TOTEM - Tandem Optimised Turbo Encoded Multimedia

Claude Desset

8th March 2007

ESA Microelectronic Presentation Days

imec

Outline

Improving satellite image transmissions

– Better quality vs. rate (and signal power)

Approach: UEP methodology

– Modeling phase
– Optimization phase
– Assumptions

Practical implementation

– Flexwave-II and T@mpo cores
– SW platform & HW integration
– Performance reults

Outline

Improving satellite image transmissions

- Better quality vs. rate (and signal power)

Approach: UEP methodology

- Modeling phase
- Optimization phase
- Assumptions

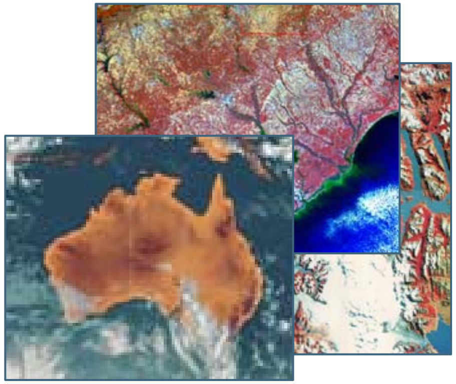
Practical implementation

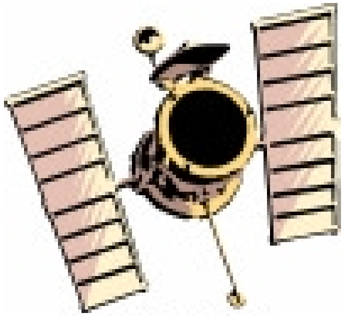
- Flexwave-II and T@mpo cores
- SW platform & HW integration
- Performance reults

Context

Satellite Communications

- Wireless transmission to Earth
- Multimedia: Images





Constrained problem

- Low power
- Low bandwidth
- High quality

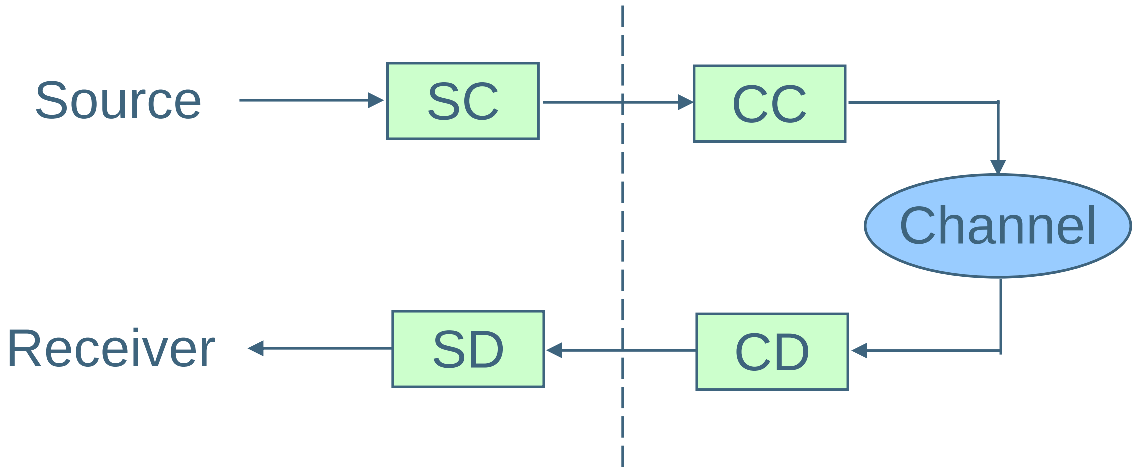
Move to scalable codecs

- Better control of
  - Rate
  - Quality
- More flexible

Joint source-channel coding

Shannon's separation theorem (1948):

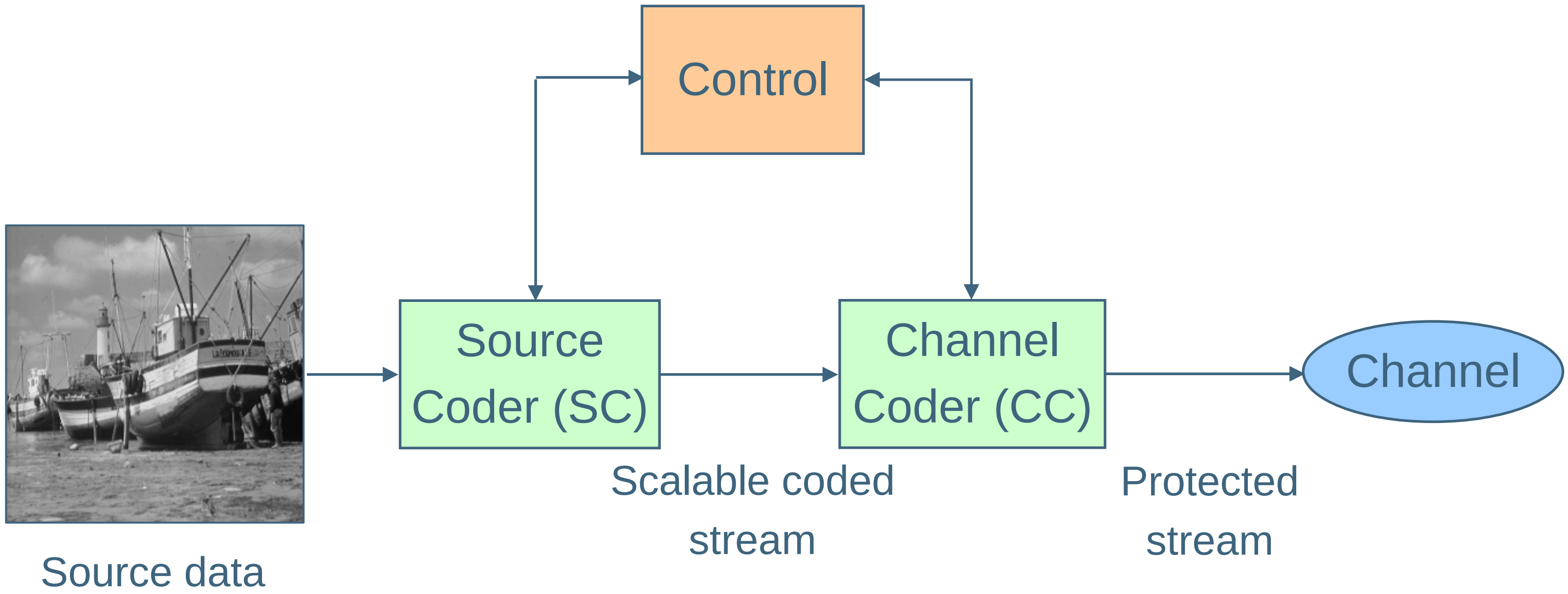
- Perfect transmission with separated Source and Channel Coding



But...

- Assumes infinite block size and unlimited complexity
- Unnecessary intermediate step (just simpler split design)
- Joint solutions lead to better performance

Tandem optimisation



Maximize received quality under rate constraint

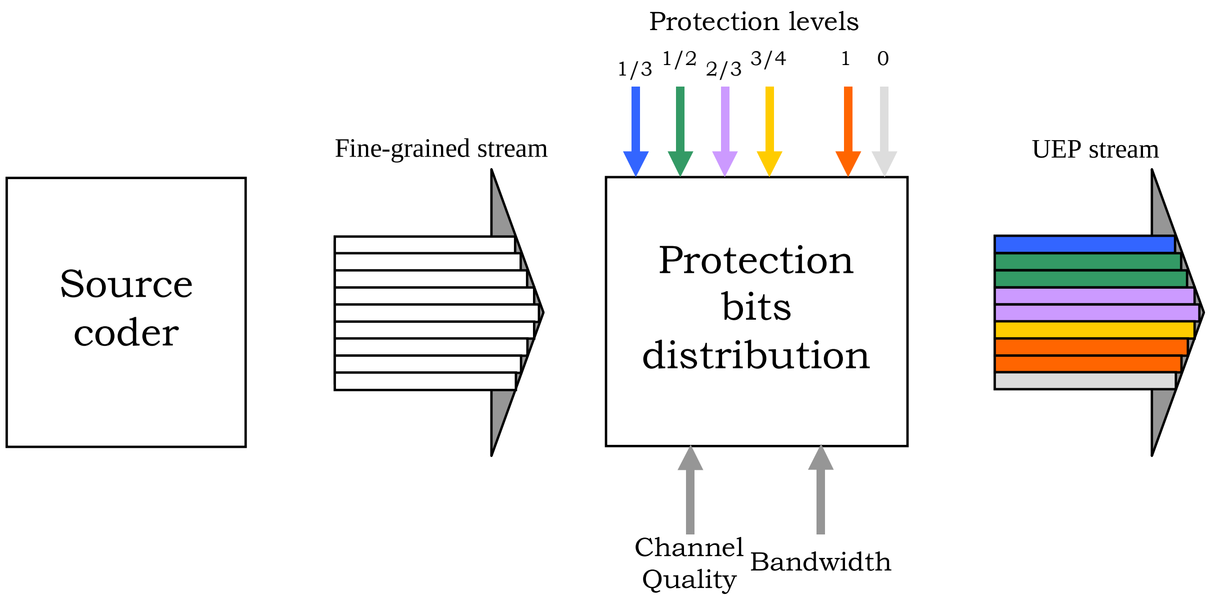
Practical option

- Reuse of existing coders without modification
- More flexible than true joint source-channel solution
- Low complexity

Unequal Error Protection

BW trade-off between Source and Channel Coding

- High compression limits accuracy of source representation
- Low protection means quality largely affected by errors
- Importance-based allocation



Issues in UEP

Typical approach in SoA

- Pick up a "base" and "enhancement" layer
- Pick up two channel codes
- Observe that it improves...

Our target

- Generic solution
  - Avoiding ad-hoc tricks specific to some coders
- Large number of substreams (and channel codes)
  - Exponential number of possibilities
  - Need for stronger methodology (not just ad-hoc)
- Pragmatic solution
  - Avoid extremely complex integer programming
  - Better exploit the knowledge of the problem

Outline

Improving satellite image transmissions

- Better quality vs. rate (and signal power)

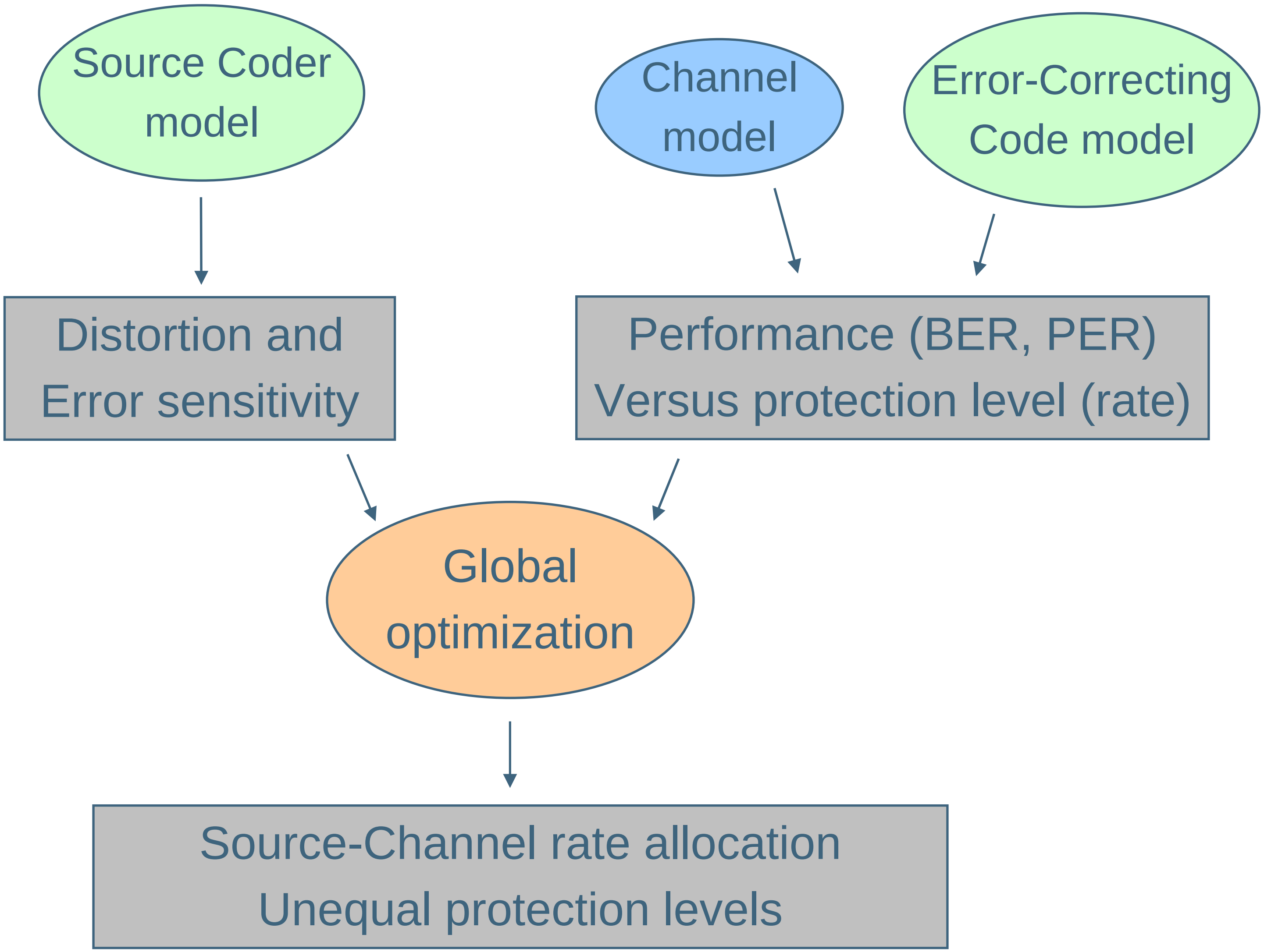
Approach: UEP methodology

- Modeling phase
- Optimization phase
- Assumptions

Practical implementation

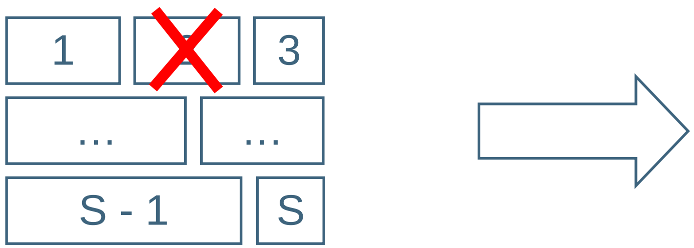
- Flexwave-II and T@mpo cores
- SW platform & HW integration
- Performance reults

UEP methodology (overview)



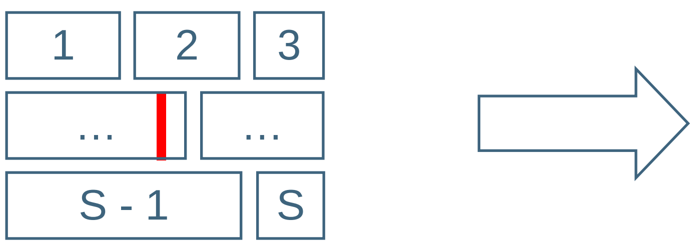
Source Model

On each substream:



D_{cut}

Distortion resulting from removing a substream

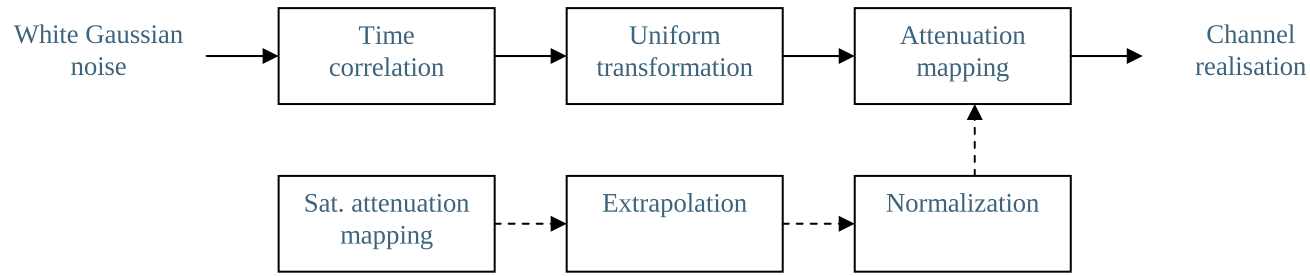


D_{bit}

Distortion resulting from inserting random bit error into a substream

All distortions computed over the whole image

DVB satellite channel model



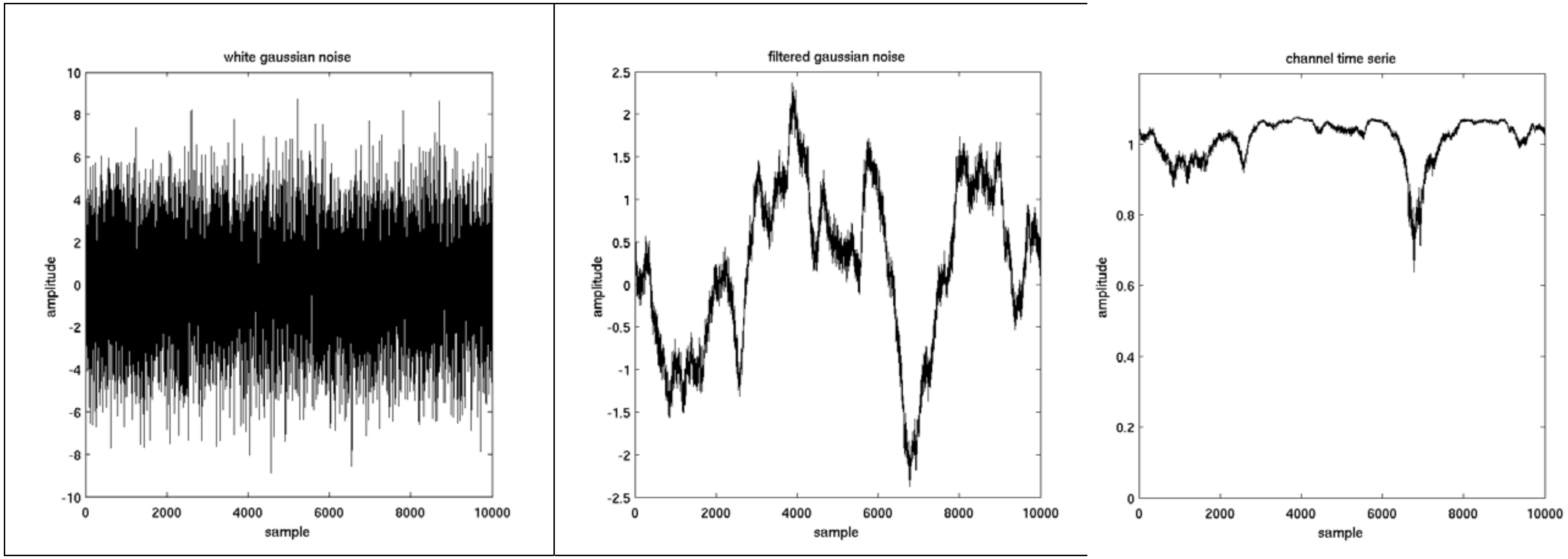


Figure 1: White Gaussian noise (left) and filtered version (right).

BER for selected channel and channel coder

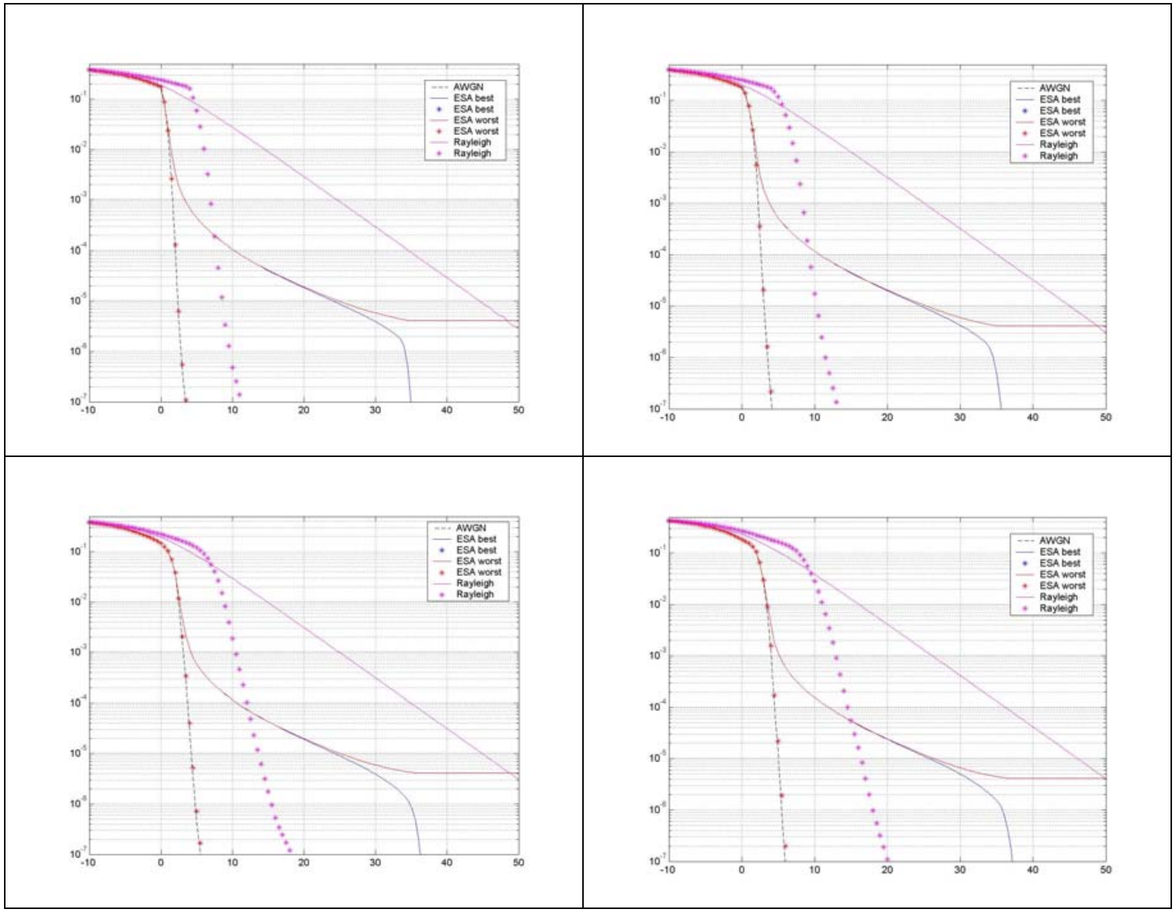
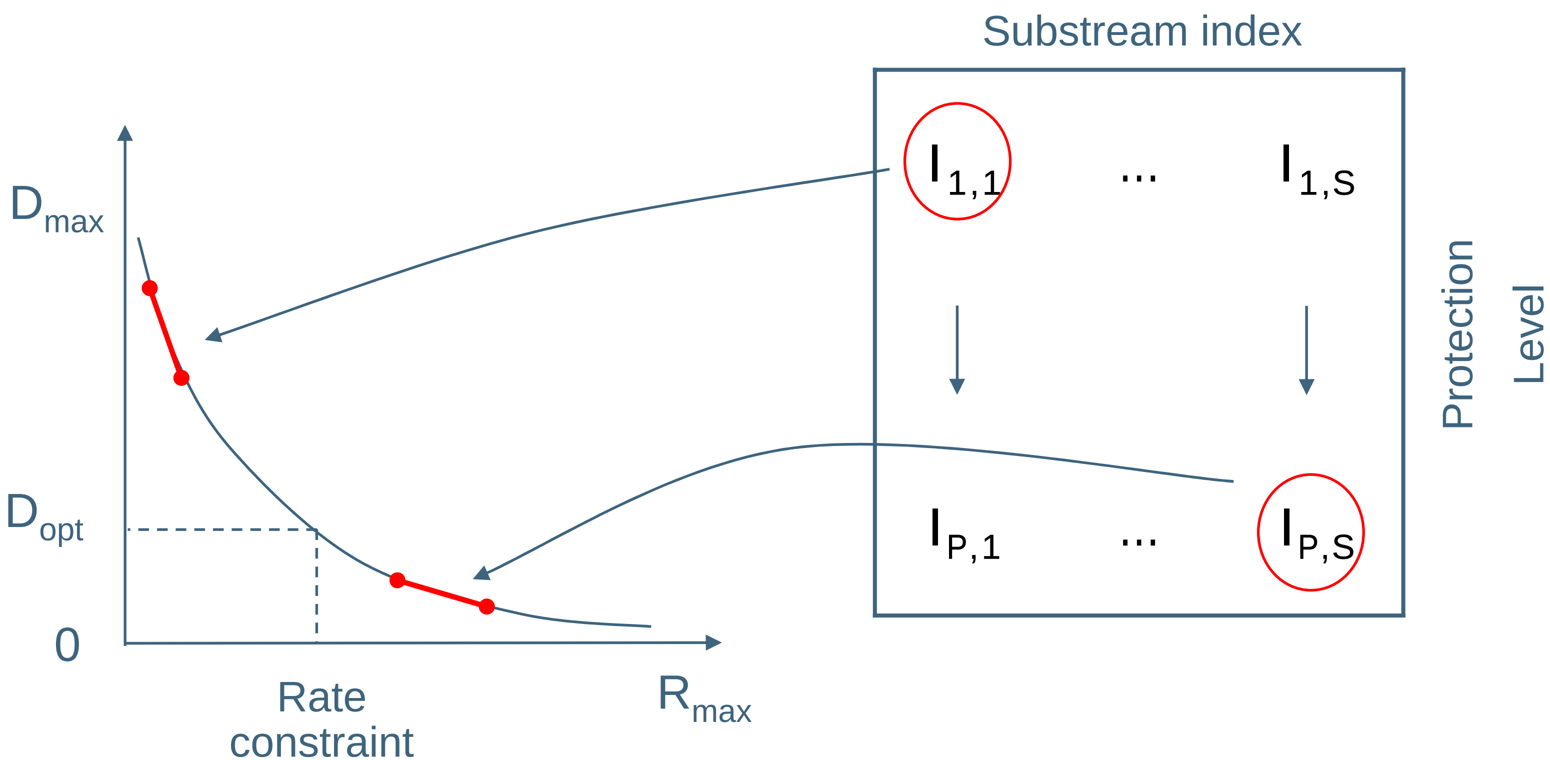


Figure 10: Performance for fast fading (stars) and slow fading (solid curves), both Rayleigh (magenta) and ESA channel (red/blue). Coderate is 1/3 (upper left), 1/2 (upper right), 2/3 (lower left) and 3/4 (lower right).

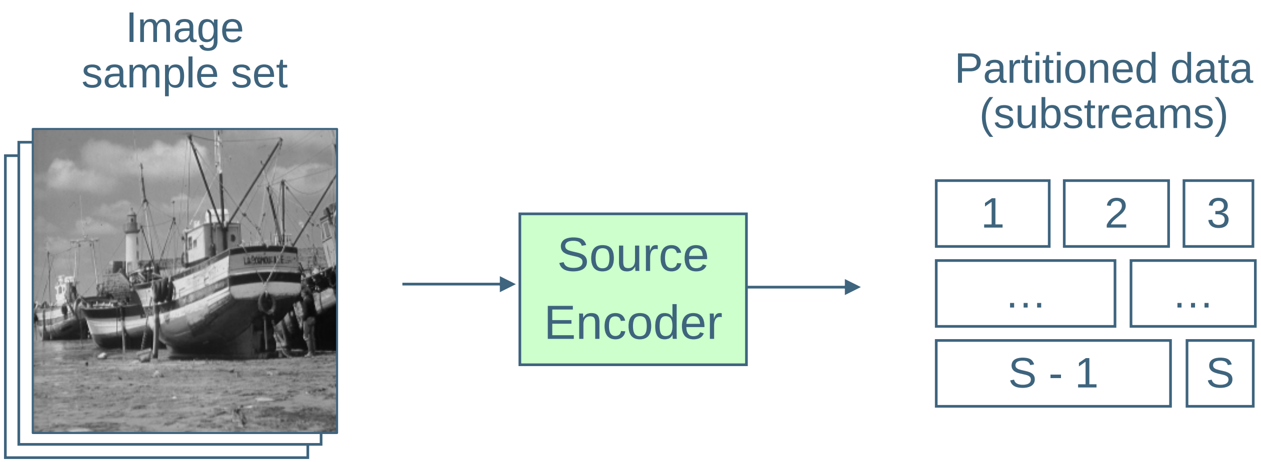
Global optimisation

Global model for each substream $D = D_{cut} + p \cdot L \cdot D_{bit}$

Sorting of specific importance values $I = \frac{\Delta D}{\Delta R}$



General assumptions



Source scalability

Substreams have uniform importance internally

Substream sizes are error-free (short headers)

Channel error independence

Central assumption: distortion additivity

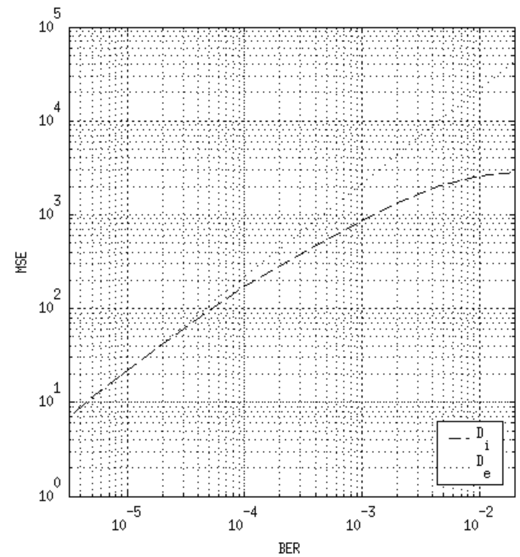
Additivity over substreams

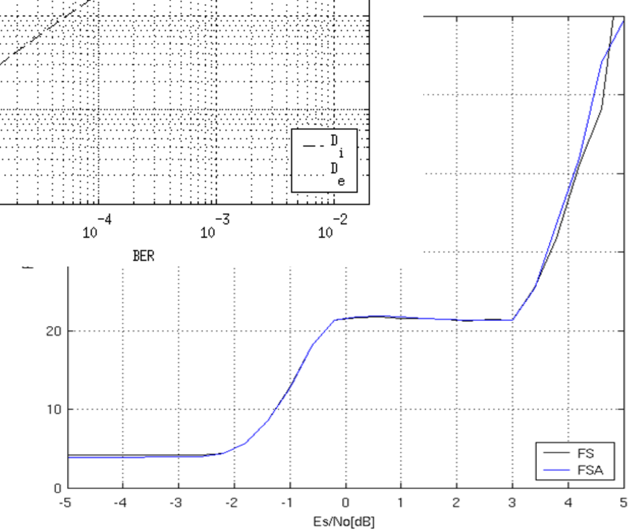
$$D = \sum_{substreams} D_i\left(q_c, PL_i\right)$$

- Huge complexity reduction

Validation in two ways

- Analysis of the deviation (additivity mismatch)
- Comparison to full search on a simplified example





Linearity inside substreams

- Allows further simplification of the modeling phase

Outline

Improving satellite image transmissions

- Better quality vs. rate (and signal power)

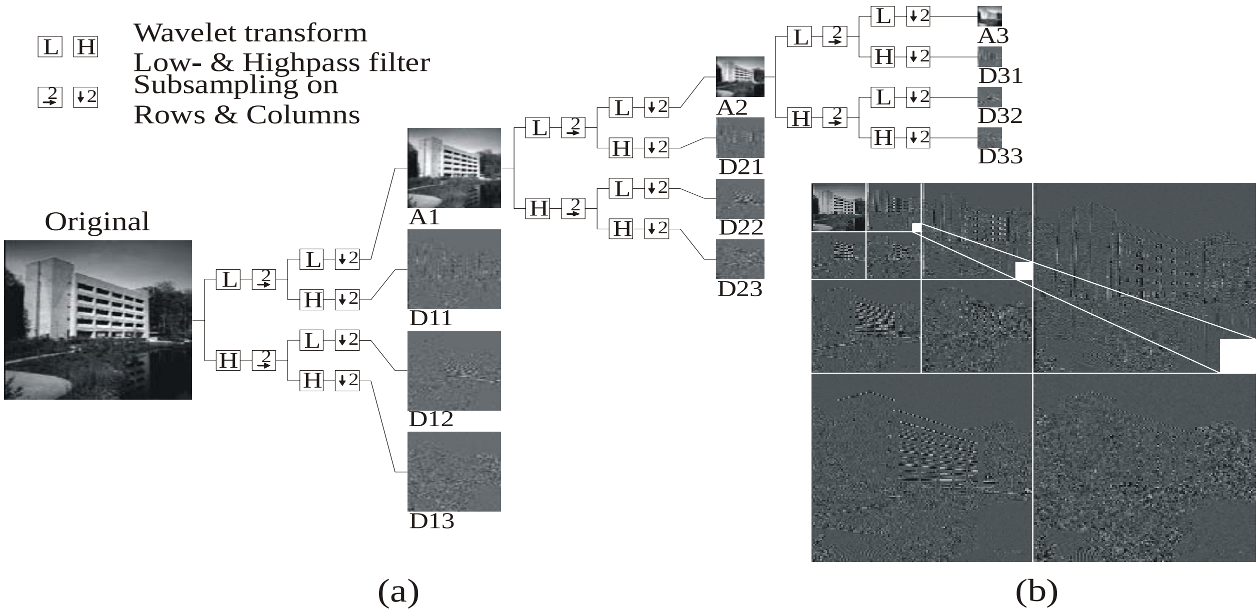
Approach: UEP methodology

- Modeling phase
- Optimization phase
- Assumptions

Practical implementation

- Flexwave-II and T@mpo cores
- SW platform & HW integration
- Performance reults

Source coder: Flexwave-II (wavelet-based)



Dedicated component for space applications

Gradual enhancement / degradation of the image (scalability)

Fine granularity (400 substreams)

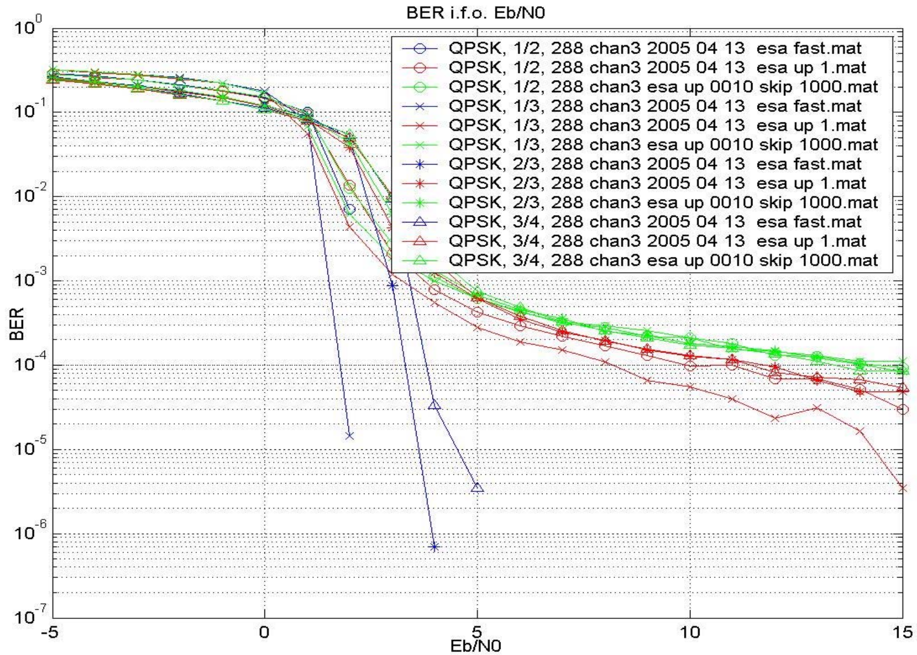
Good compression ratio (between x2 and x40)

Channel coder: T@mpo (turbo codec)

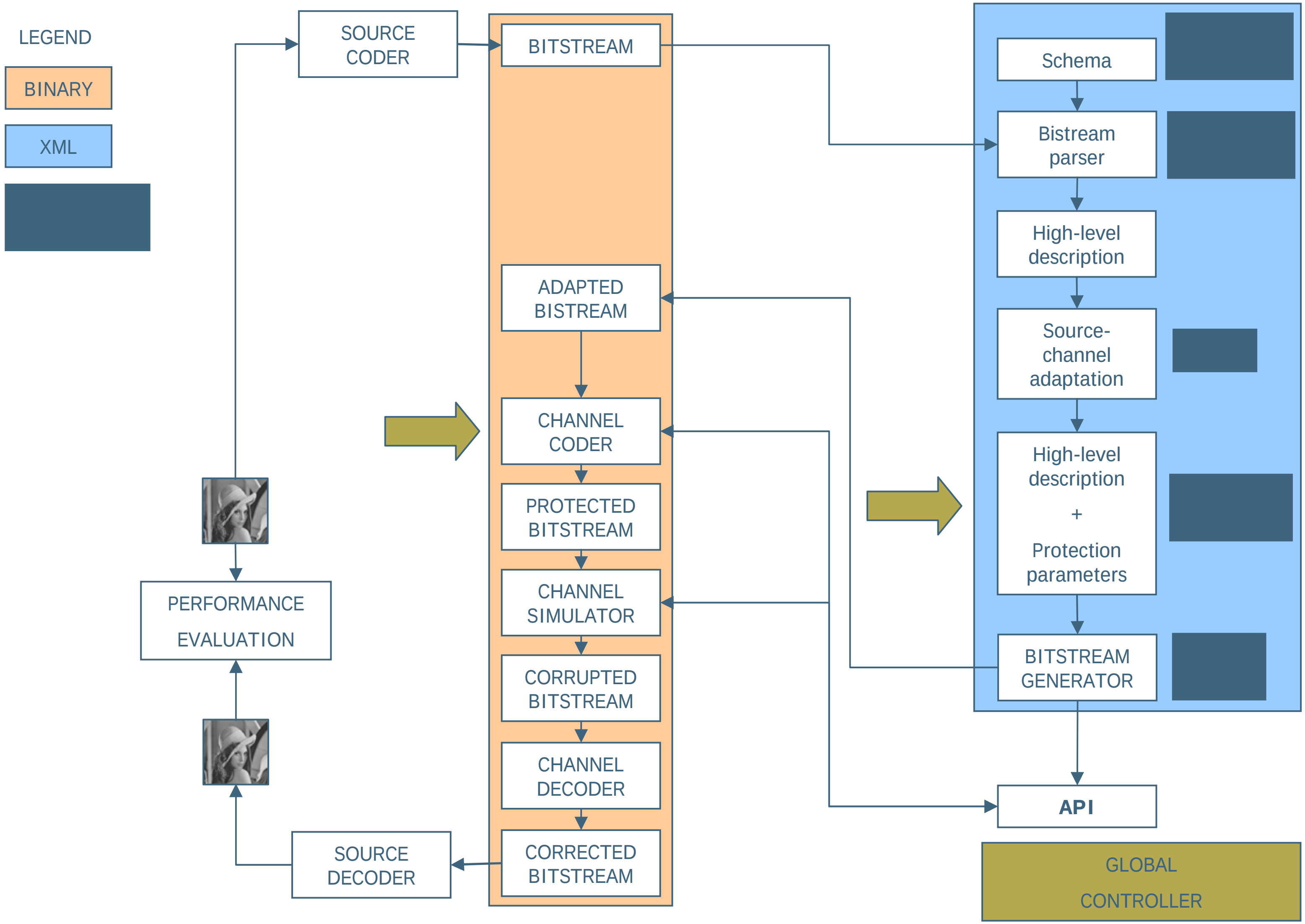
Pre-existing chip, selected parameters:

- Maximum length (288) for best performance
- Various rates for UEP: 1/3, 1/2, 2/3, 3/4

Behaviour on top of fading channels explored



SW Framework Structure



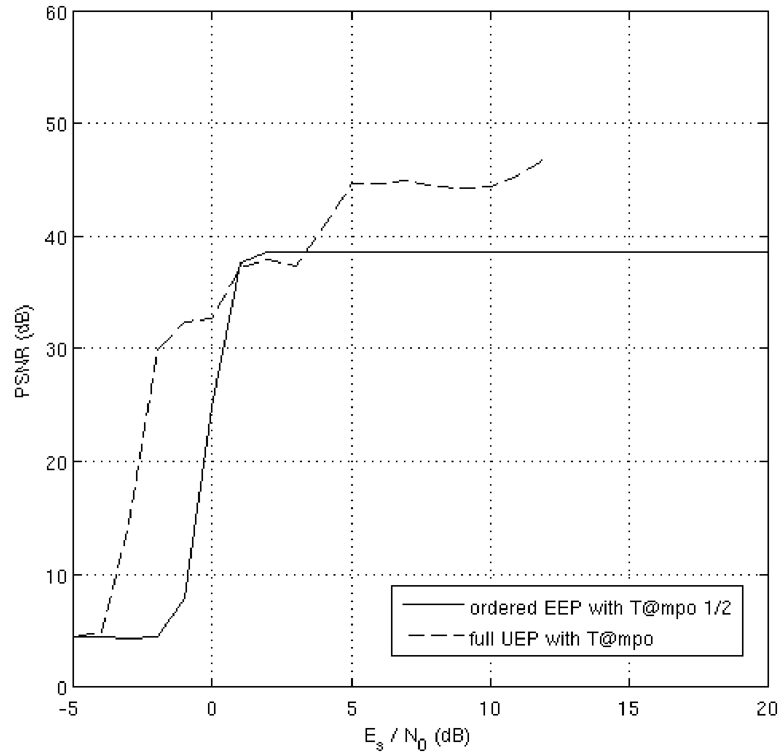
Comparison between UEP and EEP

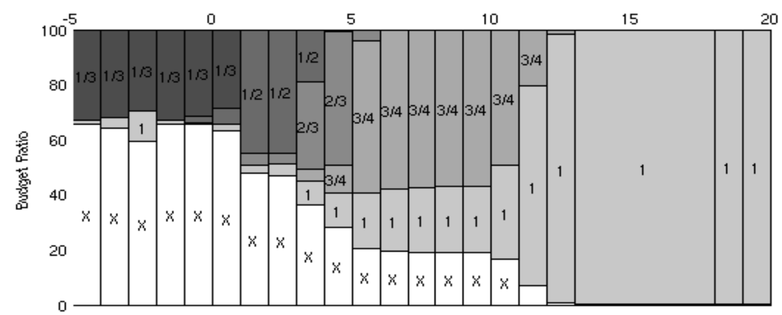
Setup

- T@mpo
  - 4 levels for UEP
  - 1 level for EEP (1/2)
- AWGN channel
- 100% budget

Conclusions

- UEP outperforms EEP
- EEP overprotects the codestream on good channels
- EEP underprotects the codestream on bad channels
- UEP dynamically adapts to the channel, while EEP is only good on a small channel range

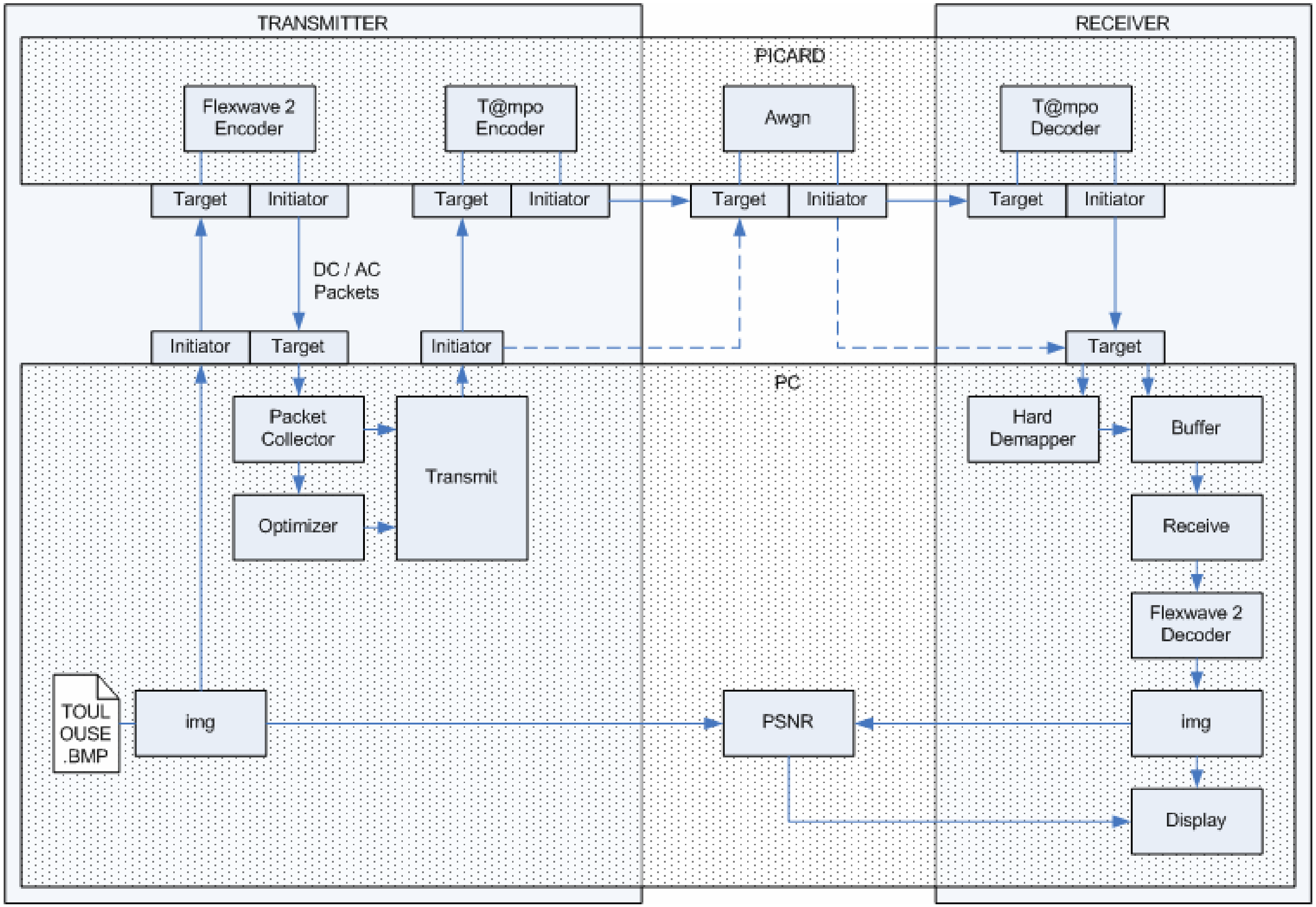




Internal structure of the HW Platform

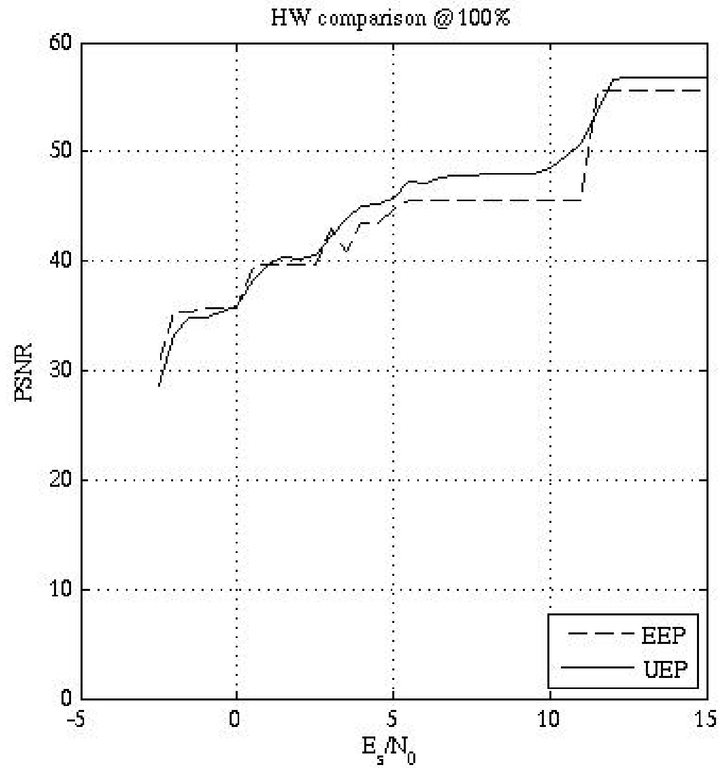
HW

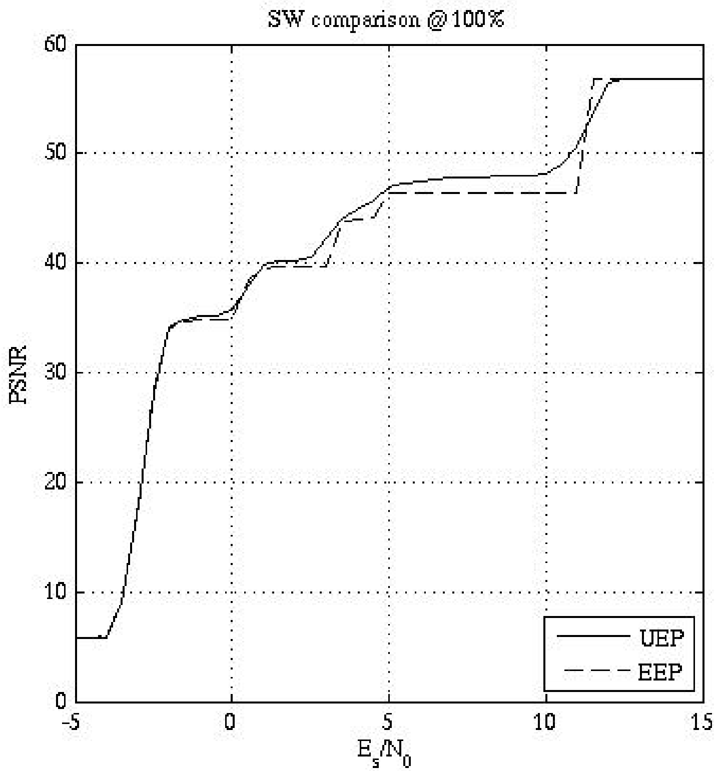
SW



The HW platform after transmission

Final validation (HW/SW, optimized UEP/EEP)





UEP outperforms EEP by 1-2dB

- Non-linear UEP extension
- Advanced EEP (run-time selection of best code, more complex)

Excellent match between HW and SW

Conclusions

UEP improves (rate, power, quality) trade-offs

- Best split of rate between source and channel coders
- 1-2 dB extra gain thanks to differential protection

Practical solution thanks to our methodology

- Design-time modeling of source and channel
- Low-complexity run-time selection (sorting-based)

Generic solution

- Illustrated on pre-existing source and channel codecs
- SW and HW demonstration validating the approach

aspire invent achieve

imec