A bit-streaming, pipelined multiuser detector for wireless communications

Sridhar Rajagopal and Joseph R. Cavallaro
Rice University
{sridhar,cavallar}@rice.edu

This work is supported by Nokia, TI, TATP and NSF
Direct

Reflections

Jointly detect data of all users
Benefits of multiuser detection

Error rate vs. SNR

- Single-user (channel estimation + detection)
- Multi-user estimation + Single-user detection
- Multi-user (channel estimation + detection)
Motivation

Implement multiuser detection for 3G wireless CDMA base-station receivers

Challenges:

- large complexity
- block based algorithms (latency)

Unable to meet real-time requirements (3GPP)

- 128 Kbps for 32 users with spreading = 32 chips/bit
Contributions

Developed a simple architecture for asynchronous multiuser detection for CDMA \[+\,\times\]

Bit-streaming
- reduced latency
- eliminates window edge computations
- lower memory requirements

Pipelined stages
- higher throughput (with more hardware)

Real-time implementation for multiuser detection now possible for 3GPP!
Asynchronous multiuser interference

Interference due to past, current and future bits of other users
Multistage Parallel Interference Cancellation (PIC)

Received Signal
\[ r_1 \ldots r_D \]

Channel Estimate
\[ A = [A_0 \ A_1] \otimes I(D) \]

Conventional code matched filter
\[ y^0 = \text{Re}[A^H r] \]
\[ \hat{d}^0 = \text{sign}(\hat{y}^0) \]

Channel Estimate
\[ B = A^HA - \text{diag}(A^HA) \]

PIC Stage 1
\[ y^1 = y^0 - B\hat{d}^0 \]
\[ \hat{d}^1 = \text{sign}(\hat{y}^1) \]

Stage 2
\[ \hat{d}^2_1 \ldots \hat{d}^2_D \]

Stage 3
\[ \hat{d}^3_1 \ldots \hat{d}^3_D \]

Detected bits
\[ d_1 \ldots d_D \in K \{+1,-1\} \]

\[ A \in ND \times KD \]
\[ y \in K \]
\[ r \in N \]
Latency - variable [Worst case (1st bit) \( \approx D \times \text{latency per bit} \)]

2 extra edge bit computations per stage.
Bit-streaming the multiuser detection algorithm

Tri-diagonal Block Toeplitz matrix $B$ $[KD \times KD]$ $D$- detection window length

$$
\begin{bmatrix}
A_0^H A_0 & A_0^H A_1 & 0 & 0 & 0 \\
A_1^H A_0 & A_0^H A_0 + A_1^H A_1 & A_0^H A_1 & 0 & \\
\vdots & \ddots & \ddots & A_0^H A_1 & 0 \\
\vdots & & 0 & A_1^H A_0 & A_0^H A_0 + A_1^H A_1 & A_0^H A_1 \\
0 & 0 & \cdots & A_1^H A_0 & A_0^H A_0 + A_1^H A_1 &
\end{bmatrix}
\begin{bmatrix}
\hat{d}_1 \\
\hat{d}_{i-1} \\
\hat{d}_i \\
\hat{d}_{i+1} \\
\hat{d}_D
\end{bmatrix}
$$

$$
y^m_i = y_i^m - L\hat{d}^m_{i-1} - C\hat{d}_i^m - R\hat{d}^m_{i+1}
$$

$L, C, R \in K \times K$ Savings in memory by $D^2$

$y \in K$
Pipelining the multiuser detector

Latency = 2*latency per bit (D/2 speedup over block)
eliminated edge bit computations.
Pipelined architecture for multiuser detection
FPGAs for pipelining

DSPs not suitable for exploiting bit-level parallelism
FPGAs - Flexibility of ASICs
Good for parallelism and bit-level operations

```
Received bits

DSP [x]
FPGA1 [+]
FPGA2 [+]
FPGA3 [+]

MF → PIC (Stage 1) → PIC (Stage 2) → PIC (Stage 3)
```

Detected bits
Performance Comparisons

$t_{MF} = O(K)$
$t_{PIC} = O(K^2)$
Prototype chip built @ Rice

Number of users supported: 4
Area available: 3000x3000 inside the pad frame
Area used: ~85%
CMOS micron process: 0.5 micron
Chip speed: 2Mbps
http://www.owlnet.rice.edu/~sunbeam/422/
Summary

Simple, bit-streaming pipelined multiuser detector
Avoids block computations
  - Savings in memory by $D^2$
No edge bit computations in a window
  - $2/D$ computational savings per stage
Lower constant latency by $D/2$.
Leads to a Real-time DSP implementation for 3GPP.