

ラベルの交差を許した点ラベル配置における最適化問題

Optimization Problems in Point-Feature Label Placement with Intersecting Labels

情報工学専攻 尾野 航平

Information and System Engineering ONO Kohei

概要

地図などにおいて、適切な位置に注記の書かれたラベルを対象物の点に配置する問題を点ラベル配置問題という。本研究では、配置後に手動で交差を容易に解消できるようなラベル配置を計算することを目的とする。そのために、交差しているラベル対の数と交差している面積の二乗和を新たな最適化基準として提案し、これらを最小化するラベル配置を考える。また、これらの問題に対し、発見的解法を提案し、計算機実験を行ない評価する。貪欲算法に基づくアルゴリズムを何度も行ない、ラベル配置を得ることで、良い解が出力できる。

キーワード： ラベル配置問題，点ラベル配置，交差数最小化，交差面積最小化，貪欲法

1 序論

地図作成においては、描かれた対象物の説明をするために文字などによる注記が行なわれる。注記（ラベル）は長方形の中に書かれると考える。注記対象が点であり、適当な位置、大きさなどの制約を満たすようにラベルを配置する問題を点ラベル配置（NLP）問題といい、一般的に、NLP問題はNP困難である。

これまで、NLP問題はラベルの交差がないことを前提に解かれてきたが、近年、ラベル同士の重なり（交差）を許す自動配置に関する研究がされている。de Berg と Gerrits は、ラベルを単位正方形とし、他のラベルと交差していないラベル（フリーラベル）数を最大化するフリーラベル最大化問題を提案し、近似アルゴリズムを与えた [1]。また、動的な点に対し、フリーラベル最大化問題のアルゴリズムを離散的に使用し、その間を補間するアルゴリズムを与えた [2]。

動的な点に対するラベル配置問題の応用例として、航空管制における監視レーダーシステムが挙げられる。このシステムでは、すべての動く航空機に対し、航空機名などの情報をもつラベルを表示している。重なってしまっても、ラベルを削除することはせず、手動でラベルの位置を移動させることにより、知りたい情報を得ている。このシステムにおいて、フリーラベルを多くしようとすると、多くのラベルが重なっている箇所が現れる傾向がある。そのような箇所では、ラベルの情報を得るためにラベルを移動する作業は手間がかかる。この問題を解消するために、フリーラベル数とは異なる最適化基準を作る必要があると考えた。そこで、本稿では、交差しているラベル対の数（交差数）と交差している面積の二乗和を新たな最適化基準として、用いることにより、手

動処理に有効かどうかを検証する。

2 ラベルの交差を許した最適化問題

平面上に与えられた n 個の点の集合を P とし、各点は点の右上、右下、左上、左下に正方形の4つのラベル候補を与えたものを入力とする。指定された最適化基準により、全点に対して、4つの候補から必ず1つを選び、そのラベル位置を解として出力する。ここでは、ラベル同士が辺で接している場合を交差とみなさない。また、ラベル同士が重なることに加え、ラベルが他の点と重なることも許す。交差を手動で解消するための手間を考慮し、どのラベル候補を選ぶかを決める最適化基準として、以下の2つを提案する。

・交差数最小化問題

この問題の目的関数は、交差しているラベル対の数（交差数）であり、これを最小化する。図1は、同じ点の配置に対して、フリーラベル最大化と交差数最小化を行なった最適解を示している。(a)はフリーラベル数3、交差数10、(b)はフリーラベル数0、交差数4である。

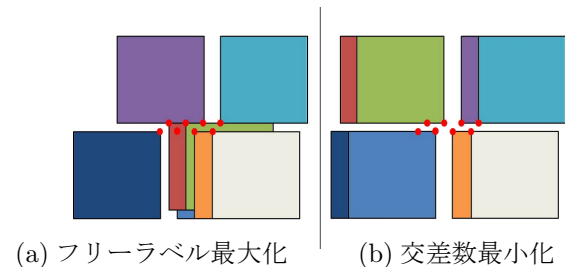
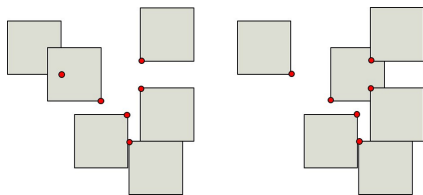


図1. 既存と提案の最適化問題の解の違い

・総交差面積最小化問題

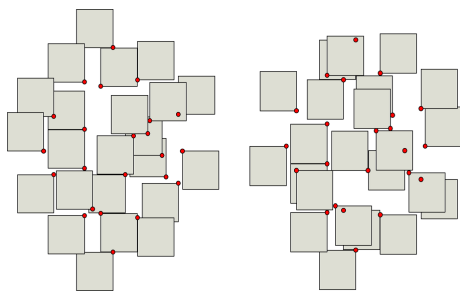
この問題の目的関数は、交差しているラベル対の交差面積の二乗和（総交差面積）であり、これを最小化する。この目的関数は、大きな重なりを減らすことを表す。単

純に面積の総和であると、大きな重なりによるものと少しの重なりが複数あるもので同じ総和になることがある。その場合、後者が適していると考え、交差面積の二乗を行なう。図2は同じ点の配置に対して、面積の総和と面積の二乗和で最小化を行なった最適解を示している。また、図3は同じ点の配置に対して、交差数と面積で最小化を行なった最適解を示している。



(a) 面積総和の最小化 (b) 面積二乗和の最小化

図 2. 最適解の違い



交差数 8

交差数 11

(a) 交差数最小化 (b) 交差面積最小化

図 3. 2つの最適化問題の解

ラベルを交差せずに配置可能であるかという決定問題が NP 完全 [3] であることから、交差数最小化問題、総交差面積最小化問題はともに NP 困難である。したがって、本稿では、発見的手法によって、実用的な解を求めるアルゴリズムを提案することを考える。

3 基本アルゴリズム

提案する解法に共通して用いられる貪欲算法に基づく基本アルゴリズムは以下の通りである。

基本アルゴリズム

1. 初期状態を求める。
2. P の各点を処理する順序を決定する。
3. 2で求めた順に、各点の4つのラベル候補に対して交差数または交差面積を計算し、最も値が良いラベル候補を決定して配置を更新して行く。値の良いラベル候補が複数あった場合は候補ラベルの優先順位に従って選択する。

初期状態とは、点に処理を行なう前の各点に対するラベルの状態のことである。初期状態の求め方、点の処理順序、ラベル候補の優先順位の決め方の組合せによって、異なるアルゴリズムを構成できる。また、上記の基本アルゴリズムの結果として得られた解を初期状態とし、基本アルゴリズムを反復し、実行する手法も考えられる。

3.1 初期状態の生成方法

初期状態の生成方法は次のもの考えた。

- (1) 配置なし：すべての点にラベルを配置しない状態のものを初期状態として扱う。
- (2) 大きな正方形の配置：各点において、4つの候補ラベルをすべて配置したラベルの2倍の辺長をもつ正方形（配置可能領域）を配置する（図4）。
- (3) 基本アルゴリズムの解による配置：基本アルゴリズムで各点に選ばれたラベル候補を配置する。

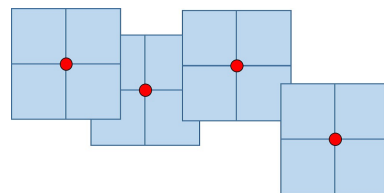


図 4. ラベル配置可能領域 (水色)

基本アルゴリズムのステップ2において、 i 番目の点を処理することを考える。 $i-1$ 番目までの点には、それまでに選択したラベルを配置し、 $i+1$ 番目以降の点には初期状態で決めた四角形（空集合の可能性もある）を配置する。これらに対して、 i 番目の点の4つのラベル候補それぞれに、交差数や交差面積を求め、同じ値であった場合には、優先順位に従って、 i 番目のラベルの位置を決定する。開始時に、ラベルの交差や交差面積を計算するために(1)、(2)を用いる。(3)は反復の場合のみ使用する。

3.2 点の処理順序の決定方法

点の処理順序の決定方法は次のもの考えた。

- (1) ランダム
- (2) x 座標の昇順・ y 座標の昇順
- (3) 点の密集度の降順：ある点の近くに、その点のラベルと交差する可能性のあるラベルを持つ点があるかどうかを次のようにして評価する。3.1で述べた配置可能領域を考える。交差数最小化では、その点の配置可能領域と交差する他の点の配

置可能領域の数を、総面積最小化問題では、その点の配置可能領域と交差する他の点の配置可能領域の面積の和を密集度とする。

3.3 基本アルゴリズムの計算速度

基本アルゴリズムのステップ3で、交差数や交差面積の計算を行なうが、配置を行なおうとしている点以外のすべての点に配置されているラベルと交差判定を行なうことは無駄である。それは、与えられた点集合とラベルのモデル、サイズによって、どのように配置をしてもラベルが交差しない点対が存在し得るからである。そのような無駄な計算を省くため、アルゴリズムを実行する前にどの点と点にラベルを配置したら交差する可能性があるのか覚える必要がある。交差する可能性のある点同士を辺で接続したグラフを考えたときの平均次数を d_{ave} とする。点の数が n 個のとき、基本アルゴリズムを1回実行する計算時間は $O(nd_{ave})$ である。

4 提案アルゴリズムと計算機実験

3.1, 3.2で述べたものを組み合わせた手法の有効性を計算機実験によって確かめた。組み合わせた提案アルゴリズムは以下ようになる。

提案アルゴリズム

1. 優先順位, 反復条件, 終了条件の決定.
2. **while** (反復条件)
基本アルゴリズムの実行.
3. **if** (終了条件) 実行終了
else 優先順位の変更.

組み合わせはたくさんあるが、その内のいくつかを紹介する。実験に用いたデータは、航空管制システムのある瞬間の78機の航空機の位置を表している。

4.1 交差数最小化問題

ラベル候補は4個あるため、ラベル候補の優先順位は24パターン存在する。交差数最小化問題に対し、航空機のインスタンスを用い、次の手法で実験を行なった。航空機インスタンスの最適値は5である。

アルゴリズム1. 開始時の初期状態はなし、反復を行なう際の初期状態は1つ前の反復で得られた解を初期配置とする。点の順序は x 座標の昇順である。ラベル候補の優先順位は、ランダムに1パターンの中から選択し、基本アルゴリズムを1回行なう。上記の動作を“定められた回数”反復させ、そのときの解を覚える。以上の流れを1回の実行とする。実行回数に関しても“定められた回数(秒数)”だけ反復し、一番良かった解を出力する。

アルゴリズム1の条件を次のように設定し、比較する。

条件1 アルゴリズム1における1回の実行あたりの基本アルゴリズムの反復回数を15とし、100回実行する。

このとき、100回の実行時間は0.13秒程度であり、100個の解の平均値は5.71、最良値は5、最悪値は7であった。つまり、最終的に交差数5の解を出力する。

条件2 アルゴリズム1における1回の実行あたりの基本アルゴリズムの反復回数を150とし、100回実行する。

このとき、100回の実行時間は1.04秒程度であり、100個の解の平均値は5.03、最良値は5、最悪値は6であった。つまり、最終的に交差数5の解を出力する。

基本アルゴリズムの反復回数の増加に伴い、実行時間は増加するが、1実行あたりの解の交差数は少なくなることが見られる。

条件1と条件2を1000回行なったときの、最終的な解の最良値と最悪値と、それぞれの出現回数を求めた。その結果、1000回のすべてにおいて、最適値5が得られたため、最悪値は出現しなかった。

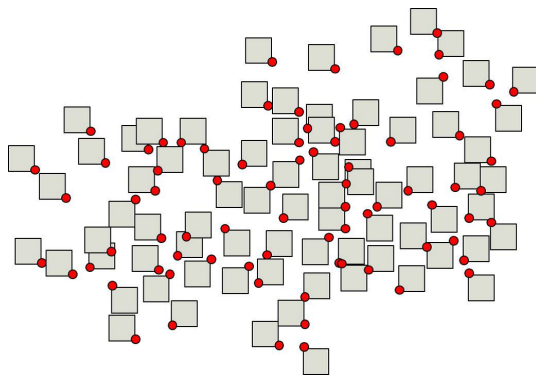
アルゴリズム1では、多くの反復を行なうため、無駄な反復があり計算時間を費やしていることが考えられる。次に、反復回数や優先順位の定め方を考慮するため、以下の手法で実験を行なった。

アルゴリズム2. 開始時の初期状態はなし、反復を行なう際の初期状態は1つ前の反復で得られた解を初期配置とする。点の順序は x 座標の昇順である。ラベル候補の優先順位は、最初は左候補を優先して基本アルゴリズムを解が更新されなくなるまで反復する。同じ操作を、右優先、上優先、下優先と優先順位を変えて実行する。

アルゴリズム3. 開始時の初期配置は大きな正方形、点の順序は密集度の降順である。反復以降の初期状態と優先順位はアルゴリズム1と同じにし、基本アルゴリズムを反復する。

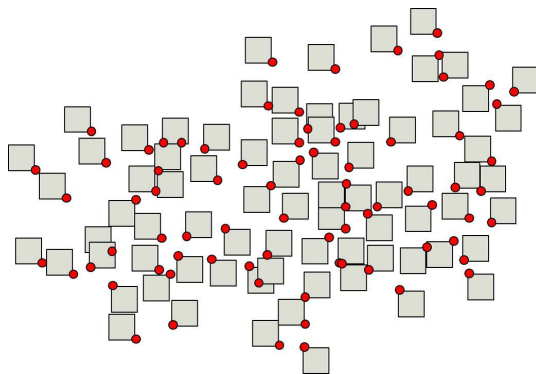
アルゴリズム2, 3を実行した結果が図5と図6である。このインスタンスに対する最適解は交差数5であり、アルゴリズム3では最適解が得られた。

点数が多いとき、局所解まで基本アルゴリズムを反復させる手法を適用すると計算時間がかかる。例えば、一様ランダムに発生させた3000点を各点間の距離の比率を保ったまま縮小させ、最大次数 d_{max} と平均次数 d_{ave} を変更させたものに対して、アルゴリズム3を実行する。



交差数 6

図 5. アルゴリズム 2 の出力



交差数 5

図 6. アルゴリズム 3 の出力

表 1 から点が密集していると、多くの点との交差判定を行なう必要があり計算時間がかかる。リアルタイムで解を求めたいという要求のときには、計算時間に数秒もかけることができない。よって、基本アルゴリズムの反復が少なくなる反復条件と終了条件で実験を行なった。

表 1. 各点の密集具合の変化による計算時間

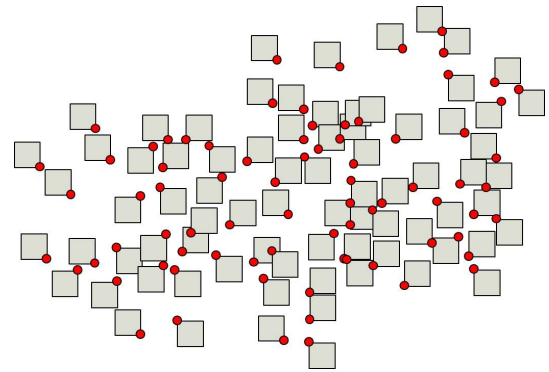
最大次数	平均次数	時間 (秒)	反復回数
$d_{max} : 28$	$d_{ave} : 12.8$	0.16	26
$d_{max} : 1391$	$d_{ave} : 928.0$	14.00	28
$d_{max} : 2999$	$d_{ave} : 2999.0$	30.00	19

その結果、基本アルゴリズムを 2 回くらいしか行なえないときは、点の順番はランダムの方が良い結果が得られた。4 回以上の反復を行なうことができる場合、密集した点から行なう方が良い結果が得られた。

4.2 総交差面積最小化問題

総交差面積最小化問題に対し、アルゴリズム 1 の目的関数を変更し、実行すると良い解が得られる。また、アルゴリズム 2 と同じ条件で目的関数のみを変更し、航空機のインスタンスを用い、実行した結果が図 7 である。図 5 と図 7 を比較すると、交差数はそれぞれ 6 と 12 で

あり、総交差面積は図 5 は図 7 の約 63 倍であった。



交差数 12

図 7. 総交差面積最小化問題に対する出力

5 結論と今後の課題

ラベルの自動配置後に手で交差を解消させる手間を省くことを目的とした、交差数最小化問題と総交差面積最小化問題を提案した。これらの問題に対し、様々な条件で基本アルゴリズムを反復する発見的解法によって、実験を行なった。計算時間に余裕がある場合、反復数と実行回数を多くし、一番良い解を得ることが有効である。時間に限りがある場合、密集した点から配置をして反復すると良い結果を得ることができる。しかし、反復がほとんどできないような用途を考えると、配置する点をランダムに選んだ方が良い結果が出やすい。また、交差面積を小さくしようとすると交差数は増加するトレードオフの傾向が見られた。

今後の課題として、近似アルゴリズムの設計、交差解消の手間の評価、動的な点に対する実験が挙げられる。

謝辞

本研究を進めるにあたり、適切な御指導、御指摘をして頂きました中央大学理工学部情報工学科の今井桂子教授、森口昌樹助教に心から感謝致します。また、多くの助言をしてくれた学生各位に感謝致します。

参考文献

- [1] M. de Berg and D. H. P. Gerrits, "Approximation algorithms for free-label maximization," *Computational Geometry*, 45(4), 153–168, 2012.
- [2] M. de Berg and D. H. P. Gerrits, "Labeling moving points with a trade-off between label speed and label overlap," In H. Bodlaender and G. Italiano, editors, *Proceedings of the 21th Annual European Symposium on Algorithms, Lecture Notes in Computer Science*, 8125, 373–384, 2013.
- [3] M. Formann and F. Wagner, "A packing problem with applications to lettering of maps," In *Proceedings of the 7th Annual ACM Symposium on Computational Geometry*, 281–288, 1991.