

---

## Contents

<b>1</b>	<b>Introduction</b> .....	1
1.1	An example of a Burrows-Wheeler Transform .....	3
1.2	Genesis of the Burrows-Wheeler Transform .....	5
1.3	Transformation .....	8
1.4	Permutation .....	11
1.5	Recency .....	12
1.6	Pattern matching .....	13
1.7	Organization of this book .....	14
1.8	Further reading .....	16
<b>2</b>	<b>How the Burrows-Wheeler Transform works</b> .....	19
2.1	The forward Burrows-Wheeler Transform .....	19
2.2	The reverse Burrows-Wheeler Transform .....	23
2.3	Special cases .....	29
2.4	Further reading .....	31
<b>3</b>	<b>Coders for the Burrows-Wheeler Transform</b> .....	33
3.1	Entropy coding .....	33
3.2	Run-length and arithmetic coder .....	38
3.3	Move-to-front lists .....	39
3.4	Frequency counting methods .....	42
3.5	Inversion Frequencies (IF) .....	43
3.6	Distance coding .....	44
3.7	Wavelet trees .....	45
3.8	Other permutations .....	46
3.9	Block size .....	47
3.10	Further reading .....	48
<b>4</b>	<b>Suffix trees and suffix arrays</b> .....	51
4.1	Suffix Trees .....	51
4.1.1	Basic notations and definitions .....	52

4.1.2	Construction of a suffix tree	54
4.1.3	Ukkonen's suffix tree algorithm	57
4.1.4	From implicit suffix tree to true suffix tree	64
4.1.5	Farach's recursive construction	66
4.1.6	Generalized suffix trees	73
4.1.7	Implementation issues	74
4.2	Suffix arrays	75
4.2.1	Traditional string sorting	76
4.2.2	Suffix arrays via suffix trees	78
4.2.3	Manber-Myers suffix sorting algorithm	78
4.2.4	Linear-time direct suffix sorting	81
4.3	Space issues in suffix trees and suffix arrays	85
4.4	Further reading	88
<b>5</b>	<b>Analysis of the Burrows-Wheeler Transform</b>	<b>91</b>
5.1	The BWT, suffix trees and suffix arrays	93
5.2	Computational complexity	95
5.2.1	BWT first stage — the transform	95
5.2.2	BWT second stage — coding the transformed text	95
5.3	BWT context clustering property	97
5.3.1	Context trees	97
5.3.2	Estimation using context trees	100
5.3.3	BWT and context trees	103
5.4	Analysis of BWT output	104
5.4.1	Theoretical distribution of BWT output	104
5.4.2	Empirical distribution of BWT output	105
5.5	Analysis of BWT compression performance	119
5.5.1	Definitions and notation	120
5.5.2	Performance using recency ranking	123
5.5.3	Performance without LGT	129
5.5.4	Performance using piecewise constant parameters	132
5.5.5	Performance on general sources via empirical entropy	133
5.6	Relationship with other compression schemes	135
5.6.1	Context-based schemes	135
5.6.2	Symbol ranking schemes	148
5.7	Further reading	149
<b>6</b>	<b>Variants of the Burrows-Wheeler Transform</b>	<b>153</b>
6.1	The sort transform	154
6.1.1	Forward sort transform	154
6.1.2	Inverse sort transform	155
6.1.3	Performance of the sort transform	159
6.2	Lexical permutation sorting	163
6.2.1	Sorting permutations	164
6.2.2	Lexical permutation sorting algorithm	167

6.3	The extended BWT .....	168
6.3.1	Sort order between strings .....	168
6.3.2	Performing the extended BWT .....	169
6.3.3	Inverting the transform .....	170
6.4	Sort-based context similarity measurement .....	173
6.4.1	Context similarity measurement and ranking .....	173
6.4.2	The prefix list data structure .....	175
6.4.3	Relationship with the Burrows-Wheeler Transform .....	178
6.4.4	Performance of the prefix list .....	180
6.5	Word-based compression .....	180
6.5.1	General word-based compression .....	181
6.5.2	Word-based Burrows-Wheeler Transform .....	183
6.6	Further reading .....	185
<b>7</b>	<b>Exact and approximate pattern matching .....</b>	<b>187</b>
7.1	Exact pattern matching algorithms .....	188
7.1.1	Brute force matching .....	189
7.1.2	The Knuth-Morris-Pratt Algorithm .....	190
7.1.3	The Boyer-Moore algorithm .....	195
7.1.4	The Karp-Rabin algorithm .....	197
7.1.5	The shift-and method .....	199
7.1.6	Multiple pattern matching .....	200
7.1.7	Pattern matching with don't-care characters .....	204
7.2	Pattern matching using the Burrows-Wheeler Transform .....	207
7.2.1	Boyer-Moore pattern matching using the BWT .....	209
7.2.2	BWT-based exact pattern matching with binary search .....	209
7.2.3	BWT-based exact pattern matching with suffix arrays .....	214
7.2.4	Pattern matching using the FM-index .....	215
7.2.5	Algorithm improvements with overwritten arrays .....	220
7.3	Performance of BWT-based exact pattern matching .....	221
7.3.1	Compression performance .....	222
7.3.2	Search performance .....	224
7.3.3	Array construction speeds .....	231
7.3.4	Comparison with LZ-based compressed-domain pattern matching .....	232
7.4	Approximate pattern matching .....	233
7.4.1	Edit distance: dynamic programming formulation .....	234
7.4.2	Edit graphs .....	236
7.4.3	Local similarity .....	237
7.4.4	The longest common subsequence problem .....	239
7.4.5	String matching with $k$ differences .....	244
7.4.6	The $k$ -mismatch problem using the BWT .....	247
7.4.7	$k$ -approximate matching using the BWT .....	253
7.5	Hardware algorithms for pattern matching .....	255
7.5.1	An equivalent hardware algorithm .....	256

7.5.2	A brief review of other hardware algorithms . . . . .	258
7.6	Conclusion . . . . .	259
7.7	Further reading . . . . .	260
<b>8</b>	<b>Other applications of the Burrows-Wheeler Transform . . . . .</b>	<b>265</b>
8.1	Compressed suffix trees and compressed suffix arrays . . . . .	266
8.1.1	Compressed suffix trees . . . . .	267
8.1.2	Compressed suffix arrays . . . . .	270
8.2	Compressed full-text indexing . . . . .	275
8.2.1	Full-text indexing using CSTs and CSAs . . . . .	276
8.2.2	Searching on compressed suffix trees . . . . .	277
8.2.3	Searching on compressed suffix arrays. . . . .	278
8.3	Bioinformatics and computational biology . . . . .	278
8.3.1	DNA sequence compression . . . . .	279
8.3.2	Analysis of repetition structures . . . . .	280
8.3.3	Whole-genome comparisons . . . . .	281
8.3.4	Genome annotation . . . . .	282
8.3.5	Distance measure between sequences and phylogeny . . . . .	283
8.4	Test data compression . . . . .	284
8.4.1	Nature of test data . . . . .	285
8.4.2	BWT-based test data compression . . . . .	286
8.5	Image compression, computer vision and machine translation . . . . .	287
8.5.1	Image compression . . . . .	287
8.5.2	Shape matching . . . . .	292
8.5.3	Machine translation . . . . .	294
8.6	Joint source-channel coding . . . . .	296
8.6.1	General source coding via channel coding . . . . .	297
8.6.2	BWT-based joint source-channel coding . . . . .	298
8.7	Prediction and entropy estimation . . . . .	299
8.8	Further reading . . . . .	301
<b>9</b>	<b>Conclusion . . . . .</b>	<b>305</b>
<b>A</b>	<b>Notation . . . . .</b>	<b>309</b>
<b>B</b>	<b>Ongoing work on the Burrows-Wheeler Transform . . . . .</b>	<b>313</b>
B.1	BWT-related web sites . . . . .	313
B.2	Ph.D. theses relating to the Burrows-Wheeler Transform . . . . .	314
	<b>References . . . . .</b>	<b>317</b>
	<b>Index . . . . .</b>	<b>341</b>