# 求解线性约束问题的微粒群优化算法

陈战平[1,2]

(1. 南京师范大学 计算机科学与技术学院, 江苏 南京 210046;
2. 江苏省信息安全保密技术工程研究中心, 江苏 南京 210097)

[摘要] 直接用微粒群算法求解约束优化问题存在收敛速度慢和精度低的缺点,研究了一种求解线性约束问题的微粒群优化算法. 通过引入拉格朗日乘子将约束优化问题转化为无约束优化,先利用拉格朗日对偶原理,将拉格朗日乘子和优化参数分离出来,然后分别采用微粒群算法进行优化. 另外,为了使微粒群算法更好地收敛到全局最优解,设计了一个突变的微粒群算法. 最后通过低通滤波器的设计证明该方法的效果优于不带约束的微粒群算法.

[关键词] 线性约束, 优化, 微粒群算法

[中图分类号] TP301.6　[文献标识码] A　[文章编号] 1672-1292(2010) 04-0026-05

## Particle Swarm Algorithm for Linear Constrained Optimization Problem

Chen Zhanping[1,2]

(1. School of Computer Science and Technology, Nanjing Normal University, Nanjing 210046, China;
2. Jiangsu Research Center of Information Security and Privacy Technology, Nanjing 210097, China)

**Abstract**: The shortage in slow convergence rate and low convergence precision exist when the particle swarm algorithm is directly used to solve the constrained optimization problem. In this paper, we are concerned with an new particle swarm algorithm, which can be used to solve the linear constrained problems. In our method, the constrained optimization problem is first translated into a non-constrained optimization one by introducing the Lagrange multipliers, and then by using the Lagrange duality principle, the Lagrange multipliers and optimization parameters are separated, which will be optimized respectively by using the particle swarm algorithm. Moreover, in order to make the particle swarm algorithm converge to the global optimization solution, an improved particle swarm algorithm with mutation is proposed. Finally, a design example of a low-pass FIR filter shows that our method is better than the particle swarm algorithm without constraints.

**Key words**: linear constraint, optimization, particle swarm algorithm

标准微粒群算法在优化过程中,种群中的最优解常常几代都无变化,这很容易产生“早熟”现象,使优化过程陷入局部最优[1,2]. 另外,在科学研究和工程实践中, 许多优化问题都带有一定的线性约束条件[3-5]. 直接使用微粒群算法求解时,为了满足约束条件,需要对初始产生种群的个体以及新产生的个体进行约束条件判断,不满足约束条件的个体就舍去并重新产生,直到满足约束为止. 这种方法在保证种群满足约束的同时,明显地影响了微粒群算法的求解速度,可能将一个最优解丢弃了. 为了使微粒群算法能更好地收敛于全局最优解,并能快速、准确的求解带线性约束的优化问题. 本文对微粒群算法进行了改进,改进后的微粒群算法在找到每代的最优解时,保留最优解,并对种群进行变异,以扩大寻优的空间,使微粒群算法能更好地收敛于全局最优解. 同时,为了更好的解决带约束线性约束的优化问题,引入拉格朗日乘子法,将线性约束的优化问题转化为无约束的优化问题,再利用对偶原理[6]能精确地解决线性问题的特性,对拉格朗日乘子和要求解的问题分别用改进的微粒群算法优化. 该方法很好的解决了一般微粒群算法不易求解的约束线性问题. 并通过对带线性约束的 FIR滤波器的优化设计,验证了该方法的有效性.

收稿日期: 2010-08-06
通讯联系人: 陈战平,讲师,研究方向: 测控系统与计算机管理信息系统的开发与应用. E-mail: czpcjx@163.com

## 1 微粒群算法的改进

微粒群算法将寻优的参数组合成群体, 再通过环境的适应度使群体中的个体向好的区域移动. 微粒群的个体 (这里称作微粒) 代表问题的一个可能解, 每个微粒具有位置和速度两个特征. 设 $D$ 维搜索空间中, 第 $i$ 个微粒位置可以表示成 $X_i = [x_{i1}, x_{i2}, \cdots, x_{iD}]$, 微粒的速度表示成 $V_i = [v_{i1}, v_{i2}, \cdots, v_{iD}]$. 微粒位置坐标对应的目标函数值即可作为该微粒的适应度, 算法通过适应度来衡量微粒的优劣. 算法首先初始化一群随机微粒, 然后通过迭代找到最优解. 在每一次迭代中, 微粒通过跟踪两个"极值"来更新自己: 一个是微粒本身所找到的最优解, 即个体极值 $P_i = [p_{i1}, p_{i2}, \cdots, p_{iD}]$; 另一个是整个微粒群目前找到的最优解, 称之为全局极值 $P_g = [p_{g1}, p_{g2}, \cdots, p_{gD}]$, 最后输出的 $P_g$ 就是算法得到的最优解. 微粒在每一次迭代中找到上述两个极值后, 微粒通过一定规则进化, 以第 $i$ 个微粒的第 $j$ 维从 $n$ 代进化到 $n+1$ 代为例:

$$v_{ij}^{n+1} = v_{ij}^{n} + c_1 r_1^n (p_{ij}^n - x_{ij}^n) + c_2 r_2^n (p_{gj}^n - x_{ij}^n), \tag{1}$$

$$\begin{cases} v_{ij}^{n+1} = v_{\max}, & \text{if } v_{ij}^{n+1} > v_{\max}, \\ v_{ij}^{n+1} = -v_{\max}, & \text{if } v_{ij}^{n+1} < -v_{\max}, \end{cases}$$
$$x_{ij}^{n+1} = x_{ij}^{n} + v_{ij}^{n+1}. \tag{2}$$

其中, 加速常数 $c_1$ 和 $c_2$ 均为正实数, 表示将每个微粒推向 $P_i$ 和 $P_g$ 的加速度的权重, 取值一般在 $[0, 2]$, $r_1$ 和 $r_2$ 是范围在 $[0, 1]$ 内取值的随机数, $v_{\max}$ 是常数, 它一般取 2.0 由进化方程 (1) 分析可知, 微粒的速度只取决于微粒自身经历的最好位置 $P_i$ 和所有微粒经历的最好位置 $P_g$. 微粒群优化算法的结构相对简单, 运行速度很快. 但是, 算法运行过程中, 如果某微粒发现一个当前最优位置, 其他微粒将迅速向其靠拢. 如果该最优位置为一局部最优点, 微粒群就无法在解空间内重新搜索, 因此, 算法陷入局部最优, 出现了所谓的早熟收敛现象. 为了使算法具有良好的全局寻优能力, 出现很多改进的算法, 其中, Shi[7] 等引入惯性权重 $w$, 其作用在于维护全局和局部搜索能力的平衡, 一般可将它设为随迭代的次数线性减小, 如由 0.9 到 0.4[7]. 这样, 进化方程 (1) 变为:

$$v_{ij}^{n+1} = w v_{ij}^{n} + c_1 r_1^n (p_{ij}^n - x_{ij}^n) + c_2 r_2^n (p_{gj}^n - x_{ij}^n). \tag{3}$$

在加了惯性权重的基础上, 当微粒 $X_i$ 经历全局最好位置时, 该微粒只能做匀速运动, 随着 $w$ 的不断减小, 该微粒几乎不变, 仍有陷入局部解的可能. 为了解决这个问题, 作者对微粒群算法进行了改进, 设计了一个突变微粒群算法. 即在当微粒经历全局最好位置时, 保存这个最好位置, 同时重新随机产生一个新的个体, 也就是说当产生了最优解时, 增加一个扰动, 扩大寻找的空间, 更有利于找到全局最优解. 改进的算法流程如下:

**Step 1** 初始化, 设定 $v_{\max}$、$c_1$、$c_2$、$w$ 和最大迭代次数 $M$ 的值, 产生原始种群并计算种群中个体的适应度 $f$, 记下 $P_i$ 和 $P_g$, 保留 $P_g$ 的值, 同时随机产生出得到 $P_g$ 的相应新个体;

**Step 2** 如果迭代次数等于 $M$ 则转 Step 5, 否则转 Step 3;

**Step 3** 种群按式 (3) 和式 (2) 进化, 计算个体适应度 $f_s$;

**Step 4** 比较 $f_s$ 与 $f$; 如果 $f_s \leqslant f$ 则修改 $P_i$, 产生新的 $P_g$, 保留 $P_g$ 的值, 同时随机产生出得到 $P_g$ 的相应新个体, 否则, 转 Step 2;

**Step 5** 输出最优个体.

用了一个非线性的测试函数对突变微粒群算法进行测试, 并与一般微粒群算法 (加入惯性权重) 比较, 设测试函数:

$$f(x) = \frac{\sin^2\sqrt{x_1^2 + x_2^2} - 0.5}{[1 + 0.001(x_1^2 + x_2^2)]^2} - 0.5, \qquad x \in [-100, 100], \tag{4}$$

该函数的全局极小点是 (0, 0), 而在距全局极小点约 3.14 范围内的隆起有无限多全局极大点, 取值为 −0.990283, 因此很容易陷入局部极小点, 由于该函数的强烈震荡性质以及它的全局最优点被局部最优点所包围的特性, 使得一般算法很难找到它的全局最优解.

实验中, 突变微粒群算法和标准微粒群算法中参数取值为: $c_1$ 与 $c_2$ 都取 0.5, $w$ 从 1.2 到 0.4 随迭代次数线性减少, $v_{\max} = 2.5$ 对上述问题进行了 50 次仿真计算, 群体规模为 20, 最大进化代数为 500 用 $F$ 表示

平均收敛代数, $R$ 表示平均收敛率, 算法结果见表1.

实验结果表明, 虽然都是随机优化算法, 但突变微粒群算法能够更好更准确地找到最优解, 标准微粒群算法易陷入局部解, 收敛不如突变微粒群算法稳定.

表 1　两种算法的比较结果

**Table 1　Comparative result between the two algorithms**

| 算法 | 误差 | 平均收敛代数 $F$ | 平均收敛率 $R$/% |
|---|---|---|---|
| 突变微粒群算法 | 0.000 01 | 102.45 | 100 |
| 标准微粒群算法 | 0.000 01 | 138.23 | 45 |

## 2　带线性约束的优化问题

标准的约束最优化问题:

目标函数为:

$$\min f(\boldsymbol{X}) = \min f(x_1, x_2, \cdots, x_n). \tag{5}$$

约束条件为:

$$\begin{cases} g_i(\boldsymbol{X}) = g_i(x_1, x_2, \cdots, x_n) \leqslant 0, & i = 1, \cdots, q, \\ h_k(\boldsymbol{X}) = g_i(x_1, x_2, \cdots, x_n) = 0, & k = 1, \cdots, l. \end{cases} \tag{6}$$

式中, $\boldsymbol{X}$ 为设计向量, $\boldsymbol{X} = (x_1, x_2, \cdots, x_n)^{\mathrm{T}}$; $x_i$ 为第 $i$ 个设计变量, $i = 1, 2, \cdots, n$; $f(\boldsymbol{X})$ 为目标函数; $g_j(\boldsymbol{X})$ 为第 $i$ 个不等式约束; $h_k(\boldsymbol{X})$ 为第 $k$ 个等式约束; $n$ 为设计变量个数; $q$ 为不等式约束的个数; $l$ 为等式约束的个数. 当(5)、(6)都是线性时, 称为线性约束, 否则称为非线性约束.

常用的求解约束优化问题的方法是引入拉格朗日乘子, 构造拉格朗日函数, 将约束优化问题转化为无约束优化问题. 通过引入拉格朗日乘子 $\lambda, \mu \geqslant 0$, 由式(5)和(6)构成的拉格朗日函数:

$$L(\boldsymbol{X}) = f(\boldsymbol{X}) + \lambda g(\boldsymbol{X}) + \mu h(\boldsymbol{X}). \tag{7}$$

这样, 由(5)和(6)式表示的约束优化问题转化为 $\min L(\boldsymbol{X})$, 这是一个无约束的优化问题. 但是, 优化的参数变为多个, 即: $\boldsymbol{X}, \lambda, \mu$, 故无法用直接微粒群算法进行优化求得最优的设计变量向量 $\boldsymbol{X}$. 而根据拉格朗日对偶原理可知, 一个给定的线性约束问题都存在一个与其相关联的另一个线性约束问题[6]. 这两个问题互为对偶, 例如, 对公式(5), 其对偶问题可表示为 $\max F(\boldsymbol{X}) = \max F(x_1, x_2, \cdots, x_n) = \beta$, $\beta \leqslant \alpha$ 时为弱对偶. 两个对偶问题可以相互转化, 从而导出新的求解途径. 且在解线性约束问题时, 利用对偶理论可以使问题更精确且简便易行[6].

为此, 本文设计了一个算法, 利用对偶性原理, 将 $\boldsymbol{X}, \lambda, \mu$ 分离出来, 分别采用突变微粒群算法进行优化. 用对偶性原理求解线性约束优化问题的微粒群算法结构如图1所示. 求解步骤如下:

(1) 随机产生一组 $\boldsymbol{X}$ 作为 $\boldsymbol{X}^c$, 通过在拉格朗日乘子的取值范围内选取适当乘子 $\lambda^c, \mu^c$, 采用微粒群算法进行优化, 使得:

$$\min L(\boldsymbol{X}, \lambda^c, \mu^c) = \alpha, \tag{8}$$

得到一个最优的 $\boldsymbol{X}$.

(2) 将 $\boldsymbol{X}$ 作为 $\boldsymbol{X}^c$, 再利用拉格朗日对偶式, 采用微粒群算法进行优化, 使得:

$$\max F(\boldsymbol{X}) = \max L(\boldsymbol{X}^c, \lambda, \mu) = \beta, \tag{9}$$

得到一个最优的拉格朗日乘子 $\lambda, \mu$.

(3) 再将最优 $\lambda, \mu$ 作为 $\lambda^c, \mu^c$, 代入式(8), 重复(1)、(2), 直到 $\boldsymbol{X}, \lambda, \mu$ 不再发生变化时停止. 便可找到满足约束的最优解.

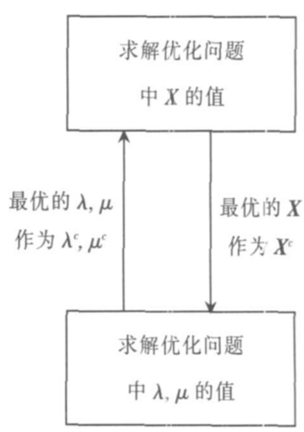


**图 1　求解结构**

**Fig.1　Solving method**

由拉格朗日对偶定理可知, 若 $\alpha$ 有限, $\exists \boldsymbol{X} \in R^n$, 使得 $f(\boldsymbol{X}) \leqslant 0$, $g(\boldsymbol{X}) \leqslant 0$, $h(\boldsymbol{X}) = 0$, 且 $\alpha = \beta$, 则(8)和(9)有最优解. 这样, 通过对偶性原理将 $\boldsymbol{X}$ 与 $\lambda, \mu$ 分离出来, 分别进行优化, 在进行若干次迭代后便完成了对线性约束问题的优化.

## 3　实例分析

FIR 滤波器在满足一定的对称条件下, 可以实现严格的线性相位. 而对 FIR 滤波器的设计是一个带有线性约束的一个最优化问题[8], 即在满足一定的约束条件下, 希望所设计的 FIR 滤波器频率特性与理想的

低通滤波器频率特性的误差最小.

设 FIR 滤波器的单位冲激响应 $h(n)$ 是有限长的 $(0 \leqslant n \leqslant N-1)$, 即滤波器的设计阶数为 $N$, 其频率响应 $H(e^{j\omega})$ 为

$$H(e^{j\omega}) = \sum_{n=0}^{N-1} h(n) e^{-j\omega n}. \tag{10}$$

当 $h(n)$ 为实序列时, 可将 $H(e^{j\omega})$ 表示成:

$$H(e^{j\omega}) = \pm |H(e^{j\omega})| = H(\omega) e^{j\theta(\omega)}, \tag{11}$$

其中, $H(\omega)$ 为幅度函数, 是一个实数, $\theta(\omega)$ 是相位函数.

线性相位 FIR 滤波器的冲激响应满足下式:

$$h(n) = \pm h(N-1-n),$$

因而系统函数可表示为:

$$H(z) = z^{-\frac{N-1}{2}} \sum_{n=0}^{N-1} h(n) \frac{z^{-n+\frac{N-1}{2}} + z^{-n-\frac{N-1}{2}}}{2}. \tag{12}$$

当 $N$ 为奇数, $h(n)$ 为偶对称, 即 $h(n) = h(N-1-n)$. 式 (11) 与式 (12) 比较可知, 幅度函数可表示为:

$$H(\omega) = \sum_{n=0}^{N-1} h(n) \cos\left[\left(\frac{N-1}{2} - n\right)\omega\right], \tag{13}$$

$h(n)$ 关于 $\frac{N-1}{2}$ 偶对称, 而且, $\cos\left[\left(\frac{N-1}{2} - n\right)\omega\right]$ 也对 $\frac{N-1}{2}$ 偶对称, 于是式 (13) 可简写为:

$$H(\omega) = \sum_{n=0}^{r-1} \alpha(n) \cos n\omega, \tag{14}$$

其中, $\alpha(0) = h\left(\frac{N-1}{2}\right)$, $\alpha(n) = 2h\left(\frac{N-1}{2} - n\right)$, $n = 1, 2, 3 \cdots, \frac{N-1}{2}$, $r = \frac{N+1}{2}$, $\Delta(r) = [\alpha(0), \alpha(1), \alpha(2) \cdots \alpha(r-1)]$ 是一组与滤波器脉冲响应相关的系数.

理想的低通滤波器如图 2, 而用 $H(\omega)$ 逼近 $H_d(\omega)$ 的过程可表示为图 3. 由上面的分析可知 FIR 滤波器的设计归结为求一组合适的参数 $\Delta(r)$, 使得按 (14) 式求得的 $H(\omega)$ 在 $[0, \pi]$ 上与所要求的滤波器频率响应的幅度函数 $H_d(\omega)$ 的误差 $E(\omega)$ 最小. 并且要满足约束条件:

$$\begin{cases} H(\omega) = 1 & \omega = 0, \\ H(\omega_c) = 0.5 & \omega_c \text{ 截止频率}, \end{cases} \tag{15}$$

其中, 误差函数 $E(\omega)$ 是将频率分为稠密栅格, 这些栅格对应幅值与理想的幅值间的误差平方和:

$$\min E(\omega_i) = \min \sum_i [H_d(\omega_i) - H(\omega_i)]^2. \tag{16}$$

这是一个带线性约束的优化问题, 可以对本文讨论的带约束的微粒群优化算法进行求解, 得到一组最优的 FIR 滤波器设计参数 $\Delta(r)$.

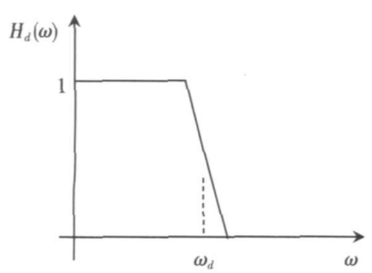


图 2 低通滤波器

Fig.2 Low-pass filter

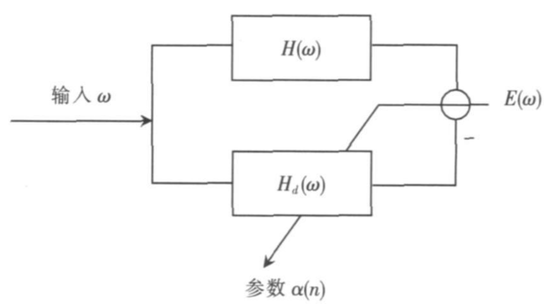


图 3 逼近过程

Fig.3 Approximation process

为了验证采用带约束的微粒群优化算法设计 FIR 滤波器的有效性, 进行了仿真实验. 下面是一个具体的实例. 设计一个低通滤波器, 幅值满足下式:

$$H_d(\omega)=\begin{cases}1 & \omega\in[0,0.2\pi],\\ 0 & \omega\in[0.3\pi,\pi].\end{cases}$$

并且滤波器设计有两个要求:$H(0)=1$;在 $\omega=0.25\pi$ 时,$H(\omega_c)=0.5$.分别用阶数 $N$ 为11和17进行设计.表1列出了一般微粒群算法、突变微粒群算法、一般微粒群约束算法和突变微粒群约束算法的实验结果.

表 2　实验结果

**Table 2　Experimental results**

| 优化算法 | 种群规模 | 迭代次数 | 约束 $H(0)=1$ 处平均误差 | | 约束 $H(w)=0.5$ 处平均误差 | | 平均耗时间 /min | |
|---|---|---|---|---|---|---|---|---|
| | | | $N=11$ | $N=17$ | $N=11$ | $N=17$ | $N=11$ | $N=17$ |
| 一般微粒群 | 200 | 200 | 0.0083 | 0.0971 | 0.1047 | 0.1881 | 28.2 | 35.4 |
| 突变微粒群 | 200 | 200 | 0.0069 | 0.0673 | 0.0408 | 0.0587 | 24.0 | 28.0 |
| 一般微粒群约束 | 200 | 200 | 0.0035 | 0.0620 | 0.0371 | 0.0435 | 12.1 | 14.7 |
| 突变微粒群约束 | 200 | 200 | 0.0021 | 0.0599 | 0.0363 | 0.0469 | 5.6 | 6.9 |

通过对表2中的数据分析可以看出,由于在初始种群及新种群产生时,非约束的微粒群算法需要对种群进行约束条件判断,明显影响了求解的速度,且通过对标准微粒群的改进,也无法提高求解的速度.而采用约束的微粒群算法,克服了这个缺点,可以大大地提高求解速度和精度.

## 4　结论

本文在一般微粒群的基础上进行改进,并与拉格朗日法相结合,研究了带约束微粒群算法.通过拉格朗日对偶原理,将拉格朗日乘子分离出来进行优化,使得线性约束问题通过微粒群优化及少量迭代可得到最优解.该算法可以用于解决工程中带约束的问题.最后,通过对低通滤波器的设计,验证了改进算法的有效性.

[参考文献](References)

[1] Boucher C, Noyer J. A hybrid particle approach for GNSS applications with partial GPS outages[J]. IEEE Trans Instrum Meas, 2010, 59(3): 498-505.

[2] 陈保娣,曾建潮.改进的吸引扩散微粒群算法[J].控制理论与应用, 2010, 27(4): 451-456.
Chen Baodi, Zeng Jianchao. Modified attractive and repulsive particle swarm optimization[J]. Control Theory and Applications, 2010, 27(4): 451-456. (in Chinese)

[3] Naka S, Genji T, Yura T, et al. A hybrid particle swarm optimization for distribution state estimation[J]. IEEE Trans on Power Systems, 2003, 18(1): 60-68.

[4] 王万雷,杨静萍,薄洪光.基于微粒群和满足质量约束的组炉方案优化方法[J].控制理论与应用, 2010, 27(4): 509-512.
Wang Wanlei, Yang Jingping, Bo Hongguang. Optimization of charge design with quality constraints based on particle swarm optimization[J]. Control Theory and Applications, 2010, 27(4): 509-512. (in Chinese)

[5] 朱海梅,吴永萍.一种高速收敛粒子群优化算法[J].控制与决策, 2010, 25(1): 20-24.
Zhu Haimei, Wu Yongping. A PSO algorithm with high speed convergence[J]. Control and Decision, 2010, 25(1): 20-24. (in Chinese)

[6] 胡适耕,施保昌.最优化原理[M].武汉:华中理工大学出版社, 2000: 134-138.
Hu Shigeng, Shi Baochang. Optimization Theory[M]. Wuhan: Huazhong University of Technology Press, 2000: 134-138. (in Chinese)

[7] Shi Yuhui, Eberhart R. A modified particle swarm optimizer[C]// Proc IEEE Int Conf on Evolutionary Computation. Anchorage, 1998: 69-73.

[8] 赖晓平. FIR滤波器约束MinMax设计算法[J].系统工程与电子技术, 2002, 24(2): 84-88.
Lai Xiaoping. Constrained minimax design algorithm for FIR filters[J]. Systems Engineering and Electronics, 2002, 24(2): 84-88. (in Chinese)

[责任编辑:刘　健]