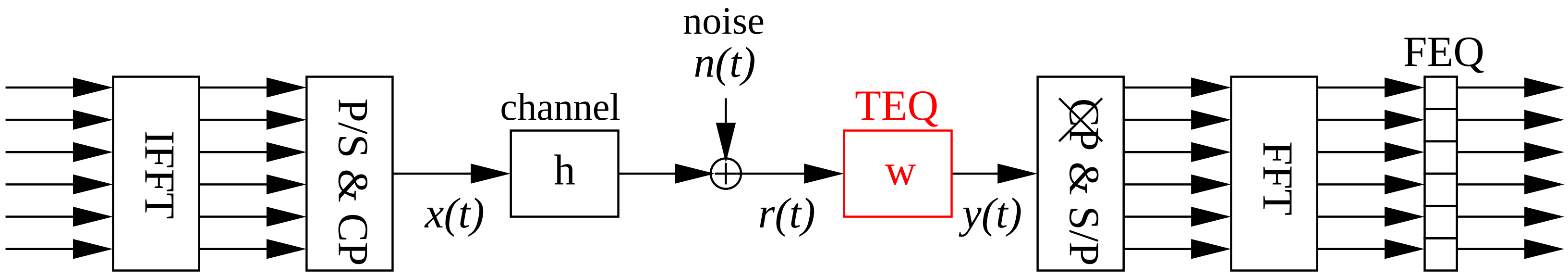
Low Complexity MIMO Blind Adaptive Channel Shortening

Richard K. Martin, John M. Walsh, and C. Richard Johnson, Jr.

The IEEE Int. Conf. on Acoustics, Speech, and Signal Processing
Montreal, Quebec, Canada
May 21, 2004

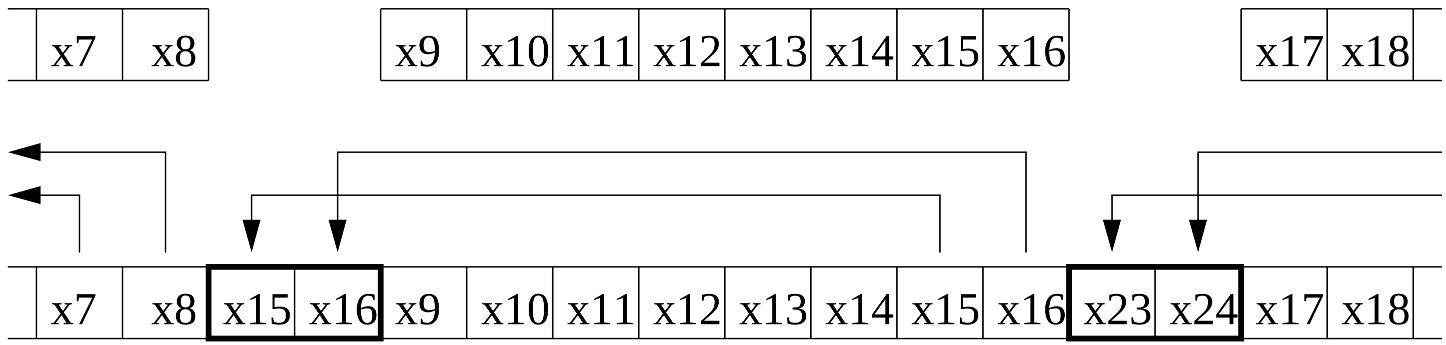
The authors are supported in part by NSF Grant CCR-0310023, Applied Signal Technology, Texas Instruments, and the Olin Fellowship from Cornell University.

Multicarrier system model



- Analog viewpoint: signal is divided into bins. Bin i is modulated by a carrier at frequency $i \cdot f_c$, where $0 \le i \le N-1$.
- Digital viewpoint: efficient implementation by IFFT.
- Goal: change frequency selective channel into N parallel flat fading channels.
- A bank of complex scalars, a Frequency-domain Equalizer, (FEQ) inverts the flat fades.

What is the cyclic prefix (CP)?



- Before each block is transmitted, the CP is prepended.
- $\mathbf{H}$ is the channel convolution matrix ($\mathbf{y} = \mathbf{H}\mathbf{x}$)
- If we add the CP, $\mathbf{H} \rightarrow \widehat{\mathbf{H}}$, a circulant matrix.
- Fact: DFT matrices diagonalize any circulant matrix.

$$\mathcal{F} \; \widehat{\mathbf{H}} \; \mathcal{F}^H = \Lambda = \text{diagonal}$$

- All of this is only true if the CP+1 is as long as the channel.
- CP length is fixed. Sometimes the channel is longer.

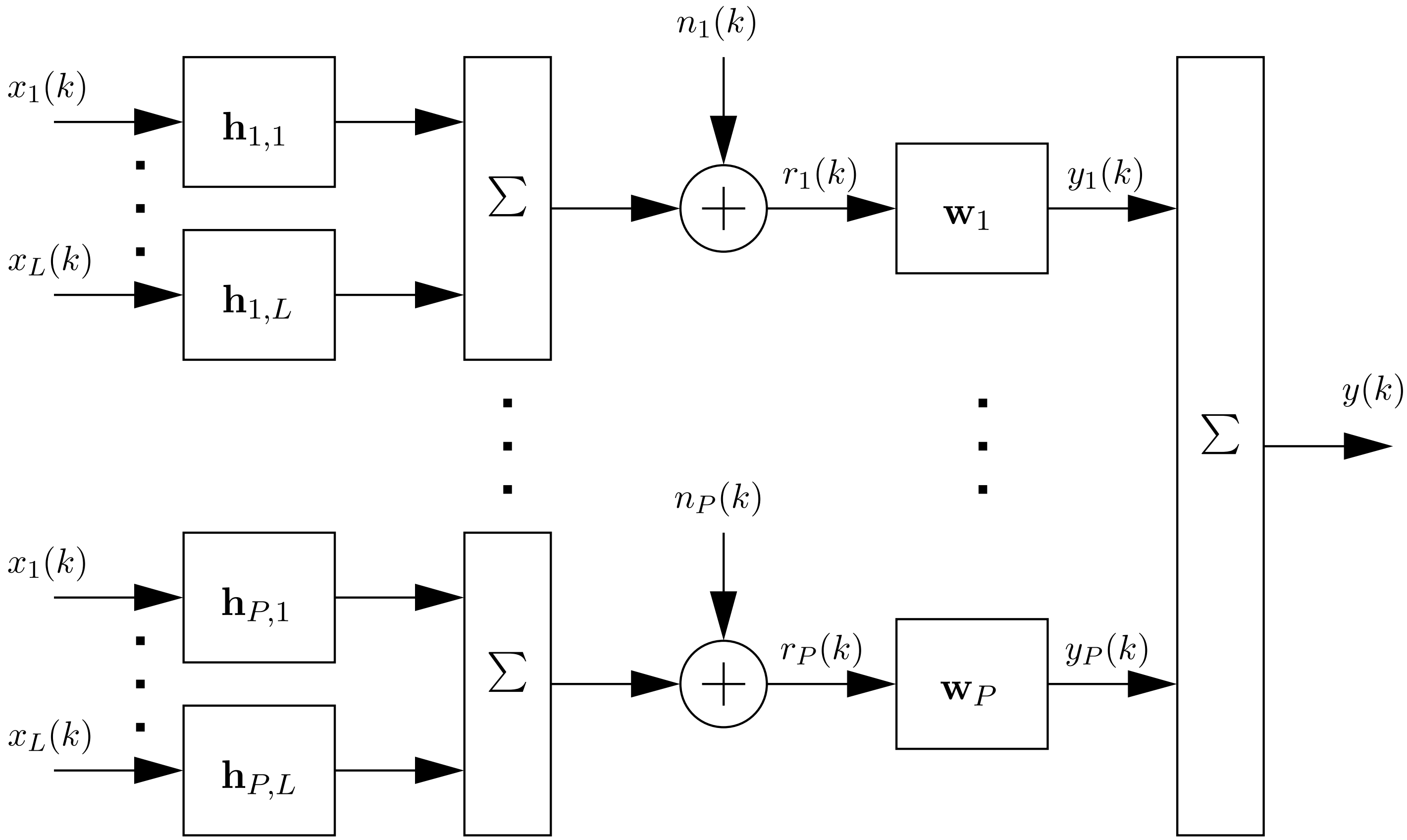
Parameter selection guidelines

1. The block size must be short enough to minimize the channel's time variations within each symbol. (Keep N small.)
2. The guard interval must be long enough to exceed the delay spread of most of the channels that will be encountered. (Keep ν large.)
3. The throughput loss due to the use of the guard interval must be kept as small as possible. (Keep ν much smaller than N.)

These guidelines cannot always be satisfied.

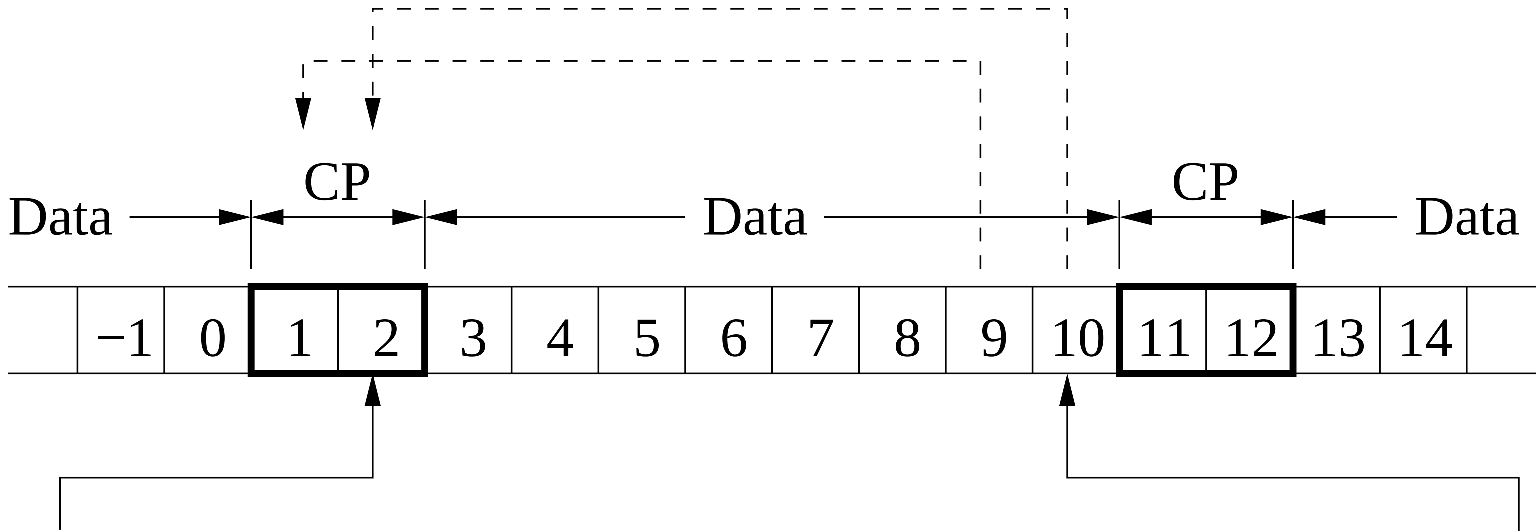
Solution: ignore guideline #2, and use a channel shortening filter.

MIMO channel shortening model



- L users or L transmit antennas
- P receive antennas or oversampling by P

MERRY: some intuition



$$
\begin{aligned}
y(2) &= c_0x(2) + c_1x(1) + c_2x(0) + c_3x(-1) + c_4x(-2) \\
&= c_0x(10) + c_1x(9) + [c_2x(0) + c_3x(-1) + c_4x(-2)] \\
y(10) &= c_0x(10) + c_1x(9) + [c_2x(8) + c_3x(7) + c_4x(6)]
\end{aligned}
$$

- The transmitted data in the CP and the end of the symbol are equal.
- If the channel is short enough, then the last sample in the received CP equals the last sample in the received symbol.

The MERRY algorithm

- Idea: try to force a particular sample in the CP to equal the corresponding sample in the symbol.
- Algorithm: perform a (stochastic) gradient descent of

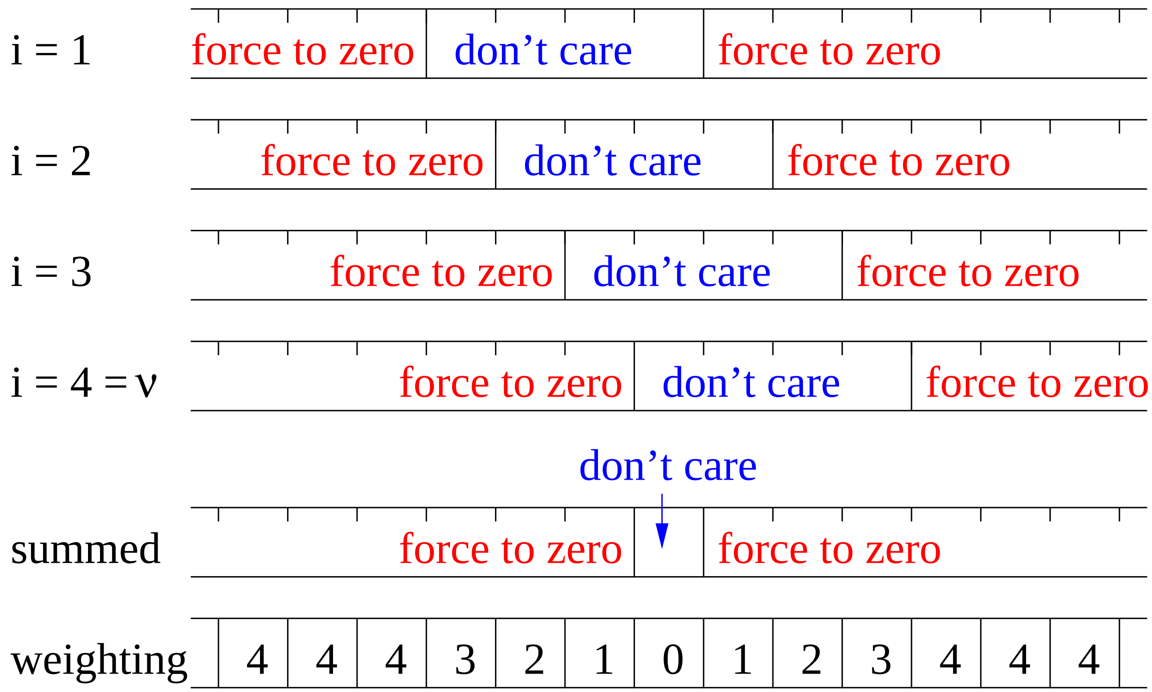
$$J_{MERRY} = \mathrm{E}\left[|y(\nu + \Delta) - y(\nu + N + \Delta)|^2 \right], \tag{1}$$

for some Δ in the set $\{0, \ldots, M-1\}$, where M is the symbol length.

- A constraint is needed to prevent $\mathbf{w} = \mathbf{0}$.
- Multicarrier Equalization by Restoration of RedundancY (MERRY):

$$\boxed{\begin{aligned} \tilde{\mathbf{r}}(k) &= \mathbf{r}(Mk + \nu + \Delta) - \mathbf{r}(Mk + \nu + N + \Delta) \\ e(k) &= \mathbf{w}^T(k)\ \tilde{\mathbf{r}}(k) \\ \hat{\mathbf{w}}(k+1) &= \mathbf{w}(k) - \mu\ e(k)\ \tilde{\mathbf{r}}^*(k) \\ \mathbf{w}(k+1) &= \frac{\hat{\mathbf{w}}(k+1)}{\|\hat{\mathbf{w}}(k+1)\|_2} \end{aligned}} \tag{2}$$

The FRODO cost function



- Why not form more than one "cyclic difference" in MERRY?
- Result: variable window size, formed by an intersection of up to ν different MERRY windows
- FRODO: Forced Redundancy with Optional Data Omission

Division-free update rule: preliminaries

- Instead of minimizing the "wall" with respect to the energy in the channel shortener, we can minimize with respect to the energy in the "window" of the effective channel.
- Recall that $\mathrm{E}\left[\left|y(\nu+\Delta)-y(\nu+N+\Delta)\right|^2\right]$ equals the "wall" power.
- It turns out that $\mathrm{E}\left[y^*(\nu+\Delta)y(\nu+N+\Delta)\right]$ is the "window" power.
- These two quantities are quadratic in the channel shortener, i.e. they equal $\mathbf{w}^T\mathbf{A}\mathbf{w}$ and $\mathbf{w}^T\mathbf{B}\mathbf{w}$.
- We want to solve

$$\mathbf{w}_{opt} = \arg\max_{\mathbf{w}} \frac{\text{window power}}{\text{wall power}} = \arg\max_{\mathbf{w}} \frac{\mathbf{w}^T\mathbf{B}\mathbf{w}}{\mathbf{w}^T\mathbf{A}\mathbf{w}} \tag{3}$$

FRODO: Division-free update rule

- Chatterjee, *et al.* proposed an iterative algorithm for computing the maximum generalized eigenvalue and its eigenvector:

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu \left(\mathbf{B}\mathbf{w} - \mathbf{A}\mathbf{w}\left(\mathbf{w}^H \mathbf{B}\mathbf{w}\right)\right) \tag{4}$$

- We can remove the expectations from $\mathbf{A}$ and $\mathbf{B}$ to get stochastic approximations. Combining this with Chatterjee's general algorithmic form, we get an adaptive generalized eigen solver.
- The new update rule is:

$$\boxed{\begin{array}{l} \text{Given } \Delta \text{ and } i_o, \text{ for symbol } k = 0, 1, 2, \ldots, \\ \qquad \tilde{\mathbf{r}}(k) = \mathbf{r}(Mk + \nu + \Delta) - \mathbf{r}(Mk + \nu + N + \Delta) \\ \qquad e(k) = \mathbf{w}^T(k)\ \tilde{\mathbf{r}}(k) \\ \qquad y_{i_o}(k) = y(Mk + i_o + \Delta) \\ \mathbf{w}(k+1) = \mathbf{w}(k) + \mu\ y_{i_o}(k)\left(\mathbf{r}^*(Mk + i_o + \Delta) - y_{i_o}^*(k) e(k) \tilde{\mathbf{r}}^*(k)\right) \end{array}}$$

Initialization

- We have

$$\mathbf{A} = \mathrm{E}\left[\left(\mathbf{r}_{\nu+\Delta}(k) - \mathbf{r}_{\nu+N+\Delta}(k)\right)\left(\mathbf{r}_{\nu+\Delta}(k) - \mathbf{r}_{\nu+N+\Delta}(k)\right)^T\right]$$
$$\mathbf{B} = \mathrm{E}\left[\mathbf{r}_{\nu+\Delta}(k)\ \mathbf{r}^H_{\nu+N+\Delta}(k)\right]$$

where k is the block number.

- We can initialize near the correct location by computing

$$\widehat{\mathbf{A}} = \frac{1}{K}\sum_{k=1}^{K}\left(\mathbf{r}_{\nu+\Delta}(k) - \mathbf{r}_{\nu+N+\Delta}(k)\right)\left(\mathbf{r}_{\nu+\Delta}(k) - \mathbf{r}_{\nu+N+\Delta}(k)\right)^T$$
$$\widehat{\mathbf{B}} = \frac{1}{K}\sum_{k=1}^{K}\mathbf{r}_{\nu+\Delta}(k)\ \mathbf{r}^H_{\nu+N+\Delta}(k)$$

and then computing the extreme eigenpair of matrices $\left(\widehat{\mathbf{A}}, \widehat{\mathbf{B}}\right)$.

- For a good estimate, $K \approx L_w$ or greater.

Synchronization

- Hitherto, all equations presume a good choice of delay Δ.
- Proposed heuristic choice: if Δ_{peak} optimizes the energy in a window of the channel, then choose

$$\Delta = \Delta_{peak} + \left\lfloor \frac{L_w}{2} \right\rfloor \tag{5}$$

- To get Δ_{peak}, first compute

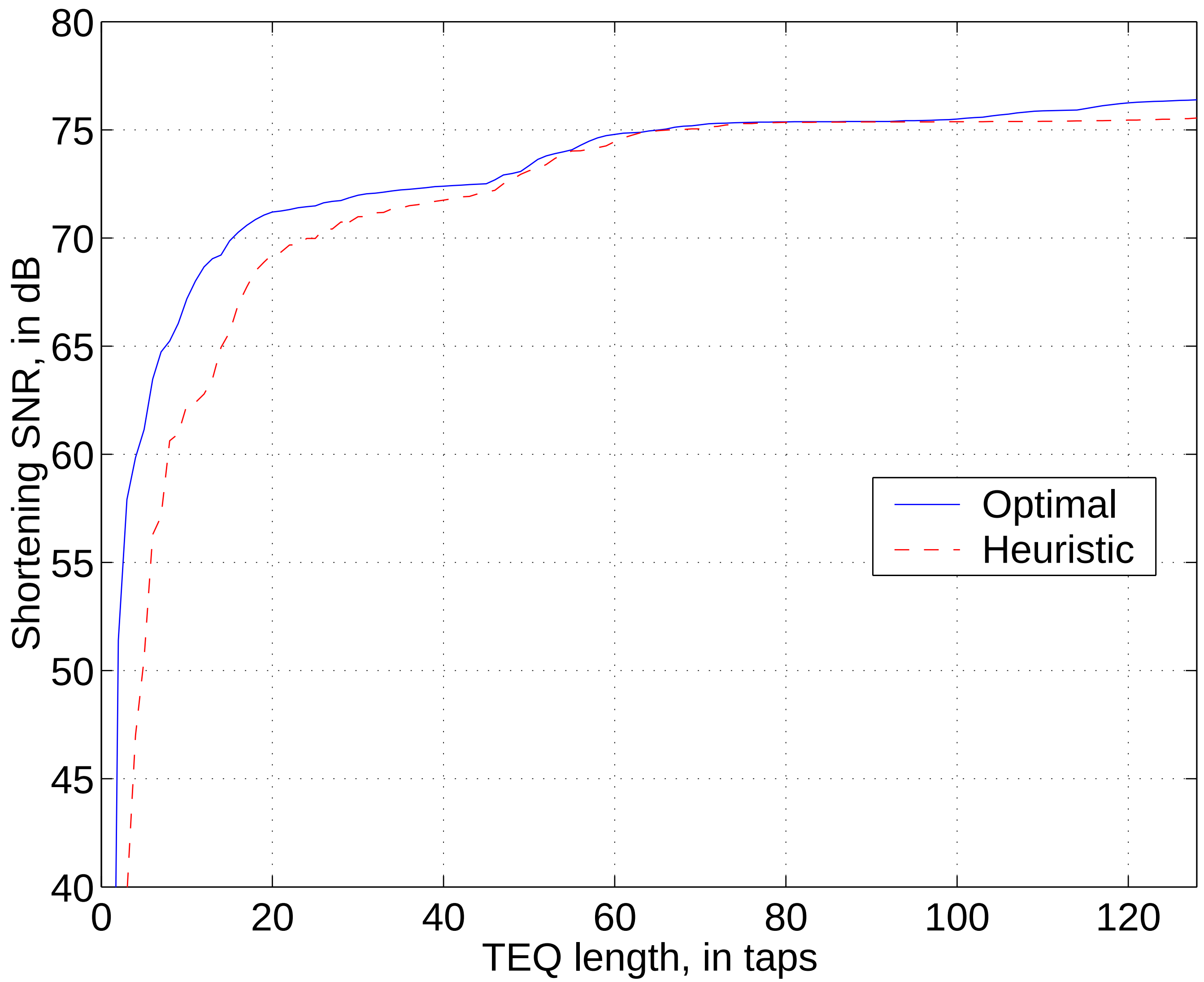
$$J_h(\Delta) = \sum_{k=1}^{K} |r(Mk + \nu + \Delta) - r(Mk + \nu + N + \Delta)|^2$$

  Like the MERRY cost, this gives the "wall" energy of the channel.

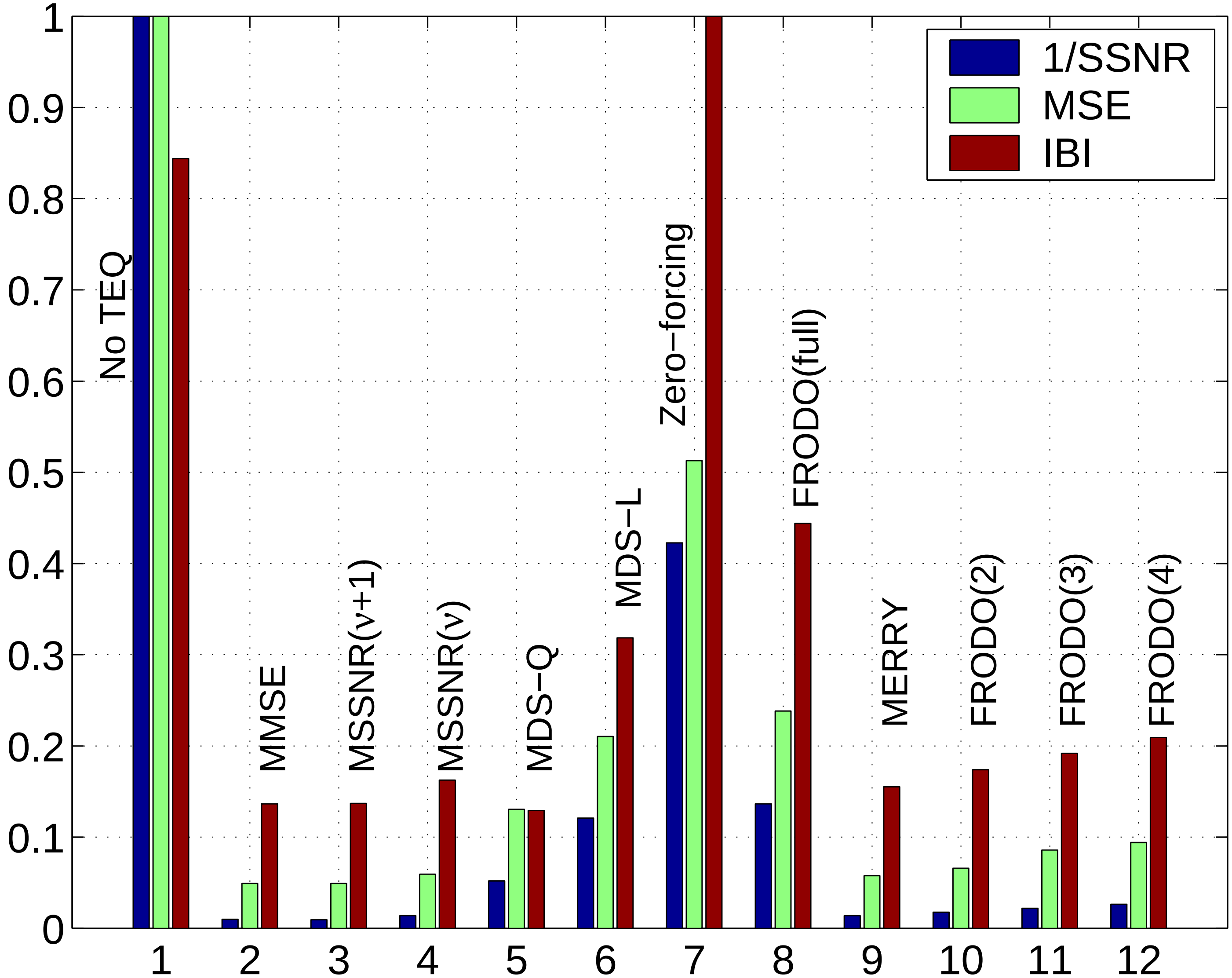
- The smaller the wall energy, the larger the window energy, so

$$\widehat{\Delta}_{peak} = \arg \min_{0 \le \Delta \le M-1} J_h(\Delta) \tag{6}$$

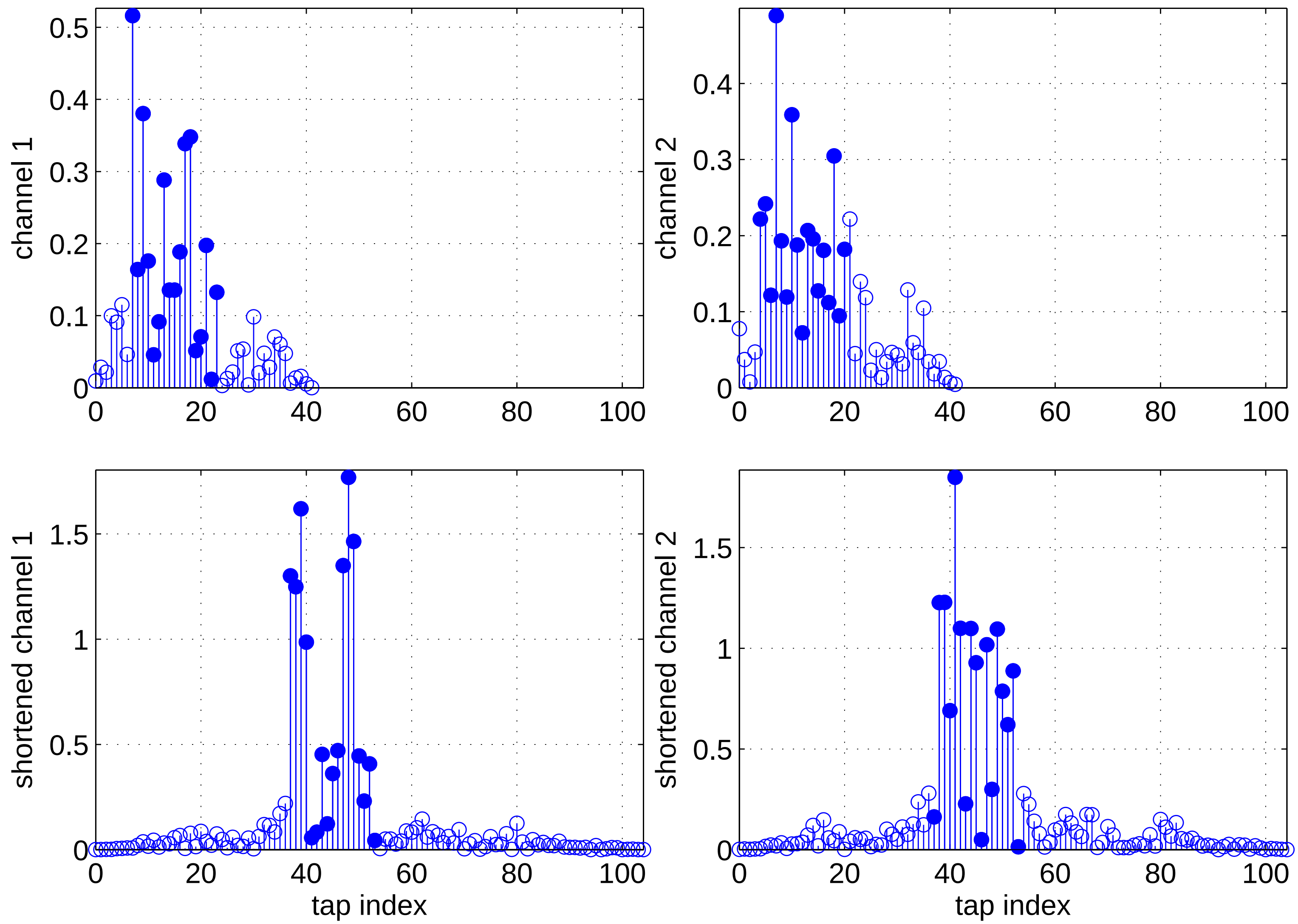
Performance of a heuristic delay choice



Design comparison



Performance of MIMO Merry



Conclusions

- We have extended MERRY in various ways, and dubbed the new algorithm "FRODO."
- FRODO avoids the computationally expensive square root and division of other channel shorteners, including MERRY.
- FRODO allows MIMO channel shortening and multiple comparisons between the CP and data.
- The use of more comparisons increases convergence speed but degrades asymptotic performance.
- We proposed blind initialization and synchronization techniques for a channel shortener.