DCC IEEE

Presented at the 2011 IEEE
Data Compression Conference
Snowbird, Utah, USA — March 29–31

On the Use of Stronger Synchronization to Boost Compression by Substring Enumeration

Danny Dubé
Danny.Dube@ift.ulaval.ca
/
Université Laval
Canada

This work was funded by the Natural Science and Engineering Research Council of Canada.
NSERC CRSNG

Substring enumeration [5]

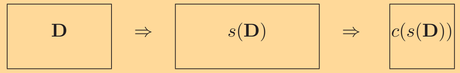
Length	Substrings							
0	8×ϵ							
1	6×0						2×1	
2	4×00				2×01		2×10	
3	3×000			1×001	2×010		1×100	1×101
4	2×0000		1×0001	1×0010	1×0100	1×0101	1×1000	1×1010
5	1×00000	1×00001	1×00010	1×00101	1×01000	1×01010	1×10000	1×10100
6	1×000001	1×000010	1×000101	1×001010	1×010000	1×010100	1×100000	1×101000
7	1×0000010	1×0000101	1×0001010	1×0010100	1×0100000	1×0101000	1×1000001	1×1010000
8	1×00000101	1×00001010	1×00010100	1×00101000	1×01000001	1×01010000	1×10000010	1×10100000

Naïve substring enumeration for '01000001'

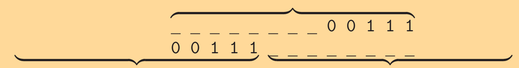
Use of synchronization codes

Instead of:

a pre-processing step inserts a synchronization code:



An unreliable 5-bit scheme



Experimental results (in bpc)

	ISITA'10 [4]									DCC'11					
Bits/Car.	BWT	PPM	Anti	k=0	k=1	k=2	k=3	k=4	n=13	n=12	n=11	n=10	n=9	n=8	n=7
bib	2.07	1.91	2.56	1.98	1.95	1.92	1.92	1.91	1.90	1.89	1.89	1.89	1.89	1.88	1.88
book1	2.49	2.40	3.08	2.27	2.26	2.25	2.25	2.25	2.25	2.25	2.25	2.25	2.25	2.29	2.33
book2	2.13	2.02	2.81	1.98	1.96	1.95	1.95	1.94	1.94	1.93	1.93	1.93	1.93	1.93	1.95
geo	4.45	4.83	6.22	5.35	5.21	4.98	4.81	4.70	4.63	4.58	4.59	4.58	4.58	4.58	4.57
news	2.59	2.42	3.42	2.52	2.49	2.46	2.45	2.44	2.43	2.43	2.43	2.43	2.43	2.42	2.42
obj1	3.98	4.00	4.87	4.46	4.53	4.43	4.32	4.24	4.17	4.03	4.05	4.02	4.01	4.00	3.99
obj2	2.64	2.43	3.61	2.71	2.69	2.59	2.53	2.49	2.47	2.45	2.46	2.45	2.45	2.45	2.44
paper1	2.55	2.37	3.17	2.54	2.51	2.48	2.47	2.46	2.44	2.41	2.41	2.41	2.41	2.41	2.41
paper2	2.51	2.36	3.14	2.41	2.39	2.38	2.38	2.37	2.36	2.35	2.35	2.34	2.35	2.34	2.34
paper3	—	—	—	2.73	2.70	2.69	2.68	2.67	2.65	2.63	2.63	2.63	2.63	2.63	2.63
paper4	—	—	—	3.20	3.16	3.13	3.13	3.10	3.07	3.02	3.02	3.02	3.02	3.01	3.01
paper5	—	—	—	3.33	3.29	3.27	3.24	3.22	3.19	3.12	3.13	3.12	3.12	3.11	3.10
paper6	—	—	—	2.65	2.61	2.58	2.56	2.55	2.52	2.50	2.50	2.49	2.50	2.49	2.49
pic	0.83	0.85	1.09	0.77	0.84	0.82	0.82	0.82	0.81	0.81	0.81	0.81	0.81	0.81	0.81
progc	2.58	2.40	3.18	2.60	2.58	2.54	2.52	2.50	2.48	2.44	2.44	2.44	2.44	2.44	2.43
progl	1.80	1.67	2.24	1.71	1.70	1.69	1.68	1.67	1.66	1.65	1.65	1.65	1.65	1.64	1.64
progp	1.79	1.62	2.27	1.78	1.76	1.73	1.71	1.70	1.68	1.66	1.66	1.66	1.66	1.66	1.65
trans	1.57	1.45	1.94	1.60	1.58	1.53	1.52	1.50	1.48	1.47	1.47	1.47	1.47	1.47	1.46

Occurrences and numbers of occurrences

The data to compress, denoted by $\mathbf{D}$, is drawn from $\{0, 1\}$, has length N, and is considered to be *circular*.

A substring w *occurs at position* p in $\mathbf{D}$, denoted by $w \in_p \mathbf{D}$, if:

$$\exists u, v \in \{0, 1\}^* \,.\, \exists i \in \mathbb{N} \,.\, u\, w\, v = \mathbf{D}^i \text{ and } 0 \leq |u| = p < N.$$

A substring w *occurs* in $\mathbf{D}$, denoted by $w \in \mathbf{D}$, if:

$$\exists p \in \mathbb{N} \,.\, w \in_p \mathbf{D}.$$

The *number of occurrences* of a substring w in $\mathbf{D}$, denoted by C_w, is:

$$|\{p \in \mathbb{N} \mid w \in_p \mathbf{D}\}|.$$

The following equations hold:

$$C_{0\,w} + C_{1\,w} = C_w = C_{w\,0} + C_{w\,1}, \qquad \text{for any } w \in \{0, 1\}^*, \quad \text{and}$$
$$\sum_{w \in \{0,\,1\}^n} C_w = N, \qquad \text{for any } n \in \mathbb{N}.$$

Synchronization schemes

Per-byte mappings only; padding bytes with 9 bit strings:

$$M(b_1\, b_2 \ldots b_8) = w_1\, b_1\, w_2\, b_2 \ldots w_8\, b_8\, w_9.$$

A *k-bit scheme* inserts k bits per byte, where $k = |w_1 \ldots w_9|$.

A k-bit scheme is *n-reliable* if it is possible to determine the phase (a number among $0, \ldots, k+7$) of any substring of the synchronized data that is at least n bits long.

	n	k	Synchronization scheme
ISITA'10 [4]	—	0	_ _ _ _ _ _ _ _
	—	1	_ _ _ _ _ _ _ _ 0
	—	2	_ _ _ _ _ _ _ _ 0 1
	—	3	_ _ _ _ _ _ _ _ 0 1 1
	—	4	_ _ _ _ _ _ _ _ 0 1 1 1
	13	5	_ _ _ _ _ _ 0 _ _ 0 1 1 1
DCC'11	12	8	_ _ _ 0 _ _ 1 0 0 _ _ 1 _ 1 1 0
	11	8	_ _ _ 0 _ 0 _ _ 1 1 0 _ _ 1 1 0
	10	10	_ _ _ _ 0 _ 0 1 1 _ _ _ 0 1 0 0 1 1
	9	10	_ _ _ _ 0 0 0 1 1 _ _ _ _ 0 1 0 1 1
	8	15	_ _ _ 0 0 0 1 0 _ _ _ 1 1 1 0 1 _ _ 1 1 0 0 1
	7	20	1 1 0 1 _ 1 _ 1 1 0 0 _ 0 _ 0 1 0 0 _ 0 _ 1 1 0 0 _ 1 _

Pseudo-code for CSE

Send N; **Send** C_0
For l **:=** 2 **to** N **do**
  For every $w \in \mathbf{D}$ **such that** $|w| = l - 2$ **and** $\min(C_{0\,w}, C_{1\,w}, C_{w\,0}, C_{w\,1}) \geq 1$ **do**
    Predict and send $C_{0\,w\,0}$

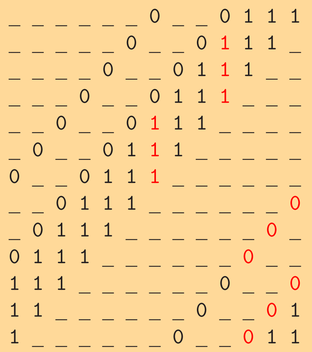
Compressor performing Compression by Substring Enumeration (CSE).
Linear time and space complexities [6].

Phase unawareness

...!W*... ⟼ ...00100001010101110010101 0...

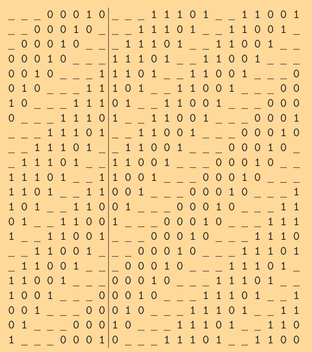
Benchmark files are made of bytes; each byte is mapped to 8 bits.
CSE is unaware of the original bytes and the phase of the bits.
Bits on different phases are likely to have different statistics.
CSE's predictions on mixed-phase substrings are likely to be suboptimal.

A 13-reliable 5-bit scheme



Demonstration of the 13-reliability of the scheme.

An 8-reliable 15-bit scheme.



Demonstration of the 8-reliability of the scheme.

Conclusions

- Using synchronization schemes stronger than those used in [4] does not achieve significant improvements.
- Future work: to address the phase unawareness problem directly by modifying CSE by annotating bits with their respective phase.
- Future work: to develop a variant for images and lossy variants for different types of data.
- Related work: prediction by partial matching [2], the Burrows-Wheeler transform [1], anti-dictionaries [3], LZ77 [7], and LZ78 [8].

References

[1] M. Burrows and D. Wheeler. A block sorting lossless data compression algorithm. Technical Report 124, Digital Equipment Corporation, 1994.

[2] J. G. Cleary and W. J. Teahan. Unbounded length contexts for PPM. *The Computer Journal*, 40(2/3):67–75, 1997.

[3] M. Crochemore and G. Navarro. Improved antidictionary based compression. In *Proceedings of the International Conference of the Chilean Computer Science Society*, pages 7–13, 2002.

[4] Danny Dubé. Using synchronization bits to boost compression by substring enumeration. In *Proceedings of ISITA*, Taichung, Taiwan, October 2010.

[5] Danny Dubé and Vincent Beaudoin. Lossless data compression via substring enumeration. In *Proceedings of the Data Compression Conference*, pages 229–238, Snowbird, Utah, USA, March 2010.

[6] Danny Dubé and Hidetoshi Yokoo. The universality and linearity of compression by substring enumeration. In *a submission to the International Symposium on Information Theory*, Saint Petersburg, Russia, August 2011.

[7] J. Ziv and A. Lempel. A universal algorithm for sequential data compression. *IEEE Transactions on Information Theory*, 23(3):337–342, 1977.

[8] J. Ziv and A. Lempel. Compression of individual sequences via variable-rate coding. *IEEE Transactions on Information Theory*, 24(5):530–536, September 1978.