Brigham Young University
Electrical & Computer Engineering

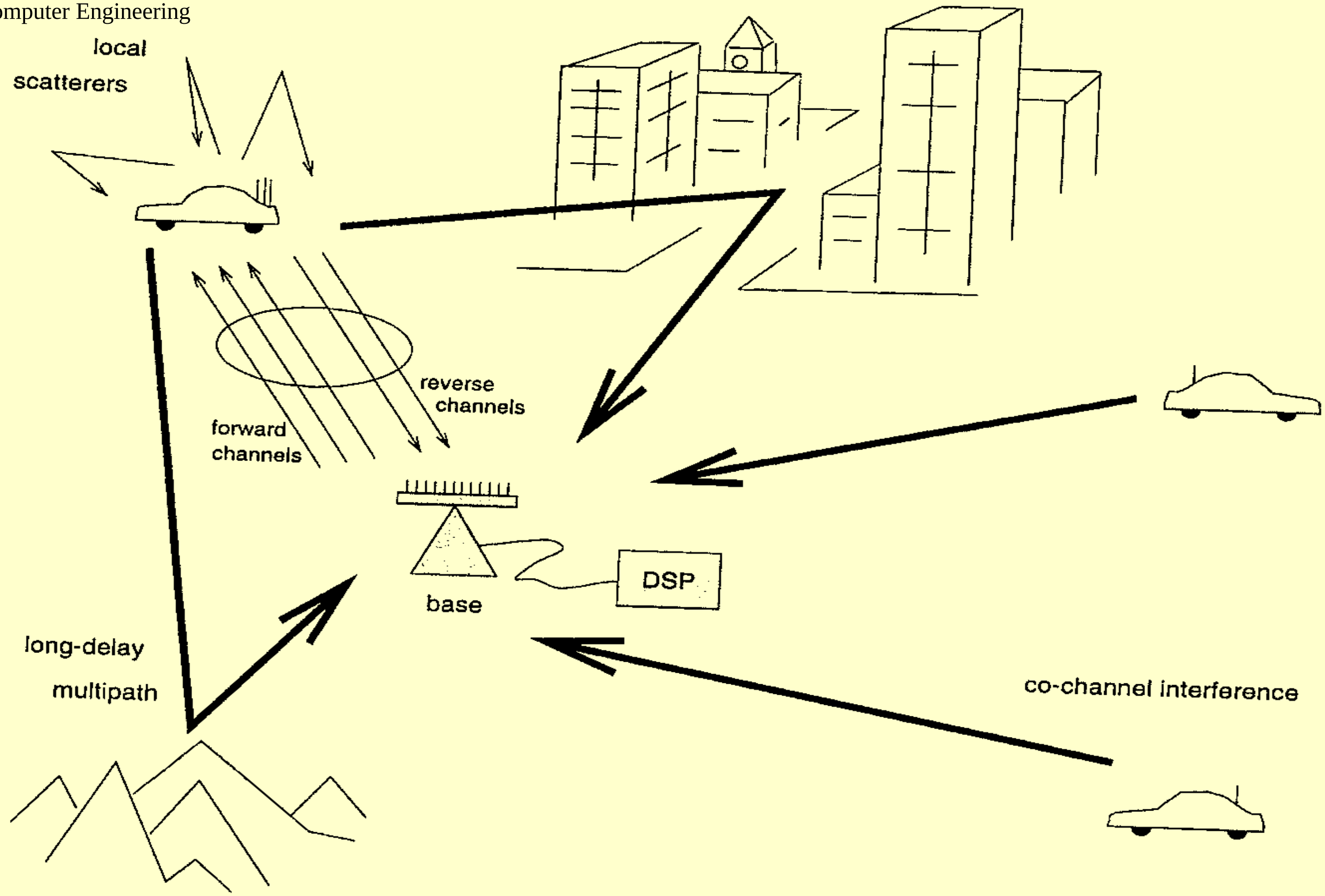
Simultaneous Source Decoding and Blind Equalization of Multipath Fading Channels

A. Lee Swindlehurst & Jake Gunther

Dept of Electrical & Computer Engineering
Brigham Young University
Provo, UTAH

Typical Scenario

Brigham Young University
Electrical & Computer Engineering



Brigham Young University
Electrical & Computer Engineering

Problem Statement

Recover desired user's signal (bits) in the presence of:

- noise
- co-channel interference
- multipath
  - delay, angle, doppler spread

Our focus: frequency selective fading
unstructured channel model
negligible doppler spread
no training data (blind)
exploit signal alphabet & coding

Standard Approach

(1) Estimate Channel

- training sequence
- blind (HOS, CMA, subspace, etc.)

(2) Equalize Channel

- MMSE
- Zero-forcing filter
- ML sequence estimation (Viterbi)

Drawbacks

- channel estimation & equalization performed in separate steps
- blind channel estimation algorithms do not exploit signal alphabet nor coding
- Viterbi is very expensive if both channel and coding memory are exploited
- block channel estimation & equalization methods not suited for time-varying scenarios

New Approach

- combines equalization & channel estimation into a single step
- channel estimate need not be explicitly calculated
- exploits known signal alphabet and coding
- recursive solution allows tracking of time-varying channels

FIR Channel Model

$$x_n^i = \begin{bmatrix} h_0^i & h_1^i & \cdots & h_{L-1}^i \end{bmatrix} \begin{bmatrix} s_n \\ s_{n-1} \\ \vdots \\ s_{n-L+1} \end{bmatrix}$$

$$= \boldsymbol{h}^i \boldsymbol{s}_L(n)$$

x_n^i = output of channel i at sample n

s_n = n^{th} transmitte d symbol

h_k^i = k^{th} tap of FIR channel i

Vector Channel Model

Assume we have *M* such channels:

$$\mathbf{x}(n) = \begin{bmatrix} x_n^1 \\ \vdots \\ x_n^M \end{bmatrix} = \begin{bmatrix} \mathbf{h}^1 \\ \vdots \\ \mathbf{h}^M \end{bmatrix} \mathbf{s}_L(n)$$

$$= \mathbf{H} \, \mathbf{s}_L(n)$$

- $\mathbf{H}$ is an *MxL* channel matrix
- all channels need not have maximum length *L*
- we assume that $\mathbf{H}$ is full rank

An Alternative Model

Brigham Young University
Electrical & Computer Engineering

FIR model not necessarily always the “best”; can use structure based on physical parameters:

$$\boldsymbol{x}(n) = \sum_{k=1}^{K} \boldsymbol{a}(\theta_k)\, s(n - \tau_k)$$

$$\tau_k = \text{time delay of } k^{th} \text{ multipath arrival}$$

$$\theta_k = \text{spatial parameters (DOA, pol, etc)}$$

$$\boldsymbol{a}(\theta_k) = \text{array response to arrival } k$$

Brigham Young University
Electrical & Computer Engineering

Stack to Create Low-Rank Subspace

$$\boldsymbol{\chi}(n) = \begin{bmatrix} \boldsymbol{x}(n) \\ \boldsymbol{x}(n-1) \\ \vdots \\ \boldsymbol{x}(n-P) \end{bmatrix} = \begin{bmatrix} \boldsymbol{H} & & & \\ & \boldsymbol{H} & & \\ & & \ddots & \\ & & & \boldsymbol{H} \end{bmatrix} \boldsymbol{s}_{L+P}(n)$$

$$= \boldsymbol{H} \quad \boldsymbol{s}_{L+P}(n)$$

- choose P so that $\boldsymbol{H}$ is "tall"
- key idea: if $\boldsymbol{H}$ is full rank, then *span(**H**)* completely determines $\boldsymbol{H}$

Brigham Young University
Electrical & Computer Engineering

Standard Approach Revisited

(1) Collect enough data

$$\boldsymbol{X}(n) = [\chi(n) \quad \chi(n-1) \quad \cdots \quad \chi(n-N)]$$

so that *span(**H**)* can be estimated.

(2) Determine ***H*** from *span(**H**)*

(3) Use Viterbi with branch metrics

$$\left\| \boldsymbol{x}(n) - \hat{\boldsymbol{H}}\,\hat{s}_L(n) \right\|^2$$

to exploit known signal alphabet

Brigham Young University
Electrical & Computer Engineering

An Alternative View: Row Subspaces

$$\begin{aligned} \boldsymbol{X}(n) &= \left[\chi(n) \quad \chi(n-1) \quad \cdots \quad \chi(n-N)\right] \\ &= \boldsymbol{H}\left[\boldsymbol{s}(n) \quad \boldsymbol{s}(n-1) \quad \cdots \quad \boldsymbol{s}(n-N)\right] \\ &= \boldsymbol{H}\,\boldsymbol{S}(n) \end{aligned}$$

- Note that $\boldsymbol{S}(n)$ is a Hankel matrix
- Let $\boldsymbol{G}(n)$ be a basis for $null(\boldsymbol{X}(n))$:

$$\boldsymbol{X}(n)\,\boldsymbol{G}(n) = 0$$

- $\boldsymbol{H}$ full rank => $\boldsymbol{S}(n)\,\boldsymbol{G}(n) = 0$ as well

Brigham Young University
Electrical & Computer Engineering

A Question

- From $span(\boldsymbol{H})$ we can get the channel $\boldsymbol{H}$ (which is used to find $\boldsymbol{S}(n)$ by Viterbi, etc)
- Since $\boldsymbol{S}(n)\,\boldsymbol{G}(n) = 0$, can we get $\boldsymbol{S}(n)$ from $span(\boldsymbol{G})$?

YES, but only an <u>unstructured</u> $\boldsymbol{S}(n)$

NO, if we want signal alphabet & coding structure

Exploiting Hankel Structure

Each of the following equations has a row in common :

$$
\begin{array}{ll}
\boldsymbol{S}(n)\boldsymbol{G}(n)=0 & \text{row } 1 \\
\boldsymbol{S}(n-1)\boldsymbol{G}(n-1)=0 & \text{row } 2 \\
\vdots & \vdots \\
\boldsymbol{S}(n-Q+1)\boldsymbol{G}(n-Q+1)=0 & \text{row Q}
\end{array}
$$

call this row

$$\boldsymbol{b}(n)=\begin{bmatrix} s_n & s_{n-1} & \cdots & s_{n-N} \end{bmatrix}$$

then

$$\boldsymbol{b}(n)\boldsymbol{G}(n)=\boldsymbol{b}(n)\begin{bmatrix}\boldsymbol{G}(n) & \boldsymbol{G}(n-1) & \cdots & \boldsymbol{G}(n-Q+1)\end{bmatrix}=0$$

Brigham Young University
Electrical & Computer Engineering

Some Observations

- With d co-channel users, $\boldsymbol{b}(n)$ has d rows instead of one.
- If $\boldsymbol{H}$ is full rank, and Q large enough so that $\boldsymbol{G}(n)$ is “fat”, there is only one solution $\boldsymbol{b}(n)$
- *Idea:* Find vector in nullspace of $\boldsymbol{G}^T(n)$ with elements from signal alphabet and coding constrained.

A Recursive Solution

- Set $\hat{\boldsymbol{b}}(n) = [s_n \quad \hat{s}_{n-1} \quad \cdots \quad \hat{s}_{n-N}]$ and solve for s_n by minimizing $\gamma(n) = \left\| \hat{\boldsymbol{b}}(n)\boldsymbol{G}(n) \right\|^2$
- Use rank revealing URV for fast $\boldsymbol{G}(n)$ update
- Insensitive to imprecisely known channel length (structure of $\boldsymbol{H}$ not explicitly used)
- Use Viterbi with branch metric $\gamma(n)$; trellis only requires memory for coding depth
- Solution separates for multiple users: single trellis per user

Example 1

Brigham Young University
Electrical & Computer Engineering

random channel
$L = 5$

4 element array

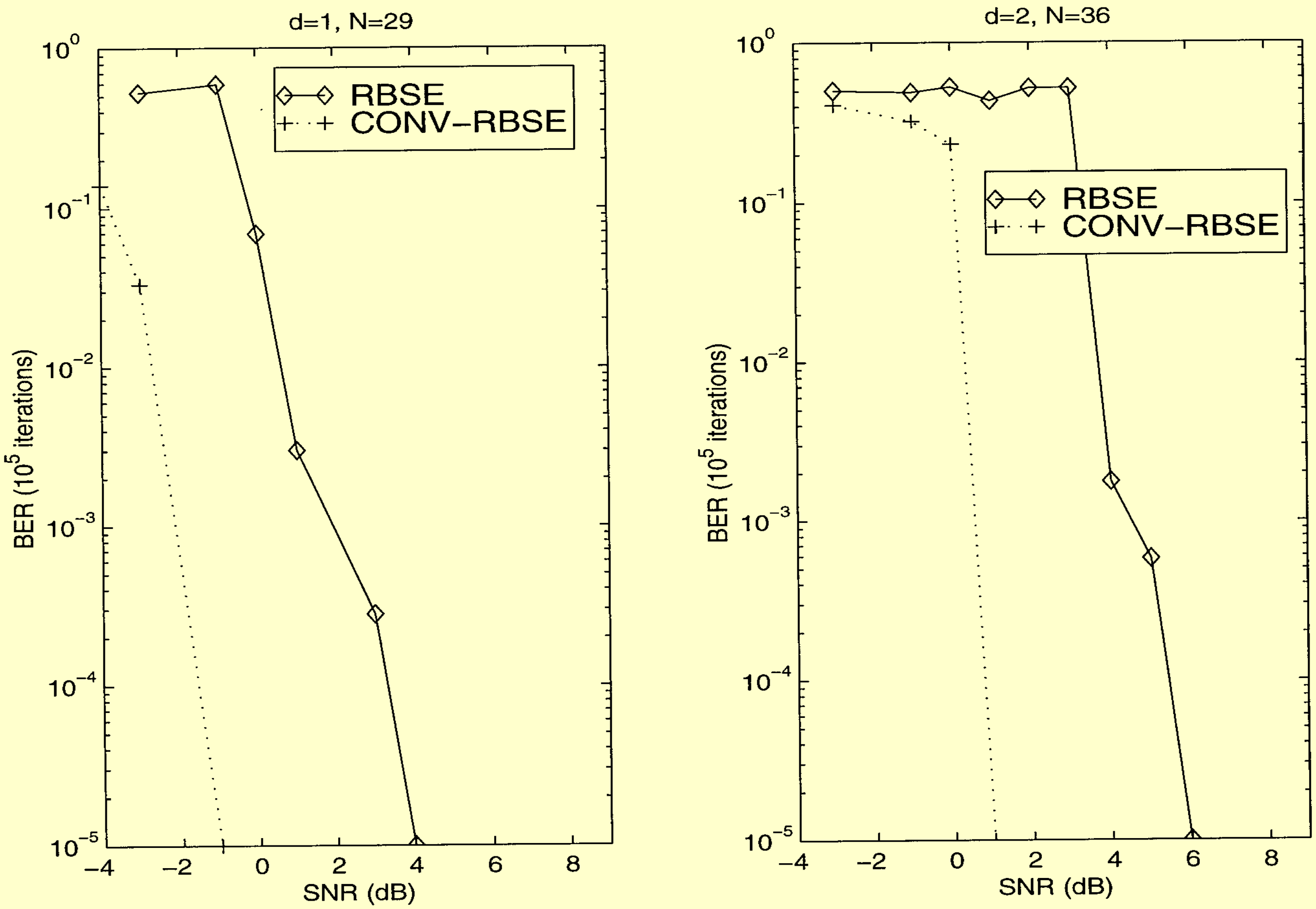
2 users

oversample by 2
$M = 8$

BPSK signals

rate 1/2 conv. code, depth 3

stacking factors
$P = 2, Q = 6$



Brigham Young University
Electrical & Computer Engineering

Example 2

random channel
$L = 5$

4 element array

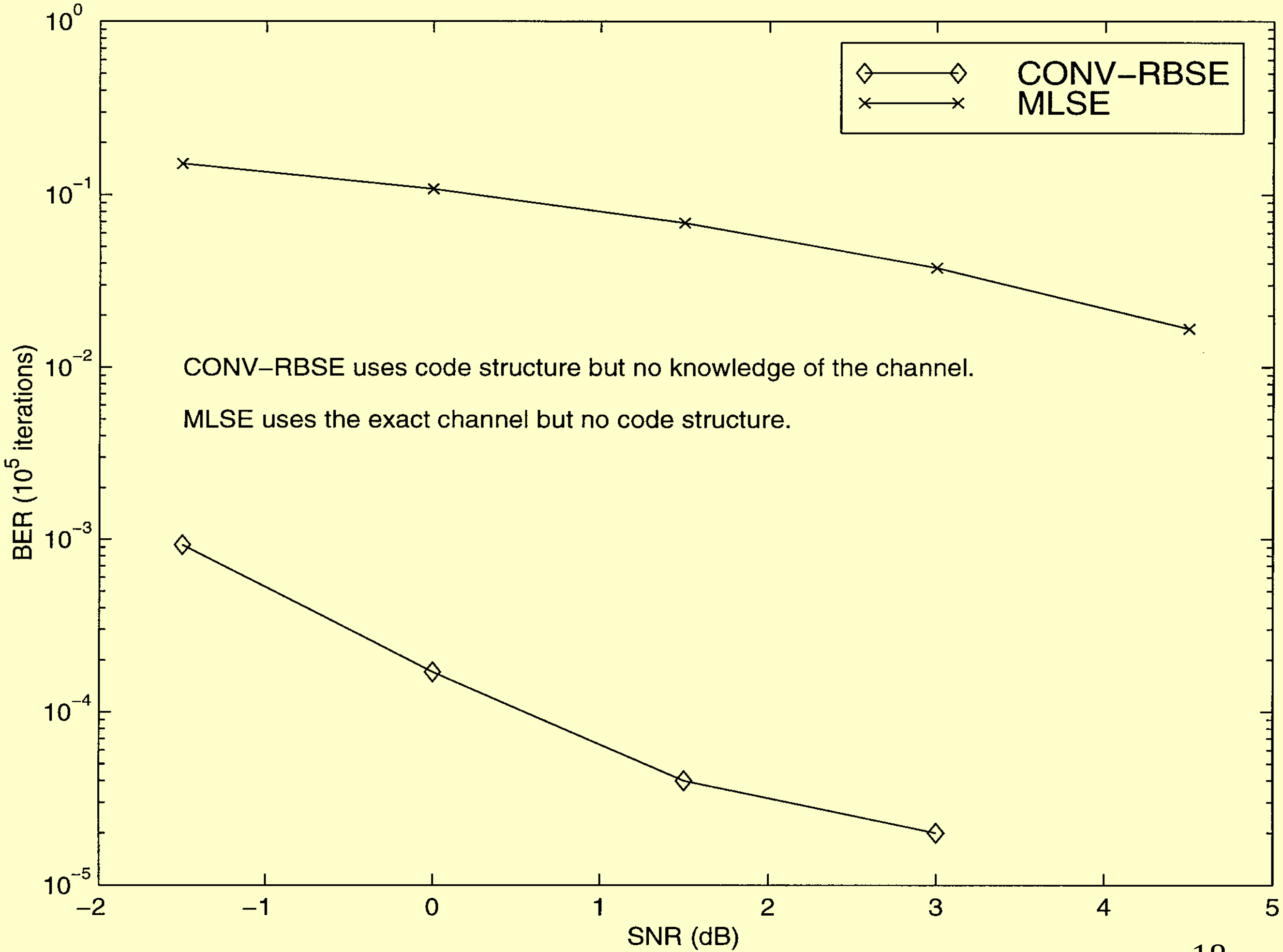
1 user

oversample by 2
$M = 8$

BPSK signals

rate 1/2 conv. code, depth 3

stacking factors
$P = 3, Q = 7$



Brigham Young University
Electrical & Computer Engineering

Summary

- Standard approaches implement channel estimation, equalization, sequence estimation separately
- New algorithm combines all into single step
- Recursive solution allows for:
  - exploitation of finite signal alphabet
  - exploitation of coding structure
  - tracking of time-varying channels
- Single Viterbi trellis per user only requires enough memory to cover coding depth