

# 一种强干扰下超弱水声信号 自适应盲分离的快速算法\*

倪晋平<sup>1,2</sup>, 马远良<sup>2</sup>, 张忠兵<sup>2</sup>

(1. 西安工业学院, 西安 710032; 2. 西北工业大学 航海工程学院, 西安 710072)

摘要: 许多盲源分离算法对超弱信号收敛速度变慢甚至失效。文章提出一种算法, 用 FASTICA 算法所得的分离矩阵作为一种基于输出去相关自适应盲分离算法的初始值, 算法的收敛速度得到明显改善。算法的学习速率通过一种自适应算法选取, 从而降低了算法收敛速度对学习步长的依赖。仿真结果表明, 文中提出的算法对超弱信号特别有效, 算法收敛速度大大增强。该算法可以用于分离强干扰下的弱水声信号。

关键词: 盲源分离; 弱信号检测; 快速独立成分分析; 水声

中图分类号: TN911.7, TB566 文献标识码: A

## An adaptive fast algorithm for blind sources separation with super weak signals in the stronger jammer

NI Jin-ping<sup>1,2</sup>, MA Yuan-liang<sup>2</sup>, ZHANG Zhong-bing<sup>2</sup>

(1. Xi'an institute of Technology, Xi'an 710032, China;

2. College of Marine Engineering, Northw estern Polytechnical University, Xi'an 710072, China)

**Abstract:** Either most of algorithms for blind source separation fail with the super-weak signal, or convergence speed of the algorithm is slow. This paper presents an adaptive algorithm as follows. Using the separate matrix yielding from the FASTICA algorithm as original separate matrix may improve the converge speed of the adaptive algorithm which based on decorrelation of output signals. Learning rate were chosen by an adaptive algorithm such that the convergence speed wasn't sensitive to learning step. Computer simulation demonstrates the algorithm is good especially for super-weak signals. The convergence speed of present algorithm is higher in contrast to other algorithms. Present algorithm can be used to separate super weak underwater acoustics signals in the stronger jammer.

**Key words:** Blind sources separation; weak signal detection; fast independent component analysis; underwater acoustics.

### 1 引言

盲源分离是在仅知道传感器观测数据的情况下, 从中分离源信号的过程, 已广泛应用于通讯、地质勘测、生物信号检测、语音增强等领域。在高信噪比条件下, 大多数算法十分有效。然而, 在核爆检测和安静型潜艇的检测中, 目标信号十分微弱, 用常规的处理算法已很难分离源信号, 许多算法收敛速度变得相当缓慢, 有的甚至失效。A. Cichocki<sup>[1]</sup>等人在 HJ 神经网络<sup>[2]</sup>算法的基础上使用一种全反馈型的神经网络, 基于输出信号去相关而建立目标函数, 用随机梯度降方法自适应搜寻分离矩阵, 该算法(以下简称 AC 算法)对从强信号中分离弱信号和混合矩阵近似奇异的情况下十分有效。由于该算法的收

敛速度和收敛性与选取的学习速率序列密切相关, 对源信号混合比强弱相差悬殊的情况, 收敛速度变慢, 因此很难应用于实际工程。Aapo Hyvarinen<sup>[3]</sup>于 1997 年提出了一种基于输出信号峰度最大(最小)化的盲源分离的算法—FASTICA 算法, 该算法仅在迭代几步后收敛到分离矩阵, 收敛速度明显优于 AC 算法; 然而, 由于该算法对观测数据进行了预白化处理, 算法对源信号混合比强弱相差较大的情况失效。该算法是一种块处理算法, 很难在线实现。本文在以上两种算法的基础上, 针对在线实现自适应盲源分离的目的, 提出一种快速盲源分离算法。提出的算法综合以上两种算法的优点, 对分离超弱信号十分有效。最后将该算法引入复数信号, 从而可以用于分离强干扰下的超弱水声信号。

### 2 算法

#### 2.1 模型

给定  $n$  个统计独立的源信号,  $S^T(t) = [s_1(t),$

收稿日期: 2000-10-30; 修回日期: 2001-03-22

作者简介: 倪晋平(1965-), 男, 陕西乾县人, 博士研究生, 研究方向: 阵列信号处理、盲信号处理。

$\dots, s_n(t)]$ ,  $t$  表示时间,  $T$  表示转置。假设用  $m$  个传感器接收  $n$  个源信号, 由于每个源信号到达每一个传感器的途径各异, 每个传感器接收的信号是  $n$  个源信号按一定的系数的混合。

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

式(1)中  $X(t) = [x_1(t), \dots, x_m(t)]^T$  为传感器的观测数据,  $N(t) = [n_1(t), \dots, n_m(t)]$  为传感器接收的噪声, 假设为零均值的高斯信号,  $A \in C^{m \times m}$ ,  $A$  的元素可以是常数, 也可以是函数。本文只限于讨论  $A$  的元素为常系数的情况, 即线性瞬时混合型的盲源分离。因为  $N(t)$  的引入, 使得盲分离问题变得相当困难, 本文忽略  $N(t)$ , 在仿真实验中只考核了在  $SNR > 0$  情况下算法的性能, 并作以下假定:

- (a)  $n$  个源信号统计独立, 零均值。传感器的数目与源信号的个数相等, 即:  $m = n$ ;
- (b) 源信号中最多只能有一个是高斯信号;
- (c) 源信号的峰度小于于零;
- (d) 混合矩阵列满秩。

盲源分离过程是仅知  $X(t)$ , 寻找一个分离矩阵  $W$ , 使  $W = A^{-1}$  则

$$Y(t) = WX(t) = S(t) \quad (2)$$

式(2)中  $Y(t)$  为盲源分离的输出(以下简称输出),  $Y(t) \in C^{n \times 1}$ , 由于信息存在于信号的波形中,  $S(t)$  中的元素在  $Y(t)$  中的次序变更和幅值变化是可以接受的, 因此

$$WA = I \text{ 或 } WA = P\Lambda \quad (3)$$

式(3)中  $I$  为单位阵,  $P$  为置换矩阵,  $\Lambda$  为任意非奇异对角阵。找到满足式(3)的  $W$ , 就认为实现了盲源分离。

## 2.2 输出去相关的神经网络自适应盲源分离算法(AC算法)

A. Cichocki 等人于 1996 年提出一种 HJ 神经网络盲源分离的修正算法<sup>[1]</sup>, 在网络中增添了输出自反馈。基于输出去相关建立目标函数, 并用随机梯度降方法推导出一种在线完成的自适应盲源分离算法

$$W(k+1) = W(k) + \mu(k)(I - f(Y)g^T(Y)W(k)) \quad (4)$$

式(4)中  $\mu(k)$  为学习速率序列,  $f(\bullet)$ ,  $g(\bullet)$  为非线性函数,  $W(0) = I$ 。

上述算法对混合比强弱相差较大的混合矩阵  $A$  和  $A$  近似奇异的情况十分有效, 但  $\mu(k)$  序列的选取对算法收敛的速度影响较大, 选取不合适甚至导致算法发散。

## 2.3 基于输出峰度值最小(大)的盲源分离算法

### (FASTICA 算法)

Aapo Hyvarinen 于 1997 年提出了一种基于输出峰度值最大(小)的盲源分离算法<sup>[3,4]</sup>。

首先对  $X(t)$  进行白化处理, 即

$$V(t) = MX(t) = MAS(t) = BS(t) \quad (5)$$

$M$  为白化矩阵。寻找正交分离矩阵  $W$  的一个行向量  $w_i$  的算法由下式给出

$$w_i^* = E\{vg(w_i^T v)\} - E\{g'(w_i^T v)\}w_i$$

$$w_i = w_i^* / \|w_i^*\| \quad (6)$$

式(6)中  $g(y)$  是一合适选取的非线性函数, 典型的有  $g(y) = y^3$ ,  $g(y) = \tanh(y)$ ,  $g'(y)$  是  $g(y)$  的导数, 期望值由样本数据的估计值给出。

寻找  $W$  的所有分量算法步骤如下<sup>[3]</sup>:

- (1) 对  $X(t)$  进行白化处理得到  $V(t)$ ,  $M = D^{-\frac{1}{2}}E^H$ ,  $D = \text{diag}[\lambda(1), \dots, \lambda(n)]$ , 为  $n \times n$  对角矩阵,  $E = [c(1), \dots, c(n)]$ ,  $E \in C^{n \times n}$ ,  $\lambda(i)$  表示协方差矩阵  $E[X(t)X^H(t)]$  第  $i$  个最大的特征值,  $c(i)$  为相应特征值对应的特征向量;  $A$

- (2) 随机选取  $W(0)$ , 并归一化, 使  $k = 1$ ;

- (3)  $W = E\{Vg(W^H v)\} - E\{g'(W^H V)\}W$ ;

- (4) 正交化  $W(k)$ ,  $W(k) = W(k)(W(k)^H W(k))^{-\frac{1}{2}}$ ;

- (5) 如果  $|W(k)W(k-1)|$  不接近单位矩阵,  $k = k + 1$ , 返回到第(3)步重新计算; 否则停止计算, 输出  $W(k)$ ;

- (6) 最终的分离矩阵  $W = W(k)^H M$ 。

该算法的特点是收敛速度快, 但由于对  $X(t)$  进行了白化处理, 在混合比强弱相差较大的情况下, 容易丢失弱信号, 甚至不能分离出弱信号, 而在许多情况下, 弱信号是有用的目标信号。

## 2.4 超弱信号的快速盲分离算法

仿真实验发现 AC 算法对  $W(0)$ 、 $\mu$  的选取比较敏感。我们采用下式自适应选取  $\mu$ , 可以降低算法对  $\mu$  的敏感性。

$$\mu(k+1) = \mu(k) - (1 - \alpha_0 \|I - f(Y)g^T(Y)\|)\mu(k) \quad (7)$$

式中  $0 < \alpha_0 < 1$ ,  $\alpha_0$  调节  $\mu$  的变化速度。采用(7)式选取  $\mu$  序列, AC 算法对  $\alpha_0$  以及  $\mu(0)$  的选取不太敏感, 但在  $W(0) = I$  的条件下算法对弱信号仍然存在收敛速度较慢的问题。

为了提高 AC 算法对超弱信号的收敛速度, 我们采用 FASTICA 算法得到的分离矩阵  $W$  作为 AC 算法的  $W(0)$ , 用 AC 算法实现分离, 仿真结果发

现,该算法对超弱信号收敛速度大幅度提高。

该算法具有以下特点:(1)容易在线实现。由于AC算法是在线自适应算法,我们提出的算法完全能够在线实现。若用单台计算机,选用FASTICA算法确定 $W(0)$ ,然后用AC算法自适应在线计算。若用双台计算机,一台用FASTICA算法计算 $W(0)$ ,一台用AC算法自适应在线计算。可以完成对非平稳信号的分离,真正做到在线实时实现超弱信号的盲分离。(2)自适应选取学习速率序列,放宽了算法对输入信号幅度的要求;(3)快速分离超弱信号。

按以下算法将AC算法扩展至复数域,有

$$W(k+1) = W(k) + [I - f[Y(k)]]g^H[Y(k)]W(k) \quad (8)$$

式(8)中 $f[Y(k)] = [f_1[y_1(k)], \dots, f_n[y_n(k)]]^T$ ,  $f_i[y_i(k)] = y_i(k)f_i[|y_i(k)|^2]$ ,  $i = 1, 2, \dots, n$ ;  $g[Y(k)] = [g_1[y_1(k)], \dots, g_n[y_n(k)]]^T$ ,  $g_i[y_i(k)] = y_i(k)g_i[|y_i(k)|^2]$ ,  $i = 1, 2, \dots, n$ ;  $H$ 表示复共轭转置。

(8)式算法对混合矩阵为复数和复数源信号,可以成功实现盲源分离。与实数信号相比,复数算法收敛速度慢。对相同的输入信号(信号强弱混合比相同,采样长度相同)复数算法分离弱信号的能力没有实数算法强。

对接收窄带水声信号的水声基阵,(8)式的算法可以直接用于该情况下水声信号的盲分离。对宽带水声信号可以通过一定的方法化为窄带水声信号盲分离问题。

## 3 仿真

### 3.1 收敛速度的比较

通过计算机仿真比较AC算法收敛速度。采用文献[5]中的方法定义收敛速度的评价标准。两个同维数的可逆矩阵 $A$ 、 $B$ 二者之间的距离按下式定义

$$\begin{aligned} \varepsilon(A, B) = & \sum_i \left| \sum_j |D_{ij} - 1| \right|^2 + \\ & \sum_j \left| \sum_i |D_{ij} - 1| \right|^2 + \\ & \sum_i \left| \sum_j |D_{ij}^2 - 1| \right|^2 + \\ & \sum_j \left| \sum_i |D_{ij}^2 - 1| \right|^2 \end{aligned} \quad (9)$$

(9)式中, $i$ 表示矩阵元素行序号, $j$ 表示列序号,

$$D = A^{-1}B \quad A = A \Delta_A^{-1}, B = A \Delta_B^{-1} \quad (10)$$

$$\Delta_A(k, k) = \|A_{ik}\|, \Delta_B(k, k) = \|B_{ik}\| \quad (11)$$

$\Delta_A$ 与 $\Delta_B$ 为对角方阵,矩阵的范数可以任意选取。

如果 $A = B$ 或 $B = P \Lambda A$ , $P$ 为置换矩阵, $\Lambda$ 为对角阵,则 $\varepsilon(A, B) = 0$ ,对任意两个矩阵 $A$ 和 $B$ , $\varepsilon(A, B) \geq 0$ 。对AC算法,如果 $WA = I$ 或 $WA = P \Lambda I$ ,则实现了盲分离,因此,矩阵 $I$ 与 $WA$ 的距离作为度量算法收敛速度的基本量。 $\varepsilon(I, WA)$ 愈小, $W$ 愈接近收敛值。将每次迭代的 $\varepsilon_i$ ( $i$ 表示迭代次数)总和作为收敛速度的度量,在相同迭代次数下,该值愈大,收敛速度愈慢,反之,愈快。

用下述两种方法选取 $W(0)$ :(a) $W(0) = I$ (b)采用FASTICA算法的结果。比较两种情况下算法的收敛速度。

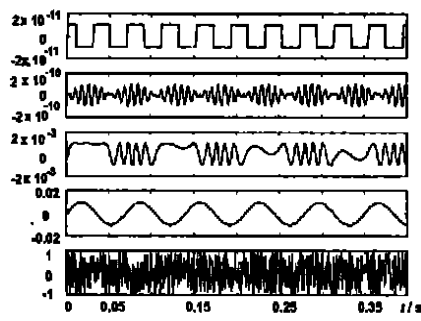
采用5个传感器接收5个源信号,源信号表示式如下

$$\begin{aligned} s_1(t) &= 10^{-11} \operatorname{sgn}(\cos(155t)); \\ s_2(t) &= 10^{-10} \sin(800t) \cdot \sin(60t); \\ s_3(t) &= 10^{-3} \sin((300t) + 6\sin(61t)); \quad (12) \\ s_4(t) &= 10^{-2} \sin(90t); \\ s_5(t) &= 2\operatorname{rpiid}(N, 'uni') \end{aligned}$$

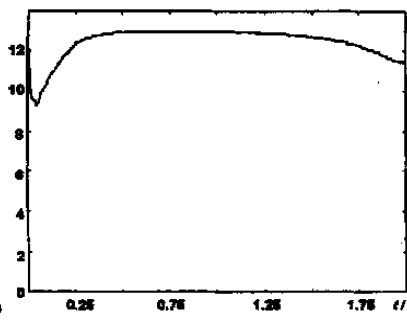
$\operatorname{sgn}$ 为符号函数, $\operatorname{rpiid}$ 为Matlab中的函数,产生独立同分布的随机数,“uni”是指在 $[-0.5, 0.5]$ 均匀分布,因此, $s_5$ 为在 $[-11]$ 均匀分布的随机数。 $N$ 为数据长度,采样频率为 $F_s = 2\text{kHz}$ ,信号长度 $N = 4000$ ,持续时间为 $2s_0$ 。混合矩阵 $A$ 随机产生, $A = [\alpha_{ij}]_{5 \times 5}$ , $\alpha_{ij} \in (-1, 1)$ 间的随机数。

图1中(a)为源信号 $S(t)$ 的波形图,图中自上而下依次为 $s_1(t)$ 、 $s_2(t)$ 、 $s_3(t)$ 、 $s_4(t)$ 和 $s_5(t)$ ,只研究 $S(t)$ 的大小,故省去量纲。以下 $X(t)$ 和图2及图3中的 $Y(t)$ 相同。图1(b)为混合以后的传感器观测信号 $X(t)$ ;图2为AC算法在 $W(0) = I$ 时的分离结果 $\varepsilon_i$ ( $i = 1, \dots, N, t = i/F_s$ , $\varepsilon$ 为一标量无量纲,以下同)和 $Y(t)$ ;图3是AC算法用FASTICA算法所得结果作为 $W(0)$ 时分离结果 $\varepsilon$ 和 $Y(t)$ 。表1给出了不同步长因子情况下,收敛速度的比较结果,表中 $\mu(0) = 0.02$ 。

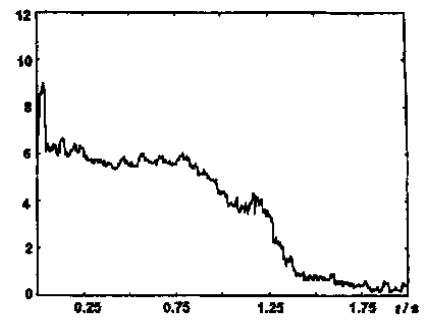
讨论:(1)在相同的迭代次数下对超弱信号进行分离,修改的AC算法收敛速度比AC算法明显提高;(2)AC算法对步长因子 $\alpha_0$ 敏感, $\alpha_0$ 愈大,收敛速度加快, $\alpha_0$ 太大,算法会发散;而修改AC算法对 $\alpha_0$ 敏感性减低,选取一合适值,既增强算法的收敛速度,又不会发散。



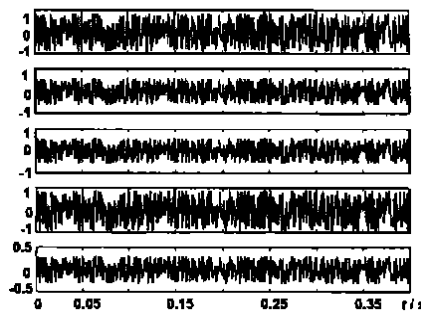
(a) 源信号  $S(t)$



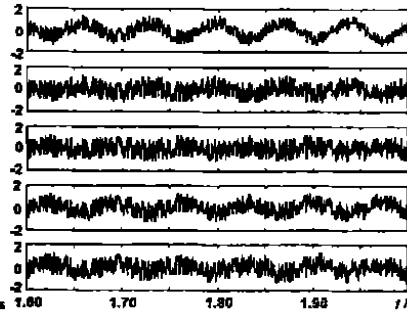
(a)  $\sigma_i$  曲线



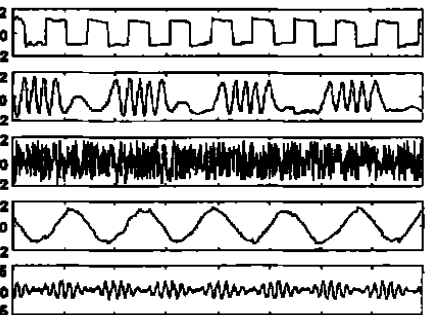
(a)  $\varepsilon$  曲线



(b) 传感器观测信号  $X(t)$



(b) 分离输出  $Y(t)$



(b) 输出  $Y(t)$

图 1 源信号和观测信号

图 2  $W(0) = I$  时 AC 算法分离结果

图 3 用 FASTICA 算法所得结果作为  $W(0)$  时的 AC 算法分离结果

表 1 不同步长下算法分离误差

$\alpha_0$	$W(0) = I$		FASTICA	
	$\sum \varepsilon_i$	$\varepsilon_i$	$\sum \varepsilon_i$	$\varepsilon_i$
0.0025	$3.4 \times 10^4$	9	$1.8 \times 10^4$	5
0.002	$3.7 \times 10^4$	9	$1.2 \times 10^4$	3
0.0015	$3.9 \times 10^4$	10	$1.0 \times 10^4$	3
0.001	$4.4 \times 10^4$	11	$1.0 \times 10^4$	3
0.0005	$4.9 \times 10^4$	12	$1.0 \times 10^4$	3

仿真试验发现,在源信号最大的幅值为 1 时,修改的 AC 算法对实数信号能分离的最弱信号的幅值为  $10^{-11}$ ,对复数信号为  $10^{-6}$ 。

### 3.2 强干扰下超弱水声信号的分离

采用三元均匀线列阵,背景噪声采用均匀噪声模型仿真,噪声为窄带高斯分布信号,均值为 0,基阵接收窄带信号的中心频率为  $f_c = 1550\text{Hz}$ ,采样频率为  $F_s = 10\text{kHz}$ ,信号长度为 4000,阵元间距  $d = 0.5$ ,即阵元接收的噪声彼此不相关,信号从  $4^\circ$  方向入射,频率为  $f_1 = 1500\text{Hz}$ ,干扰信号从  $5^\circ$  方向入射,频率为  $f_2 = 1600\text{Hz}$ ,信号与干扰功率比  $\text{SINR} = -60\text{dB}$ ,信噪比  $\text{SNR} = 10\text{dB}$ ,仿真结果表明,信号得到很好的分离。但对  $d < 0.5$  时,即相关阵元接收的噪声彼此相关的情况,分离效果变差,在此情况的信号分离算法值得进一步研究。

## 4 结论

我们对基于神经网络的盲源分离算法进行了修改,用基于输出峰度值最大(小)的 FASTICA 算法获得的结果作为 AC 算法的初值  $W(0)$ ,并用一种自动选取学习步长的算法,降低 AC 算法对学习速率选取的敏感性,该算法提高了对超弱信号的盲分离速度,并容易并行在线实现。将算法扩展至复数信号,可直接用于对窄带超弱水声信号在强干扰下的盲源分离。

参考文献:

- [1] Andrzej Cichocki, Robert E. Bogner, Leszek Moszczynski and Kenneth Pope. Modified Herault - Jutten algorithms for blind separation of source[J]. Digital Signal Processing. 1997, 9(7): 80-93.
- [2] C. Jutten, J. Herault. Blind separation of source, Part I: An adaptive algorithm based on neuromimetic architecture [J]. Signal Processing, 1991, 24(1): 1-10.
- [3] Aapo Hyvarinen and E. Oja. A fast fixed-point algorithm for independent component analysis[J]. Neural computation. 1997, 9(7): 1438-1492.
- [4] Aapo Hyvarinen. Fast and robust fixed-point algorithms for independent component analysis[J]. IEEE Trans. On Neural Networks. 1999, 10(3): 626-631.
- [5] Pierre Comon. Independent component analysis, a new concept[J]. Signal Processing. 1994, 36(2): 287-314.