

基于负熵和智能优化算法的盲源分离方法

岳克强, 赵知劲, 沈雷

(杭州电子科技大学通信工程学院, 杭州 310018)

摘要: 针对混合蛙跳算法(SFLA)更新策略会陷入局部最优、降低收敛速度的问题, 提出一种自适应阈值更新策略。根据盲源分离中常用峭度和负熵作为非高斯性的度量, 但峭度对野值敏感, 影响算法性能, 研究一种基于负熵准则的采用粒子群优化(PSO)算法和混合蛙跳算法的盲源分离方法。仿真结果表明, 基于负熵的盲分离算法性能优于基于峭度的盲分离算法, 基于SFLA的盲分离算法性能优于基于PSO的盲分离算法。

关键词: 盲源分离; 粒子群优化算法; 混合蛙跳算法; 阈值选择; 负熵; 峭度

Blind Source Separation Method Based on Negative Entropy and Intelligent Optimization Algorithm

YUE Ke-qiang, ZHAO Zhi-jin, SHEN Lei

(Telecommunication Engineering School, Hangzhou Dianzi University, Hangzhou 310018)

【Abstract】 Aiming at the problem of slowing the convergence and facing local optimization of Shuffled Frog Leaping Algorithm(SFLA) update strategy, this paper proposes a update strategy of adaptive threshold selection is raised. Kurtosis and negative entropy are used as a measure of non-Gaussian in Blind Source Separation(BSS), but kurtosis is sensitive to outliers affecting performance of BSS, it researches a criteria of negative entropy based on Particle Swarm Optimization(PSO) algorithm and SFLA. Simulation results show that the proposed BSS of negative entropy has significant performance improvement over BSS of Kurtosis and BSS based on SFLA has better performance over BSS based on PSO.

【Key words】 Blind Source Separation(BSS); Particle Swarm Optimization(PSO) algorithm; Shuffled Frog Leaping Algorithm(SFLA); threshold selection; negative entropy; kurtosis

盲源分离^[1]是指在不知源信号和传输信道参数的情况下, 根据输入信号的统计特性, 仅由传感器接收到的混合信号恢复出各个源信号的过程。由于所需信号先验知识少, 因此盲源分离方法的适用性强, 可应用于许多领域。目前常用的盲源分离算法大多采用梯度算法对目标函数进行优化确定最优解, 但其收敛速度慢。近年来采用量子遗传算法^[2]等优化算法结合峭度准则进行盲分离的研究引起了人们的注意。但是实际应用中, 峭度对野值敏感, 因此, 本文采用负熵作为非高斯性的度量, 并结合粒子群算法^[3]及混合蛙跳算法^[4-5]进行盲分离的研究。仿真结果表明, 本文算法性能较好。

1 盲源分离基础理论

线性瞬时混合模型描述如下:

$$X(t)=AS(t)+N(t) \quad (1)$$

其中, $X(t)=[x_1, x_2, \dots, x_m]^T$ 为观测信号, $x_i \in C^m$ 为实际观测到的信号; $S(t)=[s_1, s_2, \dots, s_n]^T$, $s_i \in C^n$ 为相互统计独立源信号; A 为 $m \times n$ 维实数矩阵, 称为混合矩阵, 其中, $m > n$; $N(t)=[n_1, n_2, \dots, n_m]^T$ 是由 m 个高斯统计独立噪声信号 $n_i(t)$ 构成。瞬时混合信号盲分离的目标是寻找一个分离矩阵 W , 使得分离后的信号:

$$Y(t)=WX(t)=\hat{S}(t) \quad (2)$$

其中, 恢复信号 $Y(t)$ 是源信号 $S(t)$ 的估计值。

根据中心极限定理, 多个随机变量之和趋近于高斯分布, 如果分离出的是单个独立信号分量, 则其非高斯性最强^[6]。度量非高斯性方法有多种, 本文采用峭度和负熵作为度量非

高斯性的目标函数, 简述如下:

对于零均值的信号, 四阶累积量^[2-3](峭度)是定量度量随机信号 $Y(t)$ 非高斯性的依据, 定义为

$$kurt(Y(t))=E[Y^4(t)]-3(E[Y^2(t)])^2 \quad (3)$$

高斯随机变量的峭度为零, 非高斯性越强, 峭度的绝对值越大; 因此, 当分离信号峭度的绝对值最大时, 信号的非高斯性最强, 认为达到了盲源分离。所以, 可以把峭度绝对值作为盲源分离的目标函数。

概率密度函数为 $p_y(\eta)$ 的随机信号 $Y(t)$ 其微分熵定义为

$$H(Y(t))=-\int p_y(\eta)\log p_y(\eta)d\eta \quad (4)$$

信息论的基本理论表明: 在具有相同方差的所有随机变量中, 高斯变量具有最大的熵, 这表明熵可以作为一种非高斯性的度量^[7]。

负熵 $J(Y)$ 的定义为

$$J(Y)=H(y_{gauss})-H(Y) \quad (5)$$

其中, y_{gauss} 是与 $Y(t)$ 具有相同相关(和协方差)矩阵的高斯随机信号。由上面熵的特性可知, 负熵总是非负的, 当且仅当 $Y(t)$ 具有高斯分布时其值为零。

如果用定义来计算负熵, 则需要预先估计概率密度函数,

基金项目: 电科院预研课题基金资助项目(41101040102)

作者简介: 岳克强(1984-), 男, 硕士研究生, 主研方向: 通信信号处理, 软件无线电; 赵知劲, 教授; 沈雷, 博士

收稿日期: 2009-11-23 **E-mail:** yuekeqiang@163.com

计算复杂。实际中使用非线性函数逼近概率密度函数，负熵的近似计算为

$$J(Y) \approx \sum_{i=1}^p k_i \{E[G_i(Y)] - E[G_i(v)]\}^2, i=1,2,\dots,p \quad (6)$$

其中， k_i 是正常数； v 是具有零均值和单位方差的高斯变量；函数 G_i 是非线性函数； p 是用来逼近概率密度函数的 G_i 的个数。在实验中 p 通常选择为 1 或者 2，本文选 $p=1, k=1$ ，则负熵的逼近公式为

$$J(Y) \propto \{E[G(Y)] - E[G(v)]\}^2 \quad (7)$$

本文选择 $G(Y) = (\log \cosh aY) / a(1 - a^2)$ 作为 $J(Y)$ 的估计。

选择峭度和负熵作为准则时，只有在 $E[yy^T] = I$ 的约束条件下，式(3)和式(7)有局部极大值或极小值，这样，通过最大化或最小化给定限制条件下的表达式，可以得到一组独立源。因此，在算法中，要对信号进行中心化和白化处理，以满足约束条件。

2 基于智能算法的盲源分离方法

2.1 基于粒子群算法的盲源分离方法

粒子群算法^[3]中，设粒子群中粒子个数为 S ，则在一个 D 维的搜索空间中，粒子 i ($1 \leq i \leq S$) 在第 k 次迭代时的位置信息可以表示为 $x_i^k = [x_{i1}^k, x_{i2}^k, \dots, x_{iD}^k]$ ，速度信息表示为 $v_i^k = [v_{i1}^k, v_{i2}^k, \dots, v_{iD}^k]$ ，粒子 i 到目前为止所经历的最好位置为 $p_i^k = [p_{i1}^k, p_{i2}^k, \dots, p_{iD}^k]$ ，群体中所有粒子到目前为止所经历过的最好位置为 $p_b^k = [p_{b1}^k, p_{b2}^k, \dots, p_{bD}^k]$ ，其中， b 为具有最优位置粒子的索引。这里所谓的“最好位置”即为适应度最高的解。采用这种表示方法后，粒子群算法的进化方程可描述为

$$v_{id}^{k+1} = \omega v_{id}^k + c_1 r_1 (p_{id}^k - x_{id}^k) + c_2 r_2 (p_{bd}^k - x_{id}^k) \quad (8)$$

$$x_{id}^{k+1} = x_{id}^k + v_{id}^{k+1} \quad (9)$$

其中， d 表示粒子的第 d 维 ($1 \leq d \leq D$)； ω 为惯性权重； c_1 和 c_2 为学习因子(也称加速常数)； r_1 和 r_2 为分布于 $[0,1]$ 间的随机数。为了减少在进化过程中粒子离开搜索空间的可能性，粒子速度通常限定在一定范围内，即 $v_{id}^k \in [-V_{\max}, +V_{\max}]$ 。

把粒子群算法用于基于负熵准则的盲源分离，步骤如下：

步骤 1 读取混合信号，对信号进行中心化和白化处理。

步骤 2 初始化粒子种群及移动速度，以分离矩阵作为粒子，随机产生一定数量的分离矩阵作为粒子的初始位置。

步骤 3 根据个体分离信号，并对分离后的信号进行中心化和预白化，依照式(7)计算个体的适应度值，并保存全局最优解 p_g^k 和局部最优解 p_b^k 。

步骤 4 按照式(8)和式(9)更新粒子的位置和速度。

步骤 5 重新计算适应度值，更新 p_g^k 和 p_b^k 。

步骤 6 如果达到最大迭代次数，则算法终止；否则，转至步骤 3。

2.2 基于改进的混合蛙跳算法(MSFLA)的盲源分离方法

在混合蛙跳算法^[4-5]中， P 只青蛙组成初始群体，第 i 只青蛙表示问题的解为 $X_i = (x_{i1}, x_{i2}, \dots, x_{is})$ 。将种群内青蛙个体按适应度值降序排列并分成 m 个族群，每个族群有 r 只青蛙，并满足关系 $P = m \times r$ ；对于每一个族群，具有最好适应度的解表示为 X_b ，最差适应度的解表示为 X_w ，而所有族群中具有全局最好适应度的解表示为 X_g 。对每个族群中 X_w 的更新操作一般策略为

$$D_j = \text{rand} \times (X_b - X_w) \quad (10)$$

$$\text{new}X_w = \text{old}X_w + D_j (D_{\max} - D_j - D_{\max}) \quad (11)$$

其中， D_j 表示分量 j 上移动的距离； rand 表示 $0 \sim 1$ 之间的随机数； D_{\max} 表示青蛙所允许改变位置的最大值； $\text{new}X_w$ 表示更新后的解； $\text{old}X_w$ 表示本次循环中的差适应度的解。这种对 X_w 每一维上的分量的随机处理方法易跳过全局最优解^[5]，阈值选择策略改进混合蛙跳算法可以缓解此问题，但固定的阈值使 X_w 的更新仍具有盲目性，不能够较快地收敛，因此，本文提出一种自适应的阈值选择策略，使得算法能够快速收敛到最优。

根据式(10)及式(11)，在对 X_w 不同维 j 上的分量进行更新时，判断 $|D_{j-1} - D_{j-2}|$ 与 $|D_{j-2} - D_{j-3}|$ 的大小，当 $|D_{j-1} - D_{j-2}|$ 较大时，则表明局部更新的数值范围过大，易跳过最优解，此时应采用较小的更新概率进行第 j 维的数值更新；当 $|D_{j-1} - D_{j-2}|$ 较小时，则表明局部更新的数值范围较小，易陷入局部最优，此时应采用较大的更新概率，使得第 j 维数值快速向全局最优解靠近。

综上所述，改进后的局部更新策略为

$$q_0 = \begin{cases} 0.6 - \frac{0.1g}{\text{Max}g} & |D_{j-1} - D_{j-2}| > |D_{j-2} - D_{j-3}| \\ 0.6 + \frac{0.1g}{\text{Max}g} & |D_{j-1} - D_{j-2}| < |D_{j-2} - D_{j-3}| \end{cases} \quad (12)$$

$$D_j = \begin{cases} \text{rand} \times (X_b - X_w) & q > q_0 \\ 0 & q < q_0 \end{cases} \quad (13)$$

其中， $\text{Max}g$ 为族内最大循环代数； g 为每一次更新的循环次数； $q_0 \in [0,1]$ 为局部更新阈值； q 为 $[0,1]$ 范围内随机数； D_j 更新完成以后，按照式(11)更新 X_w 。

把 MSFLA 用于基于峭度和负熵准则的盲源分离，基本步骤如下：

步骤 1 读取混合信号，对信号进行中心化和白化处理。

步骤 2 初始化青蛙种群，随机产生 P 个分离矩阵作为初始个体。

步骤 3 根据个体分离信号，并对分离后的信号进行中心化和预白化，依照式(3)和式(7)计算个体的适应度值，按照适应度值进行排序，然后青蛙进行分族并记录局部最优解 X_b 、局部最差解 X_w 和全局最优解 X_g 。

步骤 4 对族群进行局部深度搜索，按照式(12)、式(13)及式(11)更新 X_w ，如果没有改进，则用 X_g 代替 X_b 继续迭代；如果仍没有改进，则随机产生一个 X_w 。

步骤 5 重复更新操作，直到设定的迭代次数。完成所有族群的局部深度搜索后，将重新混合所有青蛙并排序和划分族群，然后进行局部深度搜索，重复执行步骤 3 和步骤 4 直到满足终止条件。

3 仿真及结果分析

仿真中采用的算法有基于粒子群峭度算法的盲源分离(KPSO-BSS)、基于粒子群负熵算法的盲源分离(JPSO-BSS)、基于改进的混合蛙跳峭度算法的盲源分离(KMSFLA-BSS)和基于改进的混合蛙跳负熵算法的盲源分离(JMSFLA-BSS)。为了便于比较，PSO 和 MSFLA 的种群数都设为 21，其中 PSO 的参数设置为： $C_1 = C_2 = 2$ ， $V_{\max} = 4$ ，迭代次数为 25；MSFLA 的参数设置为：混合迭代次数为 25，族群数 m 为 3，以及 D_{\max} 为 0.8，族内迭代次数为 10。

实验用 2 路语音信号作为源信号, 采样点数为 50 000 点, 如图 1 所示。对 2 路信号进行瞬时混合, 混合矩阵为

$$A = \begin{bmatrix} -0.129 & 1 & -0.952 & 3 \\ 0.279 & 3 & 0.624 & 5 \end{bmatrix}$$

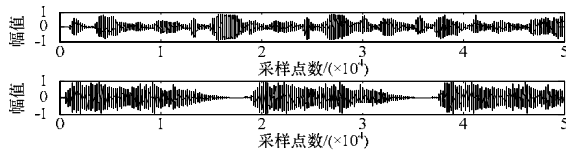


图 1 源信号

采用 KPSO-BSS, JPSO-BSS, KMSFLA-BSS 和 JMSFLA-BSS 的分离结果分别如图 2~图 5 所示。由图可见, 基于负熵准则的 JPSO-BSS 和 JMSFLA-BSS 优于基于峭度的 KPSO-BSS 和 KMSFLA-BSS, 基于 MSFLA 的盲分离算法性能优于基于 PSO 的盲分离算法。

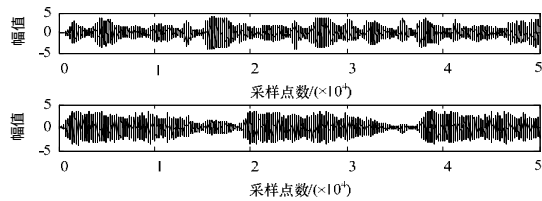


图 2 KPSO-BSS 分离

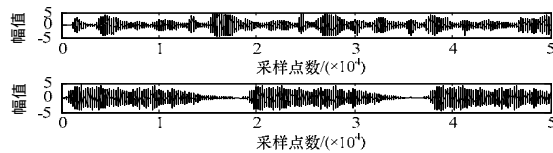


图 3 JPSO-BSS 分离

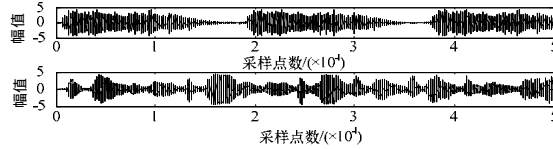


图 4 KMSFLA-BSS 分离

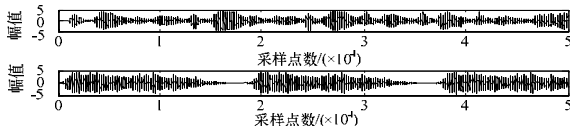


图 5 JMSFLA-BSS 分离

为检验估计信号与源信号的相似程度及算法性能, 本文选择以下 2 种性能指标:

(1) 相似系数

$$\gamma_{ij} = \gamma(y_i, s_j) = \left| \frac{\sum_{t=1}^N y_i(t) s_j(t)}{\sqrt{\sum_{t=1}^N y_i^2(t) \sum_{t=1}^N s_j^2(t)}} \right| \quad (14)$$

当相似系数越接近于 1, 源信号与估计信号越接近, 则可认为该算法分离效果更为理想。

(2) 输出信号 $y_i (i=1, 2, \dots, n)$ 的均方误差, 定义该量为

$$MSE_i = \frac{1}{N} \sum_{t=1}^N [y_i(t) - s_i(t)]^2 \quad (15)$$

$$MSE = \frac{1}{n} \sum_{i=1}^n MSE_i \quad (16)$$

其中, N 表示选取的数据长度, 即采样点数; n 表示源信号个数。当 MSE 越接近于 0, 源信号与估计信号越接近, 则可认为该算法分离效果更为理想。

KPSO-BSS, JPSO-BSS, KMSFLA-BSS 和 JMSFLA-BSS

的盲分离的相似系数如表 1 所示。由表 1 可知, 基于负熵准则的 JPSO-BSS 和 JMSFLA-BSS 优于基于峭度的 KPSO-BSS 和 KMSFLA-BSS, 基于 MSFLA 的盲分离算法性能优于基于 PSO 的盲分离算法。

表 1 2 种算法相似系数比较

| 算法 | 峭度的盲源分离 | | 负熵的盲源分离 | |
|-----------|---------|---------|---------|---------|
| PSO 盲分离 | 0.946 5 | 0.311 8 | 0.988 7 | 0.047 9 |
| | 0.312 8 | 0.948 4 | 0.047 4 | 0.978 6 |
| MSFLA 盲分离 | 0.012 1 | 0.987 9 | 0.999 7 | 0.016 9 |
| | 0.988 9 | 0.011 5 | 0.017 4 | 0.999 6 |

图 6 给出了均方误差随迭代次数增加的曲线。由图可知, 基于负熵的盲源分离方法的 MSE 较小, 且波动小, 性能优于基于峭度的盲源分离方法。

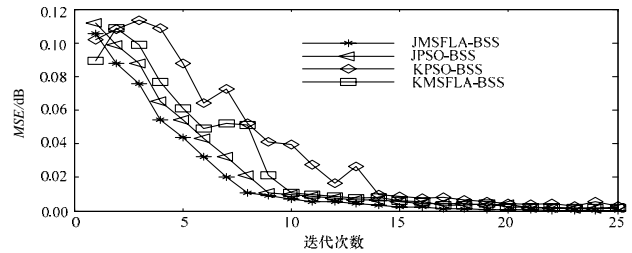


图 6 MSE 性能曲线

表 2 给出了在 25 次迭代以后的 MSE 的值。由表 2 可知, 基于负熵准则的 JPSO-BSS 和 JMSFLA-BSS 优于基于峭度的 KPSO-BSS 和 KMSFLA-BSS, 基于 MSFLA 的盲分离算法性能优于基于 PSO 的盲分离算法, JMSFLA-BSS 的性能优于其他 3 种方法。

表 2 2 种算法均方误差比较

| 算法 | 峭度的盲源分离 | 负熵的盲源分离 |
|-----------|----------|-----------|
| PSO 盲分离 | 0.003 73 | 0.001 760 |
| MSFLA 盲分离 | 0.001 81 | 0.000 582 |

4 结束语

本文介绍了粒子群及混合蛙跳算法的基本理论并提出一种改进的混合蛙跳算法, 研究这 2 种智能优化算法与负熵准则结合的盲源分离方法。仿真表明, 提出的混合蛙跳算法的盲源分离和基于负熵准则的盲源分离方法具有较优的性能。

参考文献

- [1] Jutten C, Herault J. Space or Time Adaptive Signal Processing by Neural Network Models[C]//Proc. of Int'l Conf. on Neural Networks for Computing. Snowbird, USA: [s. n.], 1986: 206-211.
- [2] 杨俊安, 李 斌. 基于量子遗传算法的盲源分离算法研究[J]. 小型微型计算机系统, 2003, 24(8): 1518-1523.
- [3] 张文爱, 刘俊豪. 基于粒子群算法的盲源分离算法[J]. 太原理工大学学报, 2006, 37(2): 169-172.
- [4] Shie Y H, Atiquzzaman M. Optimal Design of Water Distribution Network Using Shuffled Complex Evolution[J]. Journal of the Institution of Engineers, 2004, 44(1): 93-107.
- [5] Elbeltagi E, Hegazy T, Grierson D. Comparison Among Five Evolutionary-based Optimization Algorithms[J]. Advanced Engineering Informatics, 2005, 19(1): 43-53.
- [6] 张新成, 杜志勇. 盲源分离技术及其应用[J]. 电声基础, 2004, (4): 426-428.
- [7] 周宗潭, 董国华, 徐 昕, 等. 独立成分分析[M]. 北京: 电子工业出版社, 2007.

编辑 索书志