Implementation of Raptor Code on GPU

Linjia Hu1, Todor Mladenov2, Saeid Noshabadi3
1Department of Computer Science, Michigan Technological University, MI, USA
2Department of Information and Communication and Department of Nanobio Materials and Electronics, Gwangju Institute of Science and Technology, South Korea

Raptor Code principle
Raptor Code comes as an improvement to LT-Code, which performs as close as possible to the Shannon’s channel limit and provides linear encoding and decoding time. It has been chosen for the forward error correction (FEC) scheme in 3GPP and DVB-H standards. The block diagram shows the systematic Raptor encoder and decoder.

Matrix Inversion on GPU
We try to implement Raptor Codes on GPU for the purpose of processing large block and symbol size efficiently. Recommended by the 3GPP and DVB-H standard, the maximum block size is 8192, and maximum symbol size is 8192 bytes. For the large matrix, we use two types of data to memory mapping. One is “WORD”, which uses 32-bit word to store 1 bit matrix element, and the other is “PACKED WORD,” in which 32 matrix elements are packed together into a single 32-bit word.

Conclusion
Our test platform uses a 3.20 GHz Intel Core i7 quad-core CPU, a GeForce GTX 570 graphic card with 2.5 GB video memory, CUDA 4.0 and the Fedora 13 operating system. For large block size and symbol size, the speedup of PACKED WORD version decoding can approach 36, and the speedup of WORD version decoding can approach 46.

Reference