RWTH AACHEN UNIVERSITY

Processing energy vs. perfomance: Conclusions from a multi-core design

Gerd Ascheid, F. Borlenghi, M. Witte

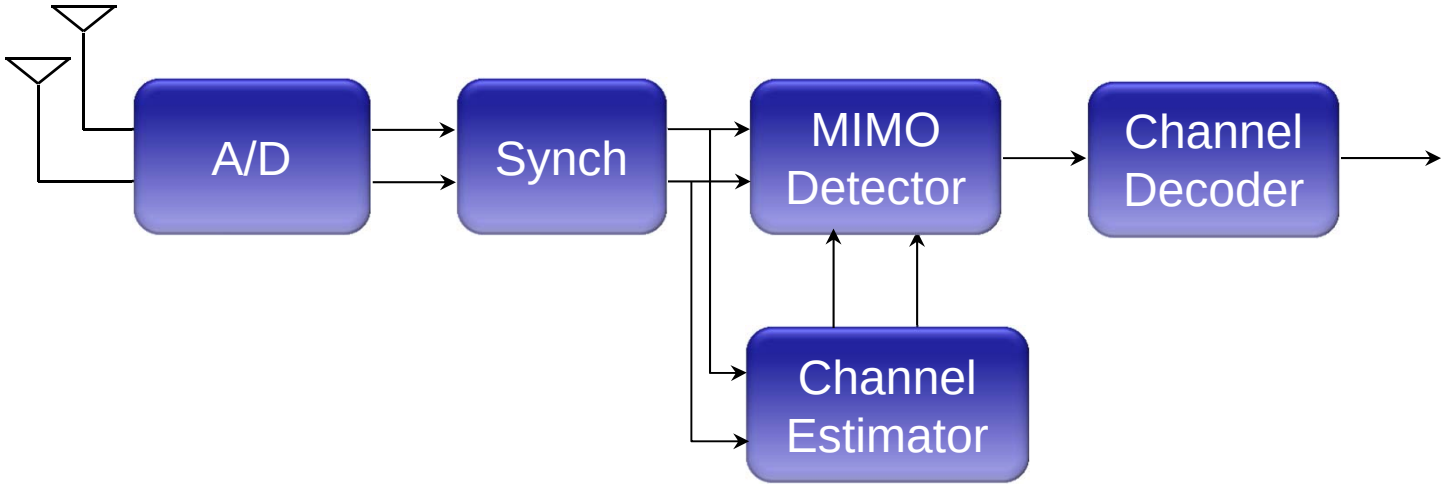
Institute for Communication Technologies
and Embedded Systems

MPSoC'13

Agenda

- **The issue: iterative MIMO receivers** — *What's the problem?*
- **IterX:**
  an iterative demapping/decoding ASIC
- **Hardware cost of performance**
- **Conclusions**

Optimum Receiver Processing

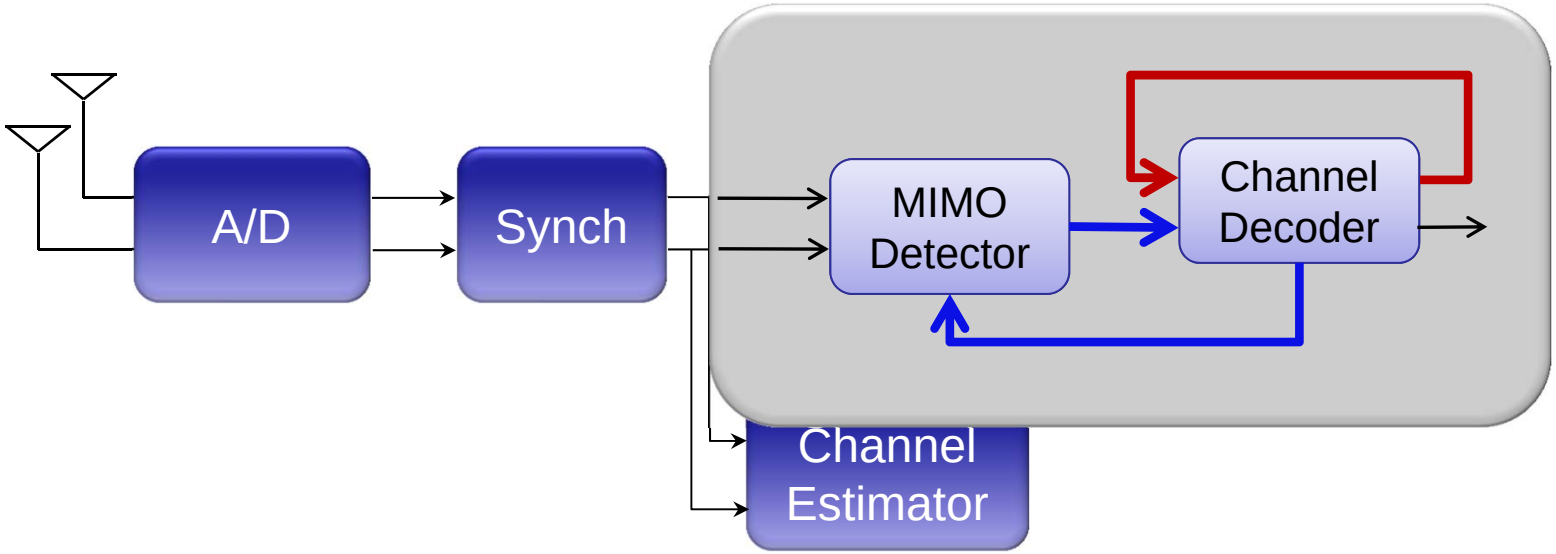


Optimum processing unfeasible, practical approach:

- Optimum estimation + likelihood information passed from left to right

⇨ **Potential performance gain by iterative optimization** (like in Turbo and LDPC decoding)

Case Study: A MIMO Doubly Iterative Receiver



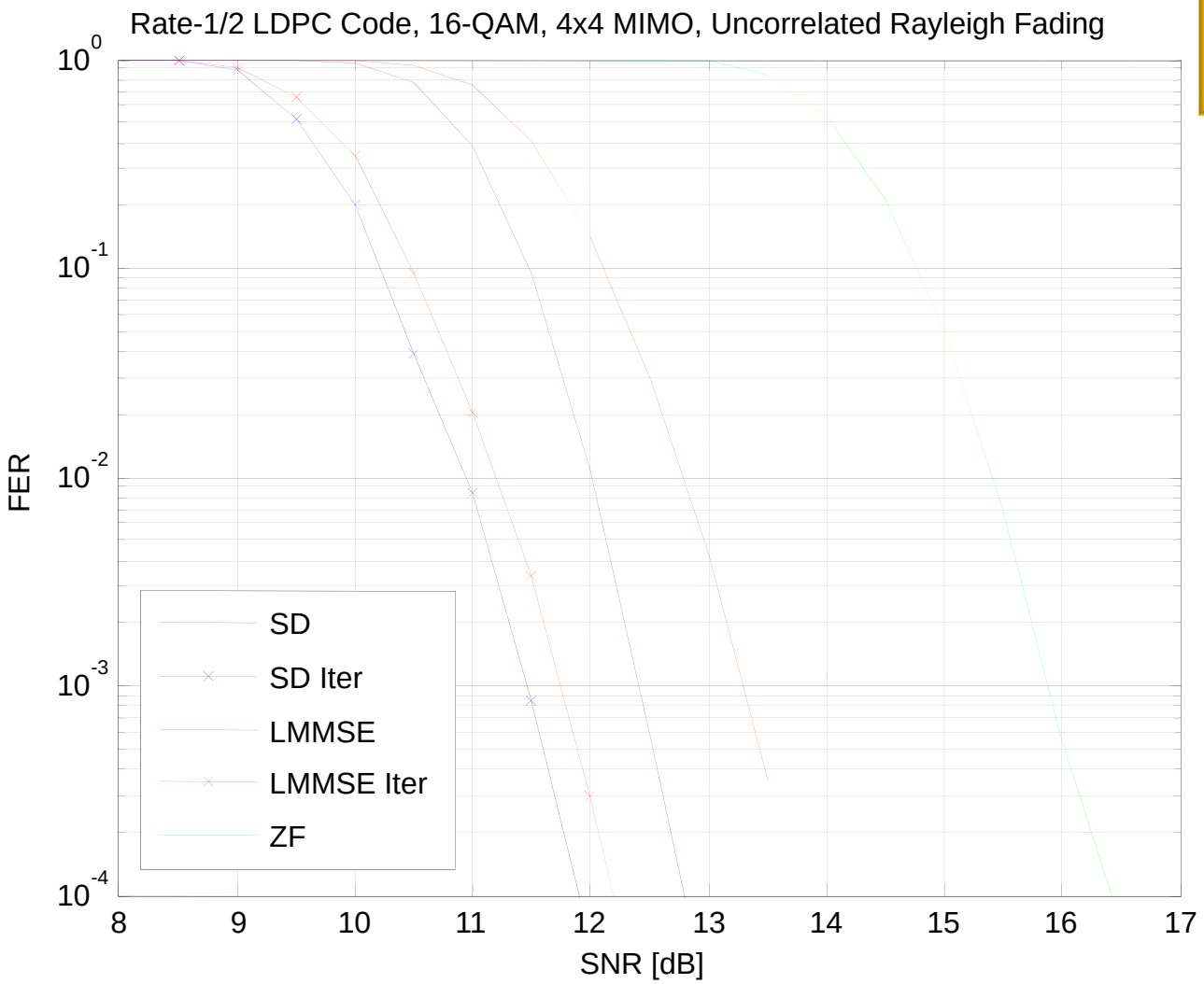
MIMO doubly iterative receiver

- Two iteration types
  - Outer iteration ($\mathcal{OI}$): MIMO detector ⇔ channel decoder
  - Inner iteration ($\mathcal{II}$): inner iteration of channel decoder
- Their execution complexity and time relation

$$\mathcal{C}_{\mathcal{OI}} = a \times \mathcal{C}_{\mathcal{II}} \qquad \mathcal{T}_{\mathcal{OI}} = b \times \mathcal{T}_{\mathcal{II}} \qquad a > 1 \qquad b > 1$$

Performance Comparison

It is worth the effort: iterative receivers yield lower FER !



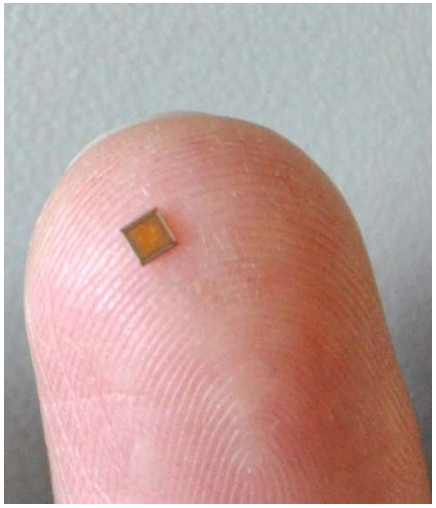
Agenda

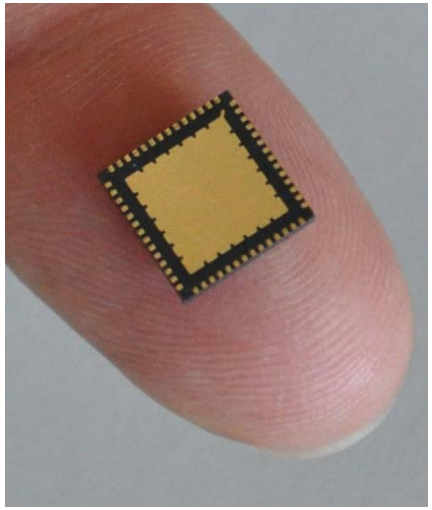
- The issue: iterative MIMO receivers
- **IterX:**
  an iterative demapping/decoding ASIC
- Hardware cost of performance
- Conclusions

A prototype

IteRX

- **A 2.78 mm² 65 nm CMOS Gigabit MIMO Iterative Demapping and Decoding (IDD) Receiver**
  - Lead designers: F. Borlenghi, E. M. Witte
  - in collaboration with A. Burg, EPFL

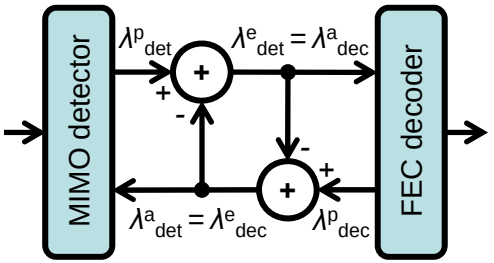


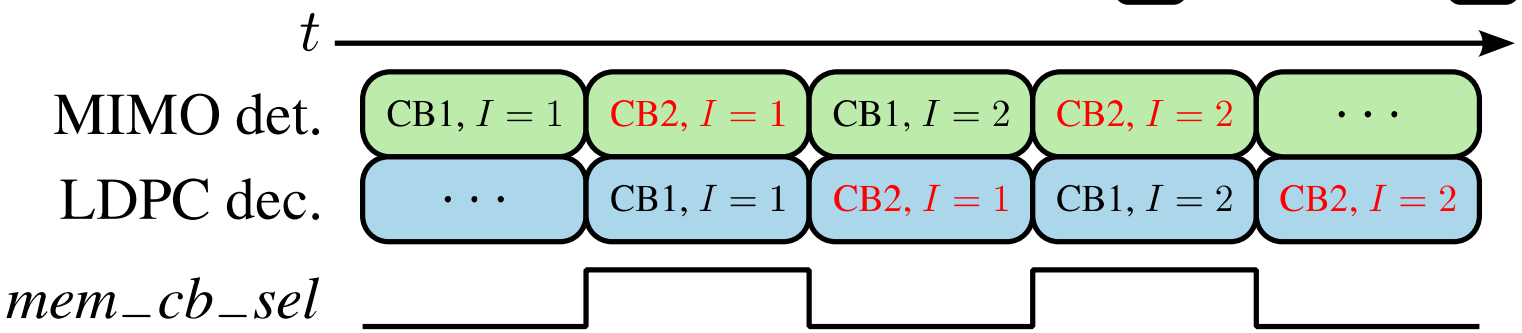


IDD System Architecture

- Codeblock (CB)-wise processing
- 2 interleaved codeblocks processed concurrently

→ Both detector and decoder always busy

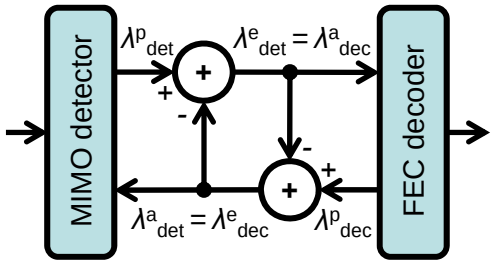


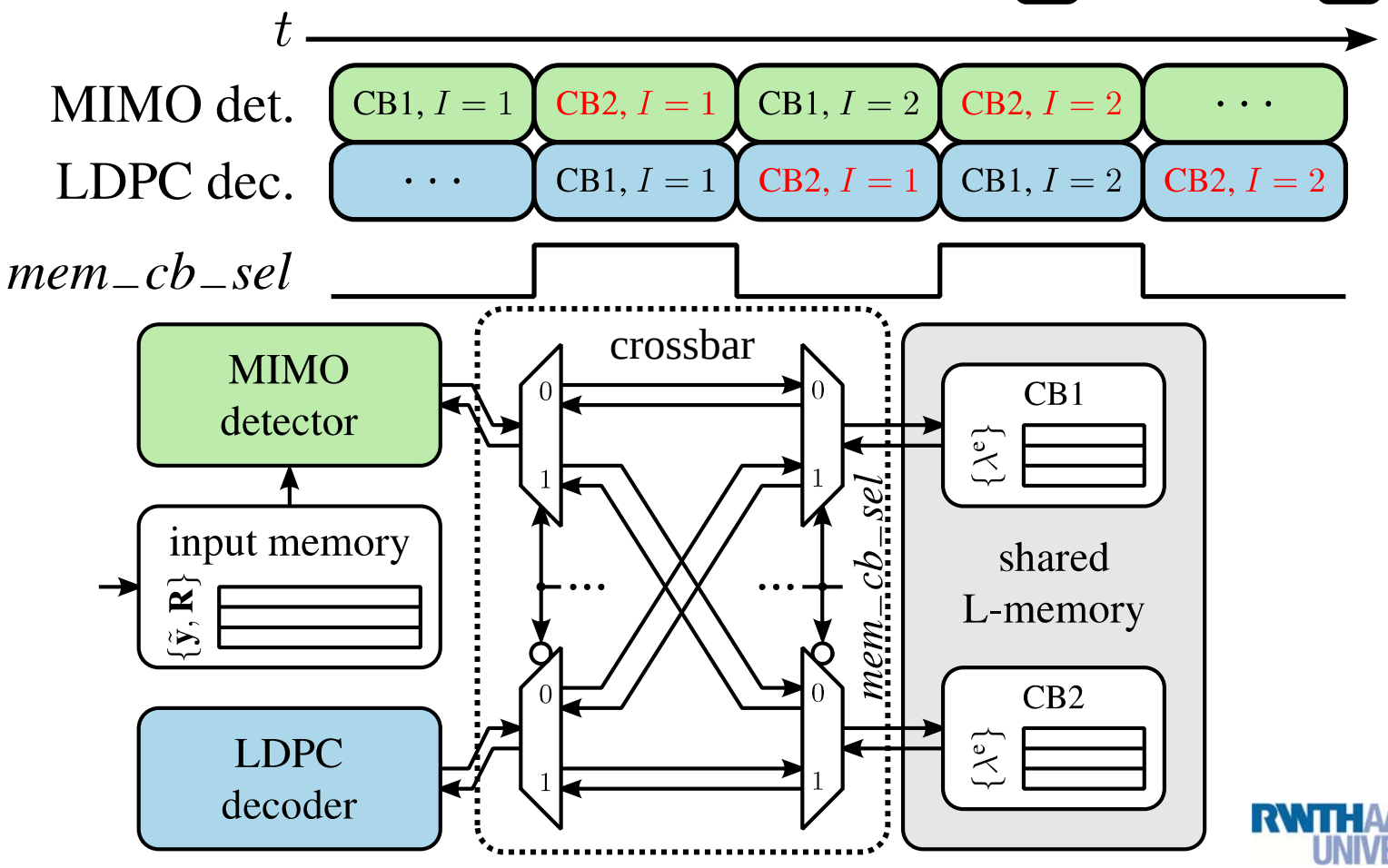


IDD System Architecture

- Codeblock (CB)-wise processing
- 2 interleaved codeblocks processed concurrently

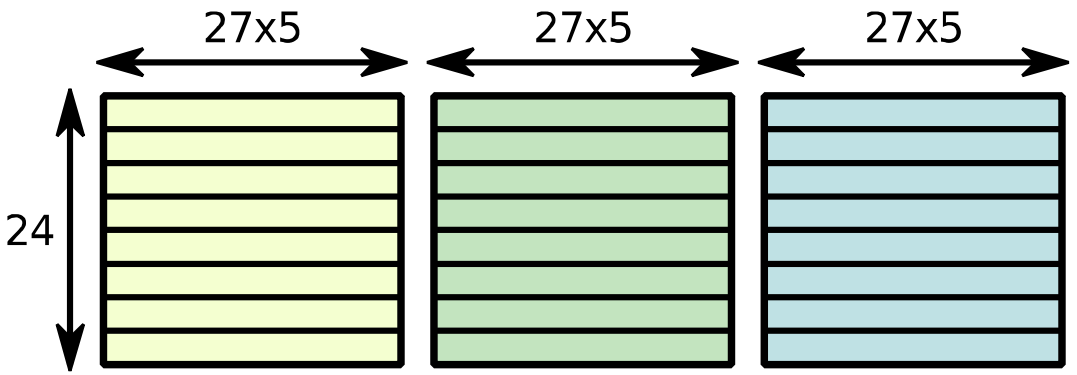
→ Both detector and decoder always busy





How to Write LLRs to the Memory?

- **LDPC requirement:** 81 LLRs (405 bits) per cycle
- **Memory** split in 3 banks matching the 802.11n code-word lengths
  ⇨ Z = {27, 54, 81}

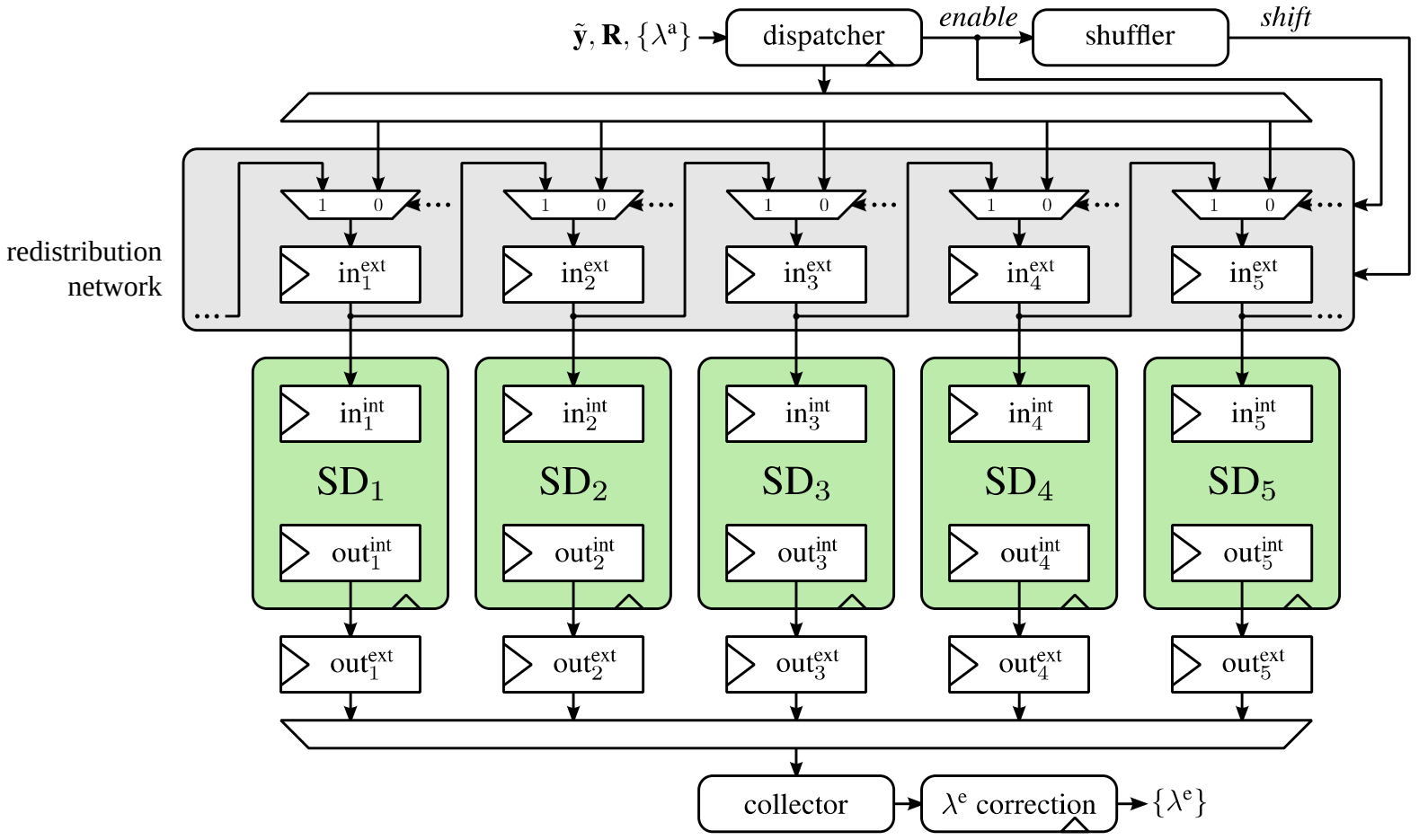


- **Detector access scheme:**
  - Vector of 4 (2x2, QPSK) to 24 (4x4, 64-QAM) LLRs
  - "Random" access due to out-of-order SD output
  - ⇨ **Alignment problem!**
  - Reorder and pack vectors? NO! 1st vector may finish last
  - Serialize access (1 LLR/cycle)? NO! Limits max system throughput

⇨ **Specialized alignment unit and custom memory structure to achieve a 1 vector/cycle throughput**

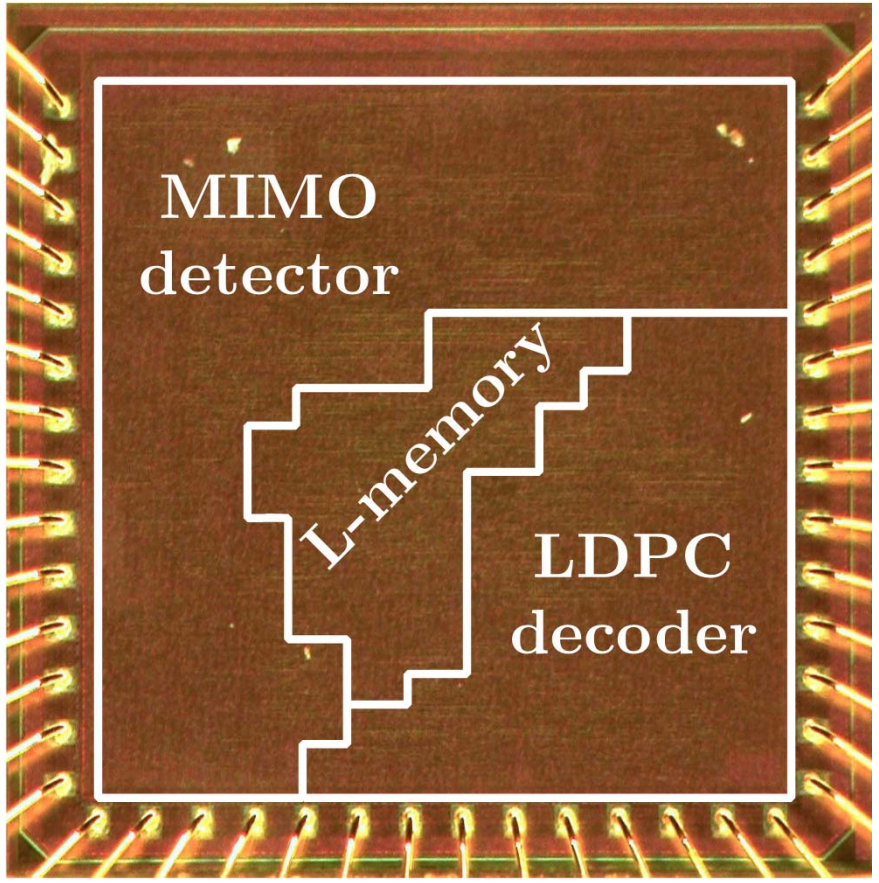
Multicore Sphere Decoder

- Multiple SD cores to sustain throughput in low SNR



Implementation Results

- **First silicon implementation of MIMO IDD baseband**
- **Core area: 2.78 mm² (1.58 MGE) in low-power 65 nm technology**
  - Detector (5 SDs): **872 kGE** (55%), 140 to 145 kGE / SD
  - Decoder: **447 kGE** (28%)
  - LLR memory: **210 kGE** (13%)
- **Runtime flexibility**
  - {2x2, 3x3, 4x4} antennas
  - {4, 16, 64} QAM
  - All 802.11n LDPC codes
- **Max. frequencies @ 1.2 V**
  - Detector: **135 MHz**
  - Decoder: **299 MHz**
- **Avg. power (4x4, 64 QAM)**
  - Detector: **175** to **245 mW**
  - Decoder: **120** to **140 mW**
- **Max. throughput > 1 Gbit/s**



Agenda

- The issue: iterative MIMO receivers

- IterX:
  an iterative demapping/decoding ASIC

- Hardware cost of performance

Algorithmic performance vs. processing energy

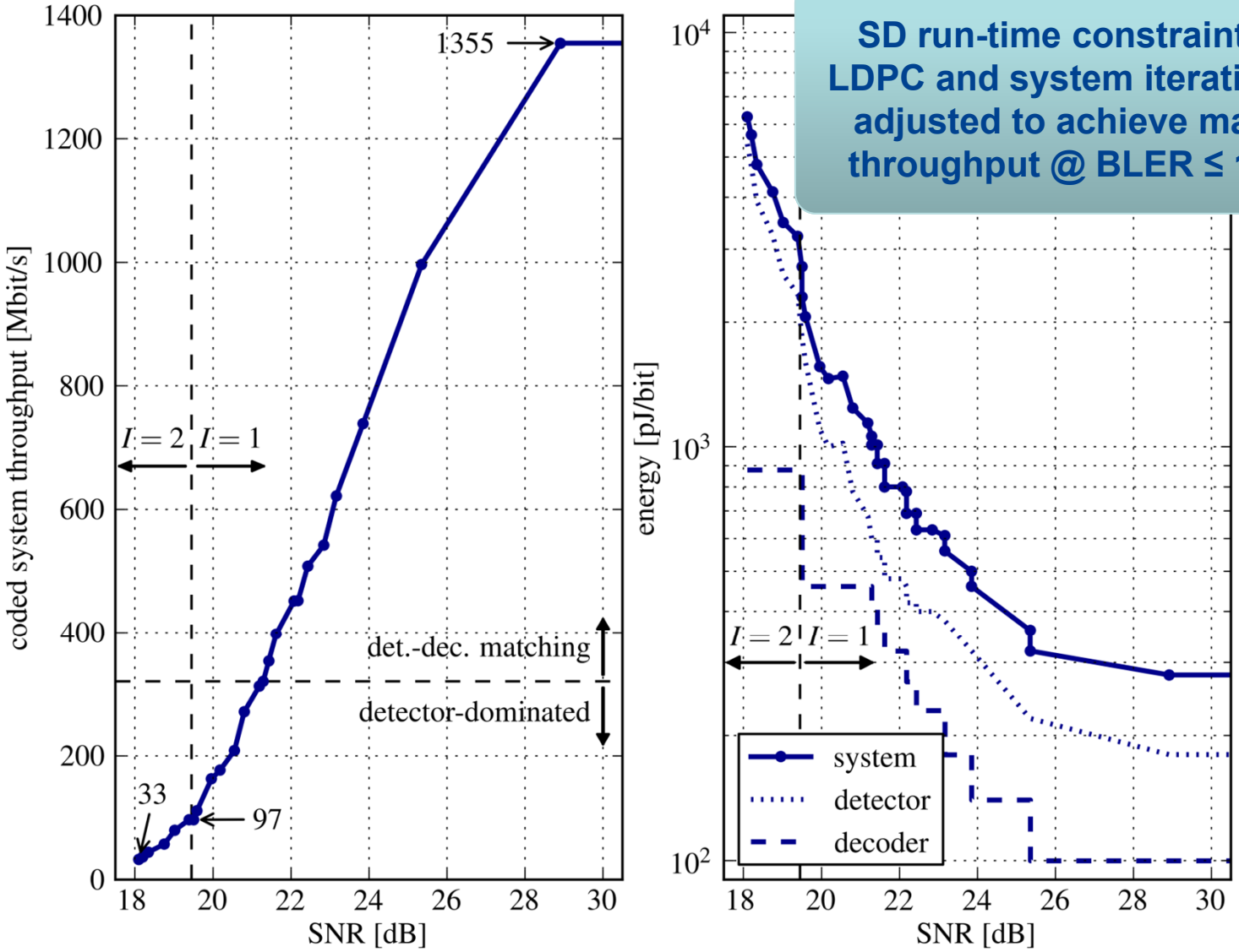
- Conclusions

Execution Time Dependencies

- **SISO MIMO demapper**
  - Sphere detector gets slower with decreasing SNR
    because less branches are pruned (i.e. more are evaluated)
- **LDPC decoder**
  - LDPC decoder gets slower with decreasing SNR
    because more iterations are required for convergence
  - There is a maximum number of reasonable iterations for low SNR
    (if convergence has not yet occured it will likely never be achieved)
- **IDD iterations**
  - Number of necessary iterations increases with decreasing SNR

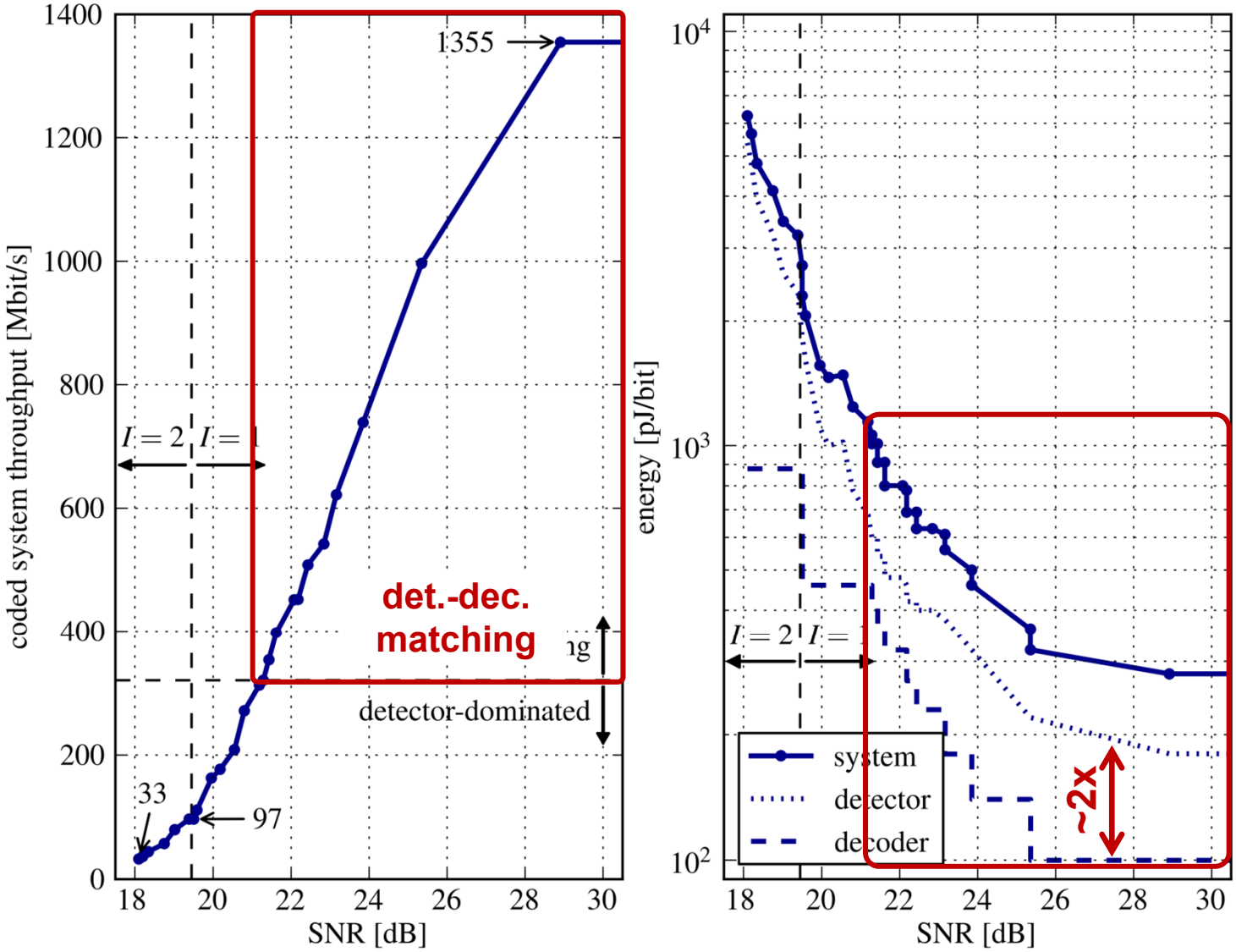
Throughput and Energy Efficiency

SD run-time constraints, LDPC and system iterations adjusted to achieve max. throughput @ BLER ≤ 1%



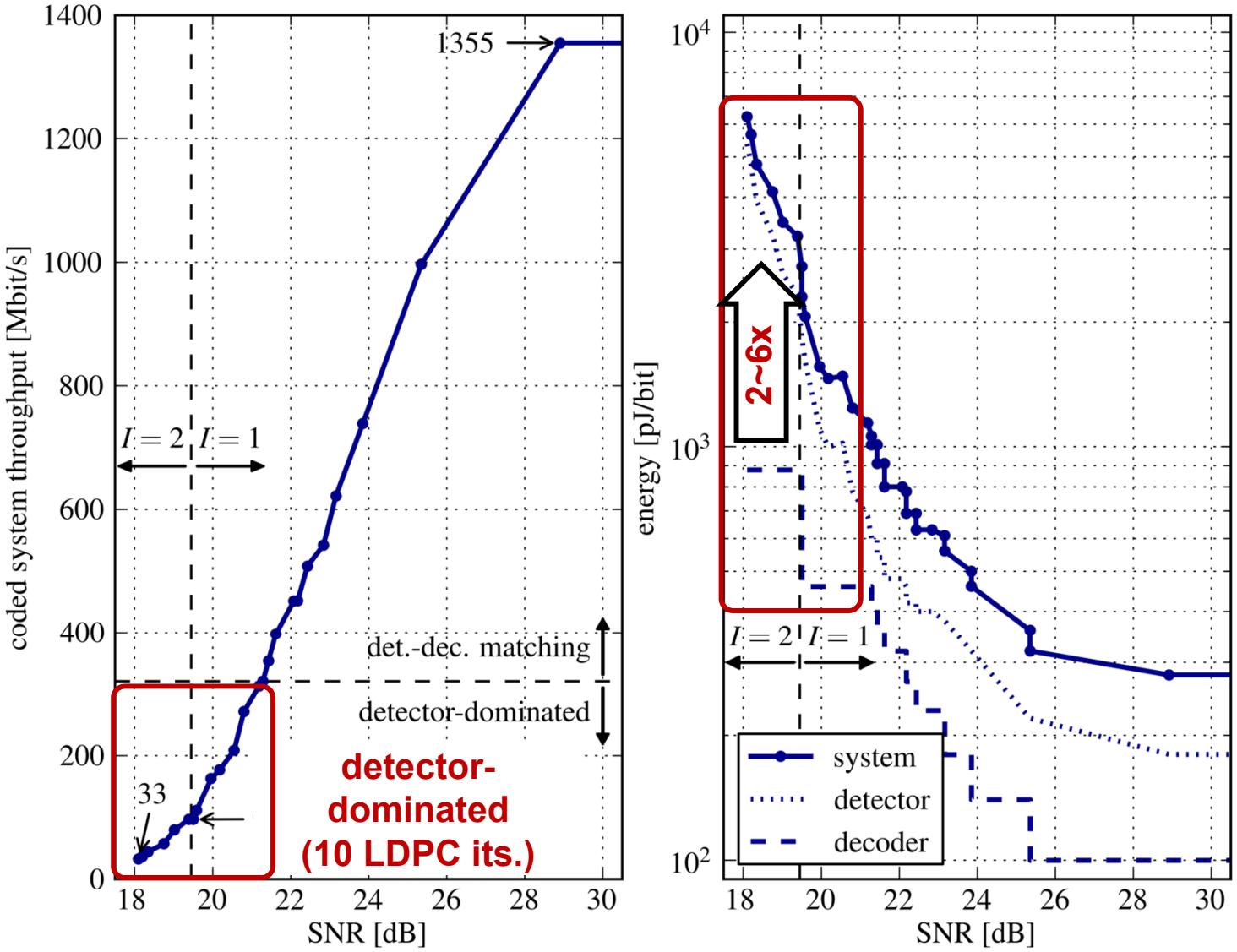
4x4 64-QAM, block length 1944, code rate ½

Throughput and Energy Efficiency



4x4 64-QAM, block length 1944, code rate ½

Throughput and Energy Efficiency



4x4 64-QAM, block length 1944, code rate ½

Agenda

- The issue: iterative MIMO receivers
- IterX:
  an iterative demapping/decoding ASIC
- Hardware cost of performance
- Conclusions

A different design view

The Previous Observations Raise a Question

**What is the maximum throughput
that can be achieved
with given hardware resources
(energy/bit or gate count)?**

Conclusions

- **Power and energy/bit are linked by the throughput:**

  Power = Energy/bit * bit/s

  bit/s: Throughput

- **Going for higher throughput in wireless communications means that we have to work backwards from the power limit (e.g. ~ 1 Watt/Chip).**
  - This tells us how much energy per bit is available
  - It may limit how elaborate processing we can do
  - Example: 100 Gb/s and 1 Watt gives us 10 pJ/bit

- **We need to think different about energy efficient processing**
  - Trade-off between algorithmic performance and processing energy
  - Asynchronous digital and/or analog signal processing ?

Thank you
for your attention !