\end{cases} \quad (3)$$

基于贪婪准则的选择算子不能够产生新的个体, 因此经选择操作后种群依然服从均匀分布.

2.2 适应度转换函数(Transform function)

算法在运行过程中, 不断检测搜索到的极值信息. 如果任两次(可根据测试函数的复杂程度灵活设定) 迭代后, 极值信息$f(x^b)$都未改变, 则认为算法陷入局部极小点, 并根据此信息将解空间划分为$S_1=\{x|f(x)\geqslant f(x^b)\}$和$S_2=\{x|f(x)<f(x^b)\}$. 因此, 可采用一种转换函数来简化复杂优化环境, 消除比局部极值点差的搜索区域, 而优于此局部极值点的函数部分保持不变. 从而使算法更容易摆脱优化问题的局部极值点, 并与种群个体均匀性分布协同作用, 保证算法能以较大机率收敛于问题的全局最优解.

本文设计了如下适应度转换函数公式:

$$\varphi(x)=\frac{(M(f(x)-f(x^b))+1)}{2}\cdot (f(x^*)-f(x))+f(x). \quad (4)$$

其中$f(x^*)$为初始化时某点的适应度函数.

$$M(x)=-\mathrm{sgn}^2(x)+\mathrm{sgn}(x)+1. \quad (5)$$

$\mathrm{sgn}(\cdot)$为符号函数, 此转换函数具有如下性质:

1) 若算法找到新点x使得$f(x)\geqslant f(x^b)$, 则$M(f(x)-f(x^b))=1$, 即拉伸所有劣于$f(x^b)$的点到$f(x^*)$, 使其位于“切平面”上.

2) 对$f(x)<f(x^b)$部分, $M(f(x)-f(x^b))=-1$, 则$\varphi(x)=f(x)$. 即函数$f(x)$中所有优于$f(x^b)$的部分保持不变.

根据上述性质可知, 定义的转换函数满足设计要求. 因为转换函数的使用并没有改变搜索目标, 只是有效降低后续搜索过程难度. 通过转换操作, 所有聚集在局部最优点x^b周围个体将提升到某一“切平面”上. 随着算法运行, 这些个体很容易掉入镶嵌在“切平面”上的凹陷区域, 即$S_2=\{x|f(x)<f(x^b)\}$. 一旦算法又一次陷入局部极小, 则再一次施行变换, 直至找到全局最优点.

2.3 算法步骤(Steps of algorithm)

根据前两小节, AtDE算法具体步骤为

Step 1 置$g=0$, 产生初始种群;

Step 2 计算种群每个个体适应度;

Step 3 评估最优个体是否位于局部极小点, 位于则令$f(x)=\varphi(x)$; 否则转Step 4;

Step 4 如果满足停机条件, 停机; 否则转Step 5;

Step 5 执行变异操作M_g得到中间种群v^g;

Step 6 对原种群x^g及中间种群v^g采用均匀交叉算子C_g, 得到中间新种群u^g;

Step 7 通过选择算子$S_g(x^g,u^g)$得到下一代新种群x^{g+1};

Step 8 置$g=g+1$, 并转Step 2.

3 评价指标(Performance indices)

为评估改进算法的优化性能, 本文通过两个评价指标分别从效率和收敛性能方面定量分析.

1) 截止代数[10].

$f(x)$在可行域内首次达到计算精度ε(ε为一很小正数)时的进化代数称为截止代数. 如果在达到预设定的最大进化代数$G_{\max}$而仍未达到计算精度ε, 则截止代数为$G_{\max}$.

2) 平均截止代数[10].

设AsG为达计算精度ε的平均截止代数, SG_i为第i次独立运行时的截止代数, R_N为独立运行次数, 则集合

$$SG=\{SG_i|0<SG_i\leqslant G_{\max},i=1,\cdots,R_N\},$$

将SG中不同元素按从小到大顺序排列, 得到新集合

$$SG'=\{SG_i'|SG_i'\leqslant SG_{i+1}',i=1,\cdots,M-1,\ M\leqslant R_N\},$$

$$C=\{C_i|0<C_i\leqslant R_N,i=1,2,\cdots,M\},$$

其中C_i 为SG_i' 对应的统计频数, 由C可得SG_i'对应的统计频率p_i 构成的集合

$$P=\{p_i|p_i=\frac{c_i}{R_N},\sum_{i=1}^{M}p_i=1,\ 0\leqslant p_i\leqslant 1,i=1,\cdots,M\},$$

则平均截止代数AsG 为:

$$AsG=\sum_{i=1}^{M}SG_i'\cdot p_i. \quad (6)$$

3) 平均优化性能.

设$f_i^*(x^g)$为第i次独立执行时, 算法运行第g代得到的最佳适应度函数. 则平均优化性能AvG_g可表示为

$$AvG_g=\frac{1}{R_N}\sum_{i=1}^{R_N}f_i^*(x^g), \quad (7)$$

其中$0<g\leqslant G_{\max}$.

4 仿真试验(Simulation test)

为验证算法效率及收敛性能, 本文采用6个基准测试函数(函数1 Rosenbrock、函数2 LevyNo.5、函数3 LevyNo.8、函数4 Schaffer、函数5 Schwefel、函数6 Generalized Schwefel), 并与DE算法和实数编码遗传算法(SGA)进行比较, 其中Schwefel及Generalized Schwefel函数为高维复杂多峰值函数. 这几个函数采用普通优化方法很难取得较好优化效果, 可有效检验算法全局搜索性能. 其中, 函数4为求全局极大值, 其余函数均求极小值.

各算法独立运行30次, 对于函数1 ~ 4, 群体规模为40, 最大迭代次数取200; 对于函数5、函数6, 群体规模为150, 最大迭代次数取500. SGA算法参数参考文献[11]设置, 交叉概率$P_c = 0.9$ 并采用自适应变异概率. DE参数参考文献[12]设置, 其中

$$CR = 0.8, F = 0.5,$$

AtDE参数采用上述DE设置. 为比较各算法收敛性能, 停机条件只设最大迭代次数而未加计算精度限制, 测试结果见表1及图1.

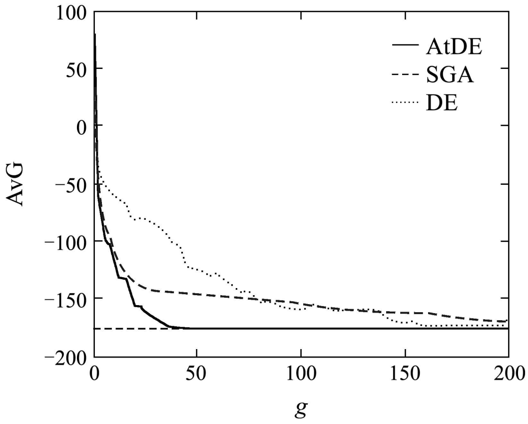


图 1　平均优化性能对比曲线

Fig. 1　Comparison of mean performance

表 1　基准测试函数的仿真结果

Table 1　Simulation results of benchmark functions

函数	理论最优	SGA		DE		AtDE	
		平均最优解	优化值方差	平均最优解	优化值方差	平均最优解	优化值方差
函数1	3905.9262	3880.5446	9.3858	3905.9262	0	3905.9262	0
函数2	−176.1375	−170.8375	0.1271	−173.0815	0.0053	−176.1375	0
函数3	0	0.0038454	0.00072	0.00014115	0.00013	0.0000725	0.00008
函数4	1	0.94813	0.00452	0.98192	0.00414	0.99928	0.00032
函数5	0	33.3693	9.0096	0.8104	0.9029	0.0121	0.1145
函数6	−12569.4866	−10065.7665	3.4061	−12167.6607	0.0538	−12569.4853	0.0021

由表1可知, SGA算法在5个具有较高寻优难度函数上的表现虽然要好于求解Rosenbrock函数, 但整体寻优性能依然劣于DE和AtDE算法. DE和AtDE算法在Rosenbrock函数上都能找到最优点, 但是在另外3个较高难度低维函数上AtDE算法表现优异, 鲁棒性也较好. 尤其在Levy No.5函数上, AtDE算法经过迭代寻优能最终找到理论最优解, 试验验证了改进算法的有效性, 如图1所示. 在高维函数Schwefel和Generalized Schwefel上, 虽然DE算法也取得了较好优化效果, 但AtDE算法由于采用了维持种群多样性和改善优化环境等措施, 优化结果仍然好于DE算法.

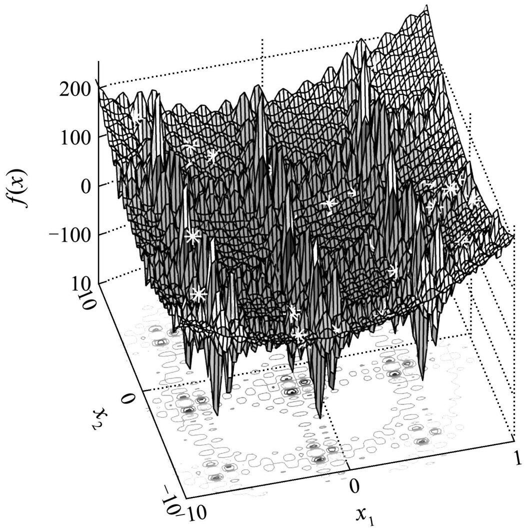
下面以Levy No.5为例, 具体研究AtDE算法寻优过程. 图2(a)描述了种群在Levy No.5原始函数上的均匀分布情况, 图2(b)显示AtDE算法在五次迭代陷入某局部极小点后, 通过对原函数转换即将局部极小点及劣于局部极小点的函数值提升到某一“切平面”, 有效降低了后续搜索难度. 随着算法运行, 最终个体掉入“切平面”的凹陷中. 其中绿色“$*$”代表个体.

实际应用中, 在保证收敛性同时更注重算法寻优效率, 3种算法求解基准函数的平均截止代数($\varepsilon = 0.001$)见表2. 实验结果表明AtDE算法优化效率有了很大改进, 明显优于SGA和DE算法.

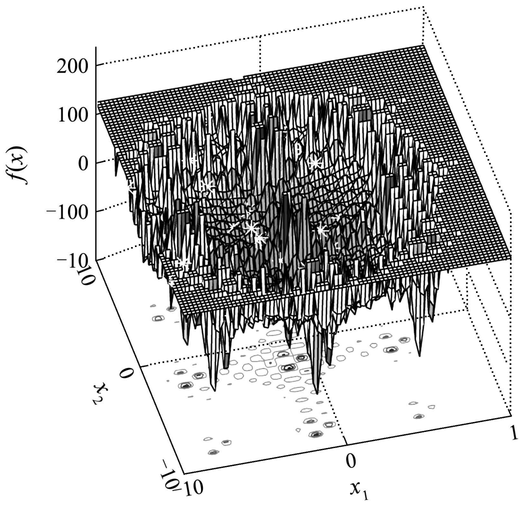
表 2　平均截至代数比较

Table 2　Comparison of the AsG

函数	SGA	DE	AtDE
Rosenbrock	200	83	42
Levy No.5	200	200	57
Levy No.8	200	145	44
Schaffer	200	200	126
Schwefel	490	451	333
Generalized Schwefel	500	475	329



(a) 在Levy No.5函数空间种群初始分布



(b) 五次迭代后在变换函数空间内种群分布

图 2 AtDE算法对Levy No.5函数优化过程

Fig. 2 Optimization process of Levy No.5 function by AtDE

5 结论(Conclusion)

以求解复杂优化问题为目标的改进算法, 其将个体均匀分布维持种群多样性和采用转换函数简化优化环境相结合.因此, 能充分利用求解信息和目标信息从而避免进化后期搜索停滞和陷入局部最优, 提高了寻优效率. 仿真结果表明, AtDE算法在收敛性能和求解效率上都表现十分优异, 具有很好的应用前景.

参考文献(References):

[1] STORN R, PRICE K. *Differential evolution-a simple and efficient adaptive scheme for global optimization over continuous spaces*[R]. America : International Computer Science Institute, 1995.

[2] 颜学峰, 余娟, 钱锋. 自适应变异差分进化算法估计软测量参数[J]. 控制理论与应用, 2006, 23(5): 744 – 748.
(YAN Xuefeng, YU Juan, QIAN Feng. Adaptive mutation differenttial evolution algorithm and its application to estimate soft sensor parameters[J]. *Control Theory & Applications*, 2006, 23(5): 744 – 748.
)

[3] URSEM R K, VADSTRUP P. Parameter identification of induction motors using differential evolution[C] *//Proceeding of the Fifth Congress on Evolutionary Computation Conference*. Canberra: [s.n.], 2003: 790 – 796.

[4] CHIOU J P, WANG F S. A hybrid method of differential evolution with application to optimal control problems of a bioprocess system[C] *//Proceeding of IEEE International Conference on Evolutionary Computation*. New York: IEEE, 1998: 627 – 632.

[5] 吴燕玲, 卢建刚, 孙优贤. 基于免疫原理的差分进化[J]. 控制与决策, 2007, 22(11): 1309 – 1312.
(WU Yanling, LU Jiangang, SUN Youxian. Differential evolution based on immunity[J]. *Control and Decision*, 2007, 22(11): 1309 – 1312.)

[6] 蔡晓芬, 钟守楠. 演化均匀优化算法[J]. 数学杂志, 2005, 25(3): 349 – 354.
(CAI Xiaofen, ZHONG Shounan. Evolutionary uniform optimization algorithm[J]. *Journal of Mathematics*, 2005, 25(3): 349 – 354.)

[7] DENG L Y, GEORGE E O. Generation of uniform variate from several nearly uniformly distributed variables[J]. *Communications in Statistics: Simulation and Computation*, 1990, 19(1): 145 – 154.

[8] 王元, 方开泰. 数论方法在统计中的应用[M]. 北京: 科学出版社, 1996.
(WANG Yuan, FANG Kaitai. *Application of the Number Theory in Statistics*[M]. Beijing: Science Press, 1996.)

[9] 汪琍, 张铃. 用网格实现交叉操作的遗传算法[J]. 计算机工程与科学, 2000, 22(1): 18 – 20.
(WANG Li, ZHANG Ling. The genetic algorithm with crossover operator employing lattice methods[J]. *Computer Engineering and Science*, 2000, 22(1): 18 – 20.)

[10] 孙瑞祥, 屈梁生. 遗传算法优化效率的定量评价[J]. 自动化学报, 2000, 26(4): 552 – 556.
(SUN Ruixiang, QU Liangsheng. Quantitative evaluation of optimization efficiency for genetic algorithms[J]. *Acta Automatica Sinica*, 2000, 26(4): 552 – 556.)

[11] 刘金琨. 先进PID控制MATLAB仿真[M]. 北京: 北京：电子工业出版社, 2005: 215 – 216.

[12] STORN R. On the usage of differential evolution for function optimization[C] *//Proceeding of the Biennial Conference of the North American Fuzzy Information Processing Society*. New York: IEEE, 1996: 519 – 523.

作者简介:

赵云涛 (1982—), 男, 博士研究生, 目前研究方向为群智能计算、优化决策, E-mail: zyt1013@126.com;

王 京 (1948—), 男, 教授, 博士生导师, 目前研究方向为先进控制理论研究及冶金中应用, E-mail: wangj@nercar.ustb. edu.cn;

宋 勇 (1974—), 男, 硕士, 讲师, 目前研究方向为轧钢自动化数学建模与控制策略研究, E-mail: songyong@nercar. ustb.edu.cn;

凌 智 (1973—), 男, 硕士, 讲师, 目前研究方向智能控制理论与应用, E-mail: lingzhi@nercar. ustb.edu.cn.