



お茶の水女子大学  
Ochanomizu University

理学部 情報科学科

第13回

卒業研究発表会

要旨集

2006年2月11日(土)

2月12日(日)

於 共通講義棟2号館201教室

# 平成17年度 理学部・情報科学科 卒業研究発表会 プログラム

場所：お茶の水女子大学 共通講義棟2号館 201大教室

2月11日(土)

9:30~10:45

宮武 莉奈	ニューラルネットPCAによる最適次元の決定	1
阪上 紗里	MinCamI Compilerにおける関数展開の制御	3
橘 春帆	左京と右京：2つの平安京ビューによるマトリクス型データの可視化	5
西山 慧子	大規模階層型データ可視化手法を用いた遺伝子ネットワークの可視化	7
菊池 愛美	実写画像とCGの合成による虹の表現	9

11:00~12:15

大瀧 杏奈	任意の頭部形状モデルへの髪型のフィッティング	11
小田 瑞穂	印象の近い音楽と画像の組み合わせを自動選出する手法	13
大山 喜冴	DIVA: 画像の印象に合わせた音楽自動アレンジの一手法の提案	15
五味 愛	大解像度画像からの類似部分画像の高速抽出 ~"VIEWGLE"の開発~	17
平野 未来	半透明複数等値面の重ね合わせの高速化	19

—— お昼休み ——

13:00~14:15

有間 久美子	格子グラフの彩色	21
玉田 高子	2次元空間におけるセルオートマトンの自己相似性	23
根岸 伸江	砂山モデルによる幾何学模様	25
倉地 彩乃	類似検索を用いた検索支援システムの開発	27
齋藤 舞	ユーザの入力キーワードに関連する時事的な情報を提供する情報推薦システム	29

14:30~15:45

渡邊 千明	グラフと強調するテキスト要約	31
奥村 奈穂子	グラフの挙動を表すテキスト生成	33
清水 美絵	音声・マルチモーダル情報を用いた情報家電との対話への取り組み	35
大西 可奈子	ユーザの目的達成を支援する知識をもった操作インタフェースの開発	37
田中 佳花	逆置み込み作用素による画像の修復	39

16:00~17:00

桑名 杏奈	S字型風車まわりの流れの数値的研究	41
水上 洋子	鉛直軸直線翼型風車における流れのシミュレーション	43
白谷 栄梨子	ビル風のシミュレーション —生活科学棟をモデルとして—	45
岡島 有希	高層ビル群によるヒートアイランド現象	47

2月12日(日)

9:30~10:45

村上 愛淑	塩味センサによる調味支援	49
杉野 碧	どの足スイッチ：キッチン用多機能フットスイッチの開発	51
早樋 沙織	トントンセンサ：キッチン向け入力インタフェースの開発	53
松崎 由香里	置いて動かす文字入力	55
辻田 眸	文鎮メタファを利用した小型情報機器向けインタフェース	57

11:00~12:15

岡野 典子	駅すきゅうずみ：無線 LAN を利用した降車駅リマインダ	59
小西 真裕美	生物対流の数値的研究	61
田中 悠紀	血管手術における効果の数値的検証	63
宮脇 梓	木星大気循環のシミュレーション	65
安田 史	温帯低気圧の簡易モデル	67

—— お昼休み ——

13:00~14:15

矢野 智美	RFID を用いた個人別情報提供システムの構築	69
郡司 京子	ウェブコンテンツを活用した仮想人物応答システムの研究	71
長澤 槇子	MP3 データの楽曲を対象とした検索システムの構築	73
千島 望	IP-SAN を用いたアプリケーション実行性能と TCP パラメータの相関関係評価	75
松井 愛子	ワイヤレス LAN 通信における TCP 輻輳ウィンドウ制御の動作解析	77

14:30~15:45

内田 智理	モバイルアドホックネットワークにおけるストリームデータ送受信の制御手法の検討	79
佐藤 有里子	ネットワークコンピューティング環境を用いた並列相関関係抽出の実行特性解析	81
久保 亜沙美	多次元尺度法とクラスター分析	83
井上 由美子	離散 2 値データからの最適決定木	85
石樽 彩乃	EM アルゴリズムによる混合コーシ分布問題	87

# ニューラルネットPCAによる最適次元の決定

宮武 莉奈 (指導教官: 吉田 裕亮)

## 1 はじめに

一般に高次元入力データの低次元化に関する方法は、データの中で分散の大きな軸を残し小さな軸を削除することにより実現される。このような、低次元化を行う重要な手法としてPCA(主成分分析)がある。

PCAとは、多くの変量の値をできるだけ情報の損失なしに、1個または少数個の総合的指標(主成分)で代表させる方法である。観測値の変量の個数を主成分に縮約するという意味で、データの次元を減少させる方法ということもできる。

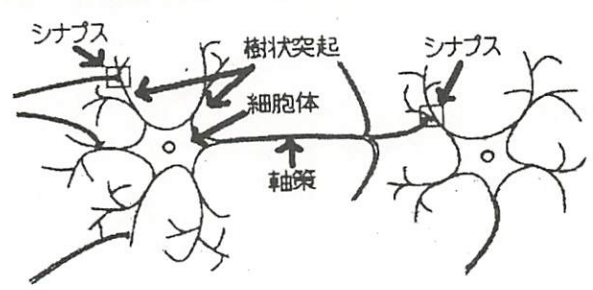
データ解析の手法のひとつにニューラルネット法がある。これは機械学習の一種であり、学習則として、入力に含まれている情報を最もよく保持するような形で出力するようにするとニューラルネットPCAが実現される。

本研究では、ニューラルネットPCA法を用い、学習則にRecursive Least Squares 学習アルゴリズムを採用し、縮約の最適次元の決定を行う手法を提案した。

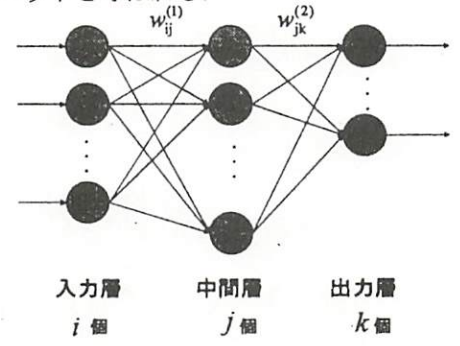
## 2 ニューラルネットワークとは

ニューラルネットワークは、生物の脳の神経細胞を模倣した計算メカニズムの総称である。人間や動物が日常的に行っている情報処理をパターン認識というが、それをコンピュータに行わせることは難しい。そのため、脳の仕組みを模倣したメカニズムによって人間のような柔軟なパターン認識を行わせる方法が研究されてきた。

図のように、人間の神経系は多くの神経細胞が結合したネットワークである。その要素である神経細胞は、多数の神経細胞から入力信号を受け取って情報処理を行い、処理結果を多くの神経細胞に出力している。



一つの神経細胞を模倣し、それらをつなげてネットワーク化したものが、下図のニューラルネットワークである。ニューラルネットワークにおいて、神経細胞はユニットと呼ばれる。

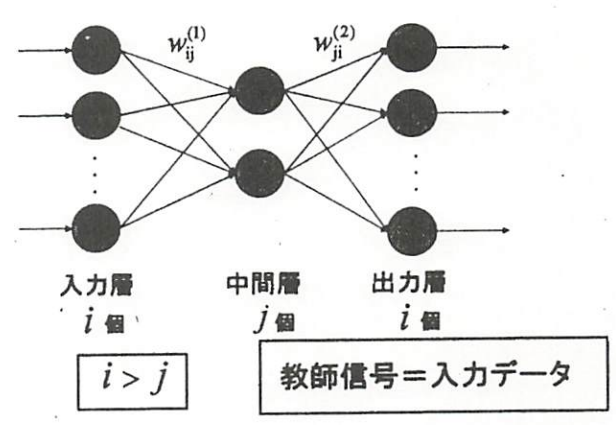


生物が学習すると正しいものと誤ったものとの判別の正答率が上がるように、ニューラルネットワークにも学習させることにより、その正答率を上げることができる。

## 3 ニューラルネットPCAとは

ニューラルネットの応用例として、ニューラルネットPCAがある。ニューラルネットPCAは、情報量ができる限り損なわず次元を縮小する。つまり、ニューラルネットワーク上ではある条件のもとでニューロンに作用する学習則は、入力に含まれている情報を最もよく保持するような形で出力するように、ニューロンを成長させることに対応する。

3層パーセプトロンによるニューラルネットPCAの特徴は、入力層と出力層のユニットの数は同じで、中間層のユニット数をそれよりも少なくしたモデルである。つまり、教師信号として入力データを用いる。



ニューラルネットPCAは、次のように実行される。各  $n$  次元の入力データ  $x$  を、 $(x_1, x_2, \dots, x_n)$  とし、データの集合をベクトル  $w$  に平行な軸へ投影した分布が、最大の分散をもつように  $w$  を学習する。

$\sum w_i^2 = 1$  の場合、 $w$  への  $x$  投影はまさに  $y = \sum_i w_i x_i$  であり、投影した分布の分散は  $y$  の分散に等しくなる。

## 4 Recursive Least Squares 学習アルゴリズム

$x(t)$  ( $t$  は学習時間) を  $N$  次元の入力データ列とし、 $p$  ( $p < N$ ) 個の主成分に対して、結合強度  $w_1, w_2, \dots, w_p$  を学習する。つまり、データの共分散行列  $C (= E[xx^t])$  の  $p$  個の主要な固有ベクトルを求めることに対応する。

以下に示すアルゴリズムは、Sanger によって提案された、第  $m$  番目 ( $m < p$ ) の固有ベクトル  $w_m$  を学習するアルゴリズムを一部改良したものである。

Step1: ネットワークの初期化を行う。  
 入力層と出力層の結合強度の初期値  $w_m(0)$  を乱数を用いて設定する。  
 更に  $P(0)$  を任意の値に設定する。

Step2: 入力層に入力データ  $x(t)$  を入れる。

Step3: ネットワークの出力 (中間層) を計算する。

$$h_m(t) = w_m^t(t-1)x(t)$$

Step4: 学習速度係数  $K_m(t)$  を定める。

$$K_m(t) = \frac{P_m(t-1)h_m(t)}{1 + h_m^2(t)P_m(t-1)}$$

Step5: 結合強度の更新をする。

$$w_m(t) = w_m(t-1) + K_m(t)[x(t) - \sum_{i \leq m} h_i(t)w_i(t-1)]$$

Step6:  $P_m(t)$  の値を定める。

$$P_m(t) = [1 - K_m(t)h_m(t)]P_m(t-1)$$

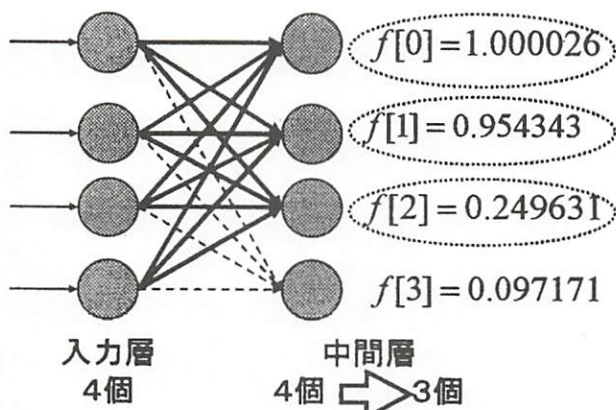
## 5 ニューラルネットを用いた最適次元の決定

機械が自動的に次元を決めるのは難しい。ここで、提案する最適次元の決定法は、RLS で得られた結合強度  $w_1, w_2, \dots, w_p$  それぞれの2乗の和を  $f[m]$  ( $1 \leq m \leq p$ ) と置き、 $f[m]$  の比を調べた。 $f[m]$  の比が小さい時、中間層のユニットをひとつ消す。これを順次繰り返して、値の差が見られなくなった時点で終了させる。最後に残った中間層のユニット数を最適次元とみなす。

## 6 実験例

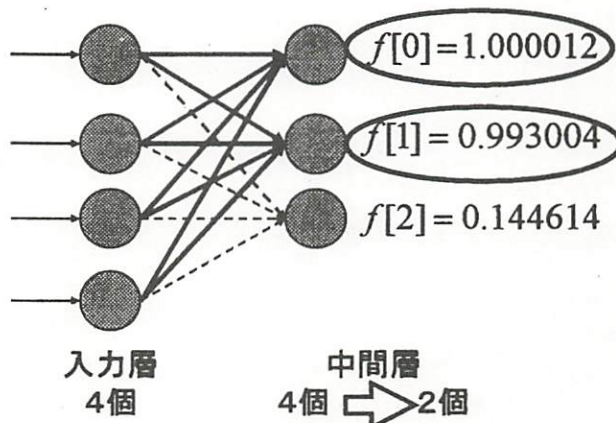
実験として、入力データは4次元、最適次元が2次元となるシミュレーションデータを用意する。

ニューラルネットPCAにおいて、入力層と出力層のユニットは同数で、教師信号は入力データと同じあるため以下の図では出力層は省略した。



最適次元を決定する場合、中間層のユニットの順序は本質的ではない。

初めに  $f[3]$  の値から、 $w_3$  を消す。



次のステップで  $f[2]$  の値から、 $w_2$  も消す。最後に  $f[1]$  は  $f[0]$  とほぼ同じ比であるため、 $w_0, w_1$  は残る。

よって残った中間層のユニット数は2個であるため、最適次元は2次元と推定できる。

## 7 まとめと今後の課題

今回、ニューラルネットPCAを用いて最適次元を決定する手法を提案した。

結合強度が弱い中間層のユニットを順次消していき、最終的に残ったユニット数を最適次元とみなした。これはネットワーク学習の面からみれば、忘却を行っているといえる。

今後の課題として、入力データのどの変量たちが残ったユニットの結合強度に関与しているかを特定できるように拡張したい。

また、実データを用いた分析を行いたい。例えば、医学分野では病気に一番影響を与えている要因の抽出が可能であると考えられる。

## 8 参考文献

- [1] 麻生秀樹, 津田宏治, 村田昇: パターン認識と学習の統計学, 岩波書店
- [2] 馬場則夫, 小島史男, 小澤誠一: ニューラルネットの基礎と応用
- [3] R. ヘクト・ニールセン, 訳 袋谷賢吉: ニューロコンピューティング
- [4] 小川順子: ニューラルネットPCA法を用いた固有値・固有ベクトルのモンテカルロ計算, 1998年度お茶の水女子大学情報科学科卒業論文
- [5] 藤井亜矢子: ニューラルネットPCA法を用いた最大固有ベクトルのモンテカルロ計算, 1997年度お茶の水女子大学情報科学科卒業論文

# MinCaml Compilerにおける関数展開の制御

阪上 紗里 (指導教官: 浅井 健一)

## 1 はじめに: Compiler とは

コンパイラ [1] とは、ある言語のプログラムを、より低いレベルの言語に変換する処理のことをいう。プログラムは文字列として字句解析に渡され、トークンとなり、さらに構文解析に渡されると構文木となって出力される。ここで出てきた構文木を各種最適化にかけ、レジスタ割り当てをし、コード生成によりアセンブリをつくり、機械語に変換される。

## 2 各種最適化における関数展開

各種最適化では、いくつかの高水準言語と低水準言語のギャップを埋める処理が行われる。コンパイラの最適化において、関数展開とは関数呼び出しをその本体で置き換えることである。この処理により関数呼び出しと復帰に要するコストを低減させ、さらに関数展開は他の最適化処理(定数伝播など)を可能にする場合がある。また関数展開が行えない場合も一部の既知の情報に特化された関数を生成することで、有効な最適化を行える場合がある。

ただしむやみに展開や特化を行うと無限ループやコード爆発を起こす危険性がある。そこで本研究では、ユーザの注釈にしたがって(1)再起変数がわかっているときは関数展開(2)関数展開が行われなかった場合には特化を行うという機能を MinCaml[2] コンパイラの最適化処理に組み込んだ。

## 3 関数展開の制御方法

関数展開が無限に繰り返されるひとつの原因として、再帰関数が停止しない場合がある。再帰関数を関数展開した場合、関数の本体部分にさらに関数自身が呼び出されるため、無限に関数展開してしまう。このことから、関数展開が停止することが明確な場合のみ関数展開が行われるようにする。具体的には「停止する条件」として、再帰に関わる変数(再帰変数)を注釈に指定し、それらが既知であれば関数展開を行う。例えば  $m$  の  $n$  乗を求める関数は

```
let rec power m n =
```

```
  if n = 0 then 1 else m * power m (n-1)
```

と定義され、この関数の再帰変数は  $n$  である。今未知の  $a$  に対して  $\text{power } a \ 2$  のように呼び出されたとする。この場合は関数展開を行うと、以下のように停止することがわかる。まず  $\text{power } a \ 2$  が

```
  if 2 = 0 then 1 else a * power a (2-1)
```

のように展開される。さらに定数伝播により条件文の実行を行うと  $a * \text{power } a \ 1$  となる。さらに  $\text{power}$

$a \ 1$  を展開していくと、もう 2 回関数展開が行われ、その結果  $a * (a * 1)$  が得られる。このように関数展開は停止する場合、強力な最適化となる。実際、得られたプログラムにおいてはひとつも関数呼び出しがされておらず、また、直接 2 乗を計算するプログラムに最適化されている。この処理により、大幅に実行時間が短縮されることが期待される。

ただ、いつでも  $\text{power}$  を関数展開して良いわけではない。再帰変数が未知の場合には、展開すると無限ループになる。例えば未知の  $b$  に対して  $\text{power } 2 \ b$  が展開されると、関数自体が本体と置き換わるので、

```
  if b = 0 then 1 else 2 * power 2 (b-1)
```

となる。 $b = 0$  かどうかはわからないので、定数伝播は行えない。さらに  $\text{power}$  の関数展開を行っても  $n$  乗の部分は 0 にはならず、展開され続けることになる。このように、延々と繰り返されてとまらない場合は関数展開を行うことはできない。

## 4 特化の制御方法

再帰変数の値が未知で関数展開を行えない場合も、再帰変数以外の値を使って最適化できる場合がある。既知の変数の値を使って新たな関数を作ることを、特化という。以下で定義される Ackermann 関数において、実際にどのように特化されるかを示す。

```
let rec ack x y =
```

```
  if x = 0 then y + 1 else
```

```
  if y = 0 then ack (x - 1) 1 else
```

```
  ack (x - 1) (ack x (y - 1))
```

今  $\text{ack } 3 \ b$  で呼び出されたとする。 $b$  が未知のまま展開すると、停止しないので特化を行うことにする。すると以下のような  $\text{ack}$  の  $x=3$  バージョンが新たに作成される。

```
let rec ack3 y =
```

```
  if 3 = 0 then y + 1 else
```

```
  if y = 0 then ack (3 - 1) 1 else
```

```
  ack (3 - 1) (ack 3 (y - 1))
```

```
in ack3 b
```

さらに定数伝播により最初の条件文が計算され、

```
let rec ack3 y =
```

```
  if y = 0 then ack 2 1 else
```

```
  ack 2 (ack 3 (y - 1))
```

```
in ack3 b
```

と変換される。ここで新しく作られた関数の本体に現れた  $\text{ack } 3 \ (y - 1)$  は、 $x=3$  バージョンとして既につくられているので、自分自身への呼び出しに置き換

える。つまり最終的に

```
let rec ack3 y =
  if y = 0 then ack 2 1 else
    ack 2 (ack3 (y - 1))
in ack3 b
```

が得られる。このように、特化によって条件文がひとつ減った新しい関数ができあがる。

## 5 filter 機能の追加

以上の処理をプログラム中に filter 機能を追加することで実現する。filter は関数を宣言する際に、

```
let rec 関数名 =
  (*{再帰変数; 特化変数}*) body
```

のようにコメントとしてユーザが記述する。これで (1) 注釈にかかれた再帰変数が既知なら関数展開を行い、(2) 関数展開を行わないときには、特化変数の値を使って、特化を行うこととする。特別な場合として、filter をかかないときには、関数展開も特化も行わない。また (\*{ }\*) は常に関数展開を行うという意味になる。

さらに再帰変数の部分には、コード爆発を回避する方法として、その範囲をかけるようにした。これらの変数が全て既知であり、なおかつ範囲内であれば関数展開が行われる。

## 6 filter の実例

power 関数の再帰に関わる変数は n, それ以外でわかっていたら置き換えて欲しい変数は m なので、以下のように filter を付け加える。

```
let rec power m n = (*{n;m}*)
  if n = 0 then 1 else m * power m (n-1)
```

この filter のもとで power a 2 と呼び出されると、先に示した関数展開が行われて a \* (a \* 1) というプログラムになる。

一方 power 3 b で呼び出された場合は、

```
let rec pwoer3 n = (*{n; }*)
  if n = 0 then 1 else 3 * power3 (n-1)
in power3 b
```

という m が 3 に特化されたプログラムが得られる。

Ackermann 関数の実行で ack 3 b の呼び出しについては filter に (\*{;x}\*) を加えれば 4 節に示したような特化が行われ、x=3, x=2, x=1, x=0バージョンが作られる。しかし 4 節に示した結果をみると、x, y 両方がわかっている呼び出しが現れている。これはあらかじめ filter として (\*{x y; x}\*) を加えてやれば先に計算できる。そうすると ack 2 1 の部分が計算され、

```
let rec ack3 y =
  if y = 0 then 5 else
    ack 2 (ack3 (y - 1))
in ack3 b
```

と特化することができる。

一方 Ackermann 関数は y の値がわかっている場合、y=0 と y=1 の場合がお互いを呼び出している場合を除いて、関数展開が停止することがわかる。そこで (\*{0<y; }\*) を filter にくわえると、y に対する if 文を実行することができ、結果として ack a 1 を展開すると、次のようなプログラムを得ることができる。

```
if a = 0 then 2 else
ack (x - 1) (if a = 0 then 1 else
              ack (a - 1) 0)
```

## 7 実行結果

本研究における関数展開の制御の効果を分析するため、プログラミング言語 OCaml のサブセット MinCaml [2] に filter 機能を追加した。もとの MinCaml のコンパイラは、関数のサイズが小さければ、関数展開を行っていた。今回の filter の効果を、これと比較することにする。実行環境を、CPU に 75MHz の SPARC を持つ Sun Blade 1000 (メモリが 512Mb) として、Ackermann 関数である ack 3 12 を実行する。

関数展開なし	9.77 秒
サイズによる関数展開	10.33 秒
(*{ ; x}*)	7.77 秒
(*{ x y; x}*)	7.35 秒
(*{ 0<y; }*)	9.77 秒

関数のサイズをみただけの関数展開では、実行時間が長くなってしまっているが、適切に制御することで、最適化できていることがわかる。

## 8 まとめと今後の課題

filter をユーザが記述することで、関数展開と特化をきめこまかく制御する方法を提案、実装した。Ackermann 関数においては、特化の効用が大きく現れた。特化を行うことで、関数展開がさらに可能になる場合があることがわかった。

現在のシステムにおいて、関数展開されたプログラムを共有することができていない。そのため、前に作られた特化の関数を使いまわすことができず、新たに同じ挙動をしめす特化関数がつくられ、コード爆発を起こす一端が見られている。また別の方向として特化用情報の範囲を書けるようにすることも考えられる。

## 参考文献

- [1] Appel, A. W. *Modern Compiler Implementation in ML*, New York: Cambridge University Press (1998).
- [2] Sumii, E. "MinCaml: A Simple and Efficient Compiler for a Minimal Functional Language," *ACM SIGPLAN Workshop on Functional and Declarative Programming in Education (FDPE '05)* pp. 27-38 (September 2005).

## 左京と右京：2つの平安京ビューによるマトリクス型データの可視化

橘 春帆

(指導教官：伊藤貴之)

### 1. 概要

情報可視化は、世の中にある一般的な情報を可視化する研究分野であり、最近になって活発に研究が進められている。情報可視化が有効に活用できると考えられるデータの一つに、数千人規模のアンケートを集計したものが考えられる。

アンケート集計結果は回答者数だけで列を持ち、質問に対する選択肢数だけ行を持つマトリクス型データの一種と考えることができる。このようなマトリクス型データを可視化する手法として、本研究では情報可視化手法である「平安京ビュー」を二つ用いて表示する手法「左京と右京」を提案する。本手法では、右京は回答者数だけのノードを有する階層型データ、左京は選択肢数だけのノードを有する階層型データを表示する。

今までのアンケート集計結果の可視化には帯グラフや棒グラフなどが使用されていた。しかし、このようなグラフでは、回答者の全体の傾向などがわかるだけで、一人一人の特徴を見たいときには、自分がグラフの面積の少ないところを見て、何度も操作をして引き出さねばならなかった。本研究では、ある特定の選択肢を選んだ回答者たちの情報をピンポイントでみることが可能になる。また、特定の回答者が選んだ選択肢だけを見ることも可能になり、少数だけど興味深い局所的傾向の発見ができると考えられる。

### 2. 関連研究

本研究に用いる情報可視化手法「平安京ビュー」[1]は大規模な階層型データの可視化に向いている。平安京ビューは階層型データ全体を一画面に配置する情報可視化手法である。葉ノード、枝ノードの直交配置を意識したアルゴリズムで、平安京の地図に似ていることから命名された。階層型データを可視化する手法には、木構造を表示する手法[2][3]や、個々のノードが持つ占有率に比例した面積で画面を分割した手法[4]がある。これらの手法に比べて、平安京ビューは階層型データを構成するすべての葉ノードを、一画面に、同じ大きさで、重なることなく表現できる。この特徴は、アンケート情報を構成するすべての回答者と選択肢を一画面に表示したい、という本研究の目的に向いているといえる。また葉ノードの個数に比例した面積で画面を分割し、葉ノードを平等に一画面に表示する手法[5]もある。この手法は平安京ビューに近い特徴を持つといえる。

### 4. 提案内容

本研究では、平安京ビューを用いた新しい試みとし

て、マトリクス型データに対する階層型データクラスタリングの結果として得られる階層型データの可視化を考える。

$n$  行  $m$  列のマトリクス型データに対して、 $m$ 次元ベクトルを持つ  $n$  個のデータ要素と見え、ベクトル間距離の近いものをボトムアップ的にクラスタリングし、結果として数段階の階層を持つ階層型データを形成する。

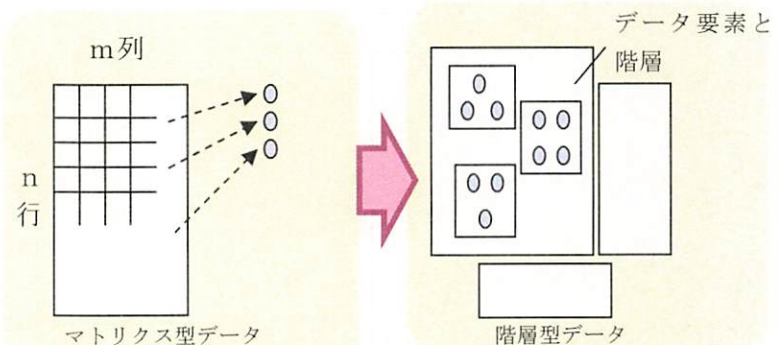


図1 マトリクス型データに対する階層型クラスタリング

$n$  行に対して1つの平安京ビュー、 $m$  列に対しても1つの平安京ビューを出し、マトリクス型データに対して2つの平安京ビューを出すことが本研究「左京と右京」の新しい点である。

今回、「左京と右京」が対象とするアンケート集計データは、選択肢、回答者を行と列にするマトリクスデータである。「右京」は回答者をノードとする階層型データを表示する。「左京」は選択肢をノードとする階層型データを表示する。「右京」および「左京」で表示する階層型データの生成にはクラスタリングを適用する。クラスタリングには Cluster3.0[6]を使用する。

右京のノードはユーザーが選んだ項目値を形と色で表す。左京のノードはアンケート選択肢に該当する人数を高さで表す。

この可視化方法では、クラスタリングした結果として得られるグループ内の相関性の真偽、強弱などを知るのに有効と考えられる。またアンケート集計結果をクラスタリングした結果の大人数のグループだけでなく少人数のグループについてもその回答の詳細を可視化するのが容易になる。

### 5. 実行結果

4節で述べた提案手法を用いて、本大学情報科学部のB1～M2学生に対して、自作した「女子大生の就職に対する意識調査アンケート」を実施した。アンケートの選択肢数は179、回答者は118であった。図2は「右京」の結果である。回答者がノードである。



赤色は選択肢「現在、親と同居している」を選んだ人、青色は現在、親と同居していない人を表す。正方形のノードは、「就職する際に、親の意見を重視しますか？」という質問に「はい」と答えた人を表していて、ひし形ノードは「はい」と答えなかった人である。図2では青色ノード、赤色ノード、ひし形ノードともに局所的に集中して見られる。例えば○部分の2箇所のクラスタはひし形ノードだけで構成されている。このことから上の2つの質問は回答者のクラスタリング結果に強く寄与していることが考えられる。

図3は「左京」の結果である。選択肢がノードで、色はある回答者が選んだものだけ赤色になっている。高さは各選択肢の回答者の人数になっている。図の○部分は、「就職活動は楽しい」「就職を見通したアルバイトをしている」の2つの選択肢が1箇のクラスタを構成している。そして両選択肢ともに、高さが低いので少数意見だとわかる。このような少数かつ興味深いクラスタに属する選択肢を2つとも選択している回答者を、すぐに発見することができた。

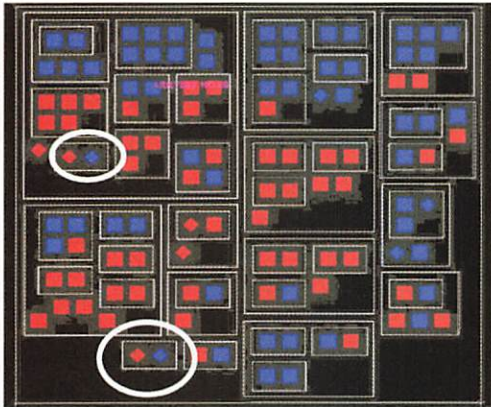


図2 「右京による回答者分布の可視化結果」

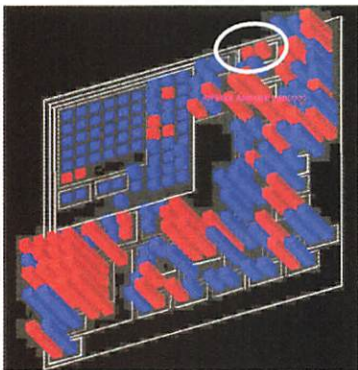


図3 「左京による選択肢分布の可視化結果」

## 6. 今後の課題とまとめ

本研究では、2つの平安京ビューを使った可視化を提案し、アンケート集計結果の可視化をした。今後、この2つの平安京ビューを一画面に表示し、対話的表

示ができるようにすることで、注目している特定の回答者、特定の選択肢の詳細表示をする予定である。具体的には、右京の回答者ノードをクリックすると、左京のその人が回答した選択肢ノードが所定の色で示されるような機能を追加したい。まや、左京の選択肢ノードをクリックすると、右京のその選択肢を選んだ回答者ノードが所定の色で示されるような機能を追加したい。

## 6. 応用例

今回、「左京と右京」でアンケート集計結果の可視化をしたが、次のような応用例も考えている。応用例の1つ目は「オンラインショッピングの可視化」である。右京に購入者、左京に商品名を表示して、商品購入の動向を可視化できると考えられる。2つ目は、「文献とキーワードの可視化」である。右京にキーワード、左京に論文名を表示して、どのキーワードがどの論文に使用されているかという傾向を可視化できると考えられる。

## 謝辞

生活科学系アンケートの採取方法に関する情報提供をしてくださった本大学生活科学部の御船美智子教授、永瀬伸子助教授、畑江敬子教授、香西みどり助教授、Cluster3.0を開発された Maichel de Hoon 氏、アンケート作成時にアドバイスしていただきました本学人間社会科学科 B4 小園絢子さん、アンケートの回答にご協力していただいた本学情報科学科の皆様にご感謝の意を表します。

## 参考文献

- [1]伊藤, 小山田, 平安京ビュー ～ 階層型データを基盤状に配置する視覚化手法, 可視化情報学会第9回ビジュアリゼーションカンファレンス, 2003.
- [2] Carriere J., et al., Research Paper: Interacting with Huge Hierarchies beyond Cone Trees, IEEE Information Visualization 95, pp.74-81 (1995).
- [3]Lamping J., Rao R., The Hyperbolic Browser: A Focus+context Technique for Visualizing Large Hierarchies, Journal of Visual Languages and Computing, Vol. 7, No. 1, pp. 33-55 (1996).
- [4]Johnson B et al., Tree-Maps: A Space Filling Approach to the Visualization of Hierarchical Information Space, IEEE Visualization '91, pp. 275-282 (1991).
- [5]Bederson B., Schneiderman B., Ordered and Quantum Treemaps: Making Effective Use of 2D Space to Display Hierarchies, ACM Transactions on Graphics, Vol. 21, No. 4, pp. 833-854 (2002).
- [6]Cluster3.0 ソフト  
<http://bonsai.ims.u-tokyo.ac.jp/~mdehoon/software/cluster/software.htm#ctv>

# 大規模階層型データ可視化手法を用いた遺伝子ネットワークの可視化

西山慧子（指導教官：伊藤貴之）

## 1. 概要

情報可視化は世の中にある一般的な情報を可視化する研究分野であり、最近になって生物情報の可視化の研究が活発に進められている。現在ヒトゲノム解読は完了しているが、DNAを構成する塩基配列が解読されただけで、その遺伝子のふるまいなどは、はっきり分かっていない。そこで現在その遺伝子の振舞いについての研究が必要とされている。その中でもマイクロアレイデータ[1]からの遺伝子ネットワーク同定問題は、バイオインフォマティクス分野における重要なトピックのひとつであると言える。

遺伝子ネットワークとは、各遺伝子をノードとし、遺伝子間にエッジがあるようなグラフ構造で、ゲノム上での位置関係、代謝、制御パスウェイ上での隣接関係、転写時の共発現率、たんぱく質相互作用など、多くの性質を表現するために用いられる。またパスウェイなどの相互関係は、エッジが方向を持つ。これら进行分析することで、遺伝子に変異したときに何が起こるか予測することができる。

このデータは数千の遺伝子群からなることもあり、複雑な連結成分をふくんでおり解釈や把握が困難であるので、何らかの方法でより興味深い遺伝子群を抽出し注目すべき対象を絞り込むことが必要である。しかしながら常に目的に叶った結果をきりだせていないのが現状である。

また遺伝子解析において、それぞれの相互関係を示す現象として、マルチドメイン（2種の遺伝子の機能を同時にもつ遺伝子）等が挙げられる。

これらの遺伝子は、発現率によるクラスタリング結果と遺伝子ネットワークを組み合わせて可視化することにより、その存在が理解しやすくなると考えられる。

そこで本研究では、例えば複雑なネットワークの中から、遺伝子のクラスタリングの結果を同時にみせつつ、ネットワーク中の注目部分を抽出できる可視化手法を提案する。ここでいう注目部分とは、例えば検索結果である。この研究結果により、研究者は全体の中から特定の遺伝子の相互関係を知りたい時、混乱せず抽出でき利用できるのではないかと考えている

Visualization of genome network using large-scale hierarchical data visualization technique

Keiko Nishiyama, Takayuki Itoh,

Ochanomizu University{nishy, itot}@itolab.is.ocha.ac.jp

## 2. 関連研究

ネットワークの可視化手法は、すでに多く発表されているが、遺伝子ネットワークの研究に向いているネットワーク可視化は、複雑な操作がいらず、簡単に注目部分だけを抽出でき、大規模なネットワークを扱える手法である。

注目部分を強調する為のネットワーク可視化の研究は、すでに多く発表されている。データ中の注目部分を、魚眼レンズでみたように拡大表示し、その周囲部分を画面の端によせる可視化手法が報告されている[2]。だが拡大表示部分の外にあるノードは、エッジで連結されているのに画面の端においやられてしまう。また、データ中の注目部分を3次元的に引き上げて、それとエッジで連結されているノードも浮き上がらせることで、データの注目部分の相互関係をわかりやすく表示する手法が報告されている[3]。だがこれも、複数のノードを同時に注目したい時の対応がなされていない。

## 3. 提案内容

本研究では、ネットワークのデータ中の注目部分の抽出に向けた情報可視化を実現するために、以下のような点を重視する。

- (1) 注目している部分を強調でき、相互作用関係を判別しやすいこと
- (2) できるだけ多くの遺伝子をクラスタ単位で表示できること

そこで(1)を満たすために、データ中の注目部分を、3次元的に引き上げることにより、そのノードとエッジでむすばれているノードの関連を強調させ、周囲との関連の表示を行う。

また(2)を満たす為、まず  $xy$  平面に、クラスタごとにノードを配置し、続いてそのノードの  $x$  座標値、 $y$  座標値を変更することなく相互関係を強調させるようにする。 $xy$  平面上のノード配置には、平安京ビュー[4]を拡張した手法を提案する。

本手法は、 $M$  個のマイクロアレイ上にそれぞれにある  $N$  個の遺伝子、 $M \times N$  のマトリクス型データを対象とし、 $N$  個の遺伝子のうち発現率傾向の近いものをクラスタリングし、平安京ビューにおける階層型データに変換する。以下の図 1 に概要を記す。本研究は、3次元空間の底面に平安京ビューを用いて、遺伝子のクラスタリング結果を可視化する。このノード一つ一つが、一つの遺伝子に対応し、クリックブルになっており、その遺伝子の詳細情報を提示することができる。

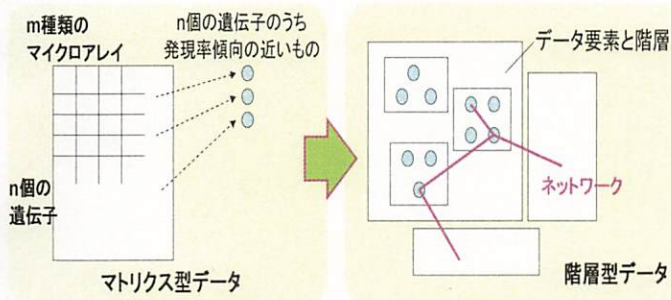


図 1: 階層型データへの変換

またその上にネットワークを重ね、注目ノードをクリックするか、または検索エンジンのような GUI で指定すると、その注目ノード、およびそれをエッジで連結されたノードを、3 次元的に引き上げて表示する。

ネットワークは、2 個のノード（遺伝子）のペア、nodeA, nodeB とし  $m$  個のマイクロアレイがあったとすると、

nodeA の発現率を  $A = \{a_1, a_2, \dots, a_m\}$

nodeB の発現率を  $B = \{b_1, b_2, \dots, b_m\}$  とする。

発現率どうしの相関性を算出する式は、

$1.0 - \text{distance}(A, B) / \text{maxDistance}$  とする。

なお  $\text{distance}(A, B)$  は A, B 間のユークリッド距離の 2 乗であり、 $\text{maxDistance}$  はその最大値である。この算出式は、クラスタリングに使用したソフトウェア Cluster3.0 にも用いられている。の値が一定値（自分で与えた値）より大きい時、この 2 つのノードを接続するエッジを表示する。

また、クラスタ間をまたぐエッジを多く持つ遺伝子を、あらかじめ、光らせるようにする。これにより、特殊な反応のありそうな遺伝子群を表示する。

本研究により、クラスタリングを行い平安京ビューで表示を行う為、整然とした表示可能になり、全体を一画面表示することにより遺伝子などの膨大なデータに対応できる。また、ネットワークを重ねることにより、相互関係が理解しやすく、クラスタリングだけでは表現しにくかった特性（マルチドメインなど）を表現できる。

#### 4. 実行結果

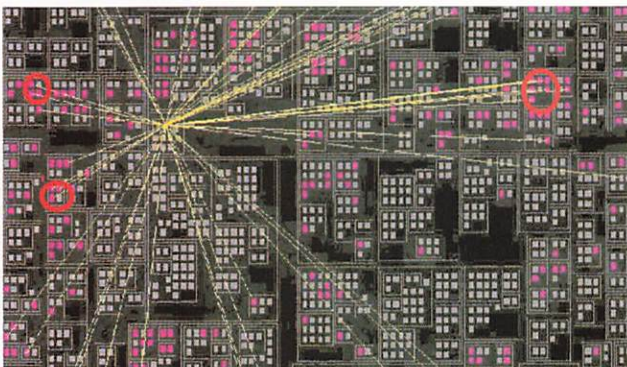


図 2: 提案手法を用いた、注視ノードが一つの実行例

本手法による可視化の例を図 2 に示す。ネットワー

クを表示させ、注視ノードを一つとし、そのノードと結ばれているエッジだけを表示している。また、クラスタ内のノードすべてと、注視ノードがつながっているクラスタを丸で囲んだ。

この実行例から、この注視ノードは、丸で囲んでいるクラスタ内すべてと、結ばれていることより、すべての属性をふくんでいるマルチドメインかもしれないという事が発見できた。また、このデータにおいて、注視ノードから出ているエッジは、先ほど光らせたノードとほとんど結ばれていることより、クラスタを多くまたぐエッジを多く持つ遺伝子同士が複雑に絡み合いネットワークを作っているという事も発見できた。

#### 5. まとめと今後の課題

本論文では、複雑なネットワークの中から、遺伝子のクラスタリングの結果を同時にみせつつ、ネットワーク中の注目部分を、3 次元的に引き上げることにより抽出できる可視化手法を提案した。

本手法において、ネットワークとクラスタを同時表示することにより、マルチドメインなどの特性を表現でき、またクラスタをまたぐエッジを多く持つ遺伝子を、色をつけることにより、おもしろい反応のありそうな遺伝子の絞込みもできることが解った。

今後の課題として、発現率の代わりに、グラフに向きのあるパスウェイを用いた遺伝子ネットワークの可視化を試みたい。また、オントロジーなどの遺伝子同士の関係性などもくわえていきたいと考えている。また、可視化された特徴的な現象が、本当に遺伝子学的に興味深い特性なのか否かの検証も必要である。

#### 謝辞

本報告のクラスタリングに関して、コロンビア大学 Mickael De Hoon 氏から、cluster3.0 のソフトを提供していただいた事を感謝いたします。

遺伝子ネットワークに関する議論に関して、東京大学 宮野教授、中谷助教授、渋谷講師、瀬々助手、井本助手から貴重なご意見を賜ったことを感謝いたします。

#### 参考文献

- [1]有田正規、遺伝子ネットワークと確率モデル、2001 年ベイジアンネットチュートリアル。
- [2]Manojit Sarcar, MarcH .Brown, Graphical Fisheyes Views of Graphs, Communication of the ACM, pp73-83, Volume 37, March 1994.
- [3]塩澤他, 「納豆ビュー」の対話的な情報視覚化における位置付け、情報処理学会論文誌, 38, 11, pp. 2331-2342, 1997
- [4]伊藤、小山田。平安京ビュー～階層型データを基板状に配置する可視化手法、可視化情報学会第 9 回ビジュアルセッションカンファレンス、2003。

# 実写画像とCGの合成による虹の表現

菊池愛美 (指導教官: 伊藤貴之)

## 1. 概要

映画やドラマ等の映像で自然現象の表現にCGを用いることがある。その中でもよく用いられるのは、発生条件が厳しい現象、例えば虹である。CGを用いて自然現象を表現する手法は、ハードウェアの発達により高度な計算技術の適用が可能になり、より写実的な表現を様々な手法で実現できるようになってきた。しかし、CGによって表現された自然現象を撮影映像に合成する技術に関しては、まだ議論の余地があると考えられる。

そこで本研究では、実写画像にCGで生成した虹をリアルに合成する表現手法を提案する。本手法では、実写画像に撮影された各物体の奥行きを仮想することで三次元仮想空間を構築する。虹は空気中の水滴内部での光の反射によって起こる現象である。本手法では、実写画像の背景物体よりも前方にある水滴内部で視点に向かって反射する総量を、視点から背景物体までの線積分によって算出する。さらに、その反射光が目が届くまでに空気中で散乱してしまう量を、視点から水滴までの線積分によって算出する。これにより、状況に応じて違った見え方をする虹を表現できると考えられる。例えば雨上がりの虹と噴水の周りに見える虹の違い、近くの建築物にかかる虹と空にかかる虹の違い、などを表現できる。

本手法において実写画像と虹を合成する考え方は、虹に留まらず空気中の浮遊物がもたらす気象現象全般の表現に応用できると考えられる。

## 2. 関連研究

虹は、大気中の水滴内部で反射した光が、波長による屈折率の違いによって分光して目に見える現象である。これをCGで表現した初期の研究に、伊藤ら[1]による研究がある。また最近では芳信ら[2]による研究がある。これは、波動工学を使うことで水滴の半径の変化を可能にしている。文献[2]と違い、本研究は波動工学を考慮せず、かわりに天空以外の物体にかかった虹を表現することを考慮する。

また、虹以外の天空気象現象のCG表現に関する研究もいくつか存在する。岩崎ら[3]の研究では散乱マップを用いて多重散乱の計算と空の色の高速表示法を提案している。虹と同じような発生条件の厳しいオーロラのビジュアルシミュレーション[4]では、発光原理を考慮してオーロラを表示する手法を提案している。

## 3. 提案内容

図1は本手法の処理手順を示したものである。本手法では、まず実写画像から三次元空間を構築し、三次元空間中の水滴分布を入力し、これらを参照して虹を表現し、最後にこれを実写画像と合成する。

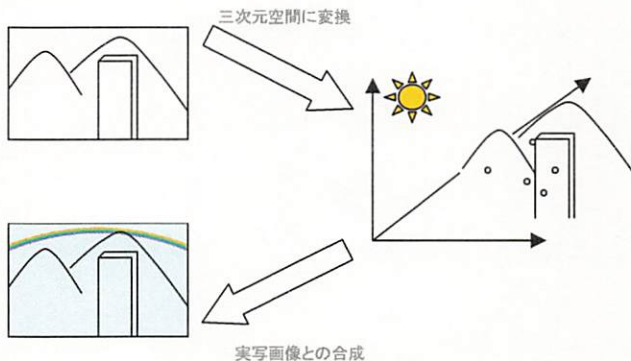


図1: 3次元仮想空間

本手法ではまず、実写画像から三次元空間を構築する。現時点の実装では、まずk-means法を用いて実写画像の画素値をクラスタリングし、代表色を得る。本手法では代表色ごとに奥行き値を入力することで、実写画像から三次元空間を構築する。空や山などの遠くに見えるものと家や人など近くに見えるものなどに区別することで、大まかな奥行きを表現できる。ただし複数の物体の色、あるいは物体色と背景色が同じ代表色で括られて場合には、本手法は上手く動作しない。この点については改良が必要である。これとは別に本手法では、三次元空間における大気中の水滴の分布、および太陽光線の入射方向を入力することを想定する。

これらの情報を用いて本手法は、虹を構成する光の色と強度を算出する。

虹の光と実写画像の合成による輝度 $I$ を求める手順は以下の通りである。空気中の水滴から反射して視点に届く光の輝度を $H$ 、それが大気中で散乱吸収せずに視点に届く割合を $S_H$ とする。また背景光の強度を $B$ 、それが大気中で散乱吸収せずに視点に届く割合を $S_B$ とすると

$$I = HS_H + BS_B \quad (1)$$

とできる。

次に、式(1)の $S_H$ を計算する。図2は、太陽光が水滴内部で反射してできる虹の光線と読み取った実写画像の物体からの光線を表したものである。

Representation of rainbows in Image based Rendering

Manami Kikuchi, Takayuki Itoh

Ochanomizu University

{manami, itot}@itolab.is.ocha.ac.jp

ここで水滴濃度  $w(x)$ 、光が減衰せずに通過する割合  $G(x)$  を仮定する。 $x$  は視線上の位置を表すものとする。水滴濃度と減衰率は位置の関数で表せるので、 $w(x)$  における水滴による光の強度は(3)に比例する値になる。

$$w(x) \prod G(t) \quad (3)$$

視点に届くのは視線上の各位置における水滴の反射光をの総和になるので、(3)を積分する。

$$S_H = \int_0^d w(x) \prod G(t) dx \quad (4)$$

$d$  は背景物体までの距離、または水滴の反射が視点に届く量が無視できるほど小さくなる距離である。

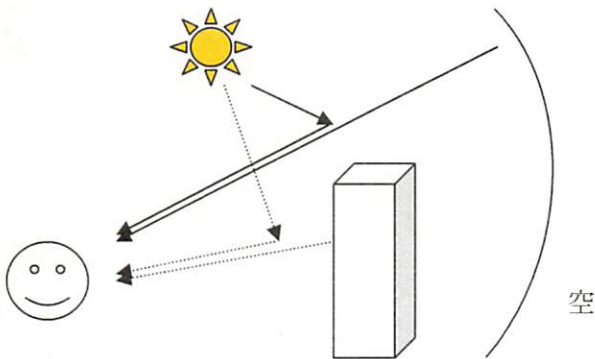


図2：視界に入る光線の種類

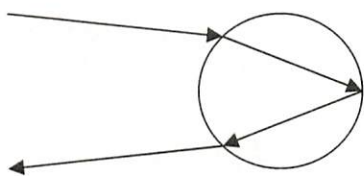


図3：水滴のモデル（主虹）

続いて虹の色を求める方法について述べる。図3は、水滴内部で光が1回のみ反射することで観察される主虹について、大気中の水滴内部の太陽光線の軌跡を現したものである。太陽光線が水滴内部に入射したとき、屈折率が波長によって変化するので、太陽光線と視線の偏角も波長により変化し、結果として分光して見えるのである。よって太陽光線と視線の偏角が与えられると、そこから反射光の波長を逆算し、その色を求めることができる。この仕組みは文献[1]にも示されている。

#### 4. 実行結果



図4：実行結果

図4は、視点の後ろから太陽光線が差し込んでいることを想定して、実写画像に虹を合成したものである。虹は手前の木の奥に分布する水滴によって発生しており、木と重なっている部分は木に隠れて見えなくなっている。

#### 5. まとめ

本報告では、実写画像にCGで生成した虹をリアルに合成する表現手法を提案した。

今後の課題として、以下の点が挙げられる。

- 太陽光線の方向をユーザが指定できるようにすること
- 画像内の様々な状況に対応できるようにすること
- k-menas 法を用いて分類する色数や、画像中の物体の奥行き値を、自動設定できるようにすること

#### 参考文献

- [1]伊藤, 牧野, 斉藤, 大石, 「レイ・トレーシング法における分光モデルに関する検討」, 日本コンピュータ・グラフィックス協会第6回 NICOGRAPH 論文コンテスト, pp. 45-50, 1990
- [2]芳信, 金田, 「波動工学に基づく虹のレンダリングと背景実写画像との合成」, 信学技報 ITS2004-77, IE2004-211, pp. 65-70, 2005
- [3]岩崎, 柳田, 土橋, 西田, 「散乱マップを用いた多重散乱の計算と空の色の高速表示法」, Journal of IIEEJ, Vol. 33, No. 4, pp. 515-522, 2004
- [4]米山, 近藤, 「発光原理を考慮したオーロラのビジュアルシミュレーション」, Visual Computing グラフィックスとCAD 合同シンポジウム' 05-19, pp. 111-116, 2005
- [5]師, 宮崎, 土橋, 西田, 「グラフィックスハードウェアを用いた空の色と大気散乱の高速レンダリング」, Visual Computing グラフィックスとCAD 合同シンポジウム' 05-4, pp. 17-22, 2005
- [6]P. E. Debevec, J. Malik: "Recovering High Dynamic Range Radiance Maps from Photographs", Proc. SIGGRAPH' 97, pp.369-378, 1997

## 任意の頭部形状モデルへの髪型のフィッティング

大瀧杏奈 (指導教官: 伊藤貴之)

### 1. 概要

コンピュータグラフィックス(CG)のための髪モデリングは、以前から活発に研究されている分野であり、既にさまざまな方法が提案されている。一般に一人の人間の頭には約10万本の髪の毛が生えており、そのデータ量は膨大なものとなる。このような膨大な数の毛髪の形状モデリングは非常に活発な研究課題である。

CGで自分の髪型をシミュレートしたい、という要求は以前からある。例えばモデルの髪型を真似してみたいとき、CGを使ってその人の髪型を自分の顔画像に当てはめられれば、その髪型が自分に似合うか確かめることができるのではないかという欲求がある。近年の髪モデリング技術により、仮にモデルの髪型がデータ化されていたとすると、それを自分の顔画像にフィッティングできれば、その要求は解決できるはずである。

しかし、任意の顔画像に任意の髪型を合成する場合、2次元画像上で合成するにしても、頭部形状の3次元モデル上で合成するにしても、もともとの髪形の持ち主と頭部形状が異なる場合には、その都度髪型を頭部形状に合わせなければならない。異なる顔の形へ髪を合わせた場合の不具合の改善について解決している研究は、現状では見当たらない。

そこで、本研究では任意の頭部形状モデルへの髪型のフィッティング手法を提案する。

本研究では、既存手法によって、ある人物の頭部形状に合わせて作られた髪型の3次元形状があることを想定する。そして、これを別の人物の3次元頭部形状にフィッティングさせる場合、起こる不具合を解決する事を目標とする。本研究を適用することで、元の髪型の大まかな形状を損なわずにフィッティング出来るようにする事ができれば、髪型の3次元形状の使い回しが可能になり、モデリングの反復作業の大幅な削減が期待できる。

ただし本研究では、髪型のフィッティングへの第一歩として、ある人の毛根パターンを別の人の頭部形状にフィッティングする技術の研究に着手した。

### 2. 関連研究

#### 2.1 髪型の3次元モデリングに関する基本手法

髪型を3次元形状としてモデリングする多くの手法は、以下の前提によって膨大な数の髪を表現している。

一般的に髪は、折れ線状のデータ構造で近似される場合が多い。典型的な例として、複数の線分や角柱を接続した形状で1本の髪(あるいは数本の髪の束)を表現する手法や、複数の三角形で(一方向に整列する)数本の髪を表現する手法がある。

また多くの手法では、力学や相互作用を考慮した計算手法により、各々の髪(または髪の束)の形状を決定する。最近の研究では、髪の毛自体にかかる重力や回転の動きはもちろん、髪

と髪との相互作用を考えた研究や、髪の流れを表現する手法や、カールやウェーブといった変わった髪形に対して、髪の流れに沿った流れ場を髪の周りに発生することによって形を決定するモデリングなどが発表されている。

#### 2.2 髪型の対話的な3次元モデリング手法

髪型の対話的なモデリング手法の代表的な手法として、「階層型のクラスター構造を用いた手法」[1]や「ペン入力による手法」[2]が挙げられる。[1]は髪を階層構造にすることにより、利用者が上位階層において髪の大局的な形状を決定したり、最下位階層において髪一本一本の形状を決定したりことができる。この入力方法により、簡単な入力操作で複雑な髪型の再現を可能にしている。[2]は利用者が頭皮面から任意の曲線を引き、その曲線に沿って円柱状に配された毛髪を発生させる。ペン入力のみで、髪の長さ、発生位置、巻き髪・縮れ髪といった髪の性質、散髪を可能にしている。

#### 2.3 実写画像からの髪や顔の3次元形状の再構成

実写画像から髪の3次元形状を再構成する方法としては[3]がある。これは光源を固定し、複数の視点から対象の髪の写真を撮影し、それぞれの髪の流れを三角測量することによって、髪の3次元情報を取り出すことを可能にしている。

また、実写画像から顔の3次元構造を再構築する手法として[4]などがある。

#### 2.4 顔と髪の合成

顔と髪の合成に関しては[5][6][7]があげられる。[5][6]は2次元画像データをもとにしたもので、目・鼻・口の位置から顔の向きを調べ、髪型と顔を合成する手法である。これらの手法は2次元の画像空間で操作を行うため、任意の方向から見た場合の画像を得るには、複数の視点からの髪型の補間を行う必要があり、また頭部形状が大きく違う人物へ合成する場合、形が崩れてしまう。[7]は1つのあらかじめ決まった顔のポリゴンに顔画像をテクスチャマッピングし、髪型を3次元形状で作成し合成したものである。この研究では頭部形状は単一モデルでしか考えられておらず、異なる頭部形状に合成した場合については、研究の余地がある。

### 3. 提案内容

本研究では、任意の頭部形状モデルへの髪型のフィッティングの簡便化を実現する手法を実現するための第一歩として、髪型の毛根パターンを任意の頭部形状モデルにフィッティングする手法を提案する。

#### 3.1 髪の束を六角柱で表現

髪の本数は非常に多く、1本1本の単位で扱うのは非現実的であるから、束で扱いたい。そこで本研究では髪の束を六角柱で近似する。六角形を用いた根拠は、頭皮を隙間無く覆うことが

できる図形のうち、六角形が最も円に一番近い図形であるからである。本研究では将来的に、この六角柱の外接円柱を用いることで、円柱で表現した髪の毛の衝突計算手法[11][12]が適用できると想定している。

また、毛根の側方抑制[7]により、髪の毛の密度はほぼ一定であると考えられることから、六角柱はできるだけ同じ大きさで、正六角形に近い形状で、かつ等密度で生成されることが望ましい。

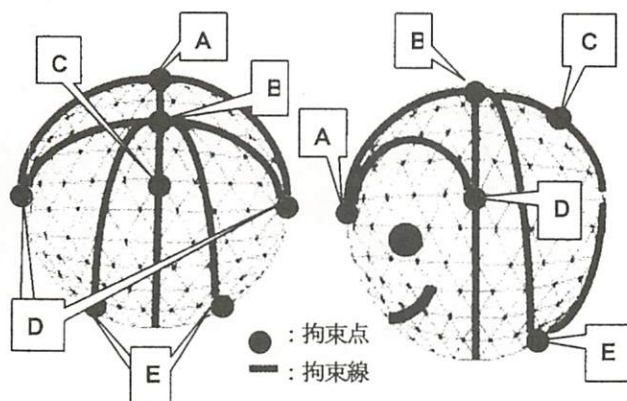
### 3.2 髪型を階層構造で表現

本研究では、髪を2段階の階層構造に格納する。上位階層は大まかな髪型の領域分けを表現し、下位階層は髪の毛束を表現するものとする[10]。任意の頭部形状に合わせて作られた髪型を別の頭部形状に移す際に、髪型のイメージを損なわないためには、この階層構造を保存したままマップする必要がある。

### 3.3 別の頭部形状へマッピングする際の拘束条件

本研究では、頭部形状を三角形ポリゴンで表現する。また六角柱の生成パターンを表現するために、本研究では毛根群が形成する曲面も三角形ポリゴンで表現し、その双対を求めることで六角形パターンを生成する。

3.2節で説明した条件を満たすためには、上位階層を構成する各領域の境界線を保存し、別の任意の頭部形状上に定義された拘束線と一致させる必要がある。この拘束点・拘束線とは、毛根集合を頭部形状にマップする際に、両者の対応を決定する点・線である。ある頭部形状Aに合わせて作った髪形形状を別の頭部形状Bへマッピングする際、何も拘束点を設けなかった場合全体的に縮小・拡大されて、元の髪型の形を大きく損ねる場合がある。このため、実際に美容師が髪を切る場合の頭部の領域の分け方を参考にし、拘束線・拘束点を設定する(図1参照)。



【図1】拘束線・拘束点の位置

A:前中心 B:トップポイント C:バックポイント D:耳 E:三弁

本研究では、この拘束線・拘束点が一致するように六角形群のマッピングを行う。マッピングの際には、毛根集合の三角形領域を毛根が分布する頭部曲面を2次元座標系に射影する。この射影は、次節で述べる最適化の際に、髪型形状と頭部形状の座標系を統一するために行われる。この処理は3次元座標系よりも2次元座標系のほうが曖昧性を生じにくい点で有利である。

### 3.4 髪の毛の大きさ・密度の最適化

3.3節で説明したように別の頭部形状へ髪型形状をマッピ

ングする際に、3.1節で説明した毛根集合を表す六角形は変形され、髪の毛の疎密差が生まれてしまう。このため、毛根集合の六角形を再分割・併合する必要がある。

この再分割・併合の簡単化のために、六角形群の双対である三角メッシュを再分割・併合する。本研究では、

- ・ 髪の毛束の本数をモデルと出来るだけ同じにする
- ・ 1つの髪の毛束の大きさを出来るだけ保存する

のいずれかを条件を満たすように、三角形の辺の長さの基準値を設定し、この基準を満たさない三角形を再分割・併合する。

## 4. まとめ

本研究では、任意の頭部形状モデルへの髪型のフィッティング手法を提案した。実行結果および今後の課題については、当日の口頭発表にて説明する。

## 謝辞

研究の全般的なアドバイスを東京大学 大学院情報学環・学際情報学府山口泰教授から頂きました。感謝の意を表します。

## 参考文献

- [1] Tae-Yong Kim, Ulrich Neumann, Interactive multiresolution hair modeling and editing, ACM Transactions on Graphics (TOG), Proceedings of the 29th annual conference on Computer graphics and interactive techniques, Volume 21, Issue 3, July 2002
- [2] Shazad Malik, A Sketching Interface for Modeling and Editing Hairstyle, EUROGRAPHICS Workshop on Sketch-Based Interface and Modeling, August 2005.
- [3] Yichen Wei, Eyal Ofek, Long Quan, Heung-Yeung Shum, "Modeling Hair from Multiple Views", In Proc. of SIGGRAPH conference, ACM, pp. 816 - 820, 2005
- [4] Yuencheng Lee, Demetri Terzopoulos, and Keith Waters, "Realistic Modeling for Facial Animation", In Proc. of SIGGRAPH conference, ACM, pp. 55-62, 1995
- [5] 富田 一乗: "画像のワーピング処理を用いたヘアスタイルの合成法", *The Journal of The Institute of Image Electronics Engineers of Japan*, 1996
- [6] 中野敦, 星野准一: "対話型仮想ファッションシステム", 第17回 NICOGRAPH 論文集, pp. 155-160, 2001
- [7] Lieu-Hen Chen, Santi Saeyor, Hiroshi Dohi, Mitsuru Ishizuka, "A System of 3D Hair Style Synthesis based on The Wisp Model", *Journal of Visual Computer* Vol.15 No.4, 1999
- [8] 松崎 貴: 最新の毛髪科学, 毛髪科学技術者協会
- [9] 北川照男, 予防医学ジャーナル 2004年7月15日号 第398号
- [10] Tae-Yong Kim, Ulrich Neumann, Interactive multiresolution hair modeling and editing, ACM Transactions on Graphics (TOG), v.21 n.3, July 2002
- [11] Byoungwon Choe, Min Gyu Choi, and Hyeong-Seok Ko, Simulating Complex Hair with Robust Collision Handling, In ACM SIGGRAPH / Eurographics Symposium on Computer Animation, pp. 153-160, July 2005
- [12] F. Bertails, T-Y. Kim, M-P. Cani, U. Neumann, "Adaptive Wisp Tree - a multiresolution control structure for simulating dynamic clustering in hair motion". Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pp 207 - 213

# 印象の近い音楽と画像の組み合わせを自動選出する手法

小田瑞穂 (指導教官: 伊藤貴之)

## 1. 概要

MIDI や MP3 などのデジタル音楽記録形式の普及により、近年では計算機上での音楽鑑賞が非常に普及している。また一方では、聴覚だけでなく視覚に訴える計算機技術も非常に発達している。そこで聴覚だけでなく視覚にも同時に刺激を与えるような音楽提供方法により、さらに直感的に音楽に親しむことが可能になると考えられる。

そこで本研究では、音楽情報を与えると、コンピュータがその音楽の印象を推測しそれに近い印象の画像を選択し、同時に提示する手法を提案する。本研究では画像や音楽の特徴を自動抽出して数値化し、その特徴から画像や音楽の印象を推測する。この推測のために本研究では、人間による印象も数値化し、特徴から印象を求める数式を用いる。この数式の決定には応答曲面法を用いる。

この研究成果は例えば、曲ごとにアイコンを作り、多くの曲を一覧表示する、という目的に応用可能であると考えられる。また、本報告で提案する手法の原理は、音楽から画像を選ぶだけでなく、画像から音楽を選ぶ、という逆の目的にも応用が可能である。

## 2. 関連研究

大山らは画像に印象の合うように音楽を自動アレンジする手法を提案している[1]。また印象に基づいて楽曲を検索するシステムについての研究がある[2]。しかし、この研究では画像と音楽の組み合わせは実現していない。また、類似度の高い曲に類似度の高い色を割り当てるインタフェースの研究[3]も存在している。

## 3. 提案内容

本研究では、「特徴値」「評価値」という2種類の数値をもって、画像および音楽を評価する。ここで特徴値および評価値の定義は以下のとおりである。

**【特徴値】** 画像および音楽の特徴を表す値で、コンピュータによる自動抽出が可能なもの。

**【評価値】** 画像または音楽に対して、人間による主観的な印象を表す値。本研究では、画像や音楽の印象を表

す数種類の形容詞を用意し、その形容詞の各々に対して適用度を主観評価させた値を評価値として用いる。例えば「明るい」という形容詞に対し、与えられた画像や音楽の適用度を0から10までの11段階のいずれかの値で主観評価した値を用いる。よって本研究では、用意した形容詞の数だけの評価値が用いられる。なお本研究では、画像と音楽の両者に対して同じ形容詞を用いるものとする。

ここで特徴値と評価値を、以下のように定式化する。

画像の項目の特徴値を  $P(p_1, p_2, \dots, p_l)$  とする。

音楽の項目の特徴値を  $M(m_1, m_2, \dots, m_m)$  とする。

両者共通の評価値を  $A(a_1, a_2, \dots, a_n)$  とする。

(ただし  $l, m, n$  は正整数)

$P$  から  $a_k$  を求める数式を  $a_k = f_k(P)$  とする。

$M$  から  $a_k$  を求める数式を  $a_k = g_k(M)$  とする。

関数  $f$  と  $g$  は応答曲面法を使って求める。応答曲面法とは、 $a_k$  と  $x_i$  を与え、その点群の近傍を通るような曲面を生成できるように係数  $\beta_0, \beta_i, \beta_{ij}$  を決定する方法である。本報告では関数  $a_k$  を、応答曲面法による以下の数式により表現する。

$$a_k = \beta_0 + \sum \beta_i x_i + \sum \beta_{ii} x_i^2 + \sum \beta_{ij} x_i x_j$$

$a_k$ : 人間による画像や音楽の印象の、 $k$  番目の評価値

$x_i$ : コンピュータが計算する画像( $P$ )や音楽( $M$ )の特徴

本研究では最初に準備段階として、 $s$  個のサンプル音楽と  $t$  個のサンプル画像を用意する。これらを実験者に提示し、評価者がもつ印象を数値化することで、評価値  $a_1 \sim a_k$  を決定する。つまりここには  $(s+t)$  個の評価値が存在することになる。

続いて  $s$  曲のサンプル音楽について、特徴値  $M(m_1 \sim m_m)$  を求める。この結果を以下の式

$$a_1 = g_1(M)$$

:

$$a_k = g_k(M)$$

に代入し、 $g_1 \sim g_k$  の各々に対して応答曲面法を適用し、係数  $\beta_0, \beta_i, \beta_{ij}$  を確定する。

同様に、 $t$  枚のサンプル画像について、特徴値  $P(p_1 \sim p_m)$  を求める。この結果を以下の式

$$a_1 = f_1(P)$$

:

$$a_k = f_k(P)$$

に代入し、 $f_1 \sim f_k$  の各々に対して応答曲面法を適用し、

A method for automatically coupling images and music those bring similar impressions.

Mizuho Oda, Takayuki Itoh

Ochanomizu University

{miz-oda, itot}@itolab.is.ocha.ac.jp



係数  $\beta_0, \beta_i, \beta_{ij}$  を確定する。

以上により  $a_1 \sim a_k$  を求める数式が完成する。これ以降、新しい音楽データや画像データを追加しても、 $a_1 \sim a_k$  を求める数式を用いることで、計算機は画像や音楽の印象を推測できる。

表 1. 特徴値と評価値の一覧

画像 P( $p_1, \dots$ )	評価値 A( $a_1, \dots$ )	音楽 M( $m_1, \dots$ )
色相	明暗	休符含有率
彩度	静動	平均音程
明度	重軽	テンポ
面積	攻防	単位時間当音数
要素数	華素	
	幸悲	
	硬柔	
	複単	

以下に具体的な特徴値と評価値を挙げる。特徴値は音楽や画像から数値として取り出しやすくするため、音楽は MIDI を、画像は SVG を使い、そこから数値として取り出しやすいものを特徴値として抜粋した。評価値は SD 法 (Semantic differential method) [4] が提示する形容詞群の中からいくつかを抜粋し、これについて評価者に採点させた結果を用いている。本研究で用いた特徴値と評価値の一覧を、表 1 に示す。

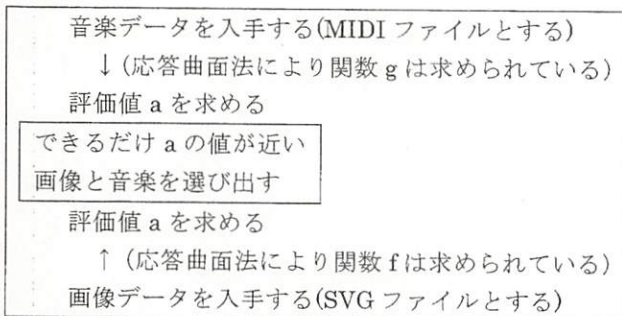


図 1. 印象の近い画像と音楽の選出

本研究では、評価値をベクトルであるとみなし、その距離が小さいものを「印象が近いもの」と判断する。例えば A の項目が 3 個のとき、画像の評価値を  $A_1(a_{11}, a_{12}, a_{13})$  とし、音楽の評価値を  $A_2(a_{21}, a_{22}, a_{23})$  と仮定する。3 次元座標系で  $A_1$  と  $A_2$  を座標にプロットするし、印象の近いものを選ぶには 2 点の距離が最小である点を探し出せばよいことになる。2 点間距離は  $(a_{11}-a_{21})^2 + (a_{12}-a_{22})^2 + (a_{13}-a_{23})^2$  によって求められるので、距離が最小の画像を選び出せばよい。A の項目を 3 項目から  $n$  項目に拡張したときには、 $\sum_{k=1}^n (a_{1k}-a_{2k})^2$  の値が最小になる組み合わせを与える画像を選べばよい。以上の処理をまとめると図 1 のようになる。

#### 4. 実行結果

まず準備段階として、サンプル画像とサンプル音楽に対して特徴値と評価値を求め、この値から散布図を生成した。この散布図から、評価値に対応する各々の印象語について、相関性の強い特徴値を画像と音楽に対して視覚的に選択した。この結果を表 2 に示す。

この結果を用いて、各々の評価値ごとに応答曲面法を適用して数式をつくり、評価値の距離の近い画像と音楽を選択する。現時点では、画像、音楽の両方の特徴値を 2 つ以上適用可能な、上から 4 項目だけについて応答曲面法を適用した。選択結果とその評価については講演にて論じる。

表 2. 相関性の強い特徴値の視覚的な選出結果

画像の特徴値	評価値	音楽の特徴値
明度, 彩度, 面積	静的な ⇔ 動的な	音程, 音数
明度, 彩度, 数	重い ⇔ 軽い	テンポ, 音程
明度, 彩度, 面積	攻撃的 ⇔ 保守的	音程, 音数
明度, 彩度, 面積	華やか ⇔ 素朴な	テンポ, 音程, 音数
彩度	硬い ⇔ 柔らか	休符含有率
彩度, 明度	明るい ⇔ 暗い	テンポ

#### 5. まとめ

本研究では、印象の近い音楽と画像を組み合わせる手法を提案した。今後の課題として、表 2 の下 2 項目についても応答曲面法を適用できるようにしたい。また、音楽については、数値以外の特徴 (例えば音色など) や、1 曲の中での曲想の変化などを考慮できるようにしたい。画像については、静止画だけでなく CG アニメーションなども対象にしたい。

#### 謝辞

応答曲面法は京都大学小山田耕二教授からいただきました。この場をお借りして御礼申し上げます。

#### 参考文献

- [1] 大山, 伊藤, DIVA: 画像の印象に合わせた音楽自動アレンジの一手法の提案, 第 68 回情報処理学会全国大会講演予定, 2006.
- [2] 熊本, 太田 印象に基づく楽曲検索システムにおける程度語の理解 The 18<sup>th</sup> Annual Conference of the Japanese Society for Artificial Intelligence, 2004
- [3] 後藤, 後藤, Musicream: 楽曲を流してくっつけて並べることのできる新たな音楽再生インタフェース, WISS2004, pp.53-58, 2004.
- [4] Osdoog, C.E., Suci, G.J. Tannenbaum, P.; The measurement of meaning, Univ. Illinois Press, (1957)

## DIVA: 画像の印象に合わせた音楽自動アレンジの一手法の提案

大山喜冴 (指導教官: 伊藤貴之)

### 1. 概要

映画やテレビCMの制作において、画像と音楽は密接な関係にある。また例えば個人でも、「自分で制作したホームページに印象の合った音楽を載せたい」という感想は、多くの人が一度は持つような感想だろう。しかし音楽に精通していない人は、画像に印象の合う音楽を選べない場合があり、画像と音楽の相関性のない一見不釣合いなホームページをつくる場合が多い。

最近では画像に合った音楽を検索できるシステムの研究[1]や、画像に合った音楽の自動作曲システムの研究[2]が盛んに行われている。しかし、入力画像の雰囲気マッチする音楽が、必ずしもシステムに登録されているとは限らない。また、このシステムにおいて選曲された楽曲や自動作曲された音楽が、ユーザーの好きな作曲家・演奏家による楽曲、ユーザーの好みのメロディである保証はない。

この問題を解消するために我々は、ユーザーが任意の音楽と画像を入力した際に、画像の印象に合わせて音楽を自動アレンジする手法の研究を進めている。本研究は現時点では、画像から色分布を算出し、その色分布から連想されるリズムパターンを導き、そのリズムパターンを音楽に適用して自動アレンジする。

本手法ではまず前処理として、被験者に質問を課し、その被験者が所定の色から所定のリズムパターンを連想する度合いを「連想度」として数値化する。続いて任意の画像を入力し、その画像中における所定の色的重要性度を算出し、それにリズムパターン連想度を乗じ、これを累算する。この処理により、所定のリズムパターンの入力画像からの連想度を算出する。この連想度が最大であるリズムパターンを用いて、入力音楽を自動アレンジする。

### 2. 関連研究

音楽の持つ感性と画像の持つ感性をマッチングにする方法には、多様なメディアデータに対し書誌情報による通常の検索に加え感性検索も可能とするマルチメディア感性データベース管理システムの研究[3]や言語情報と画像情報のマッチングし、自由な表現が可能な自然言語を検索キーとした言語・画像型のシステムの研究がある[3]。楽曲からリズム認識をする研究[4]、楽曲からメロディを抽出する研究[5]も本研究の楽曲を解析する際に関連付けられそうである。

しかし、画像の印象に基づいて音楽を自動アレンジする研究は今のところ見つかっていない。

### 3. 提案内容

論文[6]では、音楽の印象と色彩の印象に相関性があることを実証する実験結果が示されている。例えば色相と音楽の印象の相関性には、「赤には迫力のある音楽が似合う」「緑には明るい音楽が似合う」という回答が多いとの結果が出ている。また、明度や彩度と音楽の印象の相関性には、「明度が高くなるに連れて音楽を明るく感じる傾向にある」「彩度の高低と音楽の力強さの度合いが対応する傾向にある」という回答が多いとの結果が出ている。このことから画像の持つ「色」が音楽に与える影響は大きいと考えられる。

また文献[7]においても、画像の持つ印象は主に配色が重要である、という実験結果が報告されている。このことから本研究では、画像の特徴のうち配色に着目する。画像からの他の特徴(例えば画像中の物体形状)の抽出は、現時点では未着手である。

一方、楽曲の構成要素には「調(長調/単調)」「テンポ」「旋律(上昇/下降)」「音高(高/低)」「和声(単純/複雑)」「リズム(固定/流動)」などがある。文献[4]では、この構成要素の中でも「リズム」→「旋律」→「和声」→「音高」の順で印象に残りやすく、楽曲を特徴付ける大きな要素となっていると述べている。このことから本研究では、音楽の構成要素のうちリズムパターンに着目してアレンジを行う。他の構成要素は、現時点ではアレンジには用いていない。

以上の背景により本研究では、入力画像中の色分布を用いて、画像からの連想度が高いと推定されるリズムパターンを特定し、これを音楽に適用させて自動アレンジすることを考える。

まず本研究では準備段階として、被験者に十分に多くの色を提示して、各々の色に対して印象の近いリズムを選択させる。ここで回答対象となる色の制定方法について述べる。まず、色特徴の解釈手法は HSV 色空間を用いることにする[8]。本研究では現段階では 68 色を被験者に提示している。68 色の内訳は以下の通りである。有彩色については、太陽光をプリズムで虹色 7 色に分割すること[9]から 7 段階に、彩度および明度を 3 段階に分類する。無彩色は 5 段階に分類する。以上の処理により本研究では、被験者に提示する 68 色を得る。本研究ではこの 68 色と同時に、複数のリズムパターンを被験者に提示し、各色に対するリズムパターンの連想度を回答させる。本研究では、 $i$  番目の色に対する  $j$  番目のリズムの印象の近さを  $R_{ij}$  と記述する。

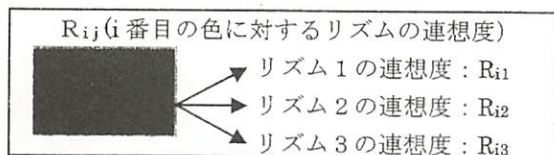


図1: 68色それぞれが持つリズム連想度

続いて、画像を68色に分類し、その画像中における68色の重要性の割合を算出する。ここで色重要度は

- ・ 画像中に占める面積
- ・ 隣り合う色との差分
- ・ 色自体の持つ印象

等により算出される、その色の画像中における印象の強さを意味する。

本研究では、ある画像中における  $i$  番目の色の重要度を  $C_i$  と記述する。このとき、この画像に対する  $j$  番目のリズムパターンの連想度は、式(1)で表現される。

$$C_1R_{1j} + C_2R_{2j} + \dots + C_{68}R_{68j} \dots (1)$$

本研究では各々のリズムパターンに対して、式(1)を用いて連想度を算出する。そして、この値が最も大きいリズムパターンを、入力画像に最も印象の近いリズムパターンであると判断し、自動アレンジに用いる。

#### 4. 実行結果

本研究ではまず準備段階として、同じメロディ A に対して異なる 7 種類のリズムパターンを適用した 7 曲を、被験者に鑑賞させた。同時に所定の 68 色を被験者に提示し、各々の色からどのリズムパターンを連想したかを回答させ、この回答結果から値  $R_{ij}$  を得た。同様に上記の実験で行った楽曲とは違うメロディ B に対して、メロディ A と同様に 7 種類のリズムパターンを適用した 7 曲と画像 25 枚を提示し、各々の画像からどのリズムパターンを連想したかを選択させた。25 枚の画像は様々な風景画像を用い、メロディ A, B は著者の一人である伊藤によって作曲された楽曲を使用した。アンケート回答者は男女 7 人である。

続いて本研究では、この 25 枚の参照画像について、以下の 2 種類の方法によって自動選択されたリズムパターンと、準備段階で被験者が選択したリズムパターンとを比較した。

[方法 1] 所定の 68 色の各々が占める重要度  $C_i$  を算出し、式(1)を用いて各々のリズムパターンの連想度を算出し、連想度が最大であるリズムパターンを自動選択。  
[方法 2] 25 枚の画像から連想されたリズムパターンから 68 色が持つリズムパターン連想値  $R_{ij}$  を逆算して求め、式(1)を用いて各々のリズムパターンの連想度を算出し、連想度が最大であるリズムパターンを自動選択。

この実験では各々の被験者の回答から導いた  $R_{ij}$  値の傾向が被験者ごとにあまり似通っていないことに対し、特定の画像から選ばれたリズムパターンは非常に似通っていた。また、68 色から選ぶ場合でも画像より選ぶ場合でも、リズムパターンの選ばれる頻度に非常に偏

りがあった。特に 68 色からリズムパターンを選ぶ場合に偏りが多く、うまく実験ができないことがあった。全くあまり選ばれないリズムパターンはメロディと印象の合わないものである場合が多かった。[方法 1]における一致率は現時点で 44%であり、[方法 2]における一致率は 76.8%であり、色よりも画像を提示するほうが準備段階の方法として有効である傾向が現れた。一致率の高い画像は彩度や明度の低い色の面積が多い画像であった。反対に明度・彩度が高い色が多く使われている画像では一致率が極端に低かった。

#### 5. まとめ

本論文では、画像から色分布を求め、その色分布から最も連想されるリズムパターンを用いることで、画像に印象が近くなるように音楽を自動アレンジする手法を提案した。

今後の課題として

- ・ 画像からの配色以外の特徴抽出
- ・ リズムパターン以外の構成要素を変化させる音楽の自動アレンジ
- ・ 楽曲の持つ雰囲気も考慮し、より多彩な楽曲にアレンジができるようなリズムパターンの作成
- ・ 学習アルゴリズムなどの適用により、よりユーザーの好みを反映するシステムの研究開発なども考慮していきたいと考えている。

#### 謝辞

本研究の被験者の方々に感謝の意を表します。

#### 参考文献

- [1] 古賀、下塩、画像に合った音楽の選定技術、ヒューマンコミュニケーション基礎研究会技報、平 11-9、1999.
- [2] 佐藤、英画像から音楽を自動演奏「ピクチャーメロディー」v1.2
- [3] 坂井、大塚、宮崎、マルチメディア感性データベース YAMAKAN、第 13 回データ工学ワークショップ (DEWS2002)、2002.
- [4] 感性(印象語)語による検索  
<http://www.slis.keio.ac.jp/~ueda/semi/99onsei.html>
- [5] 武田、西本、嵯峨山、確率モデルによる多声音楽演奏の MIDI 信号のリズム認識、情報処理学会論文誌、Vol. 45, No. 3, pp.670-679, March, 2004
- [6] 安達、岩宮、色彩と音楽が互いに及ぼす影響--ショパンのエチュードを手がかりに、第 5 回学生のための研究発表会講演論文集(日本音響学会九州支部)、pp.13-16、2003
- [7] 北島、土居、画像の構成を手掛かりとした音楽の検索、平成 15 年度情報処理学会関西支部支部大会 講演論文集、pp39-42、2003.
- [8] 原田、感性語句を用いた自然言語文による画像データベースの対話的検索、静岡大学博士論文、工博甲第 175 号、1995.
- [9] 大林、銀河の道 虹の架け橋、小学館、1999.

## 大解像度画像からの類似部分画像の高速抽出 ～"VIEWGLE"の開発～

五味愛 (指導教官: 伊藤貴之)

### 1. 概要

インターネット、デジタルカメラやカメラつき携帯電話の普及に伴い、今では誰もがデジタル画像を作成することが容易となった。大量の画像情報を活用できる環境の増加に伴い、それらの膨大なデータを効率的に検索することに対する需要はますます高まっている。同じ画像サイズの写真データを比べる類似画像検索は研究が盛んに行われている。一方で、小解像度画像に写る物体の形、色を把握し、大解像度画像の特定の一部分にその物体を自動発見するという検索に関する研究には、まだ議論の余地がある。

本研究は既存の画像認識技術を応用し、大解像度画像の中から、入力画像に類似する部分画像を高速に自動発見する一手法を提案する。

### 2. 関連研究

特定領域の画像のみを対象とする画像検索システムとして、有名なものは QBIC[1] などがある。QBIC は初めての商用画像検索システムであり、類似画像やユーザのスケッチによって検索することができる。

最近の画像検索では、色に関する特徴量抽出結果を用いる手法[2]、色と構図から特徴量を抽出した手法[3]がある。また画像の境界線抽出結果を用いる手法として、ウェーブレット変換を用いる手法[4]も発表されている。ただし、これらはどれも同じ画像サイズの写真データを比べる類似画像検索の研究である。

大解像度画像から類似部分画像を抽出する研究としては、指定した人間の顔を高速抽出する研究事例がある[5]。

### 3. 提案内容

本研究は、大解像度の写真データの中から、ユーザの求める一部分を高速かつ的確に検索するシステムを提案する。

#### (1) 概要

大解像度画像の中から、検索したい物体が写っている部分を切り取る、という問題を以下のように定式化する。

1. 入力画像を  $I$ 、大解像度画像を  $I'$  とする。
2.  $I'$  から切り取った長方形領域を  $I''$  とする。このとき  $I''$  を、 $I'$  の左から  $x$  画素目、上から  $y$  画素目、を左上端とし、幅  $w$  画素、高さ  $h$  画素のサイズを有する長方形領域とする。
3.  $I$  から特徴量を抽出し、 $I'$  の中から類似物体が存在しそうな場所を切り取り、 $I''(x,y,w,h)$  をいくつか生成する。 $I$  との画像類似度を判定し、類似度が高い  $I''$  を出力結果として提示する。

このとき  $I''$  の抽出のために、 $x,y,w,h$  に対して想定しう

る全ての値を用いていたのでは、膨大な処理時間がかかる。そこで次のようなプロセスによって、高速に適切な  $I''$  を探し出すことを考える。

1. ステップ 1 では、 $I'$  の中から、 $I$  に対して類似度の高い長方形領域  $I''$  が存在する可能性のある部位を、高速に絞り込む。

2. ステップ 2 では、ステップ 1 で絞り込んだ部位において、複数種類の  $I''$  を切り出して  $I$  との類似度を判定し、類似度が最大である  $I''$  を少ない反復回数で探し出す。

#### (2) 検索画像に適したアルゴリズムの自動選択

画像類似度判定アルゴリズムはすでにたくさんの手法が提案されている[1]～[4]。これらの手法をステップ 1 とステップ 2 で、また画像の性質によって使い分けることで、高速な画像検索を可能にする。現段階ではどのアルゴリズムがどのステップ、またどのような画像に適するのか、実験を進めている最中である。現時点では[4]に記述された手法の一つを用いている。この手法ではまず、K-means 法で入力画像および大解像度画像に減色を施し、減色結果が同一である隣接画素をグループ化してラベリングする。ラベリング結果として得られる各々の領域に対して重心を算出し、任意の 3 個の重心点を結ぶ三角形を生成する。この三角形を比較し類似度の高い三角形部分を抜き出し、画像類似度判定を行い、類似度の高い  $I''$  を見つける。

#### (3) 検索時の画像分割の最適化

(2) で説明した画像類似度判定アルゴリズムを前提にして、ステップ 1 および 2 の詳細を説明する。

ステップ 1 では、 $I$  をラベリングして得られる領域について、面積が上位 3 位までのものを選び、その重心を結んで三角形  $T$  を作る。そこで抽出した色情報から  $I'$  でも同じ色の組み合わせをもった三角形  $T_i$  を生成する。三角形  $T$  と  $I'$  でできたそれぞれの三角形  $T_i$  のベクトルを比較し、三角形間の類似度を算出する。

ステップ 2 では、ステップ 1 において類似度の高い三角形の存在する部位の周辺において、 $(x,y,w,h)$  の値を少しずつ変えながら複数の  $I''$  を生成し、各々について  $I$  との類似度を判定する。

ここで  $I''$  を生成する際の  $(x,y,w,h)$  の変化量が大きすぎると、類似度が最大となる部分画像を抽出しそこなう可能性が生じる。例えば  $I''$  の生成のために  $x=x_1, x=x_2 (x_1 < x_2)$  という 2 つの  $x$  座標値を採用した際に、実際には  $x=(x_1+x_2)/2$  の位置で  $I''$  を生成するのが最適である、ということが起こりえる。かといって、想定しうる全ての  $x$  値を対象として比較していたのでは、膨大な時間がかかってしまう。

そこで、最適な $I'$ を発見するために、最適化問題の目標関数等に近似手法を用いた手法の一つである応答曲面法を用いる。応答曲面法とは、設計変数と状態量の間を何点かの解析もしくは実験結果を用いて近似した関数である。一般的には、解析や実験点のパラメータ設定に実験計画法が使われ、関数の近似に最小二乗曲面が使われる。応答関数として2次多項式を用いた場合、応答曲面は式(1)のようになる。

$$y = \beta_0 + \sum_{i=0}^n \beta_i x_i + \sum_{i=0}^n \beta_{ii} x_i^2 + \sum_{i < j} \beta_{ij} x_i x_j \quad (1)$$

これを本研究の問題におきかえると以下のようにになる。まず実験計画法を用いて、少数の $(x, y, w, h)$ 値の組み合わせ $I'$ を類似度判定の回数分だけ用意をする。この各々の $I'$ に対して画像類似度を数値化し、式(1)に代入する。ただし式(1)において、 $y$ は類似度であり、 $x_0 \sim x_3$ は $(x, y, w, h)$ の各数値である。これを連立方程式として、これらの最小二乗曲面を生成することにより、 $\beta$ の各値を算出する。この最小二乗曲面は一般的には画像類似度が最大となる1点をもつので、その点における $(x, y, w, h)$ を算出することで、類似度が最大となる部分画像を推定する。この処理を反復することで、少ない回数の画像類似度判定により、高速かつ的確に、 $I$ に類似する画像 $I'$ を抽出できる。

#### 4. 実行結果



図1：(左)入力画像 $I$  (右)大解像度画像 $I'$

本手法の実行例として、図1に入力画像 $I$ および大解像度画像 $I'$ の例を、また図2に、本手法によって発見された類似部分画像 $I''$ の例を示す。筆者らの実装では、 $I'$ で生成された三角形54,621個のうち、類似度の高い三角形11個をステップ2へ進めることができた。



図2：類似部分画像 $I''$

この結果により、本手法の適用により、大幅な判定回数削減が可能となり、的確な類似部分画像を得られていることがわかる。ただし現時点での筆者らの実験では、応答曲面法を用いる時に $w, h$ の値を固定し、 $(x, y)$ の値の組み合わせだけを用いている。今後は $w, h$ も変数に加えて最適化することで、更に精度を向上したい。

#### 5. まとめ

本報告では、大解像度画像の中から、入力画像に類似する部分画像を高速に自動発見する一手法を提案した。

今後の課題として、画像類似度判定の精度と速度を上げ、さまざまな画像を使い、実験を繰り返し改良したい。

本研究の最終目標は、入力画像の撮影時視線方向を高速かつ的確に検索するシステムの構築である。このシステムは例えば、「最も人気のある記念写真アングル」を検索するシステムとして実用できると考えられる。筆者らはこのシステムを“VIEWGLE”と命名し、開発を進めている。

#### 謝辞

京都大学小山田教授から、応答曲面法のプログラムを提供して頂きました。本大学増永教授および増永研究室M1隅田氏から、地理情報処理に関する情報を提供して頂きました。

#### 参考文献

- [1] C. Faloutsos, R. Barber, M. Flickner, W. Niblack, D. Petkovic, and W. Equitz. Efficient and Effective Querying by Image Content. *Journal of Intelligent Information Systems*, Vol. 3, No. 3, pp. 231-262, 1994.
- [2] 橋本、色彩情報に基づく画像検索システム、<http://chihara.aist-nara.ac.jp/people/95/masano-h/research-j.html>
- [3] 谷田川、色と構図に基づく画像検索ブラウザの提案 東京大学大学院平成14年度修士論文, 2002.
- [4] 福見、知的画像検索のためのニューラルネットへの知識埋め込みとルール抽出、電気通信普及事業団研究調査報告書、2003、[http://www.taf.or.jp/publication/kjosei\\_18/pdf/071.pdf](http://www.taf.or.jp/publication/kjosei_18/pdf/071.pdf)
- [5] T. Hattori, H. Kitajima, T. Yamasaki, Face Pattern Recognition And Extraction From Multi Persons Scene, *Proceedings of ICEIS 2003 (Fifth International Conference on Enterprise Information Systems)*, ACM, AAAI and IEEE, pp.92-99, 2003.

## 半透明複数等値面の重ね合わせの高速化

平野未来 (指導教官: 伊藤貴之)

### 1. 概要

ボリュームデータの可視化の重要な適用分野に、流体と人体がある。流体の可視化は、雲や波に代表される気体や液体のシミュレーション結果として出力される、複数のスカラ値やベクトル値の分布をわかりやすく表示する研究分野である。また人体の可視化は、CTスキャンで得た医療画像の輝度値から、臓器や体表の形状をわかりやすく表示する分野である。

本研究では半透明複数等値面を使い、ボリュームデータのスカラ値分布の可視化に焦点を当てる。

等値面とはスカラ値が一定であるような点の集合で、一般的には曲面となる。ボリュームデータの多くの情報を1画面に表示する手段として、半透明の等値面を複数重ねて表示する手法がある(図1(左)参照)。この手法はボリュームレンダリングに近い大局的な表現ができる上に、ボリュームレンダリングに比べ計算コストが小さいので、リアルタイムレンダリングに向いている。

ボリュームレンダリングは、ボリュームデータ全体の情報を1画面に表示する可視化手法の1つである。流体力学シミュレーションの結果を研究者が理解する為の画像だけでなく、リアルなシーンを表現する画像を生成することもできる。しかし半透明複数等値面と比べて、1枚の画像生成のためにボリューム全体を処理するため計算コストが大きくなる。また効果的な画像を生成するための伝達関数の設定に、熟練を要することが多い。

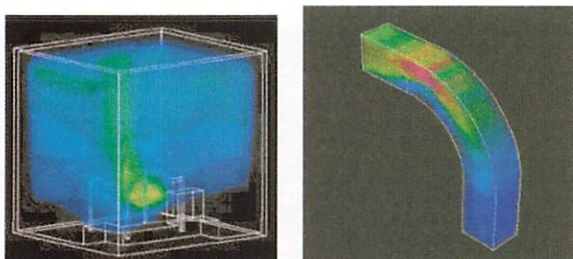


図1 (左)半透明複数等値面の画像例  
(右)ボリュームレンダリングの画像例

本研究は半透明複数等値面の高速生成を目的とする。この解決手段として本報告では、複数の等値面を同時に生成でき、かつ等値面を構成する点群を視点の奥から順に生成できる手法を提案する。本手法ではボリュームを構成するエレメントを、スカラ値の最小値および最大値、視点からの距離、の3値を座標軸とする空間に配置し、この空間上で等値面と交差するエレメントを高速検索することにより、解決手段を実現する。

### 2. 関連研究

等値面の高速生成手法に、ISSUE[1]という手法が提案されている。ISSUEでは、エレメントの各ノードのスカラ値の最小値(Smin)を横軸に、最大値(Smax)を縦軸に取り、全てのエレメントを2次元座標系にプロットする。指定したフィールド値sの等値面になりうるエレメントは図2の左上の塗りつぶされた領域上にある。

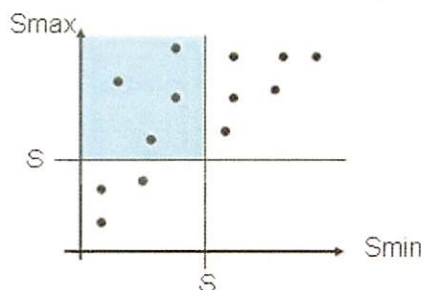


図2. ISSUEの原理

しかし、このISSUEを用いて複数の半透明等値面を表示するには、以下の3ステップを行う必要がある。

1. 複数の等値面を生成し、一時蓄える。
2. 半透明表示を実現するために、等値面を構成する点群を、視点からの距離の遠い順に並び替える。
3. 視点からの距離の遠い順に点群を描画する。

### 3. 提案内容

本研究では、半透明複数等値面の生成の高速化を狙い、ISSUEで必要だった3ステップを、以下の2ステップにする。

1. 複数の等値面を生成する際に、等値面を構成する点群を、視点から遠い順に生成する。
2. 生成した順に点群を描画する。

その為に必要な機能は、

- 複数の等値面を同時に高速生成できるようにすること
- 視点からの距離が遠い順に点群を生成できるようにすること

である。以下に、この機能を実現するためのアルゴリズムを説明する。

#### 【複数等値面を同時高速生成する方法】

複数等値面を同時に生成できるように、ISSUEを拡張する。例えば、フィールド値がs1、s2、s3の3枚の等値面を作るとする。s1の等値面になりうるエレメントは図3の①、s2は②、s3は③の領域上にある。つまり、s1、s2、s3の3枚の等値面を作るときには階段状の塗りつぶされている領域上に、必要な3枚の等値面になりうるエレメントがプロットされていることになる。

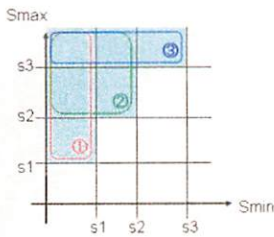


図3 複数等値面の  
同時高速生成

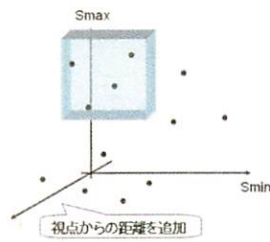


図4 視点から遠い順に  
点群を生成

#### 【視点から遠い順に点群生成する方法】

ISSUEの座標の3次元目に視点からの距離という軸を追加する(図4参照)ことにより、描画する際に遠い順に描くことを実現する。

## 4. 実行結果

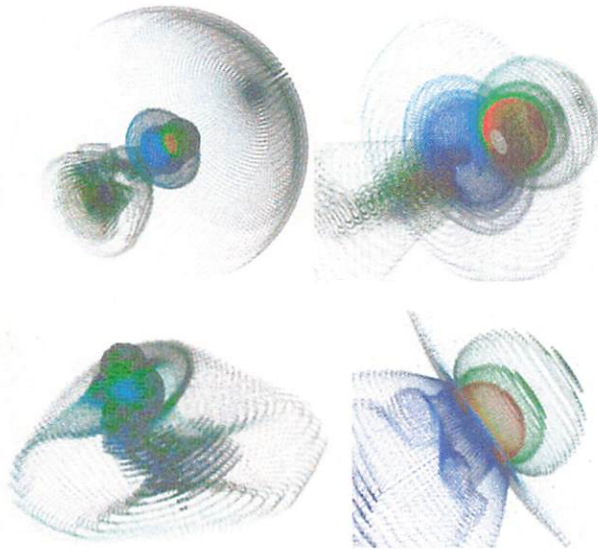


図5. 画像生成例

図5は、流れに対して垂直に置かれた円板のまわりの圧力分布を、提案手法を用いて可視化した例である。この例では6~9枚の等値面を同時生成している。

表1 処理時間測定結果

	Data1	Data2	Data3
本手法	0.96	2.92	11.61
本手法でないもの	15.2	163.64	1551.58

単位はミリ秒、エレメント数はData1で $24 \times 24 \times 24$ 、Data2で $50 \times 50 \times 52$ 、Data3で $100 \times 100 \times 102$ である。本手法を使わない場合と比べ本手法は、Data1では約15.7倍、Data2では56倍、Data3では134倍の高速化に成功した。

表2 Data1を基準にした際の処理時間の比較

	Data2	Data3
本手法	3.042倍	12.094倍
本手法でないもの	10.77倍	102.08倍
データの大きさ	9.404倍	73.785倍

データの要素数を  $N$ 、 $L$  は配列の分割数、 $K$  は1枚の等値面と交差するエレメントの数、 $M$  を生成する枚数とする。ISSUEを使った場合の処理時間は

$$O(M \log(N/L) + \sqrt{NM}/L + MK + MK \log K)$$

となる。複数の等値面のうち1枚以上の等値面と交差するエレメント数を  $K_2$  とすると、本手法の処理時間は、

$$O(\log(N/L) + \sqrt{NM}/L + K_2)$$

となる。また、実行結果で本手法を使わない場合として使った手法の処理時間は  $O(MN + MK \log K)$  となる。以上のことから、データが大きくなればなるほど本手法が有用であることは、表2からだけでなく理論的にも分かる。

今回の実行環境は、Panasonic CH-W4 (CPU 速度 1.20GHz, メモリ 0.99GB) および Windows XP である。

## 5. まとめ

本研究は半透明複数等値面の高速生成を目的とするものである。この解決手段として本報告では、複数の等値面を同時に生成でき、かつ等値面を構成する点群を視点の奥から順に生成できる手法を提案する。本手法ではボリュームを構成するエレメントを、スカラー値の最小値および最大値、視点からの距離、の3値を座標軸とする空間に配置し、この空間上で等値面と交差するエレメントを高速検索することにより、解決素手段を実現する。

今後の課題としては以下の点が上げられる。

【結果画像の改善】 図5(右下)のようにズームアウトすると結果画像に穴が空き、図5(左上)のようにズームインすると描画した領域が重なっている。これ改善すべく、等値面を Surfels として生成する[2]。

【勾配に応じた Surfels の大きさ制御】 平らな領域は Surfels を大きくし、細かな曲面になっているところは Surfels を小さくする機能を追加する。Surfels の大きさ制御により、等値面を取ってくる際の高速化と描画時間の高速化が期待できる。

他に、描画し直したときの各エレメントの勾配の再計算、ISSUE と本手法との計算時間の比較などがある。

## 6. 謝辞

流体のシミュレーションデータは東京大学人工物センター白山助教授と本学科河村研究室からご提供いただきました。この場をお借りして感謝の意を表します。

## 7. 参考文献

- [1] Han-Wei Shen, C. D. Hansen, Y. Linvat, C. R. Johnson: Isosurfacing in Span Space with Utmost Efficiency, Visualization '96, p.287
- [2] Christopher S. Co, Bernd Hamann, Kenneth I. Joy, Reading in Iso-splating: A Point-based Alternative to Isosurface Visualization, 0-7695-2028-6/03, IEEE2003.

# 格子グラフの彩色

有間 久美子 (指導教官:萩田 真理子)

## 1 はじめに

グラフの各点に, 隣り合う二点が互いに異なる色をもつように色を与えることを, グラフの点彩色という.

$G$  を点集合  $V(G)$ , 辺集合  $E(G)$  のグラフとする. 彩色に用いる色数を  $K$  として,  $C = \{1, 2, \dots, K\}$  を色集合とする.  $c$  を  $V(G)$  から  $C$  への写像とし,  $v_p \in V(G)$  に与えられている色を  $c(v_p) \in C$  と表す. このとき,

$$v_a v_b \in E(G) \Rightarrow c(v_a) \neq c(v_b)$$

が満たされていれば,  $c$  を  $G$  の  $K$  彩色という.  $G$  に  $K$  彩色  $c$  が存在するときに  $G$  は  $K$ -彩色可能という.

$$\chi(G) := \min\{K | G \text{ は } K\text{-彩色可能}\}$$

を  $G$  の彩色数 (chromatic number) という. さらに,  $d(v_a, v_b)$  を点  $v_a, v_b$  間の距離とし,  $c \in C$  に対して

$$d(c) := \min \left\{ d(v_a, v_b) \mid \begin{array}{l} c(v_a) = c(v_b) = c, \\ v_a, v_b \in V(G), v_a \neq v_b \end{array} \right\}$$

$$d := \min\{d(c) | c \in C\}$$

として, この点彩色を, 距離  $d$  の分散彩色と呼ぶ. この  $d$  がなるべく大きな値となるような点彩色, つまり同じ色を与えられた異なる二点間の距離が大きい点彩色をみつきたい.

与えられたグラフを比較的少ない色数で彩色するアルゴリズムとして, 以下のウェルシュ・パウエルアルゴリズムが知られている.

### ウェルシュ・パウエルアルゴリズム

- (0)  $V(G) = \{v_1, v_2, \dots, v_P\}$  が  $\deg(v_1) \geq \deg(v_2) \geq \dots \geq \deg(v_P)$ , ( $\deg(v)$  は次数) を満たすように並べ替える.
- (1)  $i = 1$  とする.
- (2)  $c = 1$  とする.
- (3)  $v_i$  の隣接点で色  $c$  を持つものが存在しなければ,  $v_i$  に色  $c$  を与えて (5) へ進む.
- (4)  $c = c + 1$  として (3) に戻る.
- (5)  $i < P$  ならば,  $i = i + 1$  として (2) に戻る.  $i = P$  ならば終了.

このアルゴリズムでは,  $\chi(G)$  に近い色数で, 与えられたグラフを彩色することができる. しかし, 色集合のうち番号の小さい色を優先して使うため, 距離  $d$  が大きな値をとる分散彩色は得られないことが多い.

いま, グラフ  $G$  を  $X \times Y$  個の点をもつ格子グラフ, すなわち平面  $R^2$  上の格子点

$$V(G) = \{v_{(x,y)} | 1 \leq x \leq X, 1 \leq y \leq Y\}$$

を点集合とし, 長さ 1 の 2 点を辺集合

$$E(G) = \{v_{(x,y)} v_{(x+1,y)} | 1 \leq x \leq X-1, 1 \leq y \leq Y\} \\ \cup \{v_{(x,y)} v_{(x,y+1)} | 1 \leq x \leq X, 1 \leq y \leq Y-1\}$$

としたグラフとする. 本研究は, 格子グラフ  $G$  に対して, 距離  $d$  ができるだけ大きな値となる分散彩色をおこなうアルゴリズムの作成を目指したものである.

## 2 格子グラフの最適な分散彩色

ここでは, 同じ色を与えられた異なる二点間の距離が最大となる格子グラフの彩色を考える. このような彩色を”最適な分散彩色”と呼ぶことにする. 色数  $K$  で最適な分散彩色がなされた格子グラフにおける値  $d$  を,  $d_K$  と表す. すると

$$d_K = \begin{cases} 2m+1, & \text{if } 2m^2 + 2m + 1 \leq K < 2m^2 + 4m + 2 \\ 2(m+1), & \text{if } 2m^2 + 4m + 2 \leq K < 2(m+1)^2 + 2(m+1) + 1 \end{cases}$$

が成り立って, 各  $K$  に対応する  $d_K$  は次のようになる.

$K$	2 ~ 4	~ 7	~ 12	~ 17	~ 24	...
$d_K$	2	3	4	5	6	...

以下からは, 作成したアルゴリズムを用いて彩色された格子グラフにおける値  $d$  を, 値  $d_K$  と比較し, それが有用なアルゴリズムであるかどうかを考察する.

## 3 格子グラフ彩色アルゴリズム (1)

- (0) 全ての点  $v$  に色  $K+1$  を与える.  $i = 0$  とする.
- (1) 全ての点  $v$  に対し,  $v$  と同じ色を与えられている点までの最小距離  $d(v)$  を与える.
- (2)  $x = 1, y = 1$  とおく.
- (3)  $d(v_{(x,y)})$  を最も大きくするような色  $k$  ( $1 \leq k \leq K$ ) を, 点  $v$  に与える.
- (4)  $x < X$  のとき,  $x = x + 1$  とする.  $x = X$  のとき,  $x = 1, y = y + 1$  とする.
- (5)  $x \leq X, y \leq Y$  ならば, (3) に戻る.
- (6)  $i = i + 1$  とする.

$2 \leq K \leq 24$  としたときの結果は次のとおり.

$K$	2	5	6	7	8	12	13	18	19	24
$d_K$	2	3	3	3	4	4	5	6	6	6
$d$	2	2	2	3	3	3	4	4	5	5

さらに,

- (7)  $i < 20$  ならば 1 に戻る.

とした場合の結果も上と同じ.



### 4 格子グラフ彩色アルゴリズム (2)

- (0) 全ての点  $v$  に色  $K+1$  を与える.  $i=0$  とおく.
- (1)  $x=1, y=1$  とおく.
- (2) 全ての  $k(1 \leq k \leq K)$  について, 値  $d_{v(x,y)}(c) = 0$  を用意する.
- (3) 点  $v(x,y)$  から距離1のところの色  $k(1 \leq k \leq K)$  を与えられた点が  $n$  個あるとき,  $d_{v(x,y)}(k) = d_{v(x,y)}(k) + \frac{n}{K}$  と計算する.
- (4) 値  $d_{v(x,y)}(k)$  を最も小さくする色  $k$  を, 点  $v(x,y)$  に与えなおす.
- (5)  $x < X$  のとき  $x = x + 1$  とする.  $x = X$  ならば  $x = 1, y = y + 1$  とする.
- (6)  $x \leq X, y \leq Y$  ならば, (2) に戻る.
- (7)  $i = i + 1$  とする.

このアルゴリズムの (3) の部分を

- (3') 点  $v(x,y)$  から距離1のところの色  $k(1 \leq k \leq K)$  を与えられた点が  $n$  個あるとき,  $d_{v(x,y)}(k) = d_{v(x,y)}(k) + \frac{n}{K}$  と計算する.

に置き換えたものを用いて彩色された格子グラフ  $G$  における値  $d$  を  $d^{(2)}$ ,

- (3'') 点  $v(x,y)$  から距離1のところの色  $k(1 \leq k \leq K)$  を与えられた点が  $n$  個あるとき,  $d_{v(x,y)}(k) = d_{v(x,y)}(k) + \frac{n}{K}$  と計算する.

に置き換えたものを用いて彩色された格子グラフ  $G$  における値  $d$  を  $d^{(3)}$  で表す.

$2 \leq K \leq 24$  とした場合の結果は次のとおり.

$K$	2	5	6	7	8	9	12	13
$d_K$	2	3	3	3	4	4	4	5
$d$	2	2	2	2	2	2	2	2
$d^{(2)}$	2	2	2	2	2	3	3	3
$d^{(3)}$	2	2	2	3	3	3	3	4

14	15	16	18	21	22	23	24	$K$
5	5	5	6	6	6	6	6	$d_k$
2	3	3	3	3	3	3	4	$d$
3	3	4	4	4	4	4	4	$d^{(2)}$
4	4	4	4	4	5	5	5	$d^{(3)}$

さらに,

- (8)  $i < 20$  ならば1に戻る.

とした場合の結果も, 上と同じ.

### 5 格子グラフ彩色アルゴリズム (3)

- (0) 全ての点に色1を与える.  $i=0$  とする.
- (1)  $x=1, y=1$  とする.

$$(2) \quad d_{v(x,y)} = \min \left\{ d(v(x,y), w) \left| \begin{array}{l} v(x,y) \in V(G), \\ v(x,y) \neq w, \\ c(v(x,y)) = c(w) \end{array} \right. \right\}$$

を求める.

- (3)  $W = \{w_1, w_2, \dots, w_N\} = \{w | w \in V(G), d(v(x,y), w) = d_{v(x,y)}, v(x,y) \neq w, c(v(x,y)) = c(w)\}$  とおく.
- (4)  $n = 1$  とする.
- (5) 点  $w_n$  からみて, 色  $k(1 \leq k \leq K)$  をもつ点までの最小距離  $d_{w_n}(k)$  を計算する.
- (6) 値  $d_{w_n}(k)$  を最小にする色  $k$  を点  $w_n$  に与える.
- (7)  $n = n + 1$  とし,  $n \leq N$  ならば (5) に戻る.
- (8)  $x < X$  のとき  $x = x + 1$  とする.  $x = X$  ならば  $x = 1, y = y + 1$  とする.
- (9)  $x \leq X, y \leq Y$  ならば, (2) に戻る.
- (10)  $i = i + 1$  とする.

$2 \leq K \leq 24$  としたときの結果は次のとおり.

$K$	2	5	8	9	12	13	16	17	18	24
$d_K$	2	3	4	4	4	5	5	5	6	6
$d$	2	2	2	3	3	3	3	4	4	4

さらに

- (11)  $i < 20$  ならば1に戻る.

とした場合の結果は, 次のとおり.

$K$	2	5	7	8	12	13	14	18	24
$d_K$	2	3	3	4	4	5	5	6	6
$d$	2	2	2	3	3	3	4	4	4

## 6 考察

ウェルシュ・パウエルアルゴリズムは, 与えられた色数  $K$  によらず, できるだけ  $\chi(G)$  に近い色数を用いて彩色を行おうとするものだった. 格子グラフ  $G$  は  $\chi(G) = 2$  であるから, ウェルシュ・パウエルアルゴリズムを適用した場合,  $d = 2$  となる. 一方, 作成したアルゴリズム (1) ~ (3) を用いて彩色された格子グラフでは与えられた全ての色が彩色に使われており,  $K$  の増加とともに  $d$  も増加していく分散彩色となっている. (1) ~ (3) の各アルゴリズムを格子グラフに適用したときの値  $d$  を比較すれば, 格子グラフの分散彩色という目的にたいし最も理想に近い結果を得られるアルゴリズムは, アルゴリズム (1) であると考えられる.

## 参考文献

[1] Robert A. Wilson, *Graphs, Colourings and the Four-colour Theorem*, Oxford University Press, 2002.

## 2次元空間におけるセルオートマトンの自己相似性

玉田 高子 (指導教官: 竹尾 富貴子)

## 1. はじめに

セルオートマトンとは、ある規則(遷移規則という)に従って、自己増殖していく状態を示すもので、遷移規則と初期値を設定すると、ある時刻におけるセルの状態は、直前のセルの状態により決定される。セルオートマトンは様々な分野でシミュレーションのための手法として使用されている。

1次元セルオートマトンでは、様々な遷移規則を調べていくと、描かれる図形が4種類のクラスに分類できることが分かっている(Stephen・Wolfram)。2次元セルオートマトンも遷移規則によりいくつかのクラスに分類できるかどうかを調べていく。

## 2. 1次元における各クラスの状態図の例

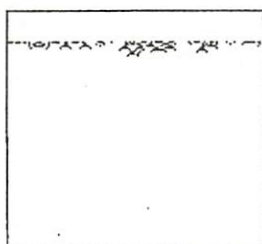


図1: Class1 (No.4)

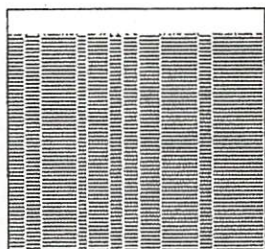


図2: Class2 (No. 3)

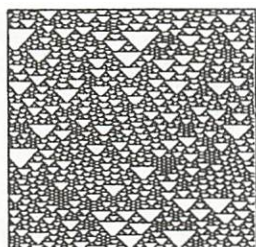


図3: Class3 (No.44)

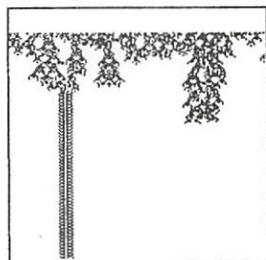


図4: Class4 (No.20)

## 3. 研究目的・内容

本研究では、1次元セルオートマトンで成り立っていることが2次元でも成り立つかどうかを調べる。

研究では、下に示す2種類の遷移規則を使用した。

$$a_i^t = c_1 a_{i+(0,0)}^{t-1} + c_2 a_{i+(1,0)}^{t-1} + c_3 a_{i+(-1,0)}^{t-1} + c_4 a_{i+(0,1)}^{t-1} + c_5 a_{i+(0,-1)}^{t-1} \pmod{M}$$

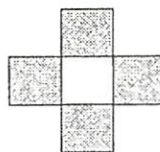
$c_i$ : 係数,  $M$ : 状態数 ( $M > 1$ )

$a_i^t$ : 位置  $i$  の時刻  $t$  での状態を表す

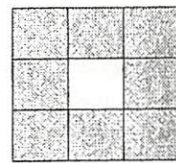
この遷移規則に使用した近傍はフォン・ノイマン近傍で、自分自身とその左右上下の近傍を指す。

$$a_i^t = c_1 a_{i+(0,0)}^{t-1} + c_2 a_{i+(1,0)}^{t-1} + c_3 a_{i+(-1,0)}^{t-1} + c_4 a_{i+(0,1)}^{t-1} + c_5 a_{i+(0,-1)}^{t-1} + c_6 a_{i+(-1,-1)}^{t-1} + c_7 a_{i+(1,-1)}^{t-1} + c_8 a_{i+(-1,1)}^{t-1} + c_9 a_{i+(1,1)}^{t-1} \pmod{M}$$

この遷移規則に使用した近傍はムーア近傍で、自分自身とその周囲8つの近傍を指す。



フォン・ノイマン近傍



ムーア近傍

## 4. 2次元セルオートマトンの自己相似性

まず、2次元セルオートマトンの自己相似性について調べるために、1点から開始して、いろいろな  $M$  と各時刻の図について調べた。調べた時刻の範囲は  $t=1 \sim 100$ 。

4-1. 状態数  $M$  の分類

①  $M$  が素数

$$M = 5, 7, 11, 13$$

②  $M$  がある素数の累乗

$$M = 8, 9, 16, 25$$

③  $M$  が素数とある素数の累乗の積

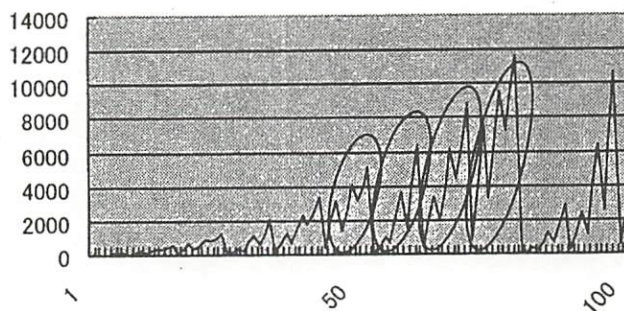
$$M = 12, 18, 20, 24$$

④  $M$  が素数と素数の積

$$M = 6, 10, 14, 15, 21$$

## 4-2. 各時刻の状態図と周期性

下のグラフの横軸は時刻を表し、縦軸は0でないセルの合計個数を表す。 $M=9$ のグラフで近傍はフォン・ノイマン近傍。



4 - 3. 状態数 M と各時刻の状態図

M の分類	図の特徴
②	$t = cp^{k-1} (c \in Z, c \geq 0)$ とすると, $pt$ と $t$ の時刻の図は相似
④	$M = p \times q$ とすると, 各時刻の図は $M = p$ と $M = q$ の場合の各時刻の図の和集合
①~④	$M = n$ とすると, $t = n, 2n, 3n, \dots$ の時に図の密度が小
①, ②	図の密度が小さくなる直前に図の密度が大になる.

表 1 : 遷移関数の  $c_1 = \dots = c_9 = 1$

この性質を証明するための定理

<p><b>定理</b> <math>M = p^k</math> の時,</p> <p>(i) <math>t = sp^{k-1} \Rightarrow a_{pi}^t = a_i^t</math></p> <p>(ii) <math>t = sp^k, \frac{i}{p} \notin Z \Rightarrow a_i^t = 0</math></p> <p>(<math>s \in Z, s \geq 0</math>), <math>t</math> : 時刻</p>
--

4 - 4. 状態数 M と剰余

M の分類	各剰余の数
①	ほぼ同じ
②	$M = p^k$ と表すと, 剰余 $p, 2p, 3p, \dots$ のところで多い
③	グラフが線対称
④	$M = p \times q (p < q)$ と表すと, 剰余 $p, 2p, 3p, \dots$ のところが多くなり, 剰余 $q, 2q, 3q, \dots$ のところでも少し多め

表 2 : 遷移関数の  $c_1 = \dots = c_9 = 1$

5. 2次元セルオートマトンにおける分類  
 $a_{i+(0,0)}^t = f(a_{i+(0,1)}^{t-1}, a_{i+(1,0)}^{t-1}, a_{i+(0,0)}^{t-1}, a_{i+(-1,0)}^{t-1}, a_{i+(0,-1)}^{t-1})$   
 $a_i^t$  のとる状態は 0 か 1. 近傍の状態の総数は 32 種類あり, これら各々に対し 0 または 1 を割り振るので, 規則の総数は,  $2^{32} = 4294967296$  (約 42 億通り). この規則をすべて調べあげるのは不可能なので, 簡略化して総和型規則で考える.

$$a_{i+(0,0)}^t = g(a_{i+(0,1)}^{t-1} + a_{i+(1,0)}^{t-1} + a_{i+(0,0)}^{t-1} + a_{i+(-1,0)}^{t-1} + a_{i+(0,-1)}^{t-1})$$

と  $g$  を定め,  $[g(5) g(4) g(3) g(2) g(1) g(0)]$  を 2 進数と見なしたときの数値をその規則番号とする. そうすると規則は 64 通りになる. 規則番号は 0~63.

(例)  $[101110]$  の場合

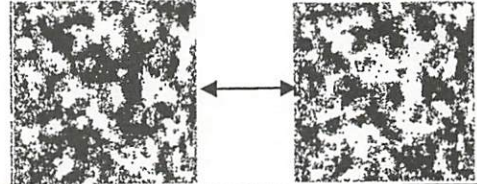
$$2^5 \times 1 + 2^4 \times 0 + 2^3 \times 1 + 2^2 \times 1 + 2^1 \times 1 + 2^0 \times 0 = 32 + 8 + 4 + 2 = 46 \leftarrow (\text{規則番号})$$

5 - 1. プログラム

- 規則番号を読み込み 2 進数に変換.
- 初期値は  $200 \times 200$  個のセルにランダムに 0 か 1 を割り振る.

5 - 2. 結果

下の図の様に「前の時刻で 0 の値を持つセルが 1 になり, 1 のセルが 0 になる」といった状態を繰り返すものや, 比較的短い時間で何点かに収束してしまうものや, すべてのセルが 0 になってしまうもの, 奇妙な模様を描くものなどが見られた. 描かれる図をもとに分類すると, 以下の様になった.



I	0, 16, 32, 48
II	11, 43
III	7, 14, 39
IV	1, 3, 15, 31, 33, 35, 47, 63
V	8, 40
VI	24, 56
VII	6, 25, 30, 38, 57, 62
VIII	4, 20, 36, 52
IX	上に入らないもの

各時刻における値が 1 のセルの合計数をカウントし, グラフを作ると, グラフの形状から先ほど分類したものは, 以下の 4 つに分類できた.

1	I
2	III, IV
3	II, VII, IX
4	V, VI, VIII

6. 今後の課題

- 2次元セルオートマトンの分類で今回はフォン・ノイマン近傍で考えたので, ムーア近傍で調べる.
- 得られた結果を理論的に証明する.

参考文献

[1] S. Takahashi, Self-Similarity of Linear Cellular Automata, *J. Comput. System Sci.* 44(1992), 114-140

[2] 森下 信, 『セルオートマトン—複雑系の具象化—』, (養賢堂 2003 年)

[3] 山本 芳人, 『Java による図形処理入門』, (工学図書 2001 年)

# 砂山モデルによる幾何学模様

根岸 伸江 (指導教官: 竹尾 富貴子)

## 1 はじめに

セルオートマトン法 (CA 法) とは, セル間の局所的な相互作用の原理に基づく計算手法である. その特徴は, 生体の組織形成, 経済, 交通, 災害などの複雑な現象を簡単なモデルで自己組織化させて再現できることであり, 様々な分野で適用されている. 自己組織化とは人の介入なしで組織が自然に形成されることである.

CA 法の一つに砂山モデルがある. これは 1987 年に Bak, Tang, Wiesenfeld が提案した自己組織化臨界現象を示す代表的なモデルであり, 空間, 時間, 状態変数を離散的な整数値で表すものである. 本研究では, 砂山モデルを用いて得られる構造のパターンについて研究をする.

## 2 2次元の砂山モデル

- $L \times L$  の 2 次元有限格子を考える.
- 各サイト  $(x, y)$  に 0 以上の整数  $Z(x, y)$  を与え, サイト  $(x, y)$  の勾配とする.
- しきい値を  $Z_c = 3$  とする.
- あるサイト  $(x, y)$  で  $Z(x, y) > Z_c$  となったとき, 以下のプロセスを行う.

$$\left. \begin{array}{l} Z(x, y) \rightarrow Z(x, y) - 4 \\ Z(x \pm 1, y) \rightarrow Z(x \pm 1, y) + 1 \\ Z(x, y \pm 1) \rightarrow Z(x, y \pm 1) + 1 \end{array} \right\} \text{(A)}$$

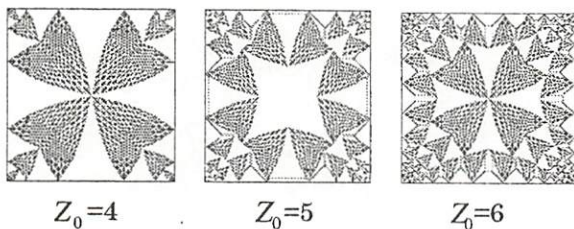
- (A) のプロセスをすべてのサイトの勾配が  $Z_c$  以下になるまで繰り返す.

ここでは閉じた境界条件とする. (A) のプロセスが行われたとき, 「なだれ」が発生したと考える.

## 3 最終状態

本研究では, 初期条件としてすべてのサイトの勾配を  $Z_0 > Z_c$  とし, (A) のプロセスを行った後の最終状態について調べた.

$Z$  の値 (0~3) によって色分けをして表した.



いずれの初期値  $Z_0$  においても上下・左右・対角線で対称な形となった. また, 直角に向かって自己相似な形が作られ,  $Z_0 - 3$  個の形が横につながって作られた.

## 4 各サイトにおけるなだれが起きた回数

初期状態から最終状態になるまでの, 各サイトで起きたなだれの回数  $C(x, y)$  を調べた. 本研究ではまず  $Z_0 = 4$  について考えていく.

### 4.1 なだれの回数と最終状態との関係

(A) のプロセスから, 最終状態において

$$Z(x, y) = Z_0 + C(x+1, y) + C(x-1, y) + C(x, y+1) + C(x, y-1) \pmod{4} \quad (1)$$

と表されることが分かる. よって, 回数と最終状態で作られる図形とは大きなつながりがあるといえる. 図形の輪郭は  $Z=3$  以外となるサイトによって作られているので, 回数の規則性を調べることで自己相似な形が作られる仕組みがわかるだろうと考えた.

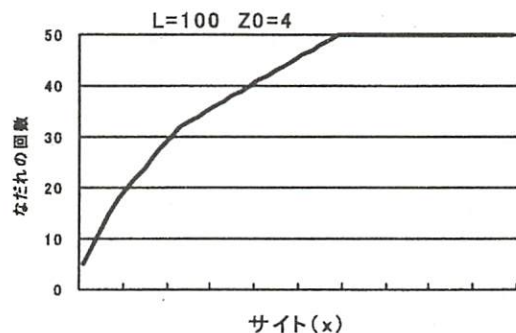
### 4.2 実行結果

中心のサイトの値が一番大きくなり, 周りのサイトにいくにしたがって値は小さくなり, 最終状態の勾配と同様に上下・左右・対角線で値が対称となった. また,  $L$  は  $\{3, 4\} \{5, 6\} \{7, 8\} \dots$  と  $C(x, y)$  の値の種類が等しい組で分けられたので, 各組を小さいほうから順に  $n=1, 2, 3, \dots (n=\lceil L/2 \rceil - 1)$  とする.

### 4.3 回数の規則性

さらに実行結果から回数の規則性を詳しく調べた. 対称となっているため格子の左上の 4 分の 1 を考える. 左のサイトから順に  $x=1, 2, 3, \dots$ , 上のサイトから順に  $y=1, 2, 3, \dots$  とする.

例として  $y=1$  のときの  $x$  と回数の関係を以下のグラフに示す.



なだれの回数は  $x$  が大きくなるにしたがって多くなる。また、グラフの傾きは徐々に小さくなり、最後は一定の値になっている。このことはどの  $y$  でも成り立ち、任意の  $y$  に対し一定の値になり始める  $x$  を  $x_y$  とすると  $x_{y+1} - x_y = 0$  または  $1$  となっていた。また、各  $y$  において一定になる値は、

$$\max_x C(x, y) = \frac{1}{2}n^2 + \frac{3}{2}n + 1 - \sum_{k=1}^{n-y+1} k \quad (2)$$

となることも分かった。

#### 4.4 輪郭が作られる仕組み

$y=1$  のとき、傾きが変わっているそれぞれの  $x$  が、各図形の頂点になっている  $x$  と一致していることが分かった。そこで、傾きが変わっている  $x$  におけるサイトが図形の輪郭を作るのに関係があると考え、仕組みを詳しく調べた。

$x > x_y$  となる任意のサイト  $(x, y)$  について  $C(x, y) = a$  とする。  $t = n - y + 1$  とおくと(2)より、  $C(x \pm 1, y) = a$ ,  $C(x, y-1) = a - t - 1$ ,  $C(x, y+1) = a + t$  となる。よって(1)より  $Z(x, y) = 3$  となる。次に  $(x_y, y)$  について考える。同様にして  $C(x_y, y) = a$ ,  $C(x_y + 1, y) = a$ ,  $C(x_y - 1, y) = a - 1$ ,  $C(x_y, y - 1) = a - t - 1$ ,  $C(x_y, y + 1) = a + t$  より、(1)から  $Z(x_y, y) = 2 \neq 3$  となるので、このサイトが図形の境界となることがわかる。以上のことは任意の  $y$  について成り立ち、対角線で対称になっていることから、一番大きい図形の輪郭（なだらかな曲線部分）が作られることになる。このようにグラフにおいて傾きが変わっている  $x$  が図形を作る上で深い関係があることは分かったが、  $x < x_y$  についてはまだ規則性がはっきりせず、また自己相似な形が作られる仕組みについては明らかになっていない。

#### 5 2つの最終状態の和

2つの最終状態のすべてのサイトにおいて、同じサイトの勾配同士を足し、再び(A)のプロセスを行う。

$Z_0 = i$ ,  $Z_0 = j$  の最終状態の勾配をそれぞれ  $Z_i(x, y)$ ,  $Z_j(x, y)$ , なだれが起きた回数をそれぞれ  $C_i(x, y)$ ,  $C_j(x, y)$  とする。2つ同じサイトの勾配同士を足したときの各サイトの勾配は、(1)より

$$= i + C_i(x+1, y) + C_i(x-1, y)$$

$$\begin{aligned} &+ C_i(x, y+1) + C_i(x, y-1) \\ &+ j + C_j(x+1, y) + C_j(x-1, y) \\ &+ C_j(x, y+1) + C_j(x, y-1) \pmod{4} \quad (3) \end{aligned}$$

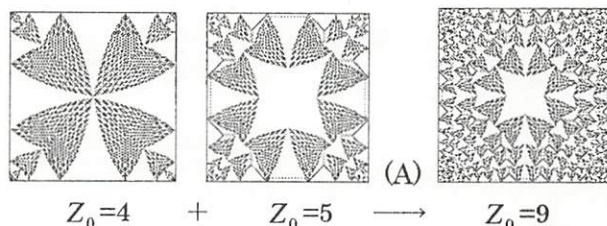
となる。次に(A)のプロセスを行ったときの最終状態になるまでに起きたなだれの回数を  $C_k$  とすると、最終状態の勾配は(1)より

$$\begin{aligned} Z(x, y) &= Z_0(x, y) + C_k(x+1, y) + C_k(x-1, y) \\ &+ C_k(x, y+1) + C_k(x, y-1) \pmod{4} \quad (4) \end{aligned}$$

と表される。また、実行結果から各サイトで起きたなだれの最終的な回数  $C_i(x, y) + C_j(x, y) + C_k(x, y)$  は  $C_{i+j}(x, y)$  と等しくなることも分かったので、(1), (3), (4)より

$$Z(x, y) = Z_{i+j}(x, y)$$

となり、結果的に  $Z_0 = i + j$  の最終状態と一致することがいえる。以下は  $i=4$ ,  $j=5$  としたときの実行結果である。



まとめとして、各サイトに勾配が足される順番、

なだれが起こる順番は関係なく、境界から勾配が失われた後でも合計でいくつの勾配が足されたかで最終状態が一意に決まることがわかった。

#### 6 まとめと今後の課題

実行結果からわかった規則性をもとに図形と結びつけて考えてきたが、自己相似な形が作られる仕組みについて明らかになっていないので今後の課題としたい。また、今回はなだれの回数に着目して考えていったので、他の視点からも考えるなどしてさらに砂山モデルの特徴を追究していきたい。

#### 参考文献

- [1] 本田勝也：「フラクタル」朝倉書店、2002
- [2] 加藤恭義, 光成友孝, 築山洋：「セルオートマトン法—複雑系の自己組織化と超並列処理—」森北出版、1998
- [3] S. H. Liu, Theodore Kaplan and L. J. Gray, Geometry and dynamics of deterministic sand piles, *Phys. Rev. Lett.* **42**(1990), 3207-3212.

# 類似検索を用いた検索支援システムの開発

倉地 彩乃 (指導教員: 小林 一郎)

## 1 研究背景と目的

今日、インターネットで公開されている情報を検索し必要な情報を得ることが日常生活の一部となってきた。現在では Google をはじめとするすぐれた情報検索エンジンが存在するが、未だに得られる検索結果の上位ページの中に自分が意図した情報が掲載されているものを見つけるのは難しい。この理由として、検索結果の中に自分が意図したものが含まれてはいるが検索質問で使用したキーワードの対象世界での使い方の違いや同音異義語の存在などから該当するページが下位に位置づけられているためなどが考えられる。本研究では、このような問題点に着目し、検索エンジンによって得られる結果の文書群の中から自分の意図したページ (以後、「代表ページ」と呼ぶ) を一つ選び、そのページと類似する情報を持つページを検索結果の文書群の中から再度検索し、ユーザに提示する手法を提案する。

## 2 提案手法による処理のながれ

提案する手法の処理のながれを以下に示す。

### step1 既存の検索システムによる文書検索

ユーザは、できるだけ多くの情報を網羅的に取り出すことができるよう、任意の検索エンジンにおいて必要最小限の個数のキーワードで検索を行なう。

### step2 取得 Web ページの特徴分析および、文書ベクトルの作成

step1 において取得された Web ページの HTML タグの分析および語の頻度情報、文書頻度情報などを求め、その特徴量に基づき文書ベクトルを作成する。

### step3 代表ページの選択

step1 において検索された Web ページの中から代表ページを一つ選択する。

### step4 類似文書の検索

step2 で作成された取得された Web ページの文書ベクトルと、step3 によって選択された代表ページの文書ベクトルとのマッチングを行い類似度を判定する。

step1 では、必要最小限のキーワードで検索を行い、予め検索結果の再現率を高く保つようにしている。実際には検索結果の分析にかかる時間を考慮して、再現率の高さを制限することになるが、初期の段階における該当ページのこぼれ落ちを防ぐことを考慮している。step2~step4 では、再現率を高くなるように検索した結果、意図しない内容のページを多く含む文書群の中から、ユーザの検索要求に対して適合率が高い文書群を判別している。これにより、ユーザの意図するページを網羅的かつ的確に検索できる処理が実現されている。

## 3 システムの概要

システムの概要を図 1 に示し以下、システムの構成要素一つずつについて以下に説明する。

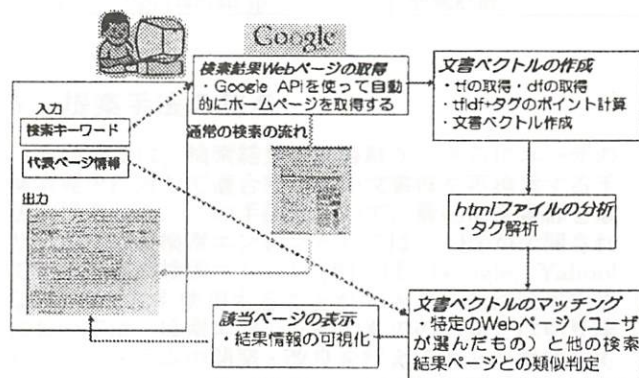


図 1: システムの概要

### 3.1 検索結果 Web ページの取得

本研究では、使用する検索エンジンとして、その API が公開されている Google を対象とした。Google は 2002 年にその検索エンジンに直接アクセスできる API が公開されており [1], 自らが開発するシステムの中から直接、Google の検索エンジンが使用できるようになった。開発したシステムは Java 言語によって構築されており、その中から GoogleAPI を呼び出し、入力されたキーワードに対しての Web ページを取得している。

### 3.2 HTML ファイルの分析

検索された Web ページの内容および特徴を抽出するために文書中の語彙の重要度を重みとする文書ベクトルを作成し、個々の文書の比較をおこなう。対象としている文書が Web ページの場合は、HTML タグによる文書の注釈の仕方により語彙の重要度が判別できるため、タグの種類に基づくポイントを重要度の計算に加えている。今回、分析対象としたタグは文字のサイズを示す H1~H6, ハイパーリンクを示す A, 太字を示す B の計 8 種類であり、それぞれにポイントを加えることとした。ポイントはタグによって語彙が強調される程度に応じた。Java には Perl の CPAN [3] のような HTML を解析するモジュールが存在しないため<sup>1</sup>, Web ページ中のタグを判別するとフラグを立て、特定のタグで囲まれた箇所を認識するプログラムを作成した。この処理を行うためには、本文中の文字を形態素レベルで判別する必要がある。そのため本研究では、奈良先端科学技術大学院大学松本研究室で開発された形態素解析システム『茶筌』 [2] を使用した。しかし、茶筌では半角英数字を適切に形態素解析することができないため、分析対象となる Web ページ本体を全角の文字に変更してからタグの分析を行った。

<sup>1</sup>以前は CJAN (<http://cjan.org/>) というプロジェクトが存在していたが、現在 (2006 年 2 月初旬) は活動を休止しているようである。

### 3.3 文書ベクトルの作成

本研究では、文書の内容を文書ベクトルで捉え、文書ベクトル同士の比較を行うことにより、文書間の類似性を判別する。文書の特徴付ける語彙として名詞と動詞に注目し、それらをベクトルの成分とした。基本的に文書ベクトルは、対象となる  $i$  番目の文書に対してベクトル成分が  $n$  個存在するときに、以下の式で表現される。

$$D_i = a_{i1}V_{i1} + a_{i2}V_{i2} + \dots + a_{in}V_{in}$$

$D_i$  :  $i$  番目の文書ベクトル

$a_{ij}$  :  $D_i$  における  $j$  番目の語彙の重要度

$V_{ij}$  :  $D_i$  における  $j$  番目の語彙

ここで、 $a_{ij}$  はベクトル成分  $V_{ij}$  の重要度を示す係数であり、各ベクトル成分の重要度はその語の  $tf \cdot idf$  値およびタグの種類に基づくポイントの加算によって決定される。

$$a_{ij} = tf_j \cdot idf_i + HTML \text{ のタグによるスコア}$$

文書ベクトルの一例を図 2 に示す。この例では「ストローク」、「サーブ」、「ラケット」をベクトル成分としている。

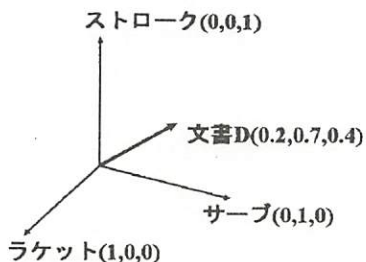


図 2: 文書ベクトルの例

### 3.4 文書のマッチング

文書ベクトルによって表示された各文書とユーザが選んだ代表ページとの類似判別が行われる。ある閾値より高い類似度をもった文書が類似度の高い順に並び替えられユーザに提示される。

文書ベクトルの類似度は以下の式によって計算される。

$$sim(D, Q) = \sum_{i=1}^t a_i q_i$$

$D$  : 文書  $D$  の文書ベクトル

$Q$  : 代表ページ  $Q$  の文書ベクトル

$a_i$  :  $D$  の  $i$  番目のベクトル成分の重み

$q_i$  :  $Q$  の  $i$  番目のベクトル成分の重み

## 4 システム実行結果

ユーザ入力を「テニス、ストローク」とした。まず最初にストロークの打ち方について記述されたブログを選択するとブログのページが上位に出現した。それに次いで上位に出現するものは「テニス」、「ストローク」のキーワードの出現頻度が多いものとなった。これは検索された Web ページの大きさに依存すると考えられる。次にテニスの練習器具のページを選択した場合、最上位に同様の内容が記されたページが出現した。このことから、本システムで提案された方法が妥当であると考えられる。

上記、の最初の例についての、Google で得られる結果と、本システムを用いて得られる結果である。

表 1: 実行結果例

出力順	Google による検索結果	提案システムによる結果
1	APL 初級者歓迎オフ	Blog:テニス強化計画
2	前腕筋群	Sportsman.jp
3	EnjoyPlayers ブログ!	EnjoyPlayers ブログ!
4	Blog:テニス強化計画	APL 初級者歓迎オフ
5	Sportsman.jp	前腕筋群

## 5 提案手法の利点

本研究では、検索結果の文書群からさらにユーザの検索要求に対して適合率が高い文書群を再検索する手法を提案した。この手法において、最初に文書群を取り出すための検索エンジンとしては、API が公開されている既存の検索エンジン (例えば、Google, Yahoo! など) のどれも使用することができる。これは本手法においては、検索エンジンそのものの開発とは切り離して、システムの開発・改良を行えることを意味している。また、ユーザが代表ページとして選ぶページは「このような雰囲気をもつ同じようなページ」というように明確なキーワードを意識して選択する必要がないため、検索を行う際にユーザが適切なキーワードを知らずとも自分の要求に的確なページを検索できることなどが挙げられる。このことから、検索に不慣れな人でも必要な情報を精度よく手に入れることができるようになる。さらにユーザがあまり内容を知らない分野についても的確に欲しい情報の内容を示しているページを検索することができるようになると思われる。

## 6 まとめ

本研究では、文書ベクトルを構築する際に、各ベクトル成分の重みとして、 $tf \cdot idf$  を利用したが、語の共起回数などを重みとした文書ベクトルの有効性など示されていることから [4]、より正確に文章の内容を捉える工夫をすることを考えている。さらに、現段階では HTML のタグごとにテキストを取り出すことができているため、取得した Web ページ全体を形態素解析して分析する手法をとらざるを得なくなっているが、CPAN のように HTML のタグ解析が高速に行えるモジュールなどが利用できるようになれば、処理の高速化が図れると考えられる。また、取得した Web ページは、さまざまな文字コードに変換したシステムに適する文字コードに変換した後も、茶筌を使ってその結果を解析するために分析するテキストの分量を制限するなどの工夫を要した。このような文字コードの問題も解決できれば、システムは高速かつ安定に動くようになると思われる。

## 参考文献

- [1] <http://www.google.com/apis/>
- [2] 奈良先端科学技術大学院大学情報科学研究科自然言語処理学講座 (松本研究室), <http://chasen.naist.jp/hiki/ChaSen/>
- [3] <http://www.cpan.org/>
- [4] 語の共起情報を用いた文書クラスタリング, 小熊淳一, 内海彰, 電気通信大学, 第 19 回人工知能学会全国大会, 2E1-01, 2005.
- [5] 文書クラスタリングを用いた Web 検索支援システム, 折原大, 電気通信大学電子情報学専攻経営システム工学講座修士論文, 2004.
- [6] 特集「テキストマイニング」, 人工知能学会誌 Vol. 16 No. 2, 2001 年.
- [7] 情報検索の理論と技術, 岸田和明, 頸草書房, 1998 年.

# ユーザの入力キーワードに関連する時事的な情報を提供する情報推薦システム

齋藤 舞 (指導教員: 小林 一郎)

## 1 研究背景と目的

インターネットの急速な普及により、容易に多量の情報を得ることができるようになった。しかし、情報量の多さゆえ、自分の要求を満足する情報を見つけることが困難になっている。実際には、検索された結果以外にもユーザにとって有益な情報が存在する可能性が大いにある。これらのことから情報を推薦するシステムの研究が盛んになってきている [1]。

本研究においては、ユーザが情報検索のために入力したキーワードに関連した時事的な情報を提示する情報推薦システムを開発することを目的とする。

## 2 安価な情報推薦システム

情報推薦の方式は以下の二つの方式に大別される [2]。

### ● コンテンツに基づくフィルタリング方式

コンテンツに基づくフィルタリング方式は、推薦する情報の内容に基づき、ユーザに推薦する情報を決定する方式。

### ● 協調フィルタリング方式

協調フィルタリング方式は、ユーザAと関心が類似した別のユーザBが高く評価した情報を、ユーザAにも薦めるといったものである。

これらの情報推薦方式では、いずれの場合も、個々のユーザにおける行動(検索)履歴などの膨大なデータを保有・管理する必要があり、有益な情報推薦を受ける代わりに、システム構築にかかるコストは高価なものとなる。

本研究においては、その様な高価な情報推薦システム構築を目指すのではなく、通常の情報検索作業への傍らで、ユーザの興味に関連する時事的な情報の存在を教示する安価な情報推薦システムの構築を目指す。具体的には、特定のニュースサイトのRSSから、その時に話題となっているニュースを取得し、そこから時事的なキーワードを抽出し、ユーザが検索のために入力したキーワードと関連する時事的なキーワードを判別し、それを用いて、検索した結果も同時に表示することにより、ユーザの興味(検索キーワード)に関連する情報も提供する情報推薦システムを構築する。

## 3 システムの概要

開発したシステムの構成を図1に示す。以下に処理の流れに沿ってシステムの構成を説明する。

### 3.1 RSSから時事ニュースの取得

gooで提供している時事ニュースにおけるRSS(RDF Site Summary)<sup>1</sup>から時事ニュースのHTMLファイルを取得する。RSSとは、Webサイトの見出しや要約などのメタデータを構造化して記述するXMLベースのフォーマットである。RSSファイルの例を図2に示す。

<sup>1</sup><http://news.goo.ne.jp/news/rss/topics/chumoku/index.rdf>

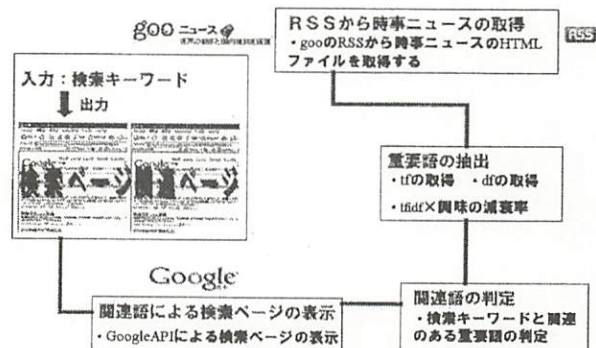


図1: システムの概要

```
<?xml version="1.0" encoding="UTF-8" ?>
<rdf:RDF xmlns="http://purl.org/rss/1.0/" xmlns:dc="http://purl.org/dc/elements/1.1"
xmlns:sy="http://purl.org/rss/1.0/modules/syndication/" xmlns:admin="http://webi
<channel rdf:about="http://news.goo.ne.jp/news/rss/topics/chumoku/index.rdf">
<title>注目のトピックス - gooニュース</title>
<link>http://news.goo.ne.jp/news/rss/topics/index.html</link>
<description>最新の注目トピックスを7本を配信します</description>
<dc:language>ja</dc:language>
<dc:creator>NTT Resonant</dc:creator>
<dc:publisher>NTT Resonant</dc:publisher>
<dc:rights>Copyright(C) 2006 NTT Resonant Inc. All Rights Reserved.</dc:rights>
<dc:date>2006-01-10T19:00:23+09:00</dc:date>
```

図2: RSS ファイル

システムは、Perl でプログラミングされており、RSSからのニュース本文の抽出などはperlモジュールCPAN [3]を用いて実装されている。

gooニュース中の注目のトピックスは約30分おきに時事ニュースが更新されている。本研究においては、12月8日から12月11日の時事ニュースを対象としており、1日3回6時間おきに4日間、計84のニュースを取得した、HTMLファイルを対象とした。

### 3.2 時事ニュースからのキーワード抽出法

取得対象となるテキストはタイトルと本文である。取得したHTMLファイルは形態素解析システム茶筌[4]によって形態素解析がなされ、名詞と未知語を抽出している。

未知語を抽出する理由は、時事的なニュースには絶えず新語が現れる可能性があるため、その様な語彙も時事的な重要度として見逃さないためである。

### 3.3 重要語の抽出

語彙の重要度は、語彙に対する $tf \cdot idf$ の値および日ごとの興味の減衰率によって決定される。また、時間によって重要語の新鮮度が変わること考慮して、日ごとの興味の減衰率を考慮した計算式により時事的に重要と判別される語彙を決定している。減衰率の計算式は、ガンマ分析に従うという仮定がなされているが[5]、本研究では指数関数で表現している。また、日ごとの興味の減衰率の式にはエビングハウスの忘却曲線[6]を使用した。

エビングハウスの忘却曲線は、ドイツの心理学者、ヘルマン・エビングハウスが自分を実験台にしてラン





# グラフと協調するテキスト要約

渡邊 千明 (指導教員: 小林 一郎)

## 1 研究背景と目的

インターネットが普及するにつれ、インターネット上の膨大な情報を利用できる人、そうでない人の格差であるデジタルデバイドという社会現象が起きている。この要因の一つとして考えられるのが、インターネットから得られる情報の内容や表示が難解であり、情報を提供する側において、ユーザが欲しい情報を欲しい形で提供するなどの工夫がなされていないためである。そのため本研究では、このような現状を踏まえ、情報の内容や表示を誰にでも理解されやすいよう情報提示の形態を動的に変化させることができる、知的コンテンツの開発を試みる。その具体例として、テキストとグラフという異なるメディア同士を協調させることにより、大まかな情報を必要とするユーザ、または詳細な情報を必要とするユーザなど、それぞれのユーザに適した情報を提示させることを目的とする。



図 1: MuST コーパスと日経平均株価の数值情報

## 2 対象コンテンツ

本研究では、日経平均株価の動向を示すテキストとグラフを対象とする。テキストデータとして、国立情報学研究所の主催で実施されている評価型ワークショップ (NTCIR) のひとつである「動向情報の要約と可視化に関するワークショップ」[3] で提供されている MuST コーパスを利用している。MuST コーパスとは、1998年から1999年の2年間分の毎日新聞を基に、ガソリン価格やパソコン出荷状況など20トピックについて時系列になっている記事を収集し、各トピックにつき3つ前後の統計量を選んで、これらの統計量の可視化に必要な要素に対して、XML文書の形式で、人手でタグを付与したものである。タグの例として、指定された統計量や出来事に言及している部分に付与されている unit タグ、「10日」「今年」「昨年」等の時刻の表現に付与されている date タグ、統計量の値そのものではないが、その値の差や比に付与されている rel タグなどがある。



図 2: システムの概要

## 3 提案手法

本研究の準備として、日経平均株価の数值情報と、MuST コーパスによって得られるその日の株価の動向情報に対応させ、グラフとテキストを関連付ける。MuST コーパスと日経平均株価の数值情報の対応の仕方を図 1 に示す。

### 3.1 処理概要

表示されたグラフから、その一部を取り出したり、目盛りの間隔を変更したりすることで、変更されたグラフの詳細度、範囲に対応して、ニュース本文が要約され、グラフの表示とテキストが協調した情報提示を行っている。要約文は、MuST コーパスに含まれるテキストの中から重要文を抽出することにより生成される。また、文に対する重要度はグラフに対する処理に応じて動的に変更され、グラフの表示状態に応じた重要文からなる要約文が生成される。システムの概要を図 2 に示す。

### 3.2 システム構成図

数值データから必要な部分を取り出し、グラフを表示させる。MuST コーパスの方も同様に、グラフの詳細度に対応して記事をサンプリングし、重要度の高い文を抽出させ要約文を表示する。これにより表示されるグラフとテキストの協調を実現している。

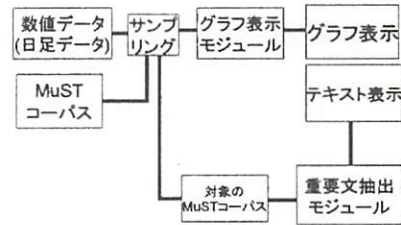


図 3: システム構成図

### 3.3 要約対象となる文の重要度の決定方法

本研究において、要約対象となる各文の重要度は、MuST コーパスに用意されているタグに基づき決定している。unit タグは、具体的にグラフの挙動 (数值情報が得られる箇所) が記載されている文に付与されていることから、その文は他の文と比べて重要度が高いとする。また、unit タグが付与されている文の中でも、「前日比」というように、短い期間の挙動について言及しているものから、「3年前と比べて」というように、長い期間の言及している文もある。この場合、日付に付与されている date タグを参考にし、同じ unit タ

グが付与されている文の中でも、より長い期間の挙動について言及している文を重要度がより高いとする。他に、見出しに付与されている HEADLINE タグを参考にし、見出しで取り上げられている話題に言及している文を重要とする。この方法として、取り出した見出しを、茶笥で形態素解析を行う。その結果から名詞のみを取り出し、その名詞が含まれている文を重要と判断する。さらに重要度のランク付けとして、見出しに含まれている名詞が、より多く含まれている文をより重要とする。

テキスト1 文1重要度3 文2 ←3 文3 ←3 文4 ←5 文5 ←6 文6 ←4	テキスト2 文1重要度2 文2 ←2 文3 ←2 文4 ←3 文5 ←4 文6 ←5	テキスト3 文1重要度3 文2 ←3 文3 ←4 文4 ←5 文5 ←5	テキスト4 文1重要度3 文2 ←3 文3 ←6 文4 ←6 文5 ←4 文6 ←4	テキスト5 文1重要度6 文2 ←5 文3 ←3 文4 ←3 文5 ←3 文6 ←2
--	--	---	--	--

新たに作られたテキスト

テキスト3 文3	テキスト4 文5
テキスト3 文4	テキスト4 文6
テキスト3 文5	テキスト5 文1
テキスト4 文3	テキスト5 文2
テキスト4 文4	

### 3.4 グラフの目盛り間隔を変更

グラフが変更され、2日おき、4日おきのよう目盛りの間隔が広がった場合、2日ごと、4日ごとのように、重要文を抽出してテキストをまとめる。さらに、それぞれから抽出されたテキストから、さらに重要度の高い文を抽出する。

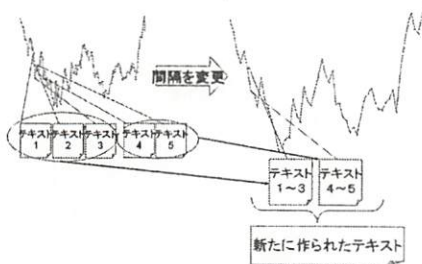


図 4: グラフの目盛り間隔の変更

この処理により、抽出されるテキストは減り、動向を示す情報の詳細度が低くなる。また、間隔が狭くなれば抽出されるテキストが増え、詳細度が高くなる。システムの実行例を図5に示す。

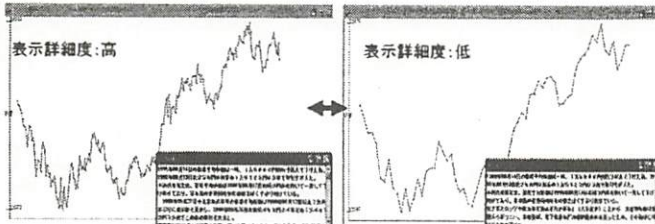


図 5: 実行例 (グラフの目盛り間隔の変更)

### 3.5 範囲を選択

グラフの一部分が選択された場合、選択された範囲にある部分のテキストから重要度の高い文を抽出する。このとき、抽出する文の長さは指定可能である。

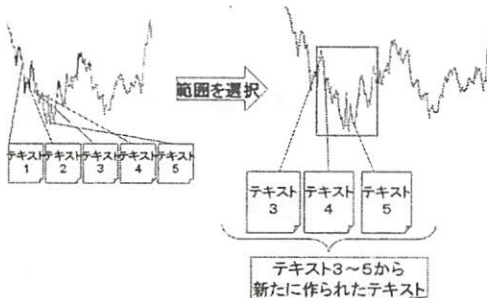


図 6: 特定箇所の情報抽出

図 7: 重要度による抽出例

この処理により、テキストも選択した部分を焦点とした内容となる。範囲が狭くなればなるほど、変更する前には抽出されなかった重要度の低い文も抽出されるようになり、その範囲のみをより詳しく説明したテキストとなる。システムの実行例を図8に示す。

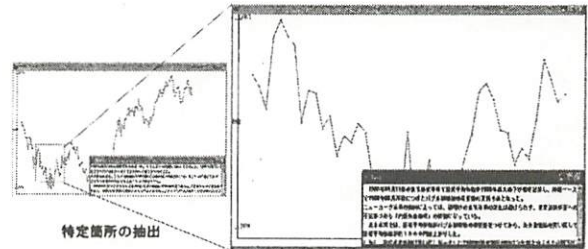


図 8: 実行例 (特定箇所の情報抽出)

## 4 まとめ

本研究では、異なるモダリティが協調することにより情報を効果的に提示する技術開発の一環として、グラフとテキストの異なる2つのモダリティ情報を用いた、グラフの表示状態に対応したテキストの表示を行った。これにより、ユーザにグラフを変更させ、ユーザの情報閲覧の焦点を判断し、その要望に対応した提示方法を提案した。今後の課題として、MuSTコーパスで用意されているタグが付与されていない文の中で、重要度が高いとする文には、新たにタグを追加し重要度を判断する基準とする等、グラフとテキストの情報がより協調する仕組みを工夫し、提示方法を自由に変化させることができるコンテンツの開発を進める。

## 参考文献

- [1] サイバーアシストコンソーシアムシンポジウム 2005 「人間中心の情報社会」の実現へ向けて—セマンティックコンピューティングの応用展開—, Oct 26 2005.
- [2] 長尾 確: アノテーションに基づくデジタルコンテンツの高度利用, IPSJ Magazine Vol.42 No.7 July 2001
- [3] 動向情報の要約と可視化に関するワークショップ, [http://www.kecl.ntt.co.jp/scl/workshop/must/must\\_index.html](http://www.kecl.ntt.co.jp/scl/workshop/must/must_index.html)
- [4] 松下光範, 加藤恒昭: 動向情報に基づく情報可視化の基礎検討, 第19回人工知能学会全国大会予稿集, 1E3-03, 2005.

# グラフの挙動を表すテキスト生成

奥村 奈穂子 (指導教員: 小林 一郎)

## 1 研究背景と目的

World Wide Web の普及により、複数の異なるモダリティの情報を利用して効果的に情報を伝えるマルチモーダル文書が利用されるようになった。そのような文書はひとつのモダリティが他のモダリティと協調する形で、読み手に対して効果的に情報を伝えている。このことをさらに進めた形態として、あるモダリティの情報を他のモダリティの情報によって表現することができれば、読み手に対してより効果的に情報を伝えることができると考える。本研究では、そのような複数のモダリティを横断する技術開発の一環として、グラフで表現される数値情報（具体的には「日経平均株価」）の動向を説明するテキストの生成手法を提案し、システムを開発することを目的とする。

## 2 システムの概要

システムの概要を図1に示す。

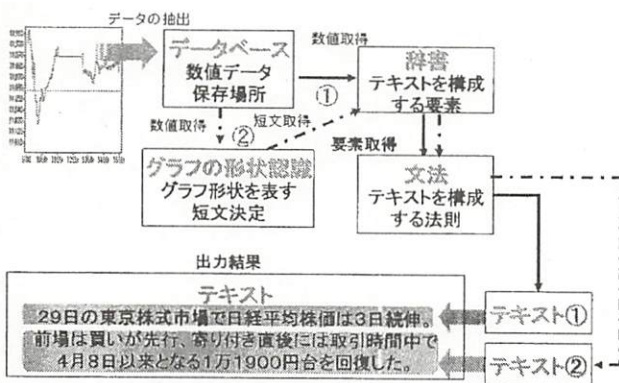


図1: システムの概要

以下、大まかな処理の流れを示し、システム各部の説明を行う。

## 3 システム構成要素

### 3.1 処理の流れ

本システムによって生成されるテキストは以下の2つのタイプに分類され、タイプごとにテキスト生成の処理の流れが異なる。

**タイプ①** グラフの形状を踏まえることなしに、データベースからの情報のみから生成できるテキスト。

**タイプ②** グラフの形状を踏まえて、かつデータベースからの情報から生成できるテキスト。

以下にタイプ②のテキスト生成の流れを示す。(タイプ①のテキストは以下の処理の流れにおいてstep2が省かれたものである。)

#### step1. データベースからの数値情報取得

当日のチャート数値情報と過去の始値・終値・高値・安値の数値情報を取得する。

#### step2. グラフの形状認識

線形最小二乗法を用いて、午前の相場である前場

と午後の相場である後場のグラフの形状を認識する。

#### step3. グラフの形状に対する語彙選択

step1 で得られた数値情報を言語表現にすることは、適切な語彙を選択することより可能である。またタイプ②のテキストではこれに加え、step2 で認識されたグラフ形状を踏まえ、グラフの挙動を表現するために適切な語彙を選択する。

#### step4. テキストを表現する文法選択

タイプ①のテキストでは、step1 で得られた数値情報を元にあらかじめ用意された文法を適切に選択。タイプ②のテキストでは、step3 で表現できる語彙に付随する時間帯、接続詞を適切に選択する。

### 3.2 データベース

システムへの入力情報は、日経平均株価 2005 年 7 月 25 日から 8 月 30 日までの分足データと始値・終値・高値・安値である。その数値データを管理する（データの追加・削除・変更を容易に操作するため、MS Access を用いてデータベースを構築した。

### 3.3 グラフの形状認識

図2に示す折れ線グラフの動向を視覚的に把握すると、「下がって、上がっている」と認識される。この様にチャートを視覚的に把握するために、本研究では、線形最小二乗法を用いて、チャートの近似曲線を作り、その近似曲線の振る舞いを捉えることにより、グラフの動向を言語で認識した。近似曲線は5次多項式で表現されており、この多項式の次数は、グラフの形状を表現している語彙の種類を実際のコーパス（約1ヶ月分の日経平均株価動向の解説記事）を分析することにより、その最適な次数を5次と導いた。5次多項式が表現する典型的な曲線の全体的な形状を11のタイプとし、その形状のパラメータの値のとり方により、さらに13種類の部分形状が導けるとした。この分類は、グラフの挙動を説明するために使われる表現の観点から導いた。任意の全体形状のタイプはどの部分形状を含むかが決まっているので、5次多項式で認識されたグラフの形状は、始めに分類された全体形状の特定のタイプを選別し、次に、その部分形状を数式的に解釈することにより最終的なグラフの形状を認識し、これを説明する適切な言語表現を選択する(表1参照)。

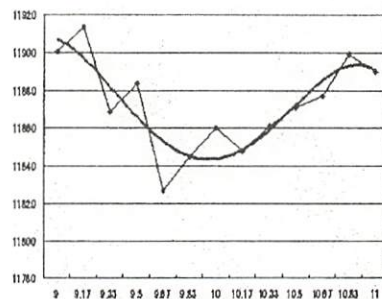


図2: 線形最小二乗法の適用例

表 1: 部分形状の数式的解釈とその諸表現

部分形状	特徴	原文+時間帯
	$ b2-b1 / MAX-MIN >0.4$	売りが優勢だった
	$ a1-a2 / max-min <0.7$	売りが広がった
	$ b2-b1 / MAX-MIN >0.4$	売りが優勢になる場面があった
	$ b2-b3 / b2-b1 >0.5$	
	$ a1-a2 / max-min <0.7$	
	$ fsh-a2 / max-min <0.7$	中ごろ過ぎにかけて
	$ fsh-a2 / max-min >0.5$	
	$ fst-a1 / max-min <0.2$	中ごろに
	$ fsh-a1 / max-min <0.2$	
	$ fsh-a1 / max-min <0.6$	中ごろ過ぎから
	$ fsh-a1 / max-min >0.45$	

3.4 辞書

2005年7月25日から8月30日までの27個のテキストを分析することにより、表現できる語彙を抽出した。部分形状で表現できる短文が55種(例:「売りが広がった」、「じり高歩調となった」、「反発」)、時間帯が9種(例:「前場」、「大引けで」)、接続詞が4種(例:「そして」、「なので」)である。

3.5 文法

上記27個のテキストを分析することにより、3種の文法テンプレートを生成した。その例を以下に示す。

- 〇日の東京株式市場で日経平均株価は〇終値は〇円〇銭(〇%)の〇万〇円〇銭だった。
- 〇日の東京株式市場で日経平均株価は〇終値は〇円〇銭(〇%)の〇万〇円〇銭で、〇円台を割り込んだ。

タイプ①のテキストはこのテンプレートからできるテキストとデータベースの数値情報のみからわかる短文からできる(例:「1日を通じて高い水準で推移した。」)。タイプ②のテキストは短文、時間帯、接続詞の適切な組み合わせ規則により生成される。その例を以下に示す。

- 時間帯によって先頭に「前場は、」、「後場は、」をつける。
- 部分形状によっては、時間帯によって「中ごろ過ぎにかけて」、「中ごろに」、「中ごろ過ぎから」などが短文の前につけられる。

4 実行例

システムの実行例を図3に示す。

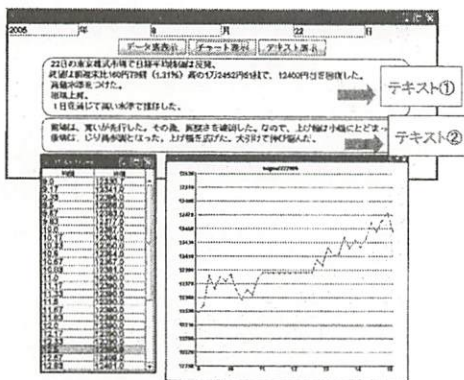


図 3: システムの実行例

図3の例では、2005年8月22日と入力すると、以下のようなテキストが生成された。(太字がシステムが判定したグラフの特徴。)

タイプ①のテキスト生成

22日の東京株式市場で日経平均株価は反発。終値は前週末比160円78銭(1.38%)高の1万2452円51銭で、12400円台を回復した。高値水準をつけた。相場上昇。1日を通じて高い水準で推移した。

タイプ②のテキスト生成

前場は、買いが先行した。その後、底堅さを確認した。なので、上げ幅は小幅にとどまった。後場は、じり高歩調となった。上げ幅を広げた。大引けで伸び悩んだ。

5 評価

システムの評価を表2に示す。

表 2: システムの評価

実際のコーパス	生成したテキスト		
状態を表現するテキスト	状態を表現するテキスト		
チャート+過去3日間のデータより判断可能	本文と一致	本文よりも詳細	不適切
119	107	76	11
状態に付随する時間帯	状態に付随する時間帯		
チャート+過去3日間のデータより判断可能	本文と一致	本文よりも詳細	不適切
61	59	42	0
数値	数値		
チャート+過去3日間のデータより判断可能	本文と一致	本文よりも詳細	不適切
181	176	0	1

表2より、状態を表現するテキストの適合率は、約89.8%である。しかし不適切な表現も出力されている。この原因としては、元のチャートの数値データを取得する際に、約10分間隔の特徴点をとってきてデータとし、さらに近似したことが考えられる。より精度の高いデータを用いることができるならば、生成される不適切な表現も減ると予想される。

6 まとめ

本研究では、あるモダリティで表現されている情報を他のモダリティで表現する一つの手法として、グラフとして表現される数値情報からその動向を説明するテキストを生成する方法を提案した。提案した手法は、グラフの形状で表現される統計データの概要を言語で表現することで、人にその内容を容易に把握させることができる。また、この手法は統計データの言語によるインデックス化を行うことができ、言語によって特定の統計データの事例を検索できる検索エンジンの中核技術として用いることが可能である。このような統計情報を言語で検索できる検索エンジンは、未だ存在していない。本研究は数値情報からテキストを生成するものであるが、このことと反対の技術(テキスト情報中に埋め込まれている数値情報から動向をグラフなどとして可視化する技術)として、現在、国立情報学研究所主催で「動向情報の要約と可視化」ワークショップ(NTCIR-5)が進められている[2, 3]。このような技術も取り入れ、双方向にモダリティの変換を行う技術を開発することも今後の課題としたい。

参考文献

[1] 小林一郎, “グラフ情報の自然言語表現に関する研究”, 日本ファジィ学会誌, Vol.3. No. 12, June, pp.406-416, 2000.  
 [2] [http://www.kecl.ntt.co.jp/scl/workshop/must/must\\_index.htm](http://www.kecl.ntt.co.jp/scl/workshop/must/must_index.htm)  
 [3] 松下光範, 加藤恒昭, “動向情報に基づく情報可視化の基礎検討”, 第19回人工知能学会全国大会予稿集, 1E3-03, 2005.

# 音声・マルチモーダル情報を用いた情報家電との対話への取り組み

清水 美絵 (指導教員: 小林 一郎)

## 1 はじめに

日常生活において情報家電が普及し、ネットワークを通じて様々な家電を利用することができる時代が到来している。多くの利便性を享受できることと共に、予想されることの一つとして、それらが持つ複雑な機能を使いこなせない人は、現在と同じく多く存在するというのである。このことから音声で対話することができるような優れたインタフェースの必要性が高まってくると予想される。これらのことを踏まえ、本研究においては、情報家電と音声対話することを目指し、音声対話技術コンソーシアム [1] における Galatea プロジェクト [2] にて開発された、擬人化音声対話エージェントのツールキット Galatea Toolkit を利用した音声・マルチモーダル対話の可能性について考察を行う。

## 2 Galatea システム概要

Galatea システムは擬人化音声対話エージェントのツールキットである。顔、声、音声合成テキスト、認識文法、対話の流れなどをカスタマイズ可能で、これを用いてエージェントを作成する。このシステムは、顔画像合成モジュール (FSM)、音声認識モジュール (SRM)、音声合成モジュール (SSM) を統合し XISL と呼ばれる対話記述言語 (音声・マルチモーダル対話管理用マークアップ言語) で記述されたシナリオに沿ってユーザと対話を実現している。図 1 に Galatea システムの概要を示す。

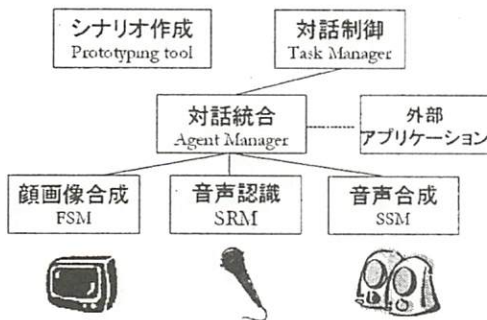


図 1: Galatea システム概要

以下、各部について説明していく。

### 2.1 顔画像合成モジュール (FSM)

図 2 に示すように、正面方向から撮影した 1 枚の写真から擬人モデルを生成できる。まずワイヤーフレームモデルと呼ばれる多面体近似モデルと顔写真を整合し、目や歯のモデルをうめ込むことで作成される。顔の表情や滑らかな口の動きはキーフレーム間を線形補間することによりアニメーションとして生成される。表情は怒り、喜び、悲しみ、嫌悪、恐れ、驚きの 6 つが操作可能でより人間らしい対話のための自然な瞬きや、他のモジュールと連携させることにより音声と唇の動きを制御する機能を持つ。



図 2: 顔画像合成モジュール

### 2.2 音声認識モジュール (SRM)

音声認識の流れは入力された音声を信号処理し、音響モデルと単語辞書、言語モデルの情報を頼りにして認識エンジンで検索し結果を出す。音響モデルとは調音により徐々に移り変わる音響的特徴を、簡単な確率モデルで表現されているもの。また、単語辞書は単語と音素列を対応付けを、言語モデルは単語の連鎖を規定している。図 3 に音声認識モジュールを示す。

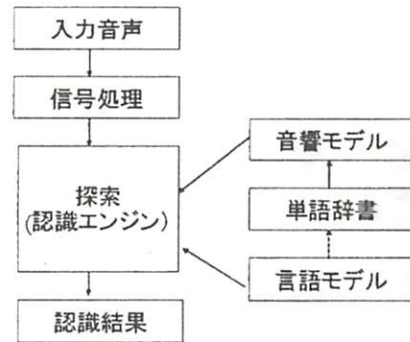


図 3: 音声認識モジュール

### 2.3 音声合成モジュール (SSM)

漢字仮名混じり文で表記された日本語テキストを合成音声に変換することができる。入力された日本語テキストは形態素解析され、「読み」や「アクセント」情報を付与、韻律が生成される。できた韻律情報を元に音声の合成波形を生成し音声として出力する。予め Galatea には男 2 名、女 1 名の基本話者モデルが提供されており、本研究では男性の話者モデルを利用している。音声合成モデルを学習する過程として音声収録してデータベースの整備、次にモデル学習をする。データベースの整備とは読み誤り、ポーズの位置、アクセントの位置をチェックし、発生にあった言語や韻律情報の修正を行うことである。次に行うモデル学習は収録した音声データから、各音素のモデルを作成し前後環境や言語的な構造に依存した精度の高いモデルを作る。

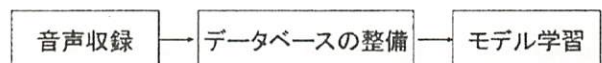


図 4: 音声合成モデル作成過程

## 2.4 シナリオ文生成

Windows 版 Galatea システムにおけるシナリオ文作成には、豊橋技術科学大学新田研究室でその仕様策定が行われているマルチモーダル対話記述言語 XISL(eXtensible Interaction Scenario Language) を利用する。XISLとは3つのモジュール(音声認識、顔画像合成、音声合成モジュール)を統合管理し、ひとつのシナリオとして記述するためのマークアップ言語である。XISLを用いたシナリオ作成の支援ツールとして、prototyping tool が用意されている。図5に prototyping tool を用いたマルチモーダル対話シナリオ作成の例を示す。これを用いて、音声認識のための言語モデルや単語の辞書を登録したりシナリオの流れを作成、画像を設定することができる。

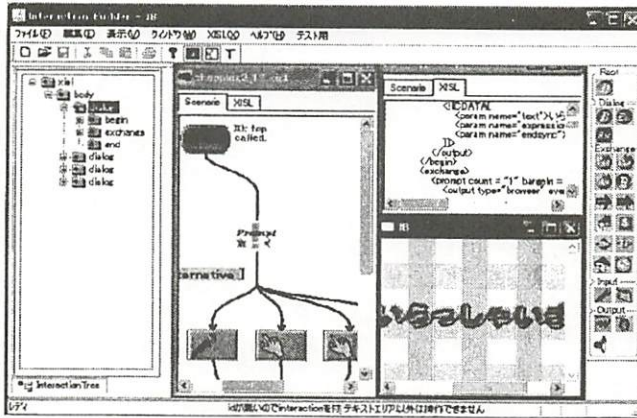


図 5: prototyping tool を用いたシナリオ作成例

像表示によるインタラクションもできるため、2つのモダリティを対象にしたインタラクションの可能性を考える。画像によるインタラクションは、種類別で考えるとキッチン系においてレシピなどの表示、AV機器のテレビやDVDレコーダではテレビ番組表、また衣服住系は操作方法を画面で示すなどが考えられる。本研究では、この中から冷蔵庫に対する対話の一例となるシナリオを作成し、Galatea システムに実装した。このシナリオの特徴はシステムがユーザを誘導していくタイプではなく、なるべく普通の会話に近づけるよう多くの分岐をもっている。また、ユーザは知りたい情報を得たらいつでも対話を終了できるように、ユーザからの発話で「ありがとう」を音声認識すると対話を終えることができるように作成した。このシナリオを Galatea システムで実行している例を図7に示す。



図 7: 実行例

## 3 情報家電との対話への取り組み

### 3.1 家庭電化製品

家庭電化製品には様々な種類が存在する。近い将来、それらが情報家電となり音声と他のマルチモーダル情報を用いて対話を行うことになる。その対話の内容や形態は家電それぞれによって異なることが予想される。これを家電の種類(キッチン系、衣服住系、AV機器系の3種類を対象とする)で分類したものを図6に示す。

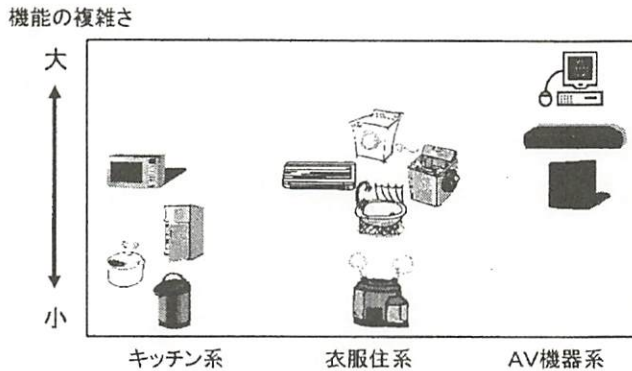


図 6: 家電の種類と機能の複雑さの関係

図6に示す関係より、複雑さや種類の近いものほど似たシナリオになることが予想できる。Galatea システムでは音声のみのインタラクションだけでなく、画

## 4 まとめ

本研究では、音声・マルチモーダル対話への取り組みとして、Galatea システムの実装を行った。また、情報家電の対話可能性を考え、家電の種類による対話シナリオの違いを考察した。そして、そのシナリオを元に Galatea システムで XISL による、冷蔵庫シナリオを作成実行した。冷蔵庫シナリオの実行には HTML 画像を作成して出力することにより、音声だけでなくマルチモーダルなインターフェイスを実現できた。今後の課題としてはさらに情報家電との対話可能性を考察し、より分かりやすく使いやすいインターフェイスを作っていく。また、実際の外部アプリケーションとの連携を実現させたい。

## 参考文献

- [1] 音声対話技術コンソーシアム <http://www.astem.or.jp/istc/>
- [2] Galatea Project <http://hil.t.u-tokyo.ac.jp/galatea/index-jp.html>
- [3] Life-Like Characters Tools, Affective Functions, and Applications Series: Cognitive Technologies Prendinger, Helmut; Ishizuka, Mitsuru (Eds.) Springer, 2004.

# ユーザの目的達成を支援する知識をもった操作インタフェースの開発

大西 可奈子 (指導教員: 小林 一郎)

## 1 研究背景と目的

近年、コンピュータの価格低下やインターネットの普及により、コンピュータを使用する機会は格段に増加している。それに伴い、さまざまな機能を備えたアプリケーションソフトウェアが提供されるようになってきた。これは一見すべての人に利便性を与えているように見えるが、実際は多くの人がソフトウェアを使いこなせていない。近い将来、情報家電が家庭に入ってきた時に、それぞれの複雑な操作をひとつの情報集約端末(例えば癒しの効果を持つペット型ロボットなど)を使って自然言語で操作できる様なインタフェースの必要性が高まると言える。この様なことを踏まえ、本研究では、言語指示で複数のアプリケーションソフトウェア(本研究においては MS Word, MS Excel, MS Outlook を対象にする)を操作するためのインタフェースの開発を目的とする。

## 2 システム概要

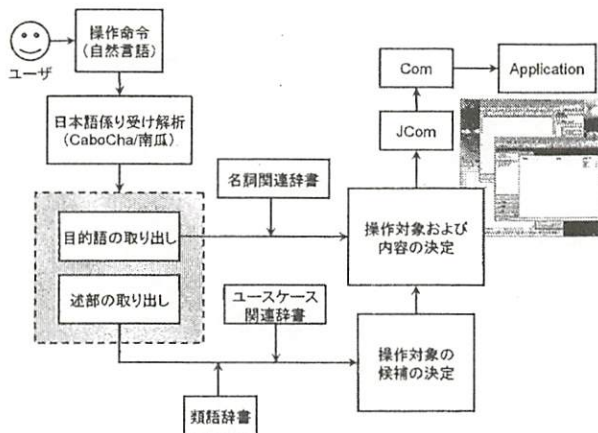


図 1: システム概要

### 2.1 入力文解析

奈良先端科学技術大学院大学で開発された日本語係り受け解析器 CaboCha/南瓜 [1] を使って、入力文を形態素(単語)の列に分解、品詞を見分け解析を行う。これを用いてシステムは、「メールを送りたい」という入力文に対して、述部として「送る」を、目的語として「メール」を抽出する。

### 2.2 アプリケーションソフトウェアのユースケース

本研究では、アプリケーションソフトウェアコマンドに直接結び付く操作のユースケースを抽出し、それらはシステム内では特定の動詞で表現されている。ユーザからの入力文中において、それらユースケースを表現する特定の動詞を発見すると、ユースケースに結び付いたコマンドが実行されアプリケーションソフトウェアが操作される。

### 2.3 操作対象の候補決定

先行研究 [2] においては、ユースケースを表現する特定の動詞を含まないユーザからの様々な発話によっても、アプリケーションソフトウェアを操作可能にするため、述部に対する類語辞書を導入し、ユーザの発話をユースケースに結び付く特定の動詞に柔軟に結び付ける工夫を行った。本研究では、複数のアプリケーションソフトウェアを同時に操作することを目的とするため、ユースケースオントロジーをシステムに持たせる。このユースケースオントロジーには、各ユースケースがどの程度、あるアプリケーションソフトウェアに從属している状態であるかを示すユースケース関連辞書が備わっている。これにより、選択されたユースケースに基づき、操作対象となるアプリケーションソフトウェアをある程度絞り込むことができる。

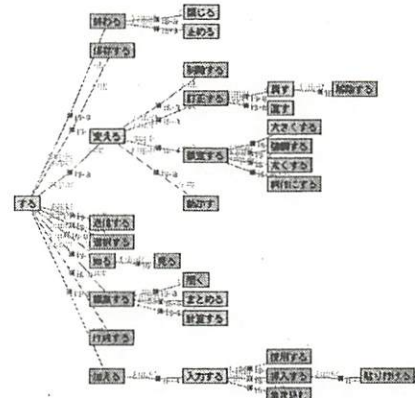


図 2: 対象ソフトウェアのユースケースオントロジー

上記オントロジーの抽出には、溝口らによって開発された AFM (Activity First Method) [3] を考慮して行い、オントロジーの作成には法造 [4] を用いた。

### 2.4 操作対象および内容の決定

動詞によって挙げられた複数のユースケースの候補をひとつに絞り込むため、本研究では更に名詞関連辞書を導入した。これは、ある名詞が各アプリケーションソフトウェアにどの程度関連しているかを数値化した値を保持した辞書で、この数値が高いほどその名詞は各アプリケーションソフトウェアとの関係が強いと判別する。名詞関連辞書を使って、候補であるユースケースに名詞関連度を付与し、この値が最大となるものを最終的に操作対象となるユースケースとする。これにより複数のアプリケーションの競合は防止され、操作の代替も可能となる。例えば「メールを書きたい」という命令を受け取った場合、システムは「書く」と「メール」に最も関連するメールエディタ、すなわち Outlook を起動しようとするが、Outlook はなんらかの理由で起動できない状態とする。そこでシステムは代替するアプリケーションソフトウェアを探すため、メールの代替となるものを推論し、「図表入りテキスト」を扱えるエディタを代わりに探す。そして上位概念であるエディタからサブクラスを探し、「図表入りテキスト」



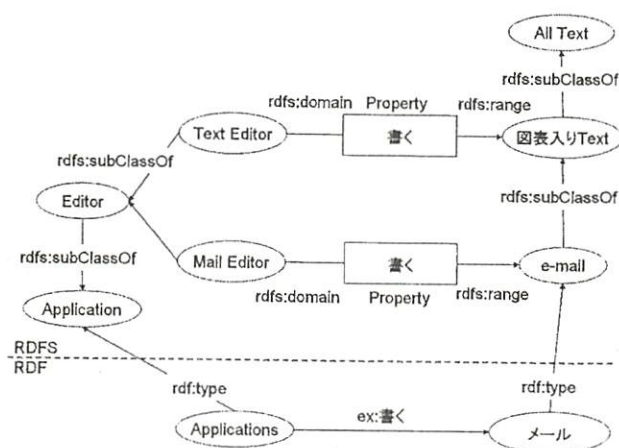


図 3: ドメインオントロジー 1

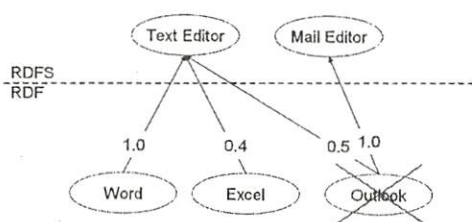


図 4: ドメインオントロジー 2

ト」を扱えるテキストエディタを発見する。それによりテキストエディタとしての機能を持つものとして、3つのアプリケーションソフトウェアが存在することがわかる。ここで Outlook は使用不可であり、Word と Excel のどちらを起動するかが問題となるが、同様に2つにもユースケース関連辞書と名詞関連辞書を基に從属関連度を付与してやると從属関連度の大小関係より Outlook の代替として使用するものは Word に決定される。

## 2.5 アプリケーションソフトウェア実行

本システムにおいては、判別されたユースケースを実際にアプリケーションソフトウェアのコマンドに結び付けるために JCom から Com を利用している。JCom とは、Java から Windows の COM (Component Object Model) を利用できるミドルウェア [5] で、Word, Excel など既存の COM の機能を Java から利用できる。また COM とは、Microsoft 社が提唱する部品化されたプログラムを作成・利用するための基盤となる技術仕様である。

## 3 処理概要

アプリケーションソフトウェアが立ち上がっていない状態で、ユーザから「図や絵を入れて招待状を書きたい」と命令された場合を例に、本システムの処理概要を説明する。

### 3.1 状況判断と解析

最初に、どのアプリケーションソフトウェアが起動して、どのような状態であるかを把握する。現在は行為を及ぼす対象が存在していないので、最適と思われるアプリケーションソフトウェアを判断し、起動実行すればよい。また、形態素解析により動詞「書く」、名詞「図」「絵」「招待状」を抽出する。各アプリケー

ションソフトウェアとの從属関連度は  $X_{word}$ ,  $X_{excel}$ ,  $X_{outlook}$  という変数で監視する。

### 3.2 動詞の処理

從属度を持つ動詞に「書く」がなく、行為を及ぼす対象が存在していないので、ユーザの発話内容オントロジーを参照し、「書く」は「作成する」のサブカテゴリであると推論し、「作成する」の從属度を監視変数にそれぞれ加える。「作成する」はどのアプリケーションソフトウェアにも等しく從属するので、監視変数はどれも 1.0 となる。

$$X_{word}=1.0 \quad X_{excel}=1.0 \quad X_{outlook}=1.0$$

### 3.3 名詞の処理

名詞「図」「絵」「招待状」の関連度を各監視変数に加える。それぞれの関連度は以下ようになる。

名詞の関連度	Word	Excel	Outlook
図	0.4	0.6	0.0
絵	0.4	0.3	0.3
招待状	0.6	0.0	0.4

これらを各監視変数に加えた結果、

$$X_{word}=1.0+0.4+0.4+0.6=2.4$$

$$X_{excel}=1.0+0.6+0.3+0.0=1.9$$

$$X_{outlook}=1.0+0.0+0.3+0.4=1.7$$

となり、從属関連度の大小関係より、システムは Word の「作成する」ユースケースから Word を起動し新規文書を作成する。

続けてユーザから命令を受け取った場合も同様の計算により、実行するユースケースを決定する。ただし、前回起動したアプリケーションソフトウェアに対応する変数には 0.1 が加算される。これは直前に使用したアプリケーションソフトウェアと、その他の候補ユースケースであるアプリケーションソフトウェアとに差異をつけるためである。

## 4 まとめと今後の課題

本研究が提案する手法を利用することにより、ユーザの命令に対して柔軟な対応が可能になり、ひとつのインタフェースで複数のアプリケーションソフトウェアが動かせるようになった。

今後の課題として、対象ドメインを拡張し、ドメインオントロジーの拡充とともに、状況判断機構を持ち、より複数のアプリケーションとの連携を図ることができると柔軟な操作インタフェースとして開発を進めていくつもりである。

## 参考文献

- [1] 奈良先端科学技術大学大松本研究室: CaboCha/南瓜 <http://chasen.org/taku/software/cabocha/>
- [2] 言葉によるアプリケーションソフトウェアの操作, 菊池友里子, お茶の水女子大学理学部情報科学科卒業論文, 3月, 2005.
- [3] タスク・ドメインロールに基づくオントロジー構築ガイドシステムの設計と開発—石油精製プラントを例として—, 石川誠一, 久保成毅, 古崎晃司, 來村徳信, 溝口理一郎, 人工知能学会論文誌, Vol17, No.5, pp585-597, 2002.
- [4] オントロジーエディタ法造 <http://www.hozo.jp/>
- [5] JCom(Java-COM Bridge) <http://sourceforge.net/projects/jcom>
- [6] 知の科学「オントロジー工学」, 溝口理一郎, 人工知能学会編集, 2005.

## 逆畳み込み作用素による画像の修復

田中佳花 (指導教官: 金子 晃)

## 1 はじめに

写真を撮る際、撮影像までの距離に応じて、レンズ系を前後に移動させる焦点調節を行う。しかし、この操作が上手にいかないとフィルム上にブレ（いわゆるピンボケ）が生じ、鮮明な画像を得ることが出来ない。これを補正するため、本研究では逆畳み込み作用素を用いる。その準備段階として、まず画像に様々なフィルタをかけることにより生ずるぼかしの性質を調べた。次に、こうしてぼかした画像を逆畳み込みフィルタで修復し、その統計データから最適なパラメータ値を導くことを試みた。最後にこれらの考察を基に種々のフィルタを用い、実際のピンボケ写真の修復を試みた。

## 2 ぼかしフィルタ

ぼかしフィルタとして、テント型フィルタ (図 1)、円錐型フィルタ (図 2)、高原型フィルタ (図 3)、Gaussian フィルタ (図 4) を用意する。

Gaussian フィルタとは、2次元ガウス分布に従う重み付きマスクを用いた線形空間フィルタの一つであり、円対称で、以下のようにフィルタの総重みが1になるよう正規化したものを用いる:

$$g_{x,y} = \frac{1}{2\pi a} \exp\left(-\frac{x^2 + y^2}{2a^2}\right)$$

ここで、 $a$  とフィルタのサイズをパラメータとし、自由に選択できるものとする。このパラメータが大きいほど画像はぼかしがより激しくなる。しかし、フィルタのサイズは17ピクセルより小さいと復元画像上にアーティファクトが出てしまうため、17ピクセル以上とする。

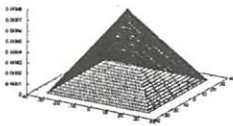


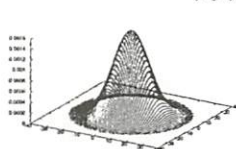
図 1: テント型



図 2: 円錐型



図 3: 高原型

図 4: Gaussian  
( $a=10, \text{size}=69$ )

## 3 畳み込み演算

畳み込み演算とは、関数  $f$  の平行移動と関数  $g$  を重ね足し合わせる二項演算であり、以下のように定義する:

$$(f * g)_{m,n} = \sum_{m'=0}^{M-1} \sum_{n'=0}^{N-1} f_{m',n'} g_{m-m',n-n'}$$

ここで、 $f$  を元画像、 $g$  をぼかしフィルタ、 $M, N$  をそれぞれ縦、横のピクセル数とすると、画像のぼかしはこ

の畳み込み演算で表現できる。よく知られているように

$$\mathcal{F}(f * g) = \mathcal{F}(f)\mathcal{F}(g)$$

という関係にある。ここで、 $\hat{g} = \mathcal{F}(g)$  はフーリエ変換を表し、以下のように定義される:

$$\hat{g}_{u,v} = \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} g_{m,n} \exp\left(-\frac{2\pi i nu}{N}\right) \exp\left(-\frac{2\pi i mv}{M}\right)$$

逆フーリエ変換も同様に定義する:

$$g_{u,v} = \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \hat{g}_{m,n} \exp\left(\frac{2\pi i nu}{N}\right) \exp\left(\frac{2\pi i mv}{M}\right)$$

すると、逆畳み込み演算は次の式で表すことができる:

$$f = \mathcal{F}^{-1} \left[ \frac{\mathcal{F}(f * g)}{\mathcal{F}(g)} \right]$$

この公式を利用することにより、ブレを生じた画像を修復することを考える。そのための準備として、適当に選んだフィルタにより自らぼかした画像を上記の公式で元に戻すことを試みる。

しかし、ぼかしフィルタをフーリエ変換すると高周波になるにつれて0に近い値 (図 5 の中央付近) になるため、それで割り算したものは戻したい画像の値に反して、非常に大きな値となってしまう、理論通りには復元できない (図 8)。これを防ぐため、高周波領域では  $\mathcal{F}(g)$  を他の値に置き換えて割るという、逆問題でいわゆる正規化の手法が必要となる。このため、どの値まで  $\mathcal{F}(g)$  で割るかを設計しなければならない。なお、様々なぼかしフィルタで実験した結果、ぼかしフィルタが滑らかになればなるほど、0 に近づくのが速く、正規化が難しいことがわかった。

例: Gaussian フィルタ ( $a=10, \text{size}=17$ )

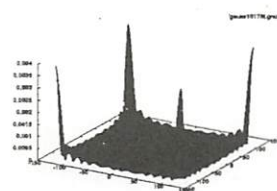


図 5: ぼかしフィルタのフーリエ変換



図 6: 原画像



図 7: ぼかし画像

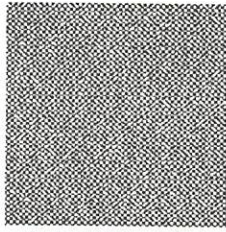


図 8:公式通り計算した復元画像

#### 4 正則化の実装

正常な元画像にぼかしフィルタをかけ、その画像を復元することを実装した。ある閾値  $\rho$  を設け、 $|F(g)|$  の値が  $\rho$  未満のところでは  $F(g)$  を  $\rho$  に置き換えたもので割り、その  $\rho$  を 0.00001 から 0.0002 まで 0.00001 刻みで動かし、画像復元を観察する。なおその際、原画像と復元画像との  $L_1$  ノルムが最小になる  $\rho$  を最適な  $\rho$  とした。ここで  $L_1$  ノルムは以下のように定義する:

$$\|a - b\|_{L_1} = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} |a_{m,n} - b_{m,n}|$$

逆畳み込み演算で使われるフーリエ変換は高速フーリエ変換アルゴリズム FFT で行った。FFT では、画像のサイズを 2 のべき乗にしなければならないため、原画像を拡張する必要がある。一定のピクセル値で拡張してしまうと、原画像と拡張部分に不連続性が生じ、実行結果にアーティファクトが出てしまう。そのため、原画像をミラー状に拡張させ、アーティファクトが出るのを防いだ。

#### 5 実行結果と考察



図 9:ぼかし画像



図 10:復元画像



図 11:  $|F(g)|$  の分布

図 9~図 11 は、Gaussian フィルタ ( $a=10, size=17$ ) での  $\rho=0.0001$  の実行結果である。図 11 は、元来の  $F(g)$  を用いた部分 (0.0001 以上) を白、 $\rho$  で置き換えられた部分を黒で表したものである。

他のぼかしフィルタでも  $\rho=0.0001$  前後においてほぼ同様の結果を得ることができた。ぼかしが激しくなるほど復元画像にアーティファクトが目立ち、原画像通りに戻すことが難しいこともわかった。

#### 6 ぼかし画像修復の試み

ピンボケした実写画像も Gaussian フィルタをかけたものと似たような画像ではないかと予想する。そこ

でその準備段階として、原画像に Gaussian フィルタをかけてぼかし画像を作成し、そのぼかし画像から元の Gaussian フィルタを推定し、半自動的に修復することを試みた。まず、sharpness 値を以下のように定義する:

$$S = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} [(f_{m+1,m} - f_{m,n})^2 + (f_{m,n+1} - f_{m,n})^2]$$

ここで、 $f$  を画像とする。画像のぼかしが激しくなる (Gaussian フィルタのパラメータ値を大きくする) ほど、この sharpness 値が小さくなっていく (図 12) ことから sharpness 値が最大になるとき、原画像に戻せるのではないかと考えた。図 13 は、 $a=10, size=19$  のぼかしフィルタをかけた画像を修復した結果である。残念ながら、この方法では原画像にもどすことが出来なかった。その他にも、ぼかし画像と修復画像との  $L_1$  ノルムが最小になる修復画像が原画像に近い画像になるのではないかと考えたが、この方法でも原画像にもどすことが出来なかった。

しかし、元のパラメータ値 ( $a, size$ ) 前後の値で逆畳み込み演算をしても、原画像に近い画像を得ることが出来た。このことから今後、ユーザが自由にパラメータを動かすことによってぼかし画像を修復できるようにしていきたい。

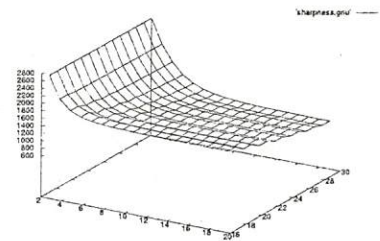


図 12:ぼかし画像の sharpness

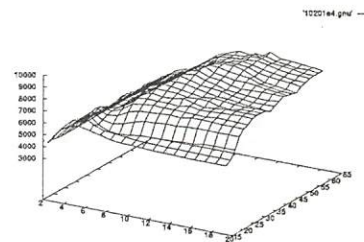


図 13:逆畳み込み演算後の sharpness

#### 7 まとめと今後の課題

正常な原画像にぼかしフィルタをかけた画像の復元を実装し、正規化パラメータの最適値を導きだした。今後、自動的に元のぼかしフィルタを探し出せるようにし、最終的に、ピンボケした実写画像の修復に対応させたい。

#### 参考文献

- [1] Bernd Jahne.: "Digital Image Processing," Springer, 2002
- [2] <http://www.owl.net.rice.edu/elec431/project95/lords/>
- [3] グロエッチュC. 数理科学における逆問題, サイエンス社, 1996年

# S字型風車まわりの流れの数値的研究

桑名 杏奈 (指導教員: 河村 哲也)

## 1. はじめに

風力エネルギーは、有害な廃棄物を出さないという点で地球環境問題の抑制に効果的である。尽きることのない風力エネルギーを得るために、風車は欠かせない道具の一つである。

本研究ではS字型風車について数値シミュレーションを行い、風車に働く力や風車の周りの流れ場などを検証した。S字型風車は、構造が単純、強風時でも回転音が静か、軸を回転させる力が強い、風向きに依存せず弱風でも回転を始められる等の特長があり、揚水などに利用される。

## 2. モデル化

### 2.1 計算領域

風車の回転半径を4、高さを2とし、計算領域には風向き方向に24、スパン方向に20、高さ方向に8の直方体をとった。(図2.1) はじめにxy平面(2次元)でのシミュレーションを行い、後に3次元に拡張した。

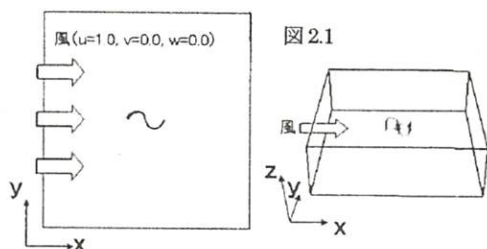


図2.1

### 2.2 格子

風車付近の流れを正確に計算するため、風車に沿った、風車に近いほど細かい格子を考える。(図2.2)

2次元計算では格子数は81×65とした。また、3次元ではこの格子を高さに41積み上げたものを用いた。

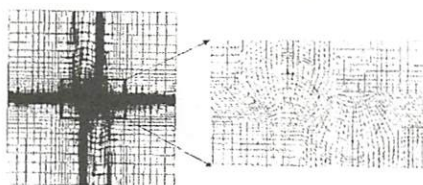


図2.2

## 3. 計算方法

### 3.1 基礎方程式

大気の流れは非圧縮性流体とみなせるので、連続の式(3.1)と、運動方程式として非圧縮性 Navier-Stokes 方程式(3.2)を利用する。

$$\nabla \cdot \mathbf{V} = 0 \quad (3.1)$$

$$\frac{\partial \mathbf{V}}{\partial t} + (\mathbf{V} \cdot \nabla) \mathbf{V} = -\nabla P + \frac{1}{Re} \Delta \mathbf{V} \quad (3.2)$$

$\mathbf{V}=(u,v)$ : 速度ベクトル、 $t$ : 時間、 $P$ : 圧力

$Re$ : レイノルズ数。本研究では $Re=2000$ とした。

### 3.2 解法

式(3.1),(3.2)を風車に固定した回転座標で表し、それを圧力項とそれ以外の項を分離して計算する、フラクショナル・ステップ法(FS法)を利用して解いた。

### 3.3 差分

$Re$  が大きいので、式(3.2)の左辺第2項(対流項)には3次上流差分を、その他には中心差分を用いた。

## 4. 風車に働く力

風車の効率を調べるため、次の用語が使われる。

$\lambda$  (周速比): 風車の回転速度/風速

$TR$  (トルク): 風車が回転する力。

$TR < 0$  のときは、回転を妨げる方向に力が働いている。

$Ct$  (トルク係数): 無次元化した  $TR$ 。

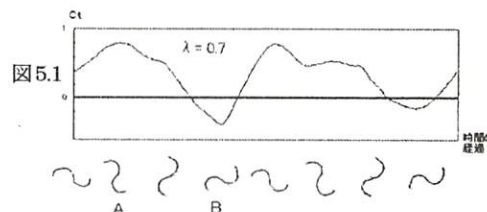
$Cp$  (パワー係数):

風の中から取り出すことのできるエネルギーの割合。

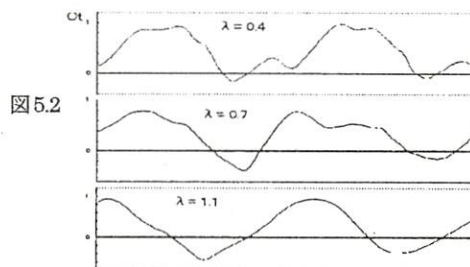
## 5. 結果と考察

### 5.1 トルク

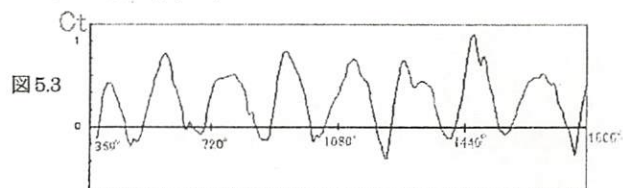
風車の角度により、 $Ct$  の大小が変わる。(図5.1) 風車がAのような位置にあるときには  $Ct$  すなわち回転力が大きく、Bのような位置では  $Ct$  が小さいことがわかる。



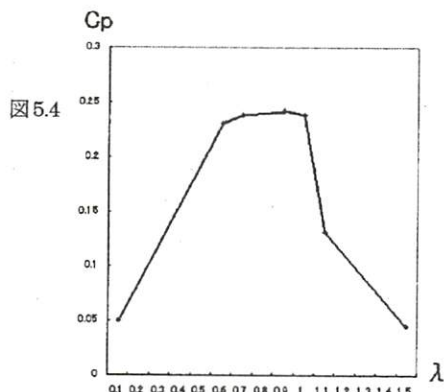
$\lambda$  を変えてもグラフが同じような形をしているため、 $Ct$  の変動の仕方は  $\lambda$  によらず  $\pi$  周期であることが分かる。(図5.2)



$Ct$  は周期的に変化する。図5.1、図5.2では1周する間の  $Ct$  を調べたが、4周する間の  $Ct$  を調べると周期的に変化している様子がよくわかる。(図5.3)



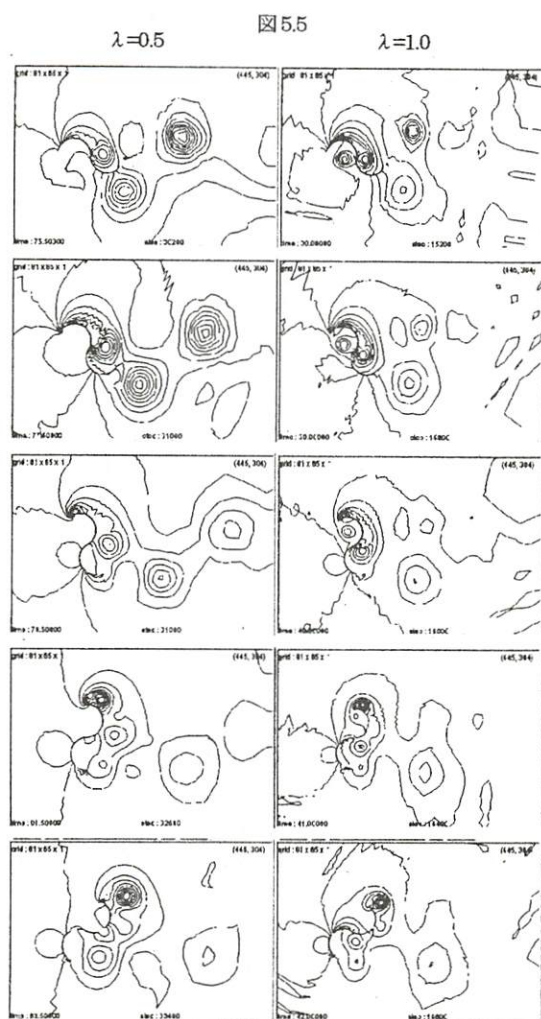
一般的に、S字型風車では出力  $C_p$  が最大になる  $\lambda$  は 0.7、出力がなくなる  $\lambda$  が 1.7 程度と言われている。それを確かめるため  $\lambda$  と  $C_p$  (時間平均値) の関係をグラフにした。(図 5.4)  $\lambda=0.6 \sim 1.0$  で高い値になっている。また、1.0 を過ぎると  $\lambda$  の値は低下し、 $\lambda=1.7$  に近づくと  $C_p$  の値が 0 に近づき、出力が小さくなっていくことがわかる。



## 5.2 流れ場

図 5.5 は圧力の等圧線を描いた図である。同心円状に見える部分は渦で、渦の内部では圧力が低くなっている。

ブレードの先端と凸部から渦ができる様子がわかる。 $\lambda=0.5$  と  $\lambda=1.0$  を比較すると、 $\lambda$  が大きい ( $\lambda=1.0$ ) ほど細かい渦がたくさんでき、風車の凹部にも渦が入り込んでいる。



## 6. 3次元計算

3次元に拡張し、トルクや流れ場を考察したい。図 6.1、図 6.2 は3次元での  $\lambda=0.6$  のときの速度ベクトルである。風車の付近で流れが乱れ、渦を作っている様子がわかる。

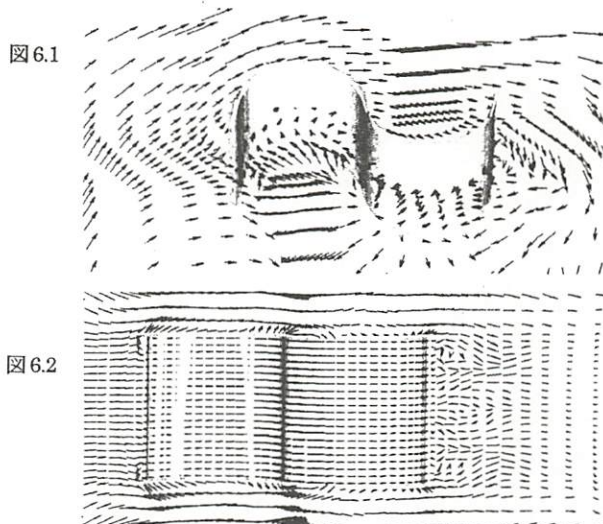


図 6.2 を見ると、ブレードの上下から風が逃げている様子が見える。風車の上下にふたを付けると風が逃げるのを防ぐことができるため、ブレードを押す力が大きくなると言われる。このように風車の形を変え、より効率のよい風車を考察したい。

## 7. まとめと今後の課題

本研究では、風車は機械的に同じ角速度で回転させたが、実際の風車は風が及ぼす力などによって回転の速さが変わる。そのようなこともふまえて、より現実に近い条件で計算し、実際の実験結果と比較したい。

時間変化に沿って状態が変わっていくため、紙面など動かない媒体上にわかりやすく結果を表現するのが難しい。結果を表現する紙面や画面は2次元であるため、3次元の風車では更に難しくなる。アニメーションなどを利用しつつ、見やすい表示の仕方を考えたい。

風力エネルギーは、自然の風を利用した安全でクリーンな動力源であるとはいえ、騒音問題、電波障害、鳥問題、景観問題などの欠点もある。また、強風が吹けば破損の危険性もあり、風車が回るも止まるも風の吹き具合次第ということで、動力が計画的に得られないという問題もある。現実に近い条件でシミュレーションを行うとともに、こうした風力エネルギーの現状を学び、地球に優しいエネルギーの供給、ひいては地球環境問題の緩和について、考えていきたいと思う。

### 7. 参考文献 (著者の五十音順)

- [1] 石松克也、篠原俊夫、詫磨史孝：  
『サボニウス風車に関する数値計算』  
日本機械学会論文集 (B編) 1994
- [2] 大槻和代：『鉛直軸型風車周りの流れの数値的研究』  
お茶の水女子大学大学院修士論文 2004
- [3] 河村哲也：『流体解析 I』朝倉書店 1996
- [4] 河村哲也：『エネルギーと風車』山海堂 2003

## 鉛直軸直線翼型風車における流れのシミュレーション

水上 洋子 (指導教員: 河村 哲也)

## 1 はじめに

近年, 地球温暖化現象が問題視される中, 風力発電がクリーンな発電法として注目されている. 風力発電に用いる風車を設計する際, 風車周りの流れを数値シミュレーションにより解析することは有用であると考えられる.

風車発電を目的とする風車は高速回転が要求されるため, 風車のブレードに生ずる揚力を利用して回転するものが適している. また, 風向の変化にかかわらず回転するためには, 風向に垂直な回転軸をもつ鉛直軸型の風車が有利である.

そこで本研究では, 発電に利用される風車の中から Fig.1 に示すような鉛直軸直線翼型風車を取り上げ, その周りの流れを解析し性能の評価を行うことを目的とした.

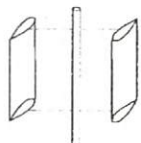


Fig.1 鉛直軸直線翼型風車

## 2 解法

## 2.1 基礎方程式

回転座標系で表した 2 次元の連続の式および非圧縮性 Navier-Stokes 方程式は以下ようになる.

$$\frac{\partial U}{\partial X} + \frac{\partial V}{\partial Y} = 0$$

$$\frac{\partial U}{\partial t} + U \frac{\partial U}{\partial X} + V \frac{\partial U}{\partial Y} - \omega^2 X + 2\omega V = -\frac{\partial p}{\partial X} + \frac{1}{\text{Re}} \left( \frac{\partial^2 U}{\partial X^2} + \frac{\partial^2 U}{\partial Y^2} \right)$$

$$\frac{\partial V}{\partial t} + U \frac{\partial V}{\partial X} + V \frac{\partial V}{\partial Y} - \omega^2 Y - 2\omega U = -\frac{\partial p}{\partial Y} + \frac{1}{\text{Re}} \left( \frac{\partial^2 V}{\partial X^2} + \frac{\partial^2 V}{\partial Y^2} \right)$$

$X, Y$ : 回転座標系での位置

$U, V$ : 回転座標系での速度

$\omega$ : 風車の回転角速度

静止座標  $x, y$  とそれに対して角度  $\theta$  で傾いた回転座標  $X, Y$  との関係を図 2 に示す. 回転座標系で表した式を用いることで, 物体に固定した格子による計算が可能になる. 静止座標と回転座標との間には以下の関係がある.

$$x = X \cos \theta + Y \sin \theta, y = -X \sin \theta + Y \cos \theta$$

$$X = x \cos \theta - y \sin \theta, Y = x \sin \theta + y \cos \theta$$

また, 静止座標における速度  $u, v$  と回転座標における速度の間には以下の関係がある.

$$u = U \cos \theta + V \sin \theta + \omega y, v = -U \sin \theta + V \cos \theta - \omega x$$

$$U = u \cos \theta - v \sin \theta - \omega Y, V = u \sin \theta + v \cos \theta + \omega X$$

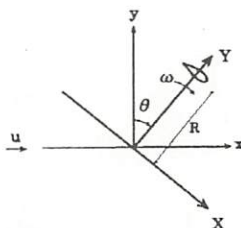


Fig.2 座標系と問題設定

## 2.2 差分方法

基礎方程式は圧力項をそれ以外の項と分離して計算する Fractional Step 法を用いて解く.

高レイノルズ数の場合にも計算結果が得られるように, 非線形項は 3 次精度の上流差分法を用いて近似する. その他の空間微分は 2 次精度中心差分, 時間微分は 1 次精度前進差分を用いる.

## 2.3 問題設定

今回は風車ブレードの枚数を 1 枚として計算を行った. 問題設定は Fig.2 のとおりである.

## 2.4 計算格子

計算に用いる格子はブレード断面の翼型を正確に実現するために境界に適合した格子を生成する. 翼の形状は NACA0012 翼型とする. また, 風車ブレード周りの風向は回転とともに変化するため, 特定の方角に格子の集中がない O 型格子とする. ブレードの近傍は精度良く計算するために格子点が密になるようにする. Fig.3 に今回使用した計算格子を示す.

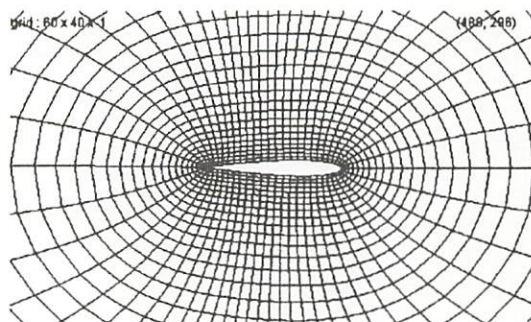


Fig.3 計算格子

## 2.5 境界条件

風車は一定の角速度で回転し, ブレード上ではすべりなしの条件を与える. また遠方境界では速度一定, 圧力は 0 とする.

## 3 計算結果

## 3.1 計算条件

風車の性能を表すために, 風車のブレード先端速度と風速との比で定義される周速比をパラメータとして用いる. 周速比  $\lambda$  は式で示すと以下ようになる.

$$\lambda = \frac{\omega R}{V_\infty}$$

$R$ : 風車の半径,  $V_\infty$ : 風速

今回, 風車半径は翼の弦長の 10 倍とし, 一様流速と弦長を基準としたレイノルズ数は 15000 とする. 周速比  $\lambda$  を 1.0 から 7.1 の間で変化させて計算を行った.

## 3.4 流れ場

Fig.4 に圧力の等値線の例を示す. Fig.5 には速度ベクトルの例を示す. 圧力の等値線と速度ベクトルから, 回転するブレード周りに渦が発生することが確認できる. また, 発生した渦は周速比によって異なる軌跡を見せる. 高い周速比においては, 回転の途中で発生し流された渦にブレードが追いつき衝突する様子を確認できた.

3.3 トルク

トルクは今回の風車の場合、ブレードに生じる揚力による回転力である。Fig.6 にトルク係数  $C_t$  の時間変化を示す。流れ場の様子とトルク係数の時間変化を確認したところ、低い周速比においては羽の先端と後端から交互に渦が放出されるときにトルク係数の変化に乱れが生じることがわかった。さらに、高い周速比においては流された渦の衝突も、発生するトルクに影響を与えることがわかった。Fig.7 には周速比とトルク係数  $C_t$  の関係を示す。

3.4 パワー係数

自然風から風車によって取り出すことのできるパワーの割合をパワー係数という。周速比と出力係数  $C_p$  の関係を Fig.8 に示す。周速比  $\lambda = 6.0$  付近で最大値を得ることがわかる。このことは実験による結果などと定性的に一致する。

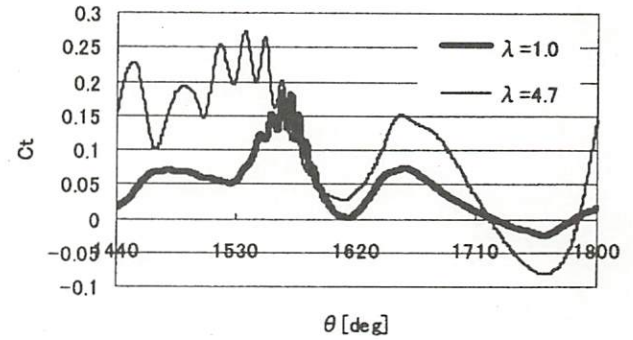


Fig.6 トルクの時間変化

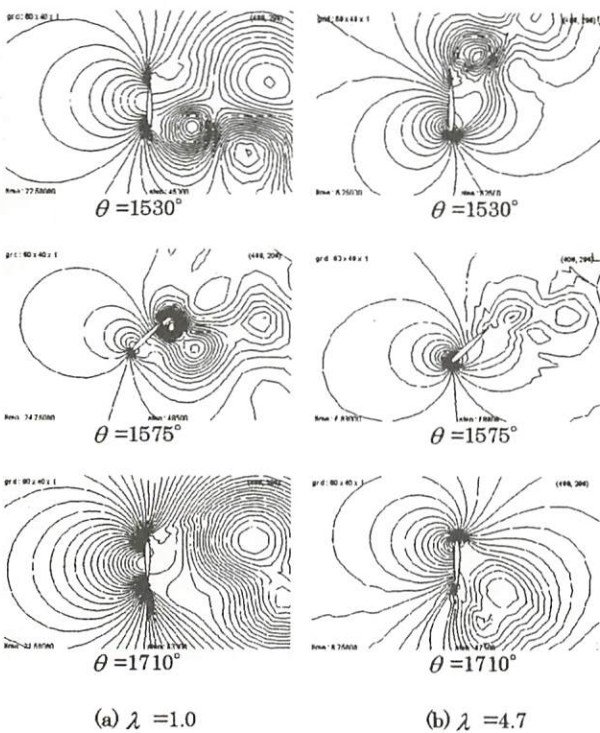


Fig.4 圧力の等値線の例

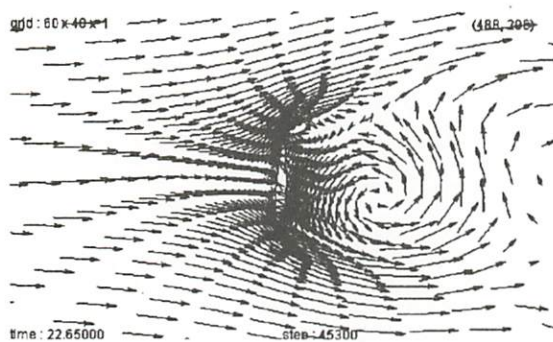


Fig.5 速度ベクトルの例  
( $\lambda = 1.0, \theta = 1530^\circ$ )

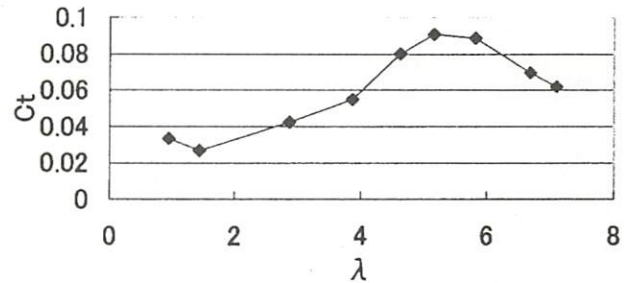


Fig.7 周速比とトルク係数の関係

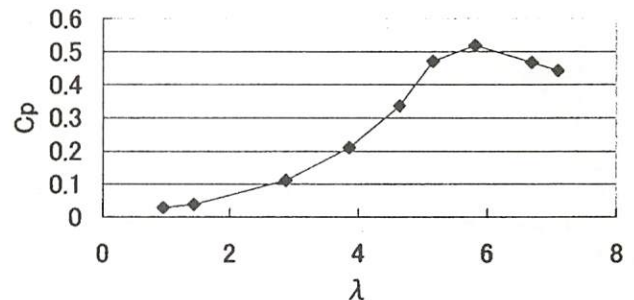


Fig.8 周速比と出力係数の関係

4 まとめ

本研究では 1 枚ブレードの鉛直軸直線翼型風車の流れを解析した。

今後は計算の 3 次元化を行い、実験値との比較を行う。また、ブレードを複数枚取り付け付けた場合の解析を重合格子法を用いて行う。

謝辞

本研究を進めるにあたり、丁寧にご指導いただきました情報科学科の河村先生と伊藤先生、総合情報処理センターの佐藤さんに深く感謝いたします。

参考文献

- (1) 牛山 泉: “風車工学入門”, 森北出版, 2002
- (2) 佐藤, 水上, 伊藤, 河村: “重合格子法による直線翼垂直軸型風車周りの流れの数値シミュレーション”, 第 37 回流体力学講演会, 2005, p207-208
- (3) 前田, 他: “風車回転翼面上の圧力分布に関する研究”, 日本機械学会流体工学部門講演会, 2005, p26

# ビル風のシミュレーション

—生活科学棟をモデルとして—

白谷 栄梨子 (指導教員：河村 哲也)

## 1 はじめに

ビル風問題は都心の高層ビル建設に伴って社会問題化した。近年高層化したマンション建設により、都心だけでなく住宅街などにおいても風による問題が生じている。このように、ビル風は私たちにより身近に感じられつつある問題である。

そこで、本研究では具体的に身近な高層建造物である、新しい生活科学部棟の総合研究棟を例にとり、それを建設することによってどのように風の流れが変化し影響を及ぼしているのかを、数値シミュレーションを行って調べる。

この棟の周りには、理学部校舎群があり特に3号館は7階建ての高層建造物と考えられる。主なビル風の種類をこの数値シミュレーションで実証し、実際目に見える形にして比較検討することを目指した。

## 2 モデル化

本研究では、具体例としてお茶の水女子大学総合研究棟生活環境研究センターを中心とした棟(生活科学部校舎群、理学部校舎群、付属図書館等)に注目し、お茶の水女子大学施設課から提供を受けた資料により、下図のようにモデル化した。

座標系として、3次元不等間隔直交座標を用い、x-zを地表面、yを上空方向にとった。計算領域は地表面を350m×350m、上空方向を60mにした。

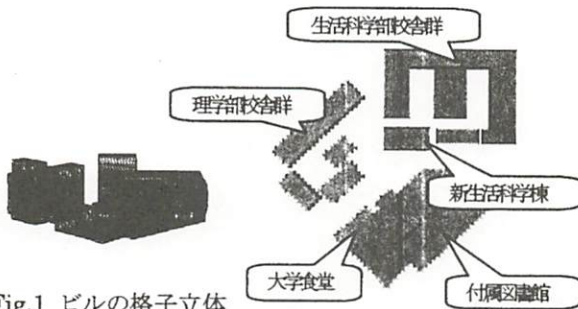


Fig.1 ビルの格子立体

Fig.2 上空から見た図

## 3 格子生成

格子数は、x軸、z軸方向に100、y軸方向に50とした。x-z平面は、生活科学棟の中心から±100mを細かくし、外に向かって粗くなるように、y方向は、地面から40mを細かくし、上空に向かって粗くなるような不等間隔格子を用いた。(Fig.3)



Fig.3 格子

## 4 計算方法

### 4.1 基礎方程式

風速は音速に比べて非常に小さいので、非圧縮性流体の流れとみなせる。したがって、質量保存を表す連続の式(1)と非圧縮性 Navier-Stokes 方程式(2)に支配される。

$$\nabla \cdot V = 0 \quad (1)$$

$$\frac{\partial V}{\partial t} + (V \cdot \nabla) V = -p + \frac{1}{Re} \Delta V \quad (2)$$

V: 速度 p: 圧力 t: 時間 Re: レイノルズ数

### 4.2 ビルの境界

複雑な形状のビルを計算に組み込むために、ビルの形状をあらわす3次元配列 BUILD(x,y,z)を用意し、ビル内部=0、流体部分=1を読み込む。はじめは建物がないと仮定して計算し、得られた結果に配列 BUILD をかけることにより、建物内部の圧力、速度が0になる

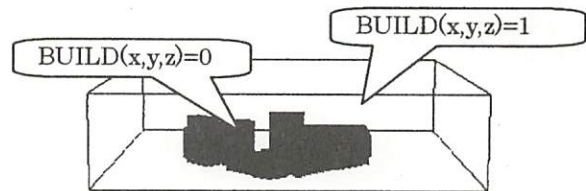


Fig.5 建物の配列

## 5 計算結果

地上1.5m付近の風速場を、新生活科学棟の建設前・後においてそれぞれ Fig.5~14 に示す。また、建設前・後の差がわかりやすくなるように、Fig.9、14 では建設後から建設前の風の大きさの差をベクトルで表示した。

施設課から提供された資料により夏期の主風向(南東風)、冬期の主風向(北西風)について検証を行った。

### 夏期(南東風)の風の動き

\*地表面での様子(上空から見た図)

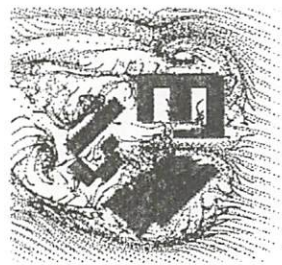


Fig.5 建設前

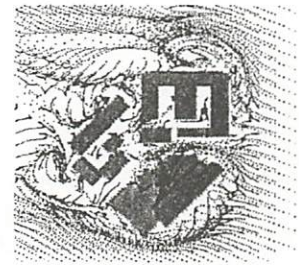


Fig.6 建設後



\*地上付近での様子（南から見た図）



Fig.7 建設前



Fig.8 建設後

\*建設前・後における風の差（上空から見た図）

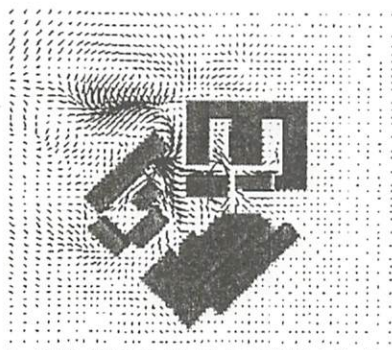


Fig.9 建設前後の風速ベクトルの差（地上1.5m）

Fig.5~9 から、生活科学部棟の東側の壁と図書館の東側の壁で剥離流が起こった。その結果、生活科学部棟の北側で渦をまきながら上空へ吹き上げがみられ、南西から流れる風が理学部棟に遮られることにより、1号館の背後である北側でも風が吹き上げられた。

また、校舎の建設により新生活科学棟の背後である西側において、風が舞い上がる現象がみられ、風の大きさも新生活科学棟と理学部1号館の北側で大きくなった。

冬期の風（北西風）の風の動き

\*地表面での様子（上空から見た図）

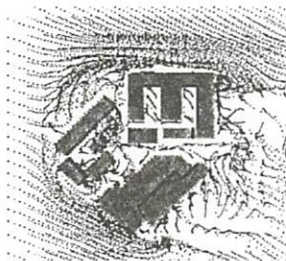


Fig.10 建設前

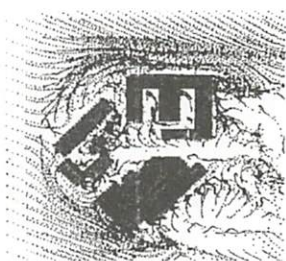


Fig.11 建設後

Fig.10~14 から、理学部1号館の東側で剥離流が起こり、1号館の南側では風が渦をまきながら上昇する現象がみられた。また、図書館の東側でも同様の現象がみられた。夏期とは異なり生活科学部棟の東側で、渦が発生している様子が観察できた。

新生活科学棟が建ったことにより、その南側と図書館の東側において風速が大きくなった。

\*地上付近での様子（南から見た図）



Fig.12 建設前



Fig.13 建設後

\*建設前・後における風の差（上空から見た図）

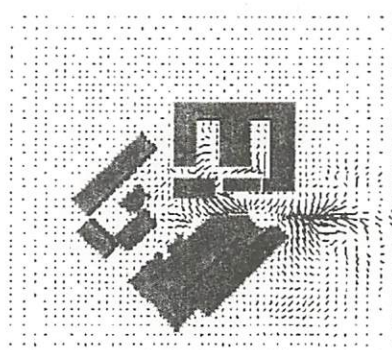


Fig.14 建設前後の風速ベクトルの差（地上1.5m）

## 6 まとめ

校舎に遮られた風は、逆流したり上下左右にそって流れた。

校舎がない場合その部分が風の通り道となり、風が分散しやすいことがわかった。建設により風は集まりやすくなり、校舎の風下側では風が強まったことがわかった。

## 7. 今後の課題

今後の課題として、実測値と比較すること、今回は直交座標系を用いたため、斜線の建物が精度よく計算できなかったのもっと格子の生成を考えて曲面などの複雑な形状の建物に対応していくこと、また実際には地面からの照り返しや建物から発生される熱による温度についても考えていきたい。

## 謝辞

本研究を行うにあたり、ご指導してくださいました指導教員の河村哲也先生、本研究室の諸先輩方に深く感謝いたします。

## 参考文献

- [1] 河村哲也：「流れのシミュレーションの基礎！」インデックス出版
- [2] 安藤常世：「工学基礎 流体の力学」培風館

# 高層ビル群によるヒートアイランド現象

岡島 有希 (指導教員: 河村 哲也)

## 1 はじめに

ヒートアイランド現象とは、都市の中心部の気温が郊外に比べて島状に高くなる現象であり、近年都市に特有の環境問題として注目を集めている。その主な原因として、オフィスや家庭、コンピュータや自動車などで使うエネルギーの増加、それによって都市周辺に生じる煙霧層が作り出す温室効果、都市の建物の凹凸やコンクリートやガラス等の日射による高温化等が挙げられる。これらのことが悪循環になって、都市の気温は年々上がり続けている。事実、地球の平均気温が100年間に0.6℃上昇しているのに比べ、東京の気温は2℃以上上昇しており、これは地球温暖化の5倍以上のスピードで上昇していることになる。

また、都市部では建物や道路の蓄熱、人口排熱などによって郊外よりも温度が高くなるために上昇気流が生じ、地上では郊外から都心へ、上空では逆の循環流が発生する。これを「ヒートアイランド循環」という。さらに、この上昇気流は「ダストドーム」と呼ばれる都市上空で汚染物質をドーム状に覆う現象を発生させる。このように、ヒートアイランド現象は、単なる熱汚染問題であるのみならず、大気汚染問題でもあると考えられる。

そこで本研究では、都市部で発生した熱や汚染物質が大気中にどのように影響を与えるかを数値シミュレーションによって実証することを目的とする。

さらに、ヒートアイランド現象対策として近年注目されている屋上緑化や河川の影響についても考える。

## 2 モデル化

本研究では、具体例として約200mの高層ビル10棟を都市部として、下図のようにモデル化をする。都市から離れるにつれて郊外が広がっているとす。座標系は3次元不等間隔格子を用い、x-z平面を地表面、y方向を上空方向にした。

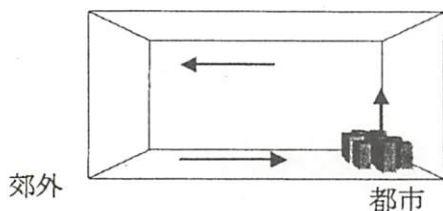


Fig. 1 : 計算領域

## 3 格子生成

格子数は、x、y方向に70、z方向に50とした。

高層ビル群の周りの流れをより詳しく観察するために、高層ビル群部分の格子を細かくして、高層ビル群から離れていくにつれて格子が粗くなるような不等間隔格子を用いた。

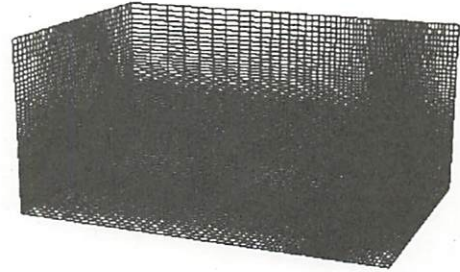


Fig. 2 : 計算格子

## 4 計算方法

### 4.1 基礎方程式

大気中の流れは非圧縮性流体とみなせるので、(1) 連続の式と、(2) 非圧縮性 Navier-Stokes 方程式を支配方程式として解析することができる。また、熱を取り扱うため、(3) 熱に関する方程式も用いた。

$$\nabla \cdot V = 0 \quad (1)$$

$$\frac{\partial V}{\partial t} + (V \cdot \nabla)V = -\nabla p + \frac{1}{\text{Re}} \Delta V \quad (2)$$

$$\frac{\partial T}{\partial t} + (V \cdot \nabla)T = \frac{1}{\text{Re} \cdot \text{Pr}} \Delta T \quad (3)$$

V: 速度、T: 温度、p: 圧力、t: 時間

Re: レイノルズ数、Pr: プラントル数

ここで、プラントル数 Pr は、流れの性質によらない物質の定数であり、0.71 とする。レイノルズ数 Re は、流体の慣性力と粘性力の比を表す量であるが、本研究では乱流の効果も考慮して 2000 とした。これらの式から、圧力と速度を分離して計算を行う MAC 法を用いて計算を行った。

### 4.2 建物の境界

建物の内部は、速度を0とする。ヒートアイランド現象の原因になる熱源は、建物の表面とコンクリート地表部に置き、20℃に設定する。屋上緑化を行った場合、建物の屋上部分のみを10℃になるようにした。

また、都市が分離している場合は高層ビル10棟中2

棟を取り除き、その部分を道路と河川とする。コンクリート舗装道路は20℃、河川は0℃に設定した。

## 5 計算結果

以下の状態で解析を行った。

1. ヒートアイランド現象
2. 屋上緑化を行った場合

また、3.都市部がコンクリートの舗装道路や河川によって2つに分離されている状態についても解析を行う。

風の速度はどの条件でも一定とし、都市から郊外への熱の流れ、都市内部の熱の流れ、また、都市からの汚染物質の流れ、郊外からの汚染物質の流れをFig.3~Fig.12に示す。

## 6 考察

まず、通常状態と屋上緑化を行った状態を比較すると、屋上の温度以外の設定は同じにもかかわらず、熱の流れについては、大きな違いが見られた。汚染物質の流れについては、量や速度に多少の違いはあるものの、都市と郊外で発生した汚染物質はヒートアイランド循環により、郊外へと運ばれていくことが分かった。このことから、ヒートアイランド現象は都市だけの問題にとどまらず、郊外にも影響を及ぼしているといえる。

また、都市が2つに分離されている場合、都市内部の熱の流れは、コンクリートの舗装道路では、ビルの高さ付近まで熱が上昇するが、河川では、河川部分で熱も分離されるため熱の広がりが小さいことが分かった。

## 7 まとめと今後の課題

本研究では、ヒートアイランド現象を考慮した高層ビル群周りの熱の流れと汚染物質の流れについてシミュレーションを行った。

今後の課題として、夏や冬など季節の変化についてシミュレーションを行うことや現実に近い気象条件を与えることが挙げられる。さらに、実在するモデルについて検証を行い、ヒートアイランド現象について考えていきたい。

## 8 謝辞

本研究を行うにあたり、ご尽力くださいました指導教員の河村哲也先生、研究室の諸先輩方に深く感謝いたします。

## 9 参考文献

- [1]河村哲也、渡辺好夫、高橋聡、岡野覚：「流体解析II」、朝倉書店、1997
- [2]尾島俊雄：「ヒートアイランド」、東洋経済新報社、2002
- [3]三上岳彦：「東京異常気象」、洋泉社、2005

## 1. ヒートアイランド現象 (Fig.3~Fig.6)

### 1.1 熱の流れ

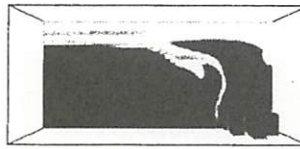


Fig.3 : 都市から郊外へ



Fig.4 : 都市内部

### 1.2 汚染物質の流れ

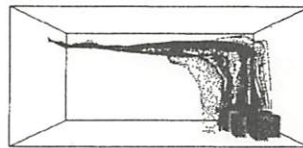


Fig.5 : 都市から

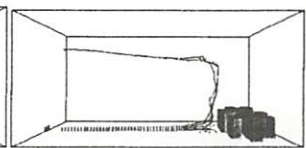


Fig.6 : 郊外から

## 2. 屋上緑化 (Fig.7~Fig.10)

### 2.1 熱の流れ

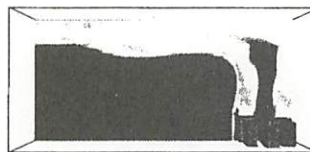


Fig.7 : 都市から郊外へ

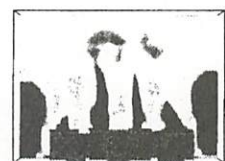


Fig.8 : 都市内部

### 2.2 汚染物質の流れ

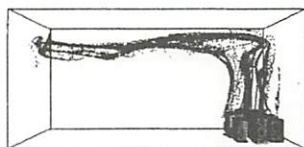


Fig.9 : 都市から

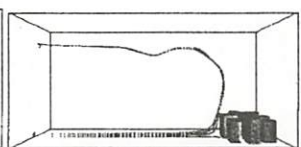


Fig.10 : 郊外から

## 3. 都市が分離している場合 (Fig.11~Fig.12)

### 3.1 熱の流れ



Fig.11 : 道路



Fig.12 : 河川

## 塩味センサによる調味支援

村上 愛淑

お茶の水女子大学 理学部 情報科学科

### 1 はじめに

正確に味を伝えることは出来るのだろうか。

例えば、遠隔地に居る友達同士が電話を使って作り方を伝えながら同じ料理を作る場合、味加減を言葉で伝えることは難しい。また、料理本を見ながら料理を作る場合は、～少々・適量などの表現から、本の著者と同じ味付けの料理を作ることは困難である。このように、空間や時間を隔てて味付け情報を伝えることは困難である。

本研究室では、コンピュータを組み込んだ未来のキッチンシステム、Kitchen of the Future の開発を進めている。このシステムは、市販のキッチンにディスプレイ、フットスイッチ、カメラ、マイクロフォンを組み込んだもので、この上で、遠隔地とTV会議によりコミュニケーションや学習を支援する等のアプリケーションが開発された [2]。このアプリケーションを使って、玉川大学と函館未来大学を結んで調理の方法を伝える実験が行われた。この実験によって、料理の作り方や手順を伝えることは可能であった。しかし、料理の見た目は同じでも、作られた料理が同じ味付けであるかどうかを確認する手段が無かった。従来のテレビ会議システムでは、味付けを伝達することが出来なかったのである。

そこで、味を数値化することで味を伝えられる支援が可能になると考えた。基本的な味の種類は甘味、塩味、うま味、苦味、酸味の5つで、その中でも塩は、最古の調味料であることが分かっている [3][4]。また、人間の体液・羊水のミネラル成分構成は海水と非常に似ており、よって塩分は体に必要不可欠なものだと言われている [1]。上記より、塩味の重要度の高さが分かる。加えて、塩分濃度は電気抵抗の測定により比較的容易に測定することが可能である。

そこで、味の濃さの指標として塩分濃度を測定することにした。これを数値化し、ネットワーク経由で伝達すれば、遠隔地の人とコミュニケーションしながら調理をする際に、同じ味付けの料理を作ることが出来るであろう。また、インタラクティブなマルチメディア料理本システムに本システムを導入すれば、料理の著者と同じ味付けの料理を作れるであろう。今回の卒業研究では、このシステムのために塩分センサ部分を開発したので報告する。

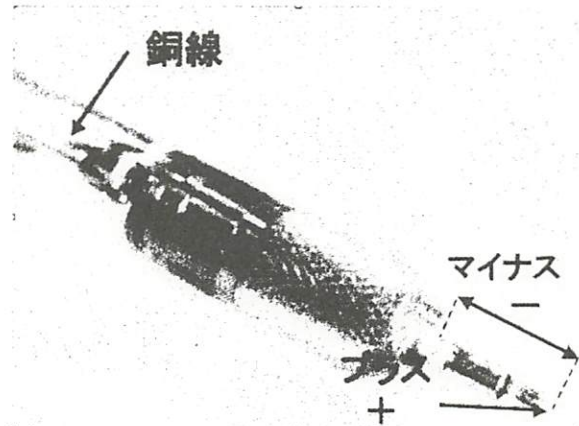


図 1: 電極棒の先端部分

### 2 塩分センサ

料理の塩分濃度を測定するために、その電気抵抗を測定するセンサを開発した。そこで、鍋や食器に入っている液体状の料理の電気抵抗を測定するために、図 1 に示すような棒状のセンサ部分を作成した。これはエンジニアリング・プラスチックの細いパイプの先端に、塗料・電気絶縁材・接着剤として使用されているエポキシ樹脂を用いて電極プラグを接着したものである。どちらも調理中に使用できるよう、100℃以上耐熱性がある材料を選んだ。

電極プラグに  $E$  ボルトの電圧を加えると、そこに接触した食材の電気抵抗  $R$  オームの抵抗により、オームの法則により、以下のように  $I$  アンペアの電流が流れる。

$$I = \frac{E}{R}$$

食材の電気抵抗  $R$  は、食材の電気抵抗率  $\rho$  [ $\Omega \cdot \text{m}$ ]、電極の面積  $S$ 、電極間の距離  $L$  により、以下のように決定される。

$$R = \rho \times \frac{L}{S}$$

食材中のイオンは主に電離した食塩によって移動するので、電気抵抗率  $\rho$  は、塩分濃度によって変化する。そこで、このセンサの電極に流れる電流を測定することで、食材の塩分濃度を知ることができる。

また  $\rho$  は、食材の温度の影響も受ける。そこで、食材の温度を測定するために、温度センサ\* を電極部分に組み込んだ。

\*高精度・摂氏直読温度センサ IC : LM35

電極に流れる電流は、直列に接続した抵抗により電圧に変換し、これをAD変換してコンピュータに取り込む。また、温度センサの出力電圧も同様にAD変換する。この部分を、PIC-BASIC<sup>†</sup>で作成した。PIC-BASICには多チャンネルのAD変換器と、シリアル入出力が内蔵されている。電極に流れる電流値に比例した電圧と、温度に比例した電圧をデジタルに変換するプログラムを作成し、PIC-BASICに組み込んだ。この結果は、RS232C経由でPCに送られる。PCでは、この値から塩分濃度を計算し、これを遠隔地に伝達するプログラムなどを開発する予定である。

現在、電気ノイズの影響で十分に安定した測定結果が得られない問題がある。バッファアンプ回路などを付加して、ノイズを除去する工夫が必要である。

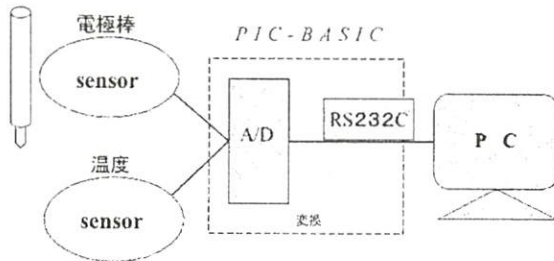


図 2: 塩分センサの全体図

### 3 応用例

本センサによって得られた塩分濃度情報を、ネットワーク経由で伝達することで、同じ料理を遠隔地で作る際に、塩分濃度が分かることにより、おなじ味付けを実現する手助けとなるだろう。また、ネットワーク上やDVDなどで提供される、マルチメディア料理指導コンテンツと連携することで、料理専門家が意図したとおりの味付けを作るシステムの実現も可能であると考えられる。

このようなシステムでは、たとえば、遠隔地の様子や料理指導コンテンツを表示するコンピュータ画面の片隅に、調理中の味付けが指導どおりの値なのかどうかを示すバーグラフなどを表示すると良いかもしれない。このグラフの上下を見ながら、味付けを行うことで、適正な味付けの料理になるように、容易に調味できるだろう。

<sup>†</sup>秋月電子製。BASIC言語でプログラムできるワンチップコンピュータ。

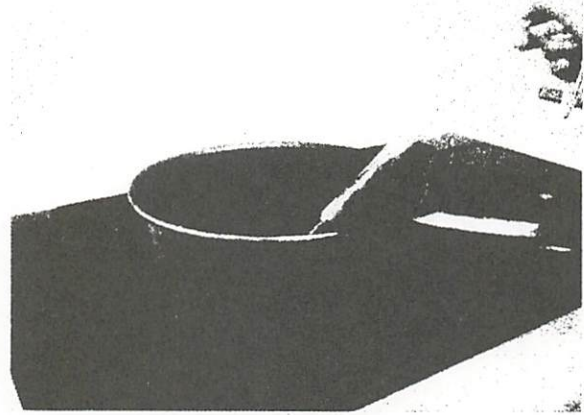


図 3: 電極棒の使用イメージ

### 4 まとめと今後の課題

本研究では、料理の味付けを測定する目的で、塩分濃度を測定するセンサを作成して、これをPIC-BASICによりコンピュータに取り込むプログラムを作成した。今後は、より安定した測定が実現できるように、センサ部分の改良を進めたい。

また、ネットワークごしに塩分情報を伝えることが出来るソフトウェアの開発をすすめたい。さらに、実際の調理場面での利用実験を行うことで、人の味覚により合致した塩分濃度の測定の実現や材料の水分差や作り手の個人差による塩分濃度測定値への影響などの検証をすすめたい。

#### 謝辞

本研究にあたり、玉川大学の宮澤 寛氏の協力を頂いた。ここに記して謝意を表する。

#### 参考文献

- [1] Web,塩の歴史館,<http://www.hakumatsu.co.jp/Hakumatsu-HP/history.htm>.
- [2] 椎尾一郎, 美馬のゆり, Frank, I., 小野哲雄, Weintraub, H.: Kitchen of the Future: レシピ作成を支援するキッチン, 情報処理学会シンポジウムシリーズ, インタラクション論文集, Vol. 2004, No. 5, pp. 237-238 (2004).
- [3] 日本専売公社村上 正祥: 最古の調味料・塩, 週刊朝日百科130. 世界の食べもの・テーマ編, pp. 258-262 (1983).
- [4] 佐々木直亮: 食塩と健康 - 5. 塩の世界史と民族学, Web,<http://hippo.med.hirosaki-u.ac.jp/sasakin/nao-h/salthhealth.htm> (1991).

## どの足スイッチ：キッチン用多機能フットスイッチの開発

杉野 碧

(指導教官：椎尾 一郎)

### 1 はじめに

近年、計算機や周辺機器の価格の低下、小型化が進み家庭内にセンサやディスプレイを組み込んだ生活支援の研究が盛んに行われるようになってきた。本研究室ではキッチンに注目し、キッチンの各4カ所の作業場（シンク、テーブル、まな板、コンロ）にLCDディスプレイを組み込んだ図1のようなコンピュータ強化されたキッチン環境 Kitchen of the Future[1](図1)の開発を行っている。このキッチンを使い、人の調理過程を記録、再生、学習するアプリケーション；遠隔地とリアルタイムで通信し、コミュニケーションを図りつつ調理するアプリケーション；テレビの料理番組の料理手順を構造化し、ユーザーの作業にあわせて1ステップずつ対応する映像と共に提示してくれる調理支援アプリケーション Happy cooking[2]。などの試作をしている。これらのアプリケーションの操作は、図1のように4カ所の作業場の各足下にあるフットスイッチにより行なっている。

これらの研究の課題として、単純なフットスイッチでは調理中に起こりうる様々な状況やアプリケーションに対応しきれていないこと、Happy cooking[2]での複数人数利用に対応出来ない事などが挙げられる。このため、フットスイッチの多機能化が求められている。

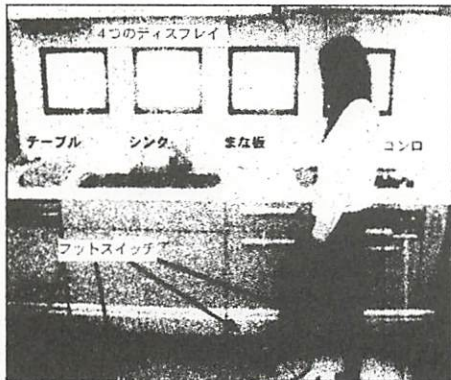


図1: フットスイッチでの Kitchen of the Future 操作

## 2 どの足スイッチ

### 2.1 どの足スイッチの提案と概要

本研究では、前述の課題に対応するために、踏む足を識別（右足なのか、左足なのか、誰の足なのか）できる



図2: RFID タグ:TI 社の 32mm ガラス封入型トランスホンダを使用

フットスイッチを開発した。

足区別の手法として、ユーザーのスリッパに RFID タグを組み込み(図2)、各フットスイッチにスリッパの ID タグを読むためのアンテナを組み合わせた。

どの足がスイッチを押したのか知るためには、スリッパの ID を確実にアプリケーションまで伝えなければならない。ID を読み取るアンテナの感度範囲が広すぎると、スイッチを押していない足の ID を読んでしまい、感度範囲が狭いと ID を認識出来ない。そこで、アンテナの感度範囲を適切に設定にすることが課題になる。また、キッチンという水や火を扱う環境下での耐久性も考慮する必要がある。

### 2.2 アンテナ試作と評価

最適な感度範囲決定のため巻き方を工夫し、様々なパターンアンテナを試作した。また、感度に影響なくアンテナを強化するために、コイル状に巻いたポリウレタン銅線 0.23mm をラミネート加工した。(図4)

RFID<sup>1</sup> のマニュアルを参照し、以下の式より巻き数を導いた (6~7 巻き)。

$$L = \frac{k \times u \times S \times n^2}{l}$$

但し、 $L$  = インダクタンス 47uH、 $k$  = 長岡係数、 $u$  = 真空の透磁率 × 比透磁率、 $S$  = コイルの断面積、 $n$  = コイルの巻き数、 $l$  = コイルの長さ、とする。

<sup>1</sup>Texas Instruments 社 (以下 TI) 製の Micro-reader, RI-STU-MRD1(134.2kHz の低周波数帯を利用する低周波リーダー)

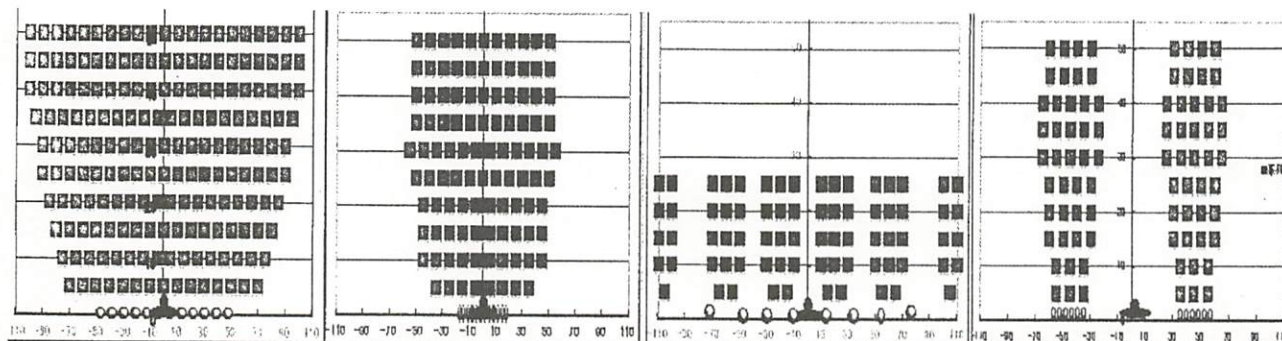


図 3: 左から, モデル [1], モデル [2], モデル [3], モデル [4]. 感度範囲を散布図化したグラフ. 横軸:アンテナの中心からの感度距離. 縦軸:高さの感度距離.



図 4: アンテナ感度評価実験方法

評価実験は図 4 の様に, タグの読み取り範囲を縦横 5mm 単位で計り, アンテナの感度範囲を検出した. (図 3) は結果を散布図で表したものである.

実験結果より以下の事が言える. モデル [1][2][4](図 3) の様に, 銅線の間隔が 6.6cm 以下ならば, 高さ 5cm まで ID タグを認識出来る. モデル [3] に示すように, 間隔 6.6cm 以上になると, 動作が不安定になり, ID の検出領域が不均一になる. モデル [4] より, アンテナを密に巻くと検出可能部分が集中してしまい, ID を読めない可能性がある. これより, 10cm 以下の幅に均一に巻いたアンテナが適していると言える. そこで本キッチンにおいて, 均等に巻いた (幅 3cm, 5cm, 7cm, 10cm) のアンテナを, 実際にフットスイッチと組み合わせて ID 読み取り評価をした.

その結果, アンテナが 5cm 以下の幅では検出域が狭くなり, ID の読み損じが見られた. そして, 10cm の場合が最も安定してタグの ID を読める事が解った. さらに本キッチンでは, フットスイッチを図 2 に示すように, キッチン下部の 10cm の蹴込み部分に設置している. このため, フットスイッチを押すためにユーザーは蹴込み部分につま先を入れることになるので, 検出領域を 10cm にする事で誤検出を防ぐ効果も期待出来る.

### 2.3 ソフトウェアの設計

スイッチを押したユーザーのスリッパ ID は, アンテナが読み取り, リーダーを通じシリアルポート経由で PC に取り込まれる. HappyCocking[2] が C# によって製作されていることより, 「どの足スイッチ」の制御ソフトも C# で製作中である. フットスイッチが押された瞬間にフットスイッチの ID および RFID の ID を含めた KeyEvent コードを上位クラスに送ることにより, どの足がどのフットスイッチを押したかの情報を, キッチンのアプリケーションに知らせる事が可能である.

## 3 まとめと今後の課題

本研究では, 調理支援のための未来型キッチン Kitchen of the Future を操作するための, 多機能フットスイッチの提案と試作をし, より確実な動作のためのアンテナ評価実験を行った. このシステムの利用により, より多くの入力操作や複数人数ユーザーへの対応が可能となるだろう. 今後の課題として, 現在開発中のソフトウェアによる, さまざまなアプリケーションへの対応と実装, また, ユーザーテストを行いキッチン環境での本システムの有用性を図っていきたい.

### 参考文献

- [1] 椎尾一郎, 美馬のゆり, Frank, I., 小野哲雄, Weintraub, H.: Kitchen of the Future: レシピ作成を支援するキッチン, 情報処理学会シンポジウムシリーズ, インタラクシオン論文集, Vol. 2004, No. 5, pp. 237-238 (2004).
- [2] 浜田礼子, 宮澤寛, 鈴木幸敏, 岡部淳, 佐藤真一, 坂井修一, 椎尾一郎: コンピュータ強化キッチンによるインタラクティブ調理支援 (2005).

# トントンセンサ：キッチン向け入力インタフェースの開発

早樋 沙織

(指導教官：椎尾 一郎)

## 1 はじめに

近年の、コンピュータやインターネットの普及に伴い、生活環境のあらゆる場所に情報通信環境が埋め込まれ、あらゆる場面でコンピュータを利用できることを実現する社会、つまりユビキタス社会が現実的になりつつある。この流れから、家庭内にも、生活支援のためにコンピュータやディスプレイ等を組み込む研究、開発が進んでおり、ユビキタスコンピューティングが浸透しつつある。

そこで本研究は、家庭内でも、人間にとって必要不可欠な食生活が営まれる場所であるキッチンに注目し、そこにあらゆるコンピュータテクノロジーを導入し、単に調理の効率化を図るだけでなく、コミュニケーションや学び場として利用しようと考えた。本研究室が開発している、キッチンにディスプレイやスイッチなどを組み込んだ未来型キッチン Kitchen of the Future では、学習やコミュニケーションの支援が実現可能である。[1] 調理過程の記録、公開、再生するアプリケーション [2]、レシピと映像を1ステップずつ対応させ、画面に提示することで、ユーザが映像と同時進行で短時間かつ最適に作業できるように支援するアプリケーション Happy Cooking[3] がある。

また、本研究の Kitchen of the Future システムでは、4箇所の作業スペース(ガスコンロ、テーブル、シンク、まな板)それぞれにディスプレイとテーブル型フットスイッチを設置した。(図1) ユーザが各エリアに移動する度に、常に目の前のディスプレイに表示される情報で、最適な調理支援を実現した。また、ユーザがその場にあるフットスイッチを踏むことで、アプリケーション操作が可能となった。これは、キッチンでは、調理中に手が汚れたり、両手が塞がっている状態が多いので、足元にスイッチがあると最適であると考えられる。しかし、HappyCookingなどのマルチメディアコンテンツを操作するには、多くの入力装置が必要になる。

## 2 トントンセンサ

### 2.1 概要

本研究では、キッチンの過酷な環境で利用する入力デバイスの開発を行った。キッチンに設置する入力デバイスは、調理の妨げになってはいけない。また、キッチンで

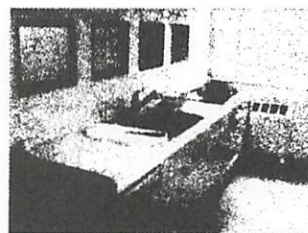


図1: Kitchen of the Future. 壁にLCD, 足元にフットスイッチが組み込まれている。

は、食料を洗ったり、切ったり等さまざまな作業が行われるため、手が汚れていたり、濡れた状態でも確実に入力操作を行う必要がある。

そこで本研究は、キッチンの中で、躊躇なく手で叩いて、入力操作ができる場所として、シンクに注目した。キッチン内におけるさまざまな作業中に、シンクの内側に触れる機会は少なく、そのためここをコンピュータ入力デバイスとして利用すれば、調理の邪魔になりにくいと考えられる。また、シンクの内側は、汚れた手で操作してもすぐに洗い流すことができる。そこで本研究では、シンクを利用してアプリケーション操作を行うインタフェースを提案する。

シンクに対する操作として、人がシンクを手などでトントンと叩く動作を利用することにした。このセンサをトントンセンサと名付けた。トントンセンサとは、シンクを叩く振動でコンピュータに入力する、装置である。2個の振動センサをシンク裏に設置して、人が叩いた振動を検出する。振動がセンサに到達する時間差を測定すれば、振動センサの間のどの部分を叩いたかを知ることができ、叩いた場所に機能を割り当てれば、キッチンでのコンピュータ操作、例えば、料理番組のDVDの再生、停止、巻き戻しなどの操作を実現できるであろう。ここでは、位置センサとして、二個のスピーカー (TRIPLE RE1601:4  $\Omega$ , 0.25W) を使用した。(図2)

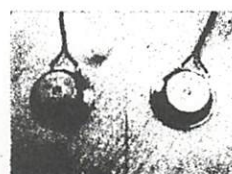


図2: 使用したスピーカー



図3に全体のブロック図を示す。シンクを叩くと、センサ(スピーカー)に振動に応じた電圧が生じる。これらをマイクアンプ<sup>1</sup>により増幅する。次に、コンパレータ<sup>2</sup>を用いて閾値以上の電圧の入力があつた場合に TTL レベルに変換する。このようにして得られた2つのセンサからの信号のうち、早く来た信号を Flip-Flop 回路にて保持した。保持された信号は、PIC-BASIC<sup>3</sup>の平行ポートに入力した。また、PIC-BASIC から Flip-Flop 回路に信号を送ることで、保持した情報をクリアできるようにした。

振動センサは、人がトントン叩く動作以外に、調理中の様々な振動を検出してしまふ。そこで、予期しない誤動作を排除するために、図4に示すキッチンの足元に設置したフットスイッチと連動させることにした。すなわち、フットスイッチを踏んだときのみ、シンクを叩くと入力できるようにした。フットスイッチの状態も、PIC-BASIC の平行ポートに入力した。

PIC-BASIC には、フットスイッチが踏まれると、Flip-Flop 回路をクリアし、振動の測定を開始し、どちらかの振動センサの入力があつたら、その結果を RS232C 経由で PC に伝達するプログラムを書き込んだ。現在のプログラムでは、2つのセンサのどちらに早く振動が伝わるかだけを検出している。このため、2種類の入力のみを受け付けることができる。将来は、時間差を測定できるようにして、2つのセンサの間に複数の識別領域を設けようようにしたいと考えている。

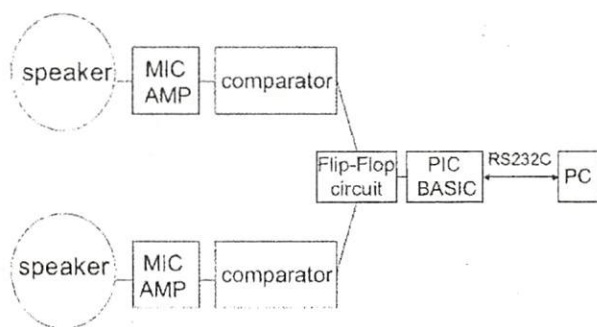


図 3: ブロック図

振動センサはシンクの裏側に設置すべきであるが、感度が得られなかったため、現在はキッチンカウンターに設置している。シンクのステンレス素材に防振加工がされており、振動が吸収されてしまったためである。今後は、振動センサの改良を行い、シンクの振動を検出できるよう工夫したい。

<sup>1</sup>ELELIT NT-5

<sup>2</sup>National semiconductor 社製 LM311

<sup>3</sup>BASIC でプログラミングできるワンチップコンピュータ

RS232C で接続された PC では、Visual Basic.net により書かれたプログラムが動作している。また、DVD 再生ソフトウェアが稼働しており、ここで DVD を再生した。Visual Basic プログラムは、PIC-BASIC からの検出結果情報に従い、DVD 再生ソフトウェアに、再生、停止などのショートカットキーイベントを送る。これによりシンクを叩くことで DVD 再生をコントロールできる。

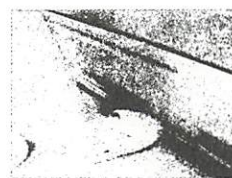


図 4: フットスイッチを踏んでいる状態

### 3 まとめと今後の課題

本研究では、未来型キッチン Kitchen of the Future システムのために、シンクの内側を叩く動作により、コンピュータ操作を行う入力デバイスを開発した。

今後、時間差を測定できるようにワンチップコンピュータ部分を強化し、多機能な入力を可能にしたい。また、実際に調理を行うユーザによる評価実験を行い、動作の安定性、設置場所の妥当性などの検証をし、改良を行いたいと考えている。さらに、Kitchen of the Future の他のプロジェクトで開発されている、Happy cooking 等の調理支援アプリケーションと連携して動作できるよう、アプリケーションとインターフェースを仲立ちするミドルウェアを開発していきたいと考えている。

#### 謝辞

Kitchen of the Future 向けの入力インタフェースの開発にあたって、玉川大学の宮澤 寛氏の協力を得た。

#### 参考文献

- [1] Siio, I., Mima, N., Frank, I., Ono, T. and Weintraub, H.: Making Recipes in the Kitchen of the Future, in *Extended abstracts of the 2004 conference on Human factors and computing systems*, pp. 1554-1554, ACM Press (2004).
- [2] 椎尾一郎, 宮澤寛, 美馬のゆり: Kitchen of the Future: 調理を記録・公開・再生するキッチン, 情報処理学会シンポジウムシリーズ, インタラクション論文集, Vol. 2004, No. 34, pp. 5-8 (2004).
- [3] 浜田礼子, 宮澤寛, 鈴木幸敏, 岡部淳, 佐藤真一, 坂井修一, 椎尾一郎: コンピュータ強化キッチンによるインタラクティブ調理支援, 情報処理学会シンポジウムシリーズ, インタラクション論文集, Vol. 2005, No. 5, pp. 49-52 (2005).

## 置いて動かす文字入力

松崎 由香里†

(指導教官：椎尾 一郎†)

### 1 はじめに

近年、RFID タグなどを物に埋めこんだ実世界指向システムの研究・運用がさかんに行われており、既に Suica や Edy など一般に広く浸透しているものも多い。筆者らは、ID リーダと動き検出装置などを一体化して操作の幅を広げようとするシステム、MouseField を提案している [2]。これは、光学マウスと RFID リーダを組み合わせたユビキタスコンピューティングのための入力デバイスである。

MouseField は、裏返しに設置された光学マウスの周囲に、RFID のアンテナを配している。これにより、RFID タグの組み込まれた IC カードなどを置くことでものの識別が行われて、そのカードを動かすことで様々な情報の操作を行うことが可能である。このデバイスを使用したものには、行き先掲示板 (図 1) や PlayStand++ などがある [2]。このデバイスは、安価で丈夫であり、防水処理なども容易にできるので、キッチンや風呂場、屋外などでの使用も可能である。よって、過酷な環境で使用される街頭キオスク端末などにも適しているといえる。

たとえば、公共インターネット端末に應用すれば、プリペイド機能をもった IC カードをかざすことで課金され、インターネットにアクセスする。そして同時にそのカードを動かすことで、一般のマウスと同様の操作によりコンテンツを操作することができる。また、銀行のキャッシュディスプレイなどへの應用も考えられる。いずれの場合も、カードを置いて動かすだけで操作できるので、ボタンやタッチパネルなどのデバイスが不要であり、安価で丈夫な端末が実現できる。これを生かすことで、従来では設置や管理にコストがかかりすぎるような場所への展開が期待できる。また、認証のためのカードを手を持ったまま操作を完了することができるので、一度ユーザの手を離れるキャッシュディスプレイ等と比較すると、高い安全性を確保することができる。

従来の MouseField では、メニューやチェックボックスなどの単純な操作のみを利用していた。しかし、街頭キオスク端末への應用を考えると、暗証番号や金額などの文字入力機能が必要になる。

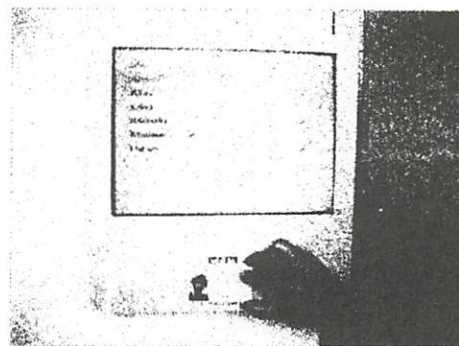


図 1: MouseField デバイスを用いた行き先掲示板

### 2 文字入力手法

MouseField では、カード等を置いて動かすことにより、通常のマウスによる指示操作を提供できる。また、ボタン等を持っていないが、カードの検出・非検出をマウスボタンの押下として使うことも可能である。ただし、カードを持ち上げる動作は、マウスボタンの操作より面倒であり、ユーザに負担を強いることになる。

ペンデバイスにより文字を入力する PDA 等では、ソフトウェアキーボードをペンでタップする手間を省く目的で、パイメニューとペンの一筆書きを利用した方式が各種提案されている [4] [1] [3]。そこで、MouseField を用いた文字入力においても、このような手法が有効であると考えた。本研究では、MouseField デバイスの上に IC カードを置き、画面上に表示した文字ボタンの上のポインタを IC カードにより動かし、パイメニューを一筆書きで選択する手法を提案する。

図 2 に本研究が提案する入力方法を示す。これは、銀行のキャッシュディスプレイなどで金額を入力する場面を想定している。ここでは、操作画面上に文字ボタン、数字ボタン等および、IC カードの移動に追従するポインタの移動を表示する。そして、このポインタが通過するボタンの内容がテキストフィールドに入力されるようにする。すなわち、ユーザはカードを動かし、入力したい文字/数字ボタンの上へポインタを移動させることで文字/数字を入力する。

文字/数字ボタンは、ポインタの移動が細小となるように、ペン入力などで利用されているパイメニュー



図 2: 上: 一筆書き入力. 下: 常に中央に戻る入力方法

同様、円周上に配置する。また、横長のカードで操作することを考えると、正円型より楕円型に配置した方がよいと考えた。数字の並びについては、場所が覚えやすいよう、時計と同様の角度に配置するようにした。また、訂正ボタンや終了ボタンを用意することで、入力の修正や終了を実現する。

一方、ボタン領域に入ったところで、画面中央などのホームポジションにポインタを移動させるようにしたら操作しやすい可能性もある。そこで、図 2 下のよう自動的に中央に戻る方式も考えられる。

さらに、図 3 のように選択肢が少ない場合は、カードの横方向移動だけを検出して、左右の移動だけを可能にすれば、操作はより簡単になる。そして、カードをデバイスから離すことで RFID リーダがカードの ID を読み取れなくなることを利用すれば、カードを持ち上げることで入力終了（入力決定）とできるであろう。

MouseField デバイスでは、移動中にカードがリーダーの読み取り領域外に一時的に動く可能性がある。そこで、適当なタイムアウト時間を設定して、操作の連続性を確保する必要がある。一方で、このタイムアウト時間を短く設定しておけば、認証後の不正アクセスを防ぐことができる。さらにセキュリティを強化する方法として、入力時の通過経路を記憶しておき、次回操

金額は1580円でよろしいですか



ボタンを選んでカードを持ち上げてください

図 3: 選択肢が少ない例。

作時に異なる経路パターンをたどった場合はエラーとなるようにすれば、暗証番号が漏れたとしても、不正アクセスを防ぐことができるであろう。

### 3 まとめと今後の課題

本研究では、MouseField デバイスの上に IC カードを置いて動かすことで、画面上に表示した文字ボタン上のポインタを動かし、一筆書きで選択する文字入力方法を提案した。今後は、本論文で提案した手法を実装し、さらに使いやすい形状等を調査する予定である。また、ユーザによる評価実験を行い、本方式の有効性を実証していきたい。

#### 謝辞

FieldMouse を用いた文字入力手法の開発にあたって、玉川大学の宮澤 寛氏の協力を得た。

#### 参考文献

- [1] Mankoff, J. and Abowd, G. D.: Cirrin: a word-level unistroke keyboard for pen input, in *UIST '98: Proceedings of the 11th annual ACM symposium on User interface software and technology*, pp. 213-214, New York, NY, USA (1998), ACM Press.
- [2] Masui, T., Tsukada, K. and Sio, I.: MouseField: A Simple and Versatile Input Device for Ubiquitous Computing, in *UbiComp 2004: Ubiquitous Computing*, pp. 319-328, Springer (2004).
- [3] Perlin, K.: Quikwriting: continuous stylus-based text entry, in *UIST '98: Proceedings of the 11th annual ACM symposium on User interface software and technology*, pp. 215-216, New York, NY, USA (1998), ACM Press.
- [4] Zhai, S. and Kristensson, P.-O.: Shorthand writing on stylus keyboard, in *CHI '03: Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 97-104, New York, NY, USA (2003), ACM Press.

## 文鎮メタファを利用した小型情報機器向けインタフェース

辻田 眸 (指導教官：椎尾 一郎)

### 1 はじめに

PDAや携帯電話などの小型情報機器では、表示画面が小さいために、サイズの大きなコンテンツを一度に表示することが困難な場合が多い。一般には、コンテンツの一部のみを表示して、スクロールバー操作や、コンテンツ部分のドラッグ操作等により、残りの部分をスクロール表示する手法がとられる。そのため、たとえば大型画面用にデザインされたWWWコンテンツを閲覧しながら、ボタンやリンクをクリックしたり、検索などの文字入力を行ったりする場合には、コンテンツのスクロールと、コンテンツの操作・編集を交互に頻繁に繰り返す必要がある。コンテンツ操作とスクロール操作の切り替えには、スクロールバーの場合は、スクロールバーまで指示装置を移動する必要がある。コンテンツ部分のドラッグでスクロールする場合は、スクロールモードへの切り替えが必要であり、この操作が頻繁に必要なことが、小画面表示デバイス进行操作する際の問題点となっていた。

### 2 文鎮メタファ

本研究では、表示コンテンツの編集操作とスクロール操作を直感的に切り替える手段として、文鎮メタファに基づくインタフェース手法を提案する。図1のように、平滑な机の上に紙片を置き、片手で紙片に文字を書き込もうとする場合、筆記具の先だけを紙の上において動かすと、文字を書くことができず紙が滑ってしまうことがある(図1左)。このような状況では、人は、手のひらを使って、紙を押さええて固定して文字を書こうとする(図1右)。このように手を使って文鎮のように紙を押さえる動作をメタファとして利用すれば、スクロールと編集操作をスムーズに切り替えるインタフェースが実現可能である。

そこで、本研究では、ペン入力可能なPDA等の手のひらが当たる部分にタッチセンサを取り付けたデバイスを提案する。これにより、手のひらがタッチセンサに触れている間は(紙が押さえられるメタファを利用して)文字入力等のコンテンツ操作が可能であり、手のひらが離れている時には(紙が自由になり筆記具で移動するメタファを利用して)ドラッグするとコンテンツがスクロールするインタフェースを実現できる。

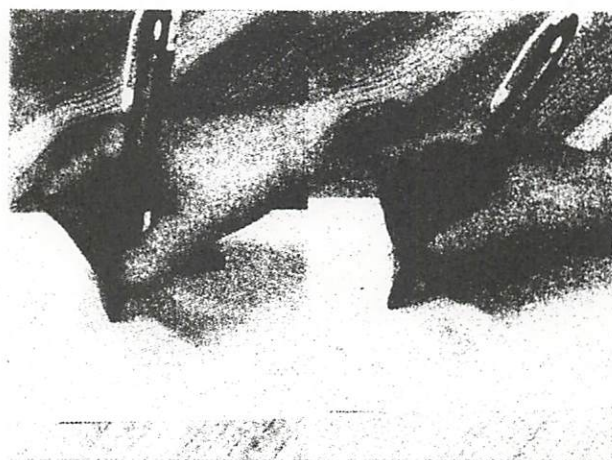


図1: 文鎮メタファの説明。机上の紙片は筆記具を動かすと移動してしまう(左)ので、書き込むには手で押さえなければならない(右)。このような手で押さえる動作を利用してスクロールと編集モードを切り替える。

### 3 応用例

文鎮メタファは様々な応用が可能である。本研究では図2のように、PDA<sup>1</sup>の手のひらが当たる部分にタッチセンサをとりつけ、様々な応用の実装と評価をすすめている。図2では、地図などのコンテンツを表示して、タッチセンサに手のひらが触れていない場合には地図がスクロールし、触れている場合には地図に手書きメモが書き込めるアプリケーションを示している。

地図の他に、たとば写真、ワープロ書類、設計図、電子カルテなどのスクロールと編集の切り替えにも応用できる。PCの大きな画面向けに設計されたWWWページのブラウザも実装できるであろう。WWWブラウザでは、手のひらを当てない状態でスクロールし、手のひらを当てることで、WWWページのリンクやボタンをペンによりクリックしたり、WWWページの検索欄などに文字を入力するなどの操作を提供する。

小型コンピュータは公共の場に携帯されて利用されることも多い。そのため、住所や電話番号など個人情報を記入したり、パスワードを入力するなど、他の人に画面を見られたくない場合もある。そこでプライベートな情報を入力(手がタッチ)している時に、バックライトの照度を下げる等の仕組みを機能させ、のぞき防止用の画面

<sup>1</sup>シャープ社 Zaurus SL-C760

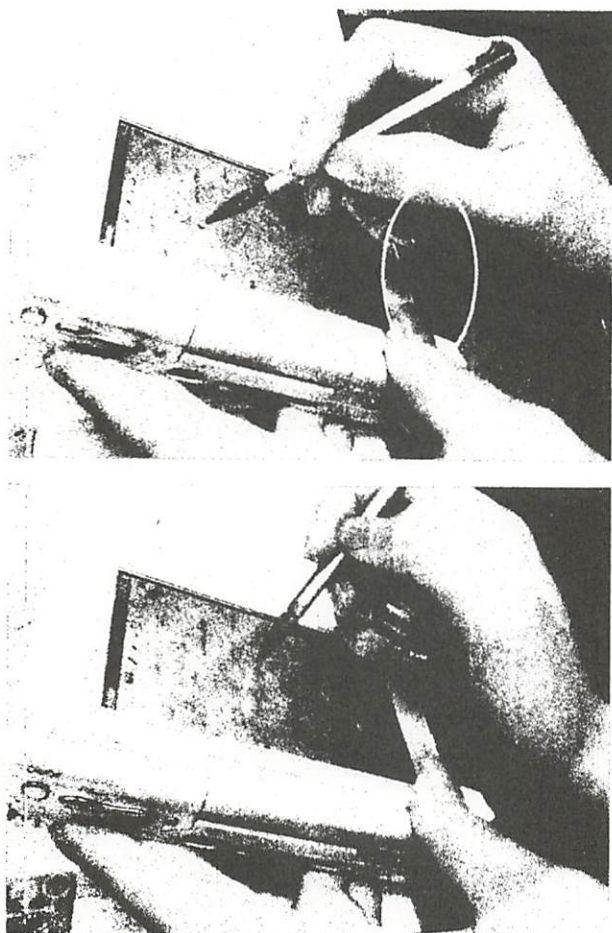


図 2: 上はセンサ (円内) に触れずにドラッグすることでスクロールしているところ, 下はセンサに触れることで手書きメモ入力しているところ。

とすることも可能である。

また写真ブラウザにおいては、手のひらで押下されたことで写真の一部が固定されたと考えれば、その状態でのペンによる左右方向のドラッグ操作は、固定された点を中心とした回転操作になると考えられる。また、固定された点とペンで上下にドラッグする点の距離が変化することから、写真が伸縮するゴム板のような性質を持っているととらえて、写真の拡大・縮小操作になるとも考えられる。そこで、タッチした場合には写真の回転や拡大・縮小モードとし、タッチしない場合にはスクロールモードとすることも自然で直感的である。

ディスプレイ一体型タブレットを間に置いて対向する二名のユーザが、両方向から同時に操作する利用環境において、接触センサをそれぞれのユーザの手のひらの当たる場所に設置すれば、どちらのユーザが画面の操作をしているのかを検出することができる。これにより、どちらのユーザの書き込みであるかを判定して記録するアプリケーション等が実現できる。

ノート PC などでは、指先の接触位置を検出するタッ

チパッドと呼ばれる指示装置が用いられている。タッチパッドの手前に接触センサを取り付ければ、通常はマウスのような相対座標入力装置として機能し、手のひらが当たっている場合には、ペンタブレットのような絶対座標入力装置として機能させることができる。これにより、例えば、手のひらを触れた状態で、タッチパッドに文字を書くことで、文字認識を行い文字入力を行うなどの機能を実装することが可能になる。

## 4 関連研究

ノート PC のタッチパッドにユーザの指が接触することを利用して、キーボードの機能を切り替える手法が提案されている [2]。本研究は、小型スクリーンを備えた情報機器を対象としている。また、携帯コンピュータの画面の周囲に指先接触位置センサを配置して、この操作によりページめくりなどを実現する手法が提案されている [1]。本研究では、単純なタッチセンサの追加により、文鎮メタファを提案したスムーズなモード切り替えの実現を目的としている。

## 5 まとめ

文鎮メタファを適用して、手のひらがタッチセンサに触れている間は文字入力等のコンテンツ操作が可能であり、手のひらが離れている時にはドラッグによりスクロールするインタフェースを考案した。今後は様々な応用の実装と評価実験を行い、本システムの有効性の検証を行っていきたい。

なお、本研究の内容は文献 [3][4] として公開する予定。またお茶の水女子大学で特許出願中である。

## 参考文献

- [1] Hinckley, K., Pierce, J., Sinclair, M. and Horvitz, E.: Sensing techniques for mobile interaction, in *UIST '00: Proceedings of the 13th annual ACM symposium on User interface software and technology*, pp. 91-100, New York, NY, USA (2000), ACM Press.
- [2] Rekimoto, J.: ThumbSense: automatic input mode sensing for touchpad-based interactions, in *CHI '03: CHI '03 extended abstracts on Human factors in computing systems*, pp. 852-853, New York, NY, USA (2003), ACM Press.
- [3] 辻田眸, 椎尾一郎: 文鎮メタファを利用した小型情報機器向けインタフェース, *インタラクション2006* (2006年3月).
- [4] 辻田眸, 椎尾一郎: 文鎮メタファを利用した小型情報機器向けインタフェース, 第68回情報処理学会全国大会 (2006年3月).

# 駅すきゅうずみ：無線LANを利用した降車駅リマインダ

岡野 典子 (指導教員：椎尾 一郎)

## 1 はじめに

近年、位置情報に関するサービスに対する需要は高まってきている。位置推定システムとして広く知られているGPSは、携帯電話やカーナビゲーションに搭載されることによって普及してきた。しかし、地球上を周回する衛星をとらえる事ができないと位置推定が行えないので、電車内や建物の陰では使用できないといった欠点がある。そこで、小型な機械で簡単に使用する事ができ、建物が多し都市や屋内で利用可能な位置推定システムの開発が望まれている。学校や企業、空港、電車、ファーストフード店をはじめとする飲食店、さらにはスーパーマーケットに至るまで、無線LAN(WiFi: Wireless Fidelity)は生活のあらゆるところで利用可能になっている。そこで、本研究では、電車内や市街地で使用できるWiFiを利用した位置推定アプリケーションの提案を行う。

## 2 関連研究

市中に存在する既存のWiFiを利用した位置推定システムは、多くの研究チームによってなされている。Intel Researchが推進するPlaceLabは多くのユーザを集めて、様々な場所でのWiFi, GSM, Bluetooth情報のフィンガープリント(そこで受信される基地局IDと強度のリスト)を収集している[1][2]。日本では名古屋大学大学院情報科学研究科河口研究室において、Locky.jpという多数のユーザの収集によって行う位置推定システムの開発が進められている[3]。これらの研究ではフィンガープリントと場所を対応付けたデータベースを作成する。位置推定する際には、実際に観測されたWiFi情報と、データベースにある情報を比較する。Intel ResearchがあるシアトルやLockyプロジェクトの中心となっている名古屋は車社会である。人々は地域を自由に移動するため、任意の場所の二次元的な位置情報が重要である。一方、東京ではJRや多くの私鉄が整備されて、電車での移動の方が車での移動よりも一般的である。そこで、一次元的に移動する本研究では、電車に特化した、WiFiによる位置推定システムの実現を行う。

## 3 降車駅リマインダ

### 3.1 目的

本研究では、WiFi情報(MACアドレス, SSID, 強度, 及びそれを検出したタイムスタンプ)を取得できる携帯機器によって、電車内においてユーザが事前に登録しておいた降車駅に到着したときに音を鳴らして知らせるアプリケーションを提案する。

本機器は常時、観測できるWiFi情報を取得し続け、容量が規定量を超えたら、古いWiFi情報から削除する。ユーザは降車駅を登録する場合、知らせて欲しい目的地にたどり着いたときに、登録ボタンを押す。登録ボタンを押す事により、メモリに残っている全てのWiFi情報を記録する。

降車駅が登録された後も電源が入っている限り、その時点でのWiFi情報を取得し続ける。観測したWiFi情報が、登録されたWiFi情報と一致したら、降車駅に着いたと判断して、音を鳴らしてユーザに知らせる。

また、乗り換えを要する移動の場合を考慮し、知らせるべき降車駅を複数登録する事も可能にする。複数のWiFi情報を登録した際には、登録してある一つの情報に一致した際に、それぞれ音を鳴らす。

### 3.2 手法

関連研究で挙げたPlaceLabやLockyプロジェクトといった、既存の無線LANによる位置推定システムでは、ある一点におけるフィンガープリントとの比較を行う。しかし、WiFiの普及が進んでいるとはいっても、近隣にあまり住宅やオフィスが無く、WiFiが整備されていない場合がある。また、大都市圏のターミナル駅付近では駅の規模が大きく、多くの線路が敷かれているため、近くにWiFi基地局が存在していても観測できない、という場合がある。本研究で提案する手法では、目的地に到着するまでの移動中のWiFi情報を利用することにより、全くWiFiが観測できない位置においても、WiFiを利用した位置推定を可能にする。

図1のようにWiFi基地局が配置されている場合を考える。このような場合において、WiFi情報は、図2のように、取得される。

降車駅に着いたことを判定する際には、以下の式を満

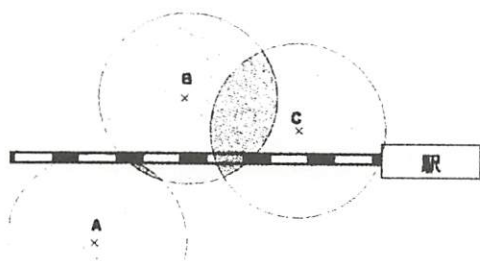


図 1: 目的地までの線路上で観測できるすべての WiFi 情報を利用する。

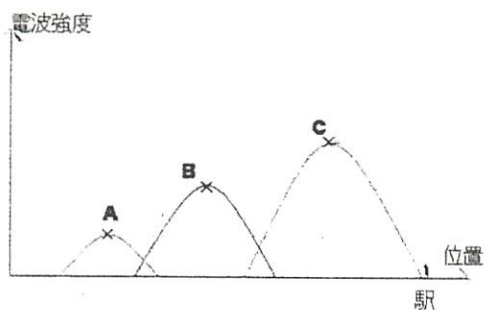


図 2: 図 1 のような基地局配置から, このような WiFi 情報が得られる。

たせば良い。

$$\sum_{i=1}^M (C_i - R_i)^2 < D$$

C は本システムの利用の際に降車駅までかかった時間, R は登録の時に, 降車駅までかかった時間, i は WiFi の種類を示す. D は 0 に近い閾値とする. 一つの WiFi 情報に対して, 電波強度のピーク時を用いて, C と R は定める。

### 3.3 WiFi パターンマッチングのための課題

WiFi の設置状況は日々変化し, 電車は高速に移動するので, 必ずしも全く同じ無線 LAN 情報の標本を取得できるとは限らない. 完全には一致しない電波情報から, 同じ位置であるということを, 評価する基準を設けねばならない. この評価の際に, 考慮せねばならない点を挙げる. まず WiFi 情報を観察する順番は, 登録されていた WiFi が全て観測されていれば, ほとんど変化しない. しかし, WiFi 基地局の設置状況は, 日々変化するので, 新しい基地局が増設されている場合も考えられる. また, WiFi 基地局には, 線路から近いものも遠いものも存在する. 線路から近い WiFi 情報は長時間観測しやすく, 遠いものは電波が届きにくいいため短時間しか観測できなかったり, 時によっては観測できない場合がある. 本研

究では, 観測できなかった部分を見捨て, WiFi 情報が一致する地点のみに関して, 式への代入を行う. 将来的には, 基地局環境の変化に対応できるように, 再学習機能を加えたい。

WiFi 情報が観測されてから降車駅までの時間を利用する事により, 降車駅付近に全く WiFi 情報が取得できない環境であっても, 降車駅を知らせることが可能になる. しかし, そのときの電車の速度や信号などの状況によって, 観測されてから降車駅までの時間は, 必ずしも常に一定にはならない. このような際には, 標本の各 WiFi 情報が観測された時から目的地までの所要時間の比率から, 利用者の観測状態を比較し, 正規化を行わねばならない。

## 4 まとめと今後の課題

電車内に特化し, 観測できる無線 LAN 情報の流れや観測時間, 電波強度を見ることによって, 従来よりも簡単に位置推定システムの実現する手法を提案した. 今後はパターンマッチングのための課題を考慮に入れて本システムの実装を行い, 評価実験を通して, 有効性を実証したい。

謝辞

無線 LAN 情報の収集アプリケーションに関して, 名古屋大学大学院情報科学研究科の河口信夫助教授に協力を得た。

参考文献

- [1] LaMarca, A., Chawathe, Y., Consolvo, S., Hightower, J., Smith, I., Scott, J., Sohn, T., Howard, J., Hughes, J., Potter, F., Tabert, J., Powledge, P., Borriello, G. and Schilit, B.: Place Lab: Device Positioning Using Radio Beacons in the Wild., in *Third International Conference Pervasive 2005, Lecture Notes in Computer Science(LNCS3468)* (2005).
- [2] LaMarca, A., Hightower, J., Smith, I. and Consolvo, S.: Self-Mapping in 802.11 Location Systems., in *UbiComp 2005* (2005).
- [3] 伊藤誠悟, 吉田廣志, 河口信夫: locky.jp:無線を用いた位置情報・測位ポータル, in *Vol.2005 No.90 2005-MBL-034* (2005), <http://locky.jp/>.

## 生物対流の数値的研究

小西 真裕美 (指導教員: 河村 哲也)

### 1. はじめに

地球上では重力があるおかげで上下の区別がついている。重力は束縛力であり地上にとどめようとする力になる。一方水中では生活の場合は立体に発展し、多くの生き物が上下方向に自由に行きかっている。水という流体の特性に応じて多くの生き物が重力に打つ勝つために様々な工夫を凝らしている。

テトラヒメナなどの鞭毛藻を含む数種の微生物では水溶液の表面に特徴的な多角形のパターンが形成されることが知られている。これを鉛直断面から観察すると微生物が上方運動と下降運動を行い、それによって対流が起きていることが見られる。このパターンが熱対流であるベナール型対流に見かけ上似ているため生物対流と呼ばれている。本研究の目的はこの現象を数値シミュレーションし、実際の観測の解析に役立てることにある。

### 2. 生物対流

テトラヒメナやクラミドモナスなどの微生物は周りの液体よりも重いものに関わらず、それらが水面近くに集まってくる。これらの多くは気液界面を目指しているのではなく、重力とは反対の方向に動こうとする性質を持つため、これを負の重力走性と呼ぶ。水面に集積した微生物の密度が培養液の密度より高いため微生物が落下することと重力走性により上昇する効果が重なり合い、培養液に密度不安定が生じて対流が起きる。

つまり、始めは均等に分布しているが、時間とともに次第に局所的な密度の隔たりができ、最終的には水面に全体にわたる一定の繰り返シパターンが形成される。

また、このパターンの形成は温度、pH、浸透圧、バクテリアの状態、紫外線、粘性、磁場、表面張力などとは無関係であることがわかっている。

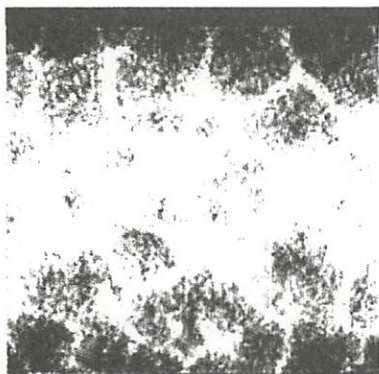


Fig.1: 生物対流による表面のパターン (実験)  
(白く見えるところが微生物の密度が大きい。)

### 3. 基礎方程式

生物対流は流れが穏やかで乱流を考慮しなくてよいので、連続の式(1)と非圧縮性 Navier-Stokes の方程式(2)を支配方程式として解析する。

$$\nabla \cdot \mathbf{v} = 0 \quad (1)$$

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} = K - \frac{1}{\rho_0} \nabla p + \nu \Delta \mathbf{v} \quad (2)$$

さらに微生物の固体密度に関する方程式(3)を加える。

$$\frac{\partial N}{\partial t} + \mathbf{v} \cdot \nabla N = \kappa_H \left( \frac{\partial^2 N}{\partial x^2} + \frac{\partial^2 N}{\partial y^2} \right) + \kappa_V \frac{\partial^2 N}{\partial z^2} - \omega_p \frac{\partial N}{\partial z} \quad (3)$$

$$\mathbf{v} = (u, v, w) \text{ 速度} \quad K = \left( 0, 0, \frac{\rho}{\rho_0} g \right) \text{ 外力}$$

$\rho_0$ : 培養液の密度  $\nu$ : 粘性係数

$N$ : 微生物個体数密度  $p$ : 圧力

$\kappa_H$ : 水平方向の拡散係数  $\kappa_V$ : 垂直方向の拡散係数

$\omega_p$ : 個々の微生物が上方に向かう速度

これらの方程式をMAC法を用いて解く。

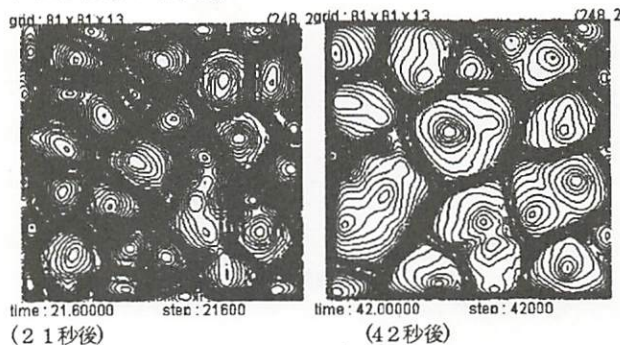
### 4. 計算結果

#### 4.1 1G 上での計算結果

容器の大きさは  $8 \text{ cm} \times 8 \text{ cm} \times 1.2 \text{ cm}$  とする。培養液が水であると仮定して密度を 1.0 とした。 $\alpha$  は細胞比重の測定から算出したもので  $5.0 \times 10^{-10}$  とした。 $\nu$  は水の動粘性係数を用いた。微生物の個体数密度の初期値は乱数で与え、 $10^5 \text{ cell / ml}$  に近くなるように調整した。水面上の微生物の密度を時間で等高線で示したものを Fig 2 に、鉛直断面内の速度を矢印で示したものを Fig 3 に表す。

#### 4.2 異なる重力下での計算結果

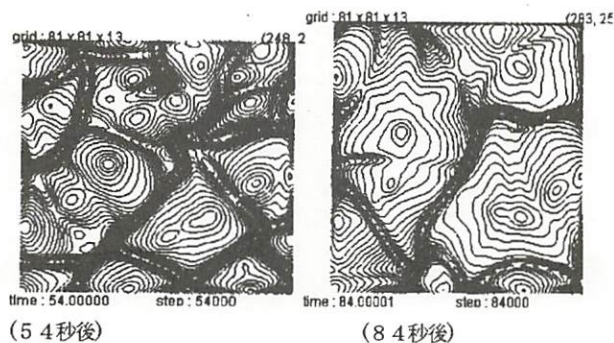
微生物に反重走性があることから重力に依存して動きが変わると予測し、重力を 0.2G、0.5G、2G、4G、6G と変えて計算を試みた。重力以外の変数は計算結果 1 のときと同じである。Fig4 は上の重力下における 1, 2, 3 回目までのパターンができるまでの時間を、Fig5 はパターンの大きさをグラフにしたものである。



(21秒後)

(42秒後)





(5.4秒後)

(8.4秒後)

Fig2: 水面上の微生物の密度

yiuu . 01 x 01 x 13

\205, 07



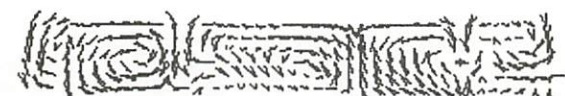
time : 21.60000

step : 21600

(2.1秒後)

yiuu . 01 x 01 x 13

\205, 07



time : 42.00000

step : 42000

(4.2秒後)

yiuu . 01 x 01 x 13

\205, 07



time : 54.00000

step : 54000

(5.4秒後)

Fig3 : 鉛直方面から見た速さのベクトル

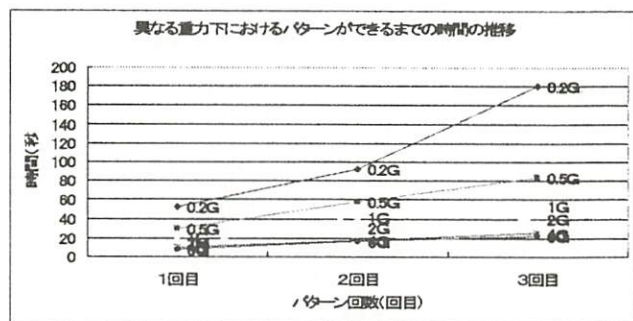


Fig4

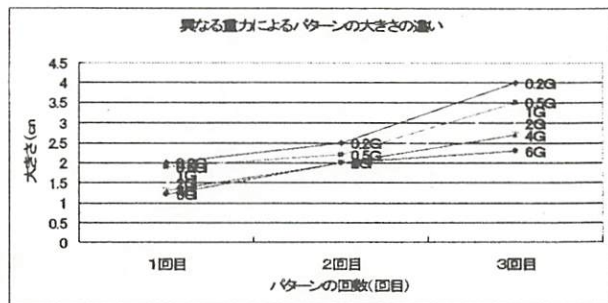


Fig5:

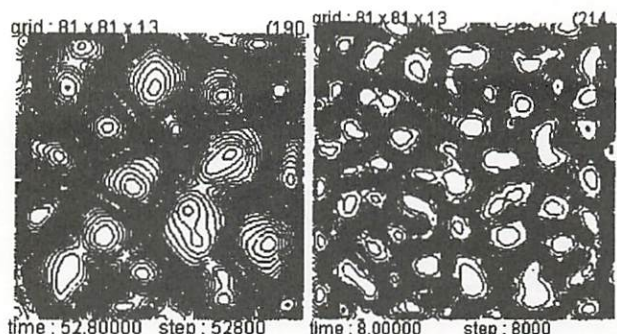


Fig6: 0.2G での一回目のパターン 6G での一回目のパターン

## 5. 考察

### 5.1 1G上での計算の考察

微生物は時間とともにその反重力性により上昇し、液体の表面に層が形成されている。その後、表面に微生物の塊がいくつもでき、1.8秒後あたりから落下し始める。2.0秒後には特徴的なパターンが形成されるが、しばらくするとパターンが崩され、2度目のパターンが作られる。1度目よりも2度目、2度目よりも3度目と回数を重ねるごとに大きなパターンが出来上がっている。

鉛直方向から見ると渦ができていのが見て取れる。渦と渦の間がパターンの帯のところに一致し、微生物の密度が大きかった。

パターンが崩れたときに底のほうの微生物の密度が一時的に大きくなるが、しばらくすると水面の密度がまた大きくなる。それを繰り返すことによりパターンが何度もできている。

### 5.2 異なる重力下での計算の考察

重力が変わってもできるパターンの形はほぼ同じであったが、重力が大きいほど小さなパターンが早くできる。重力が小さいと反重力性によって上に行こうとする力が小さくなるためだと考えられる。

## 6. まとめ

生物対流の数値シミュレーションを行うことにより、実験で見られたような特徴的なパターンを確認できた。また、重力が変化すると、パターンの大きさなどに変化が起こることもわかった。今後の課題として実際の数値に近づけたものを使い、実験でできるものと比較したい。

## 7. 謝辞

本研究を行うにあたり、ご尽力くださいました指導教員の河村先生、生物学科の最上先生、先輩方に感謝いたします。

## 参考文献

- 河村 哲也:「流体解析I」、朝倉書店、1996  
 安藤 常世:「工学基礎 流体の力学」、培風館、1973  
 兼子 小百合:「生物対流現象の数値シミュレーション」1998年度お茶の水女子大学修士論文、1999

# 血管手術における効果の数値的検証

田中 悠紀 (指導教員: 河村 哲也)

## 1. はじめに

近年、食べ物の変化により、動脈硬化や高血圧といった「血管に関わる病気」が増えている。

そこで、卒業論文の研究テーマとして、血管手術（特に浅側頭動脈-中大脳動脈吻合術）の様々な例をシミュレーションすることによって、どのような効果があるかを解析して検証した。浅側頭動脈-中大脳動脈吻合術 (STA-MCA) とは、脳の動脈が狭窄したり、閉塞したりして、脳の血流が十分でなくなった場合、それを補うために、頭皮の動脈血管（浅側頭動脈を使う）を脳の表面の動脈血管（中大脳動脈の枝を使うことが多い）につなぐ手術である。脳の血流が低下したまま放置しておくと、脳の血流がさらに低下したときに脳の神経細胞に十分な栄養が行かなくなり死亡に至るのを、この手術によってそのリスクを減らすことができる。

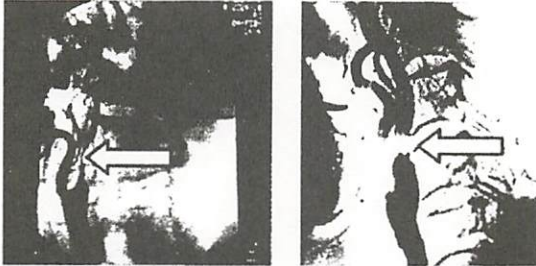


Fig.1: 頸部内頸動脈狭窄の二例 (矢印部分)

## 2. 研究の目的

一つ目の目的として、「つなぎ方（吻合の角度）を変えることによって、血管の圧力や血液の流れがどのように変化するか、その効果を検証」する。また、現在の STA-MCA では、手術を短時間で終わらせるために、浅側頭動脈の分岐血管の一本のみを中大脳動脈につなぐことが多いが、二つ目の目的として、「浅側頭動脈分岐血管二本ともに中大脳動脈につないだ場合の効果の検証」をする。

## 3. モデル化

中大脳動脈（内頸動脈の終枝）と浅側頭動脈（外頸動脈の終枝）をモデルにする。中大脳動脈は脳全体に血液を送る動脈であり、そこに閉塞ができてしまう（血液が完全に詰まってしまふ）と、脳梗塞になる。それを防ぐために、狭窄の状態、浅側頭動脈を中大脳動脈につなぎ、この浅側頭動脈から中大脳動脈へ血液を送るモデルを考えた。

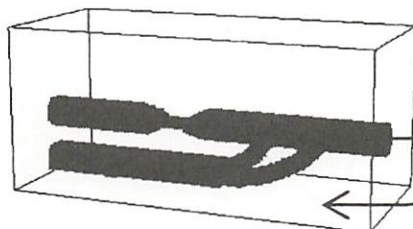


Fig.2: 血管モデル

## 4. 格子生成

格子は直交等間隔格子とする。計算に利用した格子は  $x$ ,  $y$ ,  $z$  方向に  $110 \times 50 \times 50$  である。

## 5. 計算方法

### 5.1 基礎方程式

血管内の流れの基礎方程式として、血液はニュートン流体であると仮定して、通常非圧縮性 Navier-Stokes 方程式(1), (2)を用いた。

$$\nabla \cdot V = 0 \quad (1)$$

$$\frac{\partial V}{\partial t} + (V \cdot \nabla)V = -\nabla p + \frac{1}{\text{Re}} \nabla^2 V \quad (2)$$

(1)が流体の質量の保存を表す連続の式、(2)が流体の運動量の保存を表す運動方程式である。ここで、 $V$  は流速、 $p$  は圧力、 $\text{Re}$  はレイノルズ数である。この基礎方程式を標準的な MAC 法を用いて解いた。すなわち、式(1), (2)から導かれる圧力のポアソン方程式(3)

$$\nabla^2 p^{n+1} = \frac{\nabla \cdot V^n}{\Delta t} - \nabla \cdot \{(V^n \cdot \nabla)V^n\} + \frac{1}{\text{Re}} \nabla^2 (\nabla \cdot V^n) \quad (3)$$

を用いて  $n$  ステップ目での速度  $V^n$  から未知の圧力  $p^{n+1}$  を求める。次の時間ステップの速度は式(2)の近似である式(4)

$$V^{n+1} = V^n + \Delta t \left\{ -(V^n \cdot \nabla)V^n - \nabla p^{n+1} + \frac{1}{\text{Re}} \nabla^2 V^n \right\} \quad (4)$$

から求めた。これに境界条件を組み込み、初期条件から始めて、繰り返し計算し、各時刻の速度と圧力を順次計算した。

### 5.2 血管の形状の計算法

血管は複雑な形をした物体であるため、形状を計算に組み込むのは用意ではない。そこで、血管を内部に含むような大きな長方形領域を考え、それを等間隔格子で分割した上で、血管の形状を現す三次元配列  $\text{MSK}(x,y,z)$  を別に用意した。配列  $\text{MSK}$  には血管内=1, 血管外=0, 血液の詰まった狭窄部分=0 という値を入力する。

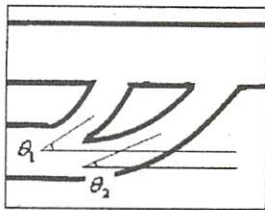
計算をする際に、初めは血管でない部分にも流れがあると仮定し、全ての格子で流れを計算する。そのようにして得られた速度の結果に先程作った配列  $\text{MSK}$  を掛け合わせる。その結果、血管内部である流体はそのままの値になり、それ以外の部分で、速度は0になる。

$$\text{MSK}(x,y,z)=1$$

$$\text{MSK}(x,y,z)=0$$

## 6. 計算結果

血管の流れの様子を以下の4通りの場合に分けて、解析を行った。また、中大脳動脈の半径=0.5, 浅側頭動脈=0.5, 浅側頭動脈分岐一本目の半径=0.25, 浅側頭動脈分岐二本目の半径=0.25 とし、分岐一本目の角度を $\theta_1$ 、分岐二本目の角度を $\theta_2$ とする。



### 6.1 つなぎ方 (吻合の角度) による解析

Case	$\theta_1$ の角度	$\theta_2$ の角度
1A	接続なし	$\theta_2 < 30^\circ$
1B	接続なし	$\theta_2 = 30^\circ$
1C	接続なし	$\theta_2 = 60^\circ$

### 6.2 分岐血管二本ともにつないだ解析

Case	$\theta_1$ の角度	$\theta_2$ の角度
2A	$\theta_1 = 60^\circ$	$\theta_2 = 30^\circ$

それぞれの Case において、

①繰り返し回数：20000回

②レイノルズ数：1000

③時間刻み  $\Delta t$ ：0.0005

④境界条件： $u=3.0$ ,  $v=0.0$

で計算し、その xy 平面 20000 回後の血管内の圧力結果と流れの結果を右に示す。

## 7. まとめと今後の課題

Case1A, 1B, 1C の3結果から、吻合の角度を急にすると、吻合された血管の上部分の圧力が高くなることを示せ、瘤のできる危険性があるので、吻合角度は緩やかなほうが良いと考えられることがわかった。また、Case1BとCase2Aより、分岐血管が一本の場合では、その吻合部分の左部に渦が出来やすくなってしまいが、分岐血管を二本吻合すると、一本の場合よりも渦ができにくく、流れが均等になりやすいことがわかった。

今後の課題としては、本研究では、簡単化のため、直交格子を使ったことにより、狭窄部分や吻合部分などが精度よく計算できていないので、格子生成方法を改めて考えて計算できるようにしたい。また、今回検証した以外での場合のシミュレーションも行っていきたいと思う。

### 謝辞

本研究をすすめるにあたりご指導いただきました、河村先生、本研究室の先輩方、脳神経外科の先生に深く感謝いたします。

### 参考文献

- [1]河村哲也:「流れのシミュレーション基礎!」山海堂 2002
- [2]田辺達三:「血管の病気」岩波書店 1999
- [3]藤田恒太郎:「人体解剖学」南江堂 1962
- [4]木村邦彦:「人体解剖学」大修館書店 1969

圧力の高低  $L=0.0$ ,  $H=5.5$



Fig.3: Case1A\_圧力

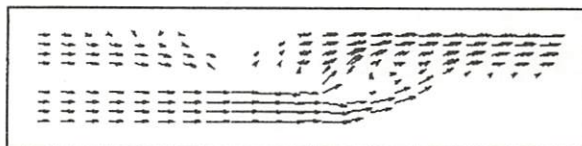


Fig.4: Case1A\_流れ



Fig.5: Case1B\_圧力



Fig.6: Case1B\_流れ



Fig.7: Case1C\_圧力



Fig.8: Case1C\_流れ

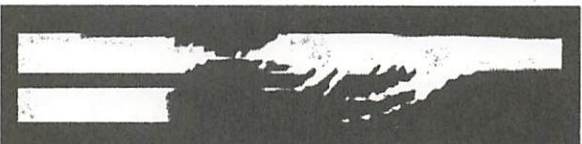


Fig.9: Case2A\_圧力

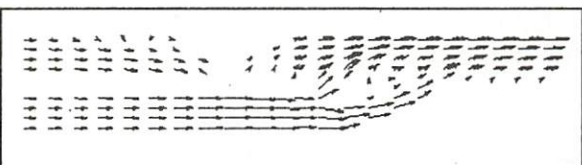


Fig.10: Case2A\_流れ

# 木星大気の循環のシミュレーション

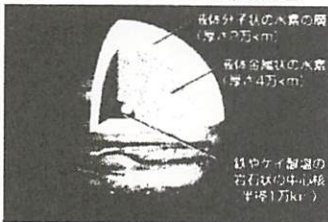
宮脇 梓 (指導教員: 河村 哲也)

## 1. はじめに

昨年、2年ぶりに飛行を再開したスペースシャトルに野口聡一宇宙飛行士が搭乗、活躍し、日本中が盛り上がった。また、小惑星いとかわには、はやぶさがタッチダウンに成功し、今後月や火星などにも探査機が送られる予定である。惑星探査もますます世界中で活発になることだろう。一方、私たちと同じ太陽系内惑星である木星はまだ未知の惑星である。木星はほとんどが気体や液体でできており、太陽系内では太陽に続いて巨大な星であり、太陽になりそになった天体とも言われている。本研究では、その木星の大気が、どのような循環をしていて縞や帯ができていのかをシミュレーションする事を目標とした。

## 2. 木星について

### 2-1. 木星の内部構造



中心(0km)-1万km : 岩石核  
1万km-5万km : 金属化した液体水素  
5万km-7万km : 液体水素  
7万km-7万1000km : 大気

表1. 太陽系惑星諸量の比較 (月を含む)

	軌道長径比	離心率	公転期比	赤道半径比	質量比	赤道傾斜比	密度比
水星	0.387	0.2056	0.24	0.382	0.055	0.38	0.98
金星	0.723	0.0068	0.62	0.949	0.815	0.91	0.95
地球	1.000 <sup>1)</sup>	0.0167	1.00	1.000 <sup>2)</sup>	1.000 <sup>3)</sup>	1.00 <sup>4)</sup>	1.00 <sup>5)</sup>
月	0.0026	0.0549	(27.3日)	0.272	0.012	0.17	0.61
火星	1.524	0.0934	1.88	0.533	0.107	0.38	0.71
木星	5.203	0.0483	11.86	11.21	317.8	2.37	0.24
土星	9.555	0.0555	29.46	9.45	95.16	0.94	0.13
天王星	19.22	0.0463	84.02	4.01	14.54	0.80	0.23
海王星	30.11	0.0090	164.8	3.88	17.15	1.11	0.30
冥王星	39.54	0.2490	247.8	0.178	0.0023	0.07	0.40

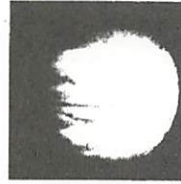
1) 1天文単位=1.496×10<sup>8</sup> km 2) 6378 km 3) 5.974×10<sup>24</sup> kg 4) 9.8m/s<sup>2</sup>  
5) 5.52 g/cm<sup>3</sup>

### 2-2. 大気の成分

組成は太陽と酷似している。  
水素: 90%  
ヘリウム: 10%

### 2-3. 木星の大気について

木星の特徴である縞模様は、明るい部分を帯、暗い部分を縞とよんでいる。また、太陽から受ける以上の熱が内部から供給されており、なおかつ自転しているため、大気は、“内部から熱いガスが運ばれ上昇し、冷やされて下降する”、という流れをしている。すなわち熱い上昇ガス流の部分が帯で温度が低くなっており、白っぽくなっている。逆にガスが下降して暖められる部分は縞である。自転しているの、帯と縞が赤道に平行に分布していることになる。



## 3. 現象のモデル化

大気部分での気体の流れを計算するため、大気のみを球面座標系( $R, \theta, \phi$ )を使って計算する。

ここで、木星液体水素までの半径方向を  $R$  方向とし、大気の厚みを半径の 1/10 とした。

## 4. 計算方法

### 4-1. 基礎方程式

質量保存を表す連続の方程式(1)、非圧縮性 Navier-Stokes 方程式(2)を支配方程式として解くことができる。

$$\nabla \cdot \mathbf{V} = 0 \quad (1)$$

$$\frac{\partial \mathbf{V}}{\partial t} + (\mathbf{V} \cdot \nabla) \mathbf{V} = -\nabla P + \frac{1}{\text{Re}} \Delta \mathbf{V} + \frac{Gr}{\text{Re}^2} \mathbf{T} \quad (2)$$

木星は中心部が高温で、太陽から受ける以上の熱が内部から供給されているため、内部熱を計算するためのエネルギー方程式(3)も用いて計算した。

$$\frac{\partial T}{\partial t} + (\mathbf{V} \cdot \nabla) T = \frac{1}{\text{Re Pr}} \Delta T \quad (3)$$

$\mathbf{V}$ : 回転速度  $P$ : 圧力  $T$ : 温度  $\text{Pr}$ : プラントル数  
 $\text{Re}$ : レイノルズ数  $Gr$ : グラスホフ数  
(レイノルズ数…物理的には流体の慣性力と粘性力の比を表す。)

本研究では、MAC 法を用いた。MAC 法は速度・圧力について直接 Navier-Stokes 方程式を解くため、3次元問題にも適用でき、境界条件が課しやすい。速度の計算には、高レイノルズ数においても安定した計算ができるように、Navier-Stokes 方程式の非線形項は、3次精度上流差分法(4)を用いて近似した。

$$f \frac{\partial u}{\partial x} = f \frac{-u_{i+2} + 8(u_{i+1} - u_{i-1}) + u_{i-2}}{12\Delta x} + \frac{|f|}{12} \frac{u_{i+2} - 4u_{i+1} + 6u_i - 4u_{i-1} + u_{i-2}}{\Delta x} \quad (4)$$

### 4-2. 格子生成

計算格子は、半径方向( $R$ 方向)に32分割、緯度( $\theta$ )方向( $0 \sim 360^\circ$ )に74分割、経度( $\phi$ )方向( $0 \sim 180^\circ$ )に76分割とした(Fig.1(a))。ただし、見やすくするため、結果の表示には Fig.1(b)のように半径方向にひきのばした格子を用いている。



Fig.1 経度方向の断面内の格子

## 5. 境界条件

木星表面では、 $\phi$  方向に回転速度  $v_\phi$  を与えた。角速度  $\omega=0.657$ 、半径  $R$  を 1 とし、 $v_\phi=R\omega$  で計算した。また内部の温度を 1 (無次元) で一定とし、大気上端では Free-slip、温度を 0 (無次元) とした。

## 6. 計算結果

以下の結果は  $Re=10000$ 、 $Pr=0.7$ 、 $\omega=0.657$  で計算したものである。また、 $t$  は無次元時間を表す。Fig.2 は経度方向の断面内の大気の循環の時間経過を熱の等値線 (シェーディング) で示す。

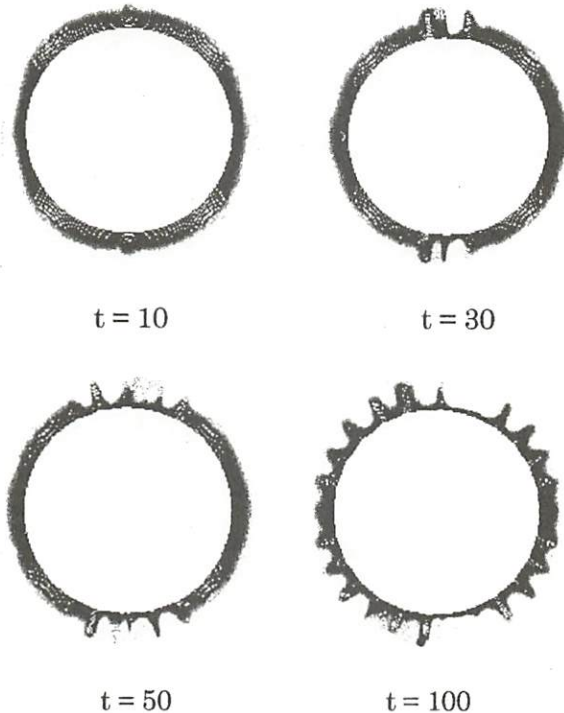


Fig.2 経度方向断面内の温度の時間変化

ここで、時間が経過するにつれて内部の熱が激しく上昇し対流が発生していることが分かる。

Fig.3 は、木星を外側から、温度(T)と圧力(P)についてそれぞれ比較した図である。ただし、高温部 (縞)・高圧部を濃く、低温部 (帯)・低圧部を薄くなるよう表示させている。

Fig.4 は、大気層の下にある、液体水素層の熱の動きを計算した結果であり、液体水素表面の温度分布と経度方向断面内の温度分布を示している。境界条件は、熱と回転速度は大気層の場合と同じで、厚みは実際の液体水素層の厚みと同じ比率とした。

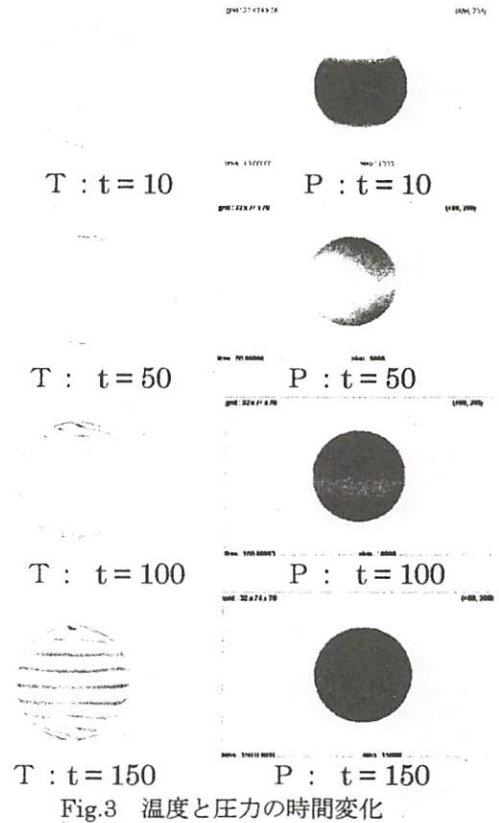


Fig.3 温度と圧力の時間変化

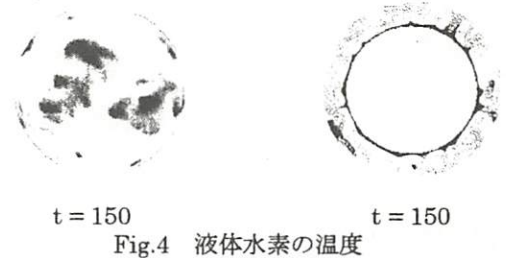


Fig.4 液体水素の温度

Fig.4 から内部熱により、液体水素層の表面は多少むらがあるものの温度がほぼ一定である事が分かる。よって、大気層を計算する時に“内部からの温度を一定”に計算したものは妥当だと考えられる。

## 7. まとめと今後の課題

本研究によって、木星のように内部が一様に高温で自転している場合、熱対流が発生して帯と縞ができることをシミュレーションによって示す事ができた。温度と圧力を比較したところ、時間が経過して模様が現れる頃から、圧力の高い領域と低い領域が、帯と縞にほぼ一致している様子が見て取れる。これにより、下から高圧の熱いガスが上昇し冷やされ、低圧になって下降し暖められていることが分かる。今後は、熱対流をさらに詳しく調べるために格子を細かくし、より木星の大気条件に近づけ、縞と帯の間に発生する渦をシミュレーションする予定である。

## 謝辞

本研究を行うにあたり、ご尽力くださいました指導教員の河村先生に深く感謝いたします。

## 参考文献

- [1]小尾信彌:「新・太陽系の科」、日本放送出版協会、1993
- [2]安藤常世:「工学基礎 流体の力学」、培風館、1973
- [3]松信八十男:「地球環境論入門」、サイエンス社、1998

# 温帯低気圧の簡易モデル

安田 史 (指導教員: 河村 哲也)

## 1 はじめに

私たちの生活と天気は密接に関係している。そして、その天気の変化の大きな原因は温帯地域に発生する温帯低気圧にある。そこで本研究では温帯低気圧に注目した。温帯低気圧は偏西風がきっかけとなり発生する。偏西風は赤道付近の暖かい空気と北極付近の冷たい空気の温度差により勢力を増し、南北に波打つように進む。気圧の谷が近づくと地上から近いところで反時計回りの空気の流れが生まれる。この流れの東側では暖かい空気が北へと移動して温暖前線をつくり、西側の冷たい空気は南へと回り込んで寒冷前線が生まれる。さらに気圧の谷が接近すると反時計回りの空気の流れも強くなって、ついに温帯低気圧が発生する。本研究では暖気と寒気をコリオリ力に見立てた力によってつりあわせることにより、不安定な前線面を作り出し、温帯低気圧の簡易モデルとみなす。長波長の波動がおきることとそのときの雲の状態を数値シミュレーションにより検証する。

## 2 モデル化

x 方向を東西方向、y 方向を南北方向、z 方向を高度とし、主に対流圏で起こる現象ということに考慮して高度よりも東西南北の領域を広くとり温帯低気圧をモデル化する。(Fig.1)

## 3 格子生成

格子は、直交等間隔格子とする。ただし、対流圏の空気の流れには鉛直方向の変化の結果が重要であると考え z 方向の間隔は x、y 方向の4分の1と細かくした。計算に使用した格子数は x、y、z 方向に  $80 \times 40 \times 80$  である。

## 4 基礎方程式

風の流れは、圧力をかけても縮まない非圧縮性の流れとみなすことができるため、連続の方程式(1)と非圧縮性ナビエ・ストークス方程式(2)およびエネルギー方程式(3)の3式を支配方程式として解くことができる。

$$\nabla \cdot \mathbf{v} = 0 \quad (1)$$

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} = -\frac{1}{\rho} \nabla p + \frac{\mu}{\rho} \Delta \mathbf{v} + \mathbf{f} \quad (2)$$

$$\frac{\partial T}{\partial t} + (\mathbf{v} \cdot \nabla) T = K \Delta T \quad (3)$$

$\mathbf{v}$ : 速度ベクトル  $p$ : 圧力  $\mu$ : 粘性率  $\rho$ : 密度

$\mathbf{f} = (2v\Omega \sin \phi, -2u\Omega \sin \phi, -g)$ : 外力項

{ $\Omega$ : 自転速度  $\phi$ : 緯度  $g$ : 重力加速度

$u$ : x 方向速度  $v$ : y 方向速度}

$T$ : 温度  $K$ : 熱拡散率  $t$ : 時間

本研究ではこれらの方程式の数値解法はフラクショナルステップ法を用いて解いた。方程式(2)の非線形項の差分近似については、3次精度上流差分を用いた。時間微分には前進差分、その他には中心差分を用いた。

## 5 簡易モデル

本研究では Fig.1 に示すように南北方向に暖気と寒気を鉛直面を境界として接して置き、暖気は緯度の低い赤道付近で暖められた空気、寒気は緯度の高い北極付近で冷やされた空気とみなした。この状態で時間を経過すると

暖気は上へ、寒気は下へと移動しようとする。しかし、実際の地球上では自転の影響により大気に対して力が働き、この見かけの力をコリオリ力という。

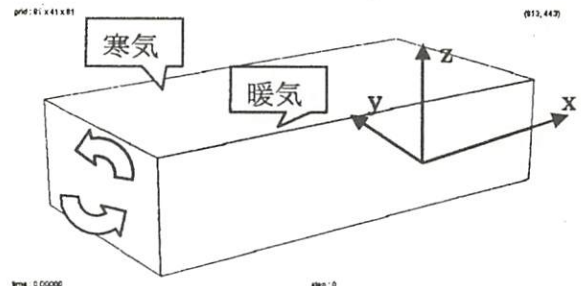


Fig.1:モデル化

## 6 コリオリ力

実際の偏西風の様子に一致するよう高度が増すほど速度が大きくなるよう設定する。速度が大きいくほどコリオリ力が強くなる。よって、コリオリ力の影響により暖気と寒気の移動はなくなり、つりあった状態になり初期の状態を保とうとする。ただし、このつりあいは不安定である。

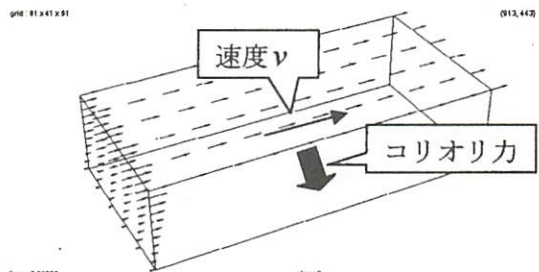


Fig.2:コリオリ力

## 7 雲の定義

雲は、大気中の温度が下がり、水蒸気が凝結したものである。本研究では大気中に無数にある水蒸気を温度を持った有限個の粒子と考え、その動きを追跡する。粒子の位置は  $\mathbf{r}^{t+\Delta t} = \mathbf{r}^t + \mathbf{v}^t \cdot \Delta t$  で求まる。ここで粒子の速度  $\mathbf{v}^t$  は周りの8つの格子点から補間するものとする。本研究では、各粒子が存在する周りの格子点における温度が、設定した温度を下回った時点で雲と判定する。本研究では降雨による水分量の減少はないと仮定しているため、温度が設定温度を超えるとまた水蒸気に戻るものとする。また、雲の発生に伴う流れの影響は考慮していない。

## 8 雲の出力

以上の方法では、各粒子ごとに速度・位置の計算をしているため、その数が計算時間に及ぼす影響は大きい。そこで少ない粒子での計算を実現するため、各格子において粒子の密集度を表す量を以下のように定義して、これを雲として出力することにした。ここで、 $d_i$  は  $i$  番目の粒子と格子点との距離とする。

$$Cld(j, k, l) = \sum_{i=1}^{imax} \exp(-d_i^2)$$

このように定義することにより、有限個の粒子の隙間をうめることができ、さらに一箇所に複数の粒子が集まっ

たときの様子を、より正確に表現できると考えられる。

## 9 計算結果

以下の結果は  $Re=18000$ ,  $\Omega=0.4$  で計算したものである。南北方向として、緯度が  $30$  度～ $60$  度の範囲を考えている。まず、Fig.3～Fig.4 に地表面付近の速度ベクトルの図を示す。

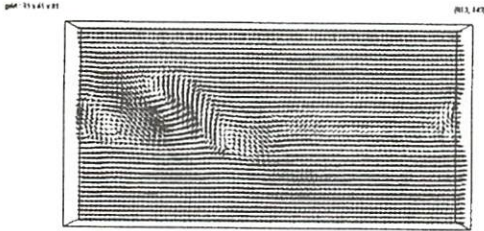


Fig.3:地表面付近の風速  $t=80$

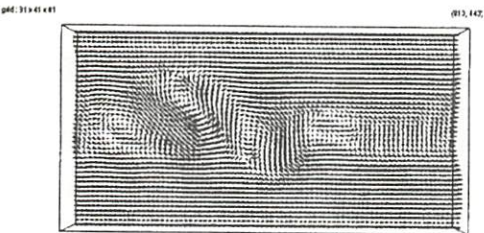


Fig.4:地表面付近の風速  $t=100$

ここではこのモデルでの偏西風波動にあたる波が大きく波打っている様子が見て取れる。Fig.5 に  $y$  方向から見た鉛直面内の等圧線を示す。

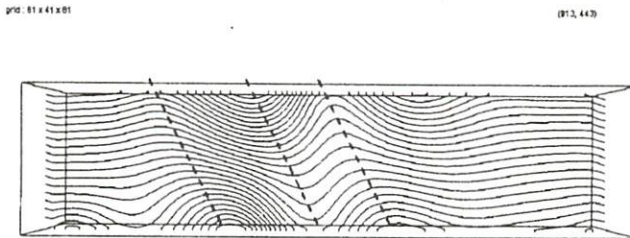


Fig.5:鉛直面内での等圧線  $t=100$

破線のように気圧の尾根と気圧の谷を見ることができ、発達しつつある偏西風波動の東西鉛直断面内の構造と一致する。

Fig.6～Fig.8 は地表面付近の等圧線である。

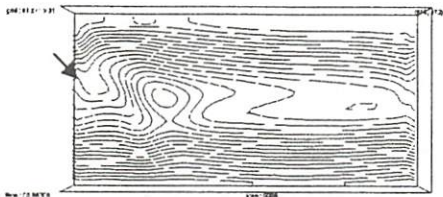


Fig.6:地表面付近での等圧線  $t=50$

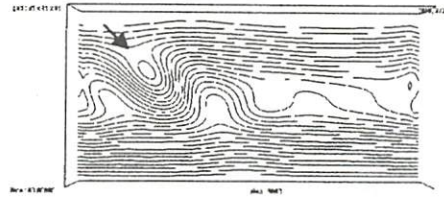


Fig.7:地表面付近での等圧線  $t=80$

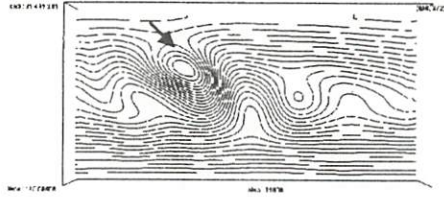


Fig.8:地表面付近での等圧線  $t=110$

温帯低気圧や移動性高気圧と思われる波動が西から東へ動く様子がみられる。Fig.9～Fig.10 は  $z$  方向高度の高いところから地表面に向かって見たときの雲と等圧線を表示させたものである。

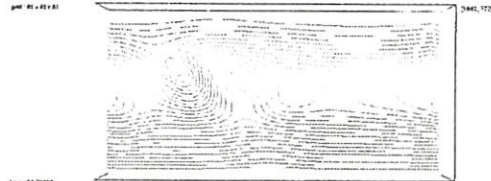


Fig.9:水平面内での雲と地表面付近の等圧線  $t=80$



Fig.10:水平面内での雲と地表面付近の等圧線  $t=90$

温帯低気圧と思われる波動の左側に常に厚い雲があることが見て取れる。温帯低気圧の左側に寒冷前線、右側に温暖前線ができるはずなので、寒冷前線にある厚い雲つまり積乱雲と考えることができる。

## 10 まとめと今後の課題

本研究で用いた簡易モデルによって、東西方向に気圧の尾根・谷および温帯低気圧、移動性高気圧と思われる波動が生じ、その発生や発達過程と雲の状態をシミュレーションすることができた。今後の課題として球座標系にすることでより実際の現象の状態に近づけること。雲の表示は単純な温度によるものであったが、鉛直方向の動きに応じて露点を考慮した雲の表示にすること。雲の発生による流れの影響を考慮すること。計算領域などを実際の大気条件に少しでも近づけることなどがあげられる。

### 謝辞

本研究をすすめるにあたり、ご尽力くださいました指導教員の河村先生、諸先輩方に深く感謝いたします。

### 参考文献

- [1] 河村哲也: "流体解析 I", 朝倉書店, 1996
- [2] 河村哲也: "流れのシミュレーションの基礎!", 山海堂, 2002
- [3] 土屋なお子: "安定成層中の山越え気流による雲の発生", お茶の水女子大学卒業論文, 2005
- [4] 岸保勘三郎・田中正之・時岡達志: "大気の大循環", 東京大学出版会, 1982
- [5] 小倉義光: "一般気象学", 東京大学出版会, 1984

# RFIDを用いた個人別情報提供システムの構築

矢野 智美 (指導教官: 増永 良文)

## 1 はじめに

RFIDとはRadio Frequency Identificationの略で無線ICタグを対象となる人や物に取り付け、無線通信を用いて対象物の認証・管理を行うシステムのことで、近年RFIDを用いた技術の需要が高まってきており、様々な分野で用いられ始めている[1].

一方、サービス業界において、顧客管理(CRM:Customer Relationship Management)が重要なキーワードとなっている[2].これは、顧客のことを十分に理解し、その顧客一人一人にあった製品やサービスを提供することにより顧客満足度を高め、継続したビジネスへとつなげていくことを目的とした経営手法のことである.

そこで、本研究では顧客にあった、顧客の為だけの情報を、顧客の欲しい時に提供するための「RFIDを用いた個人別情報提供システム」を提案する.本システムでは、特に綿密な顧客管理が重要視されると思われる百貨店業界を舞台に設定する.ある特定の百貨店に来店した顧客が、店舗入口に設置されているリーダにタグをかざすと、その顧客の嗜好に合った催事情報や商品情報が顧客の携帯電話にメール配信されるというものである.このシステムが、付加価値を高めて消費者に選択されるサービスにつながるであろうと考える.本稿ではこのシステムの概要、実装、今後の課題等を述べる.

関連研究として上海婦人用品商店は、2005年3月27日のリニューアルに合わせRFIDを活用する最先端のCRMシステムを導入した.NECをはじめ、凸版印刷などのIT系各企業の技術で構築されたシステムは、接客向上と顧客満足向上に大きな役割を果たすものと考えられる[3].

## 2 システム概要

### (1) 事前準備

RFID情報提供サービスを開始するに先立ちサービス利用を希望する顧客に下のアンケート(図1)に答えてもらう.このアンケートは実験対象の伊勢丹新宿店で定期的に行われる催事のキーワードを参考に選択肢としたものである.そのアンケート情報を顧客データベースに登録する.

### (2) 来店情報獲得

図 1: 顧客アンケート

百貨店店舗にRFIDリーダ/ライタを設置しておく.来店した顧客が店内のリーダ/ライタにICタグ(顧客の携帯電話に装着されているものとする)をかざす.

### (3) 顧客情報獲得

顧客のICタグから取得したIDにもとづき、顧客データベースからその顧客の名前・メールアドレス・キーワードを取得してくる.

### (4) 顧客別催事情報獲得

顧客キーワードを検索キーワードとして、百貨店ホームページ(モバイル用)からGoogle検索をする.

こうして得られたWebページを顧客向けの情報と考える.

### (5) メール配信

Google検索結果のうち、記事名とURLを顧客の携帯電話にメール送信する.

顧客はメール本文のURLをクリックすると、情報のあるWebページを閲覧することが出来る.(図2)

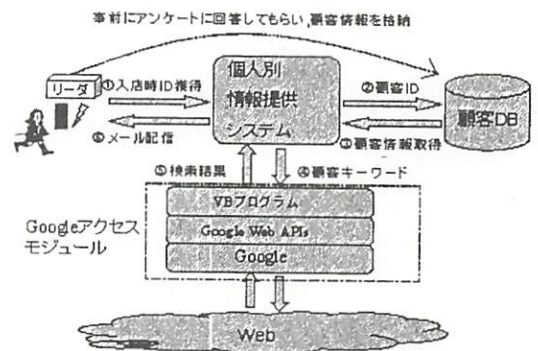


図 2: システムの概要図

## 3 システムの構築

### 3.1 実装環境

以下の仕様と機器でシステムを構築する.

- ・ 開発言語: Visual Basic.NET 2003 Professional
- ・ データベースシステム: MySQL Server 5.0
- ・ ICタグ: TesStar(TOPPAN2.45GHzRFIDシステム) 対応
- ・ RFIDリーダ/ライタ: TesStar 対応
- ・ FOMA 端末: 全角 5000 文字までのメールを受信出来る



### 3.2 顧客プロフィール

顧客データベース (図 3) の各属性を以下に示す [4].

- 顧客 ID: 唯一識別子として IC タグの UID を格納
- 名前: メール配信時, メッセージを表示させるため
- 性別: 女性の場合「レディス」男性の場合「メンズ」をキーワードとして登録
- メールアドレス: 顧客の携帯電話のアドレス
- 生年月日
- サイズ: S・M・L のいずれかをキーワードとして登録
- その他: 「リクルート」, 「ブライダル」, 「マタニティ」, 「ベビー」等のキーワードから関心があれば選んでもらい, 選択されたキーワードをそのまま登録.

id	name	mailaddress	birth	size	keyword	keywo
F82202928003407	masuzaki yoshiko	tokoni@h.its.ocha.ac.jp	1980-02-12	s	bridal	baby
F82202928004402	yano tomasi	toanoni-bd_1983.10.14-hh_ab@docomo.ne.jp	1983-10-14	s	lady	NULL

図 3: 顧客データベース

### 3.3 情報検索

百貨店ホームページモバイル版を対象に Google Web APIs を用いて Google 検索をする. 検索キーワードとして顧客データベースから取得してきたキーワードを入力. また, すでに終了してしまっている催事情報を得ないようにするため, daterange 構文を用いて日付範囲指定検索をする [5]

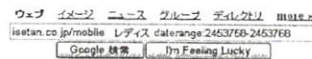


図 4: 検索実行例

### 3.4 メール送信

メール送信プログラムにより, 取得してきた催事情報 (タイトルと URL) を本文とするメールを送信する. 仮に一致する情報が得られなかったケースがあったとしても, 来店への感謝の言葉となる.



図 5: 送信メール例

### 3.5 実験と考察

今回, 百貨店を伊勢丹新宿店として実験を行った. まずキーワード (レディス, ブライダル, ベビー) で検索を行う. それぞれ 2~16 件の検索結果が得られた. しかしこれ

らの中にはすでに終了している催事の情報などが含まれていた. そこで日付範囲指定検索を行うことにした. ここで日付の範囲の選択が問題になると考え, まず範囲を一日に指定した. 前回よりもかなり絞られて有効な情報が得られた. しかしこの中にも終了済みの催事情報が混じってしまった. この日付範囲指定検索で用いられる日付というのはあくまでもページの内容が Google データベースに登録された日のことで, ページの内容の作成日とはまったく関係がないためだと考えられる. また, ホームページ中の「What's New」のページから手動で検索を行うのであれば有効な情報のみを得ることが出来たが, Google 検索からでは, トップページに最新情報が掲載されていないと取得が出来ない. ここで, 伊勢丹以外にモバイルサイトを持つ百貨店を調べ (高島屋, 三越, 大丸, 松屋) 同様の実験を行った. しかし, トップページに最新情報が掲載されているサイトはなかったため, やはり日付範囲指定検索である程度絞るやり方が良いだろうと考えられる.

## 4 まとめと今後の課題

本稿では, 百貨店を訪れる顧客に対して, 入店と同時に顧客向けの情報をメール配信するシステムを提案し, その主要な機能を実装した. 本システムではかなり限られたキーワードで情報検索を行っている. しかし, より充実した情報を顧客に提供するためには, より豊富なキーワードを用意し, 顧客のプロフィールをより細かく獲得すべきだと考える. また, 購入履歴を基にキーワードを追加していくなど RFID の特性を活かした様々な応用サービスも実現する必要がある. さらに, 配信メールに関連する画像などを添付するといったような様々なコンテンツを増やしていくことが可能であると思われる. こういったことが実現すれば, 当初の目的であった付加価値を高めて消費者に選択されるサービスになるであろう.

### [謝辞]

最後に本研究を進めるにあたりご助言をいただいた, 本学情報科学科講師の渡辺知恵美先生, 本学大学院の金野紋子先輩に深く感謝致します.

### 参考文献

- [1] "TOPPAN FORMS RFID-ing"  
<http://rfid.toppan-f.co.jp/solu/index.html>
- [2] "IBM みんなの RFID GTO"  
<http://www-6.ibm.com/jp/gto/feature/rfid/index.html>
- [3] "RFID テクノロジー - 無線 IC タグの実用化情報"  
<http://itpro.nikkeibp.co.jp/rfid/index.html>
- [4] 増永良文 "リレーショナルデータベース入門 [新訂版]" (2003), サイエンス社
- [5] Tara Calishain, Rael Dornfest "GOOGLE HACKS" (2004), オーム社

# ウェブコンテンツを活用した 仮想人物応答システムの研究

郡司 京子 (指導教官: 増永 良文)

## 1 はじめに

近年、古代の遺跡や都市を仮想世界で再現する技術が進んでいる。そこには人間が住んでいたはずだが、それらは再現されていない。人が再現された場合、ユーザはその人にさまざまな問合せをしたくなる。しかし仮想人物のデータを事前に揃えておくことは不可能に近い。

一方、近年インターネットが普及しウェブ上には大量の情報が存在しており、その情報発信者は企業や団体のみに限らず個人にまで及んでいる。さらに2002年春、Google Web API がリリースされたことにより、ウェブ上の豊富で膨大な量のコンテンツを対象として、さまざまな検索とそれに基づいた処理を行うことができるようになった[1]。

そこで本研究ではこれらの情報と機能を活用して過去に存在した人物をコンピュータに仮想的に作り上げて対話することを目標として、先行研究を拡張し、オンライン化によるリアルタイムな対話を実現できるシステムを構築する。

## 2 仮想人物応答システム～先行研究～

先行研究として、古和美由紀はインターネット上のウェブコンテンツによる歴史上の人物の情報を活用して、その歴史上の人物を仮想的に作りあげ、事実に関する単純な質問(時間の質問)に対して対話できるシステムの実装を行った[2]。仮想人物は“お札”の人物(福沢諭吉、樋口一葉、新渡戸稲造、夏目漱石、野口英世)に絞られている。古和のシステムの概念を図1に示し、その要点を以下に示す。

1. GoogleWebAPI を使用して人物関連ページの URL を取得し、それらのページをあらかじめ手動でダウンロードする。
2. ダウンロードしてきたファイルの HTML タグを除去し、プレーンテキスト化する。
3. 対話インタフェースの質問文からキーワードを抽出し、そのキーワードの類義語も交えて検索式を作成。
4. キーワード及び類義語の検索式でプレーンテキストを全文検索することによって、該当のセンテンスを得て回答候補を抜き出す。
5. 回答候補を集めランキング化し、最上位のものを回答として対話インタフェースに返す。

先行研究では事実に関する単純質問(時間の質問のみ)、例えば「誕生日はいつですか?」に対する実装ができていた。そして、音声によるインタラクションを実現した。しかし、先行研究では関連ページ取得とそのページの HTML タグ除去をして、プレーンテキストファイルを用意することをあらかじめ手動で準備しておく必要があった。これをオフラインでの実装ということにする。

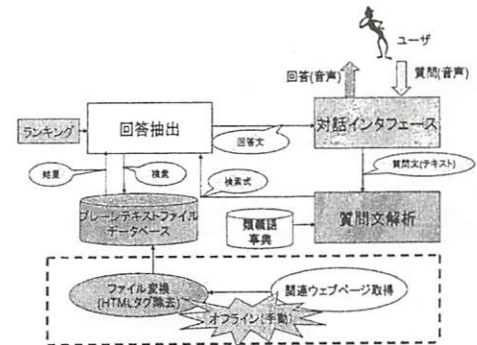


図 1: 先行研究システム概念図

## 3 オンライン化仮想人物応答システム

本研究では仮想人物の対話を、先行研究の課題としても挙げられていたが、対話とはリアルタイム性があること対話であるのではないかという観点から、先行研究のオフラインのモジュールをオンライン化しようということである。ここでいうオンライン化とは人物関連ウェブページ自動取得、取得した HTML ファイルの即自動 HTML タグ除去によるプレーンテキストファイル化を一連のシステムの流れに組み込むことをいう。また先行研究に倣い、本研究でも「時間」に関する質問を想定する。オンライン化された仮想人物応答システムの概念を図2に示す。このようにして、オンライン化により仮想人物とのリアルタイムな対話を図ろうと考える。

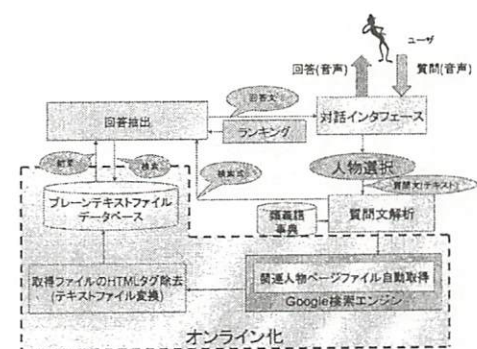


図 2: オンライン化仮想人物応答システム概念図

### 3.1 ウェブページ自動取得

対話インタフェースの人物選択より GoogleWebAPI を使用して Google 検索エンジンに自動アクセスし、該当の人

物関連ページ URL を取得する。そしてウェブから人物関連ページを得て、そのウェブページをおのおの HTML ファイルとして自動でダウンロードする [3][4]。本研究では Google 検索上位 10 件のページをファイルとしてダウンロードしている。

### 3.2 回答抽出

ダウンロードした HTML ファイルから即自動的に HTML タグを除去する。その結果がプレーンテキストファイルデータベースに格納される。回答抽出モジュールは対話インタフェースより得た質問のキーワード及び類義語の検索式によって HTML タグを除去したプレーンテキストファイルをセンテンスごとに分析し、回答候補があるセンテンスを抽出する。そして抽出されたセンテンスから回答候補を抜き出す。回答候補をランキング化し、最上位のものを回答とする。

### 3.3 対話インタフェース

回答抽出モジュールから返ってきた回答を出力する。ユーザより受け取った質問からのキーワードから全文検索して返ってきた回答を音声出力変換ソフトで読み上げる。例としては、「誕生日はいつですか?」を受け取り、「天保 6 年です。」などと音声により出力する。

### 3.4 実装環境

以下の仕様で対話システムを構築する

- ・ OS: WindowsXP Home Edition
- ・ 音声入力変換ソフト: AmiVoice SDK 5
- ・ 自動ウェブページ取得: 言語 Java
- ・ データ検索: 言語 Perl
- ・ 対話インタフェース: 言語 VisualBasic
- ・ 音声出力変換ソフト: NEC SmartVoice6.0

### 3.5 実験結果

構築したシステムを使って実験を行った。最初、例えば「福沢諭吉」と発話して、人物を指定する。続けて、「誕生日はいつですか?」と発話する。システムが質問処理を行い、「天保 6 年です。」と答える。様々な人物を想定して、同様な実験を行ったところ、平均 3 分を要した。人間の対話に関して許容待ち時間は平均 3 秒といわれているため、大幅な時間ロスとなっている。オンラインでリアルタイムでファイルを取得、即自動変換することはできてレスポンスタイムの短縮も課題となる結果になった。平均 3 分要した原因については現在検討中である。そして正答率も、ファイル取得数が少ないために先行研究の結果と比べてやや低い結果となった。オンラインでリアルタイムで時間はかかったが正答が導けた例を図 3 に示す。



図 3: 対話結果画面

## 4 まとめと今後の課題

本研究での対話インタフェース、質問文解析とウェブデータ検索のプログラムは先行研究をベースにした。本研究で新規開発した部分は Java で GoogleWebAPI から Google 検索上位 10 件の URL を取得し、それらのウェブページ 10 件をファイルとして自動ダウンロードするプログラム、古和の質問文解析、回答抽出をする Perl プログラムの改良、そしてそれらを連結してオンラインでシステム全体が稼動するようにするプログラムである。その結果、先行研究の課題のうち、オンラインによる実装を達成することができた。しかし、オンライン化したものの回答時間に問題があること、参照ウェブページが 10 件ということもあり正確な答えを導くのが難しい場合もあるという結果となっている。また、もう一つの課題の「場所」、「イベント」に関する質問には対応できていないのでこちらも充実させたシステムにする必要がある。

### [謝辞]

本研究を進めるにあたり貴重なご助言をいただいた本学情報科学科講師の渡辺知恵美先生、大学院の榎美紀先輩、隈田知代先輩に深く感謝致します。

### 参考文献

- [1] Tara Calishain, Rael Dornfest: “Google Hacks-プロが使うテクニック&ツール 100 選”, オライリー・ジャパン, 2003.
- [2] 古和美由紀: “ウェブコンテンツを活用した仮想人物応答システム, お茶の水女子大学理学部情報科学科卒業論文, 22p, 2005.
- [3] “Java で HelloWorld!ファイル入出力編”  
<http://www.hellohiro.com/fileio.htm>
- [4] “Java で HelloWorld!URL 編”  
<http://www.hellohiro.com/url.htm>

# MP3データの楽曲を対象とした検索システムの構築

長澤 槇子 (指導教官: 増永 良文)

## 1 はじめに

近年, iPodやネットワークウォークマンなどのMP3プレーヤが流行している. MP3とは, デジタル化された音声を圧縮する技術の一つであり, CDなどの音声を, 約10分の1のデータサイズまで圧縮することができる規格である. したがって, MP3を用いることにより, ハードウェアの負担も大幅に削減され, 無理なく音楽を再生することが可能となる. また, 圧縮するので, たくさんの楽曲をパソコンに取り込んだり, 手軽に持ち運ぶことが可能となる. それに伴い, 大量の楽曲のデータをユーザの分かりやすいように管理し, ユーザの聴きたい曲を効率よくスムーズに検索できる体系を備える必要がある.

既に, iTunesやSonicStage, Winampなどの音楽プレーヤソフトには, 様々な条件による楽曲検索機能がある. 例えば, アーティスト名やアルバム名, 再生回数の多い曲や少ない曲の検索などである. しかし, それらの楽曲検索機能に, 歌詞を検索条件とした機能はない. そこで本研究では, 歌詞を検索条件とした楽曲検索システムを構築することとした.

このシステムが実現すれば, ユーザが曲名やアーティストの名前が分からない場合でも, 歌詞の一部さえ分かっていたら楽曲の検索が可能となる.

## 2 関連研究

音楽検索システムに関する関連研究を以下に紹介する.

- (a) 旋律による楽曲検索システム [1]
- (b) ハミングによる楽曲検索システム [2]
- (c) 感性語による楽曲検索システム [3]

(a) では, 曲データの旋律を時間順に音高と音価に記号化した文字列で表し, テキストベースの旋律データベースにすることにより検索を行う. このシステムでは, 楽曲のデータを記号化するのに大変時間を要するという難点がある.

(b) では, ユーザがマイクからハミングを入力し, 採譜ソフトによりMIDIに変換され, 類似検索をし, 結果をクライアントPCに返信するというものである. データベースにはMIDI形式のメロディデータを使用

する必要がある.

(c) では, 楽曲構成要素(調, テンポ, 旋律, 音高, 和声, リズム)から楽曲の特徴量を抽出し, 「悲しい」や「明るい」といった感性語で問合わせをし, 最も特徴量の相関の高い楽曲を出力するといったものである.

その他, サビ区間の検出やメロディの抽出などの音楽検索システムに関する研究が行われている.

## 3 楽曲検索システムの概要

### 3.1 システム構成

本システムの構成を図1に示す. 楽曲一曲につき一つのテキストファイルを準備し, その楽曲の歌詞そのものをテキストデータとして格納しておく. 歌詞データの取得にはiTunesと連携させて自動で歌詞を取得することが出来る, フリーソフトのLyricsMasterを使用した. iTunes内にも歌詞データを入れておくことができるが, あらかじめ歌詞データの入ったファイルを歌詞データベースに格納しておき, インデックスを作成することによって検索の効率化を図った.

まず, 問合せインタフェースにユーザがキーワードを入力する. 検索モジュールにより, 入力されたキーワードを歌詞に含む楽曲を検索し, 検索結果として(一般に複数の)楽曲名とアーティスト名を表示させる. 表示された結果からユーザが聴きたい曲を一曲選択し, ユーザのiTunes内のライブラリの中から, 選択した曲とマッチする曲を再生する. ライブラリには, 楽曲名やアーティスト名, 音情報などが格納されている.

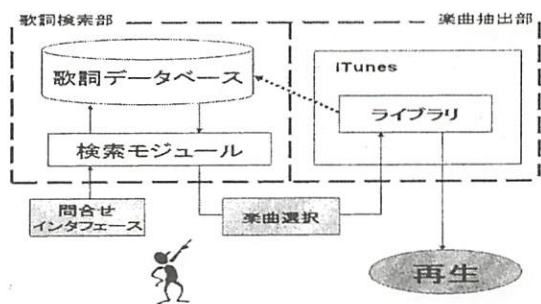


図1: 楽曲検索システムの構成

### 3.2 Namazuを用いた歌詞データの全文検索

ユーザが要求しているキーワードを歌詞の中を含む楽曲を検索するために, ファイルの全文検索を高速に行うことができるサーチエンジンであるNamazuを

使用する。Namazuの全文検索の仕組みは、あらかじめ対象のファイル群（本システムでは歌詞データベース）を基にインデックスを作成し、検索時にそれを参照する。インデックスには、例えば、“未来”という単語が含まれるファイルはAとB，“永遠”という単語が含まれるファイルはAとC，というような情報が格納されている。“未来”という単語を検索するときは、AとBというファイルを探し出し表示する。Namazuの検索の仕組みを図2に示す。

またインデックスの作成では、どの単語がインデックスに登録されるかが重要なポイントとなる。Namazuでは日本語をわかち書きにし、インデックスを生成している。本システムでは、KAKASIを使用した。KAKASIとは、漢字かなまじり文をひらがな文やローマ字文に変換するプログラムであり、日本語をわかち書きにするためのツールである。例えば、「新たな未来」という文は、「新た な 未来」というようにわかち書きされ、「新た」、「な」、「未来」がキーワードとしてインデックスに登録される。

Namazuは、HTMLやtext、Word、Excel、PDFなどテキストファイル全般を扱うことが可能である[4]。

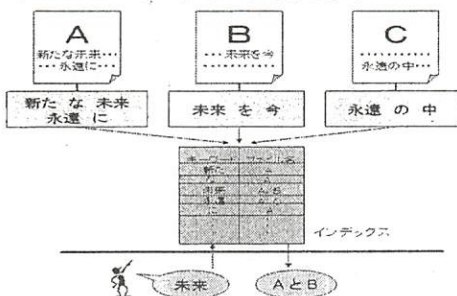


図2: Namazuの検索の仕組み

## 4 楽曲検索システムの実装

### 4.1 実装環境

- 問合せインタフェース: Visual Basic.NET
- 歌詞検索: Namazu
- 楽曲名・アーティスト名取得: Perl
- 音楽プレーヤソフト: iTunes 6.0.2

### 4.2 ライブラリからの楽曲抽出法

本システムでは検索により取得した楽曲名とアーティスト名から、自分の所有しているiTunes内のライブラリの中にあるMP3データの楽曲を再生することを実現している。iTunes内のライブラリからの楽曲抽出には、音楽再生ソフト開発ツールであるiTunes COM Windows SDK(Apple社)を使用した。以下に使用した関数について説明する。

- LibraryPlaylist()  
iTunesのメインライブラリを返す
- ItemByName()  
指定されたタイトル名の楽曲を抽出

- Play()

指定された楽曲を再生する

まずiTunes内のライブラリをLibraryPlaylist()により取り出し、ItemByName()を使用し、ユーザの選択した曲をライブラリ内から抽出する。最後にPlay()で楽曲を再生する。

### 4.3 実装結果

図3はこれまで述べてきた手順で実装を行うことによって得た楽曲検索システムの実装結果である。検索結果画面に表示されている楽曲名とアーティスト名から、ユーザの聴きたい曲を一曲選択し、再生ボタンをクリックすれば、iTunesと連動し曲が再生される。また、複数のキーワードを入力し、「AND」、「OR」、「NOT」を選択することにより、AND検索、OR検索、NOT検索が可能である。歌詞の詳細を見たい場合は詳細表示ボタンをクリックすれば表示されるようにした。



図3: 楽曲検索システムの実装結果

## 5 まとめと今後の課題

本稿では、ユーザが入力したキーワードを歌詞を含むMP3データの楽曲を対象とした楽曲検索システムを提案し、実装した。今後は、メロディデータを検索条件として、楽曲を検索するシステムの構築を実現したい。そして、今回構築した歌詞をキーワードとした検索と合わせて、キーワードに“未来”という歌詞をもち、メロディが“ドミノ”である楽曲というような検索を行う検索システムを構築していきたい。

### [謝辞]

本研究を進めるにあたりご助言・ご助力いただいた、本学情報科学科講師の渡辺知恵美先生、本大学大学院の隅田知代先輩に深く感謝致します。

### 参考文献

- [1] 山本順人: “音楽データベース”, 情報処理, Vol.29, No.6, pp.599-607, 1988
- [2] 小杉尚子, 小島明, 片岡良治, 串間和彦: “大規模音楽データベースのハミング検索システム”, 情報処理学会論文誌, Vol.43, No.2, pp287-298, 2002
- [3] “感性語(印象語)による検索”  
<http://www.slis.keio.ac.jp/ueda/semi/99onsei.html>
- [4] 西村めぐみ: “オープンソースソフトウェアによる全文検索データベース Web の作り方”, ソシム, 2000

# IP-SAN を用いたアプリケーション実行性能と TCP パラメータの相関関係評価

千島 望 (指導教員:小口 正人)

## 1 はじめに

近年、インターネット技術の進展などにより蓄積され利用されるデータ容量が爆発的に増加している。これに伴いストレージの増設、管理コストの増大が問題となっている。そこでストレージネットワークが登場し、その代表的なものとして FC-SAN(Fibre Channel - Storage Area Network) が広く用いられている。一方、SAN に IP ネットワークを利用した IP-SAN として iSCSI が期待されている [1]。

iSCSI は、これまで DAS(Direct Attached Storage) で使われてきた SCSI コマンドを TCP/IP パケット内にカプセル化することにより、サーバ (initiator) とストレージ (target) 間でデータの転送を行う。しかし、iSCSI は複雑な階層構成のプロトコルスタックで処理されており、またバースト的なデータ転送も多いことから、通常のソケット通信と比較して、特に高遅延環境においては性能の劣化が著しい。また、下位基盤の TCP/IP 層が提供できる限界性能を超えることはできず、最大限の性能が発揮できるよう TCP パラメータなどを制御することが求められる。

これまで iSCSI 環境におけるシーケンシャルアクセスの性能評価が行われ、TCP パラメータの制御手法が提案及び評価されてきた [2]。しかしアプリケーションを実行した時の iSCSI 環境における TCP の振舞についてはあまり知られていない。そこで本研究では、iSCSI を用いたストレージアクセスを含むアプリケーションの実行時の、TCP パラメータとアプリケーション実行性能の相関関係を評価した。

## 2 Linux TCP 実装

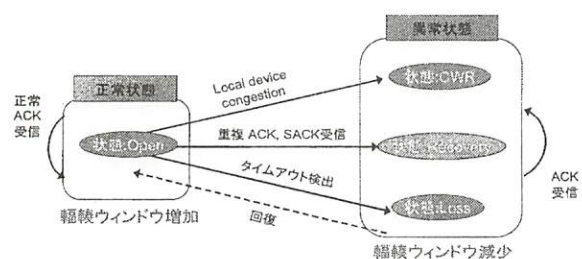


図 1: Linux TCP の状態遷移

TCP では、通信能力の制御にウィンドウサイズという概念を用いている。ウィンドウサイズとは、ホストが ACK なしに一度に送信できるデータのサイズで、TCP ヘッダに含まれる。また、データの送信側では輻射ウィンドウ、受信側では広告ウィンドウという値が決定され、このどちらか小さい方がウィンドウサイズとして用いられる。広告ウィンドウは現在の受信ウィンドウの空き容量を示しており、ACK で送信側に送られる。一方、輻

射ウィンドウは送信側の制御パラメータで、ネットワークの混雑を回避するため送信側が自主的に制限する値である。輻射制御ではこの輻射ウィンドウが利用されている。

本実験で用いた Linux OS においては、通信時の状態が正常であれば ACK 受信ごとに輻射ウィンドウは増加するが、エラーが検出されると異常と判断され、輻射ウィンドウは低下する (図 1)。輻射ウィンドウが低下する原因としては、送信側デバイスドライバのバッファが溢れることによる Local Congestion エラーを検出した場合 (CWR)、重複 ACK 又は SACK を受信した場合 (Recovery)、タイムアウトを検出した場合 (Loss) の 3 つが挙げられる。また、Linux の TCP 実装では、通信中に一度設定された輻射ウィンドウは、そのウィンドウの値を使い切らない限りは変化しないという特徴を持ち、この時スループットはほぼ一定の値で安定することが確認されている。

## 3 PostMark

本研究では、ネットワークストレージを用いる評価アプリケーションとして PostMark を利用した。PostMark はファイルシステムのベンチマークソフトウェアで、NetApp 社が実装/配布を行っている [3]。PostMark は主にインターネットサーバプログラムの性能評価を想定している。ファイルを作成した後、ファイルに対する読み書きの操作を繰り返すベンチマークであり、I/O の性能評価という側面が強い。

設定を変更できるものは次の値である。

- ・ファイル数
- ・ブロックサイズ
- ・試行回数
- ・読込追記と作成削除の比
- ・ファイルサイズ
- ・ランダム数発生種

本研究では、アプリケーションとしてこの PostMark を使い、さらにファイル数、試行回数、ファイルサイズなどの設定を変更した。

## 4 アプリケーションの実行と輻射ウィンドウの評価

### 4.1 実験概要

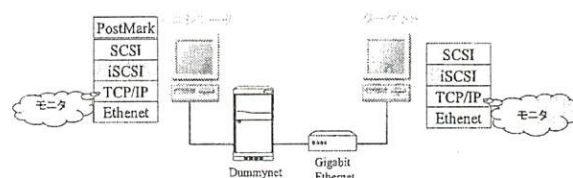


図 2: 実験システムの概要

iSCSI ストレージアクセスを行う initiator とストレージを提供する target の間を Gigabit Ethernet で接続し

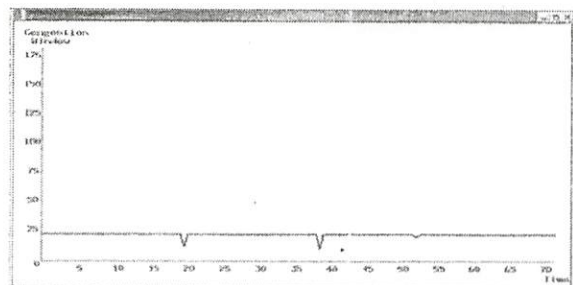


図3: 輻輳ウィンドウの変化 (広告ウィンドウ:40KB)

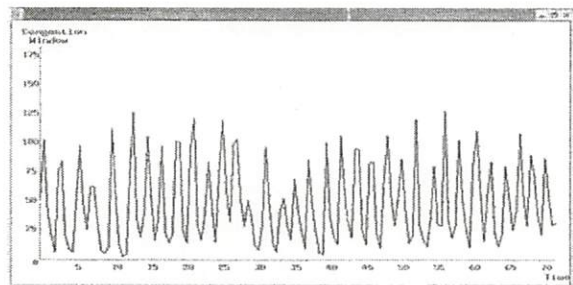


図4: 輻輳ウィンドウの変化 (広告ウィンドウ:8MB)

て実験システムを構築した。さらに、遠距離アクセスを想定して、遅延を入れるため、人工的な遅延装置である FreeBSD Dummynet を Ethernet の接続途中に挿入し、片道遅延時間を 16ms とした (図 2)。initiator と target には、OS が Linux 2.4.18-3, CPU が Intel Xeon 2.4GHz, Main Memory が 512MB DDR SDRAM, NIC が Intel Pro/1000XT Server Adapter on PCI-X (64bit, 100MHz), iSCSI は UNH-iSCSI ver.1.5.3 を用いた。この実験環境において iSCSI を起動し、PostMark の実行時に TCP 輻輳ウィンドウをモニタして、グラフを表示する。PostMark 設定状況はファイル数 15, 試行回数 15, ファイルのサイズ 9.54MB~38.15MB とした。この場合、PostMark の実行中、15 個のファイルが作成され、その 15 個のファイルが繰り返しアクセスされる。

#### 4.2 実行結果

PostMark には、read/write 速度やファイルの作成削除、読込追記などの実行結果が表示される。一方、モニタのグラフからは、輻輳ウィンドウの変化の様子がわかる。図 3, 4 は PostMark で transaction 処理が行われている時の輻輳ウィンドウの変化の様子である。

図 3 は、広告ウィンドウの設定を 40KB とした時の initiator 側の輻輳ウィンドウの様子である。この場合、輻輳ウィンドウは一定値以上に上がらず、輻輳ウィンドウが使いきれてない。一方図 4 は、広告ウィンドウを 8MB に設定した時のものである。この時、輻輳ウィンドウはある値で打ち切られてしまうことなく上昇し、ウィンドウが十分に使われていることが確認できた。

図 5 は、広告ウィンドウの設定を変化させた時の、read/write 速度、total/transaction 時間の変化の様子をグラフにしたものである。広告ウィンドウがおおよそ 120KB のところまでは、広告ウィンドウの増加に伴い、read/write 速度も上昇している。しかし、120KB をすぎるとあまり変化が見られなくなっている。これは、広告ウィンドウが十分に大きくなったため、広告ウィンドウの変化が輻輳ウィンドウに影響を与えなくなった結果

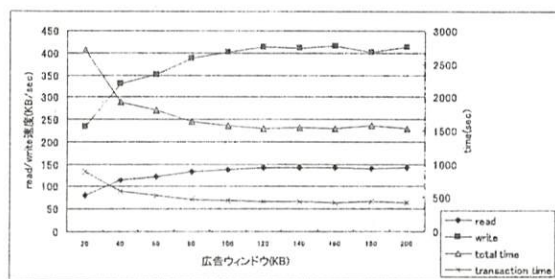


図5: read/write 速度と実行時間

であると考えられる。この時の輻輳ウィンドウの値は約 80 であったため、1 パケットの大きさが約 1.5KB であることから、 $1.5 \times 80 = 120\text{KB}$  が境目になったものと考えられる。

また、PostMark でファイルサイズの変更を行うと、それに合わせて輻輳ウィンドウの最大値も変化していることがわかった。ファイルサイズが小さいときは、輻輳ウィンドウの値は小さく、ファイルサイズが大きいときは、輻輳ウィンドウは大きな値となった。さらに、initiator と target での輻輳ウィンドウの変化の様子を比べると、initiator 側は輻輳ウィンドウが常に変化していくのに対し、target 側では輻輳ウィンドウの変化が見られるところ、小さい値で変化しないところがある。このことから、initiator 側で PostMark を実行すると、主に initiator から target へ大量のデータの送信が行われていることがわかる。target 側で輻輳ウィンドウに変化が表れている時は、同時に initiator 側でも輻輳ウィンドウが激しく変化しているため、PostMark の transaction 処理が行われ、read/write の処理が行われていると考えられる。

#### 5 まとめと今後の課題

iSCSI における PostMark の実行性能を測定し、その時の輻輳ウィンドウの時間変化を観測した。その結果、PostMark の設定の変更に合わせて、輻輳ウィンドウの値も変化していることがわかった。また、広告ウィンドウの値を制御することで、輻輳ウィンドウの値も制限でき、それによって実行性能にも影響が出ることが確認された。

今後は、さらに TCP パラメータの制御を行い、それによりアプリケーションの実行性能にどのような違いが現れるのか解明したい。

#### 参考文献

- [1] 喜連川優, ストレージネットワークング, オーム社 出版局
- [2] 豊田真智子, 山口実靖, 小口正人: "高遅延ネットワーク環境における iSCSI リードアクセス時の TCP 輻輳ウィンドウ制御手法の性能評価" SACSIS2005, pp.443-450, 2005 年 5 月
- [3] PostMark  
<http://www.netapp.com/ftp/postmark-1.5.c>
- [4] 千島望, 豊田真智子, 山口実靖, 小口正人: "iSCSI における TCP パラメータとアプリケーション実行性能の相関関係評価" 第 68 回情報処理学会全国大会, 5J-6, 2006 年 3 月

# ワイヤレス LAN 通信における TCP 輻輳ウィンドウ制御の動作解析

松井 愛子 (指導教員：小口 正人)

## 1 はじめに

近年、無線通信に対する需要がますます高まり、更なるスループットの増大が切望されている。しかし無線通信は有線通信と比べ、帯域が狭いだけでなく、通信効率が著しく低い。その原因として、ワイヤレス LAN においてはノイズ等によるエラー発生頻度が高いことに加えて、有線環境用に開発された TCP プロトコルが無線環境で効率よく機能していない可能性が考えられる。これまで有線環境においては、通信時の TCP プロトコルの振舞を解析し、TCP パラメータを制御することによって通信効率を上げる研究が行われてきた [1]。そこで本研究では、無線通信においてその手法を応用する方法を検討する。現在までに、無線通信における TCP プロトコルの研究は盛んに行われてきた [2]。しかしその多くは、スループットやレスポンスタイムの測定など、システム外部からの観察に留まっている。また無線環境に特化した TCP を構築する試みも見られる。それに対し本研究では、既存 OS のカーネル内部の TCP 実装で管理されている TCP パラメータの値を可視化し、そのデータを用いて議論を進める。

## 2 TCP パラメータの可視化の実現

本研究では無線通信時の TCP パラメータを可視化する。そして、同一条件において、有線通信時・無線通信時の TCP パラメータの振舞を比較・解析することで無線独自の問題点を発見し、無線環境で効率よく通信できるよう TCP パラメータを制御する手法を検討する。まず、カーネル内部の TCP ソースファイルにモニタ関数を挿入し、再コンパイルしてカーネルを再構築する。その結果、TCP による通信時にパラメータの値がカーネルのメモリ空間に保存され、カーネル外部からそれらの値が確認可能となる。また、モニタしたパラメータを X11 Window システムのライブラリ関数を利用し、リアルタイムにグラフ表示する。TCP パラメータの可視化手法の概念図を図 1 に示す。

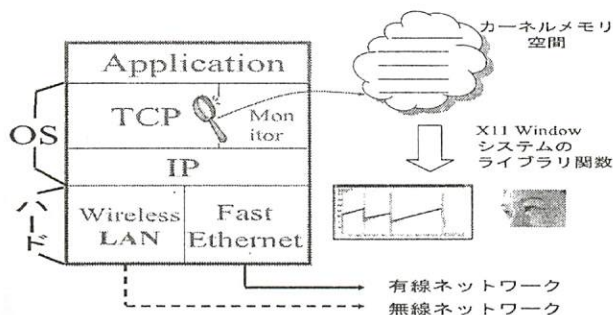


図 1: TCP パラメータの可視化手法の概念図

## 3 輻輳ウィンドウ

TCP では、輻輳制御において輻輳ウィンドウという概念を用いている。Linux OS の TCP は状態機械として実装されている。パケット送受信における処理は、その状態によって変化する。

輻輳ウィンドウとはネットワークの輻輳制御を目的としてデータ送信側が自主的に制限するためのパラメータで、受信側からの確認応答パケット (=ACK) 無しに連続送信できる最大のパケット数を表す TCP パラメータである。正常な通信時には確認応答を 1 つ受信するごとに 1 つずつ増加し、エラーが起こると急激に減少する。エラーにより輻輳ウィンドウが急激に減少した後、回復と判断されると再度正常な状態に遷移し増加を始める。単位時間あたりに受信する確認応答の個数によって差はあるものの、輻輳ウィンドウの減少時に比べると緩やかに増加していくため、TCP 輻輳ウィンドウ制御において、輻輳ウィンドウの時間変化は鋸型となる。また、Linux の TCP 実装では、通信中にひとたび設定された輻輳ウィンドウは、その値を使い切らない限りは変化せず、その間のスループットはほぼ一定の値で安定する、ということが分かっている。

## 4 基礎実験と考察

実験システムとして、2 台の PC を Fast Ethernet 有線 LAN と IEEE802.11b 無線 LAN で接続したものをを用いた。各マシンのスペックは、CPU が Pentium III 800MHz、メインメモリが 640MB、OS が Linux 2.4.18-3 である。データ送信、スループットの測定には netperf を用いた [3]。スループットは有線 LAN 通信時には平均 93~94Mbps/sec、無線 LAN 通信時には平均 3~4Mbps/sec 程度の値が出ている。有線 LAN 通信時および無線 LAN 通信時のデータ送信側の輻輳ウィンドウの時間変化をそれぞれ図 2 と図 3 に示す。

図に示されている縦の点線は、送信側デバイスドライバのバッファが溢れることによる Local Congestion エラーを検出したことを表している。

実験結果より、有線 LAN では短い周期の鋸状の波形となり、無線 LAN では長い周期の鋸状の波形となることが分かった。これは無線 LAN の時間単位あたりの送信パケット数が有線 LAN と比較して少なく、従って受信 ACK 数も少ないためであると考えられる。この結果より、無線 LAN においては、ひとたび輻輳ウィンドウの値が減少してしまうと元の値に戻るまでに長い時間がかかり、その間通信効率が大幅に低くなることが分かる。従って Local Congestion エラーが起きない程度の高い値に輻輳ウィンドウを保つことが、無線環



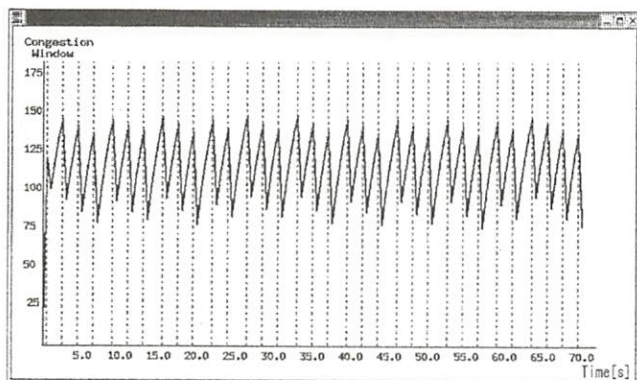


図 2: 有線 LAN 通信時の輻輳ウィンドウ個数の遷移

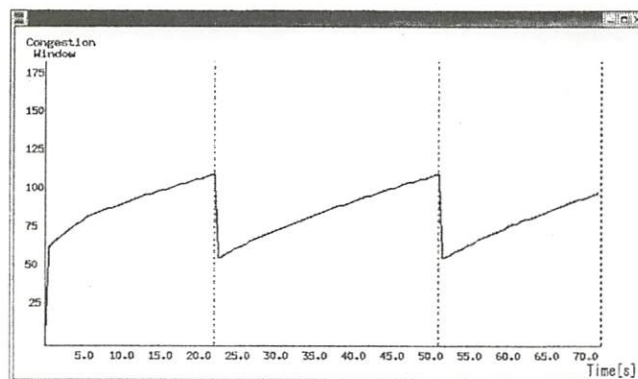


図 3: 無線 LAN 通信時の輻輳ウィンドウ個数の遷移

境で効率よく通信を行うためには重要であると考えられる。

## 5 輻輳ウィンドウ制御と通信効率の向上

### 5.1 輻輳ウィンドウ制御

無線 LAN 通信において広告ウィンドウのサイズを変更し、輻輳ウィンドウを制御する手法を検討した。広告ウィンドウとは受信側が送信側に指定するウィンドウサイズであり、輻輳ウィンドウと広告ウィンドウの小さい方が ACK 無しに連続送信できる最大パケット数となる。従って、広告ウィンドウが十分に大きい場合には、輻輳ウィンドウは広告ウィンドウに影響されることなく、そのネットワークにおける輻輳ウィンドウの最大値を取った後、Local Congestion エラーにより急激に値を下げる。広告ウィンドウが小さい場合には、最大値になる前に広告ウィンドウに成長を止められ、その値を保つ。

### 5.2 制御結果

広告ウィンドウサイズが 200K~8M バイトの時には、輻輳ウィンドウは図 3 のように鋸状の変化を見せるが、広告ウィンドウサイズが 200K バイト以下の際にはその広告ウィンドウサイズにける最大の値まで輻輳ウィンドウが成長した後、一定を保った。また、広告ウィンドウサイズが 200K バイトの時には図 4 の上の線 (1)、90K バイトの時には下の線 (2) というように、広告ウィンドウの値によって輻輳ウィンドウの最大値も変化した。

このように、広告ウィンドウサイズを Local Congestion が起きない範囲で最大にし、輻輳ウィンドウのサイズを出来る限り大きな値で一定に保つことにより、図 4(1) のようにエラーを出すことなく、効率よく通信することが出来るようになる。

## 6 まとめと今後の課題

本研究ではまず、有線・無線通信時における TCP パラメータ（主に輻輳ウィンドウ）の可視化を行った。同一条件において有線通信・無線通信を行い、TCP パラメータの振舞いを比較し、その差異を明らかにしようと試みた。その結果、無線 LAN 通信は有線 LAN 通信に比べ、ひとたび輻輳ウィ

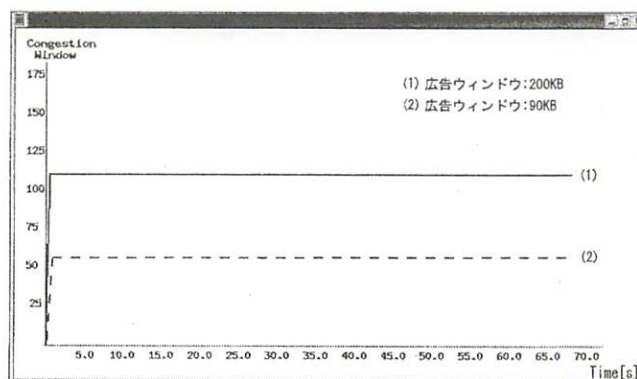


図 4: 広告ウィンドウサイズを抑えた場合

ンドウの値が減少すると再び元の値に戻るまでに長い時間がかかることが分かった。そこで、輻輳ウィンドウを Local Congestion エラーが出ない程度に高い値に一定に保つということを、広告ウィンドウの値を変える事で実現した。

今後は、更に異なる通信データ量・通信時間など様々な条件下におけるパケット通信を有線・無線 LAN 環境で行い、輻輳ウィンドウ以外の TCP パラメータについても比較・分析を重ね、無線 LAN 通信独自の問題点を発見し、解決策を検討して行きたい。

## 参考文献

- [1] 豊田 真智子, 山口 実靖, 小口 正人: "高遅延ネットワーク環境における iSCSI リードアクセス時の TCP 輻輳ウィンドウ制御手法の性能評価", 先進的計算基盤システムシンポジウム (SACSIS 2005) 論文集, pp.443-450, つくば, 2005 年 5 月
- [2] Ye Tian, Kai Xu, Nirwan Ansari: "TCP in Wireless Environments: problems and Solutions", IEEE Radio Communications, Vol.2, No.1, pp.s27-s32, March 2005
- [3] netperf: <http://www.netperf.org/>
- [4] 松井 愛子, 豊田 真智子, 小口 正人: "無線 LAN 通信における TCP パラメータの一解析", 第 68 回情報処理学会全国大会, 1S-9, 2006 年 3 月

# モバイルアドホックネットワークにおける ストリームデータ送受信の制御手法の検討

内田 智理 (指導教員: 小口 正人)

## 1 はじめに

近年インターネット上では、映像や音声などのリアルタイムのライブ放送を始めとした数多くのアプリケーションが実現されている。しかし、固定基盤を持たないモバイルアドホックネットワーク (以下 MANET) では、ストリームデータの取り扱いが難しいと言われる。MANET においてストリームデータの送受信環境が利用可能であれば、ノード間でライブ映像を送り合うなど種々のアプリケーションに用いることができ有益であると考えられる。そこで本研究では、MANET における P2P (Peer-to-Peer) 型アプリケーションのストリームデータ送受信に関する検討を行う。

## 2 MANET

インターネット上のシステムは多くの場合、クライアントサーバ型の通信形態であり、クライアントがサーバに接続することによってサービスを受けている。これに対し、P2P 型の通信はネットワークを構成するコンピュータが対等に処理を行っている。特定のサーバを用いることなく、全てのノードが他のノードに対してサービスを提供する機能を持つ。

P2P 型通信を用いて、モバイル端末同士が直接接続し、その場のみで一時的に構築されるネットワークを MANET という。MANET はアクセスポイントや基地局を必要としないので、インフラを持たない場所でネットワークを構築することが可能である。例えば、災害でインフラが機能しなくなったところや、混雑してサーバがダウンしてしまったネットワーク環境、基地局からの電波が圏外のエリアなどにおいてもコミュニケーションをとることができる。しかし、MANET では端末が常に移動するため、端末相互間のリンクが不確実であり、ネットワーク状態は頻繁に変化する。

## 3 MANET におけるストリームデータ

ストリーミングはデータを受信しながら同時に再生を行う通信方式であるため、配信されるデータ量が多く、途切れず連続的に流れ続けるのが特徴である。そのため、端末相互間のリンクが不確実な MANET 上ではストリームデータの扱いは難しいとされる。

しかし、MANET 上にストリームデータを流すことによって、以下のような便利なサービスの実現が期待できる。例えば、街を歩きながら、雰囲気の良い店に入りたいが店の中の様子が外からではよくわからないなどといった場合に、その場で MANET を構築し、ストリームデータで店内を覗き見るようなことの実現が期待できる。

## 4 MANET とストリームデータ送受信環境の構築

### 4.1 JXTA

本研究では、MANET のためのプラットフォームに JXTA を採用した。JXTA とは、Sun Microsystems 社が開発した P2P

型アプリケーションを容易に構築するための基盤ソフトウェアである [1][2]。ピア同士の自由な通信を保証し、通信経路を自動的に決定してくれるので、MANET の端末相互間のリンクに有効である。また、特定のコンピュータ言語に依存せず他の P2P ネットワークのサービスが利用可能であったり、プロトコルが単純なため、携帯電話や PDA などにも実装することが可能であるといった利点を持つ。

### 4.2 JXCube

JXTA 上でストリームデータを送受信するためには、JXCube という多機能コラボレーションツールを使うことが可能である [3][4]。JXCube を用いれば、JXTA 上で、インスタントメッセージ、ファイル交換・共有、音声・映像チャットなどの機能を利用できるようになる。

### 4.3 P2P プラットフォーム上の問題点

JXCube 上でストリームデータ送受信を行うためには、無線 LAN で接続された端末上に、Linux, Java, JXTA, JXCube の順にシステムを構築する。しかし、JXTA のようなオーバーレイネットワーク上で映像などのストリームデータを送受信すると、メッセージ単位での遅延がかなり発生すると考えられる。オーバーレイネットワークでは、プロトコルの違いを吸収するために、プロトコル変換が行われていて、その変換がアプリケーションレベルで行われているためタイムラグが発生するからである。そのため、JXCube のアプリケーション上でストリームデータを送受信する仕組みは性能面で問題がある。

## 5 研究方針

本研究では、MANET においてストリームデータの送受信が行える環境の構築手法を提案し、その上で利用できる P2P 型アプリケーションの仕組みを提案し検討する。

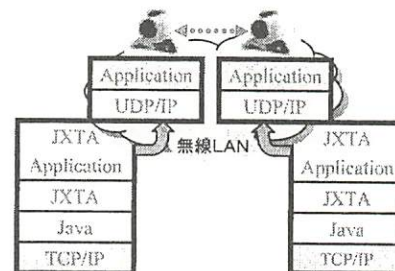


図1: 提案方式

本研究では図1に示すように、P2P でピアの発見・接続を行い、P2P のアプリケーションから UDP/IP 接続のストリームアプリケーションを呼び出す方法を提案する。更に、OS のネットワークプロトコル (UDP/IP) 上で直接動作するストリームアプリケーションと、JXTA を用いて P2P 環境から呼び出すストリームアプリケーションを比較して性能評価を行う。

## 6 実験環境

本実験は図2に示すようにPCを2台用いて行った。各PCともOSはLinux2.6.9-1.667 (FedoraCore3), Javaはj2sdk1.4.2-10, JXTAはバージョン2.3.3である。2台のPCはIEEE802.11g無線LANを用いて通信を行う。各PCにWebカメラを搭載し、そこで撮影された映像をリアルタイムストリームデータとして配信する。

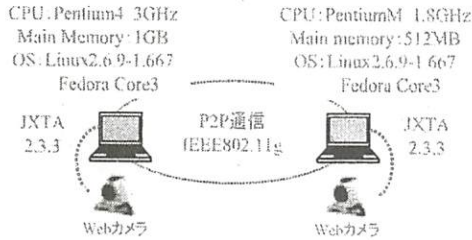


図2: 実験環境

## 7 性能評価

性能評価方法は、UDP/IP上のアプリケーションでストリームデータを送受信しているときのCPU使用率・スループットと、JXTAを用いてP2P環境から呼び出されたアプリケーションでストリームデータを送受信しているときのCPU使用率・スループットを測定して比較する。

測定した値をグラフに表したものは、2台の端末をそれぞれthink, mobileと名づけて表示する。

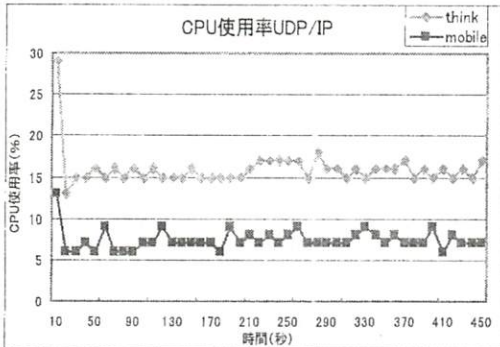


図3: CPU使用率UDP/IP

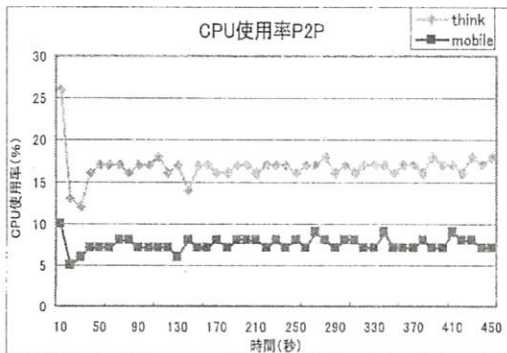


図4: CPU使用率P2P

図3のグラフはUDP/IP接続の場合のCPU使用率であり、図4のグラフはP2P環境から呼び出した場合のCPU使用率である。グラフを比べてみると、両端末ともにUDP/IP接続の場合とP2P環境の場合を比べて、ほとんど変化がない。P2P環境の方は、JXTAが動作し、そこからアプリケーションを呼び出す

オーバーヘッドがあるためCPUの負荷が高い可能性があるのだが、計測の結果CPU使用率はほとんど変わらないことがわかった。

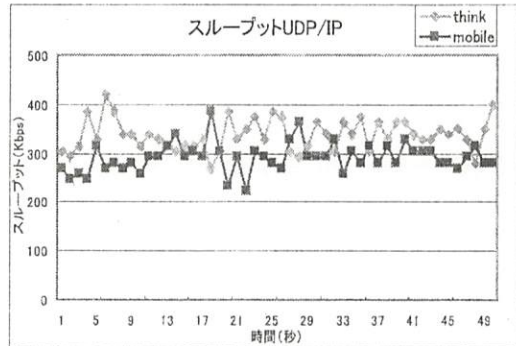


図5: スループットUDP/IP

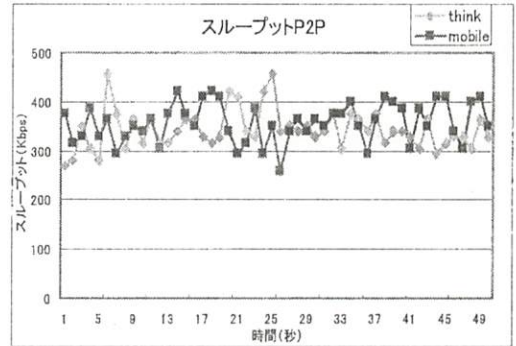


図6: スループットP2P

図5のグラフはUDP/IP接続の場合のスループットであり、図6のグラフはP2P環境を用いた場合のスループットである。スループットにも大きな変化はなく、性能的には変わらないことがわかる。

## 8 まとめ

UDP/IP上のストリームアプリケーションと、JXTAを用いてP2P環境から呼び出されたストリームアプリケーションの性能を評価した結果、CPU使用率、スループットはほとんど変化がなかった。よって、本研究で提案したJXTAを用いてP2P環境からストリームアプリケーションを呼び出す手法は有効であると言える。

## 参考文献

- [1] JXTA  
<http://www.jxta.org/>
- [2] P2P for Java/JXTA  
<http://www.wakhok.ac.jp/~maruyama/jxta/html/>
- [3] JXCube-JXTA eXtreme Cube  
<http://jxcube.jxta.org/>
- [4] JXCube  
<http://www.koyanagi-lab.org/yoshinaga/contents/jxta/jxcube.html>
- [5] 内田智理, 小口正人; "モバイルアドホックネットワークにおけるストリームデータ送受信に関する一検討", 第68回情報処理学会全国大会, 1R-10, 2006年3月



率はそれぞれ図7、図8、図9である。

まずメモリ消費量のグラフを見ると、パス2あたりのメモリ消費量が他のパスに比べて非常に大きくなっていることがわかる。特に最小サポート値を0.1%とした時にはメモリ消費量が桁違いに多くなり、このような性質を持つアプリケーションを実行する場合、PCクラスタの各ノードに十分なメモリが搭載されているか注意する必要があると考えられる。

また通信ネットワークのスループットを見ると、候補アイテム集合が多いパスでは高い値となっている。これは10台のノード間でデータをやり取りする必要があるためであろう。ただしGigabit Ethernetを用いているため、バンド幅から見てスループットはまだ余裕があると言える。

一方CPU使用率は、これらのパスにおいてトータルの使用率が100%に達している。非常に計算負荷の高いアプリケーションであり、本研究の環境ではトランザクションデータを読み出すI/O負荷よりもCPUの計算負荷が全体の実行のボトルネックになっていることがわかった。

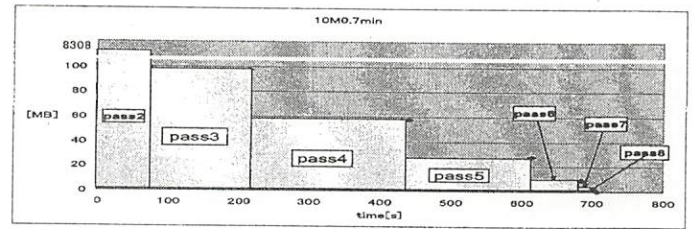


図7: メモリ使用量

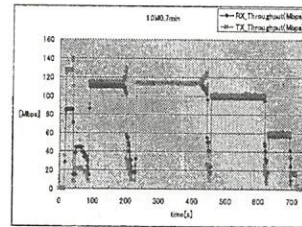


図8: スループット

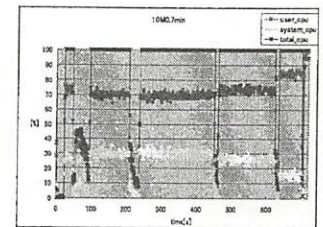


図9: CPU使用率

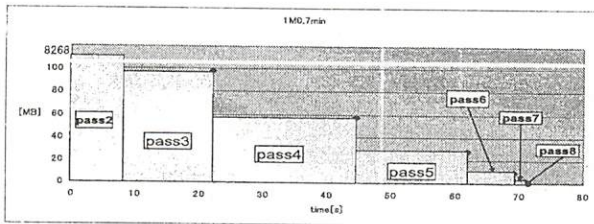


図1: メモリ使用量

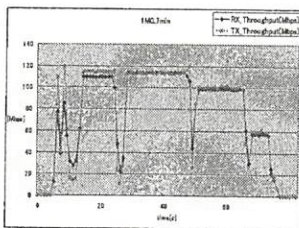


図2: スループット

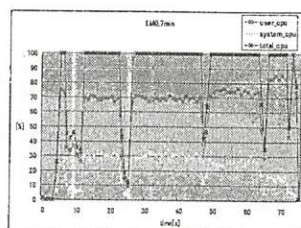


図3: CPU使用率

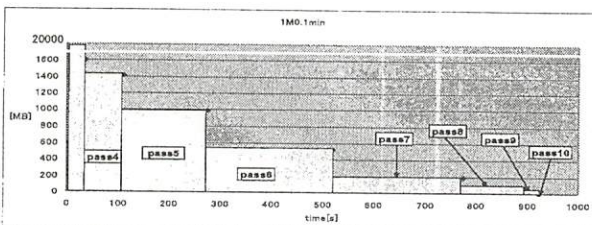


図4: メモリ使用量

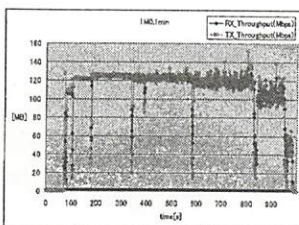


図5: スループット

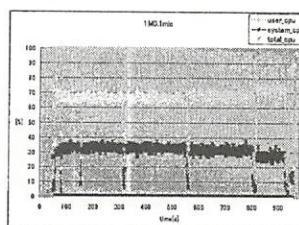


図6: CPU使用率

## 6 まとめ

本研究ではネットワークコンピューティング環境を用いて並列相関関係抽出の実行特性解析を行った。データマイニングは既にビジネス分野で広く用いられるようになっており、現在研究が行われているのは未だ実用化の見通しが不明確な領域のテーマと、本研究のように特異な振舞を見せる処理が難しい性質のアプリケーションである。後者の特性を持つ場合、実行環境のリソースを戦略的に配分することが重要となり、そのためにはデータ解析を行うマイニングアプリケーションの実行特性をさらに解析していくことが必要である。

例えば今回のケースでは、プログラム実行時にメモリ消費量が桁違いに多くなる部分があるものの、10ノードのPCクラスタを用いれば必要なメモリ量は供給できるが、一方でCPU負荷が非常に高い状態が続いているため、もっと多数台のノードを投入して計算処理を分散すれば、実行時間が短くなるであろうことがわかる。このようにデータマイニングアプリケーションの実行特性を解析し、これにより得られた情報をネットワークコンピューティング環境の構成にフィードバックする手法の確立が重要と考えられる。

## 参考文献

- [1] 福田剛他：データマイニング，共立出版
- [2] 石川博：次世代データベースとデータマイニング
- [3] R. Agrawal and R. Srikant: "Fast algorithms for mining association rules," Proc. VLDB1994, pp.487-499, Sept. 1994.
- [4] 小口正人、喜連川優："ATM結合PCクラスタにおける動的リモートメモリ利用方式を用いた並列データマイニングの実行"、電子情報通信学会論文誌、Vol.J84-D-I, No.9, pp.1336-1349, 2001年9月。
- [5] IBM Almaden Research Center, Intelligent Information Systems  
<http://www.almaden.ibm.com/software/disciplines/iis/>

# 多次元尺度法とクラスター分析

久保 亜沙美 (指導教員: 吉田 裕亮)

## 1 はじめに

クラスター分析とは、データ間に距離を定義し、距離が近いデータ同士をグループ(クラスター)にまとめる作業のことを言う。その起源は、生物学の分野での生物分類学であると考えられる。複数の特性によって決定された個体間の類似性の指標をもとに、個体の集合をいくつかのグループに分類してきた。

本研究では、パターン認識の一つとしてクラスター分析を行った。結果をデンドログラムに表すことにより、個体間の距離を一目で把握できるようにすること、更には、多次元に拡張し地図上にデータを復元することを目的とした。

## 2 クラスタリングのアルゴリズム

クラスター分析の手法は数多く知られているが、本研究では、まず1次元的表现として、階層のかつ凝集的手法でアルゴリズムを考える。

Step1) 与えられた  $n$  個の分類対象を  $n$  個のクラスターとして1から  $n$  までの番号を付与する。

Step2) 対象間の類似度行列  $D$  を算出する。

Step3) 行列  $D$  の要素の中からもっとも近いクラスターの組  $C_i$  と  $C_j$  とを併合して  $C_t$  とし、クラスター数を  $n$  から  $n-1$  に更新する。

Step4) クラスター  $C_t$  とそれ以外のすべてのクラスター  $C_k$  との間の類似度を計算し  $D$  を更新する。

Step5) 以上をクラスター数が2になるまで反復する。

Step6) 結果をデンドログラムに表す。

データ間の距離の測定法には次のようなもの等がある。

### I. 最短距離法 (single)

$$d_{tk} = \min(d_{ik}, d_{jk})$$

### II. 加重平均法 (average)

$$d_{tk} = \frac{1}{2}d_{ik} + \frac{1}{2}d_{jk}$$

## 3 1次元クラスタリングの例

Atlanta	0									
Chicago	587	0								
Denver	1212	920	0							
Houston	701	940	879	0						
Los Angeles	1936	1745	831	1374	0					
Miami	604	1188	1726	968	2339	0				
New York	748	713	1631	1420	2451	1092	0			
San Francisco	2139	1858	949	1645	347	2594	257	0		
Seattle	2182	1737	1021	1891	959	2734	2408	678	0	
Washington	543	597	1494	1220	2300	923	205	2442	2329	0

表1. 米国10都市間の飛行距離

実験データとして、表1のようなデータを用意した。これは、クラスター数を10個とした、米国10都市間

の飛行距離情報である。あらかじめ、各2都市間の距離データを行列として与えておく。

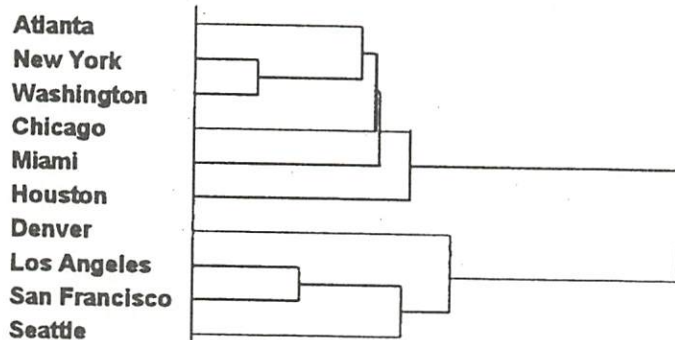


図1. 最短距離法

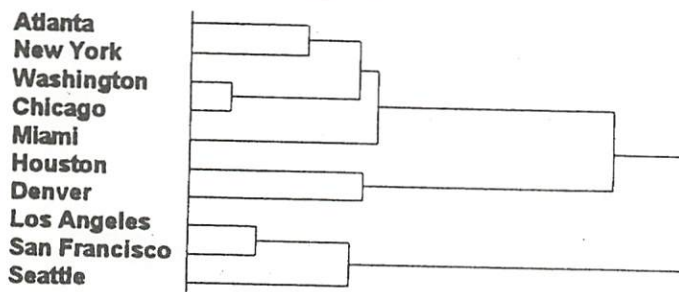


図2. 加重平均法

図1は最短距離法、図2は加重平均法を用いて作ったデンドログラムである。図1は大きく東海岸、西海岸に分かれているのに対し、図2は米国の東部、西部、中央部に分類されていることが分かる。距離の測定法の違いにより、デンドログラムの形が若干異なってくるということ、またクラスタリングの目的に合わせて距離の測定法を使い分ける必要がある、と言える。

## 4 多次元尺度法

1次元のクラスタリングでは、距離軸についてのみ表現した。同じ距離行列に基づいて、今度はデータを多次元に付置すること、ここでは2次元で地図上にデータを復元することを考える。この手法を一般に多次元尺度法という。

アルゴリズム

Step1) 距離行列  $D$  の2乗距離行列  $D^{(2)}$  を作る。

Step2) Young-Householder 変換をする。原点を

$$x_1 + x_2 + \dots + x_n = 0$$

となるように移動し、内積行列

$$P = [(x_i, x_j)]_{i,j}$$

を構成する。すなわち、



# 離散2値データからの最適決定木

井上 由美子 (指導教員: 吉田 裕亮)

## 1 はじめに

決定木は、統計的決定理論、人工知能、機械学習、データマイニング、などの分野で予測モデル構築、意思決定分析・最適化、分類問題の解決、概念・知識の記述、ルールの抽出・生成などに利用されている。意思決定理論の分野においては、意思決定と不確定条件によって分岐を繰り返す多重決定問題モデルを示したもので、プロセスと予測される結果を示す。決定理論(決定分析)の主要分析ツールである。知識工学や人工知能の分野においては、命題判定の繰り返しによる知識表現方法であり、エキスパートシステム構築の際などに使われる。データマイニングの分野においては与えられたサンプルデータ群をその属性変数の値から分類し、その繰り返しによってデータ全体を樹形モデルで表現する方法をいう。

本研究では、離散2値データを用いてある決定木を再現し、ノイズの増加とともに変化する決定木を調べるといふ実験を行うことにした。また、実験から得られた数値データに対して考察する。

## 2 決定木

入力の性質について、逐次的にテストを行うことで、最終的にその属するクラスを推測する。どのようなテストをどのような順で実施するのか、最初のテストを根のノードとし、そのテストの結果に応じて分岐した枝の先に次のテストに対応するノードをつける、という形で構成される木によって表現される。

決定木には他に2つの呼び名がある。

### 回帰木

分類に用いられるのではなく、実数値をとる関数の近似

### 分類木

従属変数が分類変数の場合

## 3 決定木のアルゴリズム

ステップ1 木の根となるノードに、全学習データに対応づける

### ステップ2

if ノードに対応づけられたデータが停止条件を満たす  
then そのノードに対応づけられたデータが属するクラスについて多数決をとり、そのノードのクラスを決める  
else ノードに対応づけられたデータに対して適用可能なテストを選択して、その結果によって子ノードを生成する。さらに、ノードに対応づけられたデータをテストで分類し、それぞれの子ノードに対応づける

ステップ3 すべてのノードで木の生成が停止するまで、ステップ2を再帰的に繰り返す

### ステップ2の停止条件

• 全てのデータが同じクラスに属する

例: 天気というテストについて、そのノードに対応づけられたデータのクラスが、全て適、または、全て不適となる

• 全てのデータが同じ結果を与える

例: 天気というテストについて、そのノードに対応づけられたデータの結果が、全て晴れ、または、全て雨となる

### 学習データの形式

学習データの形式は以下のようなレコードである。

$$(\mathbf{x}, \mathbf{y}) = (x_1, x_2, \dots, x_k, y)$$

従属変数  $y$  は分類や法則化を行う対象である。 $x_1, \dots, x_k$  などは変数であり、現象説明する。

## 4 テストの選択基準

分割前と分割後で、データ内のクラスのばらつきがどれだけ減少したかを定量化するテストの選択基準によって枝刈りをする。

$D_0$  ノードに対応づけられたデータ

$D_1, \dots, D_L$  テストの結果によるデータの背反な分割

$p(c; D_i)$   $D_i$ 内のクラス  $c$  の相対頻度

$D_i$  内のエントロピー

$$H(D_i) = - \sum_{c=1}^K p(c; D_i) \log_2 p(c; D_i)$$

テストによる情報量の増加

$$G(D_0, T) = H(D_0) - \sum_{i=1}^L \frac{|D_i|}{|D_0|} H(D_i)$$

分割の細かさを表現するもう一つの情報量

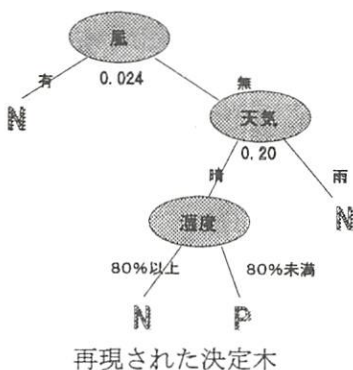
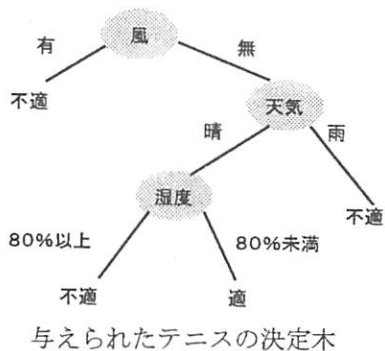
$$S(D_0, T) = - \sum_{i=1}^L \frac{|D_i|}{|D_0|} \log_2 \frac{|D_i|}{|D_0|}$$

以上より、正規化した  $G(D_0, T)/S(D_0, T)$  をテストの選択基準として用いた。



### 5 数値実験 1

以下に図示した、テニスの決定木を再現できると予想される離散2値データを用意し、実際に決定木を描き、テニスの決定木と比較する。



よって、再現されることが認められる。

### 6 数値実験 2

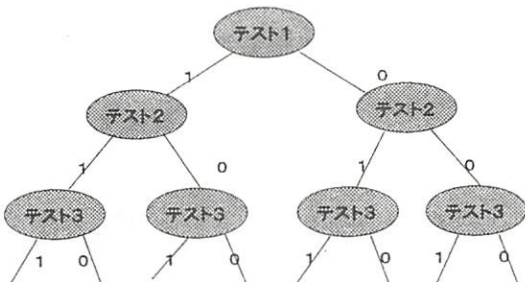
ベルヌーイ分布に従う乱数 1000 個を、0 と 1 の反転に用いてノイズを入れる。このような方法で 5%、10%、...、50%ノイズを含むデータを 10 個ずつ作る。このようにして作成したデータに対して決定木を各々描く。

不適な例

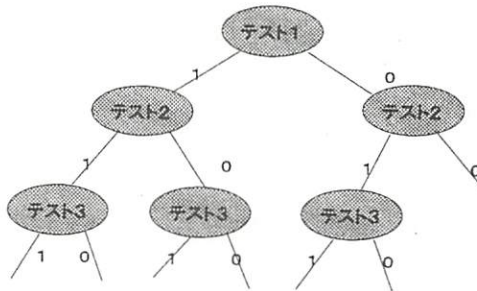
1. ノードが 1 個のみ残る。



2. ノードが全て残る。



3. 枝刈りの回数が少ない。

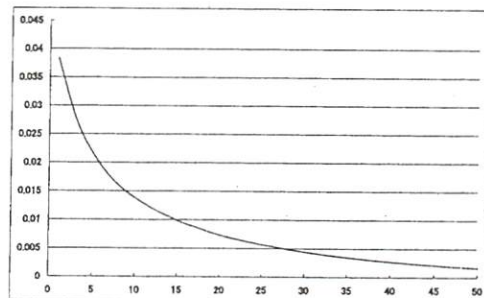


4. クラスが、ノイズを加える前後で変わる。

以上のような例がみられた。これらは 1000 個のデータの特徴を抽出できていない。

### 7 数値実験 3

数値実験 2 と同様な方法で、ノイズの量を 1% から 50% まで 1% きざみで変化させたデータをそれぞれ 100 個ずつ用意する。そして正規化した関数によって求められた値 100 個の値の平均値を計算する。下のグラフは、縦軸が平均値、横軸がノイズの量 (%) である。



正規化した関数によって求められた値 100 個の平均値 (Test1)

上のグラフは一例である。他の全てのノードのグラフも描いた。結果、全てのノードにおいて、ノイズの増加とともに平均値の増減はみられるものの、全てのノードで減少することがわかった。

### 8 今後の課題

本研究ではエントロピーに基づく評価関数を用いた。他に、データのばらつきを評価する指標に Gini 係数があり、これに基づく評価関数も考案されている。この評価関数を用いて今回と同様な手法を開発したい。さらに、これ以外の評価関数の導入も考えてみたい。そして、状況にあった評価関数を選べるような手法を考案したい。

# EM アルゴリズムによる混合コーシ分布問題

石橋 彩乃 (指導教官: 吉田 裕亮)

## 1 はじめに

実世界から得られるデータは、常に完全ではなく、ある変量に対しデータが得られていないような欠測値を含んでいる場合も多い。比較的単純な統計モデルにおいても、不完全データの存在により解析が困難になってしまう。本研究で扱う混合分布問題も、どの分布から生成されているかという情報が欠測した不完全データと考えることができる。このようなデータの解析には欠測値を考慮に入れた手法が必要となる。その代表的な手法として EM アルゴリズムがあげられる。

EM アルゴリズムは一般に、混合正規分布のパラメータの最尤推定で広く用いられる。しかし、密度関数がわかっているような混合分布にも理論的には応用できる。本研究では、混合コーシ分布への応用を考えた。しかし、Mstep における尤度方程式は、解析的に解けないため、最尤推定値を求めることは困難である。そこで、コーシ分布のパラメータの最尤推定に代わる方法を提案し、この手法を Mstep に用いた改良 EM アルゴリズムについての考察を行った。

## 2 混合分布問題

$K$  個のクラスからなる混合分布とは、 $k$  番目のクラスの確率密度関数を  $f_k(x|\theta_k)$  ( $k = 1, \dots, K$ )、混合比を  $p_k$  ( $k = 1, \dots, K$ ) とするとき、確率密度関数が

$$f(x|\theta, p) = \sum_{k=1}^K p_k f_k(x|\theta_k), \quad \sum_{k=1}^K p_k = 1$$

と与えられるような分布である。

混合分布問題とは、ここからの標本  $\{x_1, \dots, x_n\}$  が与えられたとき、パラメータ  $\{\theta_1, \dots, \theta_K\}$  と混合比  $\{p_1, \dots, p_K\}$  及び分布数  $K$  を推定することである。ここでそれぞれのデータが属しているクラスを表す潜在変量として  $z_i = (z_{i1}, \dots, z_{iK})$  を導入し、 $x_i$  がクラス  $k$  からの標本であるとき、 $z_{ik}$  を 1、そうでなければ 0 とする。すなわち理想的な完全データを  $(x_i, z_i)$  とし、観測データ  $x_i$  を  $z_i$  が欠測した不完全データと考える。

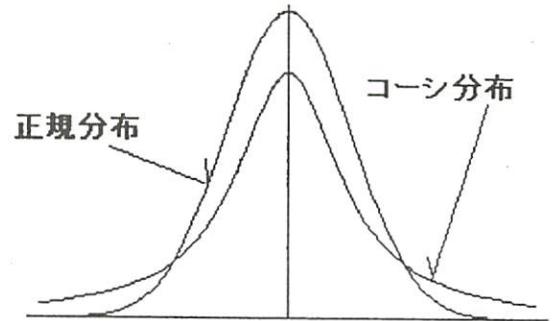
ここでは、分布数は既知として考え、パラメータ  $\theta_k$  と混合比  $p_k$  を推定する方法として EM アルゴリズムを採用する。

## 3 EM アルゴリズム

EM アルゴリズムとは、不完全データをいったん扱いやすい擬似的完全データに置き換え (Expectation Step)、この擬似完全データによりパラメータの最尤推定値を求め (Maximization Step)、さらに得られたパラメータの推定値から再び完全データを作り直し、それよりまたパラメータの推定値を求め直すという手続きを反復するものであるといえる。

本研究では、各分布がコーシ分布の場合を考える。コーシ分布は、一見、正規分布に似ているが、その性質はまったく異なり、期待値も分散も存在しない。

## コーシ分布と正規分布



一般に、密度関数は

$$f(x) = \frac{1}{\pi} \cdot \frac{c}{(x-m)^2 + c^2}$$

と与えられる。

このパラメータ  $m, c$  の最尤推定値を得ることは非常に困難である。そこでコーシ分布が中央値と四分位によって特徴付けられることに着目して  $m, c$  の最尤推定の代わりとする手法を提案する。

## 混合コーシ分布における EM アルゴリズム

$N$  個の標本  $\{x_1, \dots, x_N\}$  が与えられ、混合分布が  $K$  個の分布からなるとすると、各データの欠測値  $z_{ik}$  とそれぞれの密度関数のパラメータ  $\{(m^{(1)}, c^{(1)}), \dots, (m^{(K)}, c^{(K)})\}$  と混合比  $\{p_1, \dots, p_K\}$  は次のように推定される。

初期設定として  $m^{(k)}, c^{(k)}, p_k$  に適当な初期値を与え、各クラス  $k$  の確率密度関数を  $f_k$  とする。

### (1) Estep

各データの欠測値  $z_{ik}$  の推定

$$z_{ik} = \frac{p_k f_k(x_i)}{\sum_{k=1}^K p_k f_k(x_i)}$$

### (2) Mstep

各パラメータ  $m^{(k)}, c^{(k)}, p_k$  の推定

$m^{(k)}$ : 分布  $k$  の中央値

$c^{(k)}$ : 分布  $k$  の第 1 四分位と第 3 四分位の

中央値からの距離の平均

$$p_k = \frac{1}{n} \sum_{i=1}^N z_{ik}$$

以上より、 $f_k$  の値が更新される。

(1)(2) を繰り返し行うことにより、各パラメータの最尤推定の代わりとする。

本研究では、さらに、中央値、四分位をより正確に推定するためにブートストラップ法を適用することにした。

#### 4 ブートストラップ法

ブートストラップ法とは、経験分布に基づいて、大量にリサンプリングした標本から、ある統計量の分布を推定し、推定値の誤差評価したり、区間推定を行う手法の一つで、1979年にEfronにより提唱された。

##### 本実験におけるブートストラップ法の手順

1. 分布ごとに標本  $z_{1k}, \dots, z_{Nk}$  からランダムに重複も含めてリサンプリングする。(リサンプルの大きさは元の標本の大きさと同じとする。)

2. リサンプルから中央値、四分位を計算し、そのときの  $x_i$  の値を  $(T_m^{(n)}, T_c^{(n)})$  とおく。

3. このプロセスを  $N$  回繰り返すと、それぞれ  $N$  個の推定値  $(T_m^{(1)}, T_c^{(1)}), \dots, (T_m^{(N)}, T_c^{(N)})$  が得られる。その平均をとり、 $m^{(k)}, c^{(k)}$  の推定値とする。

#### 5 実験

分布数や、混合比、パラメータの値を変えた以下のようなシミュレーションデータに適用した。実際に乱数データによるヒストグラムと、得られた推定値との比較により、分離精度を確かめた。

真値 (分布数 2, データ数 2000)

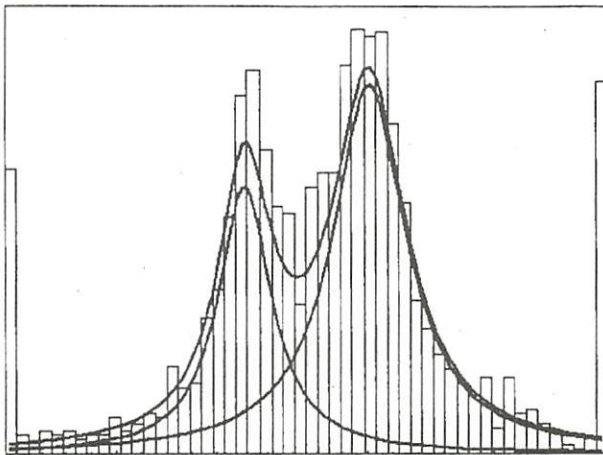
$$\text{分布 1 : } m^{(1)} = 0.0, \quad c^{(1)} = 1.0, \quad p_1 = 0.3$$

$$\text{分布 2 : } m^{(2)} = 5.0, \quad c^{(2)} = 2.0, \quad p_2 = 0.7$$

推定結果

$$m^{(1)} = 0.127, \quad c^{(1)} = 1.296, \quad p_1 = 0.344$$

$$m^{(2)} = 5.076, \quad c^{(2)} = 1.801, \quad p_2 = 0.656$$



真値 (分布数 3, データ数 5000)

$$\text{分布 1 : } m^{(1)} = 0.0, \quad c^{(1)} = 1.0, \quad p_1 = 0.2$$

$$\text{分布 2 : } m^{(2)} = 1.0, \quad c^{(2)} = 2.0, \quad p_2 = 0.6$$

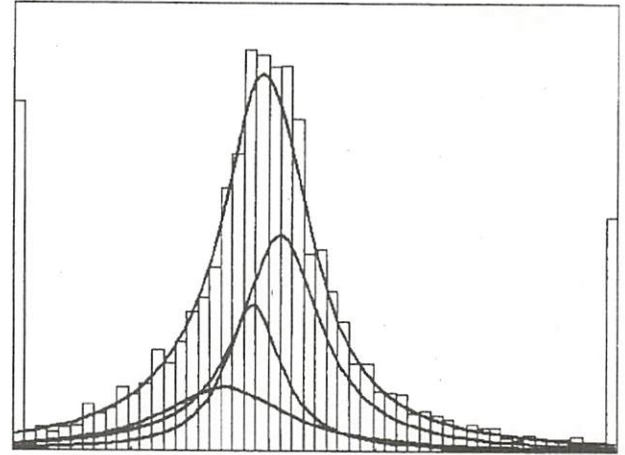
$$\text{分布 3 : } m^{(3)} = -2.0, \quad c^{(3)} = 3.0, \quad p_3 = 0.2$$

推定結果

$$m^{(1)} = -0.012, \quad c^{(1)} = 1.386, \quad p_1 = 0.245$$

$$m^{(2)} = 1.100, \quad c^{(2)} = 1.976, \quad p_2 = 0.517$$

$$m^{(3)} = -1.177, \quad c^{(3)} = 3.121, \quad p_3 = 0.238$$



以上のような結果より、多くの混合コーシー分布に対し、かなり良い分解能力を持つことがわかった。ただし、混合している分布の  $m, c$  の値が近いときや、混合比に極端な差があるときは、その混合分布が1つの分布に近くなるために正常な分離ができない場合も観測された。

#### 6 まとめと今後の課題

今回の実験により、混合コーシー分布は Mstep において中央値と四分位により、最尤推定の代わりに EM のようなアルゴリズムを適用して分離できることがわかった。ただ、正常に分離できない場合もあるので混合分布を標準化し、その差をカルバックライブラー情報量 (KLI) を用いて測り、一つの分布にどの程度近いのかの指標としたい。

また、本研究では、分布数が既知のものとして取り扱ったが、AIC 等により、観測データに最適な分布数の推定を行いたい。

さらに、コーシー分布は正規分布に比べ、裾が重いため、収束しにくいことが、分離のあいまいさにつながっていると思われる。この点についても検証していく必要がある。

#### 参考文献

[1] 中澤瑞穂: 二次元混合分布の分離問題, 1996 年度お茶大情報科学科卒業論文

[2] 麻生英樹, 津田宏治, 村田昇: パターン認識と学習の統計学, 岩波書店

[3] 統計学入門, 東京大学出版会

[4] 添田喬, 太田光雄, 大松繁: 数理統計の基礎と応用, 日新出版