

## 全ラベル配置のための領域決定問題

傳 保 能 幸<sup>†</sup> 今 井 桂 子<sup>††</sup>

### 概要

本研究では、全ラベル配置のための領域決定問題を定義し、この領域決定問題を解くためのアルゴリズムを提案する。近年の情報技術の発展により、計算機や携帯端末でデジタル地図を利用する機会が増加している。それに伴ない、地図の各要素に対して文字情報(ラベル)を計算機で自動注記する問題が盛んに研究されてきた。しかし、この種の問題の多くが NP 困難であることが示されており、ユーザの指定するラベルサイズで全てのラベルを配置することは一般に難しい。また、地図を利用する際に広域図だけでなく、詳細図が必要なことも多い。そこで、拡大表示する際にユーザが要求する点を中心に、ユーザの指定したラベルサイズで、重ならずラベル配置することを考える。よって本研究では、拡大表示する領域を決定するために、全ラベル配置のための領域決定問題を定義し、そのアルゴリズムを提案し、実装する。

## Area Decision Problem for Placing All Labels

TAKAYUKI DEMBO<sup>†</sup> and KEIKO IMAI<sup>††</sup>

### Abstract

The label placement problem is one of the most important problem in geographic information systems, cartography, graph drawing, interface design. In general maximizing the number of labels and maximizing the size of the labels are NP-hard. When we search information for a site in a digital map, we first get the position of the site in a wide-area map and then we find detailed information near the site in a closeup map. In the closeup map, all labels in the area should be placed in suitable size. Therefore, we have to decide the area in the wide-area map which we scale up as the closeup map. We name this an area decision problem for placing all labels. In this paper, we propose an algorithm for this problem and also show experimental results.

### 1. はじめに

地図の多くは、交差点や駅などの点や道路や川などの辺、建物や国などの領域に対して、文字情報(ラベル)が配置されている。従来、文字情報を注記する作業は、職人によって行なわれてきた。しかし、近年の情報技術や地理情報システムの発達に伴ない、地図をデジタルデータとして管理、加工する機会が増加している。そのため、計算機上で地図やグラフの各要素に対して、ラベルを自動注記する問題が研究されている。この問題のことをラベル配置問題という。

さらに、管理、加工された地図データを計算機や携帯

端末上で利用する機会も、近年増加傾向にある。これらの地図を利用する人々の多くは、主に現在地や目的地、経路などの探索ツールとして地図を利用している。そのため、全体を表示している地図だけでなく、ユーザの必要に応じて、現在地や目的地の周辺の詳細な情報を得るための地図も必要な場合がある。例えば、店の位置を探索する場合、都心では店が密集しているため、広域図では表示していないか、ラベル同士が交差しているため、店の細かい場所の特定がしにくい。このような場合、その周辺の詳細図を表示すれば、細かい道の把握やどの建物に店が入っているのかなどの詳細な情報を得ることができる。

この詳細な情報を得るための地図では、ユーザが各点周辺の情報を要求した場合に表示するため、その点周辺の全てのラベルを重ならず配置する必要がある。本研究では、ディスプレイに表示された広域図における点を、ユーザの必要に応じて指定し、その点周辺の詳細図を表示することを考える。ここでいう広域図と

<sup>†</sup> 中央大学大学院 理工学研究科 情報工学専攻

Information and System Engineering Course, Graduate School of Science and Engineering, CHUO University

<sup>††</sup> 中央大学 理工学部 情報工学科

Department of Information and System Engineering, Faculty of Science and Engineering, CHUO University

は、入力された全ての点を表示するような、広範囲を表示する地図であり、広域図の一部の領域を拡大表示した地図を詳細図と呼ぶ。本研究では、縦  $h$ 、横  $w$  の長方形の決まった大きさの詳細図を表示することにする。さらに、各点のラベルの文字数を  $s_{p_i}$  としたとき、縦  $\sigma$ 、横  $\sigma \times s_{p_i}$  の長方形ラベルを用いる (図 1 (a))。このとき、ラベルの高さ  $\sigma$  をラベルサイズと呼ぶ。

既存の地図システムは、一度ラベル配置問題を解き、縮尺にかかわらず、ラベルをその解を用いて配置している。そのため、詳細図でもラベルが重複、あるいは配置できないことがある。しかし、詳細図はユーザの要求に応じて表示しているため、このような問題が起こることは望ましくない。

大きさの固定された詳細図を表示する方法の 1 つとして、全ての点のラベルを、できるだけ大きなラベルサイズで、重ならず配置し、その一部を拡大表示する方法がある。しかし、この場合、点が密集している点周辺の詳細図と、そうでない点周辺の詳細図で、表示されるラベルの数に大きな差が生じてしまう。また、もう 1 つの方法として、先に拡大表示する領域を決定し、拡大表示する方法もある。この場合、もし領域を大きく取りすぎれば、ラベルが重複、あるいは配置できないことがある。逆に、領域を小さく取りすぎれば、表示されるラベルが極端に少なくなり、表示される情報が減少してしまう。

以上より、詳細図に表示するラベルを、指定したラベルサイズで、重ならず配置するためには、拡大表示する領域を巧く決定する必要がある。

よって本研究では、そのような領域を巧く決定する問題を、全ラベル配置のための領域決定問題として定義し、この問題に対するアルゴリズムを提案する。さらに、計算機実験を行ない、本研究で提案したアルゴリズムを用いると、高速に詳細図に表示する領域を決定できることを示し、拡大する位置によって、領域の大きさが柔軟に変化することを示す。

本論文の構成は以下の通りである。まず 2 節では、ラベル配置問題について述べ、本研究で扱う全ラベル配置のための領域決定問題を定義する。3 節では、本研究で用いた既存のアルゴリズムと提案アルゴリズムについて述べる。4 節では、ポップアップシステムを用いて計算機実験を行なった結果を示す。最後に、5 節で結論と考察を述べる。

## 2. ラベル配置問題

平面上の地図やグラフの点や辺、領域などの各要素に対して、文字情報 (ラベル) を配置する問題をラベ

ル配置問題という。中でも、点に対してラベルを配置する問題を点ラベル配置問題という。本研究では、点ラベル配置問題について考える。また、配置されるラベルは以下の 2 つの規則を満たすものとする。

- ラベルは、他の点やラベルと重なってはならない。
- ラベルは、指し示す点が一意に定まるように配置されなければならない。

点ラベル配置問題のラベルの配置モデルとして、Fixed-position Models と Slider Models がある。Fixed-position Models は、ラベルの配置候補位置を離散的に定義したモデルである。例として、ラベル配置候補位置が 4ヶ所ある 4-position model がある (図 1 (a))。Fixed-position Models は、ラベルの配置候補位置が数ヶ所に限定されているため、扱いやすく、実装も比較的容易である。一方、Slider Models はラベルの配置候補位置を連続的に定義したモデルである。例として、点がラベルの上下左右のどの辺に接してもよい 4-slider model がある (図 1 (b))。Slider Models は Fixed-position Models に比べ、ラベルの配置候補位置が多い反面、実装が難しい。

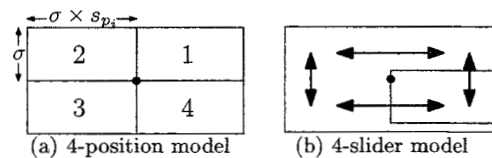


図 1 配置モデルの例

点ラベル配置問題には、ラベル数最大化問題とラベルサイズ最大化問題の 2 つの最適化問題が考案されている。ラベル数最大化問題は、ラベルサイズを固定した上で、配置できるラベルの数を最大化する問題である。そのため、ラベルサイズを指定することができるが、全てのラベルを配置できる保証がない (図 2)。一方、ラベルサイズ最大化問題は、全てのラベルを配置できる、最大のラベルサイズを求める問題である。そのため、全てのラベルを配置できるが、ラベルサイズを指定することができない (図 3)。

ラベル配置問題は、最適化の種類や配置モデル、ラベルを配置する対象などによって細かく分類され、それぞれ研究されている。また、それらの多くは NP 困難であることが知られている。ラベル配置問題に対する論文のリストは [3] に載っている。

### 全ラベル配置のための領域決定問題

広域図では、ユーザの指定するラベルサイズで全てのラベルを重ならず配置することは難しい。よって

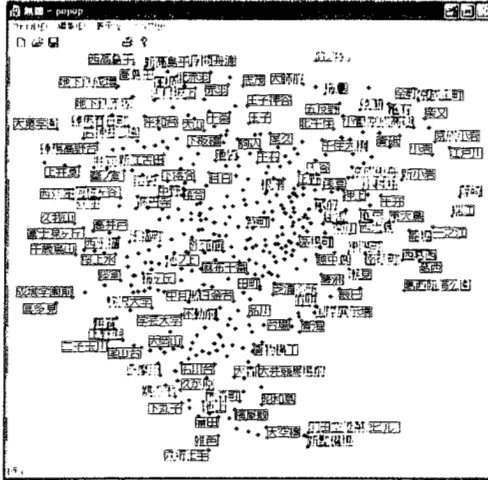


図2 ラベル数最大化

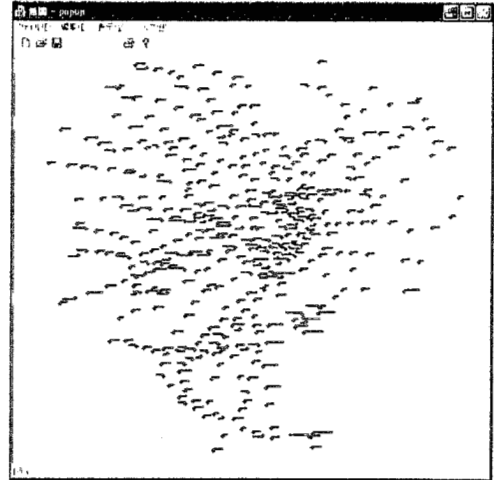


図3 ラベルサイズ最大化

本研究では、まずディスプレイに表示された広域図における点を、ユーザの必要に応じて指定し、その点周辺の詳細図を表示することを考える。詳細図は、縦  $h$ 、横  $w$  の長方形の決まった形とする。詳細図には、広域図の一部の領域を拡大表示し、表示される全てのラベルをユーザの指定するラベルサイズで、重ならず配置する。また、ユーザが指定する点が、詳細図の中心に表示されるものとする。本研究では、そのような条件を満たすように、広域図において詳細図に表示する領域を決定する。この領域を決定する問題を、全ラベル配置のための領域決定問題という。

ユーザが指定した点周辺のラベルを、指定したラベルサイズで重ならず配置し、詳細図として表示する。この最も単純な方法として、広域図においてラベルサイズ最大化問題を解き、その出力結果を拡大する方法がある。しかし、この方法では、ユーザの指定する点周辺の点の密集度によって、表示されるラベル数に大きな差が出てくる。詳細図に表示するラベルの数が多すぎると、地図がラベルで埋め尽くされてしまい、文字ではなく図の情報が失われてしまう。逆にラベルが少ない場合、情報が少なすぎて、探索に支障の出る場合がある。したがって、各点周辺の点の密集度に応じて、拡大表示する領域を決定する必要がある。よって、以下のような条件を考える。

- 詳細図の形は縦  $h$ 、横  $w$  の長方形とする。
- ユーザが要求する点  $p_{obj}$  を、詳細図の中心に表示する。
- 詳細図上のラベルの大きさは  $size_{pop}$  とする。
- 詳細図上のラベルは全て重なってはならない。

- 詳細図に占めるラベルの面積は、詳細図の面積の  $ratio$  倍以下とする。

これらの条件を全て満たす、最大の領域を求める問題を、全ラベル配置のための領域決定問題という。

### 3. アルゴリズム

本節では、全ラベル配置のための領域決定問題に対するアルゴリズムを提案する。まず、提案アルゴリズムの中で用いるラベル配置問題の既存アルゴリズムについて述べる。次に、全ラベル配置のための領域決定問題を解くための前処理について述べる。最後に、全ラベル配置のための領域決定問題を解くアルゴリズムについて述べる。

入力として、各点  $p_i \in N$  と、詳細図の高さ  $h$  と幅  $w$ 、ユーザの指定するラベルサイズ  $size_{pop}$ 、ユーザが指定する点  $p_{obj} \in N$ 、各点の座標  $(x(p_i), y(p_i))$  が与えられるとする。

#### 3.1 ラベルサイズ最大化問題のアルゴリズム [2]

本研究では、ラベルサイズ最大化問題を解く手法として [2] を用いる。この手法では、4-position model (図 1 (a)) を用いる。

次に、[2] のアルゴリズムで用いる用語を定義する。ラベルサイズが  $\sigma$  のとき、他の点を含むようなラベル配置候補位置を  $\sigma$ -dead (図 4 (a)),  $\sigma$ -dead でなく他のラベル配置候補位置と重なっているラベル配置候補位置を  $\sigma$ -pending (図 4 (b)), それ以外の  $\sigma$ -dead でも  $\sigma$ -pending でもないラベル配置候補位置を  $\sigma$ -alive (図 4 (c)) という。図 4 の (a) が  $\sigma$ -dead, (b) が  $\sigma$ -pending, (c) が  $\sigma$ -alive の例である。

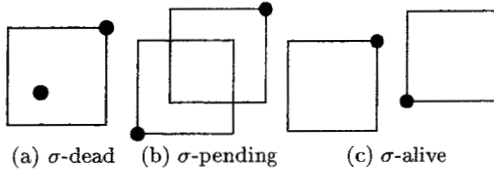


図4 ラベルの状態

### $\sigma$ -決定問題

平面上の  $n$  個の点集合を  $P = \{p_1, p_2, \dots, p_n\}$ , 全ての点  $p_i$  のラベル配置候補位置の集合を  $\Lambda$  とする。さらに、各点  $p_i$  に配置されるラベルの配置位置を 1 つずつ決め、 $\lambda_i$  とし、配置を  $\Lambda_p = \{\lambda_1, \lambda_2, \dots, \lambda_n\}$  とする。このとき、ラベルサイズ  $\sigma$  において、基本規則を満たす  $\Lambda_p$  が存在するか判定する問題を  $\sigma$ -決定問題という。 $\sigma$ -決定問題は、ラベル配置候補位置が 2 つ以下の場合 2SAT 問題に帰着し、多項式時間で解くことができる。しかし、ラベル配置候補位置が 3 つ以上ある場合、NP 完全である。そのため、[2] では、ラベル配置候補位置が 3 つ以上ある場合、2 つになるまで削除し、2SAT 問題に帰着することで、計算時間を多項式時間に抑えている。

文字数を考慮したラベルサイズ最大化問題のアルゴリズム [2]

4-position model に対する、ラベルサイズ最大化問題を解くアルゴリズム [2] を以下に記す。また、 $\sigma_l$ ,  $\sigma_u$  をそれぞれ、ラベルサイズ最大化問題の解の上限と下限とする。

1.  $\sigma_l := 0$  とする。各ラベル配置候補位置が  $\sigma$ -dead となる最小のラベルサイズのうち、最大のものを  $\sigma_u$  とする。
2.  $\sigma := \frac{\sigma_l + \sigma_u}{2}$ 。
3.  $\sigma$  において、ラベルを全て配置できるか判定する。
  - (a)  $\sigma$ -dead なラベル配置候補位置を削除する。
  - (b)  $\sigma$ -alive なラベル配置候補位置を持つ点は、そのラベル配置候補位置にラベルを配置し、他のラベル配置候補位置を削除する。これを  $\sigma$ -alive なラベル配置候補位置を持つ点が、なくなるまで行なう。
  - (c) ラベル配置候補位置が 3 つ以上残っている点に対して、各ラベル配置候補位置が  $\sigma$ -dead になる最小な  $\sigma$  を求め、 $\sigma$  の大きい 2 つを残し、残りのラベル配置候補

位置を削除する。

- (d)  $\sigma$ -alive なラベル配置候補位置を持たない点について、 $\sigma$ -決定問題を解く。
4. 3 (d) において、全てのラベルを配置できる場合は、 $\sigma_l := \sigma$  として 2 へ。配置できない場合は、 $\sigma_u := \sigma$  として 2 へ。
  5. 全てのラベルを配置できる最大のラベルサイズ  $\sigma_{res}$  が求まるまで、2, 3, 4 を繰り返す。
  6. 解を構成する。
    - (a)  $\sigma_{res}$ -alive なラベル配置候補位置を持つ点に関してはその中から 1 つ選ぶ。
    - (b) 残りの点に関しては  $\sigma_{res}$ -決定問題の解でラベルの配置位置構成する。

### 3.2 ラベルの再配置アルゴリズム [1]

前項で述べた手法は、4-position model を用いてラベルサイズ最大化問題を解いている。しかし、4-slider model の方が、ラベル配置候補位置が 4-position model より多いため、4-slider model を用いた方がラベルどうしを離して配置できる。よって、4-slider model を用いてラベルを配置した方が、ラベルがどの点に対して配置されているのかがわかりやすい。よって、4-position model を用いて配置したラベルの位置を初期位置として与え、4-slider model を用いてそのラベルを再配置する。これにより、配置済みのラベルが配置不可能になることなく、ラベルどうしをより離すことができる。本研究では、その手法として [1] の初期位置を考慮したアルゴリズムを用いる。

### 3.3 前処理のアルゴリズム

前処理では、各点  $p_i$  に対して、ラベルサイズ  $size_{p_i}$  を求める。これは、 $size_{p_j} \geq size_{p_i}$  となる点  $p_j \in N$  の集合を  $S$  としたとき、 $S$  に対してラベルサイズ  $size_{p_i}$  で、重ならずラベルを配置できるという性質をもつ。よって、 $size_{p_i}$  は、点  $p_i$  が詳細図上に表示される場合における、表示する領域と拡大倍率の下限となる。前処理では、ラベルの配置モデルとして 4-position model を用いる。さらに、ラベル配置候補位置は、右上から反時計回りに 1 から 4 と番号付けされているものとする (図 1 (a))。また、 $T_{p_x q_y}$  を点  $p$  のラベル配置候補位置  $x$  と点  $q$  のラベル配置候補位置  $y$  が重なる最小のラベルサイズとする。前処理のアルゴリズムを以下に示す。

#### 提案アルゴリズム

1.  $N$  に対してラベルサイズ最大化問題を解き、その解のラベルサイズを  $\sigma$  とする。
2. ラベル配置候補位置どうしが重なる最小のラベ



ルサイズ  $\tau_{p_x q_y}$  を全て求める。

3.  $\tau_{p_x q_y} > \sigma$  を満たす全ての  $\tau_{p_x q_y}$  からなる集合を  $\Sigma$  とする。
4. 最も値の小さい  $\tau_{p_x q_y} \in \Sigma$  を  $\sigma$  とする。
5.  $N$  に対して、 $\sigma$ -決定問題を解く。
  - (a) 全てのラベルが配置可能ならば 7 へ。
  - (b) 1 つでもラベルが配置不可能ならば 6 へ。
6. ラベルを配置できなかった各点  $p_i \in N$  に対して、 $size_{p_i} := \sigma$ ,  $N := N \setminus \{p_i\}$  とする。
7.  $N \neq \emptyset$  となるまで、 $\Sigma := \Sigma \setminus \{\sigma\}$  とし、4 へ。

1 のラベルサイズ最大化問題を解くアルゴリズムとして [2] を用いた。

### 3.4 表示する領域と拡大倍率を求めるアルゴリズム

本項では、前処理で求めた  $size_{p_i}$  を用いて、詳細図で表示する領域と拡大倍率、拡大表示する点集合  $M \subseteq N$  を求める。本研究では、詳細図の形を固定しているため、拡大倍率が決まると、詳細図で表示する領域も決定される。

入力としてラベルの占有率  $ratio$  も与える。 $N'$  は  $N$  の部分集合とし、詳細図に表示する候補の点の集合とする。 $s_{sum}$  は詳細図上に配置するラベルの文字数の和とする。 $l_j$  は、点  $p_{obj}$  と点  $p_j \in N \setminus \{p_{obj}\}$  の距離に相当する。アルゴリズムを以下に示す。

#### 提案アルゴリズム

1.  $p_i := p_{obj}$ ,  $M := \{p_{obj}\}$ ,  $size_{ans} := size_{p_{obj}}$ ,  $s_{sum} := s_{p_{obj}}$  とする。
2.  $p_j \in N \setminus \{p_{obj}\}$  のうち、
 
$$|x(p_j) - x(p_{obj})| \leq \frac{w}{2} \cdot \frac{size_{pop}}{size_{ans}}$$

$$|y(p_j) - y(p_{obj})| \leq \frac{h}{2} \cdot \frac{size_{pop}}{size_{ans}}$$
 を満たす点の集合を  $N'$  とする。また、各点  $p_j \in N'$  に対して、
 
$$l_j = \max \left( \frac{|x(p_j) - x(p_{obj})|}{h}, \frac{|y(p_j) - y(p_{obj})|}{w} \right)$$
 を求める。
3.  $l_j$  が最小の  $p_j \in N'$  を  $p_i$  とする。
4.  $s_{sum} + s_{p_i} > \frac{h \cdot w \cdot ratio}{size_{pop}^2}$  ならば 7 へ。
5.  $size_{ans} \leq size_{p_i}$  ならば 5 (a) へ。そうでないならば 5 (b) へ。
  - (a) 拡大倍率  $\frac{size_{pop}}{size_{ans}}$  に拡大表示する際に、点  $p_i$  のラベルが詳細図上に全て配置できるならば、5 (d) へ。配置できない可能性があるならば、6 へ。
  - (b) 拡大倍率  $\frac{size_{pop}}{size_{p_i}}$  に拡大表示する際に、点  $p_i$  と各点  $p_j \in M$  のラベルが、全て詳細図上に配置できるならば、5 (c) へ。

配置できない可能性があるならば、6 へ。

- (c)  $size_{ans} := size_{p_i}$  とし、5 (d) へ。
- (d)  $s_{sum} := s_{sum} + s_{p_i}$ ,  $N' := N' \setminus \{p_i\}$ ,  $M := M \cup \{p_i\}$  とし、3 へ。

6. 拡大倍率  $\frac{size_{pop}}{size_{ans}}$  と、点集合  $M$  を出力する。

また、4 を除いたアルゴリズムでも、指定されたラベルサイズで、全てのラベルを重ねずに配置できる拡大倍率を求められる。しかし、詳細図上に配置されるラベルが多くなりすぎて、画面がラベルで埋めつくされる可能性がある。4 はこのような状況を防ぐため、詳細図に占めるラベルの面積を抑える処理である。

## 4. 計算機実験

前節で提案したアルゴリズムを実装し、計算機実験を行なう。実験環境を以下に示す。

CPU : Pentium M 1.7 GHz

メモリ : 1 GB

OS : Microsoft Windows XP

コンパイラ : Microsoft Visual Studio 2003 C++

また、本研究では、目的地や現在地の探索を念頭においているため、広域図と詳細図を同時に見るといった要求が考えられる。そのような要求にも応えられるようにするため、本研究では、ポップアップシステムを用いることで、広域図と詳細図を同時に表示できるようにした。ポップアップとは、新しいブラウザウィンドウを自動的に開き、別の内容を表示することである。

#### 鉄道駅に対する実験

実験データとして、昭文社から 2005 年に刊行された Mapple 2500 より、東京 23 区内の鉄道駅を用いた。実験では、ラベルの占有率  $ratio$  を 0.4 とし、詳細図の大きさは高さ 200、幅 300 とした。また、詳細図でのラベルサイズは 16 とした。

日本橋を入力した実験結果を図 5、図 6 に示し、お花茶屋を入力した実験結果を図 7、図 8 に示す。図 5 と図 7 は広域図で、黒枠で囲まれた領域が、各点を入力としたときの、全ラベル配置のための領域決定問題を解いて出力された領域である。また、図 6 と図 8 は黒枠で囲まれた領域の詳細図である。日本橋周辺は、他の駅が非常に密集しているため、拡大表示する領域が非常に小さくとられている。一方、お花茶屋周辺は、周辺にあまり駅が密集していないため、拡大表示する領域が大きくとられている。このことから、周りに点が密集している点とそうでない点で、拡大表示する領域が変化していることがわかる。

また、入力する点やラベルサイズを変更しても、全

