DBSJ Journal, Vol.11, No.1
June 2012

Universal SAX: 空間充填曲線を利用したSAXの多次元時系列データへの適用

Universal SAX: Applied SAX to the Multidimensional Time Series Data Using the Space Filling Curve

大西 史花† 渡辺知恵美‡

Ayaka ONISHI Chiemi WATANABE

大規模データや多次元時系列データを効率的に取り扱うためには索引の作成が不可欠であり，今まで多くの研究者が様々な種類の索引を提案してきた．その中に，Lin らが提案した，時系列状のデータを文字列に変換する Symbolic Aggregate Approximation (SAX) という手法がある．この手法を介することにより実数値データに対して優秀な自然言語アルゴリズムを適用出来るようになり，マイニングに役立てることが可能となった．しかし多次元時系列データに対して SAX を適用して索引を作成する既存研究は，そのほとんどが次元を削減するなどの処理を行うため元データの情報量を欠落させている．
よって本稿では元データの情報量を欠落させない多次元時系列データのための索引である Universal SAX を提案する．本手法は SAX を多次元時系列データに対して適用し，元データの情報を保持したまま文字列に変換することで索引を作成する．

In order to effectively deal with multidimensional time series data or large scale data, index is essential. So many researchers have proposed various kinds of indexes. One of the techniques is Symbolic Aggregate Approximation (SAX) by Lin et al. SAX transform time series data into a symbolic representation. User can apply useful approaches which are only defined for symbolic representation by SAX to time series. However Existing researches which create an index after apply SAX to multidimensional time series data would be missing a part of amount of information of the original data because most approaches execute operation which is reduced the dimensionality. In this paper the authors propose a Universal SAX. This approach create an index after apply SAX to multidimensional time series data, and transform time series data into a string while retaining the information of the original data using space filling curve.

1. はじめに

大規模データベースにおけるデータの取り扱いにはデータ量の削減や処理時間短縮の面から索引の作成が不可欠であり，今まで多くの研究者が様々な種類の索引を提案してきた．取り扱う対象データの種類も，1次元のデータ（文字列など）のみではなく多次元データ（時系列データや画像データなど）を取り扱う場面が増えてきている．

我々は先行研究において，Linらが提案した時系列データを符号化する手法であるSymbolic Aggregate Approximation (SAX) [6] をスマートハウス内に設置されたセンサの推移値に対して適用し，住宅内の出来事（イベント）を検出するための索引の検討を行った[9]．この論文では，住宅内のイベントを効率よく検出する索引の作成を目指し，まず対象データとしてx y z 軸からなる3次元データであるセンサの推移値を取り扱った．その3次元センサデータに対してSAXを適用するために，特に変化が大きい軸であったx軸1つのみを選び，SAXを使用して各センサ値を時間軸で等しく区切りその区間ごとに平均値に置き換えたあと，1次元のデータである文字列に変換しその文字列を索引として利用した．しかしこの手法では，まず本来3次元であるデータからx軸のみを使用している点と次元を落として索引を作成している点から，元データの情報量に対して情報欠落が生じている．

また，画像データ等の多次元データを扱いやすくするために他表現に変換するような手法もこれまでに多く提案されてきているが[3][10]，それらの手法は，画像内の形状（2次元）をその重心から境界線までの距離を使用して時系列データ状（1次元）に変換するなど，そのほとんどが変換の過程において次元を削減している．扱うデータによってはその欠落した情報内に有用な情報が入っているという可能性は捨てきれない．

よって本稿では，多次元データの空間的距離関係を出来る限り保ったままSAXを適用するUniversal SAX (U-SAX) を提案する．この手法で我々は空間充填曲線の一種であるヒルベルト曲線を用いている．空間充填曲線とは多次元空間を1次元空間に写像する手法の1つで，この手法を用いることにより多次元空間内の近接した領域を符号に割り当ててSAX索引を作成することが出来る．ヒルベルト曲線による1次元化はデータの分布に依存しないためデータの前処理などをほどこす必要が無い．よって多次元時系列データであれば一般にどのようなデータでも適用できる．もちろんSAXで符号化することによって[6]で示されているように類似検索，例外検出，モチーフ検出など各種マイニング操作を多次元時系列データに対して効果的に行うことが出来る．

本稿の構成は以下の通りである．第2節では本研究で使用するメイン手法であるSAXについて述べ，第3節では多次元データへのSAX適用に関する既存研究について述べ，第4節では今回新たに提案するUniversal SAX索引について述べ，第5節では提案手法の評価実験について述べる．そして，第6節にてまとめと今後の課題を述べる．

2. SAX

SAX とは，時系列データの表現手法の 1 つで，時系列データを文字列に量子化する手法である．パラメータ値として，あらかじめ文字列長 w（もしくは 1 つの文字に変換するデータ数）と文字種類 a の 2 つを決定する．これらの値はデータに依存するため実験的に求める必要がある．

SAX による時系列データの文字列化の手順を以下と図 1 によって示す．図 1 では破線が SAX を適用するセンサデータである．

† 学生会員 お茶の水女子大学大学院人間文化研究科博士前期課程 onishi.ayaka@ is.ocha.ac.jp
‡ 正会員 お茶の水女子大学理学部情報科学科 chiemi@ is.ocha.ac.jp

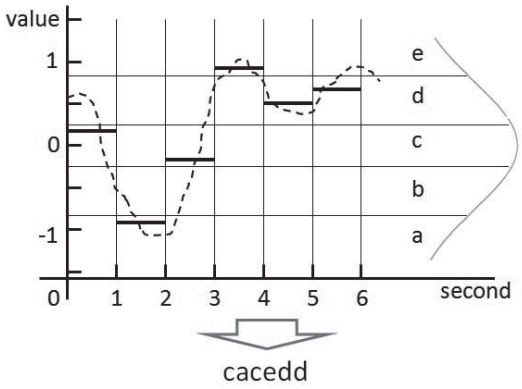


図 1: SAX による時系列データの文字列化

(1) 時間軸を等間隔に区分（縦実線）.
(2) 区間ごとの平均値を算出（横太線）.
(3) 正規分布に従って， a ，b ， c …とアルファベットを割り振り，正規分布の各面積が等しくなるような分割線（横実線）を定める.
(4) (2) で求めた平均値を(3) で定めた分割線に従って文字に変換（下部）.

SAX を適用して作成された文字列には性能の良い各種マイニング操作（類似検索，分類，モチーフ検出，特徴抽出など）を行うことが可能である．これは SAX の近似距離が実距離の下界を抑えることを保障しているということが大きい. 例えば 2 つの時系列データ C と Q の 2 つを比較する際，距離尺度としては一般的にユークリッド距離（図 2 の(A)）等を使用することが多い. それに対して SAX での文字列 $(\hat{Q}, \hat{C})$ 間の距離は図 2 の式 (B) で示されるように文字間の距離 $dist(\hat{q}, \hat{c})$ の二乗の和で定義されている．n は元データのデータ数である.

$$(A)\quad \text{元データ}\quad D(Q,C)\equiv\sqrt{\sum_{i=1}^{n}(q_i-c_i)^2}$$

$$(B)\quad \text{SAX表現}\quad MINDIST(\hat{Q},\hat{C})\equiv\sqrt{\frac{n}{w}}\sqrt{\sum_{i=1}^{w}(dist(\hat{q}_i,\hat{c}_i))^2}$$

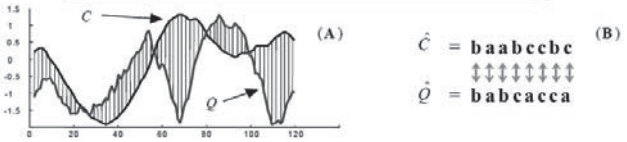
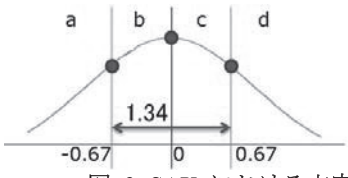


図 2: SAX における文字列間の距離の定義と概念図

文字間の距離 $dist(\hat{q}, \hat{c})$ は文字 $\hat{q}$ がとりうる値の最大値と $\hat{c}$ がとりうる値の最小値によって定義される．例えば図 3 の左側は時系列のデータの値の分布を示している. SAX 索引は量子化の前に値の正規化を行うため，正規分布に従っている．これはデータの離散化を行う際その符号の出現確率は等しいことが望ましいからである [1][8]．この図では時系列の値を 4 種類のアルファベットに分割している．ここで dist(a,d) は a のとりうる最大値 -0.67 と d のとりうる最小値 0.67 の差となるので 1.34 となる．このような距離の定義により，SAX によって求められる文字列間の距離（つまり元のデータの近似距離）は元のデータの実距離の下界を抑えていることが保障されている.

ここで, SAX における文字間の距離は入力データに依存せず，領域の分割数（符号の数）によってのみ決まることに注意したい．そのため，SAX は文字間の距離の表（図 3 の例の場合は図中右の表）をあらかじめ計算しておき，距離表を保存すれば 2 つの時系列，つまり 2 つの SAX 文字列の各文字間の距離は距離表を参照するだけで求めることが出来る.



	a	b	c	d
a	0	0	0.67	1.34
b	0	0	0	0.67
c	0.67	0	0	0
d	1.34	0.67	0	0

図 3: SAX における文字間の距離の図示と表

以上から分かるようにSAXの利点として，時系列間の距離の下界を抑えることが保障されていること，実装が容易であること，文字列用に定義された手法等も適用出来るようになること等が挙げられる.

3. 多次元データへの SAX 適用に関する既存研究

今回我々は多次元時系列データに対して次元を削減しないまま SAX を適用した Universal SAX 索引を作成する手法を提案する．関連研究として，同様に多次元のデータに対して SAX を適用し索引を作成した Li ら[7]や田中ら[12]の研究がある.

[7]では多次元データに対して重心距離法を用いた後 SAX を適用している．重心距離法とは形状の情報をその重心から形状境界までの距離として 1 次元表現する手法である. Li らはこの手法によって得られた 1 次元表現に対して SAX を適用することにより，大規模画像データセットから例外画像を動的に検出する手法を提案した.

[12]では多次元データに対して主成分分析を用いた後 SAX を適用している. 主成分分析とは複数の変数から成るデータの相関を少数の合成変数で表現する手法である．この手法で表現される合成変数は必要に応じて次元数を調節することが出来る．また第一主成分に一番多くの情報量が含まれるという性質上，扱うデータによっては第一主成分のみを使用すれば十分で，多くのデータ量を削減することが出来る．田中らはこの手法を用いてモーションセンサから得た 3 次元センサデータを 1 次元表現に変換し，そのデータに対して SAX を適用することにより動的に未知頻出パターンを検出する手法を提案した.

これらの手法ではどちらも多次元データを SAX に適用するために次元削減を行っていることに注意したい．更に，これらの手法は対象データの特徴や使用用途によっては次元削減を行っていても有効に機能することが各論文にて示されているが，次元削減を行うことによって情報が著しく欠落する場合のデータも存在する.

まず重心距離法についてであるが，この手法は前述のように重心からの境界までの距離を使用して時系列状データに変換している．したがって，元データ上で等距離に存在する異なる場所の値は全て同じ情報として扱われるという特徴を持つことになる．この特徴は，回転や移動を許す検索の場

合は問題にならないが，軌跡情報など絶対的な位置を必要とするデータの場合では情報を著しく欠落させることになる．例えば球面状を推移するデータの挙動があるとする．このデータに対して重心距離法を適用した結果の時系列は，重心からの距離が全て等しいため直線になる．このような直線データには明らかに軌跡情報が含まれていないので，軌跡情報検索を行うことは現実的でない．

また主成分分析については，分散の大きいデータに対して第一主成分のみを使用することは，情報欠落量が膨大になってしまうため適切ではなく，検索精度に問題が生じる場合が考えられる．

4. 多次元データのための Universal SAX 索引の提案

この節では, SAX を使用した多次元時系列データのための Universal SAX 索引について述べる．簡単のために，まず多次元時系列データとして 2 次元データを考える．

本手法では多次元時系列データの情報を維持したまま SAX を適用するために空間充填曲線の 1 つであるヒルベルト曲線を用いている．よって 4.1 節にて空間充填曲線について，4.2 節にて Universal SAX 索引の作成手順について述べる．

4.1 空間充填曲線

空間充填曲線とは多次元空間を曲線で埋め尽くしていくような曲線のことで，多次元空間を 1 次元空間に写像する手法の一つとして使用される．空間充填曲線にはヒルベルト曲線[4] （図 4 左図）や Z-ordering[5] （図 4 右図）などがある．これらの手法を用いることにより，多次元空間上で距離の近いデータは 1 次元空間写像後も比較的近い位置で表されるため，多次元データの情報を維持したまま SAX を適用することが出来る．

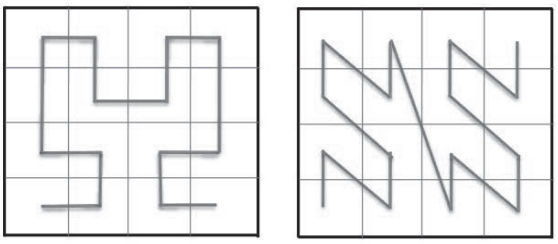


図 4: ヒルベルト曲線と Z-ordering

実際に 2 次元空間のデータを 1 次元空間に写像する手順を，ヒルベルト曲線を例に示す．以後，説明のためヒルベルト曲線によって 1 次元化されたデータをヒルベルトコードと呼ぶ．

(1) 多次元空間のそれぞれの軸の値を $2^k \times 2^k$ の格子状になるよう量子化し，ヒルベルト曲線を描く．
(2) ヒルベルト曲線の開始ピクセルから，曲線上を辿りながら(1)で決定した量子化線を越えるたびに番号を 1 加算し，各ピクセルに数値を割り振る．
(3) ヒルベルトコード化したい対象のピクセル を(2)で割り振った数値に変換する．

ヒルベルトコードの生成例を，図 5 に示す．まず，この 2 次元空間の X 軸と Y 軸を $k = 2$ で量子化し（橙色線）ヒルベルト曲線を描く（緑線）．次に，ヒルベルト曲線の開始ピクセルである左下から順に曲線を辿りながら，各ピクセルに数値を割り振る．最後に，ヒルベルトコード化したいピクセル（ここでは a_0, a_1, a_2, a_3 とする．）を先ほど割り振った数値に変換する（図 5 右図）．

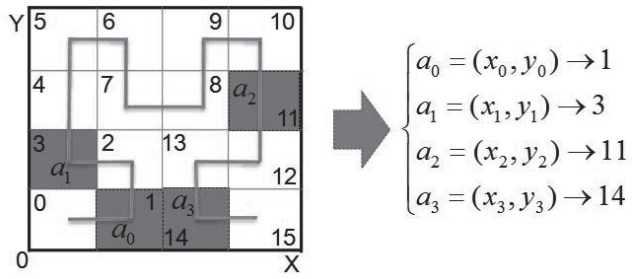


図 5: ヒルベルトコードへの変換手順の図示

このような手順によって変換されたヒルベルコードでは，図 5 の a_0 と a_1，a_0 と a_2 のように多次元空間上で距離の近いデータほどヒルベルトコード変換後の数値間の差も小さくなることが分かる．しかし稀に a_0 とa_3 のように多次元空間上での距離は近いのにヒルベルトコード変換後の数値間の差は大きくなる場合もある．このような場合は Z-ordering でも発生し，加えて Z-ordering ではその曲線の描き方故にピクセル間のジャンプが発生するためヒルベルト曲線よりもその頻度は高くなる．その点ヒルベルト曲線はジャンプが無いため Z-ordering よりも良い距離特性を持っている．

以上のような空間充填曲線を用いた多次元空間は，範囲検索や k-近傍検索や空間結合など空間問合せの手法が確立されている[3].

4.2 多次元時系列データのための Universal SAX 索引の作成手順

今回提案する多次元データに対して SAX を適用し Universal SAX 索引を作成する手順について，2 次元データの Universal SAX 作成手順を例として以下と図 6 によって示す．

(1) 対象データをそれぞれの次元ごとで標準正規分布に正規化し，対象データ（黒実線）を $t_0, t_1, \ldots, t_n$ まで等間隔に区分し，区間ごとの中央値$x_0, x_1, \ldots, x_{n-1}$（黒点）を決定する．
(2) 多次元空間を $2^k \times 2^k$ の格子状になるよう量子化（橙色線）する．ここでの k はヒルベルト空間の解像度（resolution）となる．
(3) 空間充填曲線の一つであるヒルベルト曲線を用いて（緑色点線）(2)で正規化された対象データをヒルベルトコード列 (51,46,40,38,27,28, 10,17,15,1) に変換し，時系列状のデータとして表す．
(4) 各次元で行った正規化の領域が等しくなるように分割を行い，それぞれの領域にアルファベットを割り振り（図 6 の(4)左図）文字列に変換する（図 6 の(4)左図下）．

(4)の右図は正規分布を分割した後の多次元空間のイメージである．

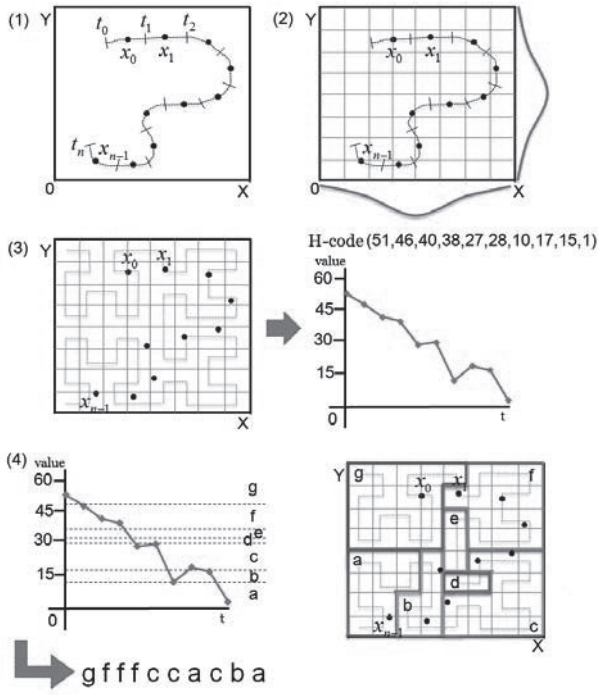


図 6: 2 次元データの 2 次元 Universal SAX 索引の作成手順

本提案手法では手順(2)において対象データをそれぞれの次元ごとに標準正規分布に正規化しているため，従来の SAX の手法とは符号の存在確率の等面積分割法が異なる．例えば図 6 のような 2 次元データに対してそれぞれの次元ごとに正規化を行うと存在確率は空間の中心に近付くにつれ高くなる（図 7 上図）．このような空間においてヒルベルト曲線を適用すると，符号の存在確率分布は 2 次元データの場合だと三峰分布となる（図 7 下図）．よって本提案手法では手順(4)においてこの三峰分布を等面積に分割する必要がある．そのために，我々は乱数を用いたシミュレーションを複数回行うことにより近似解を求める計算手法であるモンテカルロ法を用いて三峰分布の等面積分割を行った．

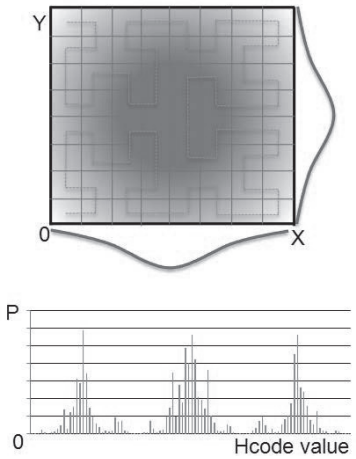


図 7: Universal SAX における符号の存在確率とヒルベルトコード上での符号の存在確率の図示

本手法を用いて 2 次元空間を分割した例を図 8 の左に示す．ここでは手順(2)で述べたヒルベルト曲線解像度 k を 7，符号の種類 a を 49 としている．k=7 の場合，各軸のセル分割数は 2^7=128 となり，つまり各次元は 128 分割で量子化されている．この図から，符号の存在確率が高い中央付近は分割数が細かくなっていることが分かる．しかし反対に，存在確率の低い外側の部分の領域が大きくなってしまい，例えば緑の領域の左上にある黒点と赤の領域にある右下の黒点は実際には距離が離れているにもかかわらず領域同士は隣接しているためこのままでは距離がゼロになってしまう．よって我々はこのような状況を回避するために新たなパラメータとしてブロック解像度（Block resolution, Br）を導入する．これは領域が一定範囲以上の大きさになれば強制的に分割をおこなうしきい値で，これによって外側の領域が大きくなり過ぎることによる実際の距離との差を少なくすることが出来る．図 8 の右は，k=7，a=49，Br＝5（2^5）とした場合の 2 次元空間の例である．

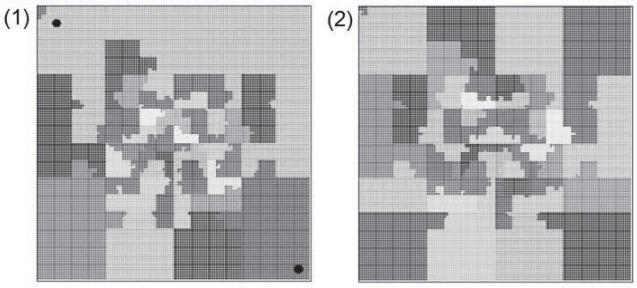
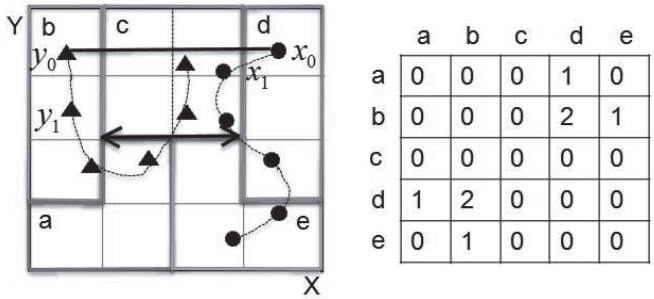


図 8: (1)Br 導入前の 2 次元空間の分割例，
(2)Br 導入後の 2 次元空間の分割例

Universal SAX 文字列の文字間の距離は，SAX の手法にならい順序が先の文字がとりうる値の最大値と順序が後の文字がとりうる値の最小値によって定義することにする．例えば，図 9 左図において元のデータ内の点 x_0,y_0 との間の距離 $dist(x_0,y_0)$ の符号割り当て後の距離は $dist'("d","b")$ となり，この時の距離は空間区分の 1 マスの距離を 1 とすると 2 となる．また x_1,y_1 との間の距離 $dist(x_1,y_1)$の符号割り当て後の距離は $dist'("c","b")$ となり，この時の距離は 0 となる．なお，元データ間の距離と文字列変換後の距離は，SAX の最大の特徴である距離の下界を抑えるために常に $dist(x_0,y_0) \geqq dist'("d","b")$ となることに注意する．

このような文字間の距離の値は，図 9 の右図のように SAX の手法と同じく距離の表としてあらかじめ用意することが可能であるため，この値を毎回計算する必要はない．



	a	b	c	d	e
a	0	0	0	1	0
b	0	0	0	2	1
c	0	0	0	0	0
d	1	2	0	0	0
e	0	1	0	0	0

図 9: 元データ間の距離と文字列変換後の距離の図示

このように作成される Universal SAX 索引には以下のような利点がある．

1. 多次元空間上での距離が保たれた 1 次元表現の符号索引が作成できる．
2. ヒルベルト曲線を使用することにより空間区分間のジャンプが起こらない．
3. 多次元ベクトル間の距離を求める処理時間を大幅に削減することが出来る．

1 については，空間充填曲線の多次元空間上での距離を保ったまま 1 次元表現するという特徴を生かし，加えて SAX を適用し文字列表現に変換することで，文字列に対する多様で有用なアルゴリズムやデータ構造を利用することが出来るようになるということで，これはこの索引の最大の利点である．

2 については，本手法において Z-ordering ではなくヒルベルト曲線を採用した理由でもある．Z-ordering を使用すると，今回の提案手順の(4)において領域の分割を行った場合に，図 10 のように同じ区分として分割が行われたが区分同士が離れてしまうということがある（図 10 の青，橙，赤の区分領域）．このように領域が分割された場合，例えば図中の 2 つの黒丸間の距離は同じ領域内の点同士であるので 0 となり，黒丸と黒三角間の距離は 1 となる．しかし実際の空間上の距離は黒丸同士よりも黒三角との間の距離の方が近く，最適な区分分割が出来ているとは言えない．これに対してヒルベルト曲線は元々の曲線の描き方から領域のジャンプが存在し得ないため今回の手法にとっては都合が良い．よって本手法では計算の容易な Z-ordering でなく空間ジャンプのないヒルベルト曲線を用いている．

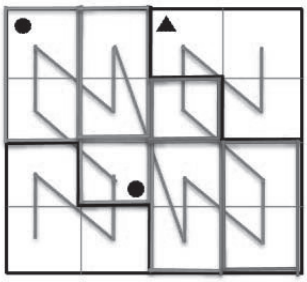


図 10: Z-ordering によって分割された多次元空間の例

3 については，SAX における文字間の距離があらかじめ表として用意することが出来るということに起因する．通常多次元データ間の距離を測る場合，1 点について次元ごとに距離を測定するという計算を毎回行う必要があり，次元数が多くなれば更に計算量は増大してしまう．しかし SAX の場合は距離の表から値を参照してくるだけで良いので次元数には全く依存しない．よって我々の提案手法では計算処理時間を大幅に削減することが出来る．

5. 評価実験

我々は提案手法を適用した場合のデータ量の比較実験と、提案手法を用いたクラスタリング実験を行った．実験に使用したデータは UCI[11]の Machine Learning Repository 中の Character Trajectories Data Set である．このデータは 2858 個の実際の筆跡データセットで，ペンタブレットによる入力を軌跡と筆圧からなる 3 次元のデータとして取得したものである．当レポジトリに関する各時系列データの説明が見つからなかったため，我々はこのデータセット内から視覚的に類似していると思われる 6 グループ（1 グループに 3 個のデータ）を抽出し，x 軸と y 軸を使用した 2 次元データを対象に実験を行った．図 11 は実験に使用したデータの一部を視覚化したものである．{data1, data2, data3}, {data11,data12, data13}, {data21, data22, data23}, {data101, data102, data103}, {data171, data172, data173}, {data226, data227, data228} がそれぞれ視覚的に類似したものと判定した．比較対象として使用したデータは、データ量比較実験については実数値である元のデータと，各次元それぞれに SAX を適用した文字列データ（説明上 MdSAX とする．），MdSAX と USAX に文字列圧縮を施したもの（今回はランレングス圧縮を使用）の 5 つである．クラスタリング実験については元データと USAX の 2 つを使用している．

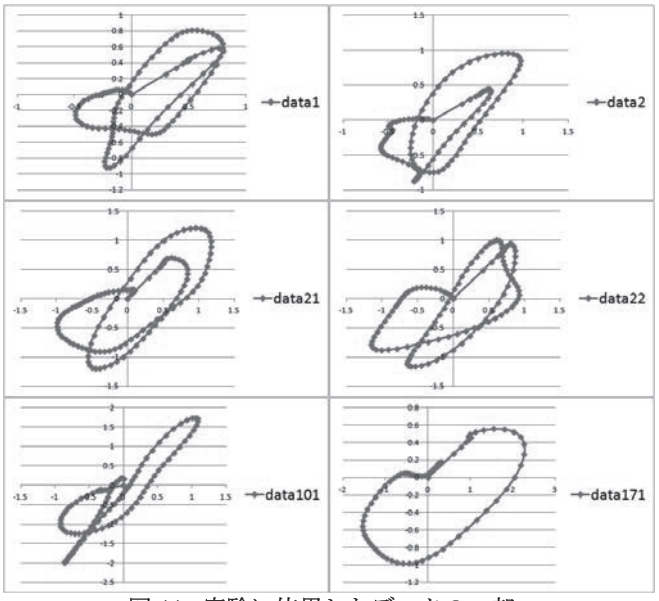


図 11: 実験に使用したデータの一部

まずデータ量比較実験の結果を図 12 に示す。SAX を適用したどの手法も、元データより非常に少ないデータ量で表現できていることが分かる．特に提案手法である Universal SAX は元データの約 1/16、MdSAX の約 1/2 のデータ量となっており、さらにテキスト圧縮を施した圧縮 Universal SAX は元データの約 1/80 のデータ量となった．この結果から、提案手法 Universal SAX は非常に少ないデータ量で多次元データを表現できるという結果を得ることが出来た．

クラスタリング実験の結果は図 13 の通りである．図の上部は元データに対してクラスタリングを行った結果、図の下部は Universal SAX を適用した文字列データ（符号数= 53）である．赤線で囲ったデータは正解したと判断した部分である．結果を見てわかるように，Universal SAX は元データの結果とほぼ同等の精度でクラスタリングを行うことができている．また、元データではうまくクラスタリングすることが出来なかった{data101, data102, data103}を Universal SAX ではクラスタリングすることができていることもわか

る．このように，Universal SAX は元データと遜色ないマイニングを行うことができるといえる．

	元データ	MdSAX	USAX	圧縮 MdSAX	圧縮 USAX
データ量(KB)	7946.14	1017.17	501.76	219.84	93.01

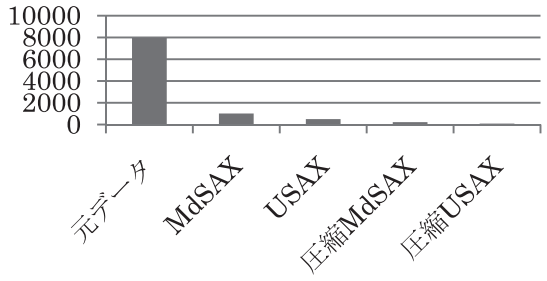


図 12: データ量の比較結果

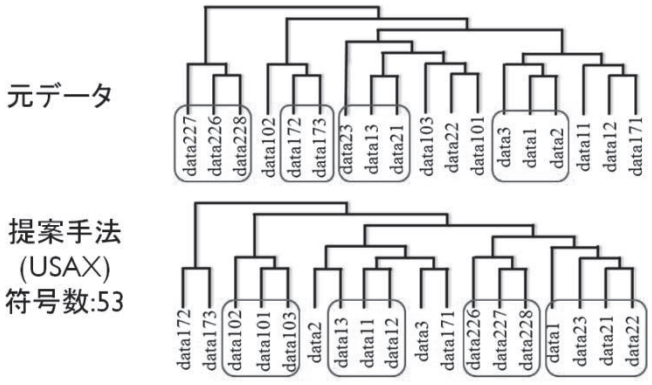


図 13: 評価実験での階層型クラスタリングの結果

6. まとめと今後の課題

本稿では，多次元時系列データに対して SAX を適用した Universal SAX 索引を提案した．この手法では，多次元時系列データに対して空間充填曲線のひとつであるヒルベルト曲線を適用し，空間的距離関係を出来る限り保ったまま SAX により符号に変換する．評価実験の結果，提案手法である Universal SAX は多次元時系列データのマイニングに有効な索引であることを示した．

今後は，多次元時系列データに対する既存手法との比較実験や，提案手法の改良を行っていく．

[謝辞]

本研究を進めるにあたり，ご助言いただいたお茶の水女子大学の伊藤貴之教授に深く感謝いたします．

[文献]

[1] Apostolico A, Bock M. E, and Lonardi, S.： Monotony of Surprise and Large-Scale Quest for Unusual Words. In proceedingsof the 6th Int'l Conference on Research , " In Computational Molecular Biology, 2002 Washington, DC, April 18-21. pp 22-31.

[2] Christos FALOUTSOS. ： 5.1 SPACE FILLING CURVES, " In SEARCHING MULTIMEDIA DATABASES BY CONTEXT , 1996, pp27-34

[3] Dengsheng Zhang and Guojun Lu. : Review of shape representation and description techniques, " In Pattern Recognition, Vol. 37, 2004.

[4] D. Hilbert. :Über die stetige Abbildung einer Linie auf ein Flächenstück, " In Math. Ann. 38, 1891, pp459-460.

[5] J.A. Orenstein. :Spatial query processing in an object-oriented database system, " In SIGMOD, 1986, pp326-336.

[6] Jessica Lin, Eamonn Keogh, Stefano Lonardi, and Bill Chiu. : A Symbolic Representation of Time Series, with Implications for Streaming Algorithms," In SIGMOD workshop, 2003.

[7] Li Wei, Keogh, E. and Xiaopeng Xi. : SAXually Explicit Images: Finding Unusual Shapes," In the IEEE International Conference on Data Mining series (ICDM), 2006.

[8]Lonardi, S. ： Global Detectors of Unusual Words: Design, Implementation, and Applications to Pattern Discovery, " In Biosequences. PhD thesis, Department of Computer Sciences, Purdue University, August, 2001.

[9]大西史花 and 渡辺知恵美.： スマートハウスのセンサデータに対する SAX を利用したイベント検出の検討, " In deim2011, March 2011

[10]Sven Loncaric. : A survey of shape analysis techniques, " In Pattern Recognition, Vol. 31, 1998.

[11]UCI. ： http://archive.ics.uci.edu/ml/index.html," February 2012

[12]Yoshiki Tanaka and Kuniaki Uehara. ： Discover Motifs in Multi Dimensional Time-Series Using The Principal Component Analysis And The MDL Principle, " In Workshop on Machine Learning and Data Mining in Pattern Recognition(MLDM), 2003.

大西 史花 Ayaka ONISHI

お茶の水女子大学大学院人間文化創成科学研究科博士前期課程在学中．日本データベース学会学生会員．

渡辺 知恵美 Chiemi WATANABE

お茶の水女子大学理学部情報科学科講師．情報処理学会会員．日本データベース学会会員．