

引出し線ラベル配置に対する解法と実装

下原 史義*

今井 桂子†

* 中央大学大学院 理工学研究科 情報工学専攻

† 中央大学 理工学部 情報工学科

概要 平面上に n 個の点集合が与えられているとき、これらの点に対しラベルを配置する問題を NLP (Node Label Placement) 問題という。ここでは配置できるラベルの数が最大となるような問題を考える。この NLP 問題に対して、引出し線を用いた配置モデルが提案された。また、グラフ描画、メッシュ生成、引出し線を用いていないラベル配置においては力学を応用した解法が研究されている。本稿では引出し線を用いた NLP 問題に対してこのような解法を応用したアルゴリズムを提案し、計算機実験をおこなう。

An Algorithm for Node Label Placement Problems with Leader Lines

Fumiyoshi SHIMOHARA*

Keiko IMAI†

*Information and System Engineering Course,

Graduate School of Science and Engineering, Chuo University

†Department of Information System and Engineering, Chuo University

Abstract In this paper, we consider the node label placement (NLP) problem. Given a set of n points in the plane, the problem is maximizing the number of labeled points. For the NLP problem, a new label layout model using leader lines has been proposed. On the other hand, some algorithms based on dynamics have been investigated in graph drawing, mesh generation, and the label placement without leader lines. In this paper, we propose an algorithm based on dynamics for NLP with leader lines, and experimental results are also shown.

1 はじめに

平面上に描画された地図やグラフに対して、描画上の対象物に文字注記を行う問題をラベル配置問題という。地図において点は都市や建物を、辺は川や路線を、領域は湖や地域などを表している。こういった対象物に対して適切な位置に文字注記を行うことは、地図の自動描画において重要な問題である。[8] では様々なラベル配置問題のモデルやアルゴリズム、文献を紹介している。点に対してラベルを配置する問題は NLP (Node Label Placement) 問題と呼ばれ、ラベル配置問題の中でも最も基本的な問題として研究が進められている。本研究ではこの NLP 問題を主に考える。

NLP 問題には、ラベル数最大化とラベルサイズ最大化という 2 つの最適化問題がある。ラベル数

最大化ではラベルの大きさを固定した上で、他のラベルや入力点と重ならないようなラベルの数を最大化する。一方、ラベルサイズ最大化では、全ての点に対して配置できるようなラベルのサイズを最大化する。本研究ではラベル数最大化について取り扱う。

NLP 問題のラベル数最大化の配置モデルとしては、点とラベルが接するように配置するモデルの研究が盛んにおこなわれている。従来のモデルは固定位置モデル (Fixed-position Model) とスライダモデル (Slider Model) が知られている。点のスライドを許す辺の数が k 本のスライダモデルを k -slider model と呼ぶ。スライダモデルでは少なくとも最適解の半分はラベルが配置できる $O(n \log n)$ 時間の近似アルゴリズム [7] や、最適解

を求めるアルゴリズム [3] が知られている。

これらのモデルは点と接するようにラベルを配置するため、点が密集している場合にはどうしてもラベルを配置できないことがある。このようなとき、実際の地図ではしばしば注記する対象から離れた空白に注記をおこない、線分で結ぶことによって対象物と注記を明確にしているような例が見られる。このような線分のことを地図用語で引出し線 (Leader Line) という。

この引出し線をラベル配置問題に適用したモデルが [5] によって提案され、[4], [5] ではそのモデルに対するアルゴリズムが提案された。

グラフ描画やメッシュ生成など他の問題においては、物理モデルを応用した解法が提案されている [1], [2]。これらの手法はまず描画上の要素を物理的な対象とみなし、お互いの位置関係から引力や斥力といった影響力を加える。そしてその影響力によって要素を移動させていき、配置をおこなうといったものである。[6] ではこのような配置手法を引出し線のないラベル配置問題に応用する方法について述べている。

本研究では引出し線を用いたラベル配置問題に対して、[6] の考えを応用した解法を提案し、実際の地図データを用いた計算機実験をおこなう。

2 引出し線ラベル配置

引出し線を用いたラベル配置のモデル化については、[4], [5] で議論されているが、本研究では以下のような定義のもとでの最適化を考える。

平面上の配置領域 A と A の内部に存在する n 個の点集合 $P = \{p_1, p_2, \dots, p_n\}$ 、各点に注記されるラベルの集合 $L = \{l_1, l_2, \dots, l_n\}$ が与えられたとする。ラベルは長方形で定義し、回転による配置はしない。またラベルは配置するための参照点を持ち、ここではラベルの中心点を参照点とする。各ラベルの参照点が $Q = \{q_1, q_2, \dots, q_n\} \subset A$ に配置されたとき、以下の制約を満たすラベルを配置可能ラベルと呼ぶ。

- 配置領域の内部に配置されている
- 他のラベルと重ならない
- 入力点と重ならない
- 引出し線と重ならない

引出し線は配置されたラベルの内部または境界のどれか一点と、注記する点を結んだ線分で定義で

きる。今回は配置されたラベルの中心点から注記する点を結んだ線分を引出し線としており、描画ではそのうちのラベルの内部の部分削除している (図 1)。

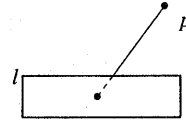


図 1. 引出し線の定義

このような定義のもとで、配置可能ラベル数を最大化する問題を取り扱う。[4], [5] では引出し線の交差を許していないが、実際の地図では引出し線が交差しているような例も見られる。よって本研究では引出し線の交差を許すことにする。以降このような引出し線を用いたラベル配置問題を引出し線ラベル配置と呼ぶ。

3 力学モデルを応用した配置手法

この節では [6] で提案している配置手法について説明する。本研究で提案する手法はこの手法を引出し線ラベル配置に応用したものである。[6] の手法はあるラベルを考えたときに、他のラベルや点に対して適切配置領域と呼ばれる領域を考える。ラベルが適切配置領域の外にあれば、現在の位置から適切配置領域の最も近い位置に移動させるような力として影響力を発生させる。この影響力はラベルが他の描画要素に近付きすぎると斥力がかかり、注記する要素と遠すぎると斥力がかかるようになっている。

この影響力に力学の運動方程式を適用し、ラベルを適切な位置に移動させていく手法である。

アルゴリズム

手順 0 ラベルをあらかじめ定めておいた初期位置に配置する。ラベルの初速度は 0 とする。

手順 1 各ラベルについて、他のラベルや点から受ける影響力の合力を求める。

手順 2 各ラベルの合力、速度から運動方程式を計算し、それによってラベルを単位時間分移動させる。

手順 3 手順 1 に戻る。

実際はユーザーが手順 2 から 3 の繰り返し回数を指定するか強制的に終了させることで、最終的な配置を得る。

4 引出し線ラベル配置に対する解法

[6] の手法を引出し線モデルに応用する方法について考える。

ラベル配置問題の基本的な制約として以下のようなのがある。

- ラベルは他の描画の対象となる要素と重なってはいけない。
- ラベルはどの対象物を指しているのか明白でなければならない。

引出し線モデルは注記する対象とラベルを線で結んでいるので 2 つ目の制約は満足している。引出し線を用いるので、ラベルは注記する対象の近くに配置されなくてもよい。よって提案手法では [6] で用いられている適切配置領域は考慮しない。その上で 1 つ目の制約を満足するような影響力を与えてやればよい。

4.1 影響力の定義

ラベルを l 、入力点を p 、 l が作る引出し線を s_l とする。これらラベル、入力点、引出し線といった描画の対象になる要素をオブジェクトと呼ぶことにする。影響力はラベルの中心点にかかるものとする。ラベル l は他のオブジェクトと交差する場合、それぞれのオブジェクトから以下のような影響力がかかるものとする。

- 他のラベル l' から受ける影響力 $F_1(l, l')$
- 入力点 p から受ける影響力 $F_2(l, p)$
- 他のラベルが作る引出し線から受ける影響力 $F_3(l, s_{l'})$
- l が作る引出し線に重なるラベルから受ける影響力 $F_4(l, l')$

$F_1(l, l')$ は l' の中心点から l の中心点に向かう方向にかかるものとする。この場合 l' には $F_1(l, l')$ とは逆向きに力 $F_1(l', l)$ がかかることになる (図 2)。 F_1 と同様に $F_2(l, p)$ は p から l の中心点に向かう方向にかかるものとする (図 3)。 $F_3(l, s_{l'})$ は $s_{l'}$ に垂直で、 $s_{l'}$ から p 向かう方向にかかるものとする。それに対して、 $F_4(l, l')$ は $F_3(l, s_{l'})$ の逆向きにかかるものとする (図 4)。

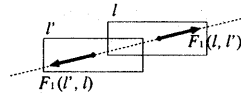


図 2. 影響力 F_1

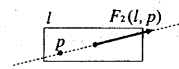


図 3. 影響力 F_2

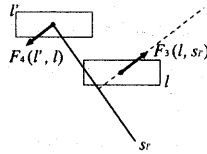


図 4. 影響力 F_3 と F_4

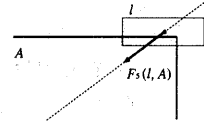


図 5. 影響力 F_5

影響力は重なっている対象と近ければ近いほど強い力を、遠ければ弱い力をかけるべきである。そこで影響力の大きさはラベルの中心点と他のオブジェクトとの距離の関数で与える。以後、距離はユークリッド距離とする。 l の中心点と l' の中心点との距離を $d_{ll'}$ 、 p との距離を d_{lp} 、 $s_{l'}$ との距離を $d_{ls_{l'}}$ とおく。つまりそれぞれの影響力の大きさ $|F|$ は $|F_1(l, l')| = f_1(d_{ll'})$ 、 $|F_2(l, p)| = f_2(d_{lp})$ 、 $|F_3(l, s_{l'})| = f_3(d_{ls_{l'}})$ 、といった関数で与え、 $|F_4(l, l')| = |F_3(l, s_{l'})|$ とする。

影響力を与える関数は、どのような関数を与えるかによって結果が大きく変化すると考えられる。今回はユーザーが決定するものとし、関数の具体的な例は計算機実験で明記する。

また、ラベル l が配置領域 A の外に移動してしまった場合、ラベルを配置領域内に移動させるような影響力を与える必要がある。よってラベルが他のオブジェクトから受ける影響力と同様に定義をおこなう。 l が A の外部に配置された場合、 l の中心点から A の中心点に向かう方向に影響力 $F_5(l, A)$ がかかるものとする (図 5)。影響力の大きさは l の中心点と A の中心点の距離を d_{lA} とすると $|F_5(l, s_{l'})| = f_5(d_{lA})$ といった関数で与える。

このように影響力を定義すると、あるラベルの中心点が他のラベルの中心点や入力点、引出し線に重なる場合は方向が定義できず影響力が発生しない。そこでこのような場合にはランダムな方向に微小な影響力を与えるといった例外処理をおこなう。

4.2 アルゴリズム

前述した影響力を用いて以下のアルゴリズムでラベル配置をおこなう。

引出し線ラベル配置に対するアルゴリズム

手順 0 ラベルをあらかじめ定めておいた初期位置に配置する。ラベルの初速度は 0 とする。

手順 1 全てのラベルの移動距離が 0 になるか、指定回数に達するまで次の手順を繰り返す。

手順 1-a 各ラベルについて、他のラベルや点から受ける影響力の合力を求める。

手順 1-b 各ラベルを合力の向きに、その大きさだけ移動させる。

手順 2 全てのラベルの移動距離が 0 になるか、指定回数に達するまで次の手順を繰り返す。

手順 2-a 各ラベルについて、他のラベルや点、引出し線から受ける影響力の合力を求める。

手順 2-b 各ラベルを合力の向きに、その大きさだけ移動させる。

手順 1 ではラベルと点のみから受ける影響力を計算し、手順 2 から引出し線を考慮した影響力の計算をおこなっている。これはラベルを適当な位置に移動させてから引出し線を考慮に入れることによって、アルゴリズムの初期段階における無駄な計算を避けるためである。

[6] は運動方程式を計算しているため、ラベルを移動させる際に繰り返しの前の移動速度が次の移動に考慮されている。しかし提案手法において運動方程式を考慮するとラベルが必要以上に重なってしまったため、提案手法では考慮していない。

5 計算機実験

前節で述べたアルゴリズムについて計算機実験をおこなった。[4] でおこなわれた実験と同じ入力データを用いて実験をおこない、その結果に対しても [4] で提案されている手法との比較をおこなった。

5.1 実験方法

入力データは国土地理院の平成 9 年度版数値地図 2500 (空間データ基盤) から、東京 23 区内における地下鉄路線図の駅の座標を抽出したものと、横浜市中区周辺から代表点の座標を抽出したものをを用いている。ラベルの高さは全て一定とし、フォントサイズから決定する。また、各ラベルの幅はフォントサイズに注記の文字数をかけたものとしている。フォントサイズは東京 23 区地下鉄路線図のデータにおける実験では 6pt、横浜市中区周辺の実験では 4pt としている。

[4] ではいくつかアルゴリズムが提案されているが、入力によって解の精度が異なる。そこでそれぞれ入力に対して最も解の値がよかったものを [4] の手法の結果として取り扱う。また、[4] では配置領域を入力として考慮していないため、比較のためにおこなう実験では配置領域 A と配置領域を考慮した影響力 F_3 は考慮していない。

影響力を与える関数は、ラベル l の対角線の $1/2$ の長さを r_l としたとき、以下のように与えた。

- $f_1(d_{uv}) = \frac{1}{2}(-d_{uv} + r_l + r_v)$
- $f_2(d_{lp}) = \frac{1}{2}(-d_{lp} + r_l)$
- $f_3(d_{ls_i}) = \frac{1}{4}(-d_{ls_i} + r_l)$

これらはラベルが他のオブジェクトと交差している場合、距離が短い場合は大きい影響力を、距離が長い場合は小さい影響力を与え、ラベルの大きさを考慮した関数になっている。

各ラベルの初期配置は、各ラベルの参照点がそれぞれ注記する入力点と一致するように配置した。また、提案手法の手順 1、手順 2 の最大反復回数はそれぞれ 100 とした。各実験はそれぞれのアルゴリズムについて、配置可能ラベル数 N_l 、計算時間 T 、引出し線の交差数 N_{s_i} を求める。提案手法はランダムな処理が含まれているため、実験を 100 回おこなったデータに対して平均と標準偏差を求めて比較の対象とした。実験環境は [4] の手法、提案手法ともに CPU が UltraSPARC-III (750MHz)、メモリが 512MB となっている。

5.2 実験結果

東京都 23 区の地下鉄路線図のデータにおける実験結果を表 1、ラベルの配置図の一つを図 6 に示す。また横浜市中区周辺のデータにおける実験結果を表 2、ラベルの配置図の一つを図 7 に示す。図

6, 7 はどちらもすべてのラベルが配置可能ラベルとなった結果である。

表 1. 地下鉄路線図に対する実験結果 ($n = 178$)

アルゴリズム	N_l	T (秒)	N_{s_l}
[4] の手法	144	4.43	0
提案手法 (平均)	177.01	10.39	0.46
(標準偏差)	0.99	52.40	8.05

表 2. 横浜の地図に対する実験結果 ($n = 287$)

アルゴリズム	N_l	T (秒)	N_{s_l}
[4] の手法	250	154.22	0
提案手法 (平均)	285.87	44.99	14.66
(標準偏差)	45.24	206.97	63.02

[4] は引出し線の交差を許していないため一概に比較はできないが、平均で見ると、どちらのデータに対しても提案手法は [4] の手法よりも多くのラベルを配置することができた。しかし、横浜市中区周辺のデータの結果では標準偏差が大きい。また、引出し線の交差が多い場合は非常に見にくい図になってしまうこともあった。

6 おわりに

引出し線ラベル配置に対して、[6] を応用した手法について研究した。計算機実験の結果、実験した入力データでは、[4] で提案されたアルゴリズムよりも多くのラベルを配置できることを示した。しかし提案手法は結果にばらつきがでてしまうことが多い。実験で与えた初期配置では、始めの段階で例外処理が一斉に起こるため、ラベルがランダムな方向に移動してしまう。その方向が結果にばらつきを与えていると考えられる。

提案手法における影響力はオブジェクト同士の交差関係によって与えられるものであった。だが、引出し線を用いる場合は配置領域全体を見た上で空いている空間にラベルを配置する、という考えが自然である。これは点が密集している領域から空いている領域に向かってラベルを移動させるような、ラベルの配置位置に応じた影響力を新たに加えることによってより自然な配置になると考えられる。

今後の課題としては、有効な初期配置や初速度の設定などの前処理、配置後の交差しているラベルの効率的な削除などの後処理の検討、ラベルの配置位置に応じた影響力の検討、引出し線を用いていない従来のモデルに対する近似アルゴリズムとの比較することなどが挙げられる。

謝辞

本研究の一部は文部科学省研究費補助金 (基盤研究 (C)) の補助を受けている。

参考文献

- [1] 阿部 昇, 増田 澄男, 山口 一章, “辺がラベルをもつ無向グラフの描画法,” 電子情報通信学会論文誌 (A), Vol.J86-A, No.8, pp. 848-859, 2003
- [2] IBM 基礎研究所, “Bubble Mesh ... Physically-based Triangulation,” 1998.
<http://www.trl.ibm.com/projects/meshing/bubble/bubble.htm>
- [3] G. W. Klau and P. Mutzel, “Optimal labelling of point features in the slider model,” *Proc. 6th Annual International Computing and Combinatorics Conference (COCOON'00)*, Lecture Notes in Computer Science 1858, pp. 340-350, 2000.
- [4] 大塚善仁, “引出し線を用いたラベル配置問題,” 中央大学大学院理工学研究科情報工学専攻修士論文, 2003.
- [5] 大塚 善仁, 今井 桂子, “引出し線を用いたラベル配置問題,” 情報処理学会研究報告, 2001-AL-79, pp. 53-58, 2001.
- [6] 佐藤 聡, 有川 正俊, “力学モデルに基づく地理データの動的表示システム,” 地理情報システム学会 第 3 回オブジェクト指向 GIS ワークショップ予稿集, pp. 31-36, 2001.
- [7] M. van Kreveld, T. Strijk and A. Wolff, “Point Set Labeling with Sliding Labels,” *Proc. 14th Annu. ACM Symp. on Computational Geometry*, pp. 337-346, 1998.
- [8] A. Wolff, “Map Labeling,” 2002.
<http://il1www.ira.uka.de/map-labeling>



図 6. 東京 23 区地下鉄路線図のデータにおける配置結果

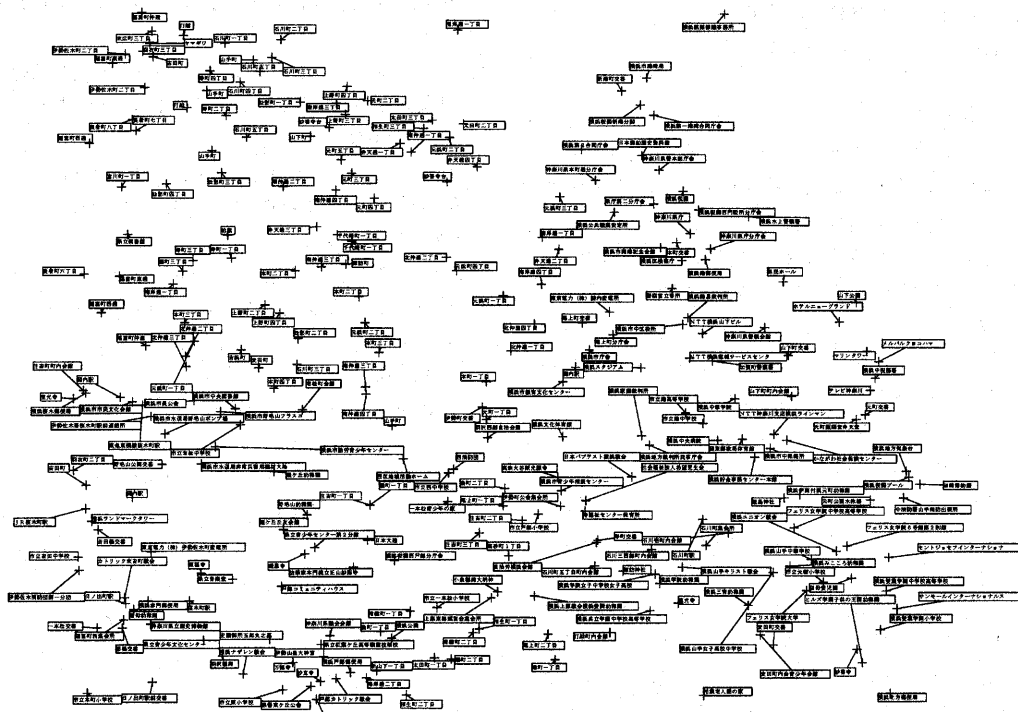


図 7. 横浜市中区周辺のデータにおける配置結果