Table 2.7 The Canterbury corpus of text used to evaluate compression methods.

File	Bytes	Content
text	152,089	The text of Lewis Carroll's *Alice's Adventures in Wonderland*
fax	513,216	A fax bitmap image (CCITT test document 5)
Csrc	11,150	C source code
Excl	1,029,744	Excel spreadsheet
SPRC	38,240	Executable object code for Sun SPARC architecture
tech	426,754	Technical writing (workshop proceedings)
poem	481,861	*Paradise Lost* by John Milton
HTML	24,603	Hypertext Markup Language source
lisp	3,721	A Lisp program
man	4,227	A Unix manual page in the *roff* format
play	125,179	Shakespeare's play, *As You Like It*

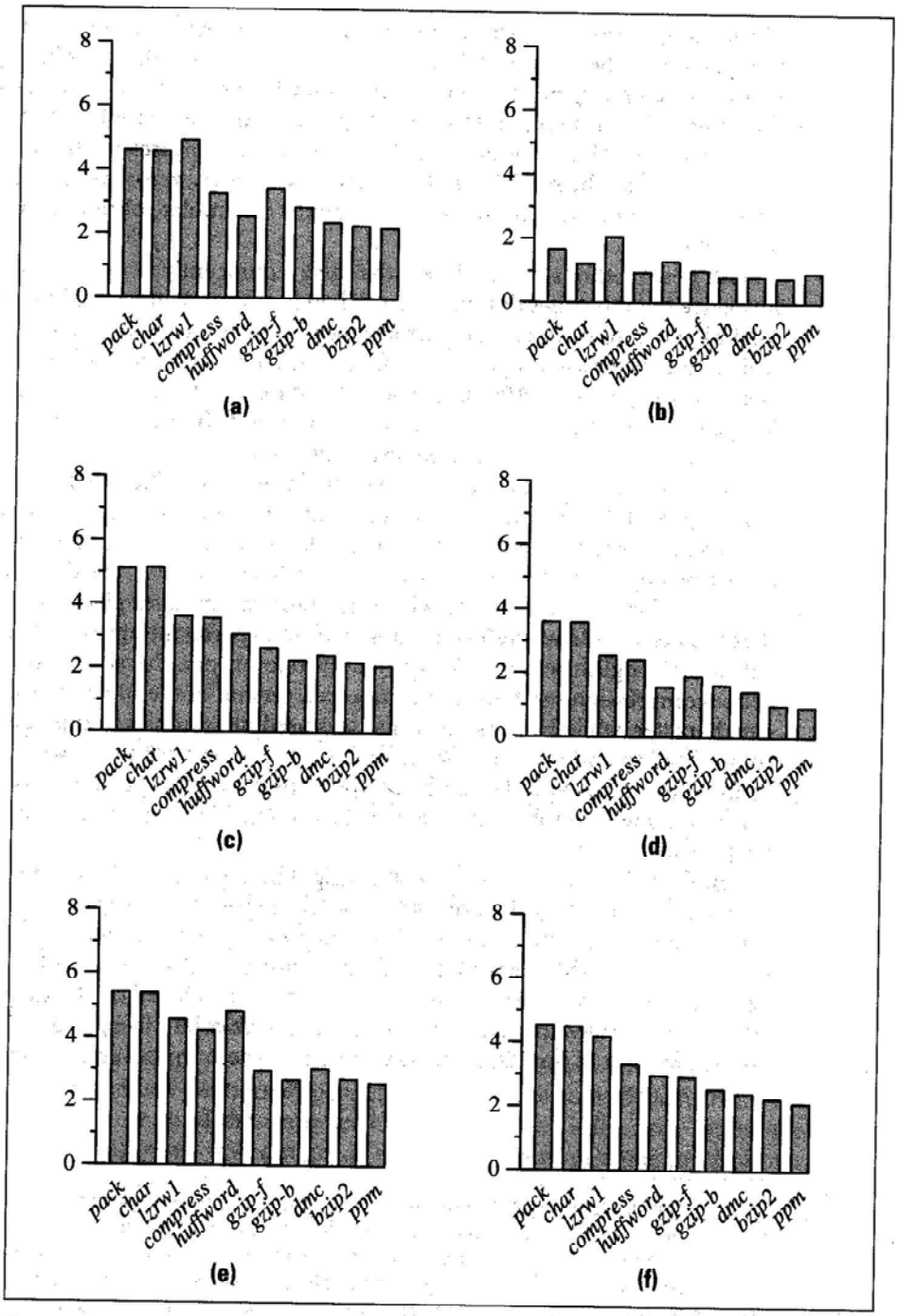

Figure 2.42 Compression performance for the Canterbury corpus, measured in bits per character: (a) English text; (b) fax image; (c) C program source code; (d) spreadsheet; (e) SPARC object code; (f) overall average.

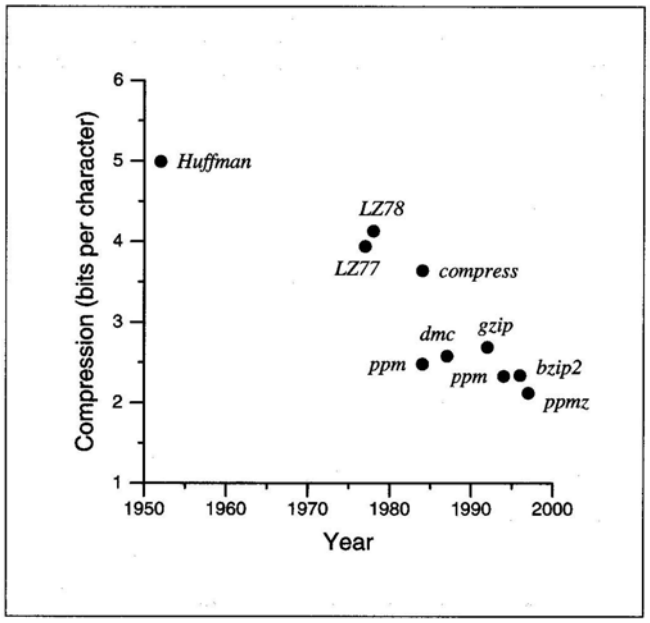

Figure 2.43 Compression effectiveness versus year of discovery of methods.

Table 2.8 Throughput (relative to the encoding speed of *compress*) for encoding and decoding, and average compression on the Canterbury corpus (bits per character and percent remaining).

Method	Relative speed		Compression	
	Encoding	Decoding	bpc	%
dmc	24.3	24.5	2.40	30.0
ppm	5.3	5.9	2.11	26.4
char	2.9	4.0	4.49	56.1
bzip2	5.5	2.0	2.23	27.9
pack	0.6	0.9	4.53	56.6
huffword	2.2	0.9	2.95	36.9
compress	1.0	0.6	3.31	41.4
lzrw1	0.7	0.4	4.18	52.3
gzip-f	1.1	0.4	2.91	36.4
gzip-b	7.0	0.3	2.53	31.6
null	0.2	0.2	8.00	100.0

File Names	Actual Sizes
Calgary	
Bib	111261
book1	768771
book2	610856
News	377109
paper1	53161
paper2	82199
paper3	46526
paper4	13286
paper5	11954
paper6	38105
Progc	39611
Progl	71646
Progp	49379
Trans	93695

File Names	Actual Sizes
Canterbury	
alice29.txt	152089
asyoulik.txt	125179
cp.html	24603
fields.c	11150
grammar.lsp	3721
lcet10.txt	426754
plrabn12.txt	481861
xargs.1	4227
bible.txt	4047392
kjv.Gutenberg	4846137
world192.txt	2473400
Project Gutenberg	
1musk10.txt	1344739
anne11.txt	586960
world95.txt	2988578

Table 1: Text files used in our tests (a) Table showing text files and their sizes from Calgary Corpus (b) Table showing text files and their sizes from Canterbury Corpus and Project Gutenberg

FileNames	Bzip2 (BPC)	Bzip2 with LIPT (BPC)
Calgary		
paper5	3.24	**2.95**
paper4	3.12	**2.74**
paper6	2.58	**2.40**
Progc	2.53	**2.44**
paper3	2.72	**2.45**
Progp	1.74	**1.72**
paper1	2.49	**2.33**
Progl	1.74	**1.66**
paper2	2.44	**2.26**
Trans	1.53	**1.47**
Bib	1.97	**1.93**
News	2.52	**2.45**
Book2	2.06	**1.99**
Book1	2.42	**2.31**
Average BPC	2.36	**2.22**

FileNames	Bzip2 (BPC)	Bzip2 with LIPT (BPC)
Canterbury		
grammar.lsp	2.76	**2.58**
xargs.1	3.33	**3.10**
fields.c	2.18	**2.14**
cp.html	2.48	**2.44**
asyoulik.txt	2.53	**2.42**
alice29.txt	2.27	**2.13**
lcet10.txt	2.02	**1.91**
plrabn12.txt	2.42	**2.33**
world192.txt	1.58	**1.52**
bible.txt	1.67	**1.62**
kjv.gutenberg	1.66	**1.62**
Average BPC	2.26	**2.17**

FileNames	Bzip2 (BPC)	Bzip2 with LIPT (BPC)
Gutenberg		
Anne11.txt	2.22	**2.12**
1musk10.txt	2.08	**1.98**
World95.txt	1.54	**1.49**
Average BPC	1.95	**1.86**

Table 2: Tables a – c show BPC comparison between original Bzip2 –9, and Bzip2 –9 with LIPT for the files in three corpuses

(a) (b) (c)

(a) **(b)** **(c)**

FileNames	PPMD (BPC)	PPMD with LIPT
Calgary		
paper5	2.98	**2.74**
paper4	2.89	**2.57**
paper6	2.41	**2.29**
Progc	2.36	**2.30**
paper3	2.58	**2.37**
Progp	1.70	**1.68**
paper1	2.33	**2.21**
Progl	1.68	**1.61**
paper2	2.32	**2.17**
Trans	1.47	**1.41**
Bib	1.86	**1.83**
news	2.35	**2.31**
book2	1.96	**1.91**
book1	2.30	**2.23**
Average BPC	2.23	**2.12**

FileNames	PPMD (BPC)	PPMD with LIPT (BPC)
Canterbury		
grammar.lsp	2.36	**2.21**
xargs.1	2.94	**2.73**
fields.c	2.04	**1.97**
cp.html	2.26	**2.22**
asyoulik.txt	2.47	**2.35**
alice29.txt	2.18	**2.06**
lcet10.txt	1.93	**1.86**
plrabn12.txt	2.32	**2.27**
world192.txt	1.49	**1.45**
bible.txt	1.60	**1.57**
kjv.gutenberg	1.57	**1.55**
Average BPC	2.11	**2.02**

FileNames	PPMD (BPC)	PPMD with LIPT
Gutenberg		
Anne11.txt	2.13	**2.04**
1musk10.txt	1.91	**1.85**
World95.txt	1.48	**1.45**
Average BPC	1.84	**1.78**

Table 3: Tables a – c show BPC comparison between original PPMD (order 5), and PPMD (order 5) with LIPT for the files in three corpuses.

(a)

FileNames	Gzip (BPC)	Gzip with LIPT (BPC)
Calgary		
paper5	3.34	**3.05**
paper4	3.33	**2.95**
paper6	2.77	**2.61**
Progc	2.68	**2.61**
paper3	3.11	**2.76**
Progp	1.81	**1.81**
paper1	2.79	**2.57**
Progl	1.80	**1.74**
paper2	2.89	**2.62**
Trans	1.61	**1.56**
bib	2.51	**2.41**
news	3.06	**2.93**
book2	2.70	**2.48**
book1	3.25	**2.96**
Average BPC	2.69	**2.51**

(b)

FileNames	Gzip (BPC)	Gzip with LIPT (BPC)
Canterbury		
grammar.lsp	2.68	**2.61**
xargs.1	3.32	**3.13**
fields.c	2.25	**2.21**
Cp.html	2.60	**2.55**
asyoulik.txt	3.12	**2.89**
alice29.txt	2.85	**2.60**
lcet10.txt	2.71	**2.42**
plrabn12.txt	3.23	**2.96**
world192.txt	2.33	**2.18**
bible.txt	2.33	**2.18**
kjv.gutenberg	2.34	**2.19**
Average BPC	2.70	**2.54**

(c)

FileNames	Gzip (BPC)	Gzip with LIPT (BPC)
Gutenberg		
Anne11.txt	3.02	**2.75**
1musk10.txt	2.91	**2.62**
World95.txt	2.31	**2.15**
Average BPC	2.75	**2.51**

Table 4: Tables a – c show BPC comparison between original Gzip -9, and Gzip –9 with LIPT for the files in three corpuses.

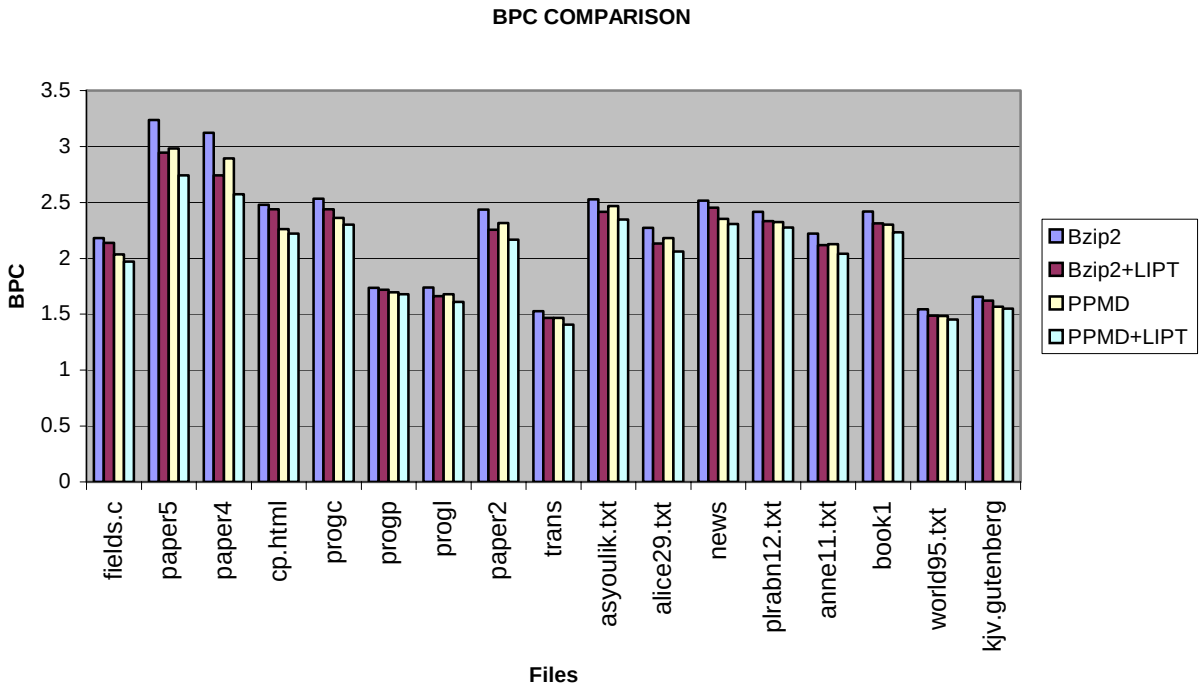

Figure 3: Bar Chart giving comparison of Bzip2, Bzip2 with LIPT, PPMD, and PPPMD with LIPT

	Original (BPC)	*-encoded (BPC)	**LIPT (BPC)**
Huffman (character based)	4.87	**4.12**	4.49
Arithmetic (word based)	2.71	2.90	**2.61**
Gzip-9	2.70	**2.52**	**2.52**
Bzip2	2.28	2.24	**2.16**
PPMD	2.13	2.13	**2.04**

Table 5: Summary of BPC Results

File	Ref.No.1	Ref. No. 2	Ref. No. 4	Bzip2 with LIPT
bib	2.05	1.94	1.94	**1.93**
book1	**2.29**	2.33	**2.29**	2.31
book2	2.02	2.00	2.00	**1.99**
news	2.55	2.47	2.48	**2.45**
paper1	2.59	2.44	2.45	**2.33**
paper2	2.49	2.39	2.39	**2.26**
progc	2.68	2.47	2.51	**2.44**
progl	1.86	1.70	1.71	**1.66**
progp	1.85	**1.69**	1.71	1.72
trans	1.63	**1.47**	1.48	**1.47**
Average BPC	2.21	2.105	2.11	**2.07**

Table 6: BPC comparison of approaches based on BWT

File	Ref. No.11	Ref. No. 7	PPMD (order 5) with LIPT
bib	1.86	1.84	**1.83**
book1	**2.22**	2.39	2.23
book2	1.92	1.97	**1.91**
news	2.36	2.37	**2.31**
paper1	2.33	2.32	**2.21**
paper2	2.27	2.33	**2.17**
progc	2.38	2.34	**2.30**
progl	1.66	**1.59**	1.61
progp	1.64	**1.56**	1.68
trans	1.43	**1.38**	1.41
Average BPC	2.021	2.026	**1.98**

Table 7: BPC comparison of new approaches based on Prediction Models

	Filesize	Bzip2 with LIPT	Word-based Huffman	% GAIN
cp.html	21333	2.03	2.497	18.696
paper6	32528	2.506	2.740	8.550
progc	33736	2.508	2.567	2.289
paper1	42199	2.554	2.750	7.134
progp	44862	1.733	2.265	23.497
progl	62367	1.729	2.264	23.618
trans	90985	1.368	1.802	24.090
bib	105945	1.642	2.264	27.477
asyoulik.txt	108140	2.525	2.564	1.527
alice29.txt	132534	2.305	2.333	1.194
lcet10.txt	307026	2.466	2.499	1.314
news	324476	2.582	2.966	12.936
book2	479612	2.415	2.840	14.963
anne11.txt	503408	2.377	2.466	3.597
1musk10.txt	1101083	2.338	2.454	4.726
world192.txt	1884748	1.78	2.600	31.551
world95.txt	2054715	1.83	2.805	34.750
kjv.gutenberg	4116876	1.845	2.275	18.914
AVERAGE		**2.169**	**2.506**	**13.439**

Table 8: BPC comparison for test files using Bzip2 with LIPT, and word-based Huffman

File	Bzip2 (BPC)	Bzip2 with LIPT (BPC)	YBS (BPC)	YBS with LIPT (BPC)
bib.ybs	1.97	1.93	1.93	**1.91**
book1.ybs	2.42	2.31	2.22	**2.16**
book2.ybs	2.06	1.99	1.93	**1.90**
news.ybs	2.52	2.45	2.39	**2.36**
paper1.ybs	2.49	2.33	2.40	**2.28**
paper2.ybs	2.44	2.26	2.34	**2.18**
paper3.ybs	2.72	2.45	2.63	**2.39**
paper4.ybs	3.12	2.74	3.06	**2.71**
paper5.ybs	3.24	2.95	3.17	**2.91**
paper6.ybs	2.58	2.40	2.52	**2.36**
progc.ybs	2.53	2.44	2.48	**2.43**
progl.ybs	1.74	1.66	1.69	**1.63**
progp.ybs	1.74	1.72	1.71	**1.70**
trans.ybs	1.53	1.47	1.48	**1.44**
Average	2.36	2.22	2.28	**2.17**

Table 9: BPC Comparison of Bzip2, Bzip2 with LIPT, YBS, and YBS with LIPT for the text files in Calgary Corpus only

File	Bzip2 (BPC)	Bzip2 with LIPT (BPC)	PPMD (order 5) (BPC)	PPMD with LIPT (BPC)	RK (BPC)	RK with LIPT (BPC)
bib	1.97	1.93	1.86	1.83	1.75	**1.72**
book1	2.42	2.31	2.30	2.23	**2.16**	**2.16**
book2	2.06	1.99	1.96	1.91	**1.83**	1.85
news	2.52	2.45	2.35	2.31	2.23	**2.21**
paper1	2.49	2.33	2.33	2.21	2.27	**2.17**
paper2	2.44	2.26	2.32	2.17	2.21	**2.11**
paper3	2.72	2.45	2.58	2.37	2.52	**2.33**
paper4	3.12	2.74	2.89	2.57	2.90	**2.63**
paper5	3.24	2.95	2.98	2.74	3.01	**2.81**
paper6	2.58	2.40	2.41	2.29	2.35	**2.27**
progc	2.53	2.44	2.36	2.30	2.31	**2.27**
progl	1.74	1.66	1.68	1.61	1.48	**1.44**
progp	1.74	1.72	1.70	1.68	1.50	**1.49**
trans	1.53	1.47	1.47	1.41	1.24	**1.23**
Average	2.36	2.22	2.23	2.12	2.12	**2.05**

Table 10: BPC Comparison of Bzip2, Bzip2 with LIPT, RK, and RK with LIPT for the text files in Calgary Corpus only

File	PPMD (order 5) (BPC)	PPMD with LIPT (BPC)	PPMonstr (BPC)	PPMonstr with LIPT (BPC)
bib	1.86	1.83	1.83	**1.81**
book1	2.30	2.23	2.23	**2.19**
book2	1.96	1.91	1.95	**1.91**
news	2.35	2.31	2.31	**2.28**
paper1	2.33	2.21	2.25	**2.14**
paper2	2.32	2.17	2.21	**2.08**
paper3	2.58	2.37	2.46	**2.25**
paper4	2.89	2.57	2.77	**2.47**
paper5	2.98	2.74	2.86	**2.65**
paper6	2.41	2.29	2.33	**2.23**
progc	2.36	2.30	2.29	**2.24**
progl	1.68	1.61	1.64	**1.59**
progp	1.70	1.68	1.65	**1.64**
trans	1.47	1.41	1.45	**1.42**
Average	2.23	2.12	2.16	**2.06**

Table 11: BPC Comparison of PPMD, PPMD with LIPT, PPMonstr, and PPMonstr with LIPT for the text files in Calgary Corpus only

.

Corpus	Average compression time without preprocessing of input file (in sec).			Average compression time when preprocessed using LIPT (in sec).		
	BZIP2	GZIP	PPMD	BZIP2	GZIP	PPMD
Calgary	0.3994	0.4861	12.8256	0.8196	1.4612	12.5968
Canterbury	2.678	3.8393	71.4106	4.5754	11.1488	67.4141
Gutenburg	4.8433	3.4233	103.5167	8.7927	12.4661	105.4828
Total time for whole corpus	7.9207	7.7487	187.7529	14.1877	25.0761	185.4937
Performance of LIPT + {BZIP2, GZIP and PPMD} over simple BZIP2, GZIP and PPMD compression techniques.				**1.7912 times slower**	**3.236 times slower**	**1.012 times faster**

Table 12: Compression times using standard compression methods and compression with LIPT transformation

.

Corpus	Average decompression time without preprocessing of input file (in sec).			Average decompression time when preprocessed using LIPT reversible transformation. (in sec).		
	BZIP2	GZIP	PPMD	BZIP2	GZIP	PPMD
Calgary	0.1094	0.0233	13.0022	0.1847	0.1058	9.4097
Canterbury	0.6506	0.1166	71.1986	1.1803	0.6977	65.0237
Gutenburg	1.1733	0.2133	100.967	2.3736	1.517	99.7866
Total time for whole corpus	1.9333	0.3532	185.1678	3.7386	2.3205	174.22
Performance of LIPT + {BZIP2, GZIP and PPMD} over simple BZIP2, GZIP and PPMD decompression techniques.				**2.31 times slower**	**6.56 times slower**	**1.0003 times faster**

Table 13: Decompression times using standard decompression methods and decompression with LIPT transformation

Table 15: BPC Comparison of Bzip2 with LIPT and Bzip2 with NIT

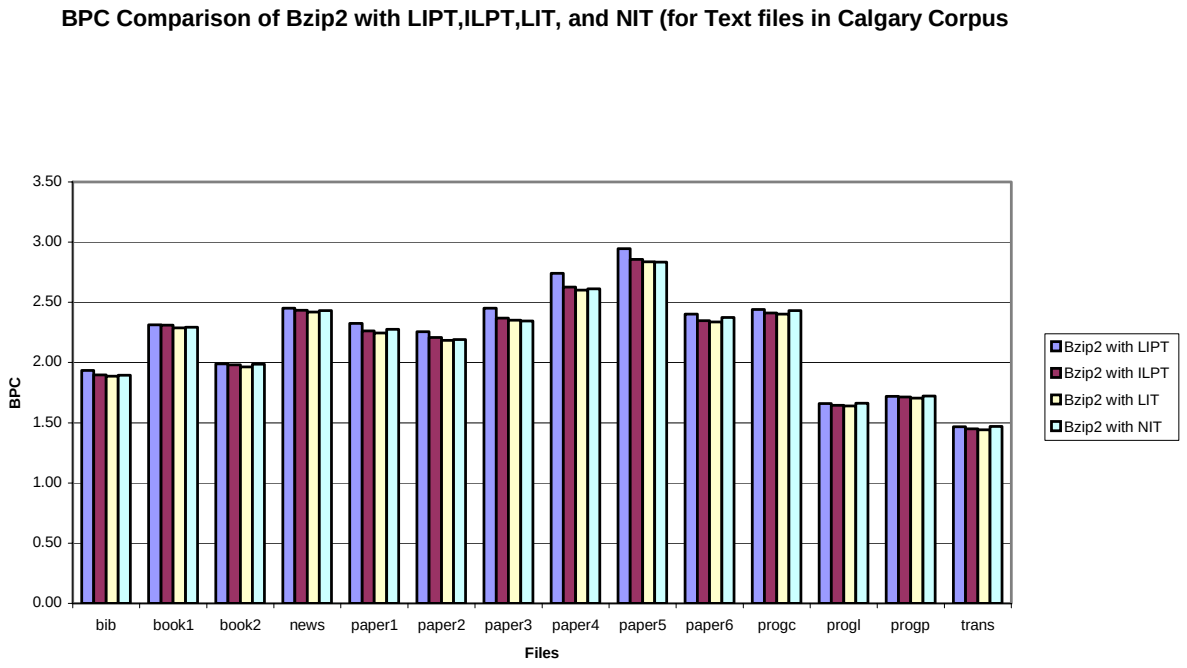

Figure 4: Chart comparing Bzip2 in conjunction with LIPT, ILPT, LIT, and NIT for Calgary corpus

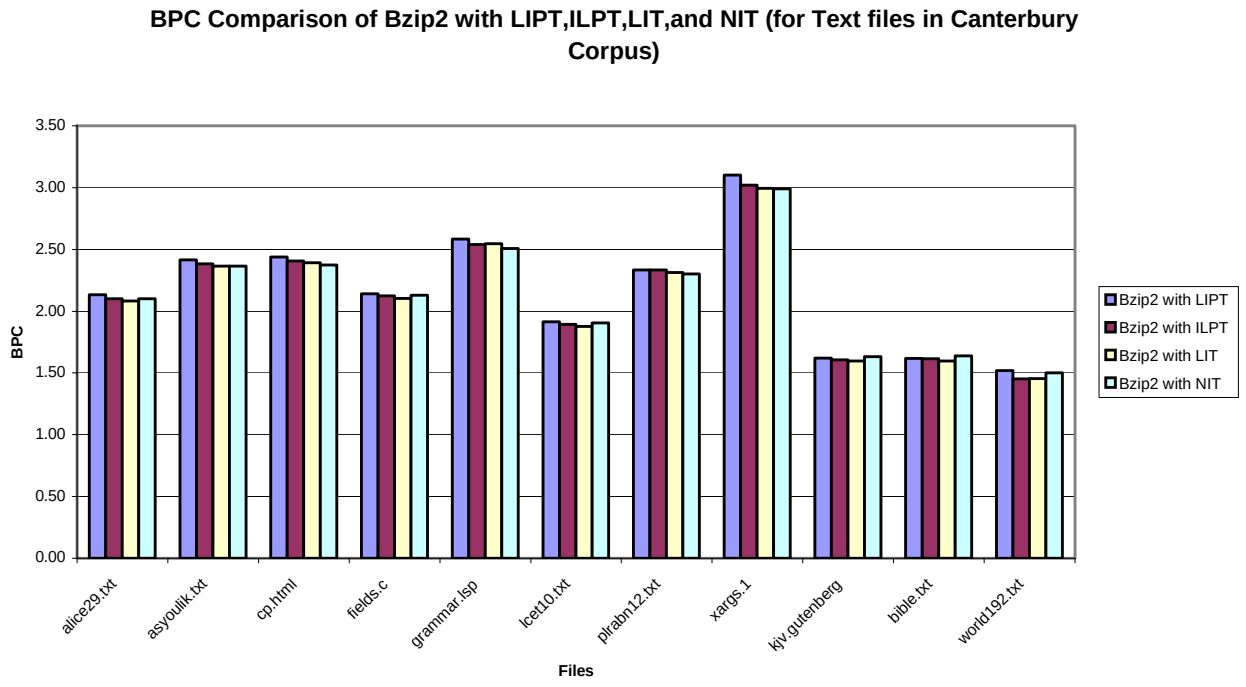

Figure 5: Chart comparing Bzip2 in conjunction with LIPT, ILPT, LIT, and NIT for Canterbury corpus

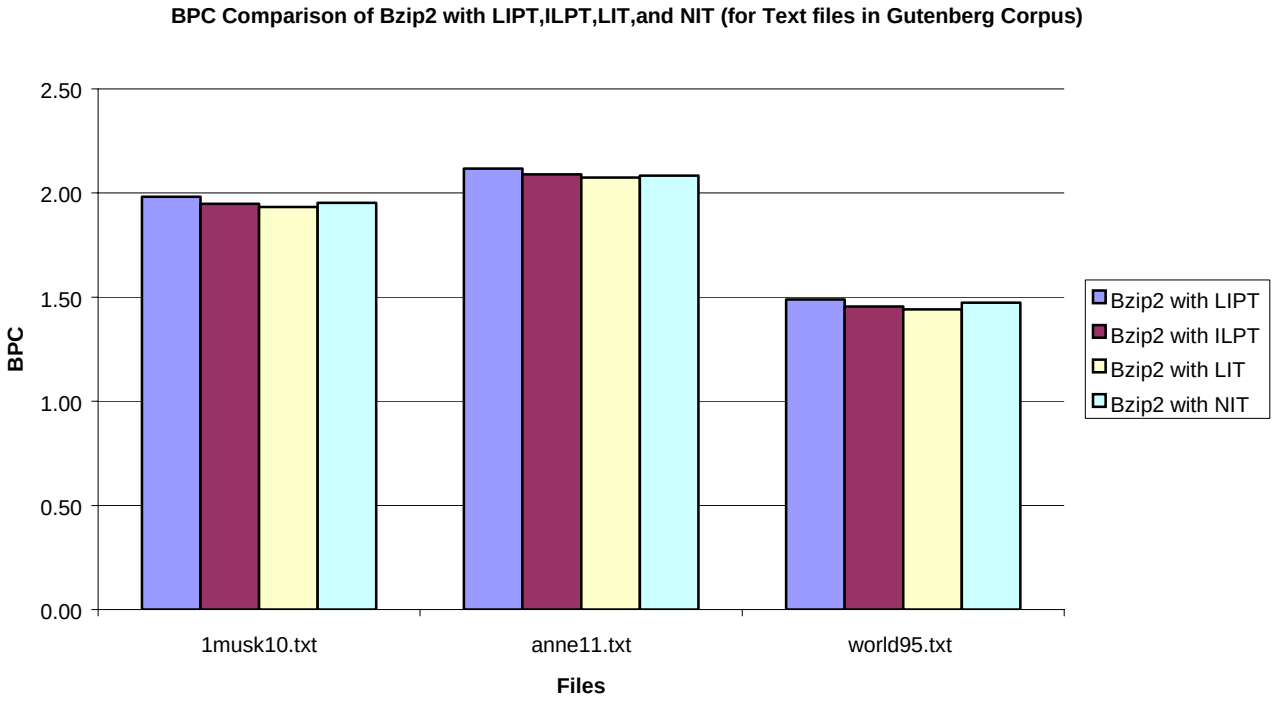

Figure 6: Chart comparing Bzip2 in conjunction with LIPT, ILPT, LIT, NIT, and 4-BIT for Gutenberg corpus

FileNames	PPMD with LIPT	PPMD with 4-BIT
Calgary		
paper5	2.74	**2.64**
paper4	2.57	**2.46**
paper6	2.29	**2.21**
Progc	2.30	**2.25**
paper3	2.37	**2.28**
Progp	1.68	**1.66**
paper1	2.21	**2.13**
Progl	1.61	**1.56**
paper2	2.17	**2.10**
Trans	1.41	**1.37**
Bib	1.83	**1.79**
news	2.31	**2.27**
book2	1.91	**1.88**
book1	2.23	**2.22**
Average BPC	2.12	**2.06**

FileNames	PPMD with LIPT	PPMD with 4-BIT
Canterbury		
grammar.lsp	2.21	2.16
xargs.1	2.73	2.62
fields.c	1.97	1.93
cp.html	2.22	2.18
asyoulik.txt	2.35	2.31
alice29.txt	2.06	2.01
lcet10.txt	1.86	1.83
plrabn12.txt	2.27	2.26
world192.txt	1.45	1.39
bible.txt	1.57	1.52
kjv.gutenberg	1.55	1.51
Average BPC	2.02	**1.97**

FileNames	PPMD with LIPT	PPMD with LIPT
Gutenberg		
Anne11.txt	2.04	2.00
1musk10.txt	1.85	1.82
World95.txt	1.45	1.38
Average BPC	1.78	**1.74**

Table 18: BPC comparison of PPMD with LIPT, and PPMD with LIT.

File	YBS (BPC)	YBS with LIPT (BPC)	YBS with LIT (BPC)
bib.ybs	1.93	1.91	**1.87**
book1.ybs	2.22	2.16	**2.14**
book2.ybs	1.93	1.90	**1.88**
news.ybs	2.39	2.36	**2.33**
paper1.ybs	2.40	2.28	**2.21**
paper2.ybs	2.34	2.18	**2.10**
paper3.ybs	2.63	2.39	**2.29**
paper4.ybs	3.06	2.71	**2.58**
paper5.ybs	3.17	2.91	**2.79**
paper6.ybs	2.52	2.36	**2.29**
progc.ybs	2.48	2.43	**2.39**
progl.ybs	1.69	1.63	**1.61**
progp.ybs	1.71	1.70	**1.69**
trans.ybs	1.48	1.44	**1.41**
Average	2.28	2.17	**2.11**

Table 19: BPC comparison of YBS, YBS with LIPT, and YBS with LIT for Calgary Corpus only

File	RK (BPC)	RK with LIPT (BPC)	RK with LIT (BPC)
bib	1.75	1.72	1.70
book1	**2.16**	**2.16**	2.15
book2	**1.83**	1.85	1.84
news	2.23	2.21	2.19
paper1	2.27	2.17	2.11
paper2	2.21	2.11	2.06
paper3	2.52	2.33	2.24
paper4	2.90	2.63	2.49
paper5	3.01	**2.81**	2.69
paper6	2.35	2.27	2.20
progc	2.31	2.27	2.24
progl	1.48	**1.44**	1.42
progp	1.50	**1.49**	1.48
trans	1.24	1.23	1.21
Average	2.12	**2.05**	**2.00**

Table 20: BPC comparison of RK, RK with LIPT, and RK with LIT for Calgary Corpus only

File	PPMonstr (BPC)	PPMonstr with LIPT (BPC)	PPMonstr with LIT (BPC)
bib	1.83	1.81	**1.77**
book1	2.23	2.19	**2.14**
book2	1.95	1.91	**1.86**
news	2.31	2.28	**2.23**
paper1	2.25	2.14	**2.06**
paper2	2.21	2.08	**2.00**
paper3	2.46	2.25	**2.16**
paper4	2.77	2.47	**2.35**
paper5	2.86	2.65	**2.54**
paper6	2.33	2.23	**2.15**
progc	2.29	2.24	**2.19**
progl	1.64	1.59	**1.53**
progp	1.65	1.64	**1.62**
trans	1.45	1.42	**1.38**
Average	2.16	2.06	**2.00**

Table 21: BPC Comparison of PPMonstr, PPMonstr with LIPT, and PPMonstr with LIT for the text files in Calgary Corpus only

List of References

1. Z. Arnavut, "Move-to-Front and Inversion Coding", Proceedings of Data Compression Conference, IEEE Computer Society, Snowbird, Utah, March 2000, pp. 193-202.
2. B. Balkenhol, S. Kurtz, and Y. M. Shtarkov, "Modifications of the Burrows Wheeler Data Society, Snowbird Utah, March 1999, pp. 188-197.Compression Algorithm", Proceedings of Data Compression Conference, IEEE Computer
3. M. Burrows and D.J. Wheeler, "A Block-Sorting Lossless Data Compression Algorithm", SRC Research Report 124, Digital Systems Research Center, Palo Alto, CA, 1994.
4. B. Chapin, "Switching Between Two On-line List Update Algorithms for Higher Compression of Burrows-Wheeler Transformed Data", Proceedings of Data Compression Conference, IEEE Computer Society, Snowbird Utah, March 2000, pp. 183-191.
5. J.G. Cleary, W.J. Teahan, and Ian H. Witten, "Unbounded Length Contexts for PPM", Proceedings of Data Compression Conference, IEEE Computer Society, Snowbird Utah, March 1995, pp. 52-61.

6. M. Effros, “PPM Performance with BWT Complexity: A New Method for Lossless Data Compression”, Proceedings of Data Compression Conference, IEEE Computer Society, Snowbird Utah, March 2000, pp. 203-212.
7. P. Fenwick, “Block Sorting Text Compression”, Proceedings of the 19th Australian Computer Science Conference, Melbourne, Australia, January 31 – February 2, 1996.
9. N.J. Larsson, “The Context Trees of Block Sorting Compression”, Proceedings of Data Compression Conference, IEEE Computer Society, Snowbird Utah, 1998, pp 189-198.
10. A. Moffat, “Implementing the PPM data Compression Scheme”, IEEE Transaction on Communications, 38(11), pp.1917-1921, 1990
11. K. Sadakane, T. Okazaki, and H. Imai, “Implementing the Context Tree Weighting Method for Text Compression”, Proceedings of Data Compression Conference, IEEE Computer Society, Snowbird Utah, March 2000, pp. 123-132.
12. J. Seward, “On the Performance of BWT Sorting Algorithms”, Proceedings of Data Compression Conference, IEEE Computer Society, Snowbird Utah, March 2000, pp. 173-182.
13. F. Willems, Y.M. Shtarkov, and T.J.Tjalkens, “The Context-Tree Weighting Method: Basic Properties”, IEEE Transaction on Information Theory, IT-41(3), 1995, pp. 653-664.
14. I.H.Witten, A. Moffat, T. Bell, “Managing Gigabyte, Compressing and Indexing Documents and Images”, 2nd Edition, Morgan Kaufmann Publishers, 1999.
15. D. Salomon, “Data Compression: The Complete Reference”, 2nd Edition, Springer Verlag, 2000.