Errata Listing for *Compression and Coding Algorithms*, Kluwer Academic Publishers, 2002

Alistair Moffat and Andrew Turpin

August 16, 2016

Preface

Chapter 1

Page 9

The claim on disk transfer rates is somewhat dated, and modern disks operate at rates in excess of 50 MB/second. Seek times are still in the 8–10 millisecond range.

[*Shingo Mamada, September 2002*]

Chapter 2

Chapter 3

Page 32

For example, when $b = 5$ the codeword lengths

should be replaced by

For example, when $n = 5$ the codeword lengths

[*Ke Shi, August 2016*]

Page 41

For example, a Rice code with $k = 1$ assigns the codewords “00”, “01”, “100”, “101”, “1100”, and “1101”, to symbols 0, $+1$, -1, $+2$, and -2, respectively, and is biased in favor of the positive values.

should be replaced by

For example, a Rice code with $k = 1$ assigns the codewords “00”, “01”, “100”, “101”, “1100”, and “1101”, to symbols 0, $+1$, -1, $+2$, -2, and $+3$, respectively, and is biased in favor of the positive values.

[*Eduardo Morras, December 2001*]

This error reflects exactly the point being made in that paragraph – that an asymmetric set of codeword lengths is inappropriate.

Page 43

In line 1 of *centered_minimal_binary_encode* it should say "set *long* $\leftarrow 2 \times n - 2^{\lceil \log_2 n \rceil}$", that is, no reference to x while the number of *long* codewords is being calculated.

[*André Osterhues, May 2004*]

Chapter 4

Page 64

At the bottom of the page we mention a decoding mechanism by Hashemian [1995], that uses k-bit tables to provide either the decoding and number of bits for a code if that length is less than or equal to k, or a pointer to another table for codes longer than k bits. This decoding method was previously used in the inflate routines in Info-ZIP's unzip and gzip, both developed in 1992 by Mark Adler.

[*Mark Adler, October 2004*]

Page 70–77

The mechanism described in this section is the work of Moffat and Turpin (1998). Somehow or other we missed citing our own paper!

[*Alistair Moffat, December 2001*]

Page 86

The first assignment (to *sym*) in step 10 of Algorithm 4.7 is not required. Then, in the second assignment in step 11, the right-hand-side expression should be $w[\textit{code\_len}] - 1$, not the "$+1$" shown in the book.

[*Andrew Turpin, October 2003*]

Chapter 5

Page 108

In Table 5.3, when coding the sixth symbol (which is a 4), the value of r that is given is incorrect, it should be 6 rather than the 4 that is shown (calculated as $60/10$).

[*Wendy Cameron, April 2004*]

Chapter 6

Chapter 7

Chapter 8

Page 220

Another theme that has been explored by a number researchers

should be replaced by

Another theme that has been explored by a number of researchers

Page 241

The inversion coding technique of Arnavut (2000) should also have been mentioned as an alternative to the MTF transformation. It is an alternative way of dealing with the localized nature of the post-BWT string. Interpolative coding (dscribed in Chapter 3) can also be used in a similar way, as described by Moffat and Stuiver (2000).

[*Alistair Moffat, December 2001*]

Page 248

Another recent paper examining searching over compressed text is by Navarro et al. (2001).

[*Alistair Moffat, December 2001*]

Chapter 9

Page 232

A recent PPM implementation of Dmitry Shkarin (2002) obtains extremely good compression, with one version reported to attain a "Calgary average" of 1.92 bits per character. A link to an executable is provided at `http://datacompression.info/PPM.shtml`.

[*Sebastian Deorowicz, August 2002*]

Page 241

The blocksize for SZIP is 4.1 MB, not 4.3 MB as listed.

[*Sebastian Deorowicz, August 2002*]

Page 241

Another recency transformation that can be used with the BWT is "Distance Coding". Deorowicz (2002) includes a description of distance coding in his recent paper. Elias (1987) also considers various recency transformations.

[*Sebastian Deorowicz, August 2002*]

Bibliography

Page 261

The third paper listed is by "Guazzo", not "Guauzzo".

[*Sebastian Deorowicz, August 2002*]

Page 263

The paper Liddell and Moffat (2002) will appear in the proceedings of the 2002 *IEEE Data Compression Conference*.

[*Alistair Moffat, December 2001*]

Page 265

The seventh paper listed is by "Pasco", not "Pascoe".

[*Sebastian Deorowicz, August 2002*]

Index

References

Z. Arnavut. Move-to-front and inversion coding. In J. A. Storer and M. Cohn, editors, *Proc. 2000 IEEE Data Compression Conference*, pages 193–202. IEEE Computer Society Press, Los Alamitos, California, March 2000.

S. Deorowicz. Second step algorithms in the Burrows-Wheeler compression algorithm. *Software – Practice and Experience*, 32(9):99–111, 2002.

P. Elias. Interval and recency-rank source coding: Two on-line adaptive variable-length schemes. *IEEE Transactions on Information Theory*, IT-33(1):3–10, January 1987.

M. Liddell and A. Moffat. Incremental calculation of optimal length-restricted codes. In J. Storer and M. Cohn, editors, *Proc. 2002 IEEE Data Compression Conference*, April 2002.

A. Moffat and L. Stuiver. Binary interpolative coding for effective index compression. *Information Retrieval*, 3(1):25–47, July 2000.

A. Moffat and A. Turpin. Efficient construction of minimum-redundancy codes for large alphabets. *IEEE Transactions on Information Theory*, 44(4):1650–1657, July 1998.

G. Navarro, T. Kida, M. Takeda, A. Shinohara, and S. Arikawa. Faster approximate string matching over compressed text. In J. A. Storer and M. Cohn, editors, *Proc. 2001 IEEE Data Compression Conference*, pages 459–468. IEEE Computer Society Press, Los Alamitos, California, March 2001.

D. Shkarin. PPM: One step to practicality. In J. A. Storer and M. Cohn, editors, *Proc. 2002 IEEE Data Compression Conference*, pages 202–211. IEEE Computer Society Press, Los Alamitos, California, April 2002.