予測残差長の量子化を用いた 可逆データ圧縮ハードウェアの性能評価

上野知洋[†1]　佐野健太郎[†1]　山本悟[†1]

概要：計算機性能の向上に伴う科学技術シミュレーションの大規模化や高精度化には，大量の浮動小数点データに対する効率的なデータ移動が求められる．しかしながら，外部メモリ帯域はプロセッサと比べて性能向上が困難であり，帯域不足による計算性能低下は計算機にとって大きな問題である．このような問題に対し，ハードウェアによる高スループットな可逆データ圧縮の適用により，帯域不足を解消する手法が提案されている．本稿は，データ圧縮ハードウェアの面積を削減する予測残差長の量子化手法を用いたデータ圧縮の性能評価を行う．本手法では，量子化の際に選択する値が圧縮性能と回路面積に大きく影響するため，諸問題の要求に応じた柔軟な構成が可能である．残差長の選択による圧縮性能と回路面積をモデル化された数値データ分布を用いて評価し，データや計算時の要求に応じた可逆圧縮を提供する数値データ圧縮ハードウェアの設計指針について考察する．

キーワード：データ圧縮，浮動小数点，回路面積，数値シミュレーション

Evaluation of Lossless Data Compression Hardware with Quantized Length of Prediction Residual

TOMOHIRO UENO[†1]　KENTARO SANO[†1]　SATORU YAMAMOTO[†1]

Abstract: Large scale and high precision scientific technological simulations require efficient data transfers because of numerous data to handle by the computers. However, an insufficiency of bandwidth is a significant cause of a decrease of performance. Some studies are proposed to solve the problem by employing lossless data compression to enhance the bandwidth. This paper presents quantized length of prediction residual for a practical small data compression hardware. An evaluation shows the proposed method achieves area reduction with moderate performance reduction. Moreover, this paper also shows a selection method according to the target application and the property of computing.

Keywords: Data compression, Floating-point operation, Circuit area, Numerical simulation

1. 序論

科学技術計算等の大規模化，高精度化に伴い，大量の数値データに対する高速処理要求が高まっている．大規模な数値計算に利用される高性能計算機は，多数のプロセッサを用いて大規模に並列化されている．このような計算機による大規模数値計算では，プロセッサにおける処理が要求する帯域に対して伝送路の帯域が不足する場合，計算性能の低下を引き起こす[1,2]．近年では infiniBand[3]等の高性能な通信規格により高帯域なネットワークを実現されているが，システムの構成やアプリケーションの種類により要求される帯域は異なる．特に大量のデータを扱うデータインテンシブな処理においては，帯域の不足は全体性能を低下させる大きな要因の一つになる[12].

この問題に対して，リアルタイムな可逆データ圧縮を用いて要求帯域を削減する手法が提案されている[4-6]．計算機内における局所的なデータ移動の際に圧縮されたデータを通信することにより，帯域不足に起因する計算性能の低下を防ぐ手法である．特に大規模データを扱う数値計算におけるデータの圧縮には，予測を用いたデータ圧縮アルゴリズムが提案されている[7,8]．これは，計算自体に影響しない可逆圧縮であり，エントロピー符号化の効果が小さい浮動小数点データに対しても効果的な圧縮を実現できる．

計算機内の局所的なデータ移動に対してこのようなデータ圧縮を適用するためには，非常に高スループットな処理が求められる[9]．この要求に対し本研究グループは，ハードウェアによる数値データ圧縮を提案し，データ圧縮を実現するハードウェアの設計，実装，および性能評価を行っている[5,6]．提案手法は，FPGA を用いたストリーム計算における専用計算パイプラインに付随する形で実装され，計算パイプラインの入出力帯域を向上させる．

本研究において設計したハードウェアは，高い圧縮性能とスループットを達成することを実証したが[6]，一方で実際の科学技術シミュレーションに適用するためには，回路面積の削減が必要となることが明らかになった．これは，複数の変数を持つ現実の数値シミュレーションでは、変数のそれぞれに必要となるデータ圧縮・展開ハードウェアを多数実装する必要があるためである．この結果，データ圧

†1 東北大学
Tohoku University

縮・展開ハードウェアが要求する総回路面積はチャネル数に比例して増加するため，実用的な問題への適用には膨大なハードウェア資源が必要になってしまう．

この問題に対し，データ圧縮・展開ハードウェアの回路面積を削減するため，予測残差長を量子化して処理を簡単化する手法が提案されている[5]．これは，回路面積増大の原因である圧縮データの符号化回路において，予測残差長の取り得る値を限定することにより回路面積を削減する手法である．回路面積の削減は圧縮性能の低下につながるが，この性能低下を軽減するため，数値データにおける予測残差長分布の偏りを利用したより一般的な手法を提案する．この手法は，採用する残差長の選択により圧縮性能と回路面積が変化するため，データやデバイスに応じて柔軟にハードウェアを生成できる．

本稿の目的は，予測残差長の量子化手法を一般化した手法の提案と，圧縮性能および回路面積への影響を調査することである．加えて，圧縮対象となるデータの統計的特徴に基づき適切な量子化を選択する基準を明らかとすることである．この際，数値データにおける残差長分布を正規分布としてモデル化し，期待値や分散を変化させて圧縮性能の評価を行う．また圧縮性能と回路面積の評価結果から，複数の選択値について比較し値の選択について議論する．

本稿の構成は以下の通りである．2 節では予測による浮動小数点数値データの圧縮手法とハードウェア化について述べる．また，回路面積の大きさとその原因となる符号化について詳しく述べる．3 節では予測残差長の量子化について述べ，それによって回路面積が削減できることを示す．4 節では，予測残差長分布のモデルを用いた圧縮性能評価と FPGA 実装における回路面積の評価について示し，量子化について議論する．5 節は本稿の結論である．

2. 予測残差を用いた数値データの圧縮

数値計算において用いられる浮動小数点データの圧縮方式として，シンボルの出現頻度に基づくエントロピー符号は適当でない．また，データ圧縮処理に許容されるオーバーヘッドは極めて小さく，完全な可逆圧縮が求められることから，数値データに対して効果的かつ．ハードウェアによる高スループット処理が可能な手法が求められる．

数値データに対する高スループットな可逆圧縮手法として，予測計算と残差の符号化を用いた手法が提案されている[7]．本研究の先行研究ではこの手法に基づき，数値的連続性を利用した一次元多項式予測を用いた手法のハードウェア化と性能評価を行った[5,6]．この研究において，実装したハードウェアは高い圧縮性能とスループットを実現したが，実用問題への適用において回路面積の増加が問題になることが分かった．以下に，データ圧縮における予測残差長の利用と回路面積の問題について述べる．

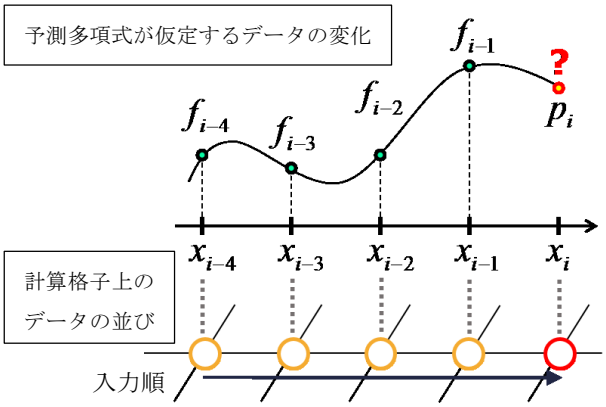

図 1　一次元多項式による数値データの予測

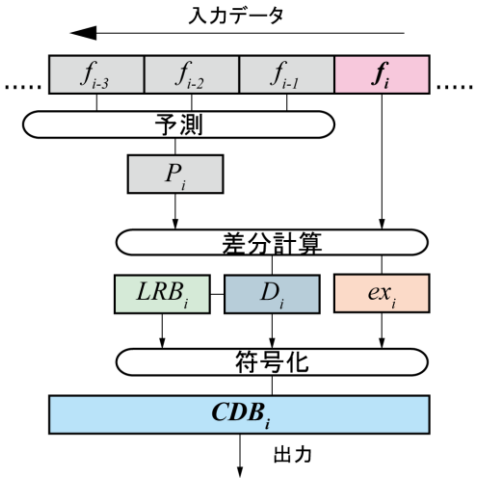

図 2　予測に基づく数値データ圧縮アルゴリズム

2.1 予測に基づく可逆圧縮アルゴリズム

数値シミュレーションにおけるデータの特徴の一つに，空間的・時間的な数値的連続性を持つことがあげられる．圧縮性流体の計算における衝撃波等の不連続な状況を除くと，数値シミュレーションのデータは連続的に変化する値を標本化により離散的に示したものといえる．このことから，隣接する，もしくは近隣のデータを使って，ある点における値を予測することが可能である．

本手法における予測計算には，(1)式に示す 3 次の 1 次元多項式を用いる．本式は 1 次元のデータ列として入力される数値データにおいて，局所的なデータの並びが 3 次の関数に従うと仮定した予測式である(図 1)．

$$p_i = 4f_{i-1} - 6f_{i-2} + 4f_{i-3} - f_{i-4} \quad (1)$$

p は予測値，fは実際に入力された値であり，直前に入力された値を用いた予測計算である．(1)式を用いた予測計算は，実際の数値データに対して高い予測精度を得られることが，先行研究により示されている[8]．

予測値を得た後，実際の値との差分を計算する．予測値と実際の値との差分は，浮動小数点演算ではなくビット列を符号なし整数と見立てて計算する．予測値と実際の値の大小関係を調べ，差分は正数として表す．予測値が実際の

値に近い場合，差分のビット列には先頭ビット(MSB)から 0 が多数連続することになる．この 0 はデータの復元に必要ないため圧縮時に削除される．可逆圧縮のため，圧縮後に送信されるデータとして，差分から 0 を除いたビット列(残差ビット)，残差ビットの長さである LRB (length of residual bits)，予測値と実際の値の大小関係を示すためのビット (ex ビット) が必要となる．

図 2 にデータ圧縮アルゴリズムの概要を示す．予測，差分計算を経て得られた 3 つのビット列は，一定長の圧縮データブロック (CDB) に変換されて出力される．圧縮アルゴリズムから，圧縮後の各データはビット長が可変であるため，物理的な帯域を効果的に利用するためにビット長が固定である CDB への変換が必要になる．CDB は圧縮後のデータを順に連結したビット列であり，そのビット長はデバイスの入出力ビット幅に依存する．

データの展開時には，入力される CDB における各圧縮データの LRB にあたるビットを参照し，残差ビットを抽出する．LRB のビット長は一定であるため， LRB の参照とシフト処理により全てのデータが順次抽出される．その後，直前に展開されたデータを用いて圧縮時と同じ予測計算を行う．圧縮時と等しい順序で展開されるため，予測値は展開・圧縮時で一致し，元のデータに完全に復元できる．

2.2 予測精度と圧縮性能

本手法における圧縮性能は，データの種類や数値計算手法に依存するほか，圧縮中でも時間とともに変動する．これは，圧縮後に得られるビット列のうち，残差ビットが可変長となるためである．ex ビットは常に 1 ビット，LRB はデータ形式により異なるが圧縮中は変化しない．残差ビットは差分計算後の有効なビット長であるので，予測値が実際の値に近い程短くなる．よって予測計算が高精度であるほど，圧縮性能は高くなる．

圧縮性能を示す値として，圧縮前のデータ量を圧縮後のデータ量で割った圧縮率を導入する．例として単精度 (32 ビット) の浮動小数点データに対する圧縮率を示す．ex ビットは 1 ビット，LRB は$\log_2 32 = 5$ビットとなる．残差ビット長を r とすると圧縮率 R_{comp} は

$$R_{comp} = \frac{32}{1+5+r} \qquad (2)$$

となる．r が取り得る範囲は 0 から 32 であるが，5 ビットの LRB で表せるように 1 から 32 とする．圧縮後のビット長は 7 から 38 であるため，圧縮率は最大で約 4.57 (32/7)，最小で約 0.84 (32/38) となる．これは 1 データに対する圧縮率であるが，r をデータ全体の平均残差長に置き換えると平均圧縮率が求まる．

予測多項式 (1) はラグランジュ補間における 3 次の補完多項式を変形したものである[8]．この予測では，予測に用いる点と予測される点の値が 3 次関数に従う場合に，予測値と実際の値が完全に一致する．予測に基づく圧縮では，予測精度が圧縮性能に直結するため，一次元だけでなく複数次元の近接点を用いた手法が提案されている[10]．また，複数の予測式を採用し，最適な予測値を動的に選択する手法も提案されている[11]．これらは高い圧縮性能を発揮するが，複雑な処理と巨大なバッファが求められるため，回路面積が増大する．これらに比べ 1 次元多項式による手法は要求される回路面積を大幅に削減できる利点がある．

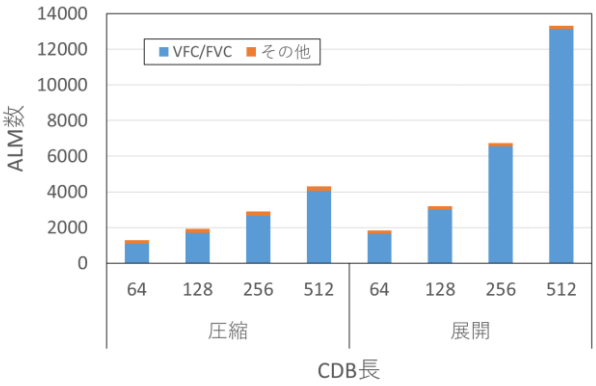

図 3　CDB のビット長に対するハードウェア面積

2.3 回路面積の問題

このデータ圧縮アルゴリズムは，高スループットかつ小さな回路により実装可能であるが，実用的な数値計算への適用にはさらなる回路面積の削減が要求される[12]．実際の数値シミュレーションを専用ハードウェアにより実現する場合，入力されるデータストリームは数値計算における各変数に対応する複数のチャネルに分割され，各チャネルは同期して処理される．本圧縮手法は数値データにおける連続性を利用するため，1 つのチャネルに対して 1 組の圧縮・展開ハードウェアが必要になり，チャネル数の増加に比例してハードウェア量が増加することになる．これに加えて，CDB のビット長もハードウェア使用量に大きく影響することも先行研究において示されている[6]．

図 3 にデータ圧縮・展開ハードウェアの FPGA 実装時に必要となる ALM 数と CDB のビット長との関係を示す．ALM (adaptive logic module) は Altera 社の FPGA における論理リソースで，同社の FPGA である Stratix V シリーズには 30 万前後の数が搭載されている．図 3 から，CDB のビット長の増加に伴い ALM 数も増加することが分かる．32 ビットデータ，CDB が 512 ビットの場合，圧縮・展開合わせて ALM の使用量は約 18000 となる．これは 1 チャネルあたりのリソース量となるため，実際の FPGA における複数チャネル計算への適用には，回路面積が問題となることが明らかである．現在の FPGA への適用に限らず，巨大な回路面積は提案ハードウェアの柔軟な運用を妨げるため，小面積化のための新しい手法が必要である．

3. 予測残差長の量子化

帯域圧縮ハードウェアの実用的な利用に向けて回路面積を削減するため，予測残差長の量子化の提案とそれを用いた圧縮手法について述べる．その中で回路面積が増大する原因と，量子化により面積削減が可能であることを示す．

3.1 回路面積増大の原因

図 3 から，CDB 長の増加に伴う回路面積の増大は，圧縮ハードウェアにおける variable-to-fixed length converter (VFC)，また展開ハードウェアにおける fixed-to-variable length converter (FVC)という 2 つのモジュールの面積の増大によるものだと分かる．これらのモジュールはそれぞれ CDB の生成と分解の処理を行うため，可変ビット長の圧縮データと固定ビット長の CDB との変換を行っているといえる．VFC を例に挙げると，可変長の圧縮データを順に連結して長いビット列を作り，一定の長さを超えた時点で CDB として固定長のビット列を出力する．

CDB は図 4 の(a)のように生成されるが，入力された圧縮データをその時点で連結されているビット長の分だけシフトする必要がある．このシフト量は，個々の圧縮データが可変長のため任意の値を取る．ハードウェアは高スループット化のためにパイプライン化されているため，この任意量のシフトを 1 クロックサイクルで実現するバレルシフタというモジュールが必要になる．w ビットの入力に対し，シフト量が S 通りある場合，バレルシフタが必要とするマルチプレクサ数 N_{mux} は，

$$N_{mux} = w\lceil \log_2 S \rceil \qquad (3)$$

となり，CDB が大きくなるほどシフト量として取り得る値 S も増えるため，回路面積が増大する．回路面積の削減には入力ビット幅を減らす，またはシフト量が取り得る値を減らすことが必要になる．

3.2 予測残差長量子化による回路面積削減の原理

入力ビット数の削減は不可能であるため，S を削減する手法が求められる．残差長の量子化は，シフト量として選択可能な値を減らしてバレルシフタを小さくする手法である．先行研究においては，単精度の浮動小数点データを圧縮する場合に残差長を 4 の倍数に限定する手法が採用されている[5]．この場合，バレルシフタの小型化に加えて，単精度浮動小数点の圧縮において，圧縮後のビット長が常に 4 ビット単位になる利点があった．予測残差長の量子化はこれを一般化して，予測残差長の効果的な選択を可能にした．また，CDB 生成時に可変長の残差ビットと固定長の LRB・ex ビットを別々に扱い，面積削減効果を高めた．

残差長の量子化には，圧縮性能低下を軽減するための残差長分布に応じた値の選択を取り入れる．予測残差長は予測精度が高い程短くなるため，数値データ自体の性質が大きく影響する．例として，2 つの 2 次元流体計算データに対する予測残差の分布を図 5，6 に示す．図 5 は一様流れ内に垂直に平板が置かれた流体計算結果であり，解像度も高い．このような単純な計算では，格子点間における値の変化が小さく，また規則的になるため高い予測精度が得られている．一方，図 6 は気体を圧縮する装置内のシミュレーションであり，曲線格子だけでなく格子点間の距離も一様ではない．また，衝撃波の発生により不連続な点も生じる．このようなデータは，1 次元多項式による予測精度が低下することがグラフから分かる．

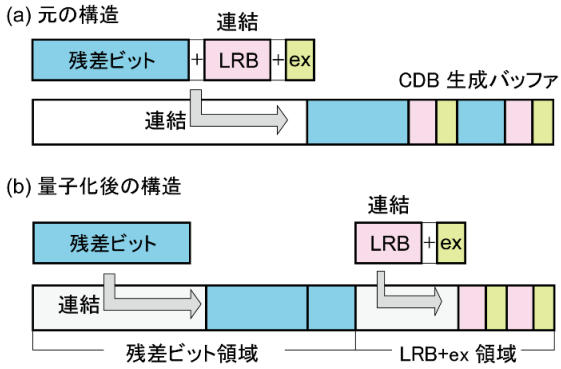

図 4　CDB の構造

一方で，共通点として予測残差長が一部に偏って存在することが見て取れる．このことを利用して，回路面積の削減に伴う圧縮性能の大幅な低下を防ぐことが出来る．残差長がほとんど存在しない部分に対して最適な圧縮を適用しても効果が小さい．残差長を選択することにより，分布が偏る領域に対してのみ効果的な圧縮を適用できれば，圧縮性能の低下を軽減できる．

3.3 量子化手法

予測残差長の量子化において重要な点を以下に示す．

- 残差長が取り得る値を少なくしてシフト量を限定しバレルシフタのサイズを削減する．
- LRB のビット長を削減し圧縮効率を上げる．
- 数値データの予測残差長分布の偏りを利用し，圧縮率の低下を軽減する．
- CDB の構造を改良し量子化に適したものにする．

LRB の長さの観点から，効率的な符号化には量子化後の残差長の種類は 2^n 通りであることが望ましい．残差長についても，連結した圧縮データの取り扱いを容易にするため，2^n 単位で量子化すべきである．また，CDB の構造は図 4 (a) に示すように，残差ビット，LRB，ex ビットをそのまま連結するものから，図 4 (b) のように残差ビット領域と LRB-ex ビット領域に分割した構造とする．これにより LRB-ex ビット領域には，常に固定長のビット列が入力され，残差ビット領域は量子化によりシフト操作が簡略化される．

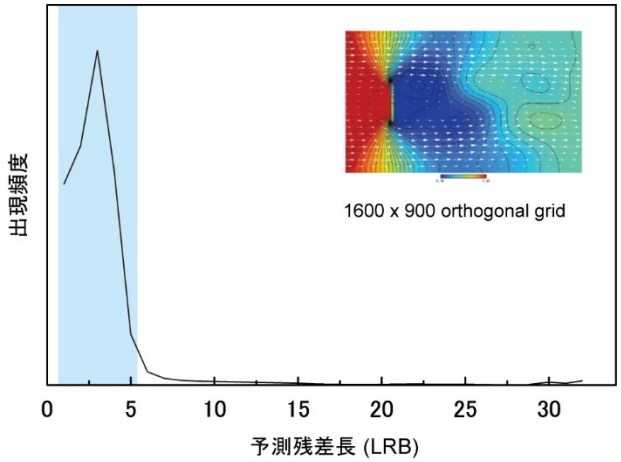

図 5　高精度直交格子による流体データの予測残差長

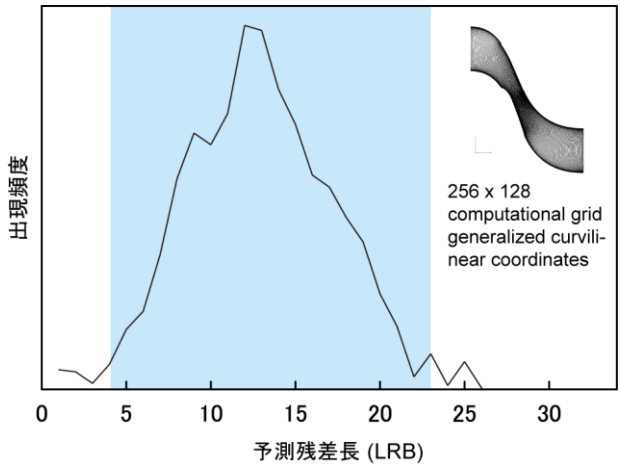

図 6　曲線座標上で生成された計算格子による流体データの予測残差長

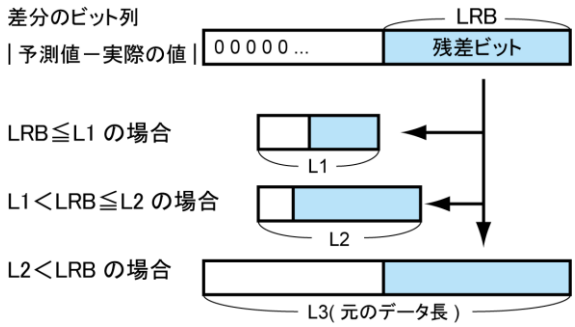

図 7　残差ビットの量子化

次に量子化した場合の符号化について述べる．量子化にはいくつかの残差長を選択し，残差ビットは選択した長さに限定する．LRB は選択した残差長の数によりビット長が決定される．単精度浮動小数点データの場合，元の LRB が 5 ビットなので，十分な面積削減を考慮すると 3 ビット以下，つまり選択する残差長は 8 つ以下となる．ここで，実装したハードウェアの制御上，LRB のうちの一つは終了判定に用いるため，実際に表現可能な数よりも一つ少ない数とする．

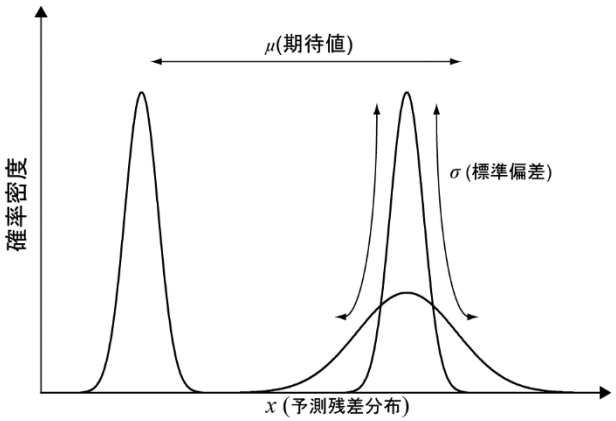

図 8　正規分布による予測残差分布のモデル化

図 7 に量子化により選択する残差長を 3 つとした場合の残差ビットについて示す．この場合，LRB は$\lceil \log_2(3+1) \rceil = 2$ ビットとなる．選択した残差長を小さい方から，L1, L2, L3と表すと，可逆圧縮を維持するために，最も大きな値 L3 は元のデータのビット数と等しくする．図 7 の場合，残差ビットの長さ L_{res} は以下のように決定される．

$$L_{res} = \begin{cases} L1 & (LRB \leq L1) \\ L2 & (L1 < LRB \leq L2) \\ L3 & (L2 < LRB) \end{cases} \quad (4)$$

さらに，値を選択する際に数値データにおける予測残差長の偏りを利用する．図 5，6 が示すように数値計算データは，予測残差長がある領域にまとまって存在する．この領域に限って効果的な圧縮を提供すれば，圧縮性能を向上させることが出来る．対象データの予測残差長分布に応じた選択のため, 次節で評価結果をもとに選択手法を考察する．

量子化の際に選択する値は，2，4，8…のように 2^n を選択することとする．これは，等間隔な量子化では数値データにおける予測残差長の偏りを利用しにくいためである．他の理由として，量子化による面積削減効果は，選択したいくつかの値の最大公約が大きい程高くなることがあげられる．これは，残差ビット領域のシフト量が常に最大公約数の倍数になるためである．

4. 評価

予測残差長の量子化の目的はできるだけ圧縮率を低下させずに回路面積を削減することである．そのため，選択する値を変えて回路面積の削減と圧縮性能の低下についての評価を行う．この結果を用いて，ハードウェアの実装より前に，数値データの性質に基づく最適な量子化の仕様を決定する方法について考察する．

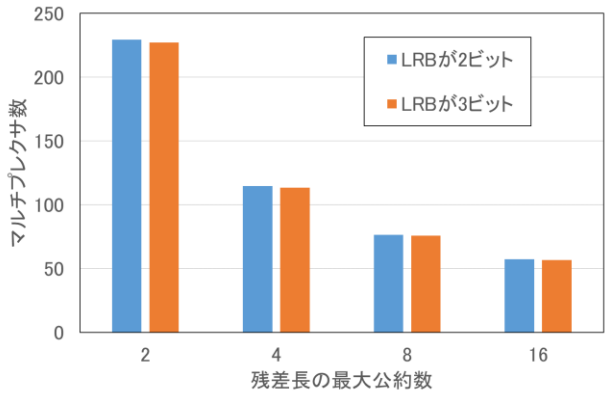

図 9　VFC 内のバレルシフタサイズ

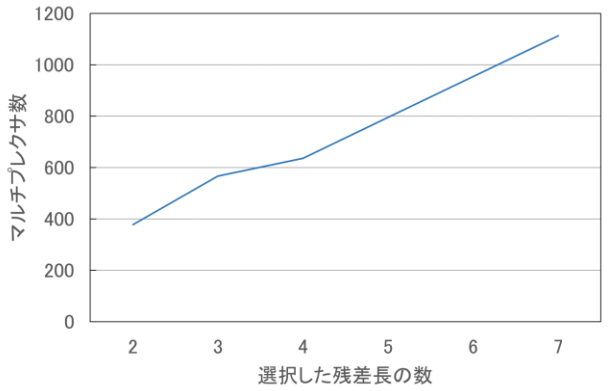

図 10　FVC 内のシフタサイズ

4.1 評価方法とデータモデル

回路面積の評価として，量子化前と量子化後のバレルシフタが使用するマルチプレクサ数の見積もりを行う．さらに FPGA 上に実装し，圧縮・展開ハードウェア全体の回路面積を評価する．

圧縮性能の評価として，量子化の際に選択する値を変えて圧縮率を評価する．圧縮性能は対象データに依存するため，図 5，6 に示した予測残差長分布を正規分布によりモデル化したものを用いて評価する．これを用いてデータに対する圧縮性能の分布を調査する．

圧縮性能評価に用いるのは，図 8 に示すように予測残差長の分布を正規分布としてモデル化したデータである．一般に圧縮性能はデータに依存する，本研究において採用した圧縮アルゴリズムも同様に圧縮性能がデータの性質に依存するため，定性的な評価が難しい．そこで，本アルゴリズムにおいて重要となる，予測残差長の分布(図 5，6)をモデル化して評価を行う．対象とするのはデータそのものではなく，予測残差長の分布を正規分布によりモデル化したものであり，期待値(平均値)と標準偏差の値を変化させた分布により圧縮性能の傾向を調べる．

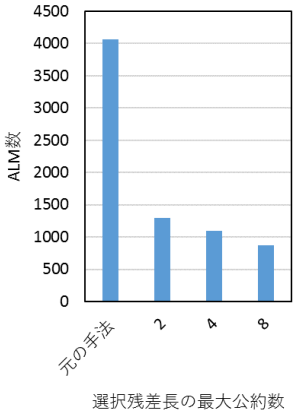

図 11　VFC の回路面積

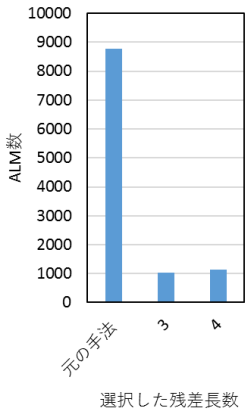

図 12　FVC の回路面積

4.2 回路面積

はじめに，予測残差長の量子化を行った場合の回路面積について，シフタサイズの見積もりと FPGA に実装しての評価とを行う．圧縮・展開ハードウェアの消費リソースの大半は，固定長と可変長の変換を行う VFC と FVC によるものである．特に図 3 が示す通り，展開ハードウェアが巨大になる傾向がみられる．これは，FVC が 2 つのバレルシフタを持つためであり，CDB が大きくなるとその分バレルシフタのシフト量の取り得る値が増加して，(3)式に示すように使用するリソースが増大するためである．VFC も 1 つのバレルシフタを持っており，FVC の場合と同様に面積増大の原因となる．

量子化によるシフタサイズへの影響は，VFC と FVC で異なる．圧縮データを連結して CDB を生成する VFC では，バレルシフタは CDB 内に既に溜まっているデータ分，入力ビットをシフトする必要があるため，量子化によって CDB 内に溜まっているビット数がある値の倍数になり，バレルシフタのサイズが削減される．一方 FVC では，バレルシフタを CDB の入力とバッファの更新との際に使っていたが，量子化と CDB 構造の改良により，入力時のシフトは必要なくなり，更新時のシフトは選択した値の固定シフタにより実現可能になる．よって，VFC のバレルシフタは小さくなり，FVC のバレルシフタは固定幅のシフタに置き換えられる．

このことを考慮して，圧縮・展開ハードウェア内のシフタサイズの見積もりを行った．圧縮対象は 32 ビットの浮動小数点データ，CDB は 512 ビットとして，VFC のバレルシフタと FVC のシフタサイズの見積もりを行った．図 9 に VFC 内のバレルシフタ，図 10 に FVC 内のシフタのマルチプレクサ数の見積もりを示す．VFC のバレルシフタは，選択した残差長の最小公倍数が大きい程面積が減少する．(3)式からわかるように，最小公倍数が大きくなるとシフト量として取り得る値の数が減少し，面積が小さくなる．LRB のビット数を決定する，選択する残差長の数は余り影響し

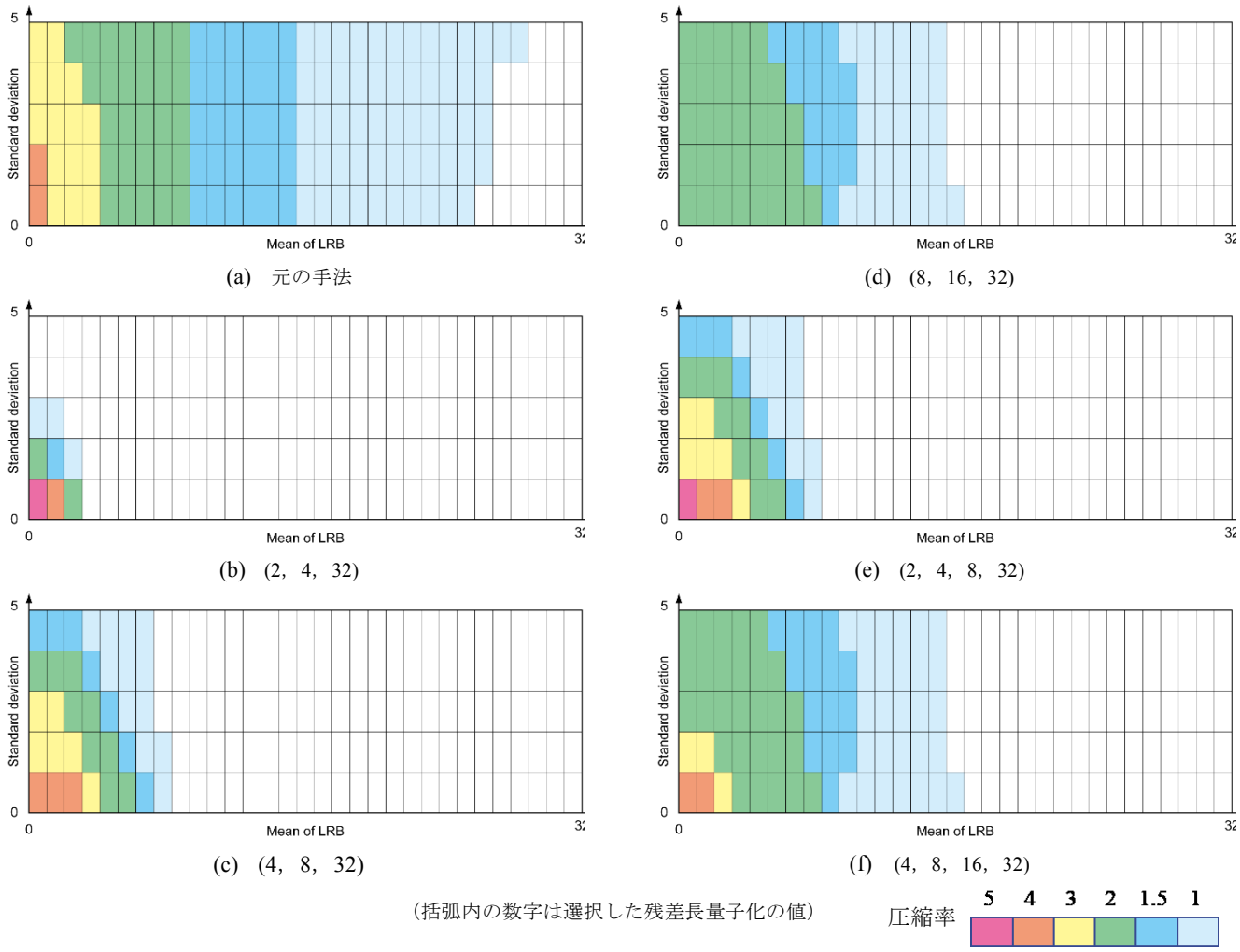

図 13　残差長の正規分布モデルに対する圧縮率分布

ない．一方，FVC からはバレルシフタがなくなっているため，選択した残差長と同じ数の固定シフタが必要になり，その数に比例して面積は増大する．また，残差長数の 3 から 4 の部分からわかるように，LRB の長さが変化すると面積増加の傾きが変化する．以上から，面積を小さくするためには，残差長の最小公倍数を大きくし，選択する残差長を少なくすることが有効である．

量子化による面積削減効果を検証するために，元の手法との面積の比較を行った．Stratix V 上に量子化を用いる手法と元の手法による VFC と FVC のハードウェアを実装し回路面積を測定した．結果を図 11，12 に示す．結果から，見積もりと同様の傾向を示していることが分かる．また，元の手法と比較して大幅に回路面積を削減できることが実証された．

4.3 圧縮性能

次に予測残差長を量子化した際の圧縮性能について評価を行った．図 8 で示した予測残差長分布の正規表現モデルを提案手法により圧縮した場合に，期待値と標準偏差に応じて変化する圧縮性能を 2 次元の分布図として図 13 に示す．各図の縦軸は圧縮した分布の標準偏差，横軸は予測残差長分布の期待値である．上に行くほど分布が広がっており，右に行くほど予測精度が下がることを意味する．各ピクセルは，期待値 1，標準偏差 1 ごとに分布を圧縮した際の圧縮率を示しており，白の部分は圧縮率が 1 以下，その他の部分は凡例に示した数値以上の圧縮率が得られたことを示している．圧縮率は，予測精度が高いほど，つまり残差長が短いほど高くなることが図にも示されている．元の手法 (a) は 1 ビット単位で圧縮を行っているため，広い範囲に対する最適な圧縮によって高い圧縮率を実現している．それに対して (b) から (f) の量子化された手法の圧縮率分布は，効果的な圧縮が可能な範囲が狭くなっている．また，元の手法が標準偏差にあまり関係なく圧縮性能を発揮しているのに対し，量子化した手法では，標準偏差が小さい場合，つまり予測残差長の偏りが強い場合に圧縮性能を発揮していることが分かる．

(b) から (d) は残差長を 3 つに制限したものであるが，これらを比較すると，3 つの値の最大公約数が大きい程，

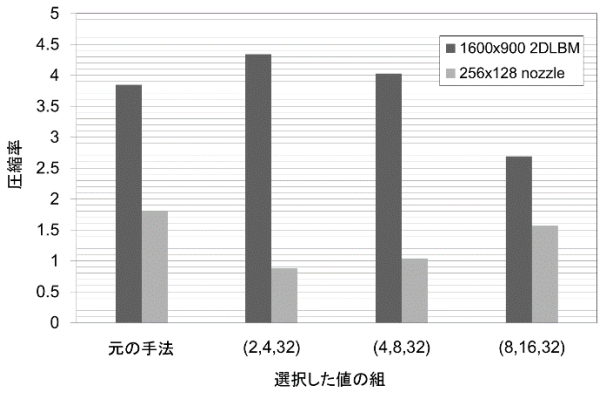

図 14　実際の数値データに対する圧縮率

広い範囲に対して効果的な圧縮が実現できている．一方，最大公約数を小さくすることにより，圧縮が効果的な範囲は狭くなるが，一部分で非常に高い圧縮性能を発揮できることも示されている．(e) と (f) は残差長を 4 つにしたものである．これらは，残差長が 3 つの結果と比べて全体に高い圧縮性能を実現している．回路面積と合わせて考えると，選択する残差長の数が多いほど圧縮性能が向上するが，回路面積も増加することになる．

最後に実際のデータに対する圧縮性能評価を行った．選択する残差長は 3 種類とし，図 13 の(b)，(c)，(d)と同じ値とする．結果を図 14 に示す．用いたデータは図 5，6 に示したデータである．色の濃い方が図 5，薄い方が図 6 のデータの圧縮結果である．

図 5 のデータは非常に予測精度が高く，残差長の偏りも強いため，(2，4，32) のように極端に予測精度の高い部分を効果的に圧縮する選択が適している．この場合，元の手法を上回る圧縮率が得られている．一方，図 6 のデータは予測精度が低く，残差長は広い範囲に分布しているため，分布全体に対して効果的な手法が適している．極端に狭い範囲に効果的な手法では，圧縮率が 1 を切っているが，(8，16，32) の場合約 1.6 とある程度の圧縮率が得られた．

この結果から，量子化の値を選択する際の基準として

- 計算格子の形状や解像度
- データに含まれる非連続要因の度合い

が挙げられる．前者についてこの結果は，計算格子が規則的でかつ解像度が高いほど予測精度が高まり，残差長分布が予測精度の高い領域に集中することを示している，この場合，(2，4，32) のような値の選択が適当である．逆の場合は，(8，16，32) のように全体に効果がある選択が適している．また，基準として示した後者は，例として流体計算における衝撃波等が挙げられ，予測残差長分布のばらつきを増やす要因となる．この傾向が強い場合には (8，16，32) のように広い領域に効果的なものを選ぶべきである．

以上の結果により，要求されるデータ圧縮性能と利用可能なハードウェアリソースの制限という 2 つの観点から，数値計算データに対する圧縮・展開ハードウェアを，データの大凡の性質からあらかじめ選択するための判断材料を提示した．回路面積に余裕があり圧縮性能を重視する場合は，残差長の量子化を行わない手法が適している．一方，回路面積の制限が厳しい場合は，残差長の量子化を適用し，かつ選択する値をできるだけ大きくすることが有効である．さらに，量子化による小面積ハードウェアを採用する場合，図 13 を参考にデータの性質に合わせて値を選択できる．

5. 結論

本稿では，大規模数値シミュレーションにおける大量の数値データにおいて，帯域不足による計算性能低下を防ぐための可逆データ圧縮ハードウェアの性能や汎用性を向上させる予測残差長の量子化について述べた．このハードウェアの実用的な利用において，巨大な回路面積が柔軟な適用を妨げる要因となっていることに対して，数値データ圧縮に用いる予測値との残差を量子化する手法の提案と，一般化し適用範囲を広げる手法を述べた，本手法は，数値データにおける予測残差長分布の偏りを利用して，回路面積を削減しつつ圧縮性能の低下を軽減することが可能である．

圧縮・展開ハードウェアの回路面積の大部分は，可変長と固定長の変換を行う VFC と FVC であり，その原因は任意のシフト量を実現するバレルシフタにあった．そこでバレルシフタのシフト量を制限して回路面積を削減するために，予測値と実際の値の差分として得られる残差ビットの長さを量子化する手法を提案した．これは，残差長を選択して圧縮データの取り得るビット長を限定することにより，ハードウェア操作を簡単化してバレルシフタのサイズを削減することが可能である．

提案した量子化により，回路面積の大幅な削減に成功した．また数値データにおける予測残差長の偏りを利用した圧縮率の大幅な低下を防ぐ残差長の選択を採用した．正規分布モデルを用いた圧縮性能評価により，データの性質に応じた効果的な残差長の選択により，元の手法に近い，あるいはそれを超える圧縮率も実現可能であることが示された．この評価結果をもとに，数値データの特性に合った量子化手法の選択を行うことにより，小面積かつ高圧縮性能なデータ圧縮ハードウェアを実現できることを示した．

謝辞　本研究の一部は、科学研究費特別研究員奨励費 42000987 および挑戦的萌芽研究 23650021 の支援により行われた．

参考文献

[1] Doug Burger, James R. Goodman, and Alain Kagi. Memory bandwidth limitations of future microprocessors. In Proceedings of 23rd Annual International Symposium on Computer Architecture, pages 78-89, May 1996.

[2] Chen Ding and Ken Kennedy. The memory of bandwidth bottleneck and its amelioration by a compiler. In Proceedings of 14th International Symposium on Parallel and Distributed Processing, pages 181-189, 2000.
[3] InfiniBand Trade Association. InfiniBand Architecture Specification: Release 1.0. InfiniBand Trade Association, 2000.
[4] Kazuya Katahira, Kentaro Sano, and Satoru Yamamoto. FPGA-based lossless compressors of floating-point data streams to enhance memory bandwidth. In Proceedings of the International Conference on Application-specific Systems, Architectures and Processors, pages 246-253, July 2010.
[5] Tomohiro Ueno, Yoshiaki Kono, Kentaro Sano, and Satoru Yamamoto. FPGA-based implementation of compact compressor and decompressor of floating-point data-stream for bandwidth reduction. In Proceedings of the 2012 International Conference on Engineering of Reconfigurable Systems and Algorithms (ERSA'12), July 2012.
[6] Tomohiro Ueno, Yoshiaki Kono, Kentaro Sano, and Satoru Yamamoto. Parameterized design and evaluation of bandwidth compressor for floating-point data streams in FPGA-based custom computing. In Proceedings of the International Symposium on Applied Reconfigurable Computing, pages 90-102, March 2013.
[7] Martin Isenburg, Peter Lindstrom, and Jack Snoeyink. Lossless compression of predicted floating-point geometry. Computer-Aided Design, 37(8):869-877, January 2005.
[8] Kentaro Sano, Kazuya Katahira, and Satoru Yamamoto. Segment-parallel predictor for FPGA-based hardware compressor and decompressor of floating-point data streams to enhance memory i/o bandwidth. In Proceedings of the Data Compression Conference, pages 416-425, March 2010.
[9] Bharat Sukhwani, Bulent Abali, Bernard Brezzo, and Sameh Asaad. High-throughput lossless data compression on FPGAs. 2011 IEEE 19th Annual International Symposium on Field-Programmable Custom Computing Machines, pages 113-116, May 2011.
[10] Lawrence Ibarria, Peter Lindstrom, Jarek Rossignac, and Andrzej Szymczak. Out-of-core compression and decompression of large n-dimensional scalar fields. Proceedings of Eurographics, 22(3):343-348, September 2003.
[11] Nathaniel Fout and Kwan-Liu Ma. An adaptive prediction-based approach to lossless compression of floating-point volume data. IEEE Transactions on Visualization and Computer Graphics, 18(12):2295-2304, December 2012.
[12] Igor Ikodinovic, Methodology for Cycle-Accurate DRAM Performance Analysis, IEEE Transactions on Computers, pages 2084-2091, vol.64, no.7, July 2015