マルチメディア通信と分散処理　68－7
グループウェア　9－7
（1995．1．26）

有限体上の画像符号化に関する検討

塩野 智樹　荒木 伸幸　永瀬 宏

金沢工業大学　工学部　情報工学科

画像や音声、ディジタルデータを統合的に扱うマルチメディアの時代を迎え、その中核技術の一つ、情報圧縮技術への関心が高まっている。各種データを圧縮符号化する方法には、圧縮、伸長の前後で情報が保存される可逆符号化と、保存されない非可逆符号化の二つがある。本報告では、有限体の概念を導入し、画像符号化において各画素値を有限体上の数とみなし、原画像と符号化画像の関係を有限体多項式で表し、その多項式の係数を送ることで、情報圧縮を実現する可逆符号化の方法を提案する。

また、有限体演算を用いた、情報の共有と差別化の実現についても述べる。

A Picture Coding by Finite Field Calculation

Tomoki Shiono　Nobuyuki Araki　Hiroshi Nagase

Department of Information and Computer Engineering,
Faculty of Engineering, Kanazawa Institute of Technology

At first, we propose a picture coding method by finite field calculation for data compression. In this method, the relationship between original data and coded data is represented by a function F(x) over a finite field. Encoder sends some of the coefficients of F(x) to decoder. Decoder reconstracts F(x) from those coefficients and reproduces the original picture data.

Next, we propose a method to define common data and own data over a finite field.

In this paper, finite field and its extension is considered, and higher compression coding and definition method of common data and own data are realized.

1 はじめに

これまでは、別々に扱われてきた画像や音声などのデータを融合し、ディジタルデータとして統合的に扱おうとする、マルチメディアへの関心が高まっている。情報圧縮技術は、このマルチメディアを実現するための重要な技術のひとつであり、例えば、ＭＰＥＧ２という規格が挙げられる。この規格は、約１００Ｍｂｐｓの現行テレビ品質の映像を圧縮符号化し、4～９Ｍｂｐｓと、約２０分の１で、実現するものである。[1]

しかし、圧縮されたデータをネットワークで伝送するなどの場合、データ量が少ないほど、より狭い伝送帯域で送ることができ、伝送チャンネル数を増やせるなどのメリットが考えられる。

本報告では、有限体の概念を導入し、画像符号化において各画素値を有限体上の数とみなし、原画像と符号化画像の関係を有限体多項式で表し、その多項式の係数を送ることで、情報圧縮を実現する符号化の方法を提案する。

一般に、符号化は圧縮、伸長の過程で、情報がそのまま保存される可逆符号化と、保存されない非可逆符号化の２通りに分けられる。有限体を用いた符号化は、前者の可逆符号化にあたり、それを既存の圧縮符号化アルゴリズムで生成された圧縮データに適用することで、さらなる圧縮の可能性を検討する。

また、これまでは、必要とするデータが同一のものであっても、複数のコンピュータがそのデータを利用する場合は、それぞれのコンピュータが、そのデータを個別に持っていなければならなかった。しかし、近年では、ネットワーク化が進み、一台のコンピュータが持つデータや画像情報を、ネットワークを介して、他の複数のコンピュータと、共有することが可能になっている。このことは、それまで、それぞれのコンピュータが、全く同じデータを持っていた、という大きな冗長性を取り除いた。しかし、今までは共有がなかったために、データが他に漏れることはなかった。ところが、共有することにより、漏れるべきではないデータが漏れてしまう、という新たな問題が発生した。現在はパスワードを設定することで、それがわからない者には、情報が漏れないなどの対策がとられているが、もしパスワードを知られると、それだけで簡単に情報が漏れてしまい、決して安全な策とは言えない。このような問題を解決するために、より完全な情報の差別化を実現する必要性が出てくる。

本報告では、有限体を用いた符号化と同様に、データを有限体上の値に変換することで、データが持つ性質が変化することを利用し、情報の共有と差別化を実現する方法についても検討する。

2 データ圧縮

2.1 有限体演算を用いたデータの圧縮の原理

有限体上では、真理値表を関数で表すことができるという特長がある。以下ではこの特長を利用し、どのようにデータを圧縮するのか、その原理を具体例を挙げて述べる。以下の内容は文献[2]に詳しいが、説明の都合上、引用させていただく。

表１　真理値表

$GF(2^3) \ni X$ インプット	アウトプット	$Y=F(X) \in GF(2^2)$
０００	００	
１００	１０	
０１０	１０	
００１	１０	
１０１	０１	
１１１	１１	
１１０	０１	
０１１	０１	

表２　$GF(2^6)$の原始元

	０１２３４５６７８９１０・・・６２	
０	１０００００１０００００	１
α	０１０００００１１０００	０
α^2	００１０００００１１００ ・・・	０
α^3	０００１０００００１１０	０
α^4	０００００１０００００１１	０
α^5	００００００１００００１	１
	→ １ → $1+\alpha$	

ここで、表１のような真理値表を考える。

$GF(2^3)$と $GF(2^2)$を含む最小の拡大体は $GF(2^5)$であり、また、$GF(2^6)$上で$\alpha^6+\alpha+1$は既約であるので、$\alpha^6=1+\alpha$によって表２のように、その原始元を特徴づけることができる。

また、$GF(p^n)$の原始元をαとし、$m \mid n$である時、

そしてその時に限って、

・$\beta=\alpha^{V}$

・$V=(p^n-1)/(p^m-1)$

で決まるβが部分体$GF(p^m)\subseteq GF(p^n)$の原始元である。したがって、これから、$GF(2^3)$と$GF(2^2)$の原始元を求めると、それぞれ、

・$\beta^3+\beta^2+1$

・$\gamma^2+\gamma+1$

となる。したがって、インプットとβ^vの、アウトプットとγ^vとの対応が、表3のようになる。

表3　対応表

$GF(2^3)\ni X$ インプット		アウトプット	$Y=F(X)\in GF(2^2)$
$0=\beta^{\infty}$	0 0 0	0 0	$\gamma^{\infty}=0$
$1=\beta^0$	1 0 0	1 0	$\gamma^0=1$
$\alpha^9=\beta^1$	0 1 0	1 0	$\gamma^0=1$
$\alpha^{18}=\beta^2$	0 0 1	1 0	$\gamma^0=1$
$\alpha^{27}=\beta^3$	1 0 1	0 1	$\gamma^1=\alpha^{21}$
$\alpha^{36}=\beta^4$	1 1 1	1 1	$\gamma^2=\alpha^{42}$
$\alpha^{45}=\beta^5$	1 1 0	0 1	$\gamma^1=\alpha^{21}$
$\alpha^{54}=\beta^6$	0 1 1	0 1	$\gamma^1=\alpha^{21}$

表3のような真理値表を表す関数は、一般に次の式で表されることがわかっている。

・$Y=F(X)=a_0+a_1X+\cdots+a_rX^r \quad (r=2^m-1)$

$X\in GF(2^m),\quad Y=GF(2^n)$

・$a_0=F(0)$

・$a_i=\sum_{X\in GF(2^m)} X^{r-i}F(X)$

$a_i=GF(2^l)\ \ (0\leqq i\leqq r)$　(l はの m,n の最小公倍数)

これより、表3の対応を表す関数を求めると、

・$F(X)=\alpha^{43}X+\alpha^{58}X^2+\alpha^{39}X^3+\alpha^{46}X^4+\alpha^{30}X^5+\alpha^{57}X^6$

が得られ、これは、有限体の性質により次のように分割される。

4乗　　4乗

・$\alpha^{43}X \rightarrow \alpha^{46}X^4 \rightarrow \alpha^{58}X^2$

・$\alpha^{39}X^3 \rightarrow \alpha^{30}X^5 \rightarrow \alpha^{57}X^6$

（αの指数は mod 63（$GF(2^6)$）で、X の指数は mod 7（$GF(2^3)$）となっている。）

この周期はフロベニウスサイクルと呼ばれ、このフロベニウスサイクルの初期値$\alpha^{43}X$と$\alpha^{39}X^3$だけが分かれば、あとは4乗することにより、F(X)を求めることができる。一般に$X\in GF(2^3)$から$Y\in GF(2^2)$へのフロベニウスサイクルは、常に、

・$X \rightarrow X^4 \rightarrow X^2$

・$X^3 \rightarrow X^5 \rightarrow X^6$

となるので、結局はXとX^3の係数α^{43}とα^{39}のみを知っていれば、F(X)を求めることができる。

ところで、$GF(2^3)$上の値は2進数3ビット、$GF(2^2)$の値は2進数2ビットの数値として考えることができる。

2.2　有限体上での圧縮符号化法

以上のことを踏まえて、データ圧縮するための基礎として有限体の持つ情報量を先ず考察する。

いま、真理値表の入出力として原始元αのべき乗、α^0、α^1、α^2、α^3…に対応するビットパターン、出力値として画素データを与え、F(X)を求め、X とX^3の係数を符号とすることを考える。この場合、復号側にフロベニウスサイクルのデータを与えておけば、係数をもとにF(X)、真理値表を復元し、もとの画素データを得ることができる。またこのことは、真理値表の入力値をアドレス、出力値をその内容と考えることで、有限体において、アドレスを指定しその内容を取り出すというメモリの機能が実現できることを意味している。

2.1で述べた原理を元に、真理値表からF(X)を求めその係数を符号とする方法を用いた時、真理値表の入力値と出力値のビット長の組み合わせを変えて求めた、符号化前後のデータ量の一部をまとめたものが、表4である。

表4　符号化前後のデータ量

入力ビット数	1	2	2	3	3	3	4
出力ビット数	1	1	2	1	2	3	1
符号化前ビット数	2	4	8	8	16	24	16
符号化後ビット数	2	4	8	8	16	24	18

4	4	4	5	5	5	6	6	・・・
2	3	4	2	3	4	2	3	・・・
32	48	64	64	96	128	128	192	・・・
36	54	64	64	96	128	136	210	・・・

表4から、符号化の前後でデータ量が等しいか、

または符号化後の方が多くなっていることがわかる。すなわち、データを有限体多項式係数に置換える、という方法のみではデータは圧縮されない。多項式のフロベニウス係数は確かに少ないが、係数を表現するために必要とするビット長が長くなってしまうからである。

自然な結論ではあるが、データ圧縮するには原データを有限体表現したときに、統計的な特性が変化することを利用しなければならない。

例えば、可逆符号化法として、ハフマン符号化法が有名であるが、この方法では、各データの頻度という統計量を用いることにより、結果として、データの圧縮を達成している。

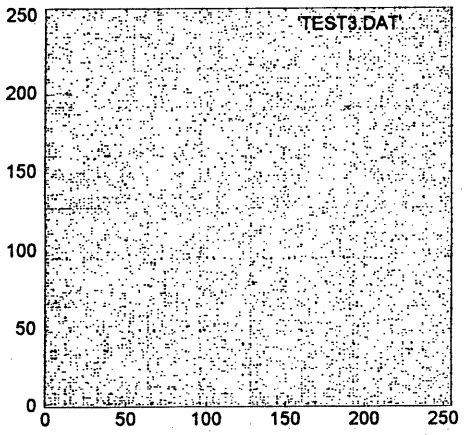


図1　ＭＰＥＧ圧縮データの分布

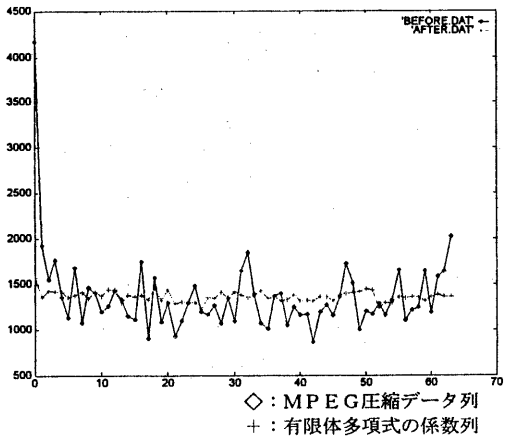


図2　頻度分布

ところで、算術計算により求められた擬似乱数列の相続く要素を座標成分とした点列は、規則的に並ぶ場合があることが知られている。[6]　図１は、ＭＰＥＧ圧縮された画像データ列の相続く８ビットを、座標成分とした点列をプロットしたものである。これらの点列は均等に分布しており、統計量を考慮してもデータの圧縮は困難であるが、データ変換により、このような分布に偏りを持たせることができれば、統計量を用いて圧縮することが可能であると考えられる。

図２は、ＭＰＥＧ圧縮データ列とそれを有限体演算で変換した多項式の係数列を６ビット単位に区切り、それぞれの６ビットデータの頻度を表したものである。この図から、有限体多項式の係数列の分布は均一であり、このままではデータ圧縮は困難である。しかし、ＭＰＥＧ圧縮データ列にはデータの分布に偏りがあり、この偏りをいっそう強調するような符号パターンの変換を行えば、再度、ハフマン符号化のような統計量を考慮した圧縮の可能性が期待できる。この符号パターンは真理値表で表され、したがって２に述べたように有限体上の多項式で計算できる。なお、実際の数値例は、現在検討を行っている段階である。

3　情報の共有と差別化

3.1　有限体を用いた情報の共有と差別化のモデルの提案

これまで有限体上で情報を圧縮することに焦点を当てて述べてきた。しかし、少し視点を変えると、有限体演算により原データが持つ情報はそのままに、その性質だけが変換されたと考えることができる。このことを利用すれば、図３のようなモデルで、情報の共有と差別化を実現することができる。

3.2　部分体と拡大体[4]

図３のモデルで、情報の共有と差別化が実現できることを説明するために、部分体と拡大体の関係について補足する。

ここで、有限体 $GF(2^3)$と $GF(2^2)$、そしてそれらを含む拡大体 $GF(2^6)$を考える。２.１に述べたように、原始既約多項式は

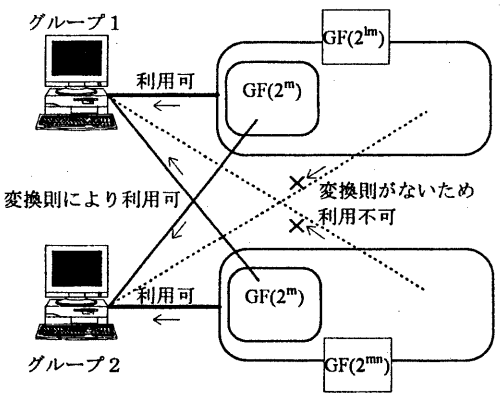
・$\alpha^6+\alpha+1$

であり、原始元αはβ、γと

・$\beta=\alpha^9$

・$\gamma=\alpha^{21}$

の関係で結ばれている。したがって、GF(2^3)の真理値表の要素β^∞、β^0、$\beta^1\cdots\beta^6$は、GF(2^6)ではα^∞、α^0、$\alpha^9\cdots\alpha^{54}$に対応し、GF($2^2$)の真理値表の要素$\gamma^\infty$、$\gamma^0$、$\gamma^1$、$\gamma^2$は、GF($2^6$)では$\alpha^\infty$、$\alpha^0$、$\alpha^{21}$、$\alpha^{42}$に対応している。すなわち、GF($2^6$)上で、要素$\alpha^\infty$、$\alpha^0$、$\alpha^9\cdots\alpha^{54}$間の関係を多項式で表せばGF($2^3$)上の真理値表と、また、要素$\alpha^\infty$、$\alpha^0$、$\alpha^{21}$、$\alpha^{42}$間の関係を多項式で表せばGF($2^2$)上の真理値表と同一の機能を実現することができる。



l,m,n：素数, GF(2^m)はGF(2^{lm})とGF(2^{mn})の部分体

図3　有限体を用いた情報の共有と差別化のモデル

3.3　原理とその評価

以下では、二つのグループで情報の共有と差別化を実現する場合を例に述べる。

u,v,l,m,nを素数とし、u=l×m,v=m×nなる関係が成り立っているとする。ここで、GF(2^u)とGF(2^v)を考える。一方のグループで扱うデータはGF(2^u)の要素で表し、他方で扱うデータはGF(2^v)の要素で表す。これらは共通の部分体GF(2^m)を含む。二つのグループで共有するデータは、この共通の部分体の要素として定義し、一方のグループ内だけで利用し、他のグループからは差別化したいデータは、それぞれGF(2^u)とGF(2^v)の固有の要素として定義する。

二つのグループは共にGF(2^m)の拡大体であるから、例えば、3.2で述べた、$\beta=\alpha^9$や$\gamma=\alpha^{21}$のような変換則を用いることによって、GF(2^m)の要素として定義されたデータを解釈することができ、データの共有が実現される。

二つのグループが、それぞれGF(2^u)とGF(2^v)の固有の要素として定義したデータは、それぞれに変換則が存在しないため、各々のグループ内でしか解釈することができず、データの差別化が実現される。

[例]　二つのグループA、Bを考える。

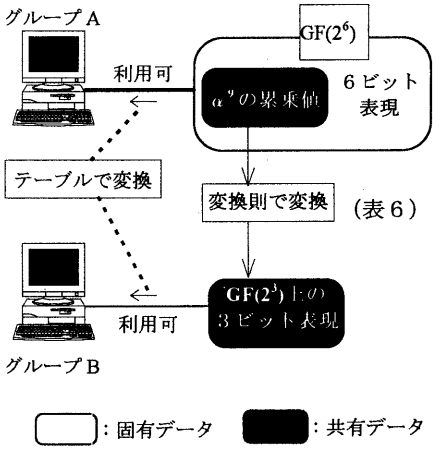


図4　情報の共有と差別化の実現例

表5　変換テーブル

原データ	GF(2^6)上の値	GF(2^3)上の値
data0 (共有データ)	000001 (α^0)	001 (β^0)
data1	000010 (α^1)	
data2	000100 (α^2)	
⋮	⋮	
data9(共有データ)	011000 (α^9)	010 (β^1)
⋮	⋮	
data18(共有データ)	001111 (α^{18})	100 (β^2)
⋮	⋮	
data54(共有データ)	010111 (α^{54})	101 (β^6)
⋮	⋮	
data62	100001 (α^{62})	

表6　変換則

β	=	α^9
↓		↘
010	=	011000

(i)グループAの定義
・固有データと共有データを持つ。
・データはGF(2^6)上（6ビット）で表す。
・表5の変換テーブルを用いて実際のデータと、GF(2^6)表現のデータを変換する。（この時共有データは変換則（$\beta = \alpha^9$）によりα^9の累乗の値として定義する。）
(ii)グループBの定義
・固有データを持たず、グループAの共有データを利用する。
・GF(2^3)上あ（3ビット）で表されたデータを解釈できる。
・表5の変換テーブルを用いて実際のデータと、GF(2^3)表現のデータを変換する。
以上の定義から設計を行なうと、図4が得られる。

我々が提案した方法は、これまでの、例えばパスワードを用いた方法などと組み合わせて用いることができるので、特に、差別化についての完全性は、より高いと言える。

また、パスワードなどの従来の方法は、ソフトウェア的に実現されてきたが、我々の方法は、ハードウェア的にも実現することが可能である。例えば、GF(2^u)上の要素は、それぞれがuビットで表現されるのでハードウェアがuビット単位のデータしか扱えないようにし、グループごとに、扱えるビット長が違うように設計することで可能となる。これで、もし、他のグループから差別化すべきデータが漏れてしまった場合でも、他のグループのハードウェアではuビット単位でデータを扱うことができないので、データを解釈することができないことになり、これにより、完全な差別化が実現できると考えられる。また、データの共有に関しても、変換則を用いて、ビット長を変換する機構を持たせておけば、実現できる。

以上で述べたことは、グループ数が少ない時や、共有と差別化の関係があまり複雑ではない時には容易に実現できるが、実際には、多くのグループ間での複雑な関係を実現することが必要となってくる。また、途中で、新たなグループが加わり、グループ数が増えたり、または、逆に減ったりすることもある。我々が、提案した方法の問題点として次のようなことが考えられる。

(i)グループの数が多い場合や、様々なグループとの間に複雑な共有、差別化の関係がある場合、グループごとに割り当てる有限体の大きさが必要以上に大きくなってしまい、ハードウェア化が効率よく行えない。

(ii)新たなグループが後から加わり、新たな情報の共有、差別化の関係を追加する場合、グループごとに割り当てられていた固有の拡大体を、すべて割り当てなおさなければならない可能性があるとともに、ハードウェアを設計し直さなければならない。

これらの問題は今回提案した方法の実用性を考えると重大なものである。差別化の完全性を犠牲にするのなら、ソフトウェア的に実現することによって、ハードウェア的な問題は解決できるが、今後、これらの問題をどう解決するかが、大きな課題である。

4 あとがき

本報告では、有限体上で、画像情報の圧縮符号化法についての検討を行った。今回の検討では、情報圧縮を達成することはできなかったが、データ列の分布の偏りを強調するような符号パターンの変換を行えば、再度、統計量を考慮した圧縮の可能性が期待でき、圧縮符号化が実現できると考えている。

また本報告では、有限体演算により、情報の共有と差別化の実現が原理的に可能であることを述べた。この方法は、ハードウェア的に実現することが可能で、特に、情報の差別化について、従来の方法よりも完全性が高いものとなっている。しかしながら、まだ問題点もあり実用的なものとは言えず、これを改善する方法の検討が必要である。

参考文献

[1] 藤原洋：最新ＭＰＥＧ教科書,アスキー出版,１９９４
[2] 高橋磐郎：組合せ理論とその応用,岩波書店,１９７９
[3] 永瀬宏,喜田泰弘,田中裕範：有限体演算を用いた計算機の検討,第１５回情報理論とその応用シンポジウム,１９９２
[4] 永瀬宏,田中裕範,喜田泰弘：有限体上の演算を用いた階層的アクセス制御,電子情報通信学会,１９９２
[5] 伊東利哉,佐古和恵：有限体上のアルゴリズムと多倍長・剰余演算の高速演算法,情報処理学会,１９９３
[6] 柏木潤,M系列を用いる乱数発生の研究,昭和５９・６０年度科学研究費補助金総合研究成果報告書,１９８６