道しるべ：データ圧縮の基礎 — ディジタルデータのロスなし圧縮

横尾 英俊　yokoo@cs.gunma-u.ac.jp
群馬大学工学部 情報工学科

はじめに

データ圧縮 (data compression) とは，たとえば，1メガバイトのデータを600キロバイトにすることである．ただそれだけのことでありながら，その内容と守備範囲は，一つの分野とは見なせないほどに広範かつ多様な側面を有するものになっている．何をどのような意味で圧縮するかによって手法が分かれるのはもちろんのこと，基礎理論から実装技術を経て特許や標準化関連の話題に至るまで，カルチャーもコミュニティも多岐にわたる．また，データ圧縮という固有の目的に加え，それ以外の目的への応用も活発である．そのような“データ圧縮”全般を見渡して道しるべを与えることは筆者には到底不可能なので，本稿ではデータ圧縮のごく限られた側面について，いくつかの切り口を紹介してみたい．なお，比較的広範の話題を扱った教科書にGibsonほか[6]がある．

データ圧縮は，大別すると，**歪み** (ひずみ) **を許す**圧縮と**歪みのない**圧縮の2種類に分類できる．**歪み**あるいは**損失**とは，圧縮したデータを元に戻したときの，再現データと本来のデータとの食い違いのことである．本稿で取り上げるのは，これらのうちの**離散データのロスなし圧縮**，すなわち，歪みをまったく許さない場合である．**歪みなし**は，**無歪み**，(情報) **無損失**，**可逆**，*lossless*，*reversible*などとも呼ばれる．典型例としては，計算機のファイルの圧縮を思い浮かべればよいであろう．現在ファイル圧縮の分野では，英文データならば，1文字あたり2ビットを切るかどうかのせめぎ合いの段階にある．

本稿の目的は，離散データの歪みなし圧縮に学術的な側面から取り組もうとする研究者や学生に対し，その諸手法への入口を提供することにある．パソコンユーザ向けの圧縮ハウツー本に書いてあることではなく，その背後にひそむ圧縮原理や基礎理論への入口である．

基礎は情報理論にあり

データ圧縮を歪みなし圧縮に限定した場合，“どのようなデータでも必ず圧縮します”というデータ圧縮法は，“どのようなデータでも”ということを字句通りに解釈する限り，存在しない．にもかかわらず，そのような“データ圧縮法”がたびたび登場するという[9]．このような謳い文句が論理的に正しくないことは容易に分かることである．むしろ，ほとんどすべてのデータは圧縮することができないのである．では，どのようなデータなら圧縮することができるのであろう．

たとえば，ある特定のデータxが与えられた場合，圧縮可能かどうかは判定できるのであろうか．実は，このような問題設定だけでは妥当な結論が得られない．それは，xをビット0で表し，x以外のデータの先頭にビット1を前置することで，どのようなxでも1ビットに圧縮できてしまうからである．したがって，圧縮可能性は特定のデーターつを対象に考えるべきものではなく，データの集合に対して考えるべきものといえる．このような立場をとるのが**情報理論**である．

情報理論では，データの集合と確率分布の組を情報源 (source) と呼び，データ圧縮を情報源符号化 (source coding) という．データ系列，すなわち情報源からの出力が無限に長い場合を考える．データ集合の任意の要素を十分長く観察したとき，この情報源の確率構造がすべて観測できるなら，この情報源は**エルゴード的**

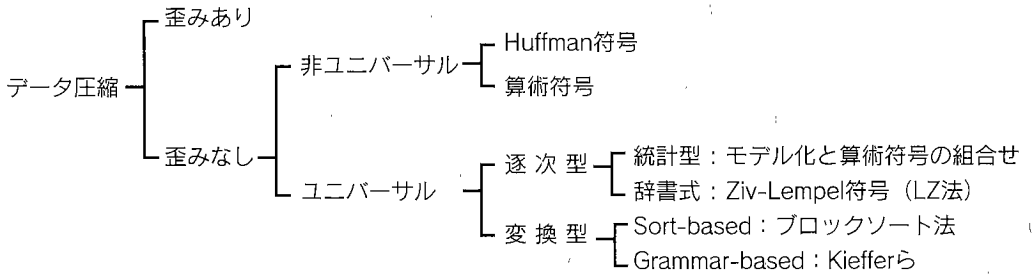
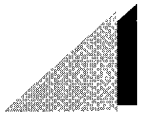


図-1 データ圧縮，特に歪みなしデータ圧縮の分類

(ergodic) であるという．エルゴード性を理解するには，エルゴード的でない例を考えるとよい．たとえば，0だけを無限に並べた系列1個と1だけを無限に並べた系列1個とから成り，両系列が等確率で選ばれる情報源はエルゴード的ではない．一方の系列をいくら長く観察しても，0と1が半々であるということは観察できないからである．さて，ある有限長のブロック（nグラム）単位のエントロピーH_nの平均増加率H_n/nが$n\to+\infty$で収束するなら，その極限値をエントロピー・レート (entropy rate) と呼ぶ[5),8)]．定常情報源ではエントロピー・レートが存在し，それがデータ圧縮の限界であることが示される．なお，歪みなし圧縮のことを，文脈によっては，エントロピー符号化と呼ぶ場合がある．

基礎となる素養をがっちり固めてから具体的な問題に取り組もうという正統派には，いま挙げたCover-Thomas[5)]，韓・小林[8)]などをまず薦めたい．論文を入口として必要に応じて基礎事項を勉強する方向をとろうということであれば，後述するWynerほか[19)]，Willemsほか[18)]，Kiefferほか[11)]が薦められる．これらの論文は，データ圧縮の最先端の話題から情報理論的側面に入っていくためのまったく異なる入射角を提供している．より基礎的な事項については，たいていの情報理論の教科書が役に立つであろう．

ユニバーサル符号



情報理論とデータ圧縮といえばまず真っ先に登場するのがハフマン (Huffman) 符号である．すでに，ハフマン符号が発表されてから来年 (2002年) で50年になろうとしている．ハフマン符号は，ファクシミリ符号化をはじめMPEGやその一部であるMP3などの画像や音声の符号化で，また，情報検索向けの圧縮法[24)]などで依然としてよく利用されている．よく知られているように，ハフマン符号の基本原理は，頻度の高い記号を短い符号語で表し，頻度の低い記号には相対的に長い符号語を割り当て，平均としてデータを圧縮しようとするものである．このことからも分かるように，ハフマン符号の構成には，情報源の確率構造を表すパラメータについての知識が要求される．一方，そのような知識を前提としないのが**ユニバーサル符号** (universal code)[10)]である．

ユニバーサル符号の研究はかなり以前からあったようであるが，1970年代半ばのZiv-Lempel符号 (LZ法) と算術符号 (arithmetic code) の出現によって理論的にも実用的にも爆発的な開花をみせ，その後の歪みなしデータ圧縮の歴史をほぼ支配しつつ今日に至っている．これらのうち，算術符号は確率パラメータの値を必要とするので，**図-1**ではユニバーサル符号には含めていない．しかし，そのような確率パラメータが1時刻ごと，あるいは1ビットごとに変化したとしても，算術符号はその変化に追従することができる．そのような意味で算術符号は万能であり，パラメータを推定する適切なモデルを組み合わせることで，ユニバーサルな圧縮法を作ることができる．モデルの例としてはPPM (Prediction by Partial Match) やCTW (Context-Tree Weighting) と呼ばれるものが代表的である[3),15)]．CTWを提案したWillemsらの論文[18)]は，モデル化に関係する重要な概念への入口として参考になる．算術符号のプログラム例は，文献13)，17) やWeb上[1)]に数多く存在している．

LZ法はファイル圧縮ツールの基礎としてよく知られ，

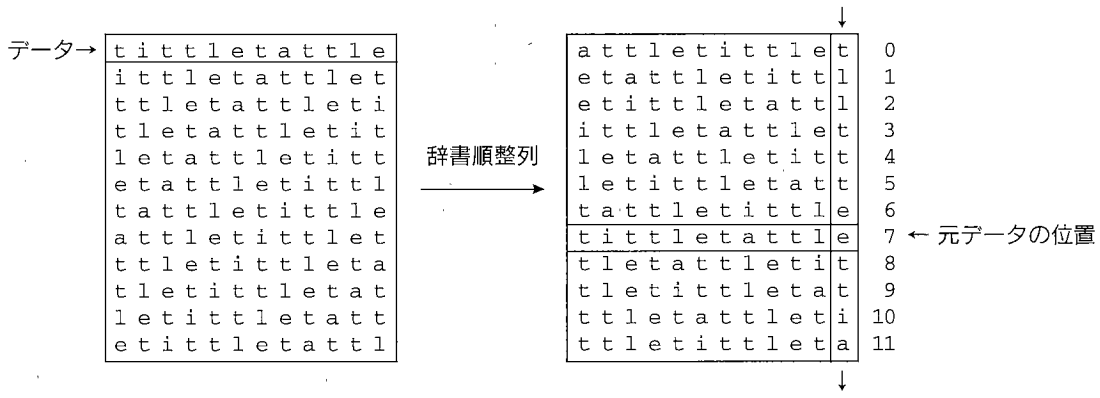
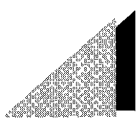


図-2 ブロックソート・データ圧縮法のためのBW変換.
データ系列'tittletattle'に対する変換結果は，<'tllttteettia',7>である.

ユニバーサル符号の代表格である．圧縮ツールに詳しくても彼らの原論文のことは知らないという場合には，やはり一度は目を通しておくべきであろう．UNIXのgzipをはじめ多くの圧縮ツールの基礎になっているLZ77法は，彼らの1977年の論文[22]で発表されている．また，LZ78法[23]もその後，多くの改良や種々の話題を生んでいる．

LZ法の誕生から今日までの歴史は，実用化や特許にかかわるものと理論研究とに大別することができる．実用化にあたっては，効率化のための種々の工夫が必要であり[15),17),20)]，それらのいくつかは特許をめぐる話題の大きな流れを形成している[9),14)]．一方，理論研究の主流は，LZ法の漸近的最良性および冗長度の評価にある．漸近最良とは，十分長いデータに対し，ユニバーサル符号の圧縮力が圧縮限界（たとえば，エントロピー・レート）に漸近的に近づくことをいう．LZ法の変種の定常エルゴード情報源に対する漸近的最良性を中心に，その達成の速度（冗長度）を含めて精力的に調べられている．この方面のソースとしては，Wynerらによる解説論文[19]がある．また，データ圧縮に関する小特集号[16]にも，LZ法の理論研究に関連する解説が企画されている．

研究の進め方 — ブロックソート法を例に



離散データの歪みなし圧縮研究へのアプローチを例を交えて簡単に紹介しよう．

この分野の目標の一つは，新しい圧縮法を最初に提案することである．しかし，そればかりを目指していたのでは“食べていけない”ので，既存の方法に関係する研究も併せて行うことになる．前章でふれた諸方法を既存のものとするなら，それに対する新手法の一つの例はBurrowsとWheeler[4]によるブロックソート（block sorting）法である．図-1に示したように，LZ法などがデータ系列をそのまま前から順に逐次的に符号化するのに対し，ブロックソート法はデータ系列をいったん別の系列に変換する．BW変換と呼ばれるその変換をデータ例'tittletattle'に対して実行した様子を**図-2**に示す．変換結果は，<'tllttteettia',7>である．変換結果の2番目の要素'7'は，データの巡回シフトを辞書式順序に整列したときの元データの位置である．

筆者は，ここまでの話を論文[4]を目にする前に聞く機会があった．この変換が可逆な変換であることを直ちに確認して — 読者も確かめられたい — の第一印象は「あっ！　やられた」というものであった．というのも，この方法には，それに関係する研究を生み出すシーズとしての面白さが十分あると考えたからである．たとえば，次のような点である．

• 分析的研究

漸近最良か否か．そうであるとするなら，どのような情報源のクラスに対し，どのような前提で証明することができるのか．BW変換が情報源の構造をある意味で壊してしまうので，この問題は，従来の同種の問題とは異なる難しさと面白さを含んでいる．完全な解明はまだなされていないが，端緒を与える研究がいくつかある[2)]．

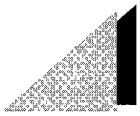
• 合成的研究

BW変換そのものはデータ圧縮法ではないので，変換

後のデータにさらにMTF (Move to Front) と呼ばれる変換やエントロピー符号化を加えなければならない．ここにさまざまな変種の可能性がある．また，BW変換そのものを実行するためのアルゴリズム—Suffix arrayと呼ばれるデータ構造の構成とほぼ等価—にも効率化の余地がある．すでに，これらの点で種々の改良を取り入れた圧縮ツール (bzip2など) が実用化されている．

上の二つの視点は，ブロックソート法に限らず，データ圧縮法研究に共通する典型的アプローチである．また，一見まったく異なる圧縮法であっても，圧縮という目的を共有する以上，相互に深く関係するということが少なくない．一般に，分析的研究と合成的研究に加え，特定の圧縮法をそれ固有の視点とは異なる角度で再解釈することにより，新たな展望を開き得る場合がある．実際，ブロックソート法もその他の方法とまったく関係がないわけではない．あるいは，異なる手法を関連づける“第3の方法”が存在する場合もある．たとえば，ブロックソート法と逐次型の従来法とを関連づける圧縮法として文脈ソート法[21]がある．さらに，再解釈の抽象度を高め一般化することで，その具体例として新たな方法を得ることもできる．最近のKiefferらの一連の研究[11),12)]は，LZ法などを文法の学習という視点でとらえた，そのような方向の研究とみることができる．Kiefferらの方法は，データを直接圧縮するのではなく，それを生成する“文法”を圧縮しようとするものである．

その他の文献など



理屈よりはとにかく手法を，という方には各種の手法を網羅しているSalomonの教科書の最新版[15]をお薦めする．講義録をWeb上で公開している例としては，Blelloch[3]のサイトがある．そのような講義録へのリンクも含めて，データ圧縮の種々の情報を集めたページとして有村のリンク集[1]が充実している．また，有村のページやそれに類似したサイトはGoogle[7]からたどることができる．

データ圧縮に関係する研究成果は，この分野の広がりを反映して，情報理論，通信理論，アルゴリズム論関連など多岐の分野にわたり，多くの雑誌や研究集会に散在している．データ圧縮に特化した国際会議としてはDCC (Data Compression Conference) がある．その他の諸会議は，上述の有村のページからたどることができる．圧縮法の性能評価を経験的に行うための標準データへもこのページ経由でアクセス可能である．ファイル圧縮法評価に使われる標準データとしては，CalgaryおよびCanterburyの両コーパスが有名である．これらのコーパスのサイトからは，評価済の圧縮法のサイトへのリンクも張られている．おそらく読者の多くは，どの方法がベストなのかという疑問を持たれるに違いない．そのような場合は，コーパスのサイトの比較結果にあたっていただきたい．ただし，性能の評価には種々の要因を含めなければならず，コーパスによる比較結果だけがすべてというわけではない．

以上，歪みなしデータ圧縮のいくつかの側面を大急ぎで紹介した．読者の一助となれば幸いである．

参考文献

1) 有村光晴: Bookmarks on Source Coding/Data Compression, <URL:http://www.hn.is.uec.ac.jp/~arimura/compression_links.html>
2) Arimura, M. and Yamamoto, H.: Asymptotic Optimality of the Block Sorting Data Compression Algorithm, IEICE Trans. Fundamentals, Vol.E81-A, No.10, pp.2117-2122 (1998).
3) Blelloch, G.: Compression, <URL:http://www.cs.cmu.edu/afs/cs.cmu.edu/project/pscico-guyb/realworld/www/compress.html>
4) Burrows, M. and Wheeler, D. J.: A Block-Sorting Lossless Data Compression Algorithm, SRC Research Report, 124 (1994). <URL:http://gatekeeper.dec.com/pub/DEC/SRC/research-reports/abstracts/src-rr-124.html>
5) Cover, T. M. and Thomas, J. A.: Elements of Information Theory, John Wiley & Sons (1991).
6) Gibson, J. D. et al.: Digital Compression for Multimedia: Principles & Standards, Morgan Kaufmann (1996).
7) Google Web Directory: Compression, <URL: http://directory.google.com/Top/Computers/Algorithms/Compression/>
8) 韓 太舜, 小林欣吾: 情報と符号化の数理, 培風館 (1999).
9) Hankerson, D., Harris, G. A. and Johnson, Jr. P. D.: Introduction to Information Theory and Data Compression, CRC Press (1997).
10) 情報理論とその応用学会 (編): 情報源符号化—無歪みデータ圧縮, 培風館 (1998).
11) Kieffer, J. C. and Yang, E.: Grammar-Based Codes: A New Class of Universal Lossless Source Codes, IEEE Trans. Inform. Theory, Vol.46, No.3, pp.737-754 (2000).
12) Kieffer, J. C., Yang, E., Nelson, G. J. and Cosman, P.: Universal Lossless Compression via Multilevel Pattern Matching, IEEE Trans. Inform. Theory, Vol.46, No.4, pp.1227-1245 (2000).
13) 荻原剛志, 山口 英 (訳): データ圧縮ハンドブック, 改定第2版, トッパン (1996).
14) 奥村晴彦: データ圧縮と特許, <URL:http://www.matsusaka-u.ac.jp/~okumura/compression/patents.html>
15) Salomon, D.: Data Compression: The Complete Reference, 2nd Edition, Springer (2000). <URL:http://www.ecs.csun.edu/~dxs/DC2advertis/DComp2Ad.html>
16) ユニバーサル符号とデータ圧縮小特集号: 電子情報通信学会論文誌 A, Vol.J84-A, No.6 (2001) (予定).
17) 植松友彦: 文書データ圧縮アルゴリズム入門, CQ出版社 (1994).
18) Willems, F. M. J., Shtarkov, Y. M. and Tjalkens, T. J.: The Context-Tree Weighting Method: Basic Properties, IEEE Trans. Inform. Theory, Vol.41, No.3, pp.653-664 (1995).
19) Wyner, A. D., Ziv, J. and Wyner, A. J.: On the Role of Pattern Matching in Information Theory, IEEE Trans. Inform. Theory, Vol.44, No.6, pp.2045-2056 (1998).
20) 山本博資: ユニバーサルデータ圧縮アルゴリズム: 原理と手法, 情報処理, Vol.35, No.7, pp.600-608 (July 1994).
21) Yokoo, H.: Data Compression Using a Sort-Based Context Similarity Measure, Computer Journal, Vol.40, No.2/3, pp.94-102 (1997).
22) Ziv, J. and Lempel, A.: A Universal Algorithm for Sequential Data Compression, IEEE Trans. Inform. Theory, Vol.IT-23, No.3, pp.337-343 (1977).
23) Ziv, J. and Lempel, A.: Compression of Individual Sequences via Variable-Rate Coding, IEEE Trans. Inform. Theory, Vol.IT-24, No.5, pp.530-536 (1978).
24) Ziviani, N. et al.: Compression: A Key for Next-Generation Text Retrieval Systems, Computer, Vol.33, No.11, pp.33-44 (2000).

(平成12年12月18日受付)