アルゴリズム 62－6
（1998. 5. 20）

多次元空間での最近傍点探索アルゴリズムの実験的解析と拡張

遠藤雅也, 今井浩
東京大学大学院理学系研究科情報科学専攻

1998年4月27日

あらまし

高次元空間での最近傍点検索問題には、マルチメディア情報検索、パターン認識や統計解析などの応用分野があり、問題のサイズはより大規模化しており、この問題を高速に解くことは非常に重要である。Clarksonが近接履歴グラフを逐次的に、概念的な点集合を用い近似的に構成し、近似的に検索をする方法を提案している。

本論文では、Clarksonの方法を一部変更した方法を提案する。具体的には、近接履歴グラフを更新する際に、これまでの履歴構造を利用するようにした。これにより、ある条件の下、検索時間を短縮することができた。また、この方法の解析および実装をし、一様分布と頻度ベクトルデータを用い、Clarksonの方法との比較を行なう。

Experimental Analysis and Extentions of Algorithms for Nearest Neighbor Search in High Dimensions

ENDO Masaya, IMAI Hiroshi
Department of Information Science, University of Tokyo

Abstract

The nearest neighbor problem for high dimensional space has applications in multimedia information retrieval, pattern recognition, and statistical data analysis. It is important to solve this problem quickly. Clarkson gives an approximate algorithm constructing a Voronoi diagram incrementally and approximately in conceptual setting.

In this thesis, we modify the algorithm. Clarkson's algorithm computes strictly nearest neighbors in constructing a Voronoi diagram incrementally. However our algorithm computes approximately using a present data structure. By this modification, we reduce a query time under a specific condition. Besides, we analyze and implement our algorithm and compare it to Clarkson's one by using uniformly distributed data and data from real world.

1 はじめに

最近傍点探索問題とは、物体を空間上の点で表したサイトと呼ばれるものの集合 S と点 q が与えられた時、点 q に最も近い S のサイト p を求める問題である。

最近傍点探索問題は、文章検索、画像検索、パターン認識、統計データ解析、データ圧縮、マルチメディアデータベースなど、幅広い応用範囲を持っている。この問題への必要性が高いのは、検索や比較など基本的な操作をする際の有効性が高いからである。

検索や比較を行なうためには、物体を測度空間内のサイトとして表現する。そして、サイトどうしが近ければ、元の物体どうしが似ていると考える。例えば、文章検索の場合、文章を測度空間のサイトにマッピングし、そこで検索する。具体的には、文章の集合から文章の特徴をよく表現するとみなせるようなキーワードを抜きだし、キーワード数を軸とするような特徴空間に文章をマッピングする。この場合、文章の非類似度をサイトの近さとしてみなす。もし、ある文章と類似した文章を探したければ、一度特徴空間に落し、そのなかで、近いサイトを検索すればよい。

一般に、キーワード数は数百と、非常に大きい。キーワード数がそのまま次元数になるため、数百次元での最近傍点探索問題を考えなければならない。Latent semantic indexing [DDF+90] のように、キーワードの空間ではなく、意味空間にマッピングし次元数を減らす方法もあるが、それでもなお次元数は大きい。そのため、高次元空間での最近傍点を高速に検索する必要が増している。もちろん、これまでに多くの方法が研究されてきたが、低次元空間でのみ有効であり、高次元空間で単純な方法より劇的に高速に解く方法はいまだ知られていない。

高次元空間の最近傍点探索問題に関して多くの研究がなされている。そのうちのいくつかを紹介する。

Indyk、Motwani、Raghavan、Vempala [IMRV97] は、locality preserving hashing を利用する方法を提案している。この方法は、10 から 20 次元までの空間では、最近傍点を有効に探索することが出来る。ハッシングは、最近傍点探索には向かないと考えられてきたので、この方法は興味深い。

Katayama と Satoh [KS97] は、SR-tree というデータ構造を提案した。S と R はそれぞれ sphere と rectangle のかしら文字である。SR-tree はデータベースでよく知られている SS-tree や R*-tree を改良したものである。しかし、一様分布の点集合を扱う場合、有効なのは約 30 次元までで、それ以上では有効でない。

Kleinberg [Kle97] は、ランダムに選ばれた直線上に点集合を投影する近似的な方法を提案した。点集合の大きさを n、空間の次元を d としたとき、この方法の検索時間は $O(n + d\log^3 n)$、前処理にかかる時間は $O(d^2 n)$、メモリ領域は $O(dn)$ である。

Clarkson [Cla97] は、分割統治法を取り入れた厳密な探索方法と、スキップリストに似た隣接履歴グラフという構造を近似的に構成する方法を提案した。後者の方法は、質問点の候補集合 Q を入力として必要とする。前者の検索時間は $(\log n)^{O(\log\log\Upsilon)}$、前処理の時間は $O(n)(\log n)^{O(\log\log\Upsilon)}$ である。後者の検索時間は $O(\log n)\log\Upsilon$、前処理の時間は Q も含めた入力サイズに対して、前者のと等しい。メモリ領域は、$O(n)K\log\Upsilon$ である。ただし、K は構成のパラメータで、Υ は、与えられた点集合に依存する値である。

本論文では、Clarkson の後者の方法を変更した方法を提案する。Clarkson の方法は、近

接履歴グラフを更新する際に最近傍点を厳密に計算していた。本論文では、最近傍点をこれまでの履歴構造を利用し近似的に最近傍点を探索するようにした。これにより、無駄な構造を減らすことが出来た。

2 準備

この章では、いくつかの定義を行ない、概要を述べる。

2.1 定義

V をある集合とし、d を V 上の距離測度とする。ここでは、有限測度空間 (V,d) を考える。すなわち、任意の V の要素 a,b,c に対して、$d(a,a)=0$, $d(a,b)=d(b,a)$、および $d(a,c)\leq d(a,b)+d(b,c)$ が成り立つと仮定する。

定義：最近傍点

測度空間 (Z,d) と Z 上の点 q が与えられているとする。このとき、Z に対して q の最近傍点 s とは、すべての Z 上の t に対して、$d(s,q)\leq d(t,q)$ となる点のことである。

定義：γ- 最近傍点

測度空間 (Z,d) と Z 上の点 q が与えられているとし、q の最近傍点を t とする。このとき、Z での q の γ- 最近傍点 s とは、$d(s,q)\leq\gamma d(t,q)$ となる点のことである。

2.2 アウトライン

3 章で Clarkson の後者のアルゴリズムについて述べ、4 章でそれをもとに変更した方法を提案する。次に、5 章で実験結果について述べ、最後に 6 章でまとめを書く。

3 Clarkson の方法

この章では、Clarkson の後者の方法について説明する。最初に、近接履歴グラフの構成を、次にそれを利用した最近傍点の探索方法を説明する。

3.1 近接履歴グラフの構成

Clarkson の方法では、サイト集合 S と質問点の候補点集合 Q が与えられたとして、近接履歴グラフ $M(S,Q)$ と呼ばれる構造を逐次的に近似的に構成する。Q と実際の質問点は同じ分布を持っている必要がある。このグラフは、ある条件の下、Voronoi 図の隣接関係を保持するグラフとみなせる。

サイト集合 S と質問点の候補点集合 Q、およびパラメータ K,γ が与えられているとする。n を S の要素数とし、S の各要素に対し、$p_1,p_2,...,p_n$ とランダムに順序をつけ、i 番目までの集合を R_i とおくことにする。$1\leq i\leq n$ に対し、Q_i を、$|Q_i|=Ki$ かつ、Q のランダムな部分集合となるように定める。

空リスト $D_j(1 \leq j \leq n)$ を用意する。p_i を加えるごとに、質問点の候補点集合 Q_i を利用し、近似的に近接履歴グラフを構成していく。p_i を R_i に加えるときに、まず、Q_i から q を 1 つ取り出し、

- R_{i-1} においての q の最近傍点 p_j を求め、
- サイト p_i が q の γ- 最近傍点であるか調べる。

もしそうであればリスト D_j に p_i を加える。Q_i の残りすべての要素についても同様のことを行なう。以下、これを p_n まで繰り返していく。構成はこれまでである。

パラメータ K が十分に大きく、$\gamma = 1$ であれば、Voronoi 図の隣接関係を構成することにほぼ等しい。なぜなら、p_i を R_i に加える際に、R_i の Voronoi 図において、p_i の領域と、ある p_j の領域が互いに接していれば、リスト D_j に p_i を加えることに相当するからである。しかし、新しいサイトを加えることによってそれまで互いに接していた領域どうしが離れることになっても、グラフの枝を消すことはないので、Voronoi 図と双対な関係にある Delaunay グラフより大きいグラフを構成することになる。

3.2 最近傍点の探索方法

サイト集合 S の近接履歴グラフが構成されたときに、点 q の最近傍点の求め方を説明する。

p_1 を解の候補とする。D_1 の先頭から p_i を取り出す。もし、p_i が p_1 より q に近ければ、p_i を新たな解の候補とし D_i を調べる。そうでなければ、D_1 から次々と要素を取り出し、p_1 と比べていく。もし、リストが空になったら、その時点での候補をこのアルゴリズムの解として返す。

4 変更点

前の章で説明した Clarkson の方法に対して、変更を 2 点提案する。1 点目は、近接履歴グラフを構成する方法の一部を変更し、2 点目は、質問点の候補点集合 Q の生成方法を提案する。

4.1 Clarkson-Endo 法

近接履歴グラフを構成する方法の変更点を説明する。

もとの方法では、p_i を R_i に加え $M(R_i, Q)$ を構成するときに、R_{i-1} における Q の要素 q の最近傍点を厳密に探索する。しかし、提案する方法では最近傍点を探索するとき、その時点までに構成された $M(R_{i-1}, Q)$ を利用して探索する。

これにより、最近傍点を近似的に探索することになるが、実は、$M(S, Q)$ の無駄な構造を減らすことが出来る。

4.2 中点法

ここでは、質問の候補点 Q を生成する方法を提案する。

p_i を加えるときの Q_i を、p_i と各 $p_j(1 \leq j < i)$ との中点の集合となるように定める。

つまり、Q は、S の中点集合となる。

これにより、Q_i は Q からのランダム抽出ではなくなる。しかし、もともと Q 自体も、質問点の候補点集合であり、分布には偏りがあるとされている。また、例えばそれは文章の頻度ベクトルのように、S の偏りと類似のものと期待される。したがって、このように Q を S の中点集合ととれば、Q の分布は S の分布をある程度は反映したものとなり、Q としては、適当であろうと推測される。

また、ユークリッド空間上で Voronoi 図を構成するという観点に立てば、この方法は、新たなサイトを加えることにより Q_i の要素の中で、最近傍点が変化するものを探している、ともいうことができる。したがって、Q_i を p_i の周辺に限定するのも、適当であろうと思われる。

しかし、集合 Q を中点に限定した方法では、当然条件が厳しくなる。これから、Gabriel グラフとの関係をふまえて、これについて述べる。

以下、K は十分に大きく、$\gamma = 1$ であると仮定する。この場合、もともとの方法では、サイトを加えるさいに Voronoi 図の隣接関係を調べることになる。これは、Voronoi 図と双対である Delaunay グラフの性質を利用しているということである。しかし中点を利用した方法では、Delaunay グラフの性質ではなく、Gabriel グラフの性質を利用している。一般に、Gabriel グラフは Delaunay グラフより集合の意味で小さい。したがって、この中点を利用する方法によって構成されるグラフも、実際の近接履歴グラフの部分グラフである。これが最近傍点探索にどの程度の影響がでるかは、実験によって確認する。

5 実験

この章では、Clarkson の方法と変更した方法について実験し、比較する。始めは、利用したデータセットについて述べる。次に、実験結果について述べる。

5.1 使用したデータセット

実験において、以下のデータセットを利用した。サイトだけでなく、候補点集合および、実際の質問点もこれらを利用している。

- 一様分布
- 頻度ベクトル（正規化したもの、重みを掛けたもの）

ただし、一様分布は、各成分を $[0, 1)$ の範囲でランダム発生させたものである。また、頻度ベクトルは、日本経済新聞から、50000 記事を取り出し、100 最頻出キーワードを抽出することによりベクトル化したものをさらに特異値分解し、次元を減少させたものである。具体的には、ユークリッドノルムで正規化した頻度ベクトルを行ベクトルとして並べた行列

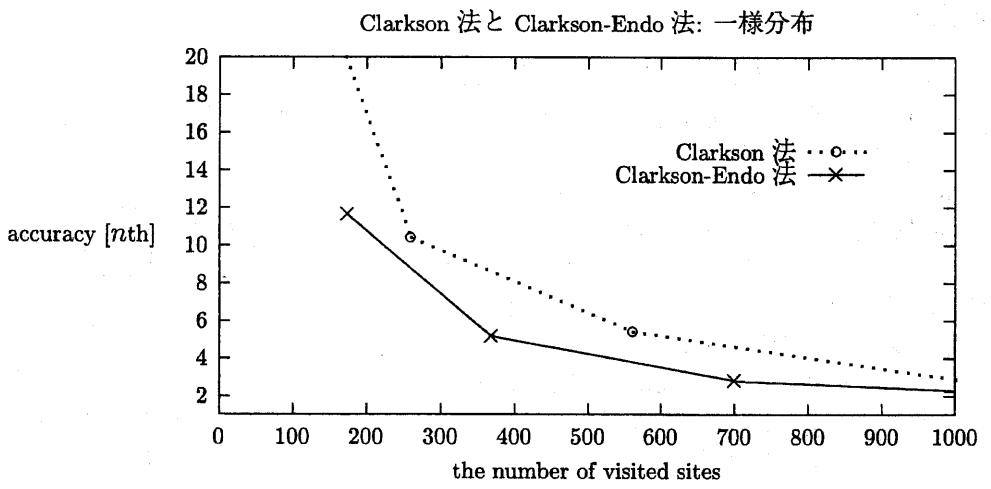

図 1: Clarkson 法と Clarkson-Endo 法の違い：一様分布

A を DST に特異値分解した。そして A の第 k 特異値までの近似行列 $A_k = D_k S_k T_k$ に対し、D_k の各行ベクトルをユークリッドノルムで正規化したものと、特異値を重みとしてかけた $D_k S_k$ の 2 通りで実験を行なった。

また、距離としてユークリッド距離を使用した。頻度ベクトルは、ベクトルどうしの角度 θ に対し、$1-\cos\theta$ で近さを測るのが普通だが、正規化した場合には以下のようになり、大小関係を比較する上では、ユークリッド距離と等しくなる。

2 つのベクトルを p, q とし、これらのなす角を θ とおく。

$$\frac{1}{2}\left|\frac{p}{|p|}-\frac{q}{|q|}\right|^2 = \frac{1}{2}\left(\frac{|p|^2}{|p|^2}+\frac{|q|^2}{|q|^2}-2\frac{p\cdot q}{|p||q|}\right) = 1-\cos\theta$$

5.2 結果・考察

図 1は、Clarkson 法と Clarkson-Endo 法の性能の違いを示している。使用したデータは、20 次元 10000 サイトの一様分布データセットである。近接履歴グラフをそれぞれの方法で構成し、1000 回質問をしたときの平均である。横軸は訪れたサイト数の平均個数、縦軸は各々の方法が返したサイトが実際は何番目であったかを示している。図は、パラメータ γ を変化させたときの様子である。図から、同じ探索回数で Clarkson-Endo 法の方がより良い解を与えていることがわかる。

これは、方法を変えたことにより、行き止まりのサイトが減少したからである。しかし、孤立したサイトは増加している可能性があるが、これが与える影響はよくわからない。

図 2は、Clarkson 法と中点法の性能の違いを示している。使用したデータは、20 次元 1000 サイトの一様分布データセットである。計算回数が同程度になるように K の値を設定

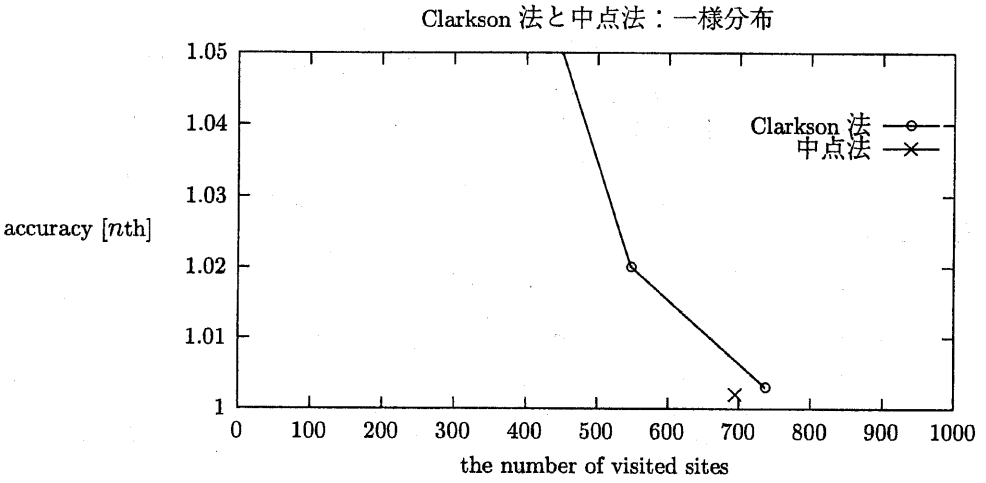

図 2: 中点法：一様分布

した。同様に、パラメータ γ を変化させたときの様子であるが、中点法は γ の影響を受けないので図のように一点になっている。図から、同じ探索回数で中点法の方がより良い解を与えていることがわかる。

図 3は、データセットによる探索回数の違いを示している。サイト数は 1000 サイトである。一様分布では、30 次元程度でほとんどすべてのサイトを探索しているが、頻度ベクトルは、40 次元で正規化したもので約 60 %、重みをつけたもので約 25 %である。これから、実際のデータは偏りをもっていて、検索をするのに都合が良いことがわかる。

また、Katayama、Satoh [KS97] において、一様分布 100000 サイトで次元と探索サイト数の関係を調べている。この結果においても、約 30 次元で限界がきている。サイト数に 100 倍の差があるが、次元に換算じて数次元の差である。その若干の分、Clarkson 法の方がよいと言えるが、今のところ、次元数に関してこの程度がこれらの方法での限界であると言える。

6 まとめ

Clarkson の方法を変更することにより、探索性能を若干改善することができた。また、候補点集合の生成法を提案し、有効性も確かめた。

頻度ベクトルを用いることにより、実際のデータには偏りがあり、それが検索回数を減少させることが実験的に確かめられた。

今後は、プログラムを改善し、より大規模なデータでも検索できるようにしたい。

また、Clarkson の方法は、測度空間を仮定している。しかし、実際はその条件を必ずしも使っているわけではない。例えば、統計の分野でダイバージェンスという量がある。どの

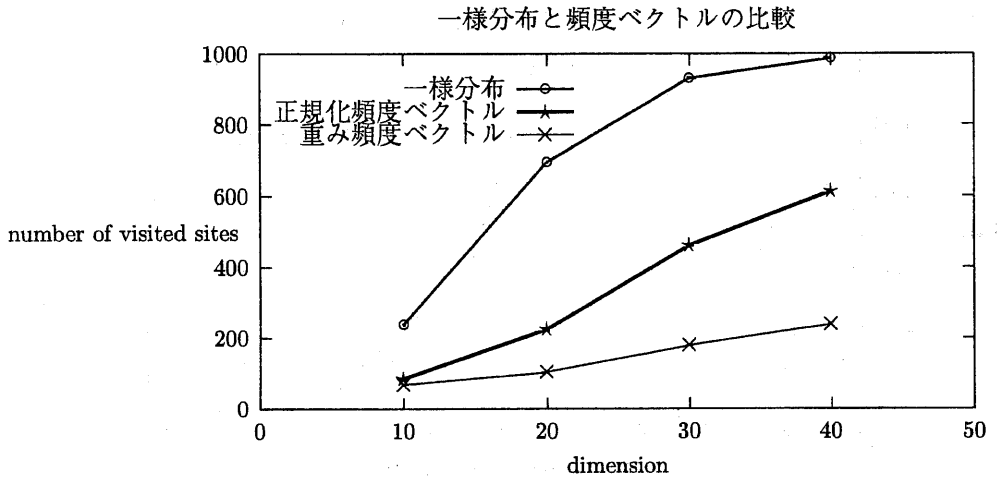

図 3: データセットによる違い

ような量が距離測度として適切であるか、実験的に確かめてみる必要がある。

参考文献

[Cla97] Kenneth L. Clarkson. Nearest neighbor queries in metric spaces. In *Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing*, pages 609–617, El Paso, Texas, May 1997.

[DDF+90] Scott Deerwester, Susan Dumais, Goerge Furnas, Thomas Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

[IMRV97] Piotr Indyk, Rajeev Motwani, Prabhakar Raghavan, and Santosh Vempala. Locality-preserving hashing in multidimensional spaces. In *Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing*, pages 618–625, El Paso, Texas, May 1997.

[Kle97] Jon M. Kleinberg. Two algorithms for nearest-neighbor search in high dimensions. In *Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing*, pages 599–608, El Paso, Texas, May 1997.

[KS97] Norio Katayama and Shin'ichi Satoh. The SR-tree: An index structure for high-dimensional nearest neighbor queries. *SIGMOD Record (ACM Special Interest Group on Management of Data)*, 26(2):369–380, 1997.