

M-022

GeoSkip : 位置情報を扱うオーバレイネットワーク構造の提案

GeoSkip : Proposal of Overlay Network Structure for Managing Location Information

中村 陽一†
Yoichi Nakamura斉藤 裕樹‡
Hiroki Saito戸辺 義人‡*
Yoshito Tobe

1. はじめに

センサネットワーク技術の発展により Location-based Service (LBS)が注目されている。実世界の位置依存情報を扱う必要性と膨大な端末やセンサからの情報を分散管理する必要性から、本研究では、位置依存情報に適した高いスケーラビリティを実現可能なオーバレイネットワークの構成手法 GeoSkip を提案する。GeoSkip は、1次元の情報を扱う従来の SkipGraph[1] の概念を応用し、2次元に拡張することで位置依存データの効率的な分散配置、検索を行う。

2. GeoSkip

2.1 位置依存情報の分散

GeoSkip では、位置依存情報を扱うため、2次元座標とその座標に存在する情報を2次元のID平面上に分散配置されたノードで管理を行う。また、従来のDHT[2,3]とは異なり、ハッシュ変換を行わずに SkipGraphs の経路制御を2次元上に拡張することで、情報の連続性を保ったネットワーク構造を保持する。以下に、GeoSkip のネットワーク構造と経路制御について述べる。具体的には、任意の座標が与えられたとき、その座標の情報を管理するノードへ到達するための経路制御手法を説明する。各ノードは、自ノードを基準とした周囲のノードの方角と距離を保持する。方角は、ユークリッド平面上で自ノードを中心に一定の角度 θ ごとに平面を分割したとき、周囲のノードがどの分割平面上にあるかを示す。ここで、 θ は式(1)(2)の条件を満たすように設定する。

$$0 < \theta \leq \frac{\pi}{2} \quad (1)$$

$$\theta \mid \frac{\pi}{2} \quad (2)$$

また分割した各面を反時計回りに順に $S_i (i = 1, 2, \dots, 2\pi/\theta)$ とする。分割平面上での周辺ノードの距離については、次のように経路表を構築する。まず、ノードごとにIDをランダムに割り当てる。経路表では、各分割平面上において最も近いノードへの到達方法(物理アドレス)を保持し、IDの接頭 l 桁 ($l=1, 2, \dots, n$) までが等しくかつ最も近いノードへの到達方法を保持し、その経路をレベル l とする。図1の左に θ を $\pi/6$ としたときの S_1, S_2, S_3 の経路表の様子、右に全体例を示す。

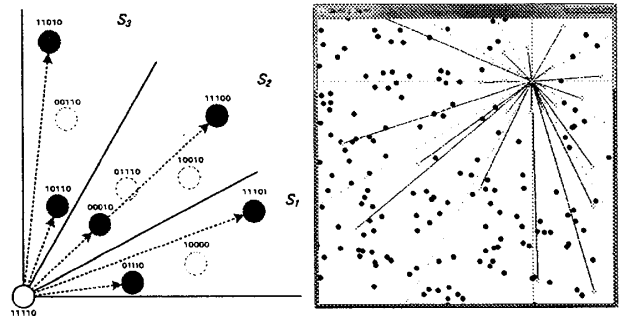
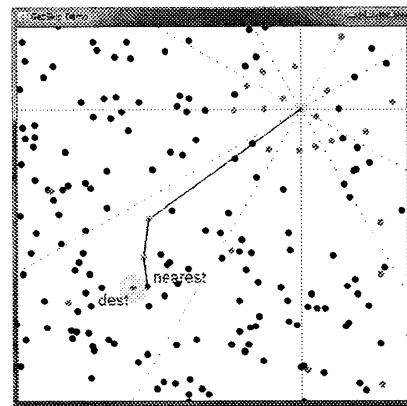
図1 平面分割角 θ を $\pi/6$ としたときの経路表の例

図2 目的地まで転送経路の例

2.2 検索アルゴリズム

検索目的の座標が与えられたとき、その座標の情報を管理するノードへ到達するための検索手法について説明する。GeoSkip では2つの手順を用いる。

近傍ノードへの到達

まず、検索目的となる座標が、経路表のどの分割平面上に入るかを判定する。例えば、 $\theta = \pi/6$ の条件下では、自ノードの座標を (x_1, y_1) とし目的の座標を (x_2, y_2) としたとき、式(3)(4)を満たすときは分割を開始からところから $\pi/6$ までの分割範囲上 (S_i) に目的の座標が存在することになる。

$$(x_2 - x_1) > 0 \quad (3)$$

$$0 \leq \frac{y_2 - y_1}{x_2 - x_1} < \tan(\theta \times 1) \quad (4)$$

次に、その分割平面上の一番高いレベルから順に目的の座標を越えかつ最も近いノードを検索し、このノードに検索要求を転送する。検索要求を受信したノードも同様の経路探索を行いさらに近いノードに転送を行う。この動作を反復して行う。最終的に、自ノードよりも近い

† 東京電機大学大学院工学研究科情報メディア学専攻

‡ 東京電機大学未来科学部情報メディア学科

* 独立行政法人科学技術振興機構 CREST

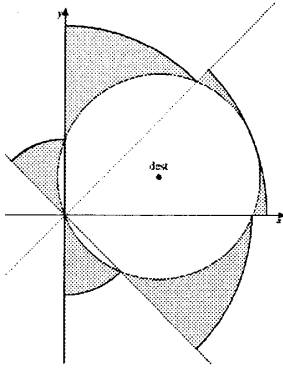


図3 絞り込みの対象となる領域

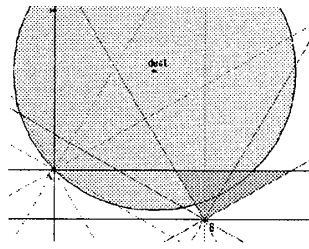


図4 検索要求を送信する対象となる領域

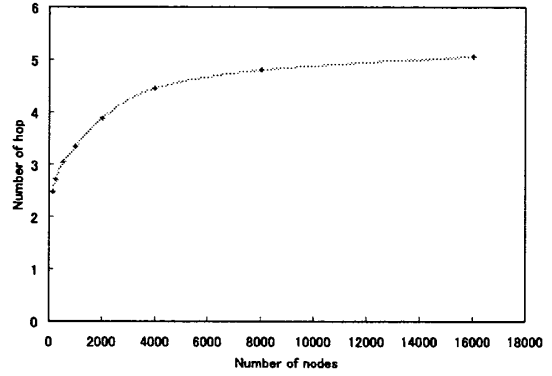


図5 ノード数に対するホップ数の平均

Algorithm 1 B received search request

```

Sba := B.AreaJudgment(A) // Bから見てAのいる領域
Sab := A.AreaJudgment(B) // Aから見てBのいる領域
Sbd := B.AreaJudgment(dest) // Bから見て目的地のいる領域
List: 絞り込み条件を満たしたノードのリスト
for i = 0 to π/2θ do
  if Sb_a - Sv > 0 do
    Node Tmp := B.getRoutingTable(Sbd+i, 0)
    // BのSv-iの経路表のレベルiのノード
    if A.AreaJudgment(Tmp) = Sab do
      List.add(Tmp)
    end if
  end if
  if Sba - Sbd < 0 do
    Node Tmp := B.getRoutingTable(Sbd-i, 0)
    if A.AreaJudgment(Tmp) = Sab do
      List.add(Tmp)
    end if
  end if
end for
if list size = 0 do
  send(A, end)
else
  send(A, List);
end if
    
```

ノードがないと判定された場合、このノードが近傍ノードとなる。図2は目的地までの転送の例である。このように効率良く検索目的の周辺までのホップ数を削減する。

目的座標への絞り込み

上記のステップにより近接ノード（ここではノード A とする）と一時的に判断されたとしても、ノード A の経路表に次のような条件を満たすノード（ここではノード B_i とする）が存在する場合、ノード A の経路表に存在しないが、より検索目的に近いノードが存在する可能性がある。その条件とは、ノード A の分割平面 $S_i (i=1, 2, \dots, 2\pi/\theta)$ において、ノード A からみて目的の座標が存在する領域を S_{ad} 、ノード A の S_i の経路表に入っているノードを B_i 、目的地を D、D を中心とした $d(A, D)$ (A と D との距離) を半径とした円と A を通る傾き $\tan(x \cdot \theta)$ の直線との交点を G_i としたとき、以下 3 つのうちいずれかが成り立つ B_i が存在する場合 (図 3 の斜線の範囲にノードが存在する) である。

- $ad+1 \leq i \leq ad + \pi/(2\theta)$ のとき $d(A, B_i) < d(A, G_{i-1})$ が成り立つ

- $ad - \pi/(2\theta) \leq i \leq ad-1$ のとき $d(A, B_i) < d(A, G_i)$ が成り立つ
- $i = v$ のとき $d(A, B_i) < 2d(A, D)$ が成り立つ

そこで近接するノードが存在するか否かを判定するために絞り込みを行う。上記の条件を満たしたノードに対して反復的に検索要求を転送する。検索要求を受け取ったノード B_i の動作アルゴリズムを Algorithm1 に示す。例えば図 4 の場合、ノード B_i の経路表に斜線の範囲にノードが存在した場合、そのノードの情報をノード A に送信する。そしてノード A はノード B_i から送られてきた情報を元に検索要求を転送する。これを反復的に行うことで絞り込みを行う。

3. 評価

本研究では、提案するアルゴリズムをシミュレーションによって評価を行った。シミュレーションの条件は次の通りである。ノード数は 125, 250, 500, 1000, 2000, 4000, 8000, 16000 の 9 通り、領域サイズは 300×300 、平面分割角度 θ は $\pi/6$ である。

図 5 はノード数ごとのランダムな 2 地点間での検索にかかるホップ数の平均を示す。その結果、ノード数の指数的な増加に対してホップ数は一定の増加に留まっている (16000 ノードで 5 ホップ)。これによりノード数の増加による規模拡張性があることが確認できる。

4. まとめ

我々は、位置依存情報に適した高いスケーラビリティを実現可能なオーバーレイネットワークの構成手法を提案した。今後は、ジオキャストの実現方法や R-tree, Grid といった従来の空間インデックスとの融合などを検討する。

参考文献

[1] James Aspnes, Gauri Shah : "Skip Graphs," Proceedings of ACM SIAM Symposium on Discrete Algorithms, pp.384-393, 2003.
 [2] Stoica, I., Morris, R., Karger, D., Kaashoek, M. F. and Balakrishnan, H.: Chord: A Scalable Peer-to-peer Lookup Protocol for Internet Applications, Proceedings of ACM SIGCOMM 2001, pp.149-160, 2001.
 [3] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker: A Scalable Content Addressable Network, Proceedings of ACM SIGCOMM 2001, pp. 161-172, 2001.