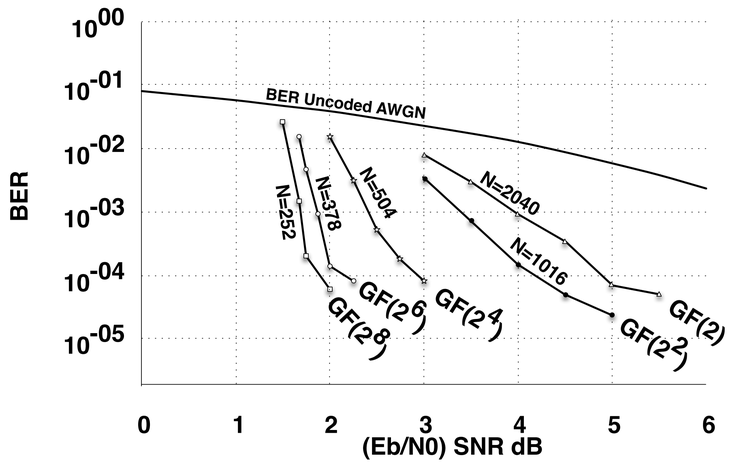
FFT-SPA NON-BINARY LDPC DECODING ON GPU

Joao Andrade[1], Gabriel Falcao[1], Vitor Silva[1], Kenta Kasai[2]

[1] Instituto de Telecomunicações, Dept. of Electrical and Computer Eng., University of Coimbra, Portugal

[2] Dept. of Comm. and Integrated Systems, Graduate School of Science and Eng., Tokyo Institute of Technology, Japan

LDPC over GF(q) Rationale



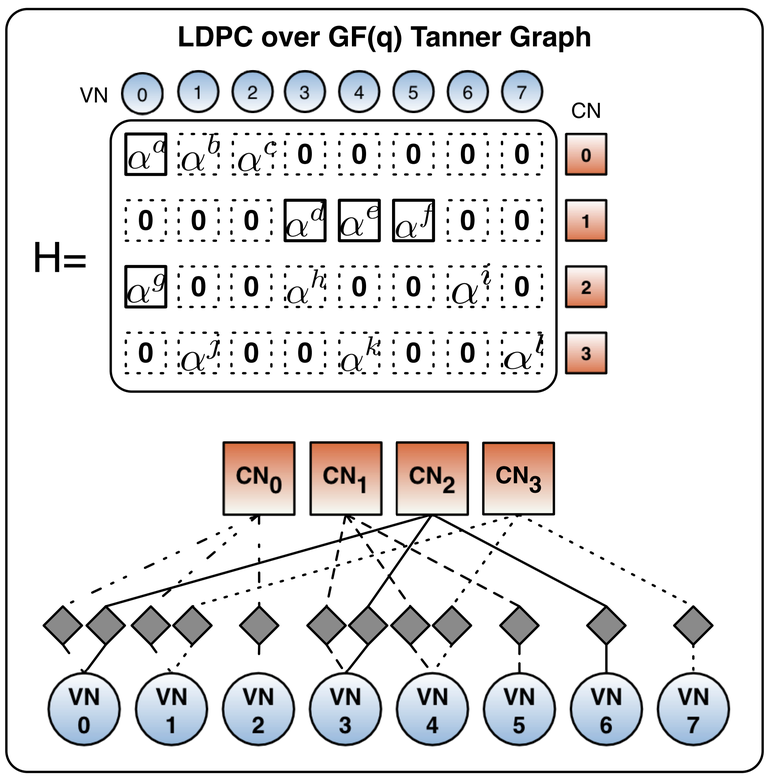
- For fixed block lengths, the higher is q, the better the performance.
- Channel capacity through q instead of N.

Problem?

- Complexity scales non-linearly with q.
- Developing dedicated hardware every time we need to test a new code? → high prog. effort.

FFT-SPA for LDPC Codes over GF(q)

- Tanner graph of GF(q) LDPC accounts for h_{ij} connecting BN_i to CN_j.



FFT-SPA over GF(q)

Initialization

$$p_v(x) = P(v_v = x|y_v) m_{cv}^{(0)}(x) = p_v(x)$$

(multC) Check-Node Processing

$$m_{cv}^{(i)}(z) = \prod_{v' \in V_c \setminus v} \mathrm{FHT}\{m_{v'c}^{(i-1)}(x)\}$$

$$m_{cv}^{*(i)}(z) = m_{cv}^{(i)}(z) / m_{cv}^{(i)}(z=0)$$

$$m_{cv}^{(i)}(x) = \mathrm{FHT}\left\{m_{cv}^{*(i)}(z)\right\}$$

(inv_edge) Depermute $m_{cv}^{(i)}(x)$ *pmfs* w/ h_{cv}

(multV) Variable-Node and APP Processing

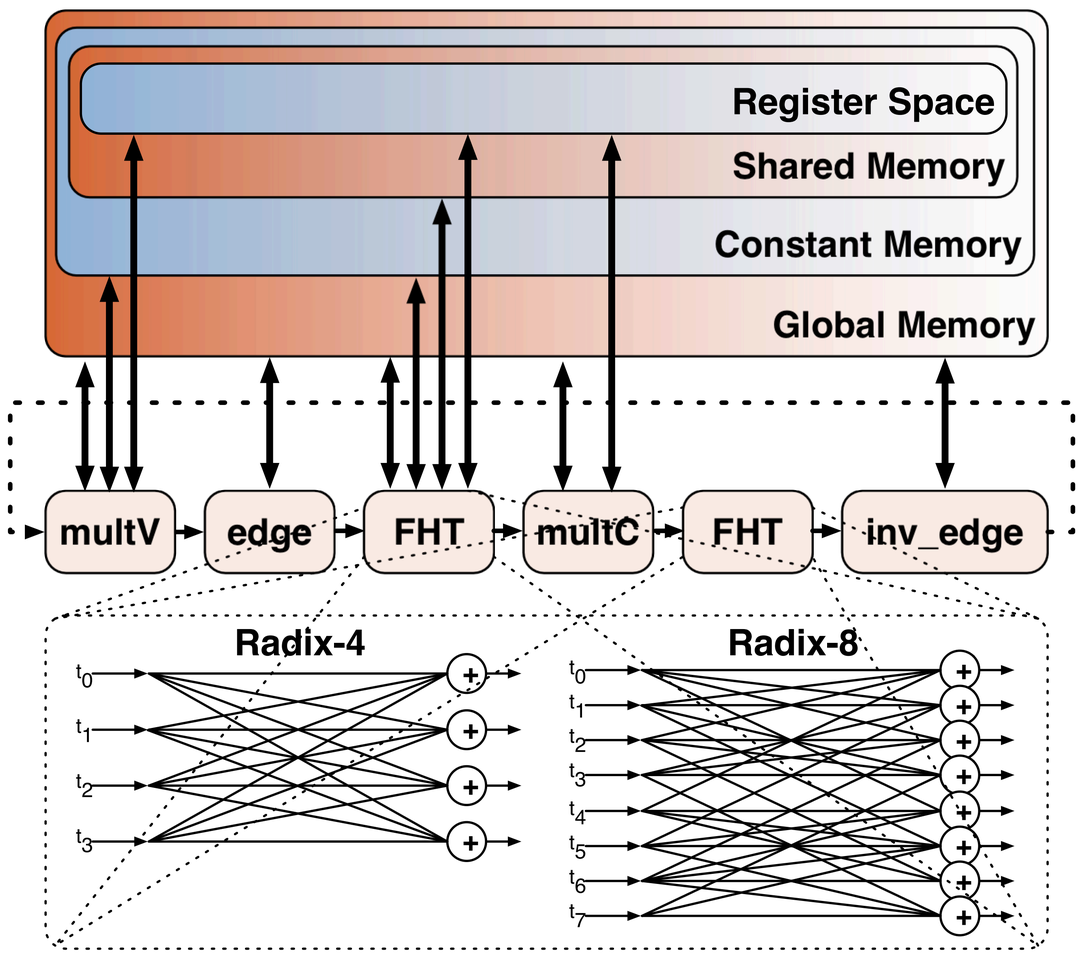
$$m_{vc}^{(i)}(x) = p_v(x) \prod_{c' \in C(v) \setminus c} m_{c'v}^{(i)}(x)$$

$$m_v^{(i)}(x) = p_v(x) \prod_{c' \in C(v)} m_{c'v}^{(i)}(x)$$

(edge) Permute $m_{vc}^{(i)}(x)$ *pmfs* w/ h_{vc}

- Parity-check evaluation complexity grows the Galois Field dimension q.

Parallel FFT-SPA Decoder on the GPU



Why the Fourier domain?

- Normalize *pmfs* on Fourier domain with DC term.
- Sum-Product Algorithm (SPA) follows $O(M \times d_c \times 2^{q \times d_c})$.
- FFT-SPA follows $O(M \times d_c \times q \times 2^q)$.

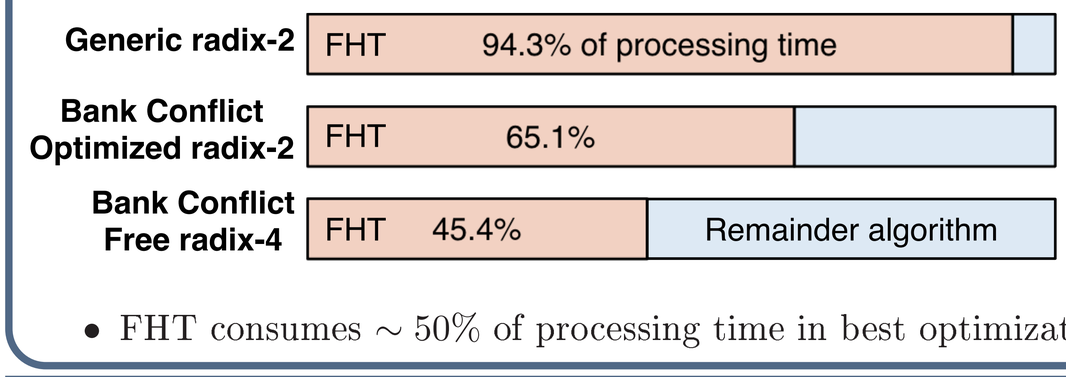
Exploring the GPU parallel features

- GPUs are well-suited accelerators for GF(2), are they for GF(q)?
- GF(q) require extra effort. Where is the most intensive kernel?
- CN and BN processing are Hadamard products of *pmfs*.
- Fourier Transform over $\mathrm{GF}(2^m)$ is the Hadamard Transform.

Challenge?

- Explore radix-n or mixed-radix Fast Hadamard Transforms (FHTs).
- Exploit the memory hierarchy to its fullest → shared memory of 32 banks.

Bottleneck: Fast Hadamard Transform



- FHT consumes ∼ 50% of processing time in best optimization.

Experimental Results

Galois Field	GF(128)			GF(256)		
Iterations	5	10	15	5	10	15
Throughput [Mbit/s]	0.42	0.21	0.14	0.26	0.13	0.08
				1.52	0.77	0.52
	3.34	**1.67**	**1.11**	**2.37**	**1.20**	**0.79**

- Proposed decoder compares w/ non-Fourier domain decoder [1].

References & Acknowledgements

[1] G. Wang, H. Shen, B. Yin, Y. Sun, and J. R. Cavallaro, *Parallel nonbinary LDPC decoding on GPU*, Proc. 46th Asilomar Conference, Nov. 2012.

This work was supported by Fundação para a Ciência e Tecnologia grants PEst-OE/EEI/LA0008/2011 and SFRH/BD/78238/2011, and by the Storage Research Consortium.

Outlook

- 5 years of GF(2) LDPC dec. on GPUs.
- 6 months of GF(q) dec. on GPUs...
- Throughput convergence (partial or total) is expected between GF(2) and GF(q) on GPUs.

· U C ·
FCTUC FACULDADE DE CIÊNCIAS E TECNOLOGIA UNIVERSIDADE DE COIMBRA
TOKYO TECH Pursuing Excellence
QREN QUADRO DE REFERÊNCIA ESTRATÉGICO NACIONAL PORTUGAL 2007.2013
POPH
UNIÃO EUROPEIA Fundo Social Europeu
it instituto de telecomunicações