Hardware Implementation of an Adaptive Minimum Bit-Error-Rate (MBER) Beamforming Algorithm

Researcher: Jor-El Sy Rivo
Adviser: Joel Joseph Marciano, PhD

Overview

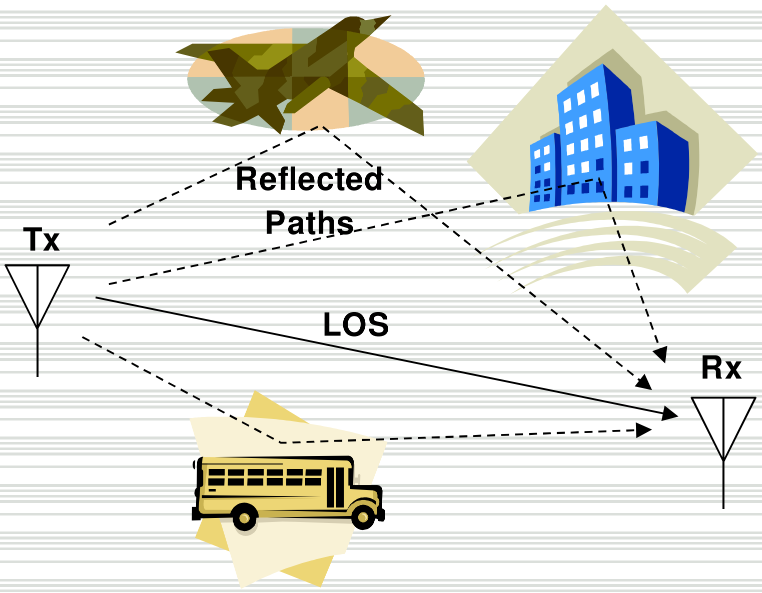
- Wireless Channel Effects
- Adaptive Array Beamforming
- MBER Algorithms
- Problem Statement
- Objectives of the Study
- Proposed Methodology
- Gantt Chart

Wireless Channel Effects

- Noise
  - Transmitter and receiver thermal vibrations
  - Black body radiation
  - Usually modeled by Additive White Gaussian Noise (AWGN)
- Interference
  - Multiple Access Interference (MAI)
  - Co-Channel Interference (CCI)

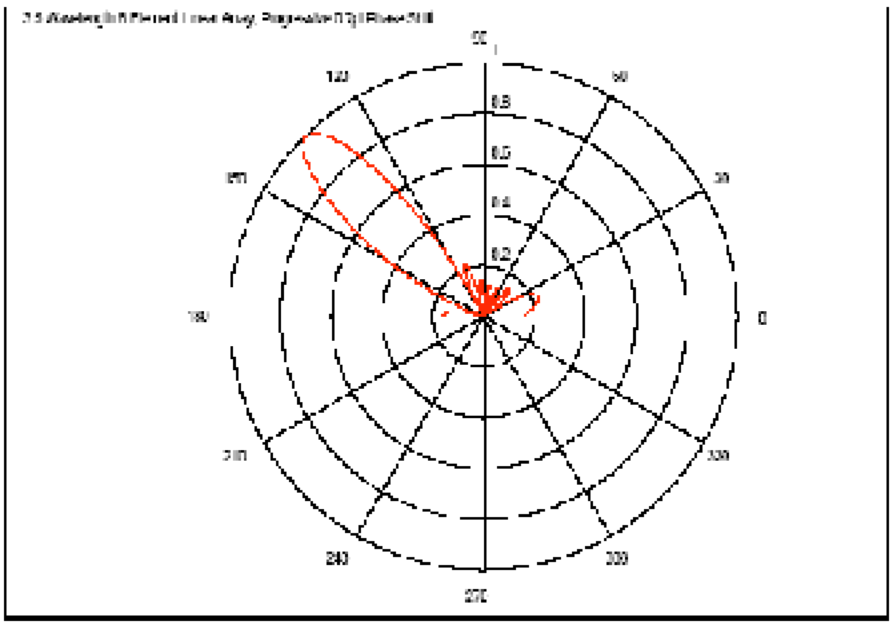
Wireless Channel Effects

- Multipath
  - Receiver signal vector sum of multiple path signals
  - May cause fading
  - May cause inter-symbol interference (ISI)



Adaptive Array Beamforming

- Antenna arrays may be used to combat wireless channel effects
- Signals received by antenna elements multiplied by weights to control beam pattern
- Beamformer output: $y(k)=\mathbf{w}^{H}\mathbf{x}(k)$



Adaptive Array Beamforming

Traditionally, Mean-Squared-Error (MSE) is minimized

Error:

$$\varepsilon(k) = d(k) - y(k) = d(k) - \mathbf{w}^{\mathbf{H}}\mathbf{x}(k)$$

Cost function:

$$\mathrm{MSE} = E\left[d^2(k)\right] + \mathbf{w}^{\mathbf{H}}\mathbf{R}\mathbf{w} - \mathbf{2}\mathbf{p}^{\mathbf{H}}\mathbf{w}$$

Adaptive Array Beamforming

- Minimizing MSE indirectly minimizes the BER
- Optimum weight vector given by:

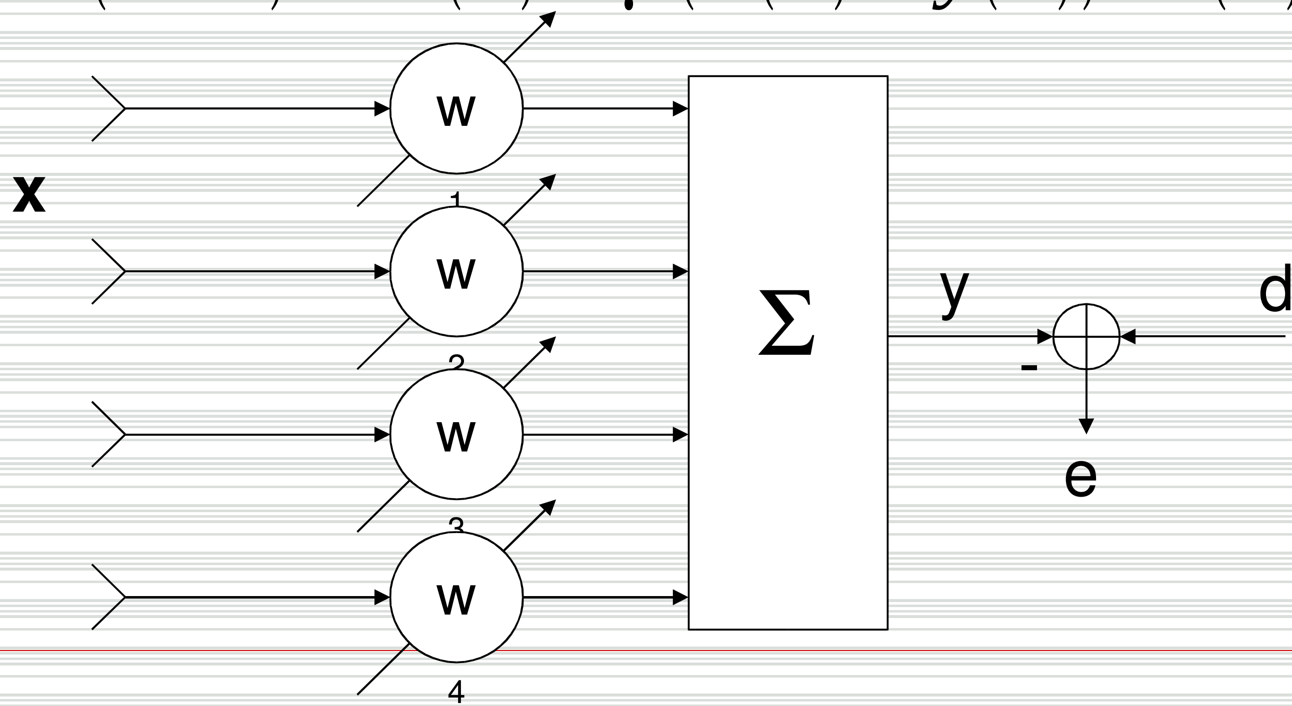
$$\mathbf{w}_{\mathrm{MMSE}} = \mathbf{R}^{-1}\mathbf{p}$$

- **R** and **S** usually not known, hence the need for adaptive algorithms

Adaptive Array Beamforming

Least-Mean-Square (LMS) algorithm used to iteratively update weights:

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu\left(d(k) - y(k)\right) * \mathbf{x}(k)$$



MBER Algorithms

S. Chen et al, “Adaptive Minimum Bit Error Rate Beamforming,” *IEEE Transactions on Wireless Communications*, April 2002.

BER should be directly minimized, not the MSE

MBER Algorithms

- For M sources using BPSK, there are $N_b=2^M$ possible sequences b_q
- Output takes values from the set:

$$\mathrm{Y}_R = \left\{ \bar{y}_{R,q} = \Re\left[\mathbf{w}^{\mathbf{H}} \bar{\mathbf{x}}_{\mathbf{q}}\right], 1 \le q \le N_b \right\}$$

$$\mathrm{Y}_R^{(\pm)} = \left\{ \bar{y}^{(\pm)}_{R,q} \in \mathrm{Y}_R, b_1 = \pm 1 \right\}$$

- Cost function:

$$P_E(\mathbf{w}) = \frac{2}{N_b} \sum_{q=1}^{N_b/2} Q\left(g_{q,+}(\mathbf{w})\right) \qquad g_{q,+}(\mathbf{w}) = \frac{\operatorname{sgn}\left(b_{q,1}\right)\Re\left[\mathbf{w}^H \bar{\mathbf{x}}_q^{(+)}\right]}{\sigma_n \sqrt{\mathbf{w}^H \mathbf{w}}}$$

MBER Algorithms

- Optimum weight vector given by:

$$\mathbf{w}_{\text{MBER}} = \arg\min P_E(\mathbf{w})$$

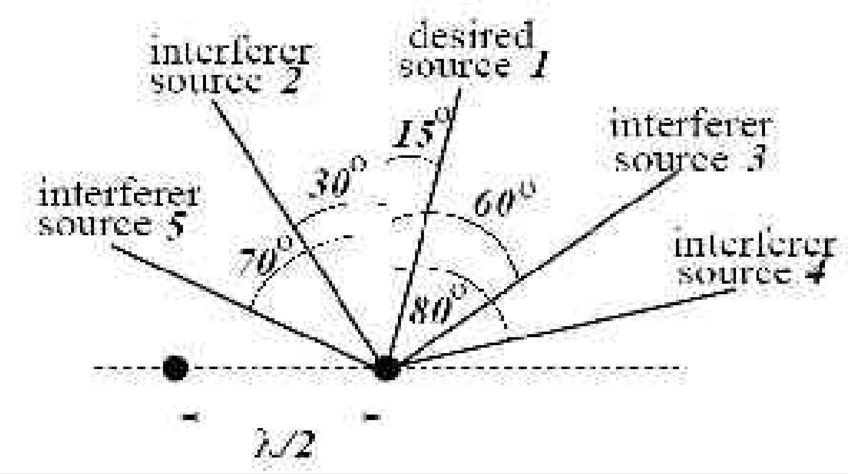
- Approximate Least Bit-Error-Rate (ALBER) used to iteratively update weights:

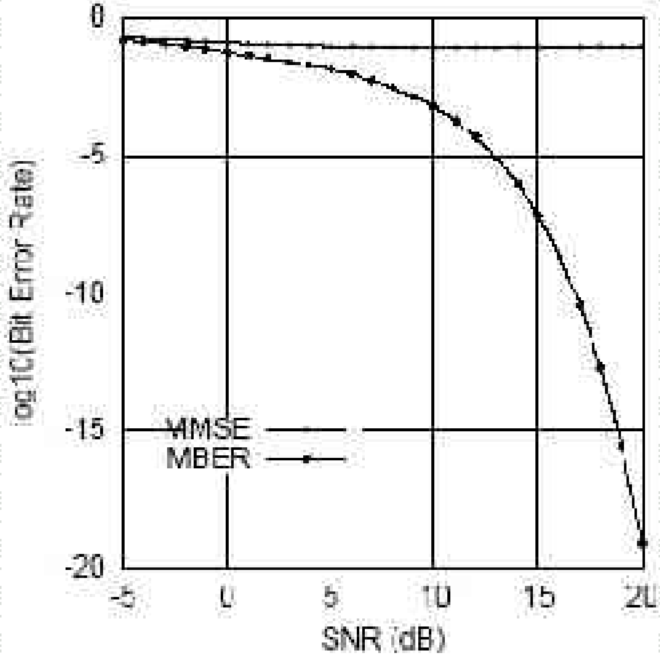
$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu \frac{\operatorname{sgn}(b_1(k))}{2\sqrt{2\pi}\,\rho_n} \exp\left(-\frac{y_R^2(k)}{2\rho_n^2}\right)\mathbf{x}(k)$$

- Has similar computational complexity to LMS algorithm
- Converges at a reasonable rate close to optimum MBER solution

MBER Algorithms

Simulation results





Problem Statement

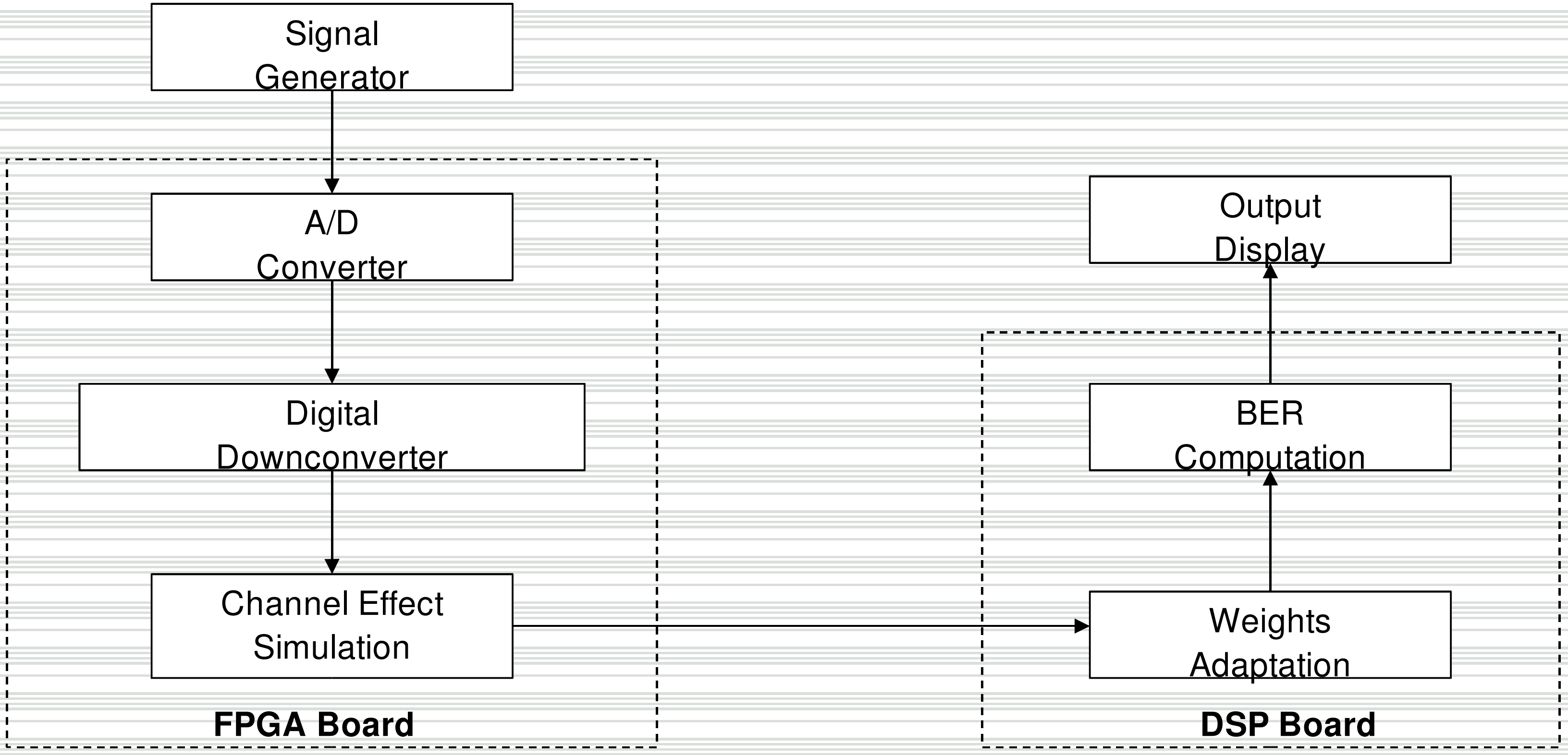
- MBER beamformers offer potential gain over MMSE beamformers
- Approximate Least Bit-Error-Rate (ALBER) algorithm was developed which has same complexity as LMS
- Feasibility of real-time implementation is yet to be tested

Objectives of the Study

- Implement ALBER algorithm in real-time on existing hardware
- Compare BER performance of ALBER versus LMS
- Compare memory and processing power requirements of ALBER versus LMS

Proposed Methodology - Implementation

Block Diagram



Proposed Methodology - Implementation

- Signal Generator
  - BPSK signals
  - IF frequency of 100 MHz
- A/D Converter
  - Signals sampled at 80 MSPS
  - 12 bits per sample
- Digital Downconversion
  - Performed in FPGA, Altera’s Stratix EP1S25 DSP Development Board

Proposed Methodology - Implementation

- Channel Effect Simulation
  - Noise
  - Co-Channel Interference
  - Multipath
  - Performed in FPGA
- Weights Adaptation
  - Performed in DSP board, TI's TMS320C6701
  - Both LMS and ALBER algorithm will be implemented

Proposed Methodology - Implementation

- BER Computation
  - Performed in DSP board
- Output Display
  - DSP board interfaced to a host computer through JTAG emulation
  - Simulation results displayed in host computer

Proposed Methodology -Testing

- Four test cases
  - AWGN
  - AWGN + CCI
  - AWGN + Multipath
  - AWGN + CCI + Multipath
- BER plots will be used as main basis of comparison for algorithm performance
- Other comparisons to be made
  - FPGA logic elements used
  - Memory space in the DSP board needed for program and lookup tables
  - Number of cycles used in DSP board

Gantt Chart

November				December				January				February				March				April				May			
1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
(1)																											
				(2)																							
												(3)															
																(4)											
																				(5)							
																(6)											

Gantt Chart

(1) Design of Digital Downconverter

(2) Coding of Channel Effects

(3) Coding of MMSE and MBER algorithms

(4) Integration

(5) Testing

(6) Documentation

Thank You!