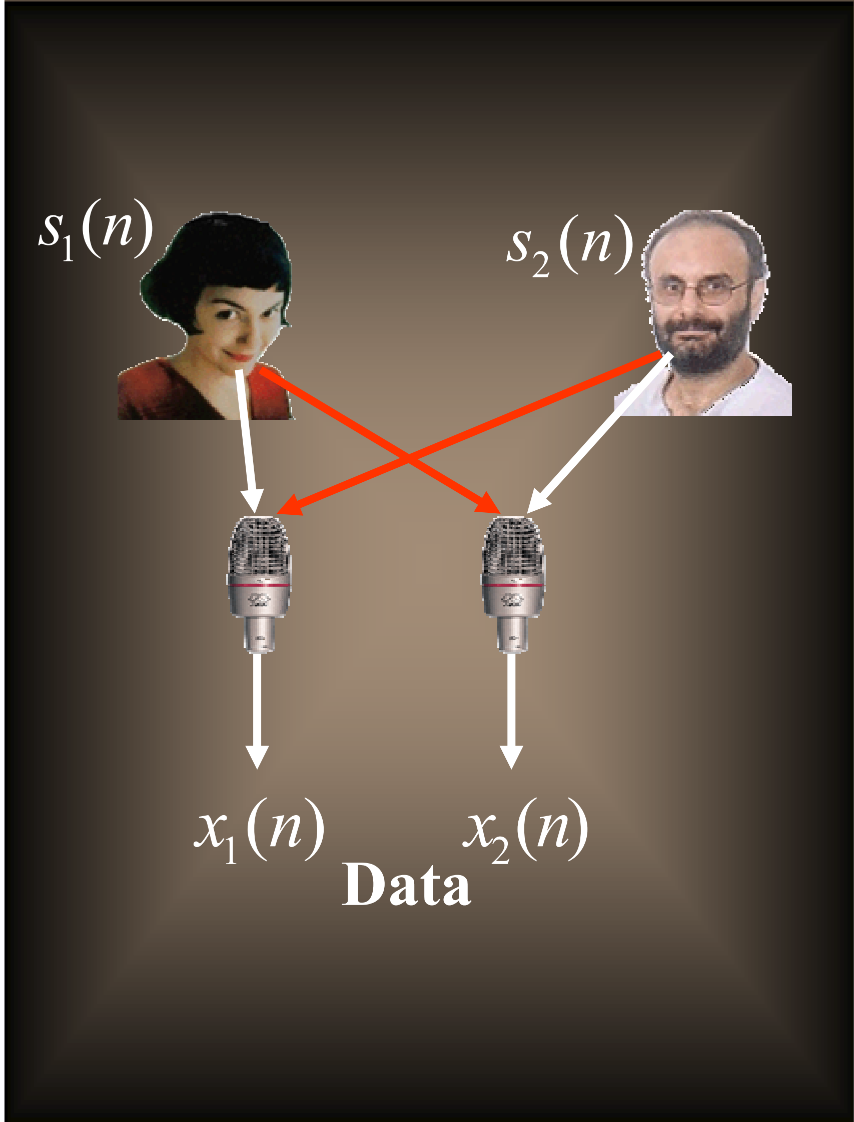
Optimum Adaptive Algorithms
for
Blind Source Separation

George V. Moustakides
INRIA, Project IRIS

Outline

- Problem definition
- Existing adaptive scheme
- General adaptive scheme
  - Imposing the correct limits
  - Imposing symmetric behavior
- Performance measure
- Optimum algorithms
- Conclusion

Problem Definition



$$x_1(n) = a_{11}s_1(n) + a_{12}s_2(n)$$

$$x_2(n) = a_{21}s_1(n) + a_{22}s_2(n)$$

$$S(n) = \begin{bmatrix} s_1(n) \\ s_2(n) \end{bmatrix}, X(n) = \begin{bmatrix} x_1(n) \\ x_2(n) \end{bmatrix}$$

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

$$X(n) = \mathbf{A}S(n)$$

If $X(n)$ available sequentially, estimate adaptively $S(n)$ or equivalently $\mathbf{B} = \mathbf{A}^{-1}$

$$S(n)=\begin{bmatrix} s_1(n) \\ s_2(n) \\ \vdots \\ s_M(n) \end{bmatrix}, X(n)=\begin{bmatrix} x_1(n) \\ x_2(n) \\ \vdots \\ x_N(n) \end{bmatrix}, \mathbf{A}=\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1M} \\ a_{21} & a_{22} & \cdots & a_{2M} \\ \vdots & \vdots & \vdots & \vdots \\ a_{N1} & a_{N2} & \cdots & a_{NM} \end{bmatrix}$$

$$X(n)=\mathbf{A}S(n), \quad N \geq M$$

Matrix $\mathbf{A}$ must be full rank

Existing adaptive technique

Let N=M=2 and $\mathbf{B}(n)$ denote the estimate of $\mathbf{A}^{-1}$ at time n.
At time n available: $\mathbf{B}(n\text{-}1)$, $X(n)$

$$\hat{S}(n) = \mathbf{B}(n-1)X(n)$$

$$\mathbf{B}(n) = \mathbf{B}(n-1) - \mu\mathbf{H}\left(\hat{S}(n)\right)\mathbf{B}(n-1),\ \mathbf{B}(0) = \mathbf{I}$$

μ is a constant (step size) with $0 < \mu << 1$.

$$\mathbf{H}(Z) = \begin{bmatrix} z_1^2 - 1 & z_1 z_2 \\ z_1 z_2 & z_2^2 - 1 \end{bmatrix} + \begin{bmatrix} 0 & g(z_1)z_2 - g(z_2)z_1 \\ g(z_2)z_1 - g(z_1)z_2 & 0 \end{bmatrix}$$

$g(z)$ is an odd univariate nonlinear function (i.e. $g(-z) = -g(z)$).

Acceptable solution(s)

$$\hat{S}(n) = \mathbf{B}(n-1)X(n) = \mathbf{B}(n-1)\mathbf{A}S(n) = \mathbf{C}(n-1)S(n)$$

Desired: $$\mathbf{C}(n) \to \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \hat{S}(n) \to \begin{bmatrix} s_1(n) \\ s_2(n) \end{bmatrix}$$

$$\mathbf{C}(n) \to \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad \hat{S}(n) \to \begin{bmatrix} s_2(n) \\ s_1(n) \end{bmatrix}$$

Acceptable: $$\hat{S}(n) \to \begin{bmatrix} \pm s_1(n) \\ \pm s_2(n) \end{bmatrix}, \quad \mathbf{C}(n) \to \begin{bmatrix} \pm 1 & 0 \\ 0 & \pm 1 \end{bmatrix}$$

$$\to \begin{bmatrix} \pm s_2(n) \\ \pm s_1(n) \end{bmatrix}, \quad \mathbf{C}(n) \to \begin{bmatrix} 0 & \pm 1 \\ \pm 1 & 0 \end{bmatrix}$$

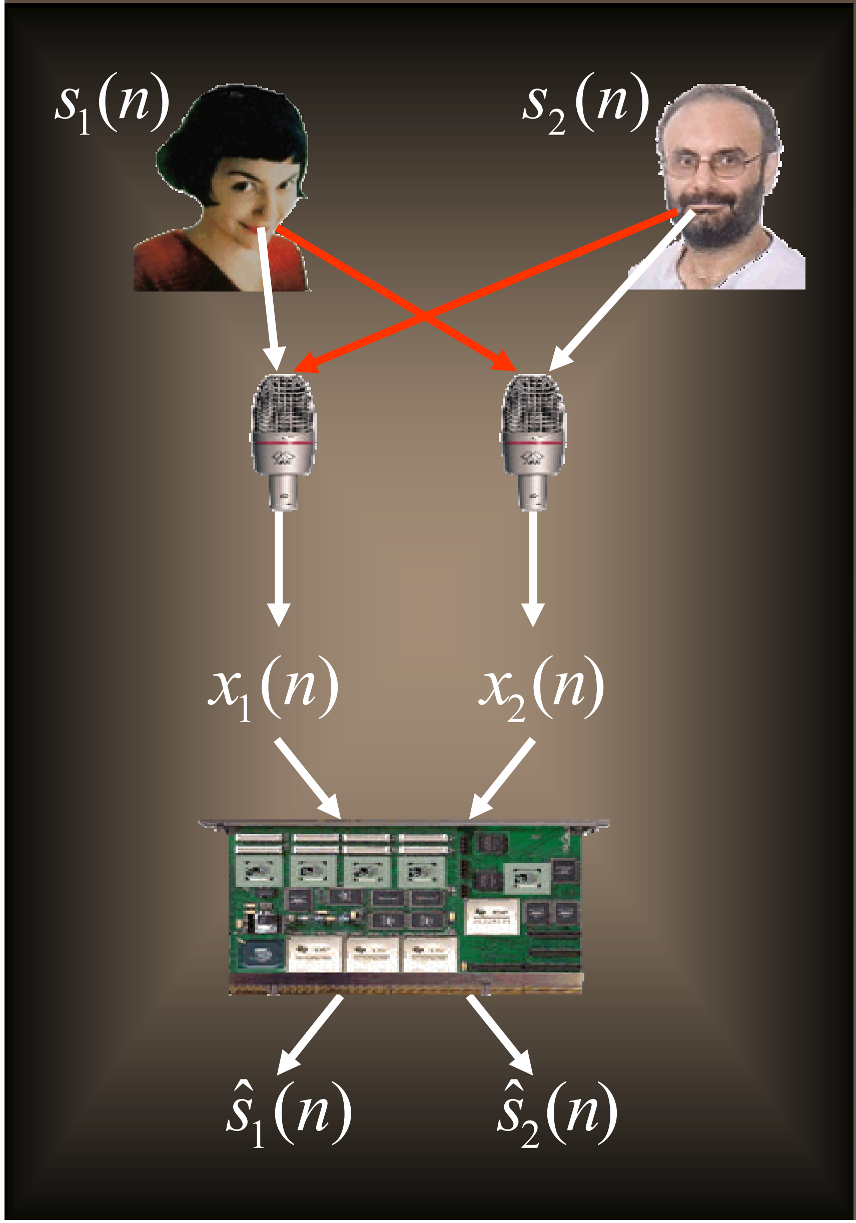
Does $\mathbf{C}(n) = \mathbf{B}(n)\mathbf{A}$ converge to one of the eight matrices

$$\begin{bmatrix} \pm 1 & 0 \\ 0 & \pm 1 \end{bmatrix} \text{ or } \begin{bmatrix} 0 & \pm 1 \\ \pm 1 & 0 \end{bmatrix} ?$$

If the source data $s_i(n)$ are independent random variables, with unit variance, symmetric probability density functions; and with at most one being Gaussian, then

- $$\lim_{n \to \infty} E\left\{\mathbf{C}(n)\right\} = \begin{bmatrix} \pm 1 & 0 \\ 0 & \pm 1 \end{bmatrix} \text{ or } \begin{bmatrix} 0 & \pm 1 \\ \pm 1 & 0 \end{bmatrix}$$
- $$\lim_{n \to \infty} E\{(\mathbf{C}(n)\text{-}E\{\mathbf{C}(\infty)\})^2\} = O(\mu)$$

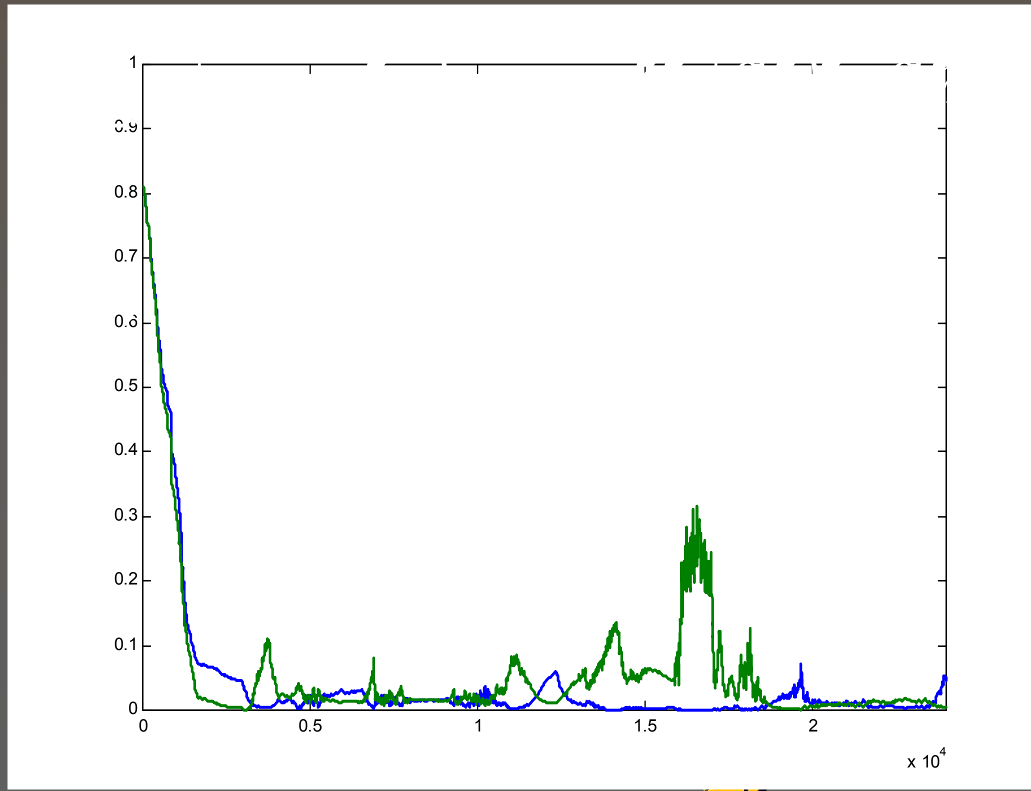
Example revisited



$$\hat{S}(n) = \mathbf{B}(n-1)X(n) = \mathbf{C}(n-1)S(n)$$

$$= \begin{bmatrix} c_{11}(n-1) & c_{12}(n-1) \\ c_{21}(n-1) & c_{22}(n-1) \end{bmatrix} \begin{bmatrix} s_1(n) \\ s_2(n) \end{bmatrix}$$

$$|c_{12}(n)|/|c_{11}(n)| \ll 1, |c_{21}(n)|/|c_{22}(n)| \ll 1$$



General adaptive scheme

$$\hat{S}(n) = \mathbf{B}(n-1)X(n), \qquad X(n) = \mathbf{A}S(n)$$

$$\mathbf{B}(n) = \mathbf{B}(n-1) - \mu\mathbf{H}\left(\hat{S}(n)\right)\mathbf{B}(n-1), \ \mathbf{B}(0) = \mathbf{I}$$

$$\mathbf{H}(z_1, z_2) = \begin{bmatrix} h_{11}(z_1, z_2) & h_{12}(z_1, z_2) \\ h_{21}(z_1, z_2) & h_{22}(z_1, z_2) \end{bmatrix}$$

$$\hat{S}(n) = \mathbf{C}(n-1)S(n), \qquad \mathbf{C}(n) = \mathbf{B}(n)\mathbf{A}$$

$$\mathbf{C}(n) = \mathbf{C}(n-1) - \mu\mathbf{H}\left(\hat{S}(n)\right)\mathbf{C}(n-1), \ \mathbf{C}(0) = \mathbf{A}$$

$$\mathbf{C}(n) = \mathbf{C}(n-1) - \mu\mathbf{H}\left(\mathbf{C}(n-1)S(n)\right)\mathbf{C}(n-1)$$

Imposing the desired limits

From the theory of adaptive algorithms we know that an algorithm of the form

$$\mathbf{C}(n) = \mathbf{C}(n-1) - \mu \mathbf{H}\Big(\mathbf{C}(n-1)S(n)\Big)\mathbf{C}(n-1)$$

converges in the mean $\lim_{n\to\infty} E\{\mathbf{C}(n)\} = \mathbf{C}$, where $\mathbf{C}$ is an equilibrium matrix satisfying the equation

$$\mathbf{C} = \mathbf{C} - \mu E\Big\{\mathbf{H}\Big(\mathbf{C}S(n)\Big)\Big\}\mathbf{C}$$

or equivalently $E\Big\{\mathbf{H}\Big(\mathbf{C}S(n)\Big)\Big\} = 0$

Since

$$\mathbf{H}(z_1, z_2) = \begin{bmatrix} h_{11}(z_1, z_2) & h_{12}(z_1, z_2) \\ h_{21}(z_1, z_2) & h_{22}(z_1, z_2) \end{bmatrix}$$

relation $E\left\{\mathbf{H}\left(\mathbf{C}S(n)\right)\right\} = 0$

is equivalent to a system of four (in general nonlinear) equations

$$E\left\{h_{ij}\left(\mathbf{C}S(n)\right)\right\} = 0, \ i, j = 1, 2$$

in four unknowns (the four elements of matrix $\mathbf{C}$) that can be solved to identify the equilibrium matrices.

If we want a specific matrix $\mathbf{C}_0$ to become an equilibrium, then the elements of $\mathbf{H}(Z)$ must satisfy

$$E\left\{h_{ij}\left(\mathbf{C}_0 S(n)\right)\right\} = 0, \; i, j = 1, 2$$

If in particular we like the following eight matrices

$$\mathbf{C}_0 = \begin{bmatrix} \pm 1 & 0 \\ 0 & \pm 1 \end{bmatrix}, \begin{bmatrix} 0 & \pm 1 \\ \pm 1 & 0 \end{bmatrix}$$

to become equilibriums, then

$$E\left\{h_{ij}\left(\pm s_1, \pm s_2\right)\right\} = E\left\{h_{ij}\left(\pm s_2, \pm s_1\right)\right\} = 0$$

Impose symmetric behavior

If there is a trajectory converging to $\mathbf{I}$ then, with the same probability, we want existence of symmetric trajectories converging to the other seven limits.

$$\mathbf{H}(z_1, z_2) = \begin{bmatrix} h(z_1, z_2) & q(z_2, z_1) \\ q(z_1, z_2) & h(z_2, z_1) \end{bmatrix}$$

$$h(-z_1, z_2) = h(z_1, -z_2) = h(z_1, z_2)$$

$$q(-z_1, z_2) = q(z_1, -z_2) = -q(z_1, z_2)$$

$$h(z_1, z_2) \neq h(z_2, z_1),\ q(z_1, z_2) \neq q(z_2, z_1)$$

$$E\left\{h(s_1, s_2)\right\} = E\left\{h(s_2, s_1)\right\} = 0$$

$$E\left\{q(s_1, s_2)\right\} = E\left\{q(s_2, s_1)\right\} = 0 \text{ (for free)}$$

Five necessary conditions

1. $\mathbf{H}(z_1, z_2) = \begin{bmatrix} h(z_1, z_2) & q(z_2, z_1) \\ q(z_1, z_2) & h(z_2, z_1) \end{bmatrix}$
2. $h(-z_1, z_2) = h(z_1, -z_2) = h(z_1, z_2)$
3. $q(-z_1, z_2) = q(z_1, -z_2) = -q(z_1, z_2)$
4. $E\left\{h(s_1, s_2)\right\} = E\left\{h(s_2, s_1)\right\} = 0$
5. $h(z_1, z_2) \neq h(z_2, z_1),\ q(z_1, z_2) \neq q(z_2, z_1)$

$$\mathbf{H}(z_1, z_2) = \begin{bmatrix} z_1^2 - 1 & z_1 z_2 + g(z_1) z_2 - g(z_2) z_1 \\ z_1 z_2 - g(z_1) z_2 + g(z_2) z_1 & z_2^2 - 1 \end{bmatrix}$$

Example

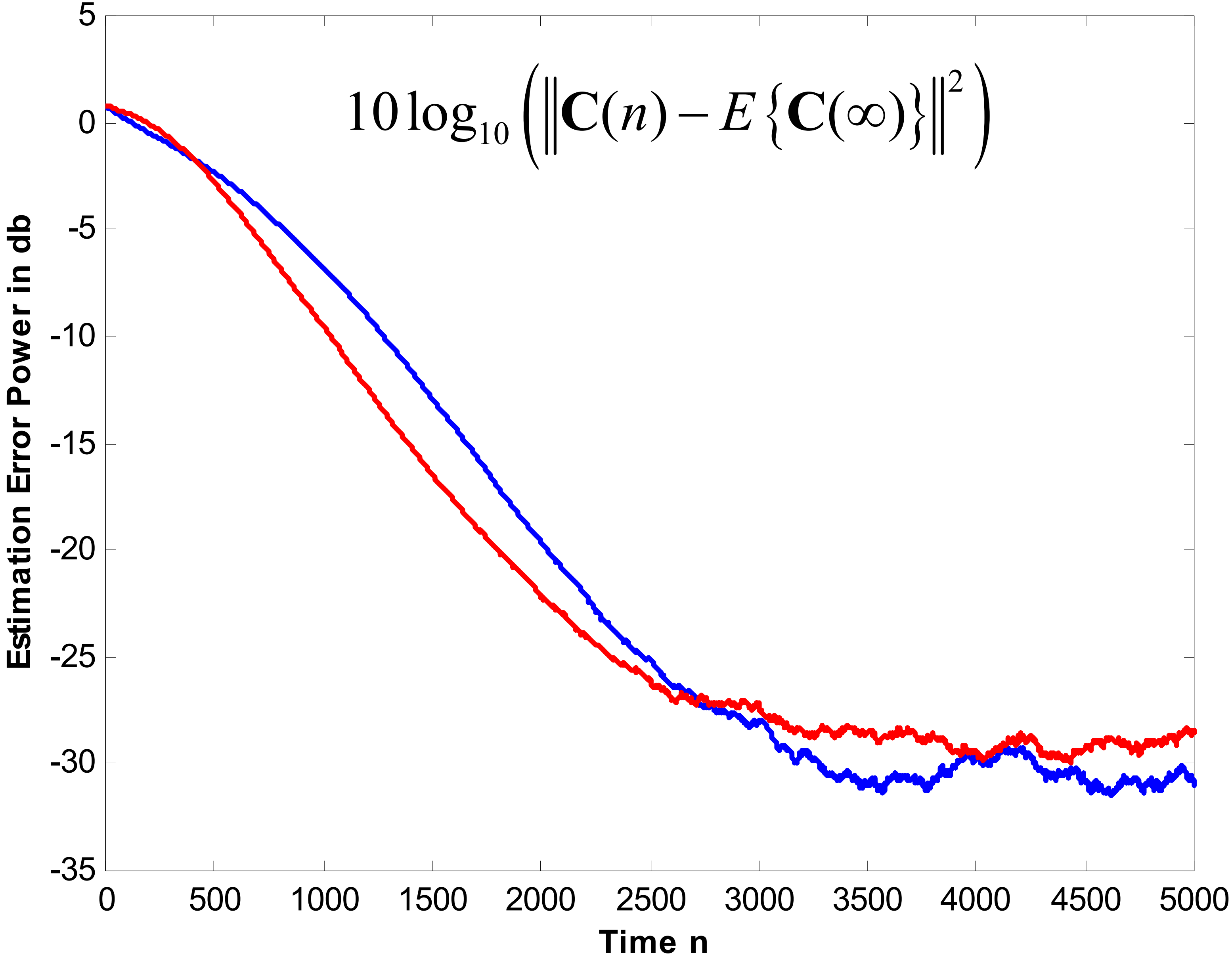
Existing scheme

$$\mathbf{H}(z_1,z_2)=\begin{bmatrix} z_1^2-1 & z_1z_2+g(z_1)z_2-g(z_2)z_1 \\ z_1z_2-g(z_1)z_2+g(z_2)z_1 & z_2^2-1 \end{bmatrix}$$

Select

$$\mathbf{H}(z_1,z_2)=\begin{bmatrix} |z_1|-1 & \dfrac{\operatorname{sgn}(z_2)g(z_1)}{1+z_1^2+z_2^2} \\ \dfrac{\operatorname{sgn}(z_1)g(z_2)}{1+z_1^2+z_2^2} & |z_2|-1 \end{bmatrix}$$

There is no whitening involved!!!



$10\log_{10}\left(\left\|\mathbf{C}(n)-E\{\mathbf{C}(\infty)\}\right\|^2\right)$
Estimation Error Power in db
Time n
5
0
-5
-10
-15
-20
-25
-30
-35
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000

Performance measure

Define a performance measure for the algorithmic class

$$\mathbf{C}(n) = \mathbf{C}(n-1) - \mu \mathbf{H}\Big(\mathbf{C}(n-1)S(n)\Big)\mathbf{C}(n-1)$$

with $\mathbf{H}(z_1, z_2)$ satisfying the five necessary conditions.

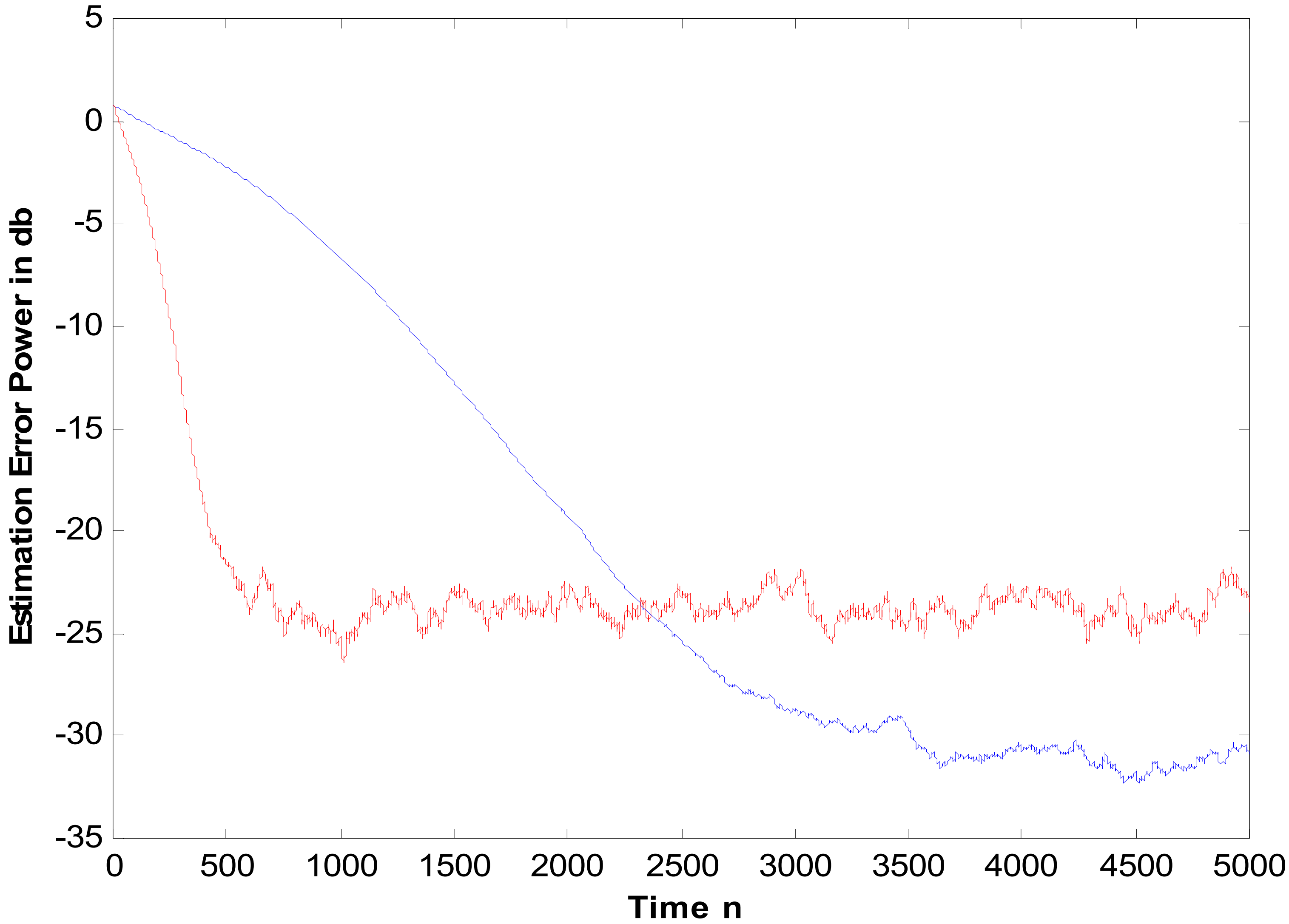
For performance there are two important quantities:

- Rate of convergence
- Steady state estimation error power

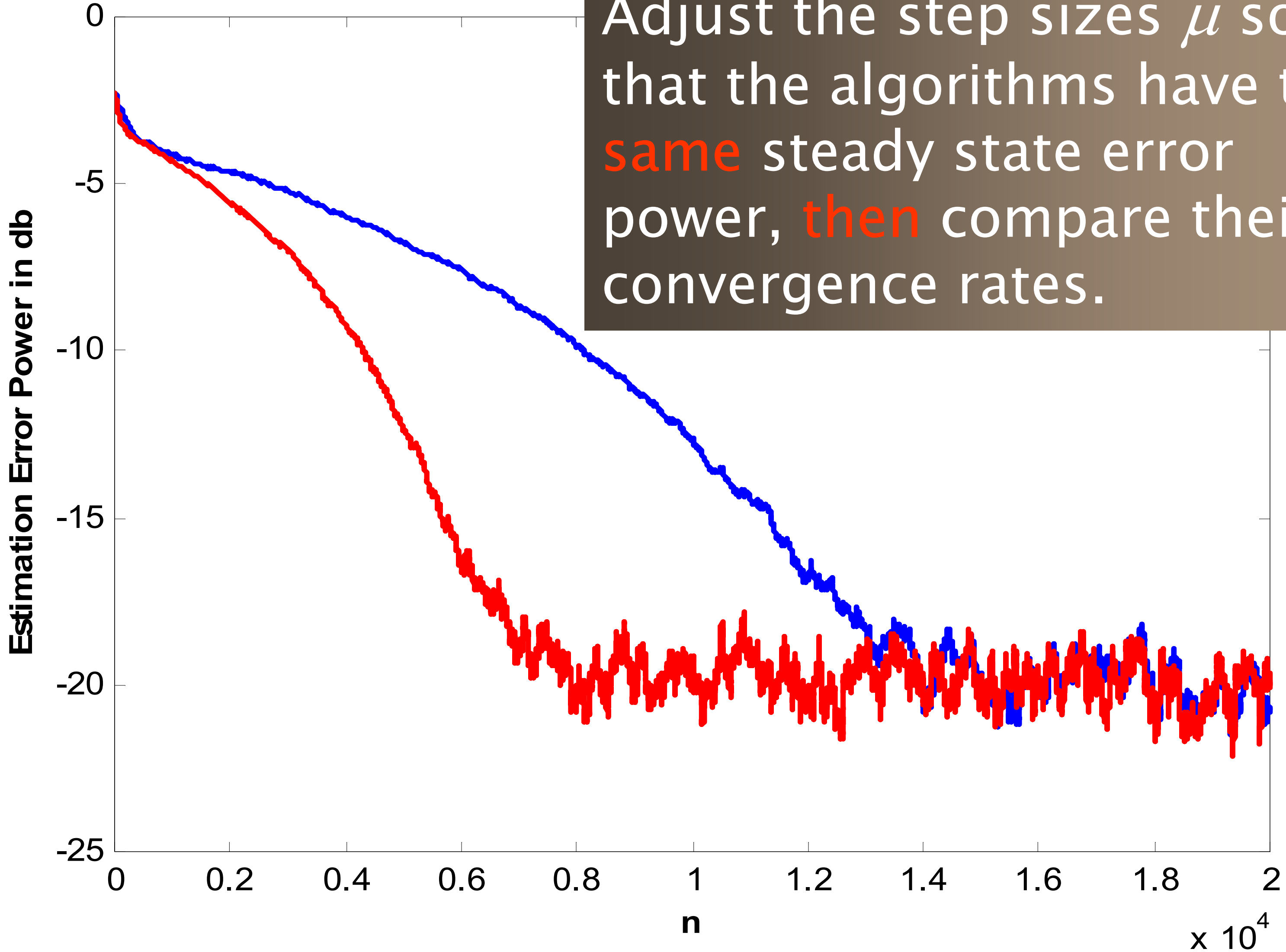
Both quantities depend on μ.

Combine the two quantities to define a fair performance measure.

$$\left\| \mathbf{C}(n) - E\left\{\mathbf{C}(\infty)\right\}\right\|^2 = \sum_{i,j} \left(\mathbf{C}_{ij}(n) - E\left\{\mathbf{C}_{ij}(\infty)\right\}\right)^2$$



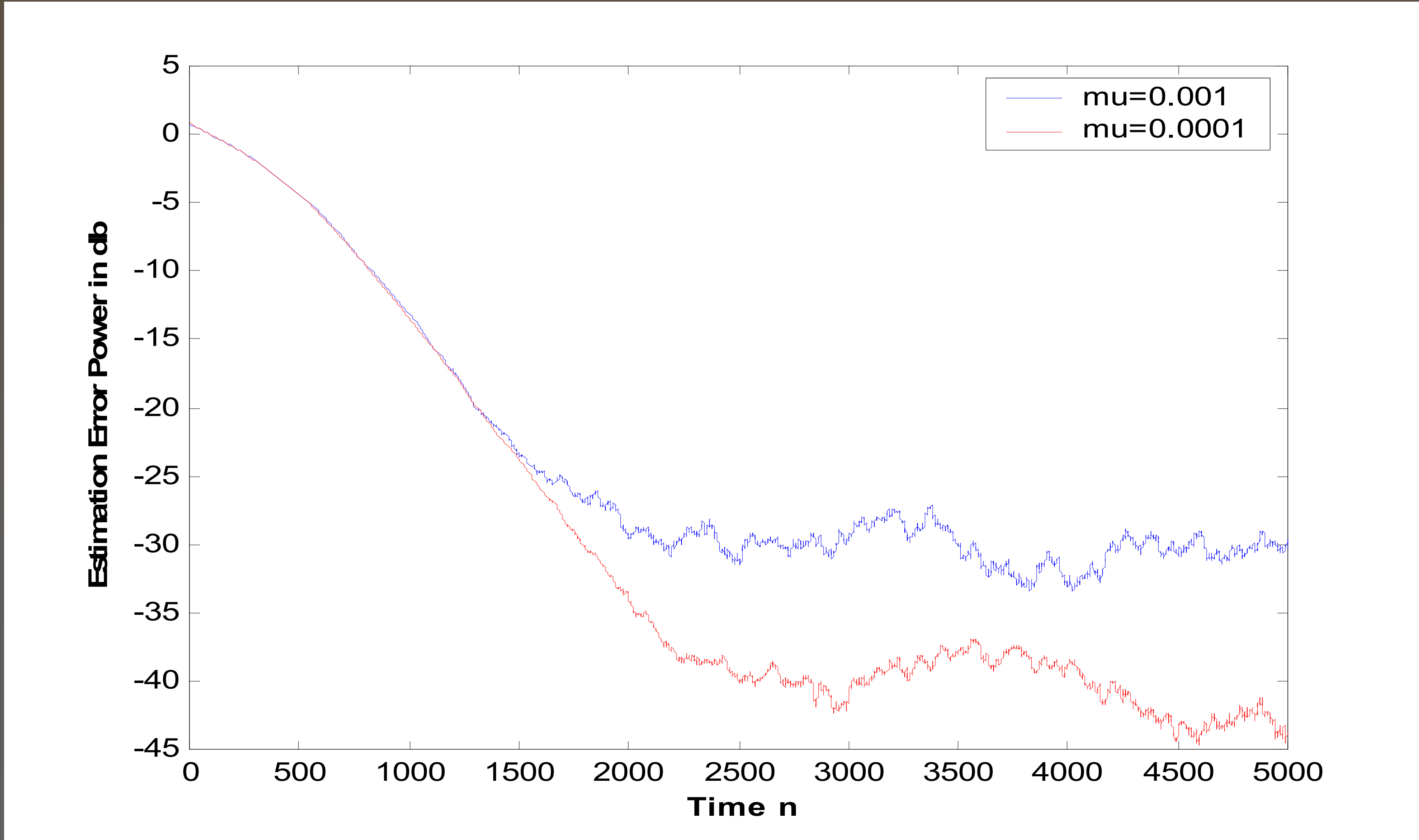
$$\left\|\mathbf{C}(n)-E\{\mathbf{C}(\infty)\}\right\|^2=\sum_{i,j}\left(\mathbf{C}_{ij}(n)-E\{\mathbf{C}_{ij}(\infty)\}\right)^2$$



Adjust the step sizes μ so that the algorithms have the same steady state error power, then compare their convergence rates.

Convergence rate

Although the recursion is nonlinear, in the vicinity of the equilibrium the algorithm exhibits a linear behavior.



$$-\lim_{n\to\infty}\frac{\log\left(\left\|E\left\{\mathbf{C}(n)\right\}-E\left\{\mathbf{C}(\infty)\right\}\right\|\right)}{n}=\mu\min_i\left\{\mathrm{Re}\left(\lambda_i\right)\right\}+o(\mu)$$

λ_i are the eigenvalues of $\mathbf{Q}_1,\mathbf{Q}_2$, where

$$\mathbf{Q}_1=E\left\{\begin{bmatrix}h(s_1,s_2)\\ h(s_2,s_1)\end{bmatrix}\begin{bmatrix}s_1l_1(s_1)-1 & s_2l_2(s_2)-1\end{bmatrix}\right\}$$

$$\mathbf{Q}_2=E\left\{\begin{bmatrix}q(s_1,s_2)\\ q(s_2,s_1)\end{bmatrix}\begin{bmatrix}s_1l_2(s_2) & s_2l_1(s_1)\end{bmatrix}\right\}$$

$$l_i(z)=-\frac{f_i{}'(z)}{f_i(z)},$$ $f_i(z)$ pdfs of the two sources

For (local) stability of the equilibrium: $\min_i\left\{\mathrm{Re}\left(\lambda_i\right)\right\}>0$

Steady state error power

$$\lim_{n\to\infty} E\left\{\left\|\mathbf{C}(n) - E\{\mathbf{C}(\infty)\}\right\|^2\right\} = \mu\,\mathrm{tr}(\mathbf{P}_1 + \mathbf{P}_2) + o(\mu)$$

$$\mathbf{Q}_i\mathbf{P}_i + \mathbf{P}_i\mathbf{Q}_i^t = \mathbf{R}_i,\ i = 1, 2$$

$$\mathbf{R}_1 = E\left\{\begin{bmatrix} h(s_1,s_2) \\ h(s_2,s_1) \end{bmatrix}\begin{bmatrix} h(s_1,s_2) & h(s_2,s_1) \end{bmatrix}\right\}$$

$$\mathbf{R}_2 = E\left\{\begin{bmatrix} q(s_1,s_2) \\ q(s_2,s_1) \end{bmatrix}\begin{bmatrix} q(s_1,s_2) & q(s_2,s_1) \end{bmatrix}\right\}$$

First adjust μ so that the steady state estimation error power is equal to (say) σ^2

$$\mu \, \mathrm{tr}(\mathbf{P}_1+\mathbf{P}_2)=\sigma^2 \quad \Rightarrow \quad \mu = \frac{\sigma^2}{\mathrm{tr}(\mathbf{P}_1+\mathbf{P}_2)}$$

Compute the corresponding convergence rate

$$\mu \min_i \left\{ \mathrm{Re}\left(\lambda_i\right) \right\} = \sigma^2 \frac{\min_i \left\{ \mathrm{Re}\left(\lambda_i\right) \right\}}{\mathrm{tr}(\mathbf{P}_1+\mathbf{P}_2)}$$

Compare the convergence rates or equivalently the Efficacies

$$\mathrm{Eff} = \frac{\min_i \left\{ \mathrm{Re}\left(\lambda_i\right) \right\}}{\mathrm{tr}(\mathbf{P}_1+\mathbf{P}_2)}$$

Larger Efficacy suggests better (faster) algorithm

Optimum adaptive algorithms

The optimum algorithm corresponds to functions $h(z_1,z_2)$, $q(z_1,z_2)$ that maximize the Efficacy for given source pdfs.

When the pdfs are different the optimization problem is still open.

When the sources have the same pdf, the optimum functions are:

$$h(z_1,z_2)=a\left(z_1 l(z_1)-1\right)$$

$$q(z_1,z_2)=bz_1 l(z_2)+cz_2 l(z_1)$$

$$l(z)=-\frac{f'(z)}{f(z)},$$

$f(z)$ common pdf

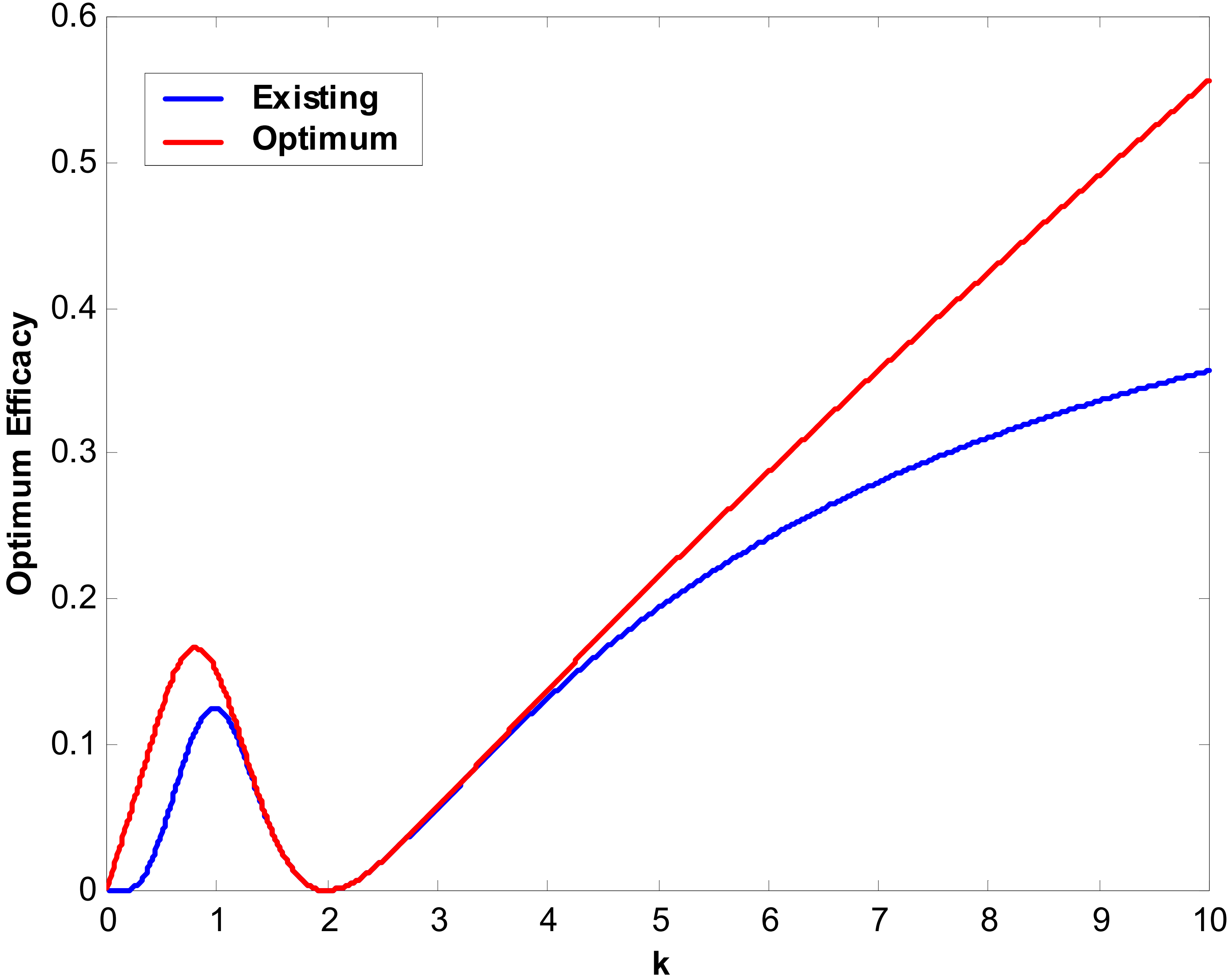
where a, b, c constants so that the two matrices $\mathbf{Q}_1$, $\mathbf{Q}_2$ have all their eigenvalues equal to unity.

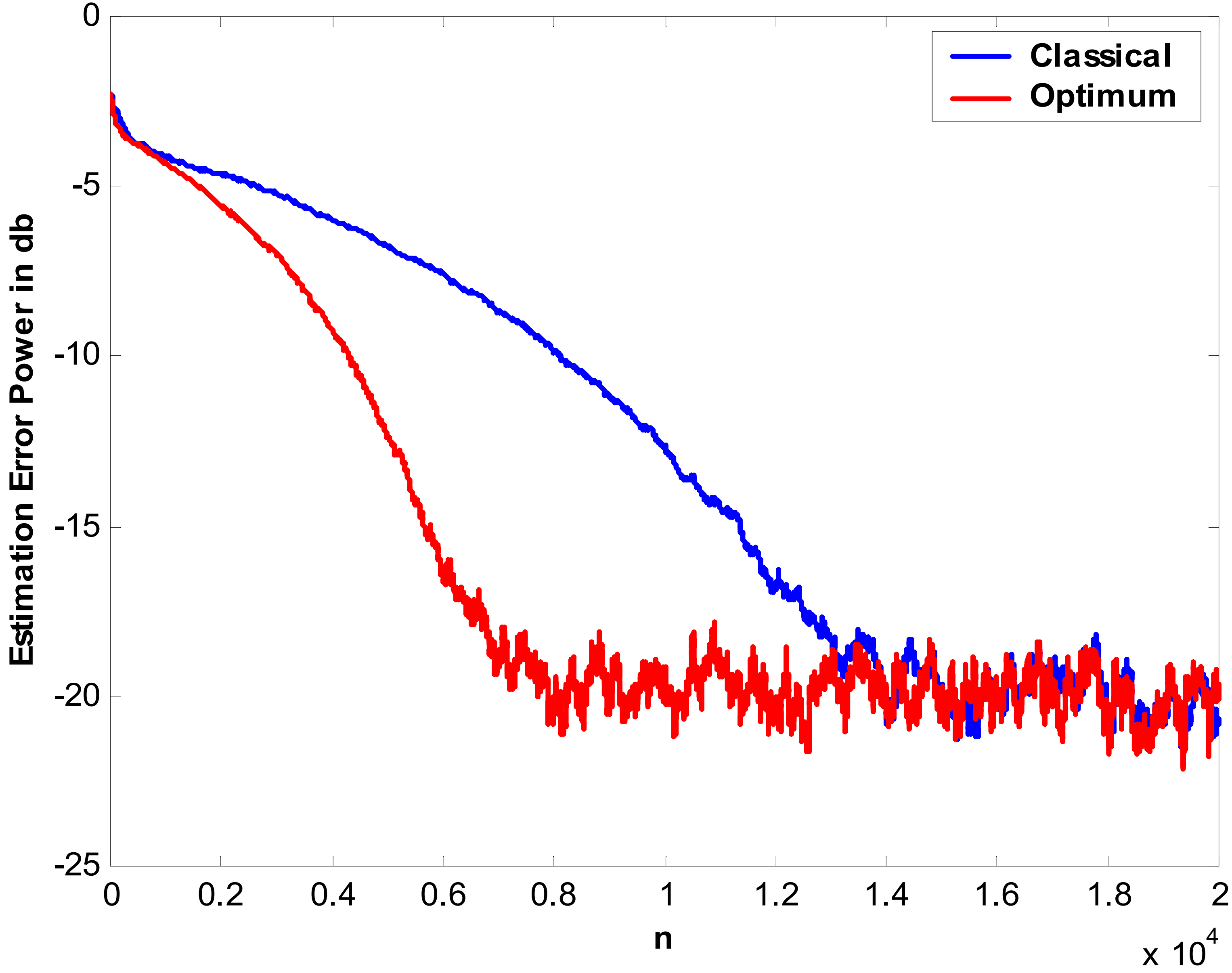
Existing scheme

$$h(z_1,z_2)=z_1^2-1,\qquad q(z_1,z_2)=z_1 z_2+\left(z_1 g(z_2)-z_2 g(z_1)\right)$$

Maximize Efficacy with respect to $g(z)$

$$q(z_1,z_2)=z_1 z_2+d\left(z_1 l(z_2)-z_2 l(z_1)\right)$$


Existing
Optimum
Optimum Efficacy
k
0
0.1
0.2
0.3
0.4
0.5
0.6
0
1
2
3
4
5
6
7
8
9
10


Classical
Optimum
Estimation Error Power in db
n
x 10^4
0
-5
-10
-15
-20
-25
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2

Conclusion

- We have presented a rich class of adaptive algorithms that can solve the blind source separation problem.
- We have introduced an analytic performance measure that allowed us to compare algorithms from our class.
- We have found algorithms that optimize the performance measure for equi-distributed sources.

Questions please?

E n D