

地理データベースにおける名前配置の後処理について*

2S-6

河北秀世 上林弥彦†
京都大学工学部‡

1 まえがき

地理データベースは、種々の地理情報をデータベースによって管理し、ユーザの要求に応じて適当な情報のみをデータベースから取り出し、地図の形でユーザに提示するものである。この場合、地理的な情報(地形情報、行政区画および市町村の情報、人口などの統計情報)は数値文字情報として管理し、それらから必要なものだけを取捨選択して地図として表示することにより、ユーザに対してさまざまなビューを提供できる。この点で地理データベースは、単に地図を管理するだけの「地図データベース」とは一線を画するものである。

従来、地図の合成の問題は「地図職人的」な観点で研究されてきた[2][4]。これは一枚の完成された地図を作り上げるという今まで人手に頼っていた作業をコンピュータにより自動化するというアプローチであり、こうして作られた地図は繰り返し利用されることになるが、それゆえ地図の品質は高いものが要求され、逆に地図を作るための時間については比較的軽視される結果となった。

一方、地理データベースの立場では、ユーザは自身の必要とする情報が取り出せるまで質問(Query)を繰り返すという状況を想定しているため、地図の合成についてはオンライン性を非常に重視している[1][3][5]。上林を中心としたグループでは既に、この立場から計算幾何学に基づいた文字配置アルゴリズムを提案している[3]。また最近では、更に高速性を追求したメッシュ基本型の文字配置アルゴリズムの開発に取り組んできた[5]。しかし高速性の追求の結果、計算幾何学による文字配置に比べて文字配置の品質が低下するという問題が生じた。

そこで本稿では、メッシュ基本型の文字配置の結果に対し、品質向上の「後処理」を付加することによって、Readabilityの高い地図を生成することを提案する。これによれば、高速性が要求される時には軽度の後処理、高品質が要求される時には重度の後処理と、要求される時間的制約に対して後処理の度合を変えることで対応でき、処理時間に見合った品質の地図が得られるという利点がある。

2 名前配置問題

地図において、ある地理図形(オブジェクト)に対する名前を配置する場所については次のような基準を設ける。

- 点図形に対しては、その点の近くに配置する。
- 線図形に対しては、線に沿って近いところに配置する。
- 領域図形に対しては、その領域の内部に配置する。ただし、領域が面積的に小さい場合は対応の分かりやすい位置に配置する。

ここで、地図における「第一名前配置問題」を、

名前を配置しようとするオブジェクトがN個あるとき、最大でいくつまで前出の基準を守って、しかも文字が互いに重ならないように名前を配置できるか。

という問題であると定義する。また、明らかにこれよりも難しい問題:「第二名前配置問題」を、

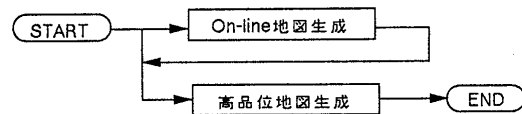


図1: 名前配置機能のサイクル

許容範囲内の Readability を実現するという制約を満たす範囲内で、「第一名前配置問題」同様に最大個数の名前を配置するにはどのように名前を配置すれば良いか。

と定義する。但し、Readability は別に定義する。ところが、各オブジェクトについて名前の配置場所の候補が二つであるという「第一名前配置問題」の簡単な場合ですら、NP 完全であることを示すことができ[1]、一般に「名前配置の問題」は、本質的に難しい問題であると言える。よって、Online 性を重視する立場からはこの問題を近似的に解くことで計算時間を短くする、というアプローチを採用している。

3 名前配置プロセスの分解

「地理データベース」の地図合成においては、最終的な地図にたどり着くまでの中間結果出力は処理の高速性が要求される「On-line 地図」であるが、最終結果の地図についてはその品質が重要視されることが多い。このことから、「地理データベース」における地図合成機能の利用形態は図1の様になると考えられる。

このとき On-line 地図としては、計算時間のことを考慮して「第一名前配置問題」の解を、高品位地図として「第二名前配置問題」の解を考えることにする。ここで、高品位地図については時間的な制約が少ないことから、従来の「地図職人的」なアプローチを簡略化して利用できると考えられる。そこで我々は On-line 地図生成の部分を中心に据え、以降「名前配置プロセス」とは On-line 地図生成のことを指すとする。

この On-line 地図生成の部分、とにかく計算時間を重視するので、問題を近似的に解くことにより計算時間の高速化をはかることが望ましい。我々は On-line 地図生成のプロセスを図2の様に2ステップに分けて考えている。

こうして得られる解は「第二名前配置問題」の解ではなく、あくまでも「第一名前配置問題」の解ではないが、「On-line 地図」の立場としては十分な品質であると考えられる。

3.1 第一名前配置問題の近似的解法

第一名前配置問題の解決には、「最大・最小配置候補数法」を利用する。これは、メッシュ基本型の名前配置アルゴリズムの一種である[3]。メッシュ基本型のアルゴリズムとは、文字の配置位置を平面上の格子点に限定し、配置位置に関する離散的な近似を行なうものである。しかし、配置位置を格子点に限定しただけでは問題のクラスが NP-完全から逃げることはできないと考

*Post-process of Name Placement Functions for Maps
†Hideyo KAWAKITA and Yahiko KAMBAYASHI
‡Kyoto University

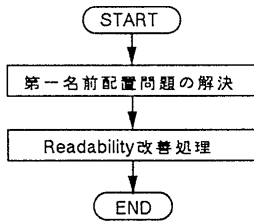


図 2: 名前配置プロセスの分解

えられるので、「最大・最小配置候補数法」では配置位置を格子点に限定した上で更に配置成功率の最大値を断念し、近似的にそれに近い値を目指している。

「最大・最小配置候補数法」では、各地理オブジェクトについての配置候補数を名前の配置位置決定に利用する。メッシュ基本型の解法では各地理オブジェクトに対する名前の配置位置候補は有限個数に限られ、しかもその候補のうちの幾つかは種々の原因(他の地理図形に邪魔されるなど)によって名前の配置はできなくなっているだろう。そのときの実現可能な名前の配置位置の数が配置候補数である。

また、「完全配置可能」という概念もここでは重要になる。ある地理オブジェクト G の名前の配置位置候補の中に、他の地理オブジェクト自身やその名前の配置位置候補と、地図平面上で交差するものが存在しない時、地理オブジェクト G は「完全配置可能である」といい、その様な配置位置は「完全配置可能位置」ということにする。

「最大・最小配置候補数法」のアルゴリズムを以下に示す。

最大・最小配置候補数法

- step 1 名前を配置したいオブジェクトをキューに登録する。
- step 2 キュー中の各オブジェクトの内、完全配置可能なオブジェクトがあれば、完全配置可能位置に名前を配置する。
名前を配置したオブジェクトについてはキューから削除する。
- step 3 キュー中のオブジェクトについて配置候補数を計算し、配置候補数がゼロのオブジェクトをキューから削除する。
- step 4 step 3 で計算した配置候補数の最小なオブジェクトをキューから取り出し、名前配置の対象とする。
- step 5 そのオブジェクトの名前配置位置候補の中で、「その位置へ名前を置いた時の、キュー中の他のオブジェクトの配置候補数の最小値」が、最大になるような配置位置を選ぶ。
- step 6 キューが尽きるまで step 2 から繰り返す。

3.2 名間配置の後処理:Readability の改善

先の「最大・最小配置候補数法」は、「第一名前配置問題」の近似的な解を与える。しかし、こうして得られた結果は、Readability の点からは決して良いものにはならないであろう。そこで、得られた結果に対して Readability の改善を行なうのがこの「後処理」である。

この様に名前配置の処理を二分化するのには次の理由による。すなわち、我々の対象としている地図は非常に密なものである。全ての地理図形に対して名前を配置できるとは限らず、そのため第一ステップとして名前の配置できるものとできないものを予め選別した後に、名前の配置できるものだけを適当な位置へ配置する必要がある。しかも、この様に名前配置プロセスを分解すると、処理時間の制約に対して「後処理」の割合を変えることによって柔軟に対応でき、処理時間に見合った品質の地図を得ることができる。

さて、この「後処理」においては Readability の改善として以下のようなことを行なう。

- 地理オブジェクトとその名前とが容易に関連づけられるように配置位置を調整する。
 - 点図形・線図形・領域図形それぞれについて統一した配置のポリシーを持つようにする。例えば、名前を地理オブジェクトの右側に配置するようにするなど。
 - 名前とそれの表す地理オブジェクトとの距離が、その名前と他のどの地理オブジェクトとの距離よりも小さくなるようにする。
 - 地理オブジェクトと名前の各組に対する表示色の塗り分け。
- 線図形欠損の制御 [5]

これらの処理も、やはり計算時間の観点から非常にシンプルなものにとどめる。

- step 1 「最大・最小配置候補数法」による解を初期配置とし、これに対し更に細かいメッシュを適用する。
- step 2 現段階での解において各地理オブジェクトの名前の配置可能な位置それぞれについて評価関数を計算し、その分散を求める。この分散の最も小さなものをキューから取り出して step 3 の対象とする。
- step 3 その名前配置可能領域内で、評価関数の最大となるところへ名前を再配置する。
- step 4 キューが尽きるまで step 2 から繰り返す。

ただし、ここでいう評価関数は Readability 改善の効果を表すもので、各 Readability 改善項基準における重み(これは目的とする地図に従って外部から与えられる)により設定される。

4 あとがき

本稿では、地理データベースにおける名前配置問題に関して、処理の高速性を重視した「On-line 地図」に対する名前配置問題を二つのステップに分けて解くことの見通しについて考察した。今後は、本稿で述べた解法の計算機上での実現および、従来の「地図職人的」解法との比較に基づいた評価を行なう予定である。

謝辞

種々御討論いただいた上林研究室の皆さんに感謝致します。

参考文献

- [1] 加藤 研児, 今井 浩: 2 または 3 自由度の文字配置問題の NP 完全性, 昭和 63 年度電気関係学会九州支部連合大会論文集, 1138 (1988).
- [2] C. B. Jones, Cartographic Name Placement with Prolog, *IEEE Computer Graphics & Applications*, Vol.9, No.5 (September 1989), pp.36-47.
- [3] N. Kojiro, K. Miura, H. Imai, Y. Kambayashi: Performance Evaluation of Automatic Name Placement Functions for Geographical Database Systems, *Proc. of the Second International Symposium on Database Systems for Advanced Applications* (April, 1991).
- [4] J.S.Doerschler and H. Freeman: A Rule-Based System for Dense-Map Name Placement, *Communications of the ACM* (January, 1992), pp.68-79.
- [5] 上林 弥彦, 河北 秀世: 地図の名前配置問題における線図形欠損の取り扱いについて, 平成 4 年度前期情報処理学会全国大会講演論文集, 4-149 (1992).