“A Robust MMSE Equalizer for MIMO Enhanced CDMA Systems”

Christian Mehlführer

chmehl@nt.tuwien.ac.at

December 7, 2006

INSTITUT FÜR
NACHRICHTENTECHNIK UND
HOCHFREQUENZTECHNIK

TECHNISCHE UNIVERSITÄT WIEN
VIENNA UNIVERSITY OF TECHNOLOGY

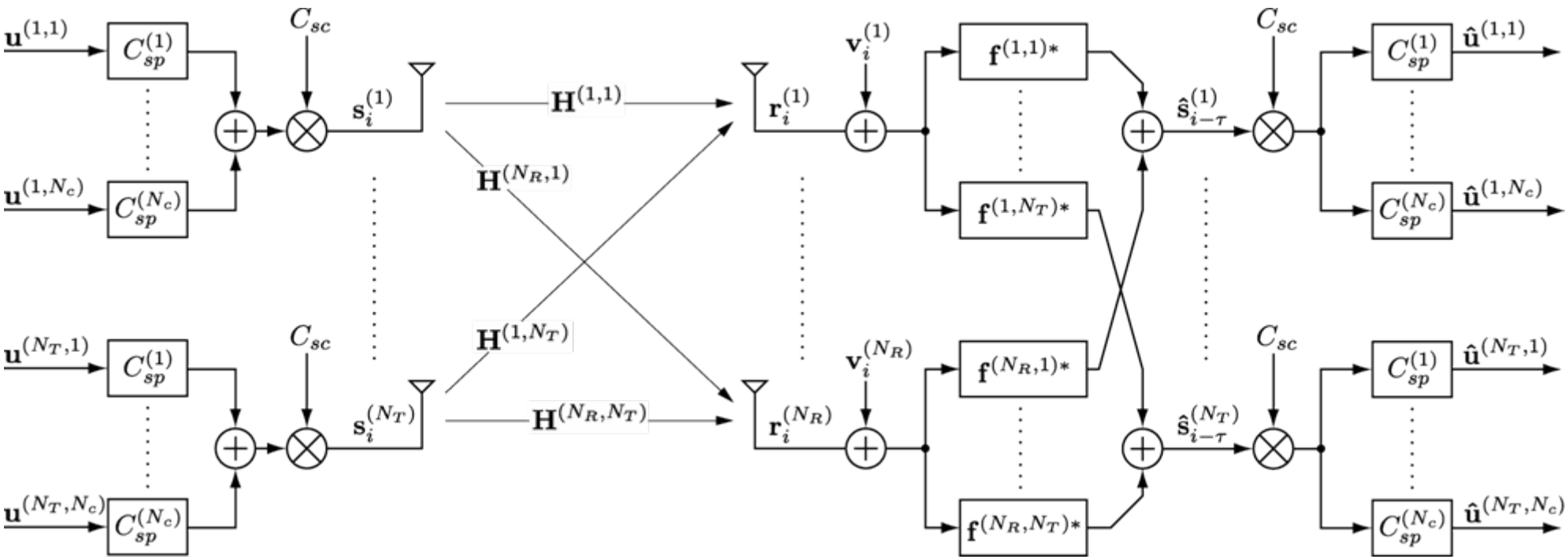
Outline

- Introduction and system model
- MMSE equalizer
- Iterative algorithm
- Simulation results
- Summary

Introduction

- In CDMA systems, users are distinguished by orthogonal spreading codes
- Frequency selective channels cause multiple access interference (MAI)
- MAI gets worse when
  - number of codes is increased
  - number of transmit antennas is increased
  - code length is decreased

System Model



N_c = number of spreading sequences

C_{sp} = spreading sequences

C_{sc} = scrambling sequence

MMSE Equalizer

System model:

$$\mathbf{r}_i = \begin{bmatrix} \mathbf{r}_i^{(1)} \\ \vdots \\ \mathbf{r}_i^{(N_R)} \end{bmatrix} = \begin{bmatrix} \mathbf{H}^{(1,1)} & \dots & \mathbf{H}^{(1,N_T)} \\ \vdots & \ddots & \vdots \\ \mathbf{H}^{(N_R,1)} & \dots & \mathbf{H}^{(N_R,N_T)} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{s}_i^{(1)} \\ \vdots \\ \mathbf{s}_i^{(N_T)} \end{bmatrix} + \mathbf{v}_i = \mathbf{H}\mathbf{s}_i + \mathbf{v}_i$$

Cost function:

$$J\left(\mathbf{f}^{(n_t)}\right) = \mathrm{E}\left\{|\mathbf{f}^{(n_t)H}\mathbf{r}_i - s_{i-\tau}^{(n_t)}|^2\right\}$$

$\rightarrow$ MMSE equalizer:

$$\mathbf{f}^{(n_t)} = \left[\mathbf{f}^{(1,n_t)T}, \dots, \mathbf{f}^{(N_R,n_t)T}\right]^T = \left(\mathbf{H}\mathbf{H}^H + \frac{\sigma_v^2}{\sigma_s^2}\mathbf{I}\right)^{-1} \mathbf{H}\mathbf{e}_\tau^{(n_t)}$$

MMSE Equalizer

$$\mathbf{f}^{(n_t)} = \left[\mathbf{f}^{(1,n_t)T}, \ldots, \mathbf{f}^{(N_R,n_t)T}\right]^T = \left(\mathbf{H}\mathbf{H}^H + \frac{\sigma_v^2}{\sigma_s^2}\mathbf{I}\right)^{-1} \mathbf{H}\mathbf{e}_\tau^{(n_t)}$$

- Direct inversion of the matrix requires divisions and high precision $\rightarrow$ can be a numerical problem in fixed-point implementations
- Idea: calculate matrix inverse iteratively
  - only mult-add operations are required
  - algorithm suitable for fixed-point implementations

Iterative Algorithm

Consider the following iteration:

$$\mathbf{W}_{k+1} = \mathbf{W}_k + \mu_k \left(\mathbf{I} - \mathbf{W}_k \mathbf{A}\mathbf{A}^H \mathbf{W}_k^H\right) \mathbf{W}_k \qquad \mathbf{A}\mathbf{A}^H = \mathbf{H}\mathbf{H}^H + \frac{\sigma_v^2}{\sigma_s^2}\mathbf{I}$$

Singular Value Decomposition $\mathbf{W}_k \mathbf{A} = \mathbf{U} \boldsymbol{\Sigma}_k \mathbf{V}^H$ leads to:

$$\mathbf{W}_{k+1}\mathbf{A} = \mathbf{U}\left[\boldsymbol{\Sigma}_k + \mu_k \left(\mathbf{I} - \boldsymbol{\Sigma}_k \boldsymbol{\Sigma}_k^H\right) \boldsymbol{\Sigma}_k\right] \mathbf{V}^H$$

$$\boldsymbol{\Sigma}_{k+1} = \boldsymbol{\Sigma}_k + \mu_k \left(\mathbf{I} - \boldsymbol{\Sigma}_k \boldsymbol{\Sigma}_k^H\right) \boldsymbol{\Sigma}_k$$

Main diagonal:

$$\sigma_{k+1}^{(i)} = \sigma_k^{(i)} + \mu_k \sigma_k^{(i)} \left(1 - |\sigma_k^{(i)}|^2\right) \qquad i = 1 \ldots N_R L_f$$

Convergence Analysis

Main diagonal:

$$\sigma_{k+1}^{(i)} = \sigma_k^{(i)} + \mu_k \sigma_k^{(i)} \left(1 - |\sigma_k^{(i)}|^2\right) \quad ; i = 1 \ldots N_R L_f$$

The "errors" of the singular values are given by:

$$\begin{aligned} e_{k+1} &= 1 - \sigma_{k+1}^{(i)} = 1 - \sigma_k^{(i)} - \mu_k \sigma_k^{(i)} \left(1 - |\sigma_k^{(i)}|^2\right) = \\ &= \underbrace{\left(1 - \sigma_k^{(i)}\right)}_{e_k} \left(1 - \mu_k \sigma_k^{(i)} \left(1 + \sigma_k^{(i)}\right)\right) \end{aligned}$$

For convergence we require:

$$\frac{|e_{k+1}|}{|e_k|} = \left|1 - \mu_k \sigma_k^{(i)} \left(1 + \sigma_k^{(i)}\right)\right| < 1$$

Step-size Selection

Convergence condition:

$$0 < \mu_k < \frac{2}{\sigma_k^{(\max)}(1+\sigma_k^{(\max)})}$$

Direct calculation of the maximum singular value requires high computational effort!

$\rightarrow$ tight upper bound for the maximum singular value is needed

$$0 < \mu_k < \frac{2}{0.25 + 2\|\mathbf{W}_k\mathbf{A}\mathbf{A}^H\mathbf{W}_k^H\|_1} \leq \frac{2}{\sigma_k^{(\max)}(1+\sigma_k^{(\max)})}$$

How To Invert A Matrix Using The Iterative Algorithm?

Iterative algorithm:

$$\mathbf{W}_{k+1} = \mathbf{W}_k + \mu_k \left(\mathbf{I} - \mathbf{W}_k \mathbf{A}\mathbf{A}^H \mathbf{W}_k^H\right) \mathbf{W}_k \qquad \mathbf{A}\mathbf{A}^H = \mathbf{H}\mathbf{H}^H + \frac{\sigma_v^2}{\sigma_s^2}\mathbf{I}$$

$$\mathbf{W}_0 = \mathbf{I}$$

$$\mu_k = \frac{1.99}{0.25 + 2\|\mathbf{W}_k \mathbf{A}\mathbf{A}^H \mathbf{W}_k^H\|_1}$$

Algorithm converges to: $\lim_{k\to\infty} \mathbf{W}_k \mathbf{A} = \lim_{k\to\infty} \mathbf{U}\mathbf{\Sigma}_k \mathbf{V}^H = \mathbf{U}\mathbf{V}^H$

$$\lim_{k\to\infty} \mathbf{W}_k \mathbf{A}\mathbf{A}^H \mathbf{W}_k^H = \mathbf{U}\mathbf{V}^H \mathbf{V}\mathbf{U}^H = \mathbf{I}$$

$$\lim_{k\to\infty} \mathbf{W}_k^H \mathbf{W}_k = \left(\mathbf{A}\mathbf{A}^H\right)^{-1}$$

Special Cases

- No Iteration, starting value $\mathbf{W}_0 = \mathbf{I}$

$$\mathbf{W}_0^H \mathbf{W}_0 = \mathbf{I} \quad \longrightarrow \quad \mathbf{f}^{(n_t)} = \mathbf{H}\mathbf{e}_\tau^{(n_t)}$$

$\rightarrow$ matched filter

- 1 Iteration, starting value $\mathbf{W}_0 = \mathbf{I}$

$$\mathbf{W}_1 = (1 + \mu_0)\,\mathbf{I} - \mu_0 \left(\mathbf{H}\mathbf{H}^H + \frac{\sigma_v^2}{\sigma_s^2}\mathbf{I} \right)$$

$$\mathbf{f}^{(n_t)} = \mathbf{W}_1^H \mathbf{W}_1 \mathbf{H}\mathbf{e}_\tau^{(n_t)}$$

$\rightarrow$ 2 matrix-vector multiplications

General Case

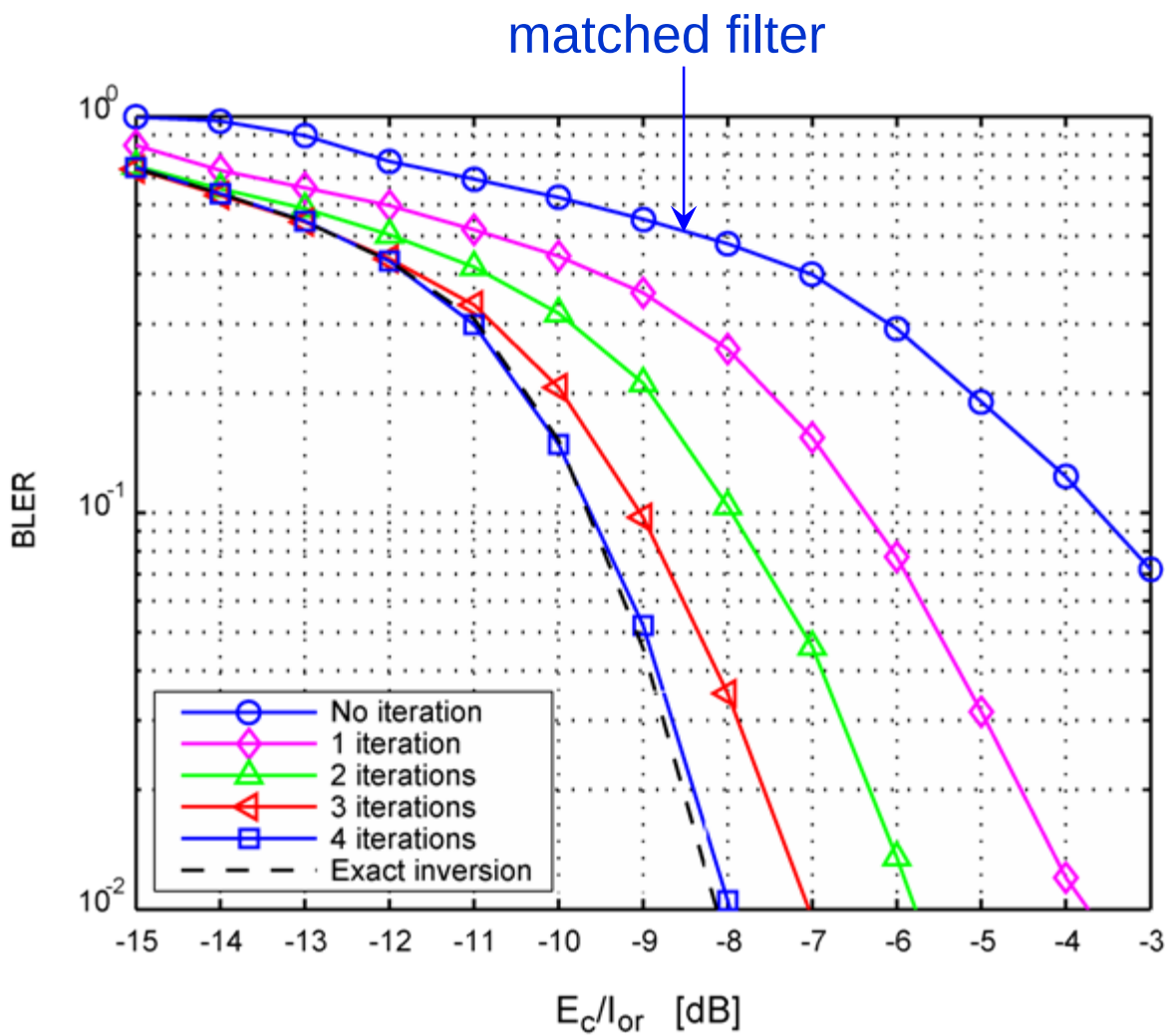
$$\mathbf{W}_{k+1} = \mathbf{W}_k \left[(1+\mu_k)\,\mathbf{I} - \mu_k \left(\mathbf{H}\mathbf{H}^H + \frac{\sigma_v^2}{\sigma_s^2}\mathbf{I} \right) \mathbf{W}_k^H \mathbf{W}_k \right]$$

$$\mathbf{f}^{(n_t)} = \mathbf{W}_{k+1}^H \mathbf{W}_{k+1} \mathbf{H} \mathbf{e}_\tau^{(n_t)}$$

$\rightarrow$ complexity roughly 2.5 matrix-matrix multiplications per iteration

once $\mathbf{W}$ is known, the equalizers for all streams are known

2x2 Low Channel Coding Rate



Simulation parameters:

Code rate: 0.2

10 codes of length 16

Pedestrian B channel model

Block fading

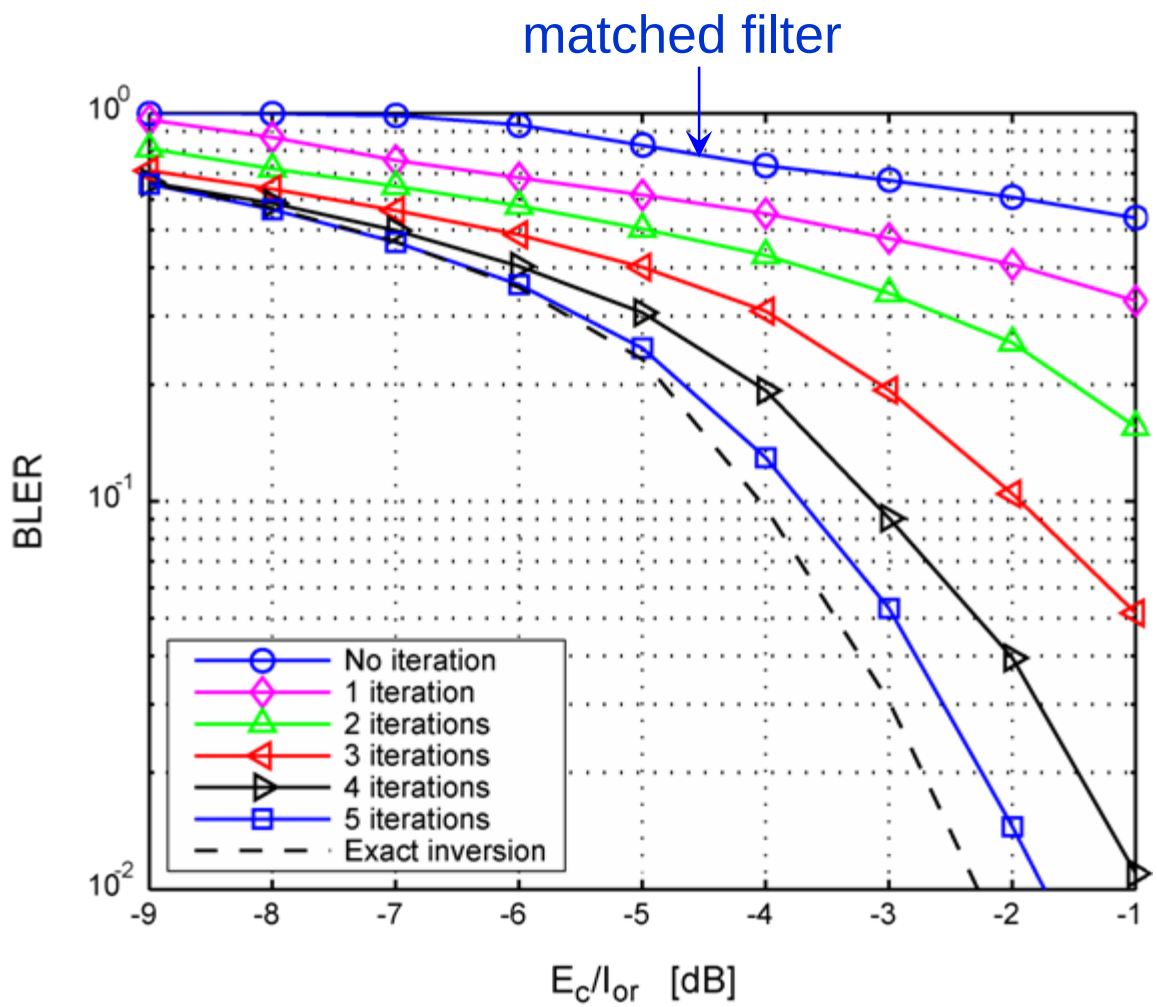
L_h = 15 chips

$L_f = 5L_h$

Perfect channel knowledge

Perfect noise knowledge

2x2 High Channel Coding Rate



Simulation parameters:

Code rate: 0.7

10 codes of length 16

Pedestrian B channel model

Block fading

L_h = 15 chips

$L_f = 5L_h$

Perfect channel knowledge

Perfect noise knowledge

Simulation Results

SNR loss in dB at BLER=0.1

Iteration No.	0	1	2	3	4	5	6	7
1x1, R=0.2	4.0	0.9	0.1	-	-	-	-	-
1x1, R=0.7	N/A	2.3	1.0	0.3	-	-	-	-
2x2, R=0.2	6.0	3.3	1.7	0.6	-	-	-	-
2x2, R=0.7	N/A	N/A	N/A	2.1	0.9	0.3	-	-
4x4, R=0.2	5.9	4.1	2.7	1.6	0.7	0.2	-	-
4x4, R=0.7	N/A	N/A	N/A	N/A	2.6	1.5	0.8	0.3

Summary

- Robust iterative algorithm for matrix inversion was presented
- Algorithm requires only mult-add operations
- Number of required iterations increases with order of the MIMO system and the code rate
- Complexity reductions for time-variant channels are expected

Thank you for your attention!