曖昧な位置を持つオブジェクトによる最近傍問合せの処理手法

飯島　裕一†　　石川　佳治††

† 名古屋大学大学院情報科学研究科　〒 464–8601 名古屋市千種区不老町
†† 名古屋大学情報連携基盤センター　〒 464–8601 名古屋市千種区不老町
E-mail: †iijima@db.itc.nagoya-u.ac.jp, ††ishikawa@itc.nagoya-u.ac.jp

あらまし　移動ロボットやモバイルセンサネットワークなどの位置情報を利用したアプリケーションでは，最近傍問合せは重要な問合せとして位置づけられているが，ノイズや測定誤差などのためにオブジェクトの位置は本質的に曖昧であるため，その曖昧さを考慮した問合せ処理手法が必要とされている．そこで本研究では，曖昧な位置を持つオブジェクトによる最近傍問合せの処理手法を提案する．
キーワード　空間データベース，最近傍問合せ，曖昧な位置，正規分布

Processing Methods for Nearest Neighbor Queries Issued by an Object with Imprecise Location

Yuichi IIJIMA† and Yoshiharu ISHIKAWA††

† Graduate School of Information Science, Nagoya University　Furo-cho, Chikusa-ku, Nagoya, 464–8601 Japan
†† Information Technology Center, Nagoya University　Furo-cho, Chikusa-ku, Nagoya, 464–8601 Japan
E-mail: †iijima@db.itc.nagoya-u.ac.jp, ††ishikawa@itc.nagoya-u.ac.jp

Abstract　A nearest neighbor query is an important notion in location-based applications such as mobile robotics and mobile sensor networks. In these application fields, query processing methods considering impreciseness are required because the location of an object is essentially imprecise due to noise and measurement errors. In this paper, we propose techniques for processing nearest neighbor queries issued by an object whose location is imprecise.
Key words　Spatial databases, nearest neighbor queries, imprecise locations, Gaussian distributions

1. はじめに

近年，移動ロボットやモバイルセンサネットワークなどの分野において，曖昧な位置情報に基づく問合せ処理技術の必要性が高まってきている．現実環境中を動き回る移動ロボットにとって，自身の位置の推定は円滑なサービス提供を行う上で欠かせないものである．移動ロボットは通常，センサからの信号や移動履歴などを基に，統計的な手法を用いて自身の位置を継続的に推定する [14] が，センサの測定誤差やモータの制御ノイズなどのために正確な位置の推定は容易ではなく，誤差を伴った推定となる．また，複数のセンサが各々の周辺の環境情報を収集するモバイルセンサネットワークにおいては，各センサの位置情報を把握することが必要であるが，現在一般的な手法である GPS による位置測定では，電波状況の悪さなどのために必ずしも十分な測位精度が得られるとは限らない [11]．加えて，GPS による位置取得は多くの電力を消費するため，各センサが電池で駆動しているような場合には極力避けたいという要求もある．以上のように，現実世界のオブジェクトの位置は曖昧な位置情報としてしか得ることができない場合が多いため，位置の曖昧さを考慮した問合せ処理手法が必要とされており，その研究が盛んになってきている．

このような背景から，本研究では，曖昧な位置を持つオブジェクトが，自らの位置から最も近くにあるオブジェクトを検索するために最近傍問合せを行うという状況を対象とする．具体的には，問合せオブジェクトの位置が**正規分布**で表現され，問合せ対象オブジェクトが確定的な位置で表される**点データ**である状況を扱う．対象とする問合せとしてユークリッド距離に基づく通常の最近傍問合せを拡張した**確率的最近傍問合せ**を定義し，この問合せを効率的に処理するための具体的な戦略として 2 つの問合せ戦略を導入する．実験では，2 つの戦略にそれらのハイブリッド戦略を加えた 3 つの戦略について，様々なパラメータ設定の下で比較を行い各戦略を評価する．

本稿の構成は以下の通りである．まず，2 節で関連研究を紹介する．次に，3 節で確率的最近傍問合せを定義し，続く 4 節でその処理手法を提案する．5 節では評価実験の結果を示し，6 節でまとめを行う．

2. 関 連 研 究

曖昧な位置情報を考慮した問合せ処理に関する研究が最近特に注目を集めている．近年なされた研究は，それぞれの対象とする状況から以下の 3 種類に分類できる．

（1） 問合せ対象オブジェクトのみ曖昧 [7], [8], [11], [13]

（2） 問合せオブジェクトのみ曖昧 [9]

（3） 両オブジェクトともに曖昧 [5], [6], [10]

本研究は 2 種類目に該当している．

別の分類基準としては位置の曖昧さを表すためのモデルが挙げられる．例えば, [11] では単純に，位置の分布が一様分布であることと，曖昧なオブジェクトが内部に位置しているような領域の存在を前提としている一方で, [5], [6], [13] では任意の確率分布の使用を認めている．ただし任意とはいえ，通常は各オブジェクトに対して，*uncertainty region* と呼ばれる，そのオブジェクトが内部に位置している確率が指定値以上であることが保証されているような領域が与えられることを前提としている．これらの研究とは異なり，本研究では位置の曖昧さが正規分布に基づいている状況を扱う．正規分布は統計やパターン認識などの分野でよく用いられる確率密度関数であることから，本研究ではその一般性に着目し，対象が正規分布であることに特化した処理技術に焦点を合わせる．正規分布の性質を効果的に用いることで，効率的な問合せ処理手法の開発が可能となる．

問合せの種類としては，主に範囲問合せ [8], [9], [13] や最近傍問合せ [7], [8], [10] などを対象に位置の曖昧さを考慮した問合せ処理が研究されている．Cheng らは [8] で，uncertainty region を用いて候補の限定を行う最近傍問合せの処理手法を提案した. [7] では 1 次元の曖昧なオブジェクトに対する最近傍問合せを検討しており，そこで提案されている問合せは確率の閾値が与えられるという点で本研究の対象とする問合せと関係している．この 2 つの研究は問合せ対象オブジェクトのみが曖昧であるという状況を対象としているが，Kriegel らによるサンプリング手法を用いたアプローチ [10] や Beskales らによる k-最近傍問合せの処理手法 [5] など，問合せオブジェクトの方も曖昧である状況を対象とした最近傍問合せの処理手法もいくつか提案されている．しかし，ここで紹介したすべての最近傍問合せ手法は本研究のように正規分布に焦点を合わせたものではない．

本研究グループではすでに，本研究が対象とする状況と同様の状況における範囲問合せの処理手法を [9] で提案しており，その一部のアイデアを本手法でも導入している．ただし，本研究が対象とするのは最近傍問合せであるため，その特徴を考慮した改良や新しい技術が必要となる．本研究では，**ボロノイ図**を効果的に使用することで効率的な問合せ処理を実現する．

3. 確率的最近傍問合せ

問合せオブジェクトと問合せ対象オブジェクトがともに確定的な位置で表されている通常の最近傍問合せの場合には，従来の最近傍問合せ処理手法を用いて簡単に問合せ処理を行うことができる．しかしながら，どちらか一方でも位置が曖昧，すなわち確率的である場合には，問合せ結果も確率的に決まるため，確率的な概念を考慮しない通常の最近傍問合せを対象とする従来の処理手法では対応することができない．本研究では問合せオブジェクトの位置が正規分布の確率密度関数によって曖昧な位置情報で表現されている状況を対象とするため，確率的な概念を取り扱えるように通常の最近傍問合せを拡張する必要がある．そこで，**確率的最近傍問合せ**（probabilistic nearest neighbor query, PNNQ）を以下のように定義する．

［定義 1］ **確率的最近傍問合せ**

d 次元空間において，問合せオブジェクト q の位置が d 次元ベクトルの座標値 $\boldsymbol{x}$ を持つ確率が，d 次元正規分布により，

$$p_q(\boldsymbol{x}) = \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left[-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{q})^t\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{q})\right] \quad (1)$$

で表されるとする．ただし，$\boldsymbol{\Sigma}$ は $d \times d$ の共分散行列であり，$|\boldsymbol{\Sigma}|$ は $\boldsymbol{\Sigma}$ の行列式を表す．$\boldsymbol{q}$ は確率分布の平均に対応している．このような q が与えられたとき，q とのユークリッド距離がすべてのオブジェクトのうちで最も小さくなる確率（q の最近傍オブジェクトとなる確率）が θ $(0 < \theta < 1)$ 以上であるようなオブジェクトの集合を返す問合せを確率的最近傍問合せ $PNNQ(q, \theta)$ と定義する．また，問合せ対象オブジェクトの集合を $\mathcal{O}$ とするとき，問合せ対象オブジェクト $o \in \mathcal{O}$ が q の最近傍オブジェクトとなる確率 $\Pr_{NN}(q, o)$ は，

$$\Pr_{NN}(q, o) = \Pr(\forall o' \in \mathcal{O}, o' \neq o, \|\boldsymbol{x}-\boldsymbol{o}\|^2 \leqq \|\boldsymbol{x}-\boldsymbol{o'}\|^2) \quad (2)$$

と表される．ただし，$\|\boldsymbol{x} - \boldsymbol{o}\|^2$ は q の位置 $\boldsymbol{x}$ と o の位置 $\boldsymbol{o}$ のユークリッド距離の 2 乗を表している．$\Pr_{NN}(\cdot)$ を用いて $PNNQ(q, \theta)$ は以下のような式で表現される．

$$PNNQ(q, \theta) = \{n \mid n \in \mathcal{O}, \Pr_{NN}(q, n) \geqq \theta\} \quad (3)$$

通常の最近傍問合せとは異なり，複数のオブジェクトが結果として返されることに注意する．また，ユーザの与える閾値 θ が高過ぎる場合，空の結果が返されることになるという点にも注意が必要である．

4. 問合せ処理手法

本節では確率的最近傍問合せの処理手法を提案する．提案手法は主に 2 次元空間における問合せ処理を対象とするが，3 次元以上の場合にも適用可能な一般的なものとなっている．

4.1 基本的なアイデア

位置の曖昧さを考慮しない通常の最近傍問合せの場合，問合せ処理にボロノイ図 [4] を用いることが一般的である．本研究は問合せオブジェクトの位置が曖昧である状況を対象とするが，通常の最近傍問合せと同様にボロノイ図を用いて問合せ処理を行う．空間上の複数の点に対して，どの点に一番近いかによって空間を分割した図がボロノイ図である．図 1 に点オブジェクト a〜h に対するボロノイ図を示す．各点の勢力範囲を**ボロノイ領域**と呼び，点オブジェクト o のボロノイ領域を V_o と表す．例えば，図 1 の陰影部分は e のボロノイ領域 V_e である．

ボロノイ図の定義から，問合せ対象オブジェクト $o \in \mathcal{O}$ のボロノイ領域 V_o 内に問合せオブジェクト q が位置している場合に o は q の最近傍オブジェクトとなる．したがって，o が q

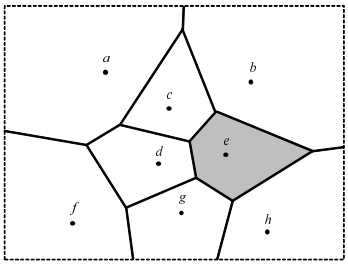

図 1　ボロノイ図

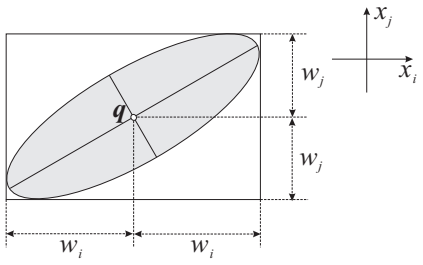

図 2　包囲矩形の利用

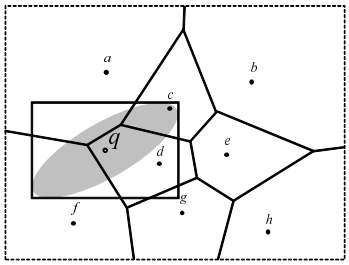

図 3　包囲矩形とボロノイ領域

の最近傍オブジェクトとなる確率 $\mathrm{Pr}_{NN}(q,o)$ は，q が V_o 内に位置する確率と言い換えることができる．つまり，$\mathrm{Pr}_{NN}(q,o)$ は式 (1) に示した確率密度関数 $p_q(\boldsymbol{x})$ を領域 V_o で積分することで計算可能である．以上の事実を踏まえ，式 (2) に示した $\mathrm{Pr}_{NN}(q,o)$ の計算式は以下のように書き換えることができる．

$$\mathrm{Pr}_{NN}(q,o) = \int_{\boldsymbol{x}\in V_o} p_q(\boldsymbol{x})d\boldsymbol{x} \tag{4}$$

すべての問合せ対象オブジェクトに対してこの式により $\mathrm{Pr}_{NN}(\cdot)$ を計算すれば，問合せを処理することができる．ただし，正規分布の確率密度関数の積分値は解析的には求められないため，$\mathrm{Pr}_{NN}(\cdot)$ の計算にはモンテカルロ法のような計算コストの高い数値積分が必要となる．加えて，各ボロノイ領域の形状が複雑な多面体であることも計算コストを高める要因となる．したがって，すべての問合せ対象オブジェクトに対して $\mathrm{Pr}_{NN}(\cdot)$ を求めることは現実的ではない．そこで本手法では，明らかに $\mathrm{Pr}_{NN}(\cdot)$ が θ に満たないといえるオブジェクトを除去し，残った候補オブジェクトに対してのみ $\mathrm{Pr}_{NN}(\cdot)$ を計算することでコストを抑える．本稿では以降，下線部の処理のことをフィルタリングと呼ぶ．このアイデアに基づく具体的な問合せ戦略として「問合せ戦略 1」と「問合せ戦略 2」を提案する．

4.2　問合せ戦略 1

問合せ戦略 1 では，2 節で紹介した uncertain region [13] の概念を導入する．具体的には，**θ 領域** [9] という，その領域の内部に問合せオブジェクトが位置する確率が $1-2\theta$ であるような領域を用いてフィルタリングを行う．定義を以下に示す．

［定義 2］ **θ 領域**

$(\boldsymbol{x}-\boldsymbol{q})^t\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{q}) \leqq r^2$ を満たす楕円体領域での確率密度関数 $p_q(\boldsymbol{x})$ の積分を考える．与えられた θ $(0<\theta<1/2)$ に対し，積分値が $1-2\theta$ になるような r の値を r_θ とする．すなわち，

$$\int_{(\boldsymbol{x}-\boldsymbol{q})^t\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{q})\leqq r_\theta^2} p_q(\boldsymbol{x})d\boldsymbol{x} = 1-2\theta \tag{5}$$

である．r_θ により以下の式で定まる楕円体領域を θ 領域と呼ぶ．

$$(\boldsymbol{x}-\boldsymbol{q})^t\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{q}) \leqq r_\theta^2 \tag{6}$$

θ 領域は問合せ時に与えられるパラメータに依存しているため，その導出は問合せ時に動的に行う必要がある．単純な方法として，問合せ時に様々な r の値に対して対応する楕円体領域での $p_q(\boldsymbol{x})$ の積分値を数値積分によって計算し，その値が $1-2\theta$ となるような $r=r_\theta$ を見つけるという方法が考えられるが，計算コストの面で現実的ではない．そこで，楕円体領域での積分を d 次元球領域での積分に変換する．まず，式 (1) において $\boldsymbol{q}=\mathbf{0}, \boldsymbol{\Sigma}=\mathbf{I}$ とした，**標準正規分布**の確率密度関数

$$p_{\mathrm{norm}}(\boldsymbol{x}) = \mathcal{N}(\mathbf{0},\mathbf{I}) = \frac{1}{(2\pi)^{d/2}}\exp\left[-\frac{1}{2}\|\boldsymbol{x}\|^2\right] \tag{7}$$

を考える．これを用いて以下の性質を導出することができる．

［性質 1］　原点を中心とした半径 r の球領域 $\|\boldsymbol{x}\|^2 \leqq r^2$ での $p_{\mathrm{norm}}(\boldsymbol{x})$ の積分を考える．与えられた θ $(0<\theta<1/2)$ に対し，積分値が $1-2\theta$ になるような半径を $\tilde{r}_\theta$ と定義する．すなわち，

$$\int_{\|\boldsymbol{x}\|^2\leqq\tilde{r}_\theta^2} p_{\mathrm{norm}}(\boldsymbol{x})d\boldsymbol{x} = 1-2\theta \tag{8}$$

である．このとき，与えられた θ に対して以下の式が成り立つ．

$$r_\theta = \tilde{r}_\theta \tag{9}$$

上記の性質が成り立つことの証明は [9] を参照されたい．この性質により言えることは，与えられた θ に対して，式 (8) に基づいて $\tilde{r}_\theta$ を計算すれば，その値がそのまま θ 領域を定めるために必要な r_θ となっているということである．具体的にどのようにして r_θ を求めるかについては後述する．

楕円体の形状を持つ θ 領域を直接フィルタリングに利用することは難しいため，図 2 に示すように，θ 領域に外接する矩形を用いることにする．この包囲矩形は，分布の平均 $\boldsymbol{q}$ から i 番目の次元について大小方向に w_i の幅を持つものとする．w_i に関して以下の性質が成り立つ．証明は [9] を参照されたい．

［性質 2］　w_i $(i=1,2,\ldots,d)$ の値は

$$w_i = \sigma_i r_\theta \tag{10}$$

として与えられる．ただし，σ_i は i 番目の次元に関する標準偏差に相当し，$(\boldsymbol{\Sigma})_{ii}$ を共分散行列 $\boldsymbol{\Sigma}$ の i 行 i 列の値としたとき，

$$\sigma_i = \sqrt{(\boldsymbol{\Sigma})_{ii}} \tag{11}$$

として定義される．

図 3 を用いて問合せ戦略 1 のアイデアを説明する．図の陰影部分が θ 領域であり，外接する矩形によって包囲されている．本戦略のアイデアは，ボロノイ領域が包囲矩形と重なりを持たないオブジェクトは解にならないという事実に基づいている．図の場合，a,c,d,f,g が候補オブジェクトとなり，残りについては除去することができる．その理由は以下の通りである．まず，θ 領域の定義から，包囲矩形の外側の領域全体での $p_q(\boldsymbol{x})$ の積分値は $1-(1-2\theta)=2\theta$ 未満である．また，$p_q(\boldsymbol{x})$ は分布の中心 $\boldsymbol{q}$ について点対称な分布であるため，$\boldsymbol{q}$ についてボロノ

イ領域 V_o と対称な領域 V_o' で $p_q(\boldsymbol{x})$ の積分を行うと，その値は V_o での積分値と等しくなる．これらの事実を踏まえると，包囲矩形と重なりを持たないボロノイ領域での $p_q(\boldsymbol{x})$ の積分値は 2 倍しても 2θ 未満であるということになる．すなわち，ボロノイ領域が包囲矩形と重なりを持たないオブジェクトは $\Pr_{NN}(\cdot)$ が θ を超えることはないとして除去できる．

問合せ戦略 1 の処理の流れは以下の通りである．まず，ボロノイ領域が包囲矩形と重なりを持つオブジェクトを検索し候補オブジェクトとする．次に，すべての候補オブジェクトに対して数値積分により $\Pr_{NN}(\cdot)$ を求め，θ 以上であれば出力する．

最後に，与えられた θ に対応する r_θ をいかにして得るかという問題を検討する必要がある．性質 1 より，式 (7) に示した $p_{\mathrm{norm}}(\boldsymbol{x})$ を球領域で積分することによって r_θ が求められるということがわかっている．しかしながら，$p_{\mathrm{norm}}(\boldsymbol{x})$ の積分値は解析的に求めることができないため，θ から直接 r_θ を計算することは不可能である．そこで逆に，適当な半径の値を選んでその半径を持つ球領域での $p_{\mathrm{norm}}(\boldsymbol{x})$ の積分値を数値積分によって計算するということを，様々な半径の値に対して行うことで，積分値から得られる θ とそのときの半径 r_θ の対応表を事前に作成しておくことにする．この表を引くことで与えられた θ に対応する r_θ を素早く得ることが可能となる．このようなアイデアは [13] でも導入されており，表は *U-catalog* と呼ばれている．U-catalog を引く際に注意すべきこととして，与えられた θ に一致するエントリが U-catalog 中に存在しない場合があるということが挙げられる．このような場合には，$\theta^* < \theta$ を満たす最大の θ^* を持つエントリの r_θ の値 r_θ^* を用いればよい．一致するエントリが存在した場合に比べて多少余分に候補オブジェクトが検索されることになるが，結果の正しさは保証される．

問合せ戦略 1 のアルゴリズムをアルゴリズム 1 に示す．ただし，U-catalog の作成と各問合せ対象オブジェクトの座標およびボロノイ領域の情報のファイルへの記録を事前に行っておくものとする．4 行目の関数 catalog_lookup は U-catalog を用いて適切な r_θ を返す．与えられた θ と一致するエントリが U-catalog 中に存在しない場合は，前述の通り，結果の正しさが保証されるような近似値 r_θ^* を返す．13 行目の条件が満たされると，その時点で残っている候補については $\Pr_{NN}(\cdot)$ が θ 以上である可能性がなくなるため，処理を終了することができる．

4.3 問合せ戦略 2

問合せ戦略 2 では，各問合せ対象オブジェクトに対して $\Pr_{NN}(\cdot)$ の上限値を求めることによりフィルタリングを行う．上限値の計算は，ボロノイ領域の**最小包含球**（smallest enclosing sphere, SES）（2 次元の場合は最小包含円）を利用して行う．例としてボロノイ領域 V_e の最小包含球を図 4 に示す．最小包含球の領域で $p_q(\boldsymbol{x})$ を積分すると，その積分値は $\Pr_{NN}(\cdot)$ の上限値とみなすことができる．球領域での積分値は事前に表を作成しておくことで簡単に見積もることができるため，最小包含球による上限値の計算は高速なフィルタリング処理を実現する上で効果的である．本戦略の詳細を以下で説明する．まず，式 (1) の共分散行列 $\boldsymbol{\Sigma}$ が単位行列であるという単純な場合を考え，次に，アイデアを一般の場合に拡張する．

アルゴリズム 1 問合せ戦略 1 に基づく確率的最近傍問合せ

```
1: procedure PNNQ-1(q, Σ, θ)
2:     C ← ∅, sum ← 0
3:     σ_i (i = 1, ..., d) を Σ から計算
4:     r_θ ← catalog_lookup(θ)
5:     {σ_i}_{i=1}^d および r_θ を用いて図 2 に示した包囲矩形を導出
6:     ボロノイ領域が包囲矩形と重なりを持つオブジェクトを検索して C に挿入
7:     foreach o ∈ C do
8:         Pr_NN(q, o) ← ∫_{x∈V_o} p_q(x)dx        ▷ 数値積分により計算
9:         sum ← sum + Pr_NN(q, o)
10:        if Pr_NN(q, o) ≧ θ then
11:            output o
12:        end if
13:        if sum > 1 − θ then
14:            return
15:        end if
16:    end for
17: end procedure
```

4.3.1 $\boldsymbol{\Sigma} = \mathbf{I}$ の場合

本節では式 (1) の共分散行列 $\boldsymbol{\Sigma}$ が単位行列である場合について考える．この場合の $p_q(\boldsymbol{x})$ は，$p_{\mathrm{norm}}(\boldsymbol{x})$ を $\boldsymbol{q}$ が中心となるように平行移動したものに等しい．

最小包含球はその半径も中心の座標もオブジェクトごとに様々であるため，異なる最小包含球に対して，その領域での $p_q(\boldsymbol{x})$ の積分値を素早く導出できるように，表を事前に作成しておく．表の作成にあたり，図 5 に示すような，原点から距離 α の点を中心とする半径 δ の d 次元の球 R を考える．このとき，$p_{\mathrm{norm}}(\boldsymbol{x})$ を R の領域で積分した値を以下の式で表す．

$$\pi(\alpha, \delta) = \int_{\boldsymbol{x} \in R} p_{\mathrm{norm}}(\boldsymbol{x}) d\boldsymbol{x} \tag{12}$$

$p_{\mathrm{norm}}(\boldsymbol{x})$ の等確率面は球形であり，原点からの距離と半径がともに等しい任意の球領域での積分値が一定であるため，このような表記を用いることができる．異なる α と δ の値の組合せに対して，数値積分により $\pi(\alpha, \delta)$ を計算し，図 6 のような表に結果を格納する．この表は戦略 1 で用いた表と同様に *U-catalog* と呼ばれ，(α, δ) のペアが与えられると対応する積分値を返す．

次に，U-catalog の使用方法について説明する．問合せ対象オブジェクト o の $\Pr_{NN}(q, o)$ の上限値を求めるためには，ボロノイ領域 V_o の最小包含球を SES_o として $\int_{\boldsymbol{x} \in SES_o} p_q(\boldsymbol{x}) d\boldsymbol{x}$ の値を計算すればよい．この値は，$\boldsymbol{q}$ から SES_o の中心までの距離を α_o，SES_o の半径を δ_o としたときの $\pi(\alpha_o, \delta_o)$ に等しいため，(α_o, δ_o) に一致するエントリを U-catalog から検索すれば簡単に得られる．得られた値が θ 以下である場合には，o を棄却できる．一方，得られた値が θ より大きいからといって，o が問合せを満たすとは限らない．なぜなら，SES_o は V_o よりも体積が大きいため，実際には問合せを満たさないオブジェクトであっても θ 以上の値が得られる場合が存在するからである．そのため，U-catalog を引いて得られた値が θ 以上であるオブジェクトは候補オブジェクトとして残す．

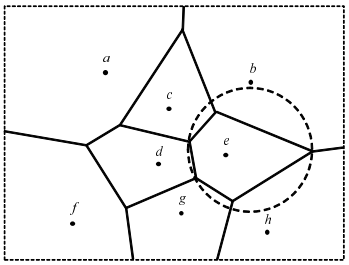

図 4　ボロノイ領域 V_e の最小包含球

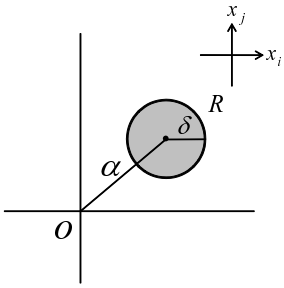

図 5　d 次元の球 R

α	δ	$\pi(\alpha,\delta)$
0.0	0.1	$\cdots$
0.0	0.2	$\cdots$
$\vdots$	$\vdots$	$\vdots$
1.0	0.1	$\cdots$
$\vdots$	$\vdots$	$\vdots$

図 6　問合せ戦略 2 の U-catalog

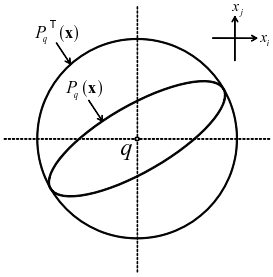

図 7　$p_q^\top(\boldsymbol{x})$ のイメージ

U-catalog のエントリ数は有限であるため，問合せ戦略 1 の場合と同様に，与えられた (α_o, δ_o) に一致するエントリが存在しないことがある．このような場合には，$\alpha_o^* \leqq \alpha_o$ かつ $\delta_o^* \geqq \delta_o$ を満たすような (α_o^*, δ_o^*) を持つエントリのうちで，$\pi(\alpha, \delta)$ の値が最小のものを見つける．つまり，(α_o, δ_o) に対応する積分値よりは大きいが，できる限りそれに近い値を返すようなエントリを見つける．一致するエントリが存在した場合に比べて候補オブジェクトとして残る可能性が高くなるが，結果の正しさは保証される．次に，これまで説明したアイデアを一般化する．

4.3.2　一般の場合

本節では式 (1) の共分散行列 $\boldsymbol{\Sigma}$ が任意である場合について考える．この場合，$p_q(\boldsymbol{x})$ は楕円体形状の等確率面を持つため，$\boldsymbol{\Sigma} = \mathbf{I}$ の場合のように単純に (α_o, δ_o) のペアによって任意の球領域での積分値を表現するというようなことは不可能であり，表を用いて積分値を求めるわけにはいかない．そこで，$p_q(\boldsymbol{x})$ の上限の関数 $p_q^\top(\boldsymbol{x})$ を導入する．

［定義 3］ **上限の関数**

共分散行列の逆行列 $\boldsymbol{\Sigma}^{-1}$ のスペクトル分解を

$$\boldsymbol{\Sigma}^{-1} = \sum_{i=1}^{d} \lambda_i \boldsymbol{v}_i \boldsymbol{v}_i^t \tag{13}$$

と表す．ただし，λ_i と $\boldsymbol{v}_i$ はそれぞれ i 番目の固有値と固有ベクトルである．このとき，

$$\lambda^\top = \min\{\lambda_i\} \tag{14}$$

と定義する．共分散行列の固有値はすべて 0 より大きいため，$\lambda^\top > 0$ が成り立つことに注意する．ここで，行列 $\mathbf{M}^\top$ を

$$\mathbf{M}^\top = \lambda^\top \sum_{i=1}^{d} \boldsymbol{v}_i \boldsymbol{v}_i^t = \lambda^\top \mathbf{I} \tag{15}$$

と定義したとき，式 (1) の $\boldsymbol{\Sigma}^{-1}$ を $\mathbf{M}^\top$ で置き換えることで得られる関数を $p_q^\top(\boldsymbol{x})$ と定義する．

$$p_q^\top(\boldsymbol{x}) = \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left[-\frac{\lambda^\top}{2}\|\boldsymbol{x} - \boldsymbol{q}\|^2\right] \tag{16}$$

$p_q^\top(\boldsymbol{x})$ の等確率面は球形である．ただし，空間全体での $p_q^\top(\boldsymbol{x})$ の積分値は 1 ではないため，厳密には $p_q^\top(\boldsymbol{x})$ は確率密度関数ではないことに注意する．$p_q^\top(\boldsymbol{x})$ は以下のような性質を持つ．

［性質 3］ 任意の $\boldsymbol{x}$ に対して，以下の式が成り立つ．

$$p_q(\boldsymbol{x}) \leqq p_q^\top(\boldsymbol{x}) \tag{17}$$

この性質を満たし，等確率面が球形の関数のうちで最良のものが $p_q^\top(\boldsymbol{x})$ である．つまり，$p_q^\top(\boldsymbol{x})$ は $p_q(\boldsymbol{x})$ の上限を与える．図 7 に同じ確率に対する $p_q(\boldsymbol{x})$ と $p_q^\top(\boldsymbol{x})$ の等確率面を示す．$p_q^\top(\boldsymbol{x})$ については等確率面が球形であるため，表を用いて任意の球領域での積分値を求めることができる．その上，表は 4.3.1 節で作成した U-Catalog をそのまま使用すればよい．具体的には，$(\alpha_o\sqrt{\lambda^\top}, \delta_o\sqrt{\lambda^\top})$ に一致するエントリを U-catalog から検索し，得られた $\pi(\alpha_o\sqrt{\lambda^\top}, \delta_o\sqrt{\lambda^\top})$ を $(\lambda^\top)^{d/2}|\boldsymbol{\Sigma}|^{1/2}$ で割ることで求められる．証明は [15] を参照されたい．性質 3 より，同じ領域で積分した場合に，$p_q^\top(\boldsymbol{x})$ の積分値が $p_q(\boldsymbol{x})$ のそれを下回ることはないため，最小包含球の領域での $p_q^\top(\boldsymbol{x})$ の積分値が θ より小さいオブジェクトは問合せを満たす可能性がないとして棄却できる．与えられた $(\alpha_o\sqrt{\lambda^\top}, \delta_o\sqrt{\lambda^\top})$ に一致するエントリが U-catalog 中に存在しない場合は，4.3.1 節で説明した通り，$(\alpha_o\sqrt{\lambda^\top}, \delta_o\sqrt{\lambda^\top})$ に対応する積分値よりは大きいが，できる限りそれに近い値を返すようなエントリを見つける．

問合せ戦略 2 のアルゴリズムをアルゴリズム 2 に示す．ただし，U-catalog の作成と各問合せ対象オブジェクトの座標，ボロノイ領域の情報，最小包含球の情報（中心点，半径）のファイルへの記録を事前に行っておくものとする．6 行目の関数 catalog_lookup は，$\boldsymbol{q}$ から SES_o の中心までの距離を α_o，SES_o の半径を δ_o として，$(\alpha_o\sqrt{\lambda^\top}, \delta_o\sqrt{\lambda^\top})$ に一致するエントリを U-catalog から検索し，$\pi(\alpha_o\sqrt{\lambda^\top}, \delta_o\sqrt{\lambda^\top})$ を返す．一致するエントリが U-catalog 中に存在しない場合は，前述の通り，結果の正しさが保証されるような近似値を返す．13 行目のソートにより，最小包含球の領域での $p_q^\top(\boldsymbol{x})$ の積分値が大きいオブジェクトから順に $\Pr_{NN}(\cdot)$ を計算できるようにしている．この値が大きいオブジェクトはボロノイ領域での $p_q(\boldsymbol{x})$ の積分値，すなわち $\Pr_{NN}(\cdot)$ も大きいと考えられるため，順序を考慮しない場合よりも問合せ処理を早く終了できる可能性が高い．

5. 評価実験

5.1 実験方法

実験には，米国加州ロングビーチの道路の線分データ [3] から各線分の中点を抽出したデータを使用した．データは $[0, 1000]^2$ の 2 次元空間上に位置するように正規化された 50,747 個の点から成る．各点を問合せ対象オブジェクトとして，2 つの問合せ戦略にそれらのハイブリッド戦略を加えた 3 つの戦略を対象に，$PNNQ(q, \theta)$ に対する性能評価を行った．ハイブリッド戦

アルゴリズム 2 問合せ戦略 2 に基づく確率的最近傍問合せ

```
1: procedure PNNQ-2(q, Σ, θ)
2:     C ← ∅, sum ← 0
3:     λ^⊤ および |Σ| を Σ から計算
4:     foreach o ∈ O do
5:         q から SES_o の中心までの距離 α_o を計算
6:         π(α_o√λ^⊤, δ_o√λ^⊤) ← catalog_lookup(α_o√λ^⊤, δ_o√λ^⊤)
7:         IV_SES_o ← π(α_o√λ^⊤, δ_o√λ^⊤)/(λ^⊤)^{d/2}|Σ|^{1/2}
8:                                   ▷ SES_o での p_q^⊤(x) の積分値
9:         if IV_SES_o > θ then
10:            C ← C ∪ {o}
11:        end if
12:    end for
13:    C 中のオブジェクトを IV_SES_o の降順でソート
14:    foreach o ∈ C do                          ▷ 先頭から順に
15:        Pr_NN(q, o) ← ∫_{x∈V_o} p_q(x)dx      ▷ 数値積分により計算
16:        sum ← sum + Pr_NN(q, o)
17:        if Pr_NN(q, o) ≧ θ then
18:            output o
19:        end if
20:        if sum > 1 − θ then
21:            return
22:        end if
23:    end for
24: end procedure
```

略は，始めに戦略 1 のフィルタリングを行い，残ったオブジェクトに対して戦略 2 のフィルタリングを行う戦略である．

式 (1) の共分散行列 $\mathbf{\Sigma}$ は以下のように設定した．

$$\mathbf{\Sigma} = \gamma \begin{bmatrix} 7 & 2\sqrt{3} \\ 2\sqrt{3} & 3 \end{bmatrix}$$

これにより，$p_q(\boldsymbol{x})$ の等確率面の形状は長軸と短軸の比が 3 : 1 で傾き 30° の楕円となる．係数 γ は分布の曖昧さの程度に対応する．この実験では，$\gamma = 10, \theta = 0.01$ を標準の設定とし，そこから値を変化させることで，γ および θ が各戦略の性能に与える影響を調べた．また，$\mathbf{\Sigma}$ を変えることで，$p_q(\boldsymbol{x})$ の等確率面の形状が異なる場合についても評価を行った．性能の評価基準には，10 回の問合せ処理の平均応答時間を用いた．

今回の実験に用いた問合せ処理プログラムでは，ボロノイ領域や最小包含球の計算などに LEDA 6.1 を使用した．LEDA [1] は，グラフ理論や幾何学計算などの分野における効率的なデータ構造とアルゴリズムを提供する C++のクラスライブラリである．また，数値積分処理には RANDLIB [2] という C 言語の乱数生成ライブラリを用いた．具体的には，RANDLIB により正規分布の確率密度関数に従って大量の乱数を生成し，それぞれの乱数がボロノイ領域内に位置しているかどうかを LEDA で提供されている関数を利用して調べた．ボロノイ領域内に位置していた乱数の個数の比率を求めれば，その比率が求める確率の推定値に相当している．この手法は**重点サンプリング法** [12] と呼ばれ，モンテカルロ法の一種であるが，通常のモンテカルロ法による計算より高速である．今回の実験では，1 回の積分計算に対して 1,000,000 個の乱数を発生させて積分値を求めるように設定した．

戦略 2 で使用する U-Catalog の作成について，登録される α および δ の値の間隔はそれぞれもう一方の値によって変化しないようにした．これは，例えば，α の値が小さいときには δ の値を密にとり，大きいときにはまばらにとるというようなサンプリング方法で U-Catalog を作成した場合，その中から最適なエントリを素早く見つけることが難しくなるためである．このようなサンプリング方法で作成した U-Catalog は効率が良い（少ないエントリ数でも多くのオブジェクトを除去できる）という利点を持つが，予備実験の結果から今回の実験では前述の制約の下で作成した U-Catalog を使用して問合せを行うことにした．使用した U-Catalog の総エントリ数は 30,925 であった．

実験用問合せ処理プログラムの開発は C++を用いて行った．実験に使用したマシンの CPU は Intel Core 2 Duo E8500 (3.16GHz)，メモリは 4GB，OS は Fedora 10 である．

5.2 実験結果

5.2.1 標準の設定の場合

標準の設定（$\gamma = 10, \theta = 0.01$）における，各戦略の応答時間を図 11 に，候補オブジェクトの個数を表 1 に示す．解として出力されたオブジェクトは 26 個であった．また，ある問合せにおける各戦略の候補オブジェクトを図 8，9，10 に示す．中心の小さな白い円は分布の平均 $\boldsymbol{q}$ を表している．やや濃い色の線でボロノイ領域が縁取られているオブジェクトが候補オブジェクトであり，ボロノイ領域が黒く塗りつぶされているオブジェクトが解オブジェクトである．図 8，10 における矩形は，戦略 1 のフィルタリングに用いる θ 領域の包囲矩形である．

図 11 に示したように，各戦略とも処理時間の大部分は $\Pr_{NN}(\cdot)$ を求めるための数値積分に費やされていた．そのため，基本的にはフィルタリングによってより多くのオブジェクトを除去できる，すなわち候補オブジェクトの少ない戦略ほど性能が良くなる．表 1 より，戦略 1 および戦略 2 をそれぞれ単独で用いた場合の候補オブジェクト数は，戦略 1 が 179 個，戦略 2 が 150 個であるが，ハイブリッド戦略では 129 個にまで減っていることがわかる．これは，戦略 1 と戦略 2 に共通の候補オブジェクトがハイブリッド戦略の候補オブジェクトとなるからである．

この実験では 26 個という少数の解オブジェクトを返すのにハイブリッド戦略の場合でも約 35 秒かかっており，処理時間が長過ぎるように思われる．これを短縮するための最も効果的な方法は数値積分に要する時間を削減することであり，数値積分に使用する乱数の個数を減らせばこれを達成できることは明らかである．例えば，今回の実験では 1 回の数値積分に 1,000,000 個の乱数を用いているが，これを 100,000 個にすることで，計算の精度は悪化するものの積分計算に要する時間をおよそ 1/10 にまで削減できる．今回の実験では，標準の設定として $\theta = 0.01$ という小さな閾値を用いたため，比較的高い精度が必要となり，様々な試行に基づいてサンプル数を 1,000,000 個に設定した．しかし，閾値がより大きい場合（例：$\theta = 0.1$）には，精度を落とすことなくサンプル数を減らすことができる．

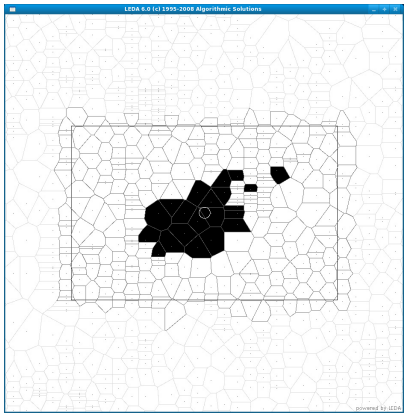

図 8　戦略 1 における候補オブジェクト

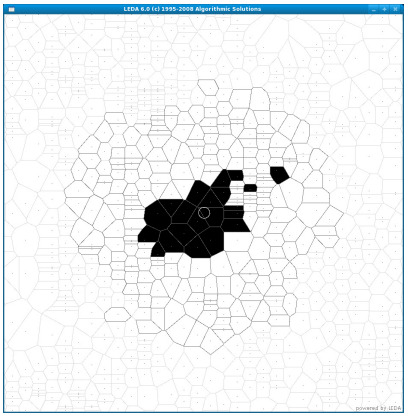

図 9　戦略 2 における候補オブジェクト

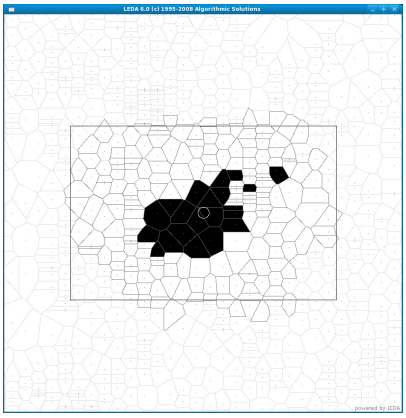

図 10　戦略 1+2 における候補オブジェクト

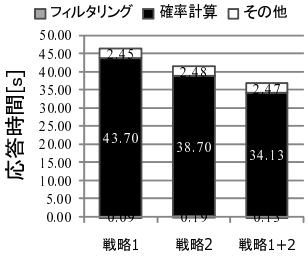

図 11　応答時間（$\gamma=10, \theta=0.01$）

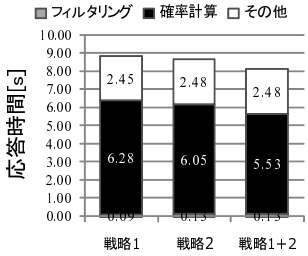

図 12　応答時間（$\gamma = 1$）

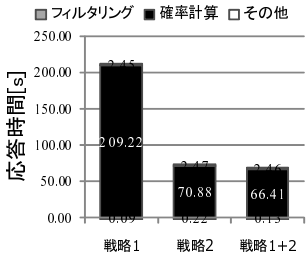

図 13　応答時間（$\gamma = 50$）

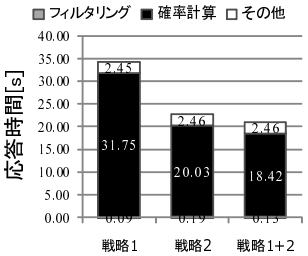

図 14　応答時間（$\theta = 0.03$）

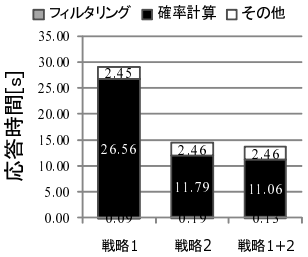

図 15　応答時間（$\theta = 0.05$）

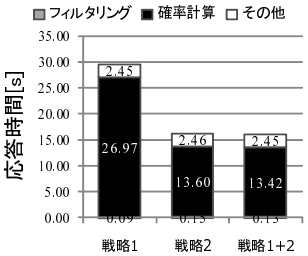

図 16　応答時間（円）

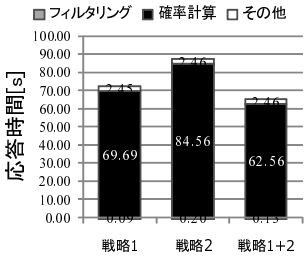

図 17　応答時間（細長い楕円）

表 1　候補数（$\gamma=10, \theta=0.01$）

戦略 1	戦略 2	戦略 1+2
179	150	129

表 2　候補数（$\gamma = 1$）

戦略 1	戦略 2	戦略 1+2
24	31	23

表 3　候補数（$\gamma = 50$）

戦略 1	戦略 2	戦略 1+2
847	276	260

表 4　候補数（$\theta = 0.03$）

戦略 1	戦略 2	戦略 1+2
128	76	68

表 5　候補数（$\theta = 0.05$）

戦略 1	戦略 2	戦略 1+2
107	44	40

表 6　候補数（円）

戦略 1	戦略 2	戦略 1+2
115	50	49

表 7　候補数（細長い楕円）

戦略 1	戦略 2	戦略 1+2
276	366	250

5. 2. 2　γ を変化させた場合（$\gamma = 1, \gamma = 50$ の場合）

標準の設定では共分散行列 $\boldsymbol{\Sigma}$ の係数 γ の値を $\gamma = 10$ としていたが，この実験では $\gamma = 1$ と $\gamma = 50$ の場合について調べた．γ を変化させると，$p_q(\boldsymbol{x})$ の等確率面の形状は変わらずに大きさが変わる．具体的には，γ を大きくすると等確率面の大きさも大きくなる．等確率面の大きさは問合せオブジェクトの位置の曖昧さの程度を表しているため，$\gamma = 10$ の場合に比べて，$\gamma = 1$ の場合には問合せオブジェクトの位置が正確になり，逆に，$\gamma = 50$ の場合にはさらに曖昧になる．$\gamma = 1$ の場合および $\gamma = 50$ の場合における，各戦略の応答時間をそれぞれ図 12，13 に，候補オブジェクトの個数をそれぞれ表 2，3 に示す．解として出力されたオブジェクトは，$\gamma = 1$ の場合が 8 個，$\gamma = 50$ の場合が 15 個であった．

図 12，11，13 より，γ が大きいほど，戦略 2 の戦略 1 に対する優位性が高くなっているのがわかる．これは，表 2，1，3 からわかるように，γ が大きいほど，戦略 2 は戦略 1 に比べてより多くのオブジェクトを除去できるようになるためである．

$\gamma = 1$ の場合について，表 2 を見ると，戦略 2 の候補オブジェクトは戦略 1 よりも多いことから，戦略 2 は戦略 1 よりも確率計算に要する時間が長くなるように思われる．しかしながら，実際には逆の結果となった．これは，アルゴリズム 2 の 13 行目に示した通り，戦略 2 ではフィルタリングに用いた最小包

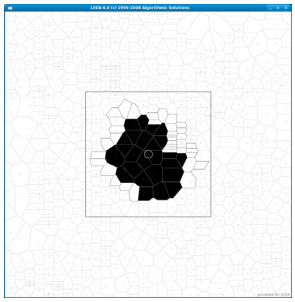

図 18　戦略 1+2 における候補オブジェクト（円）

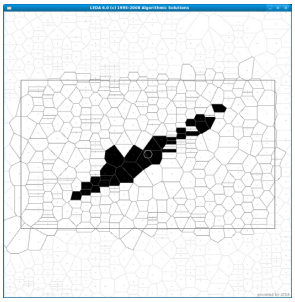

図 19　戦略 1+2 における候補オブジェクト（細長い楕円）

含球の領域での $p_q^{\top}(\boldsymbol{x})$ の積分値の降順で候補オブジェクトのソートを行うためである．問合せを満たす可能性が高そうな候補から順に確率を計算できるため，順序を考慮しない戦略 1 に比べて早く確率計算処理を終了させることができたのである．

5.2.3　θ を変化させた場合（$\theta = 0.03, \theta = 0.05$ の場合）

標準の設定では $\theta = 0.01$ としていたが，この実験では $\theta = 0.03$ と $\theta = 0.05$ の場合について調べた．各戦略の応答時間をそれぞれ図 14，15 に，候補オブジェクトの個数をそれぞれ表 4，5 に示す．解として出力されたオブジェクトは，$\theta = 0.03$ の場合が 7 個，$\theta = 0.05$ の場合が 3 個であった．閾値が大きくなるほど解オブジェクトが少なくなっているのは，確率的最近傍問合せの定義を考えれば当然のことである．

図 11，14，15 より，θ が大きいほど，戦略 2 の戦略 1 に対する優位性が高くなっているのがわかる．これは，表 1，4，5 からわかるように，θ が大きいほど，戦略 2 は戦略 1 に比べてより多くのオブジェクトを除去できるようになるためである．

5.2.4　$p_q(\boldsymbol{x})$ の等確率面の形状を変化させた場合

標準の設定では $p_q(\boldsymbol{x})$ の等確率面の形状を，長軸と短軸の比が 3 : 1 で傾き 30° の楕円としていたが，この実験では共分散行列 $\boldsymbol{\Sigma}$ を変化させることで，$p_q(\boldsymbol{x})$ の等確率面の形状が円の場合と細長い楕円（長軸と短軸の比が 9 : 1 で傾き 30° の楕円）の場合について調べた．各戦略の応答時間をそれぞれ図 16，17 に，候補オブジェクトの個数をそれぞれ表 6，7 に示す．解として出力されたオブジェクトは，前者の場合が 26 個，後者の場合が 24 個であった．また，ある問合せにおけるハイブリッド戦略の候補オブジェクトをそれぞれ図 18，19 に示す．

図 16，17 より，等確率面の形状が円である場合には戦略 1 に比べて戦略 2 の方が性能が良くなっており，等確率面の楕円形状を細長くした場合には，その逆の優劣関係になっていることがわかる．これは，表 1，6，7 からわかるように，正規分布の楕円形状が細長くなるほど，戦略 1 は戦略 2 に比べてより多くのオブジェクトを除去できるようになるためである．

6. ま と め

本研究では，位置が正規分布に従う問合せオブジェクトが最近傍問合せを発行するという状況を対象とし，閾値を導入するなどして通常の最近傍問合せを拡張した確率的最近傍問合せの効率的な処理手法を提案した．本手法では，数値積分によって正確に最近傍オブジェクトとなる確率を求めるまでもなく明らかにその値が閾値より小さいといえるオブジェクトを除去することで計算コストを削減した．このアプローチに基づく具体的な問合せ戦略として，θ 領域に基づく戦略である問合せ戦略 1 と，最小包含球と上限の関数を利用した戦略である問合せ戦略 2 を提案した．

実験では，2 つの戦略にそれらのハイブリッド戦略を加えた 3 つの戦略について，様々なパラメータ設定の下で比較を行った．その結果，問合せオブジェクトの位置の曖昧さが大きい，閾値が高い，正規分布の等確率面の楕円形状が円形に近い，というような状況では戦略 2 の戦略 1 に対する優位性が向上し，その逆の状況では戦略 1 の戦略 2 に対する優位性が向上することがわかった．ただし，実用性の観点からすると，2 つの戦略の各利点を受け継ぎ，3 つの戦略のうちで最も良い性能を示したハイブリッド戦略によって問合せ処理を行うのが良いだろう．

今後の課題としては，高次元の場合を対象とした実験や k-最近傍問合せへの拡張が挙げられる．

謝　　辞

本研究の一部は，文部科学省科学研究費（19024037, 19300027, 18200005）の助成による．

文　　献

[1] LEDA. http://www.algorithmic-solutions.com/leda/.
[2] RANDLIB. http://biostatistics.mdanderson.org/SoftwareDownload/.
[3] TIGER. http://tiger.census.gov/.
[4] F. Aurenhammer. Voronoi diagrams: A survey of a fundamental geometric data structure. *ACM Computing Surveys*, 23(3):345–405, 1991.
[5] G. Beskales, M. A. Soliman, and I. F. Ilyas. Efficient search for the top-k probable nearest neighbors in uncertain databases. In *Proc. VLDB*, pp. 326–339, 2008.
[6] J. Chen and R. Cheng. Efficient evaluation of imprecise location-dependent queries. In *Proc. ICDE*, pp. 586–595, 2007.
[7] R. Cheng, J. Chen, M. Mokbel, and C.-Y. Chow. Probabilistic verifiers: Evaluating constrained nearest-neighbor queries over uncertain data. In *Proc. ICDE*, pp. 973–982, 2008.
[8] R. Cheng, D. V. Kalashnikov, and S. Prabhakar. Querying imprecise data in moving object environments. *IEEE TKDE*, 16(9):1112–1127, 2004.
[9] Y. Ishikawa, Y. Iijima, and J. X. Yu. Spatial range querying for gaussian-based imprecise query objects. In *Proc. ICDE*, 2009 (to appear).
[10] H.-P. Kriegel, P. Kunath, and M. Renz. Probabilistic nearest-neighbor query on uncertain objects. In *Proc. DASFAA*, pp. 337–348, 2007.
[11] D. Pfoser and C. S. Jensen. Capturing the uncertainty of moving-object representations. In *Proc. 6th Intl. Symp. on Advances in Spatial Databases (SSD'99)*, pp. 111–131, 1999.
[12] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipies: The Art of Scientific Computing*. Cambridge University Press, 3rd edition, 2007.
[13] Y. Tao, X. Xiao, and R. Cheng. Range search on multidimensional uncertain data. *ACM TODS*, 32(3), 2007.
[14] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. The MIT Press, 2005.
[15] 飯島, 石川. 曖昧な位置に基づく最近傍問合せ処理手法. 情報処理学会研究報告, 2008-DBS-146-40, pp. 235–240, 2008.