Performance of Parallel Two-Pass MDL Context Tree Algorithm

Nikhil Krishnan and Dror Baron

Department of Electrical and Computer Engineering, North Carolina State University

Generously supported by

Motivation

Bottlenecks of computer/ communication systems:

- Data storage.
- Data transfer.

Compression can reduce storage and transfer requirements.

? Can parallelization help us achieve our goals of

Speed: compress length-N data in O(N/B) time with B processors?
Quality: almost as good as best serial methods?

Naïve parallelization that partitions data into independent blocks compresses poorly.

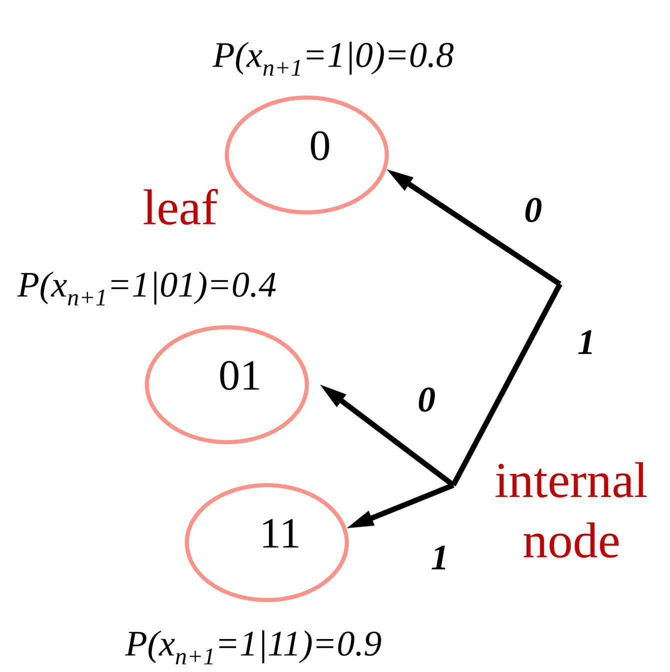
Background

Best serial universal compression has already reached bounds on compression quality [1].

Existing parallel algorithms have high computational complexity and/ or poor compression quality [2].

Source Model (Context Tree):

- Binary alphabet $X=\{0,1\}$, input sequence $x^N \in X^N$.
- State = context (past several symbols) + conditional probabilities.
- Properties:
  - ✓ More flexible than Bernoulli.
  - ✓ More compact than Markov.



Universal Compression:

- Unknown source model.
- ***Redundancy, $\rho(\cdot)$:*** excess coding length above entropy $\rho(x^N) = l(x^N) - NH$; $l(x^N)$ – coding length, H – entropy rate.
- ***Rissanen's bound:*** $\rho(x^N)$ lower bounded by $\approx 0.5log(N)$ bits per unknown parameter.

Two-Pass MDL Algorithm

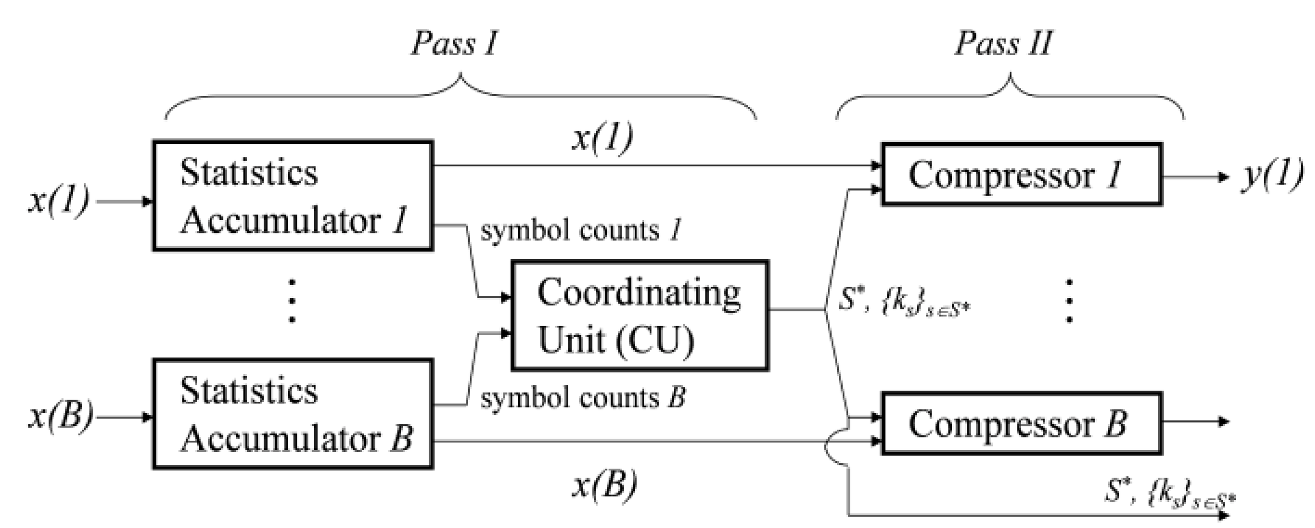
Pass I: Estimate source model and encode:

- Tree structure with |S| states.
- Quantized probability of each leaf.

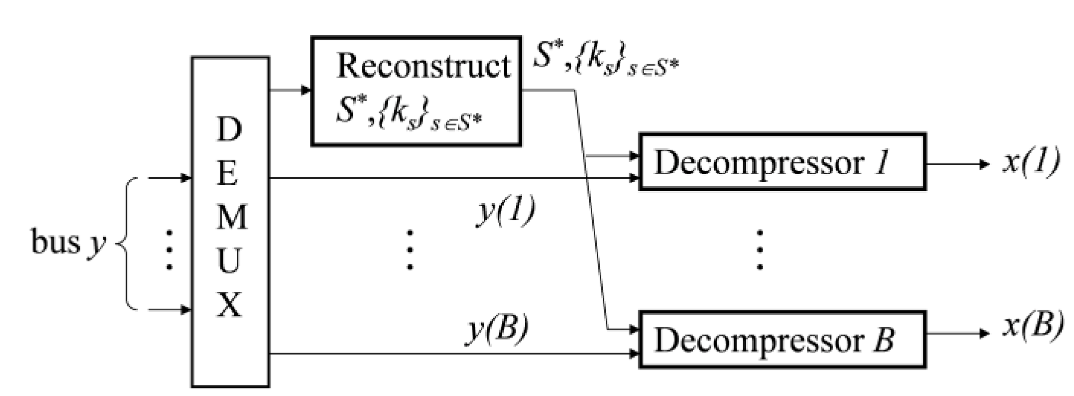
Pass II: Compress using arithmetic coding.

Careful quantizer design yields $\rho <$ 1.047 bits per state above Rissanen's bound [3].

Parallel Two-Pass MDL (PTP-MDL)



PTP-MDL compressor



PTP-MDL decompressor

Properties

Theorem 1: Worst case redundancy of PTP-MDL over maximum likelihood entropy bounded by

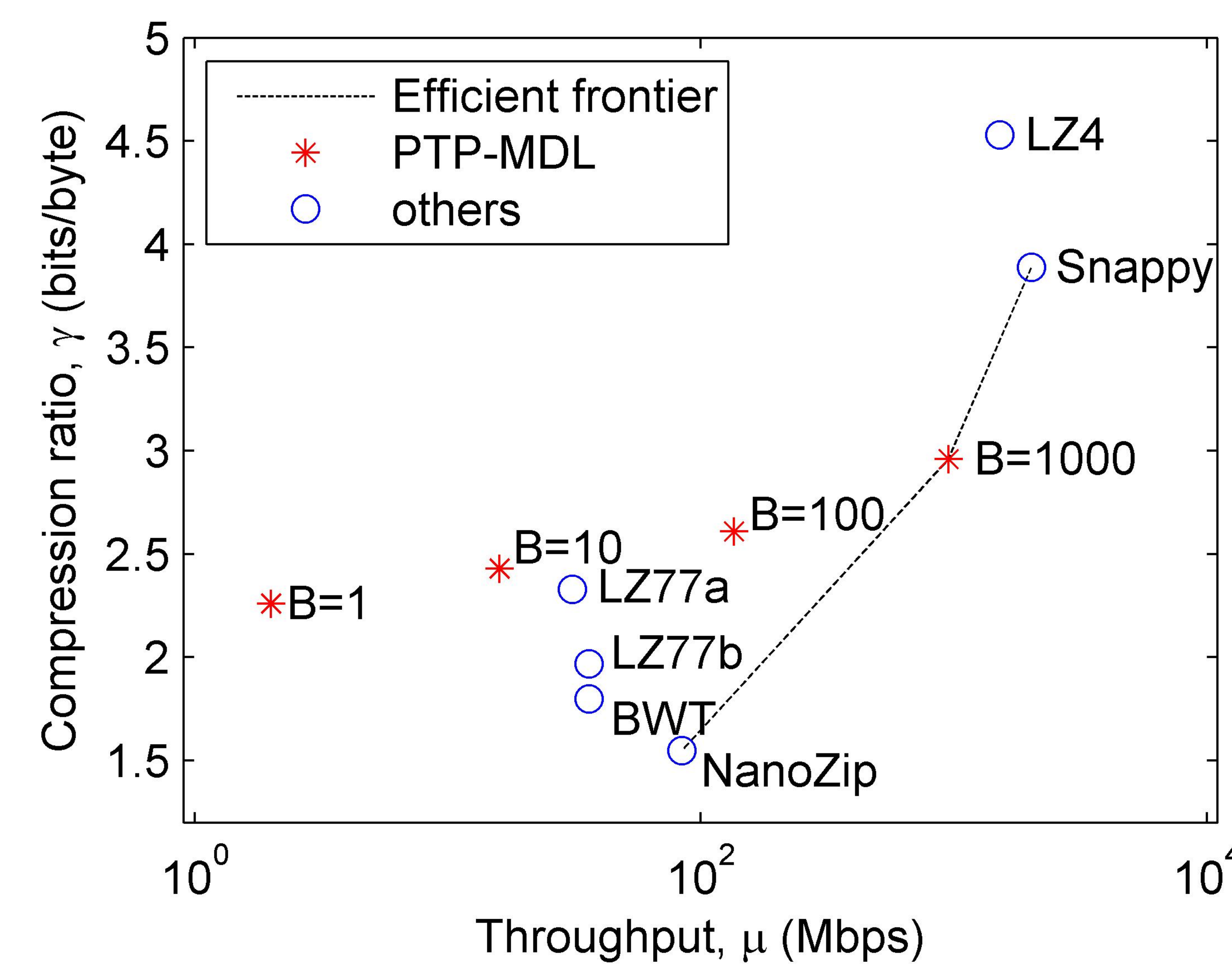
$$\rho(x^N) < B\left[\log\left(\frac{N}{B}\right)+2\right]+\frac{|S|}{2}\left[\log(N)+O(1)\right].$$

Theorem 2: PTP-MDL compressor and decompressor each require *O(N/B)* computation time.

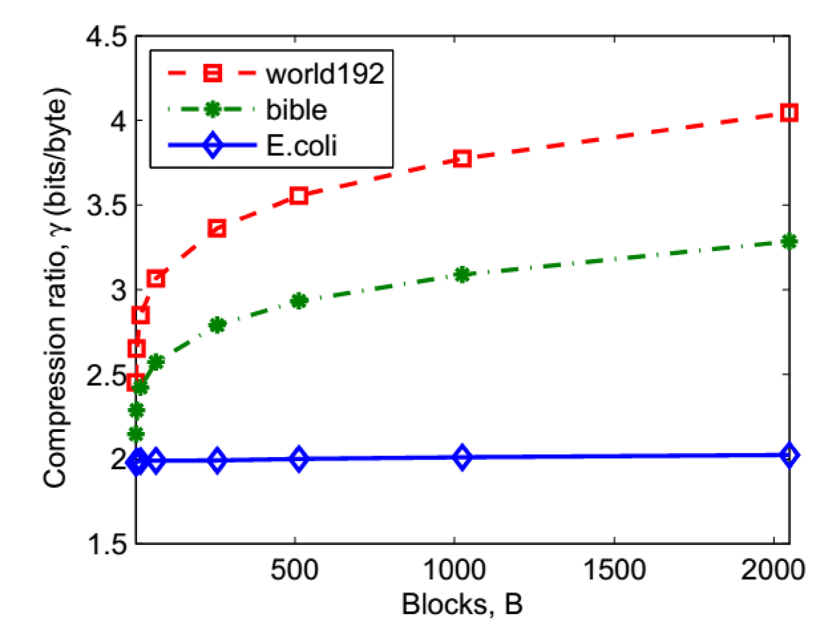
Random access: Any part of the compressed file can be decompressed without decompressing the entire file.

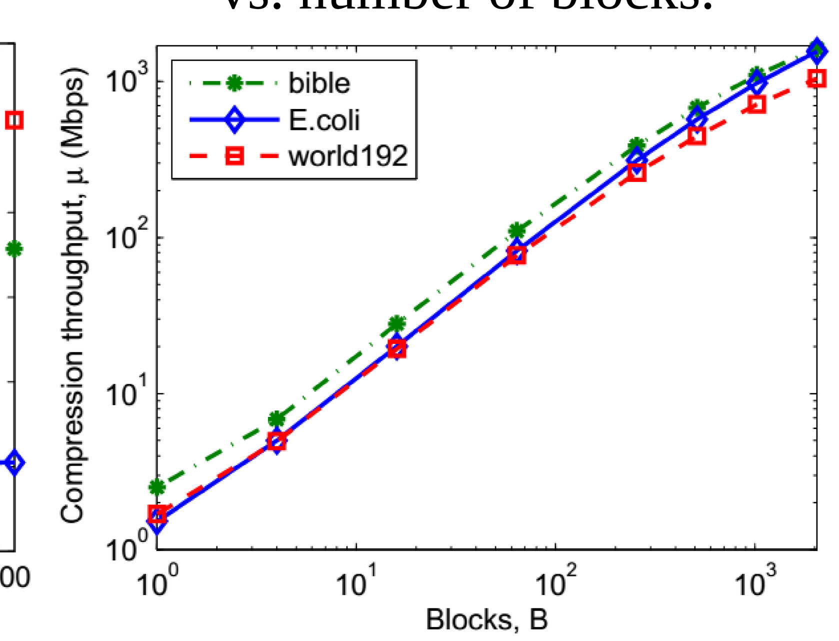
Numerical Results

- Compression ratio vs. throughput for different compressors.



- Compression ratio vs. number of blocks.
- Compression throughput vs. number of blocks.





Proposed algorithm provides good trade-off between compression efficiency and throughput on real data!

[1] M. J. Weinberger, N. Merhav, and M. Feder, "Optimal sequential probability assignment for individual sequences," IEEE Trans. Inf. Theory, vol. 40, no. 2, pp. 384–396, Mar. 1994.
[2] P. Franaszek, J. Robinson, and J. Thomas, "Parallel compression with cooperative dictionary construction," in Proc. Data Compression Conf. (DCC), Mar. 1996.
[3] D. Baron, Y. Bresler., and M. K. Mıhcak, "Two-part codes with low worst-case redundancies for distributed compression of Bernoulli sequences," in Proc. Conf. Inf. Sciences Systems, Mar. 2003.