2005－AL－103（11）
2005／11／11

任意の L_p 距離関数による検索が可能な索引構造

大西建輔[†]　小早川倫広[††]　木村彰宏[††,†††]
星守[††]　大森匡[††]

データベースに対し, 高速な ε 近傍検索を行うために, これまで様々な索引構造が提案されてきた. Yi らは, 1 つの索引構造で多くの距離モデルに対応する多モデル対応という概念を提案し, L_1 距離関数と L_∞ 距離関数での検索を可能とする距離変換規則を提案した. 我々は Yi らの距離変換規則を拡張し, L_v 距離関数を基に構築された索引空間に対し, 任意の L_p 距離関数での検索を可能とする距離変換規則を提案した. これらの研究は, ある距離関数での検索半径を変換規則により索引構造の構築時の距離関数に変換することで, 任意の L_p 距離での ε 近傍検索を可能としている.

本稿では, 検索半径の変換を行わず, 1 つの索引構造だけで任意の L_p 距離関数による検索を可能とする索引構造 (mm-GNAT) の提案を行う. この索引構造は, GNAT(Geometric Near-neighbor Access Tree) を基にしている. まず, 我々はこの索引構造で, 任意の L_p 距離関数による検索が可能であることを示す. また, 人工データと楽曲データを用いた計算機実験により, 任意の L_p 距離関数による検索が実現されていることを確認し, 有効性の評価を行った.

Index structure for arbitrary L_p norm

Kensuke Onishi ,[†] Michihiro Kobayakawa ,[††]
Akihiro Kimura ,[††,†††] Mamoru Hoshi [††] and Tadashi Ohmori [††]

For fast ε-similarity search, various index structures have been proposed. Yi et al. proposed a concept *multi-modality support*, which is an index structure for arbitrary distance model, and suggested a distance conversion rule by which ε-similarity search by L_1 norm and L_∞ norm can be realized. We proposed an extended distance conversion rule by which ε-similarity search of arbitrary L_p norm by a L_v based index structure can be realized. In these investigations search radius for a norm is converted into another norm using the conversion rule.

In this paper, we propose an index structure for arbitrary L_p norm, called mm-GNAT(multi-modality GNAT), without distance conversion rule. The index structure is based on GNAT(Geometric Near-neighbor Access Tree) which is a kind of geometric clustering for data set. We show that ε-similarity search by arbitrary L_p norm is realized by mm-GNAT. In addition, we perform search experiments by mm-GNAT on artificial data and music data. From the experiments, we can conclude that the search by arbitrary L_p norm is realized and the index structure has good search performance for queries.

1. はじめに

マルチメディアデータや時系列データ等を検索する場合, 検索目的に応じ, 元のデータから様々な特徴量を抽出し, 特徴量の空間での検索を行う. 特徴量の空間は, 多次元ベクトルの空間として表現されることが多い. また, 特徴量を用いた検索では, 必ずしも問合せと同一のデータが存在するとは限らないため, 類似検索を行うことが多い. 類似検索には, 問合せ点に近い k 個の点を出力する k 近傍検索や問合せ点から半径 ε 以内の距離にある点を全て出力する ε 近傍検索などがある. 本稿では, ε 近傍検索のためのデータ構造に関する研究を行う.

ε 近傍検索を高速に行うための様々なデータ構造 (索引構造と呼ぶ) として, R 木[5], SS 木[12], SR 木[7], VP 木[14], M 木[4], GNAT[2] 等が提案されてきた. これら以外にも多くの研究があり, それらは Böhm らによるサーベイ[1], Chávez らによるサーベイ[3] などによくまとめられている. これらの索引構造は, 距離に基づいてデータ集合を部分集合に分割し, 検索の高速化

† 東海大学理学部情報数理学科
Department of Mathematical Science, Tokai University
†† 電気通信大学大学院情報システム学研究科
Graduated School of Information Systems, The University of Electro-Communications
††† NEC フィールディング
NEC Feilding Co., Ltd.

を行っている．その基本となる考えは, 検索の出力に含まれる点を含まない部分集合を決定し, その部分集合を検索対象から除くということである．検索する必要のない部分集合を決定するために, 距離が用いられる．すなわち, これまで提案されてきた索引構造は多くが特徴量空間に導入された距離関数に依存する構造となっている．ε 近傍検索は問合せ点を中心とする ε 球に入っている点を全て出力するので, この ε 球と部分集合の間に交わりがなければ, その部分集合は検索する必要がなくなるのである．つまり, 各部分集合に対し, 距離を基にした索引構造を持たせることで, 高速な ε 近傍検索を行うことが可能となる．

Yi らは, 多モデル対応 (multi-modality support) と呼ばれる概念を提案した[13]．この概念は, 特徴量空間には, 様々な類似度モデルが導入されている可能性があるが, 各々の類似度モデルに対し, 対応が可能な索引構造を構築するという考えである．例えば, 類似度モデルを用いてパーソナライゼーションを行うことを考えてみよう．類似度モデルを各個人に対応できるようにするために, 個人毎に類似度モデルを基にした索引構造を構築するのは非常に手間のかかる作業であり, サーバ上に個人毎の索引構造を持たせるには, 非常に多くのコストが必要となる．このようなパーソナライゼーションを実現する場合に, 様々な類似度モデルに対して, 検索の可能な索引構造を構築することができれば, 多くの索引構造を構築せず, サーバにコストをかける必要がなくなるのである．

Yi らは, 類似度モデルを L_p 距離関数 (ミンコフスキー距離関数) に限定し, 任意の L_p 距離関数での検索を可能とする方法を提案した[13]．これは, L_p 距離関数と L_2 距離関数の間の関係式 (距離変換規則と呼ぶ) を提案し, L_1 距離関数と L_∞ 距離関数による検索実験を行った．Lee らはこの方法を Minimum distance という距離に対して適用した[8]．これら手法では, 検索半径を拡大することで検索を可能としている．そのため, 検索を行う L_p 距離によっては, かなり大きな領域を検索することになってしまう．そこで, 我々は任意の L_p 距離関数による ε 近傍検索を実現するための最小の検索半径を与える変換規則を提案した[15]．

これまでの研究では, 検索半径を拡大することで, L_p 距離関数による ε 近傍検索を実現してきた．我々は, 距離変換規則を用いずに索引構造を工夫するだけで, 任意の L_p 距離関数による ε 近傍検索を実現することを目的とする．このような手法では, 検索半径を拡大することなく検索を行うことができ, ミンコフスキー距離だけでなくより多くの距離関数に対応が可能な手法となる．

本論文の構成を以下に示す．2 章では, 関連研究として, Yi らの手法[13], 木村らの手法[15], GNAT[2] について説明を行う．3 章では, 任意の L_p 距離での検索が可能な索引構造である mm-GNAT を提案し, この索引構造により, 距離変換規則を利用しない任意の L_p 距離による ε 近傍検索が可能であることを示す．4 章では, mm-GNAT に対する計算機実験の概要を述べ, 5 章で, その実験結果に関する議論を行う．

2. 関 連 研 究

まず, 部分集合への分割を基にした検索の枠組みについて述べる．検索は基本的に次の 2 つのステップにより, 実現される．

ステップ 1　検索する必要の無い部分集合 (不必要集合と呼ぶ) の決定

ステップ 2　検索する必要のある部分集合 (必要集合と呼ぶ) のすべての要素に対する距離計算

ステップ 1 において, 不必要集合を決定することにより, ステップ 2 での距離計算回数が減り, 検索のコストが減ることになる．

以下では, Yi らの手法[13], 木村らの手法[15], Brin による GNAT[2] について述べる．

2.1 Yi らの手法

Yi らは次に示す距離変換規則を導き, 不必要集合の決定を L_2 距離での検索に帰着させた:

$$\mathrm{dist}_1(\boldsymbol{x},\boldsymbol{y}) \le \mathrm{dist}_2(\boldsymbol{x},\boldsymbol{y}),\quad \mathrm{dist}_\infty(\boldsymbol{x},\boldsymbol{y}) \le d^{\frac{1}{2}}\cdot\mathrm{dist}_2(\boldsymbol{x},\boldsymbol{y})$$

ただし, $\mathrm{dist}_p(\boldsymbol{x},\boldsymbol{y})$ は d 次元ベクトル $\boldsymbol{x},\boldsymbol{y}$ に対して, 次式で定義される距離関数であり, L_p 距離と呼ばれている:

$$\mathrm{dist}_p(\boldsymbol{x},\boldsymbol{y}) = \begin{cases} \left\{\sum_{i=1}^{d}|x_i-y_i|^p\right\}^{1/p} & (p=1,2,\cdots) \\ \max_{i=1}^{d}|x_i-y_i| & (p=\infty) \end{cases}$$

また, Yi らは, 明示的には示していないが, 任意の L_p 距離関数 ($p \ge 2$) に対して,

$$\mathrm{dist}_p(\boldsymbol{x},\boldsymbol{y}) \le d^{\frac{1}{2}}\cdot\mathrm{dist}_2(\boldsymbol{x},\boldsymbol{y})$$

という関係が成立することを知っていたと思われる．

2.2 木村らの手法

木村らは, Yi らの研究を発展させ, 次の距離変換規則を示した．

$$\mathrm{dist}_p(\boldsymbol{x},\boldsymbol{y}) \le \mathrm{dist}_v(\boldsymbol{x},\boldsymbol{y}) \qquad (1 \le p < v)$$
$$\mathrm{dist}_p(\boldsymbol{x},\boldsymbol{y}) \le d^{\frac{1}{v}-\frac{1}{p}}\cdot\mathrm{dist}_v(\boldsymbol{x},\boldsymbol{y}) \qquad (v \le p \le \infty)$$

この規則により, L_p 距離関数による問い合わせを L_v 距離関数を元にして構築された索引空間での問い合わせに変換することができる．　この変換規則で $p=1$, $v=2$ または, $p=\infty, v=2$ とすると, Yi らの変換規則と

なる．よって，この手法は Yi らの手法を一般化していることになる．また，L_p 距離に対する漏れのない ε 近傍検索をする場合，右辺の距離関数の係数は，最小の値となっている．

2.3 GNAT

Brin は，GNAT(Geometric Near-neighbor Access Tree) と呼ばれる索引構造を提案している．GNAT とは，元のデータ集合の部分集合と，部分集合の代表点 (分割点) と他の部分集合の存在範囲 (代表点と部分集合の間の距離の最小値と最大値の組) を保持するデータ構造である．このデータ構造は任意の距離関数に対して，構築することが可能である．

GNAT を構築する際には，まず分割点 (SP_i と記す) と呼ばれる点集合を決定する．次に，すべてのデータ点に対し，各分割点との距離を計算し，最も近い分割点をデータ点が属する分割点とする．これは，分割点の集合に対するボロノイ図を構築し，それぞれの分割点に対応するボロノイ領域に含まれるデータ点を 1 つのクラスタ (D_{SP_j} と書く) とすることと等価である．その上で，各クラスタの点と分割点の間の距離を計算し，その最大値と最小値を保持する．つまり，分割点 SP_i からみて，クラスタ D_{SP_j} がどれだけの距離の範囲に存在するかを索引構造内に，保持していることになる．以下では，この範囲をクラスタ D_{SP_j} の存在範囲と呼ぶことにする．この存在範囲を式で表現すると，次で表される．

$$\mathrm{range}(SP_i, D_{SP_j}) = \left[\min_{\boldsymbol{x} \in D_{SP_j}} \mathrm{dist}(SP_i, \boldsymbol{x}),\ \max_{\boldsymbol{x} \in D_{SP_j}} \mathrm{dist}(SP_i, \boldsymbol{x}) \right]$$

問合せ点 $\boldsymbol{q}$ での ε 近傍検索を行う場合には，SP_i から，問合せ点を中心とする ε 球がどの範囲にあるかを示す次の区間

$$[\mathrm{dist}(SP_i, \boldsymbol{q}) - \varepsilon, \mathrm{dist}(SP_i, \boldsymbol{q}) + \varepsilon]$$

と先の存在範囲 $\mathrm{range}(SP_i, D_{SP_j})$ の交差があるかどうかを判定し，交差がない場合は，クラスタ D_{SP_j} を不必要集合と判定することができる．これをすべての分割点 (クラスタ) の組に適用することで，不必要集合となるクラスタを決定することができる．ここまでが先に述べた枠組みのステップ 1 に相当する．

その上で，残ったクラスタのデータ点に対して，ステップ 2 を適用することになる．

ここで，ある距離関数 dist で構築された GNAT に対して，他の距離関数 dist′ で ε 近傍検索を行う場合について考える．このとき，枠組みのステップ 1 において，必要集合であるクラスタを，不必要集合であると判定することがある．例えば，L_1 距離関数による

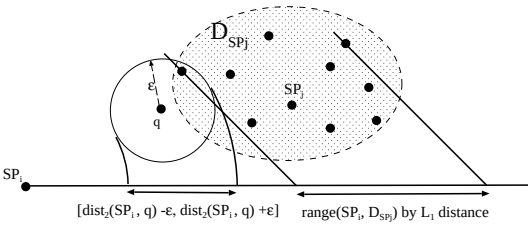

図 1 必要集合が不必要集合と判定される場合
Fig. 1 A case of regarding necessarry set as unnecessary

GNAT を構築した場合，その索引構造に保持されるのは，L_1 距離による最小値と最大値となる．このとき，L_2 距離 ε での検索を行おうとすると，同一点に対する距離としては，L_1 距離が L_2 距離よりも大きいか等しいという関係が存在するため，L_1 距離で構築された存在範囲における最小値で，必要集合となるべき集合が不必要集合と判定される可能性がある (図 1 参照)．また，L_∞ 距離で検索を行う場合にも，同様の現象がおこる可能性がある．

3. mm-GNAT

本節では，任意の L_p 距離関数での検索を可能とする mm-GNAT(multi-modality GNAT) について提案を行う．

前節で述べた現象は，検索を行う距離関数と索引構造を構築した距離関数が違うために起こる．我々は，必要集合が不必要集合と判定されないために，GNAT が保持する存在範囲を拡大するという方針を考えた．本稿では，用いる距離としては L_p 距離関数だけに絞り，1 つの索引構造構造で，任意の L_p 距離関数での検索が可能な索引構造を提案する．

L_p 距離関数では，次の関係式が成立する．

補題 1 $\boldsymbol{x}, \boldsymbol{y}$ を d 次元ベクトルとする．このとき，$\mathrm{dist}_p(\boldsymbol{x}, \boldsymbol{y})$ に対して，次の関係が成立する．

$$\mathrm{dist}_\infty(\boldsymbol{x}, \boldsymbol{y}) \le \mathrm{dist}_p(\boldsymbol{x}, \boldsymbol{y}) \le \mathrm{dist}_1(\boldsymbol{x}, \boldsymbol{y})$$

証明: $\boldsymbol{x} = (x_1, \cdots, x_d), \boldsymbol{y} = (y_1, \cdots, y_d)$ とする．$z_i = |x_i - y_i|\,(i = 1, \ldots, d)$ とすると，$z_i \ge 0$ となる．

(I) $\mathrm{dist}_\infty(\boldsymbol{x}, \boldsymbol{y}) \le \mathrm{dist}_p(\boldsymbol{x}, \boldsymbol{y})$ の証明

一般性を失わず，$z_1 = \max_i z_i$ とすることができる．

$$\begin{aligned} \{\mathrm{dist}_p(\boldsymbol{x}, \boldsymbol{y})\}^p - \{\mathrm{dist}_\infty(\boldsymbol{x}, \boldsymbol{y})\}^p &= ({z_1}^p + {z_2}^p + \cdots + {z_d}^p) - {z_1}^p \\ &= {z_2}^p + \cdots + {z_d}^p \\ &\ge 0 \end{aligned}$$

$z_i \ge 0$ なので，上式が成立する．よって，

$$\mathrm{dist}_\infty(\boldsymbol{x}, \boldsymbol{y}) \le \mathrm{dist}_p(\boldsymbol{x}, \boldsymbol{y}).$$

(II) $\mathrm{dist}_p(\boldsymbol{x}, \boldsymbol{y}) \le \mathrm{dist}_1(\boldsymbol{x}, \boldsymbol{y})$ の証明

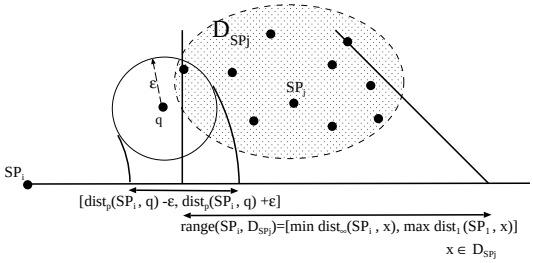

図 2 mm-GNAT での交差判定
Fig. 2 Intersection test on mm-GNAT

$$
\begin{aligned}
&\{\mathrm{dist}_1(\boldsymbol{x},\boldsymbol{y})\}^p - \{\mathrm{dist}_p(\boldsymbol{x},\boldsymbol{y})\}^p \\
&= \sum_{\substack{k_1,\cdots,k_d\geq 0,\\ k_1+\cdots+k_d=p}} \binom{p}{k_1\cdots k_d} z_1^{k_1}\cdots z_d^{k_d} - (z_1^p+\cdots+z_d^p) \\
&= \sum_{\substack{k_1,\cdots,k_d\geq 0, k_i\neq p,\\ k_1+\cdots+k_d=p}} \binom{p}{k_1\cdots k_d} z_1^{k_1}\cdots z_d^{k_d} \\
&\geq 0
\end{aligned}
$$

多項係数と各 z_i は正の値を取るため,

$$\mathrm{dist}_1(\boldsymbol{x},\boldsymbol{y}) \geq \mathrm{dist}_p(\boldsymbol{x},\boldsymbol{y})$$

が成立する. よって, 上式が証明できた. □

補題 1 を用いて, 存在範囲の拡大を行う. これは, 存在範囲の最小値を L_∞ 距離関数でとり, 最大値を L_1 距離関数で保持することで, 任意の L_p 距離関数による検索を可能とする. つまり, クラスタの存在範囲を次の式で再定義することになる (図 2 参照).

$$
\begin{aligned}
&\mathrm{range}(SP_i, D_{SP_j}) \\
&= \left[\min_{\boldsymbol{x}\in D_{SP_j}} \mathrm{dist}_\infty(SP_i,\boldsymbol{x}), \max_{\boldsymbol{x}\in D_{SP_j}} \mathrm{dist}_1(SP_i,\boldsymbol{x}) \right]
\end{aligned}
\tag{1}
$$

すなわち, mm-GNAT とは, 元のデータ集合を分割したクラスタ, それぞれのクラスタの代表点, 存在範囲 (1) で定義される索引構造である.

次に, mm-GNAT を用いて, 任意の L_p 距離関数による検索が可能であることを示す. これは, 必要集合が不必要集合と判定されないということを示せば十分である. すなわち, 元々の L_p 距離関数による範囲と問合せ点の範囲

$$[\mathrm{dist}_p(SP_i,\boldsymbol{q})-\varepsilon, \mathrm{dist}_p(SP_i,\boldsymbol{q})+\varepsilon]$$

が交差を持つ場合, 範囲 (1) との交差があることを示せばよい. このためには, 次の 2 つの不等式が示せれば十分である.

$$
\begin{aligned}
\min_{\boldsymbol{x}\in D_{SP_j}} \mathrm{dist}_\infty(SP_i,\boldsymbol{x}) &\leq \min_{\boldsymbol{x}\in D_{SP_j}} \mathrm{dist}_p(SP_i,\boldsymbol{x}) \\
\max_{\boldsymbol{x}\in D_{SP_j}} \mathrm{dist}_1(SP_i,\boldsymbol{x}) &\geq \max_{\boldsymbol{x}\in D_{SP_j}} \mathrm{dist}_p(SP_i,\boldsymbol{x})
\end{aligned}
$$

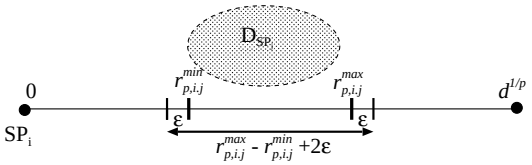

図 3 必要集合と判定される確率
Fig. 3 Probability of necessary set

ここでは, 前者の不等式だけを証明する. 後者も同様に証明ができる. 今, $\boldsymbol{y}^\infty$ を $\mathrm{dist}_\infty(SP_i,\boldsymbol{y}^\infty) = \min_{\boldsymbol{x}\in D_{SP_j}} \mathrm{dist}_\infty(SP_i,\boldsymbol{x})$ を満たす点, $\boldsymbol{y}^p$ を $\mathrm{dist}_p(SP_i,\boldsymbol{y}^p) = \min_{\boldsymbol{x}\in D_{SP_j}} \mathrm{dist}_p(SP_i,\boldsymbol{x})$ を満たす点とする. y^∞ が L_∞ 距離で最小値を与えること, 及び補題 1 を用いると,

$$
\begin{aligned}
\min_{\boldsymbol{x}\in D_{SP_j}} \mathrm{dist}_\infty(SP_i,\boldsymbol{x}) &= \mathrm{dist}_\infty(SP_i,\boldsymbol{y}^\infty) \\
&\leq \mathrm{dist}_\infty(SP_i,\boldsymbol{y}^p) \\
&\leq \mathrm{dist}_p(SP_i,\boldsymbol{y}^p) \\
&= \min_{\boldsymbol{x}\in D_{SP_j}} \mathrm{dist}_p(SP_i,\boldsymbol{x})
\end{aligned}
$$

が成立する.

以上のことから, 任意の L_p 距離に対し, 存在範囲 (1) により不必要集合を決定することができ, 残りの必要集合には, ε 近傍検索の正答となるデータ点が全て含まれていることがわかる. この必要集合のデータ点に対して, 距離計算を行うことで, L_p 距離関数による ε 近傍検索が可能となる. よって, 次の定理を得ることができた.

定理 1 mm-GNAT を用いて, 任意の L_p 距離関数による ε 近傍検索を行うことができる.

次に, mm-GNAT を用いることで, 計算コストがどの程度増加するのかを解析する. まず, クラスタ D_{SP_j} を固定する. このクラスタが ε 近傍検索において, 必要集合であると判定される確率を計算する. GNAT では, 各クラスタが必要集合であるかどうかの判定を分割点から各クラスタの存在範囲と問合せ点の ε 近傍の交わりの有無により行っている. データが $[0,1]^d$ に含まれるとすると, この存在範囲は, d 次元空間での L_p 距離の最小値 0 と最大値 $d^{1/p}$ を端点とする区間内に含まれる. データ点がこの空間では一様分布をなすと仮定すると, クラスタ D_{SP_j} が必要集合であると判定される確率は, クラスタの存在範囲の長さに比例する. さらに, ε 近傍検索を行うことを考慮にいれると, 存在範囲の長さに 2ε を加えた長さを数直線の全体 $d^{1/p}$ で割ったものが確率となることが分かる (図 3 参照).

すなわち, 必要集合となる確率は,

$$\frac{r^{\max}_{p,i,j} - r^{\min}_{p,i,j} + 2\varepsilon}{d^{1/p}}$$

で表される. ただし, $r^{\max}_{p,i,j}, r^{\min}_{p,i,j}$ は, ある分割点 SP_i

からクラスタ D_{SP_j} までの L_p 距離による最大値と最小値とする．これをすべての分割点に対して繰り返し，全てに必要集合であると判定された場合にのみ，最終的な検索対象となる．つまり，あるクラスタが必要集合と判定される確率は以下で表される．

$$\prod_{i=1}^{k}\left(\frac{r_{p,i,j}^{\max}-r_{p,i,j}^{\min}+2\varepsilon}{d^{1/p}}\right)$$

通常，ε は範囲 $r_{p,i,j}^{\max}-r_{p,i,j}^{\min}$ に比べ，かなり小さな値を取る．そこで，上式を次で近似する．

$$\prod_{i=1}^{k}\left(\frac{r_{p,i,j}^{\max}-r_{p,i,j}^{\min}+2\varepsilon}{d^{1/p}}\right)\sim\left(\frac{\widetilde{\mathrm{diff}}_{p,j}}{d^{1/p}}\right)^{k}$$

ただし，$\widetilde{\mathrm{diff}}_{p,j}$ を最大値と最小値の差分の相乗平均とする．つまり，$\widetilde{\mathrm{diff}}_{p,j}=\left\{\prod_{i=1}^{k}(r_{p,i,j}^{\max}-r_{p,i,j}^{\min})\right\}^{1/k}$ である．結局，検索対象となる点数の期待値は，次で表現される．

$$\mathrm{E}\left[\sum_{j=1}^{k}\left(\frac{\widetilde{\mathrm{diff}}_{p,j}}{d^{1/p}}\right)^{k}\cdot\left|D_{SP_j}\right|\right]$$

ここで，次の 2 つの条件を仮定する．

- それぞれのクラスタに含まれる点数 $\left|D_{SP_j}\right|$ は，どのクラスタであっても，ほぼ n/k である．ただし，n はデータ集合の総点数である．
- あるクラスタが必要集合となる確率 $\frac{\widetilde{\mathrm{diff}}_{p,j}}{d^{1/p}}$ は，他のクラスタが必要集合となる確率とは独立である．

最初の仮定を用いると，先の期待値は，次式で表される．

$$\frac{n}{k}\cdot\mathrm{E}\left[\sum_{j=1}^{k}\left(\frac{\widetilde{\mathrm{diff}}_{p,j}}{d^{1/p}}\right)^{k}\right]$$

2 つめの仮定を用いると，次の計算をすることができる．

$$\begin{aligned}\frac{n}{k}\cdot\mathrm{E}\left[\sum_{j=1}^{k}\left(\frac{\widetilde{\mathrm{diff}}_{p,j}}{d^{1/p}}\right)^{k}\right]&=\frac{n}{k}\cdot\sum_{j=1}^{k}\mathrm{E}\left[\left(\frac{\widetilde{\mathrm{diff}}_{p,j}}{d^{1/p}}\right)^{k}\right]\\&=\frac{n}{k}\cdot\sum_{j=1}^{k}\left(\mathrm{E}\left[\frac{\widetilde{\mathrm{diff}}_{p,j}}{d^{1/p}}\right]\right)^{k}\\&=\frac{n}{k}\cdot\sum_{j=1}^{k}\left(\frac{\mathrm{E}[\widetilde{\mathrm{diff}}_{p,j}]}{d^{1/p}}\right)^{k}\\&=\frac{n}{k}\cdot k\cdot\left(\frac{\mathrm{E}[\widetilde{\mathrm{diff}}_{p,j}]}{d^{1/p}}\right)^{k}\\&=n\left(\frac{\mathrm{E}[\widetilde{\mathrm{diff}}_{p,j}]}{d^{1/p}}\right)^{k}\end{aligned}$$

$\mathrm{E}[\widetilde{\mathrm{diff}}_{p,j}]$ は，$\widetilde{\mathrm{diff}}_{p,j}$ の期待値であるので，以下では，$\overline{\mathrm{diff}}_p$ と表記することにする．

この期待値に期待値自身を計算するために必要な距離計算の回数 k^2 を加えると，最終的に次の距離計算回数の期待値を得ることができる．

$$k^2+n\left(\frac{\overline{\mathrm{diff}}_p}{d^{1/p}}\right)^{k}$$

同様に，mm-GNAT の場合にも，評価を行うことが可能である．この場合は，距離の範囲が $[0,d]$ であり，$\widetilde{\mathrm{diff}}_{\mathrm{mm},j}=\left\{\prod_{i=1}^{k}(r_{1,i,j}^{\max}-r_{\infty,i,j}^{\min})\right\}^{1/k}$ となる．また，$\overline{\mathrm{diff}}_{\mathrm{mm}}$ は，$\widetilde{\mathrm{diff}}_{\mathrm{mm},j}$ の期待値である．これらの記号を用いて，期待値を表現すると，次の式が得られる．

$$k^2+n\left(\frac{\overline{\mathrm{diff}}_{\mathrm{mm}}}{d}\right)^{k}\tag{2}$$

我々は，mm-GNAT によりどの程度計算すべきデータ点の数が増えるのかを推測したいので，これら期待値の比を計算することで，次の式を得ることができる．

$$\left[\frac{k^2}{n}+\left(\frac{\overline{\mathrm{diff}}_{\mathrm{mm}}}{d}\right)^{k}\right]\Bigg/\left[\frac{k^2}{n}+\left(\frac{\overline{\mathrm{diff}}_p}{d^{1/p}}\right)^{k}\right]$$

分母と分子の初項が第 2 項に比べて十分小さな値となる場合には，分母と分子の初項は無視でき，総点数 n に依存しない次の式を得ることができる．

$$\left(\frac{\overline{\mathrm{diff}}_{\mathrm{mm}}}{\overline{\mathrm{diff}}_p}\cdot d^{1/p-1}\right)^{k}\tag{3}$$

4. 実　　験

本節では，提案する mm-GNAT がどの程度の検索性能を持つかを調べるために，次の 2 つの手法による実験を行い，比較を行った．

(1) L_p 距離による GNAT を p 毎に構築し，L_p 距離による検索を行う (標準方法)．
(2) mm-GNAT を構築し，L_p 距離による検索を行う (mm-GNAT)．

表 1 にあげる 4 つのデータ集合に対し，実験を行った．

まず，それぞれの索引構造において保持されている存在範囲の値を調べる．また，索引構造に対する検索実験を行い，問合せ点の ε 近傍に含まれる点数とそのために必要となる距離計算回数を調べる．ただし，GNAT を構築するための分割点として，データ集合の中から，無作為に 1000 点 (データ点の 1%) を選んだ．

[クラスタ構築] GNAT では，存在範囲が小さくなればなるほど検索効率がよくなるので，通常は検索に用いる距離が小さくなるようにクラスタを構築する．つまり，分割点の集合に対して，検索に用いる距離を基にボロノイ図を構築し，各ボロノイ領域をそれぞれのクラスタとすることである．mm-GNAT では，ある特定の距離関数でクラスタを構築するので，様々な距離

表 1　実験に用いたデータ集合
Table 1　Data set for experiment

	次数	総点数	分布	データの種類
DB_1	4	10 万	一様分布	人工データ
DB_2	8	10 万	一様分布	人工データ
DB_3	16	10 万	一様分布	人工データ
DB_4	20	10 万	データ依存	楽曲データ[10),16)]

表 2 mm-GNAT の構築時間 (秒)
Table 2 Computation time of mm-GNAT (sec)

クラスタ作成	DB_1	DB_2	DB_3	DB_4
L_1 距離	236	403	708	892
L_2 距離	245	415	741	840
L_∞ 距離	160	270	514	583

表 3 mm-GNAT の存在範囲
Table 3 Ranges of mm-GNAT

	DB_1	DB_2	DB_3	DB_4
mm-GNAT(L_1)	1.044	2.974	6.447	3.651
mm-GNAT(L_2)	1.020	2.941	6.405	3.617
mm-GNAT(L_∞)	1.058	3.038	6.460	3.774
GNAT($p=1$)	0.463	1.552	3.200	0.766
GNAT($p=2$)	0.221	0.533	0.777	0.176
GNAT($p=3$)	0.194	0.412	0.520	0.123
GNAT($p=4$)	0.188	0.375	0.441	0.107
GNAT($p=5$)	0.187	0.361	0.407	0.101
GNAT($p=6$)	0.188	0.355	0.390	0.097
GNAT($p=7$)	0.188	0.352	0.381	0.096
GNAT($p=8$)	0.189	0.351	0.375	0.095
GNAT($p=9$)	0.191	0.350	0.372	0.094
GNAT($p=10$)	0.191	0.350	0.370	0.094
GNAT($p=\infty$)	0.200	0.361	0.377	0.096

関数を用いて検索を行った時に, 必ずしも検索効率がよくなるとは考えられない. そこで, 我々は, L_1 距離, L_2 距離と L_∞ 距離を基準とし, クラスタを構築し, それぞれに対して, mm-GNAT の構築を行った. これは, 分割点に対し, L_1 距離, L_2 距離と L_∞ 距離でのボロノイ図をそれぞれ構築し, 各ボロノイ領域に含まれる点をクラスタとすることに相当する. 以下の実験では, クラスタの違う mm-GNAT に対して, 実験を行い, どのクラスタが mm-GNAT に適しているかの検討も行う.

[**索引構造の構築時間**] 表 1 の 4 つのデータ集合に対して, mm-GNAT の構築時間を測定した (表 2). 標準方法では, 検索を考える距離関数毎に索引構造を持つ必要があるため, 構築時間は, 距離関数の数に比例した時間が必要となるばかりか, 索引構造を保持するための記憶領域も距離関数の数に比例した記憶領域が必要となる. 一方, mm-GNAT では, 1 つの索引構造だけを構築すればよい. つまり, 表 2 に挙げた構築時間だけでよい. さらに全ての索引構造を保持する場合に比べ, 索引を保持するための記憶領域も大幅に減少させることができる.

[**クラスタの範囲範囲**] L_p 距離 ($p=1,\ldots,10,\infty$) での GNAT と mm-GNAT を構築した際の存在範囲の値から, $\overline{\mathrm{diff}}_p$, $\overline{\mathrm{diff}}_{\mathrm{mm}}$ の計算を行った. その結果が表 3 である.

[**検索実験**] 次に標準方式と mm-GNAT を用いた場合の検索効率を調べるために, 選択度 s と距離計算回数 c の関係を求めた. ただし, 選択度 s とは, 問合せ点から L_p 距離で半径 ε 以下の点数 C(つまり, 正解数) とデータ集合の総点数との比であり, 距離計算回数は, ε 近傍に含まれる全ての点を出力するまでに要した距離計算の回数である. この回数は, 式 (2) で近似される値である.

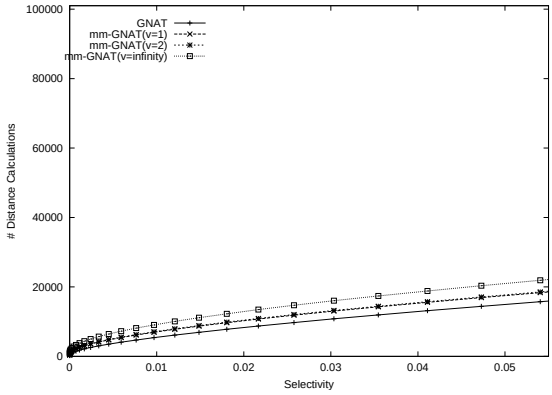

図 4 L_1 距離による検索結果 (選択度と距離計算回数, DB_1)

まず, データベースから無作為に点を選び, 問合せ点 q とする. 問合せ点 q に対し, ε 近傍検索を行い, 正解数 $C_{q,\varepsilon}$ と距離計算回数 $s_{q,\varepsilon}$ を求める. これを 1000 回繰り返し, ε 近傍検索での $C_{q,\varepsilon}$ の平均正解数 C_ε を計算する. この C_ε とデータベースの点数の比を ε 近傍検索の選択度とする. この選択度を用いることで, 異なる L_p 距離で検索を行った場合の比較が可能となる. また, $s_{q,\varepsilon}$ の平均値である平均距離計算回数 s_ε を計算する.

問合せ半径 ε を変化させながら, 選択度と平均距離計算回数を, 4 つのデータ集合に対し, 標準方法, L_v 距離を用いたクラスタで構築した mm-GNAT($v=1,2,\infty$) を索引空間として実験を行った結果を図 4 から図 9 に示す. ただし, ページ数の都合から, 4 次元人工データと 20 次元楽曲データに対する L_1, L_2 および L_∞ 距離による検索結果だけを載せている.

L_p 距離での ε 近傍検索を行う場合, 標準方法は最も平均距離計算回数が少ない. そこで, mm-GNAT での平均距離計算回数と標準方法の比を取ることで, mm-GNAT の検索効率を評価することを考える. 以降では, この比を増加比と呼ぶ. さらに, 検索する L_p 距離と増加比の関係を調べるために, クラスタを構築するのに用いた L_v 距離と検索する L_p 距離を決めた場合の増加比の最大値を選択度が 0 から 0.05 までの間で調べた. 各 p において, 増加比の最大値をクラスタの構築方法毎に表示したものが図 10, 図 11 である.

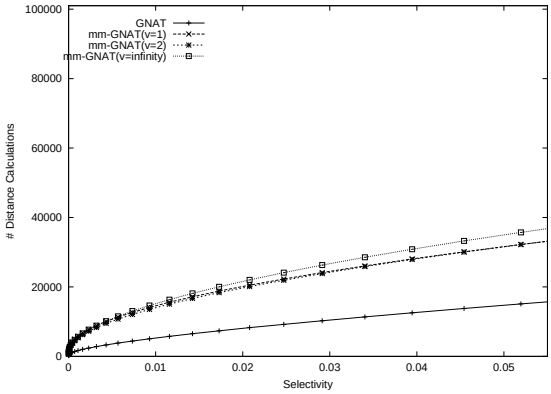

図 5 L_2 距離による検索結果 (選択度と距離計算回数, DB_1)

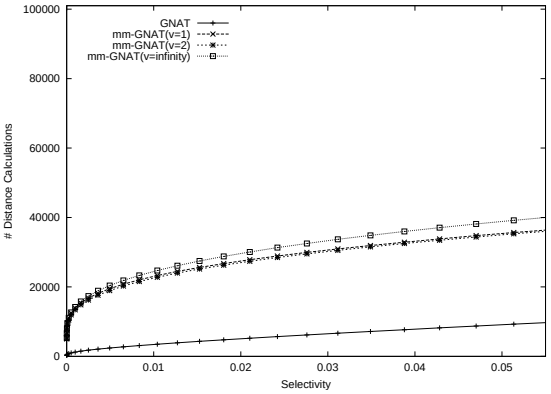

図 8 L_2 距離による検索結果 (選択度と距離計算回数, DB_4)

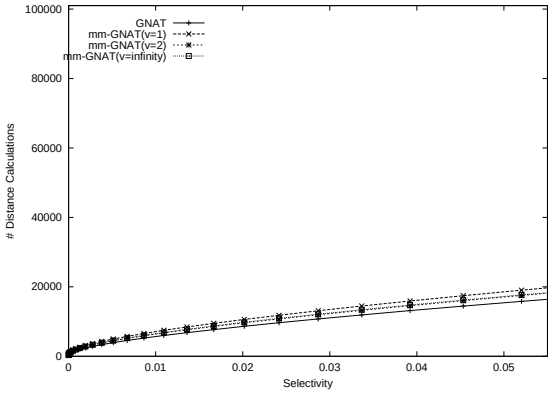

図 6 L_∞ 距離による検索結果 (選択度と距離計算回数, DB_1)

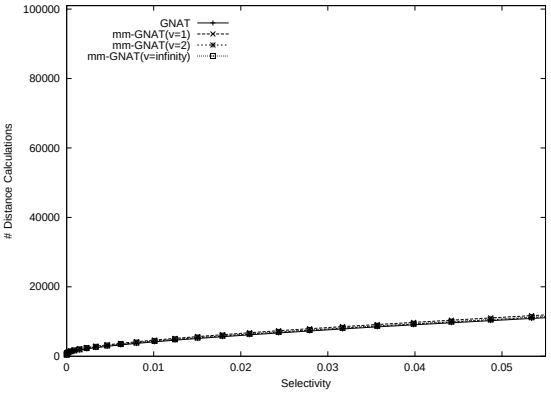

図 9 L_∞ 距離による検索結果 (選択度と距離計算回数, DB_4)

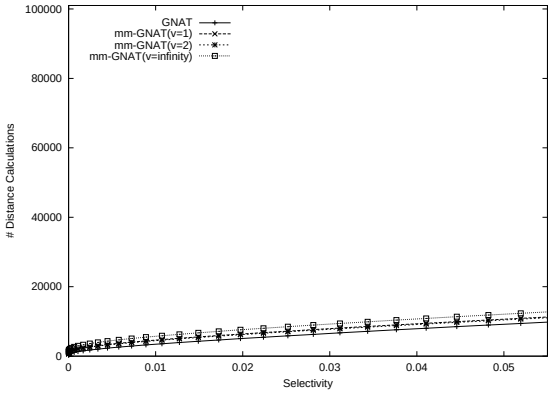

図 7 L_1 距離による検索結果 (選択度と距離計算回数, DB_4)

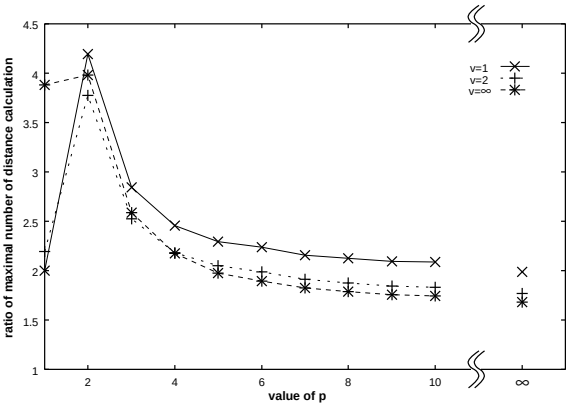

図 10 4 次元人工データ (DB_1) に対する検索距離関数 L_p と最大増加比の関係

これらの図では, 検索する L_p 距離の p の値を横軸にとり, 増加比の最大値を縦軸にとっている.

5. 議　　論

本節では, 前節までの, 理論的解析, 実験結果に基づき, 次の観点から, 議論を行う.

- 定理 1 の検証
- 選択度と距離計算回数の関係
- クラスタの違いによる性能への影響
- 理論的解析との比較

[**定理 1 の検証**] mm-GNAT を索引構造として検索実験を行ったときに得られる正解点は, 全ての場合において, 標準方法で検索を行ったときに得られた正解点

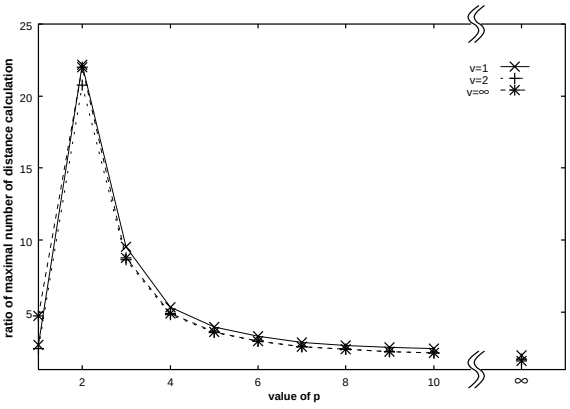

図 11 20 次元楽曲データ (DB_4) に対する検索距離関数 L_p と最大増加比の関係

と同じであったことを述べておく．これは, 定理 1 が正しいことを検索実験により検証したことになる．

[**選択度と距離計算回数の関係**] まず, 人工データである DB_1, DB_2, DB_3 について議論を行う．これらデータ集合は, 次元が $4, 8, 16$ と順に大きくなる．次元が大きくなると, どのような L_p 距離での検索を行う場合でも, 急激に距離計算回数が増え, 小さな選択度で全てのデータ点と距離計算を行うこと (以下では, 全件検索と呼ぶ) が起こるのが確認された．DB_2(8 次元) での, L_p による検索は L_2 距離の検索以外では, 絞込みの効果があった．L_2 距離の場合には非常に小さな選択度 (0.001 程度) で全件検索となった．DB_3(16 次元) での検索では, どの L_p 距離で検索を行っても, 非常に小さな選択度で全件検索となった．標準方法でも, 同様の傾向を示したため, これは mm-GNAT の問題というよりは, GNAT 自身が高次元ではあまり有効に働かないためと考えられる．また, これらデータ集合に対する検索実験では, いずれも選択度を大きくすると, 距離計算回数が単調に増加していき, 全検検索よりも多くの距離計算を行うことになる．これは, 選択度が大きくなれば, 検索半径も大きくなるためである．全件検索よりも多くの距離計算回数が必要となるのは, 必要集合の絞込みが十分に行われず, その絞込みに必要となる距離計算回数だけが大きくなるためである．

次に, 実データである DB_4 について議論を行う．BD_4 は 20 次元のデータであるため, DB_3 と同様に, 非常に小さな選択度で全件検索になるのではということが予測されたが, 実際の検索結果は, DB_1 (4 次元データ) に近い検索結果となった．DB_4 に主成分分析をかけたところ, 累積寄与率が第 4 軸までで 99.07%, 第 5 軸までで 99.52%となることが判った．つまり, 20 次元の実データであっても, 4 次元のデータとして取り扱うことが可能であることが判った．これから, 高次元の実データであっても, そのデータの特性によっては, mm-GNAT を用いることで, 十分に検索が可能となることが考えられる．

[**クラスタによる性能への影響**] ここでは, どの L_v 距離関数でクラスタを構築すれば, 検索性能のよい mm-GNAT を構築できるのかを考察する．まず, 図 4, 図 7 に着目する．これは L_1 距離による検索を行った実験結果である．この両方において, 標準方法, L_1 クラスタ, L_2 クラスタ, L_∞ クラスタの順に距離計算回数が少ないことがわかる．また, L_2 距離での検索結果である図 5, 図 8 では, L_2 クラスタ, L_1 クラスタ, L_∞ クラスタの順に, L_∞ 距離での検索結果である図 6, 図 9 では, L_∞ クラスタ, L_2 クラスタ, L_1 クラスタの順に距離計算回数が少ない．L_v クラスタと同じ L_p 距離 ($p = v$ のとき) で検索を行う場合には距離計算回数がもっとも少ない．v よりも p が大きくなる場合と, v よりも p が小さくなる場合に, 距離計算回数が増加していくことが $v = 1, 2, \infty$ の場合には, 検索実験で確認された．これは, それぞれのクラスタを L_v 距離で構築しているため, L_v 距離での検索を行う場合には, 必要集合の絞込みが適切に行われ, 距離計算の回数が少なくなるが, それ以外の距離で検索を行うとすると, 不必要集合を必要集合と判定するという現象が起こり, 距離計算回数が増加すると考えられる．また, 実験を行った以外のクラスタを構築した場合にも, 同様の結果が得られると予測される．

[**理論的解析との比較**] ここでは, 3 節で導いた計算回数比を示す式 (3) と実際の計算回数比である図 10, 図 11 について述べる．まず, 式 (3) に付いて述べる．この式は, 索引構造から決定される $\overline{\mathrm{diff}}_{\mathrm{mm}}$, $\overline{\mathrm{diff}}_p$, 分割点数 k, 次元 d, および, 検索を行う L_p 距離の p からなる距離計算回数比の推測式である．この式で p と表 3 の値を代入することで, 図 10, 図 11 と同じ横軸, 及び縦軸を持つグラフを作成することができる．表 3 の DB_1 の値 (次元は $d = 4$) と $k = 1$ の場合, $p = 2$ でピークをもつ図 10, 図 11 と同様の形状をしたグラフを得ることができた (図 12). 式 (3) は, 式全体を k 乗するので, グラフの形状は $k = 1$ の場合と同様になる．すなわち, 式 (3) は, DB_1 に対しては, 距離計算回数比の傾向を表す式となっていることが分かる．

以下では, 式 (3) について考察を行う．まず, クラスタを固定した場合について考える．クラスタを固定することで, $\overline{\mathrm{diff}}_{\mathrm{mm}}$ が一定の値を取ることになる．つまり, $d^{1/p-1}/\overline{\mathrm{diff}}_p$ がデータ構造の性能を決定することになる．実際のデータを見ると, $p = 1$ の場合には,

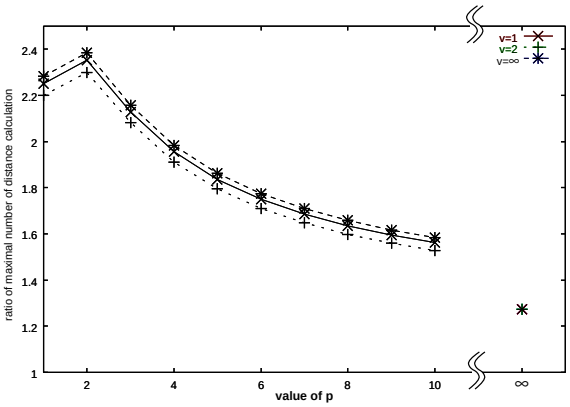

図 12 式 (3) と表 3 から得られるグラフ

$\overline{\mathrm{diff}}_p$ の値が大きく全体の値を小さくしている．p が大きくなるにつれて，$\overline{\mathrm{diff}}_p$ の値は小さくなり，変化も小さくなっていく．このときは，$d^{1/p-1}$ の部分により式全体の傾向が決定される．

次に，データ集合を固定し，どのようなクラスタが性能がよいのかについて考察を行う．データ集合は固定されているため，次元 d も固定される．ここである特定の p の場合を考えると，クラスタの変更により，値が変化するのは，$\overline{\mathrm{diff}}_{\mathrm{mm}}$ だけである．L_1 クラスタ，L_2 クラスタ，L_∞ クラスタに対する $\overline{\mathrm{diff}}_{\mathrm{mm}}$ の値を比較すると，表 3 から

$$L_2\text{クラスタ} < L_1\text{クラスタ} < L_\infty\text{クラスタ}$$

という関係があることがわかる．この関係が図 12 の全ての p に対しても成立している．すなわち，$\overline{\mathrm{diff}}_{\mathrm{mm}}$ の値の大小が性能を表すことになる．言い換えれば，クラスタの直径が小さくなるようにクラスタを構築することで，mm-GNAT の性能を向上させることができると考えられる．

6. ま と め

本稿では，距離変換規則を用いず，データ構造として，任意の L_p 距離を用いた検索を実現する手法について提案を行った．これは，L_p 距離に関する次の関係を用いて，実現がなされている．

$$d_\infty(\boldsymbol{x},\boldsymbol{y}) \le d_p(\boldsymbol{x},\boldsymbol{y}) \le d_1(\boldsymbol{x},\boldsymbol{y})$$

この関係と GNAT で保持される分割点からの最大距離を L_1 距離に，最小距離を L_∞ 距離に変更することで，任意の L_p 距離による検索を実現している．つまり，どのような L_p 距離で検索を行おうと考えたとしても，その範囲が必ず含まれるように GNAT における距離の範囲を変更したということである．我々は，この考えを導入したデータ構造 mm-GNAT を実際に構築し，計算機実験を行った．その結果，mm-GNAT により，幾つかのデータ集合に対して任意の L_p 距離関数での ε 近傍検索が実現できたことが確認された．

また，この考え方は，次のように一般化することができる．

> 任意の点 $\boldsymbol{x},\boldsymbol{y}$ に対して，次の関係を満たす距離関数が存在すると仮定する．
>
> $$d'(\boldsymbol{x},\boldsymbol{y}) \le d(\boldsymbol{x},\boldsymbol{y}) \le d''(\boldsymbol{x},\boldsymbol{y})$$
>
> ただし，d, d', d'' は別の距離関数とする．このとき，GNAT での，分割点からの最大距離を d'' で，分割点からの最小距離を d' で保持することで，距離関数 d での ε 近傍検索を実現することができる．

このような視点で，本研究をとらえると，検索可能な L_p 距離を制限することで，性能の向上を目指すことも可能である．先の論文[15] で示した関係

$$d_p(\boldsymbol{x},\boldsymbol{y}) \le d_q(\boldsymbol{x},\boldsymbol{y}) \quad (p,q \text{ は正整数であり}, p > q)$$

を用いて，

$$d_\infty(\boldsymbol{x},\boldsymbol{y}) \le d_p(\boldsymbol{x},\boldsymbol{y}) \le d_2(\boldsymbol{x},\boldsymbol{y})$$

という関係が成立することがわかる．このとき，この mm-GNAT では，L_2 距離から L_∞ 距離での検索しかできないが，L_1 距離での検索をサポートしない分だけ，性能の向上が期待される．これは，表 3 から，L_1 距離での範囲が他の距離よりもかなり大きくなるため，必要集合の絞込みのを行う際に，本来ならば不必要集合である集合を必要集合と判定していることが予測されるためである．

本稿では，GNAT のクラスタを L_1 距離，L_2 距離，および L_∞ 距離で構築した．これは，どのようなクラスタを用いれば，検索性能，すなわち距離計算回数が減少するかが判らないためである．実験結果から，検索する距離とクラスタを構築する距離が同じ場合には検索性能がもっとも良くなることがわかった．また，階層化した GNAT を用いて，予備実験を行ったが，どのクラスタを用いた場合よりも，距離計算回数が減少した．これらの実験から，クラスタ内の点と問合せ点の距離がどの L_p 距離で計算を行ってもさほど大きな変化が無いようにクラスタを構成することができれば，検索性能の向上がある考えられる．このようなクラスタを研究することは，データ構造の研究というだけでなく，様々な距離を用いたクラスタリングの変化についての研究を行うことである．さらに言うならば，様々な距離関数を用いたボロノイ図に関する研究を行うことでもある．著者は，既に任意の L_p 距離でのボロノイ領域に関する研究を行っている[9]．この研究では，同一点集合に対して，様々な L_p 距離でのボロノイ領域を構築した場合，それらのボロノイ領域の交わりは，L_1

距離でのボロノイ領域と L_∞ 距離でのボロノイ領域の交わりで与えられることを示した. このボロノイ領域の交わりをクラスタとすると, クラスタに含まれる点は, どのような L_p 距離関数で距離計算を行っても, 分割点から最も近い点となっている. この領域を用いることで, さらに mm-GNAT の性能が向上することが考えられる.

参 考 文 献

1) Böhm, C., Berchtold, S. and A.Keim, D.: Searching in High-Dimensional Spaces-Index Structures for Improving the Performance of Multimedia Databases, *ACM Computing Surveys*, Vol. 33, No. 3, pp. 322–373 (2001).
2) Brin, S.: Near Neighbor Search in Large Metric Spaces, *Proc. of the 21st International Conference on VLDB*, pp. 574–584 (1995).
3) Chávez, E., Navarro, G., Baeza-Yates, R. and Marroquin, J. L.: Searching in Metric Spaces, *ACM Computing Surveys*, Vol. 33, No. 3, pp. 273–321 (2001).
4) Ciaccia, P., Patella, M. and Zezula, P.: M-Tree: An Efficient Access Method for Similarity Search in Metric Spaces, *Proc. of the 23rd International Conference on VLDB*, pp. 426–435 (1997).
5) Guttman, A.: R-Trees: A Dynamic Index Structure for Spatial Searching, *Proc. of ACM SIGMOD*, pp. 47–57 (1984).
6) Hardy, G., Littlewood, J.E. and Polya, G.: *Inequalities*, Cambridge University Press (1952).
7) Katayama, N. and Satoh, S.: The SR-tree: An Index Structure for High-Dimensional Nearest Neighbor Queries, *Proc. of ACM SIGMOD*, pp. 369–380 (1997).
8) Lee, S., Kwon, D. and Lee, S.: Minimum Distance Queries for Time Series Data, *Journal of Systems and Software*, Vol. 69, No. 1–2, pp. 105-113 (2004).
9) Onishi, K.: Intersection of Voronoi Regions by L_p distance, *Proc. of Japan Conference on Discrite and Computational Geometry '99*, pp. 26–28 (1999).
10) Onishi, K., Kobayakawa, M., Hoshi, M. and Ohmori, T.: A Feature independent of bit rate for TwinVQ Audio Retrieval, *Proc. of 2001 IEEE International Conference on Multimedia and Expo (ICME 2001)*, pp. 409–412 (2001).
11) Weber, R., Schek, H.-J. and Blott, S.: A Quantitative Analysis and Performance Study for Similarity-Search Methods in High-Dimensional Spaces, *Proc. of the 24th International Conference on VLDB*, pp. 194–205 (1998).
12) White, D.A. and Jain, R.: Similarity Indexing with the SS-tree, *Proc. of the 12th ICDE*, pp. 516–523 (1996).
13) Yi, B.-K. and Faloutsos, C.: Fast Time Sequence Indexing for Arbitrary $\mathcal{L}_p$ Norms, *Proc. of the 26th International Conference on VLDB*, pp. 385–394 (2000).
14) Yianilos, P. N.: Data Structures and Algorithms for Nearest Neighbor Search in General Metric Spaces, *Proc. of the Fifth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pp. 311–321 (1993).
15) 木村彰宏, 大西建輔, 小早川倫広, 星守, 大森匡: 任意の L_p 距離による検索を可能とする距離変換規則, 情報処理学会論文誌：データベース, Vol. 46, No. SIG 8(TOD 26), pp. 93-105 (2005).
16) 墳崎英明, 小早川倫広, 大西建輔, 星守, 大森匡: ビットレートの異なる TwinVQ オーディオデータの類似曲検索のための特徴量, 情報処理学会第 59 回全国大会, pp. 53–54 (3) (1999).