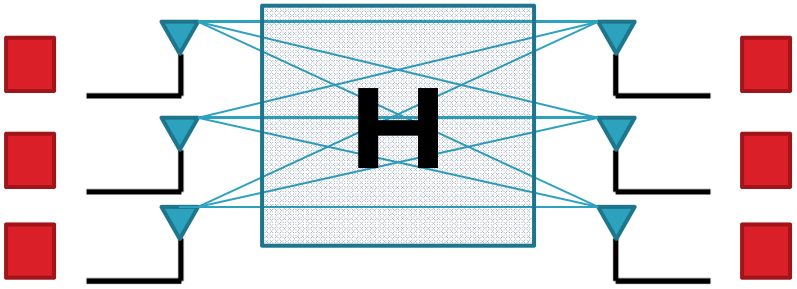
# Improving MIMO Sphere Detection Through Antenna Detection Order Scheduling

Michael Wu, Chris Dick, Yang Sun, Joseph Cavallaro

December 1, 2011

# MIMO Detection



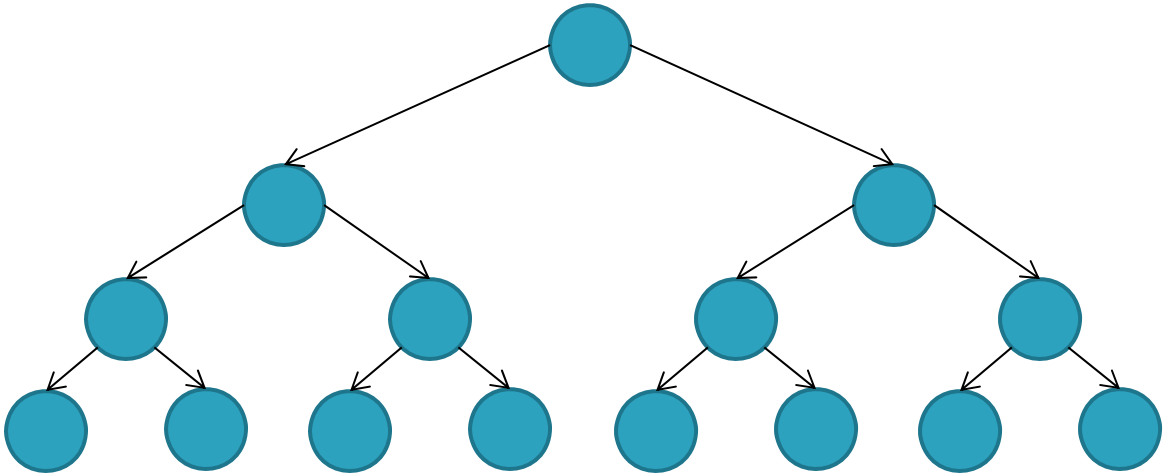

$$\begin{bmatrix} \hat{y}_0 \\ \hat{y}_1 \\ \hat{y}_2 \end{bmatrix} = \begin{bmatrix} h_{00} & h_{01} & h_{02} \\ h_{10} & h_{11} & h_{12} \\ h_{20} & h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} n_0 \\ n_1 \\ n_2 \end{bmatrix}$$

$$\hat{\mathbf{y}} = \mathbf{Rx} + \mathbf{n}$$

$$\min_{\mathbf{x} \in \Omega^n} \left\| \hat{\mathbf{y}} - \mathbf{Rx} \right\|_2^2$$

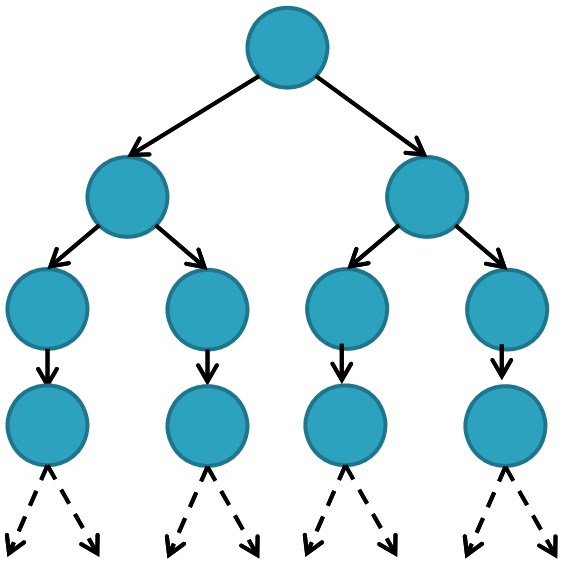
- Spatial Multiplexing
  - Increases throughput
  - Used in many wireless standards
- MIMO detector recovers original signal
- Search problem

# Tree-Search Based Detection



- Depth-first tree search
  - Variable execution time
  - Fairly sequential, slow
- Breadth-first tree search
  - More data parallel
  - Sort is the bottleneck
    - Number of comparisons + Memory requirement (KM)

# SSFE Detector

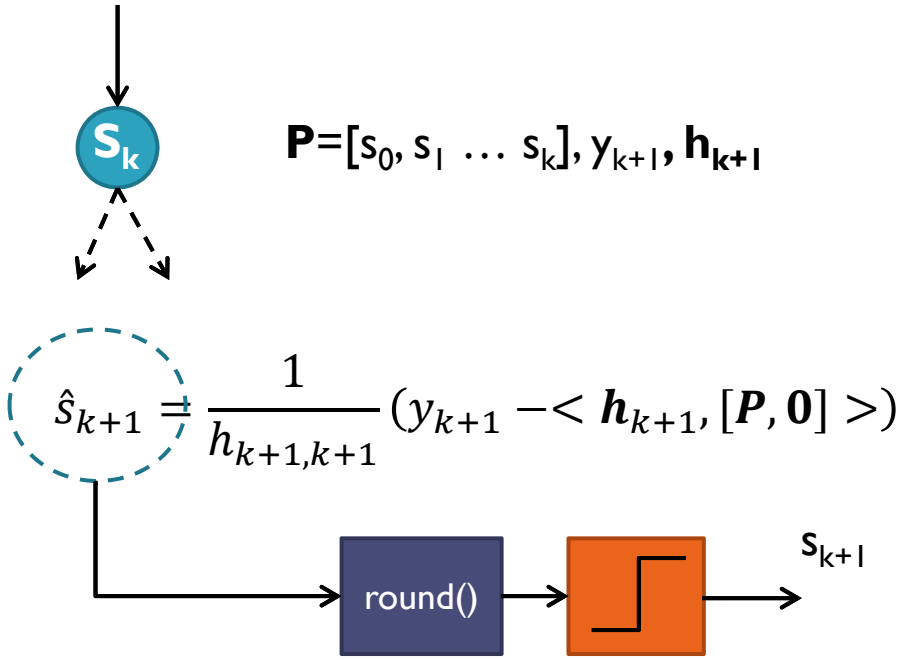


QPSK 2x2 Example

- Selective Spanning with Fast Enumeration (SSFE)
- Data parallel deterministic search
- First antenna level
  - Enumerate all modulation points
- Subsequent levels
  - Pick the best outgoing node for each path.

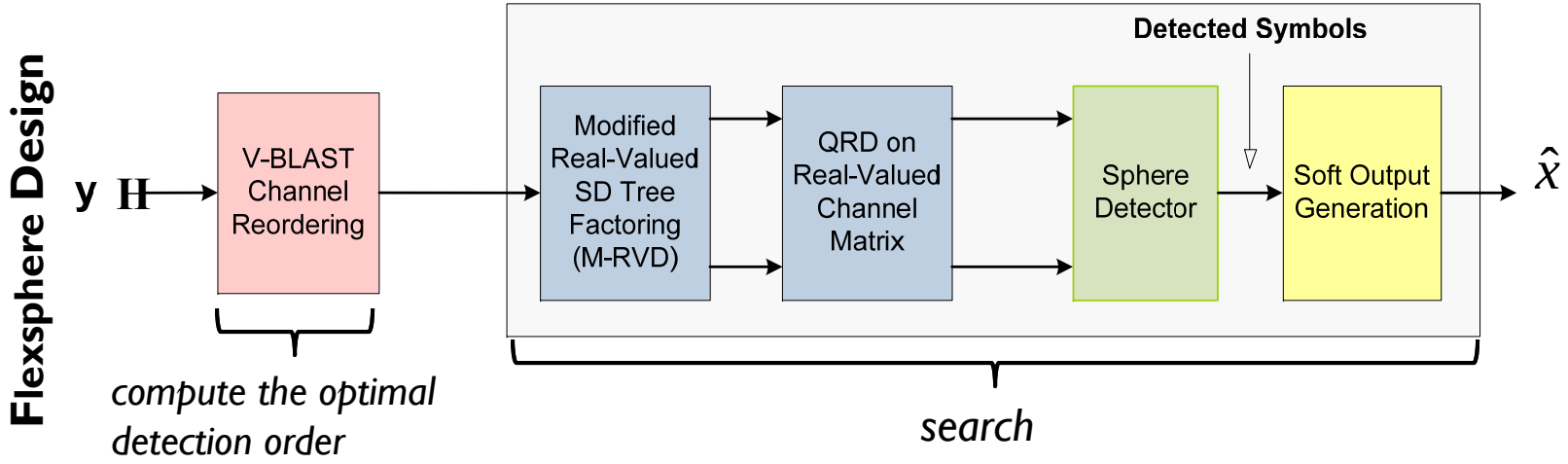
# SSFE Detector

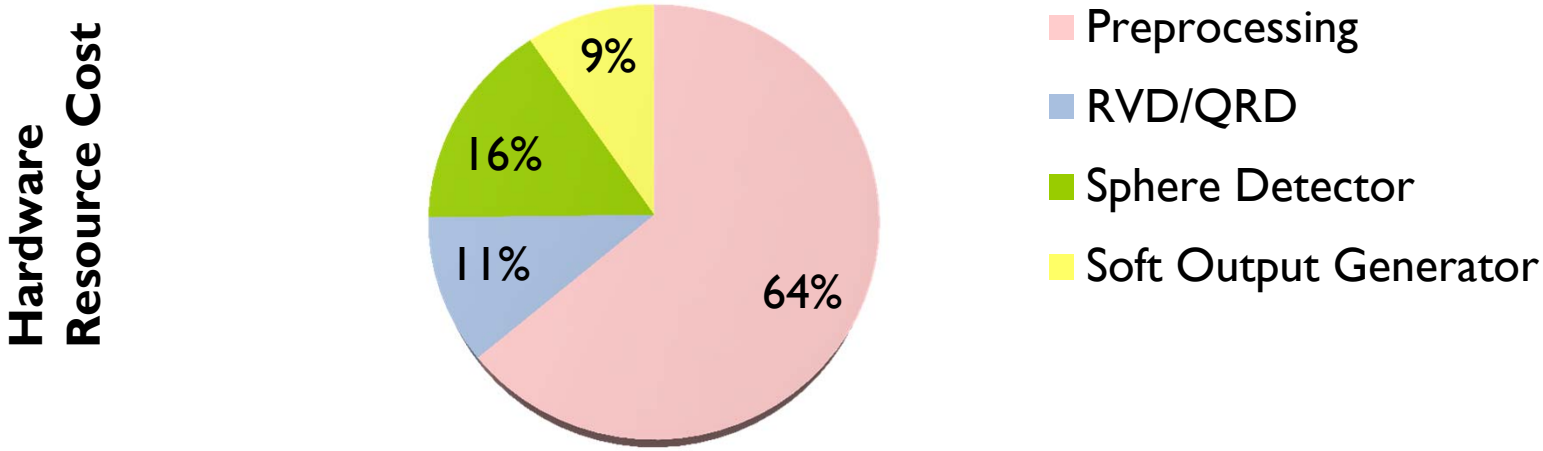
- "Sort-free" MIMO Detector
  - Picking the best outgoing node does not require sort




Schnoor-Euchner enumeration

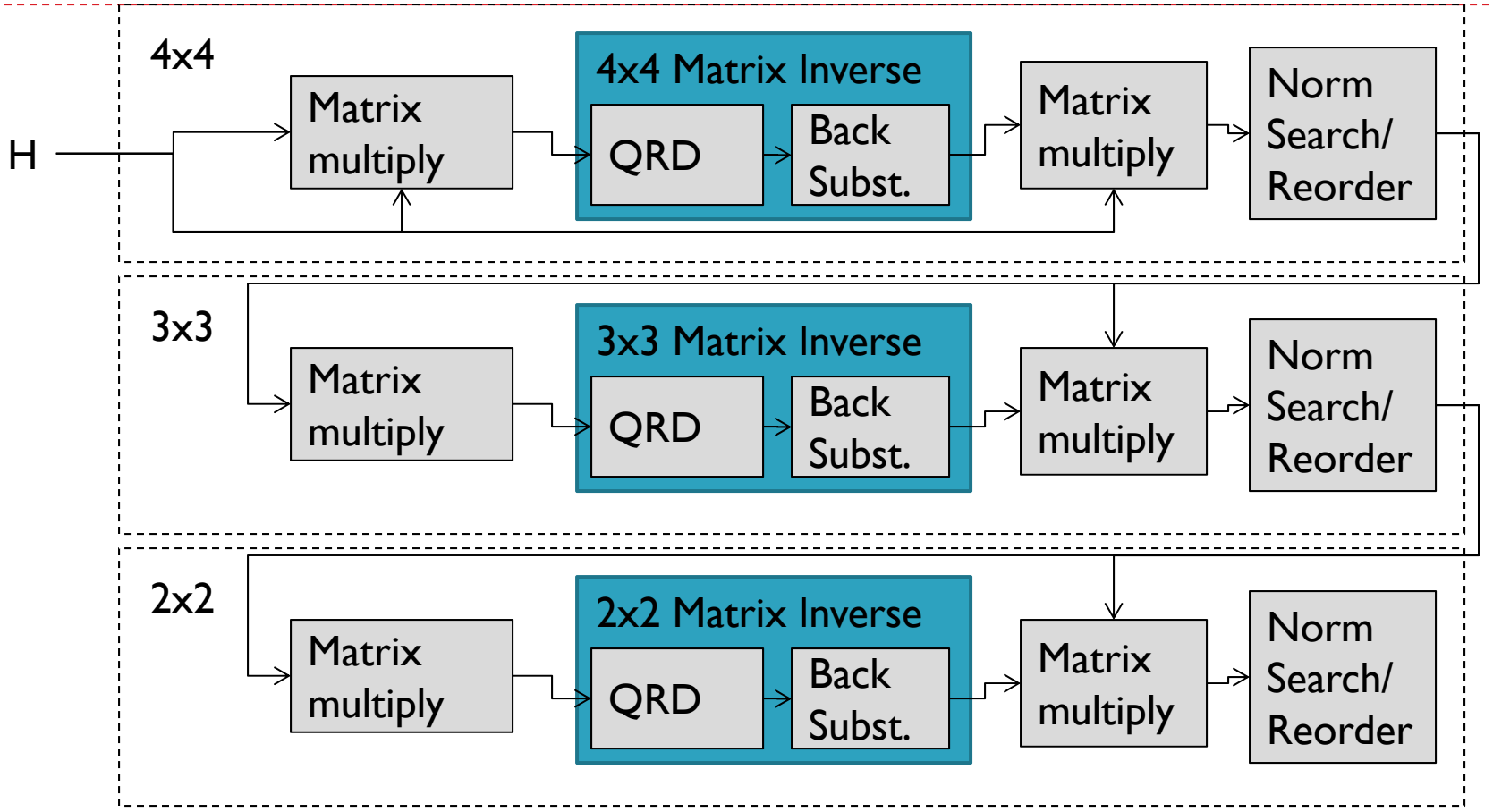
# Flexsphere Implementation



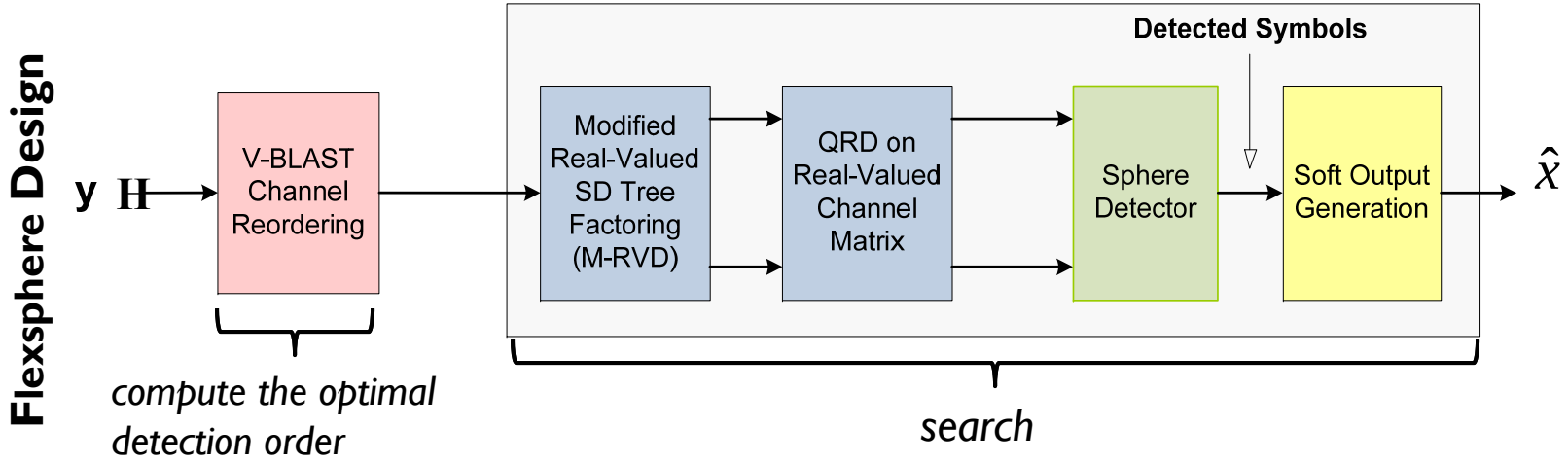

*C. Dick, M. Trajkovic, S. Denic, D. Vuletic, R. Rao, F. Harris, K. Amiri, FPGA Implementation of a Near-ML Sphere Detector for 802.16e Broadband Wireless Systems , proceedings of SDR conference, 2009*

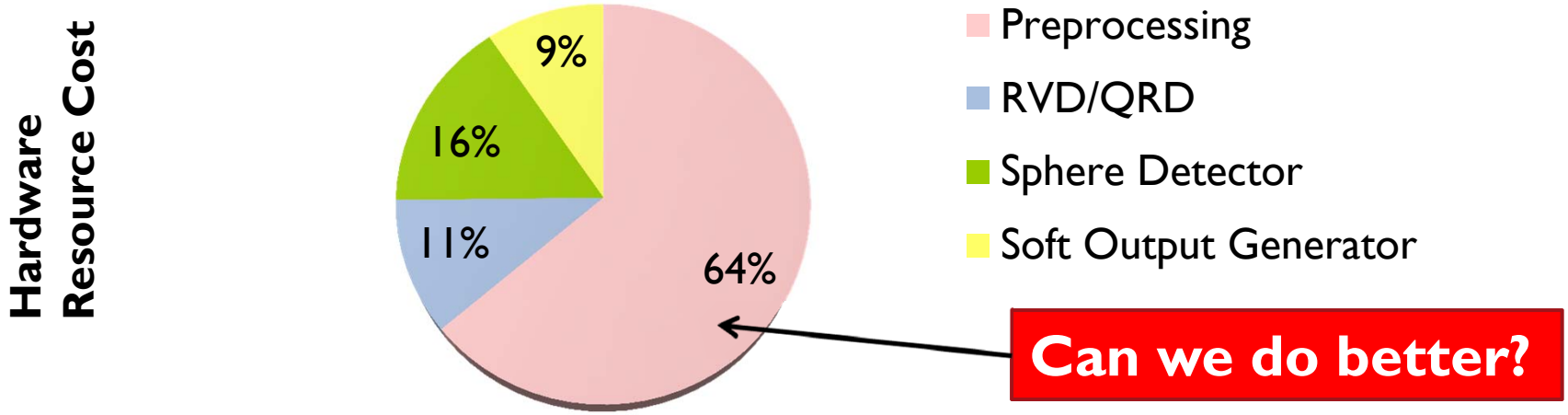
# Flexsphere Implementation: V-BLAST Reordering




- Complex block: 3 matrix inverses and 3 matrix multiplies

# Flexsphere Implementation

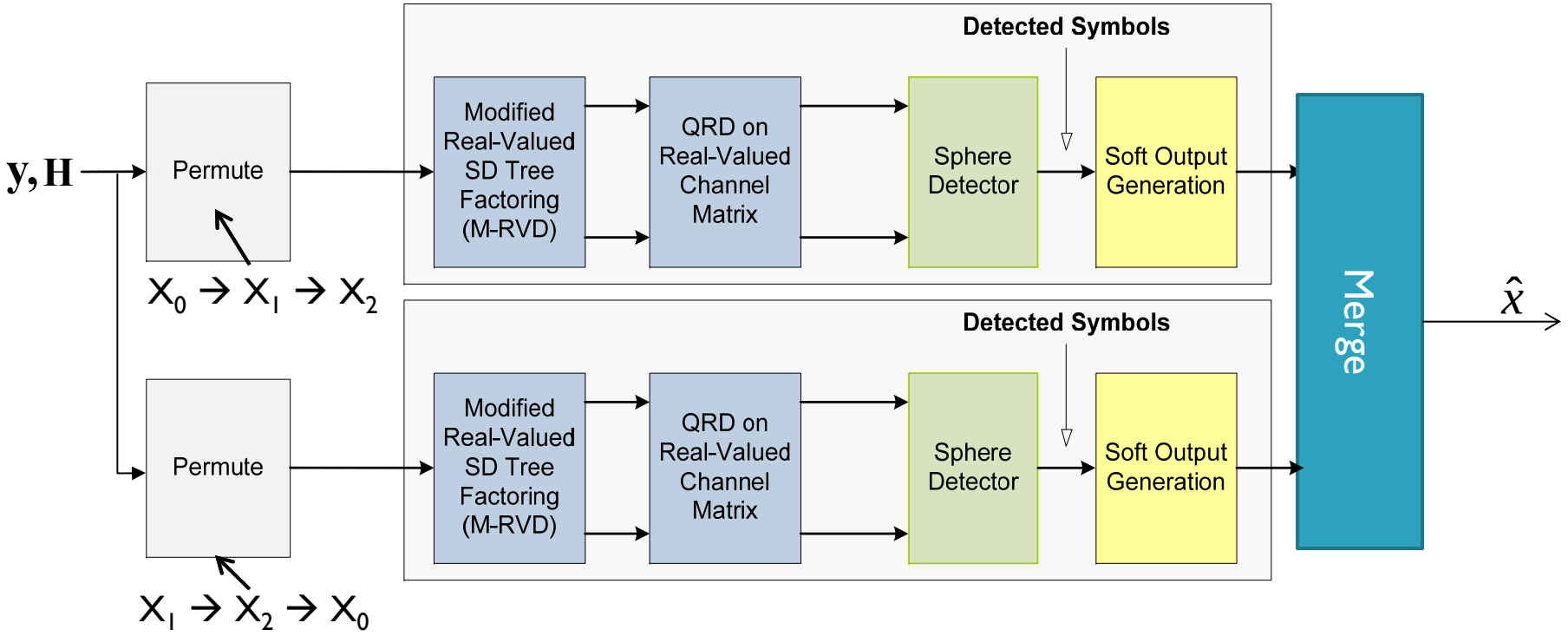



*C. Dick, M. Trajkovic, S. Denic, D. Vuletic, R. Rao, F. Harris, K. Amiri, FPGA Implementation of a Near-ML Sphere Detector for 802.16e Broadband Wireless Systems , proceedings of SDR conference, 2009*

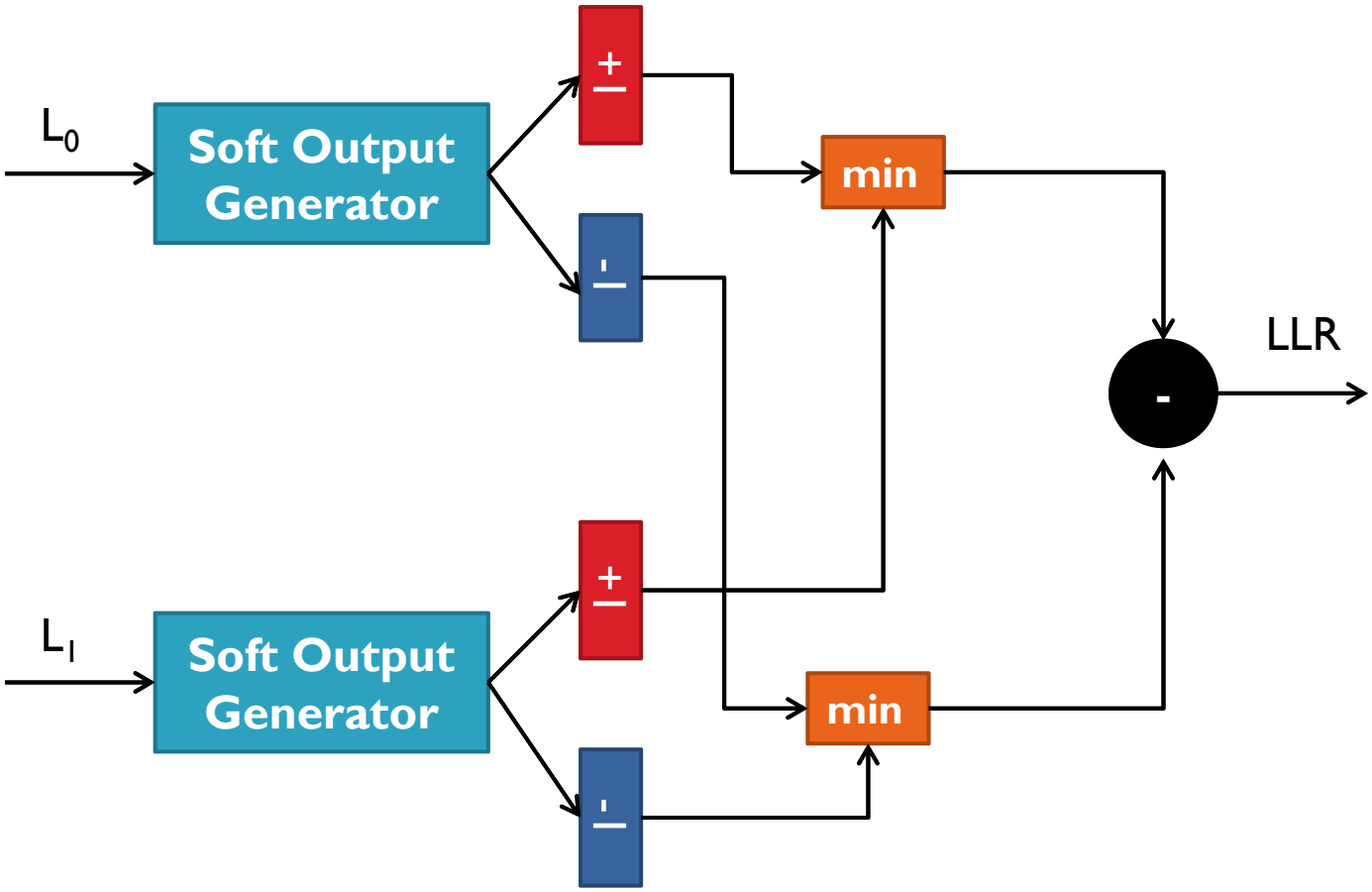
# N-Way MIMO Detector

- Get rid of the V-BLAST channel reordering block
- Duplicate search blocks depending on BER requirement.
  - Add permute block which enforces a detection order
  - Example: N = 2, two search blocks



# N-Way MIMO Detector: Merge Block

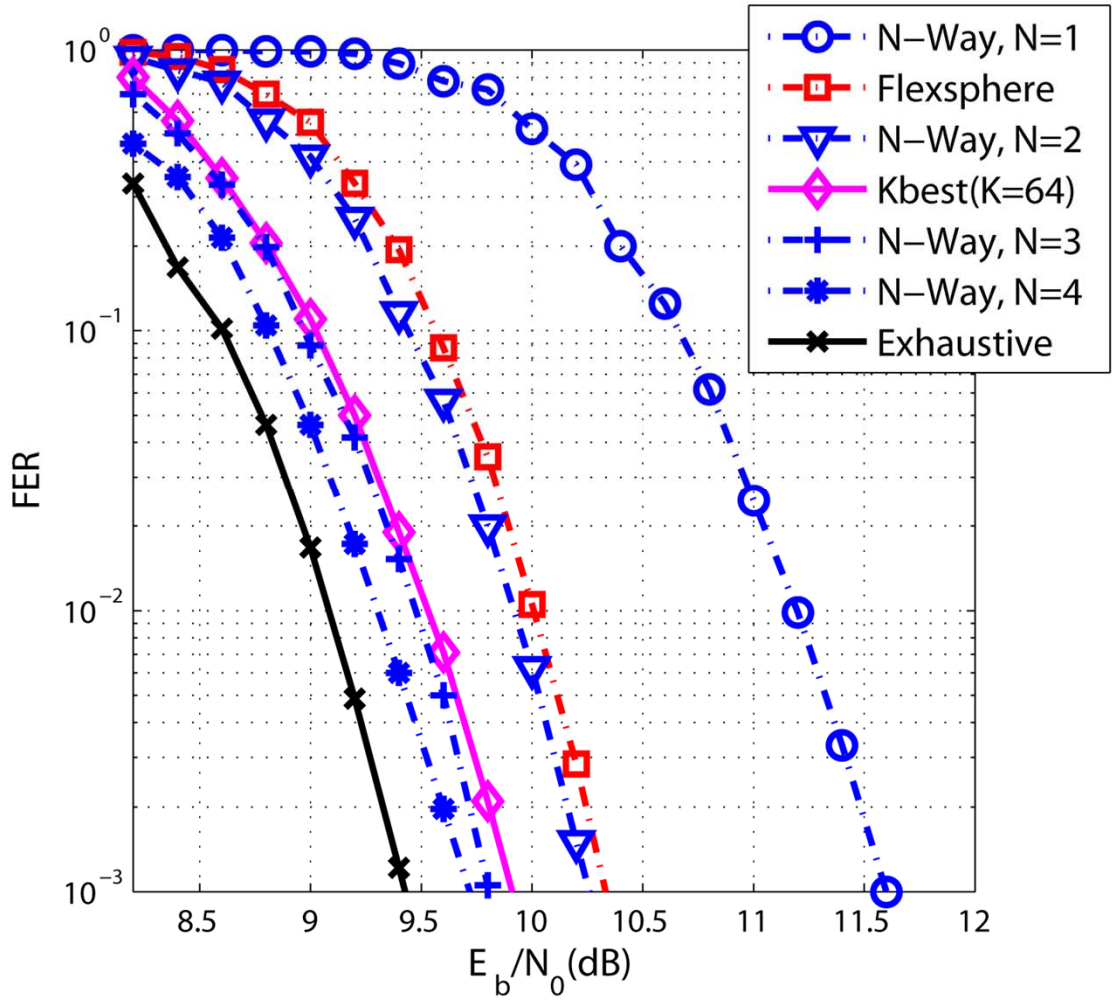
- Simple block, performs max-log-map computation



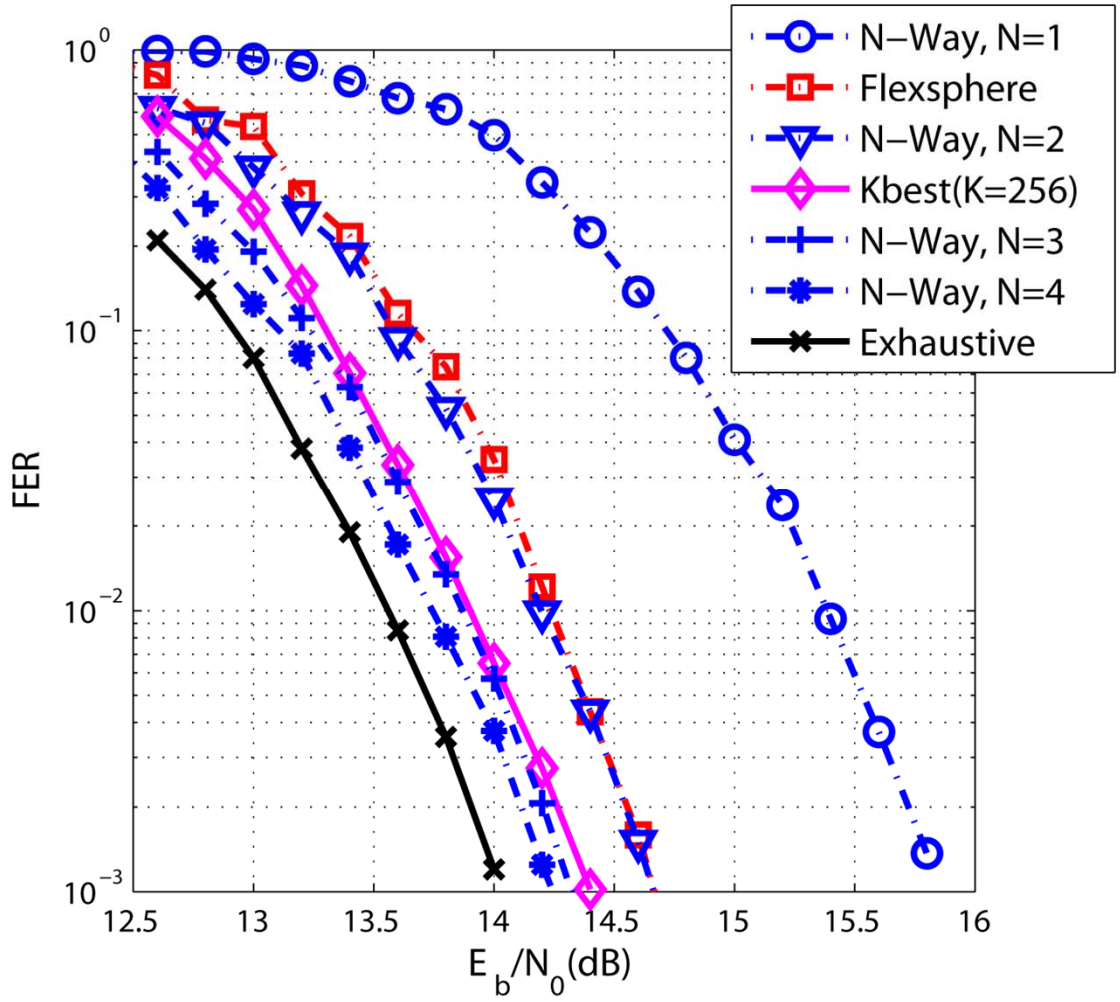
# BER Performance

- Rayleigh fading channel
- Soft Output MIMO Detector + Rate 1/2 WiMAX LDPC decoder
  - 1 outer iteration + 20 inner iteration with early termination

# BER Performance (16QAM)



# BER Performance (64QAM)



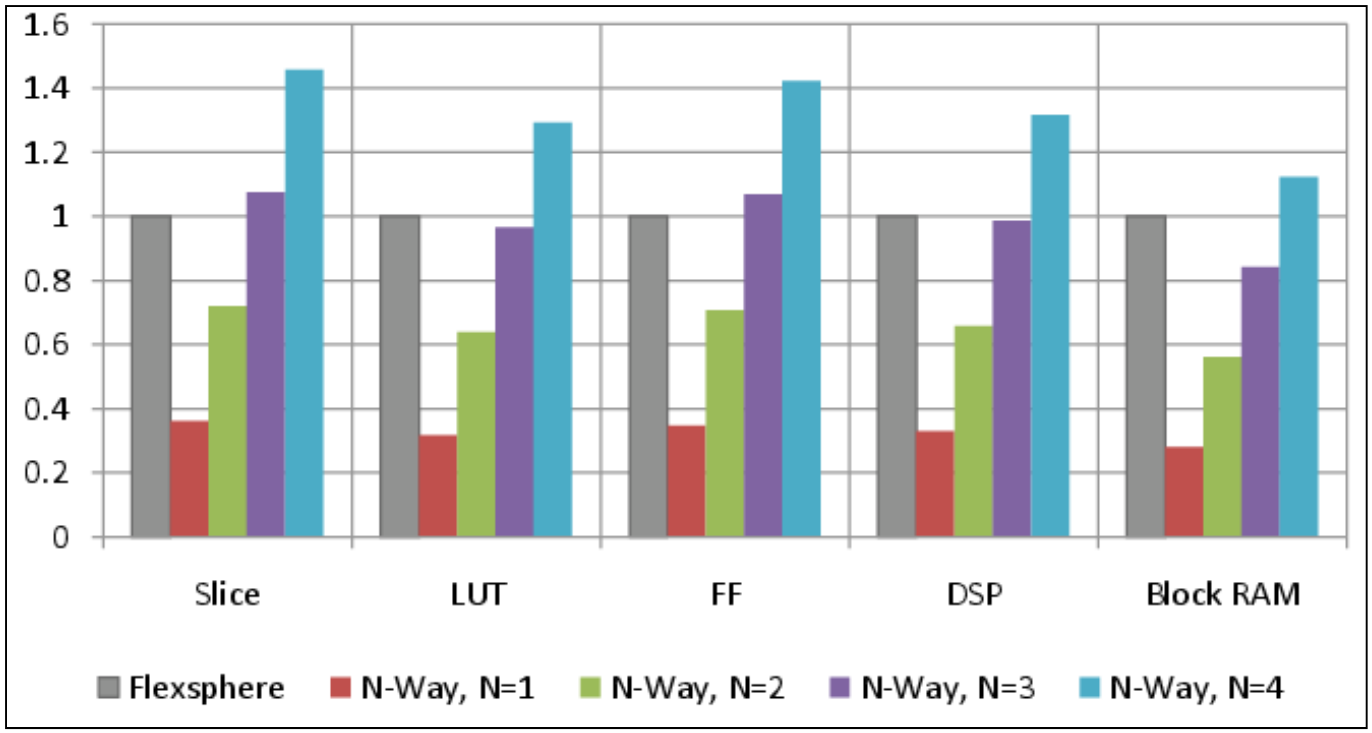
# Implementation: N-Way MIMO Detector

- Target: 83.768Mbps(WiMAX), Virtex 5 @ 225Mhz
- Total resource = N (RVD/QRD + Sphere Detector + Soft Output Generator) + Merge

| N | Slices | LUTs/FFs | DSP48 | Block RAM |
|---|---|---|---|---|
| 1 | 5,658 | 9,437/15,990 | 78 | 41 |
| 2 | 11,274 | 19,018/32,525 | 156 | 82 |
| 3 | 16,827 | 28,743/49,117 | 234 | 123 |
| 4 | 22,832 | 38,515/65,381 | 312 | 164 |
| Flexsphere | 15,657 | 29,776/45,944 | 237 | 146 |

# Implementation: N-Way MIMO Detector

- Target: 83.768Mbps(WiMAX), Virtex 5 @ 225Mhz



# Conclusion

- Scalable data parallel detection algorithm
  - Search is cheap
  - Better performance/resource compare to Flexsphere

- Target for software implementation?
  - Enumeration complexity doesn't depend on modulation
  - Increase parallelism of the detection algorithm