

P2P ネットワークにおける B+木を用いた多次元の範囲探索手法

松浦伸彦[†] 峰野博史[†] 石川憲洋[‡] 水野忠則[†]

[†] 静岡大学情報学部 [‡](株)NTT ドコモ

1 はじめに

近年、無線技術やセンサ技術の発達に伴い、各地にセンサを配置して収集したデータをサービス提供に利用しようとする動きが活発化している。その一例として、LiveE!プロジェクト [1] のように、観測所だけでなく一般家庭などにもセンサを配置し、より細やかなデータ収集を行えるセンサネットワーク展開が行われている。このようなシステムでは、世界各地から集められたセンサデータを取り扱う必要があり、温度や湿度、位置といった様々な属性を組み合わせた探索を効率よく行えることが望まれる。そこで本論文では、多次元の範囲探索を効率よく処理することが可能な B+木を用いた P2P ネットワークの構築手法と、それを利用した多次元探索手法を提案する。

2 関連研究

従来のサーバ集中型アーキテクチャでは、データの管理や探索をサーバ側で処理しなければならず、スケーラビリティ維持が困難という問題がある。このため、P2P ネットワークを用いて複数のノードを自律的に相互接続させるための研究が盛んに行われている。

P2P ネットワークは、従来のネットワーク上に仮想的な P2P 用のネットワーク (オーバーレイネットワーク) を構築してノード間で直接通信を行う。そのためスケーラビリティや負荷分散に優れており、特に膨大なデータを処理するシステムに適している。この構築手法として分散ハッシュテーブル (DHT) が注目されており、代表的な手法に Chord, Tapestry, Kademlia などが存在する。また、範囲探索や多次元探索といった柔軟な探索を行う DHT の研究も行われており、代表的な手法として、Znet [2], LL-Net [3], SkipGraph [4], P-Tree [5] 等がある。

上記手法のうち多次元の範囲探索をサポートするのは Znet と LL-Net だけであり、他手法は多重構築するなどの工夫が必要となる。また、Znet も LL-Net も扱う属性数はシステム設計時に決められており、運用中に属性数を変化させるといった柔軟性は持っていない。そのため、属性数が変化した場合に対応でき、多種多様な属性を組み合わせた探索を処理できるような、より柔軟な探索を行うための仕組みが必要であると考える。

Multi-dimensional Range Search in P2P Networks Using B+-Tree

Nobuhiko Matsuura[†], Hiroshi Mineno[†], Norihiro Ishikawa[†], Tadanori Mizuno[†]

[†]Faculty of Informatics, Shizuoka University

[‡]Service & Solution Development Department, NTT DoCoMo, Inc.

表 1: ノード参加リスト

Node (NodeID)	温度	Node (NodeID)	温度
N_0 (3)	21	N_4 (4)	NULL
N_1 (9)	30	N_5 (2)	29
N_2 (1)	15	N_6 (8)	18
N_3 (0)	27	N_7 (6)	24

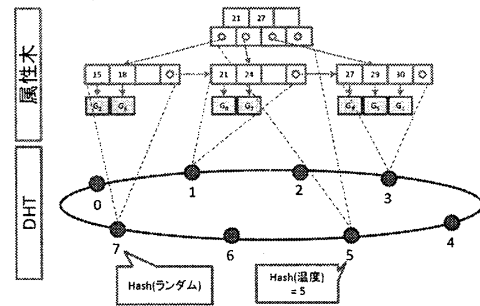


図 1: 属性木の割り当て

3 B+木を用いた多次元範囲探索手法

これまでの研究では、範囲探索や多次元探索といった柔軟な探索を行う様々な手法が考えられてきた。しかしそれら手法は、動的に属性数が変化する場合や任意の属性による探索に対応できない。もし任意の属性による多次元探索が可能であれば、より汎用的なセンサデータ共有システムを作ることができる。そこで温度や湿度といった属性ごとに木構造 (属性木) を構築し、その属性木の節 (属性ノード) を DHT 上のノード (DHT ノード) にマッピングして管理する構造化オーバーレイネットワークを提案する。これにより、新たに属性が増えた場合は属性木を新規に構築することで対応でき、木を平行に探索することで任意の属性で範囲探索を行うことが可能となる。

属性木の構築には、木構造の一種である B+木 [6] を利用する。B+木は、探索オーダが $\log_t N$ (t は属性ノードの要素数に依存する定数) と高速であり、キーの挿入・削除を繰り返しても木のバランスが保たれるという特性を持つ。そのため、常に変化し続けるセンシングデータを格納したとしても、全ての木において安定した高速な範囲探索が可能となっている。属性木の各属性ノードのマッピング方法については、根は属性名から、その他はランダムな文字列からハッシュ値を求め、同じハッシュ値を $NodeID$ として持つ DHT ノードを管理者として選ぶ。以上より、例えば表 1 の N_0 から N_7 までのノードが順にネットワークに参加した場合、温度の属性木は図 1 のように構築、割り当てが行われる。

値を探索する際は、まず根を管理している DHT ノードを調べて探索クエリを送信する。クエリを受け取った DHT

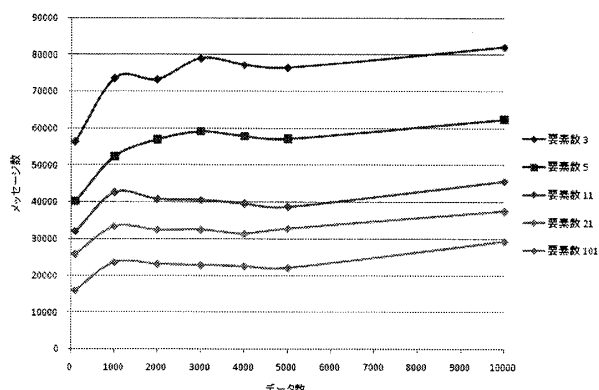


図 2: データ数を変化させた時のメッセージ数

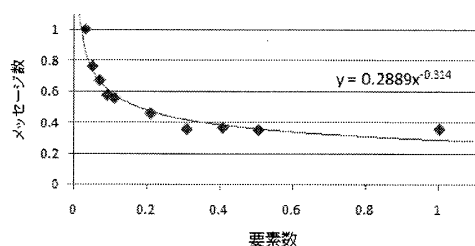


図 3: 要素数を変化させた時のメッセージ数 (正規化)

ノードは、自分が持つ属性ノードとクエリのマッチングを行い、適切な下位の属性ノードを持つ DHT ノードへとクエリの中継を行う。以上を葉の部分に到達するまで繰り返す、葉に到達した場合はクエリに一致する { 値, Node 情報 } を検索結果として返す。例えば図 1 において温度値 21 を探索した場合は、NodeID = 5 の DHT ノードへ探索クエリが送られ、NodeID = 5 → 1 と中継が行われ、探索結果 {21, N₀} が得られる。

4 シミュレーション

提案手法における属性木の要素数決定のため Overlay-Weaver [7] を利用したシミュレーションを行った。提案手法の性能は、属性木を構築している各属性ノードの要素数に非常に大きな影響を受ける。要素数が増加するにつれ、探索コストは良くなるものの、負荷分散は悪くなる。このようにコストと負荷分散のトレードオフ関係を持っており、提案手法の性能評価の前に最適な値を求める必要がある。

本評価では、Chord アルゴリズムに基づいた P2P ネットワークを構築し、提案手法に基づいた属性木を構築したときの探索にかかるメッセージ数を調査する。挿入データ数の増加に伴うメッセージ数の変化を図 2 にしめす。また要素数の増加に伴うメッセージ数の変化を、それぞれの軸の最大値が 1 となるよう各軸の最大値 (要素数 101, メッセージ数 82004) で割って正規化したものを図 3 に示す。なお、それぞれのネットワークサイズは 3000 ノード、図 3 の挿入データ数は 10000 個である。

図 2 よりデータ数の変化に伴いメッセージ数も増加するが、図 3 が示すように要素数の増加による探索コストへの影響は要素数が多くなるにつれ弱くなっていくことが

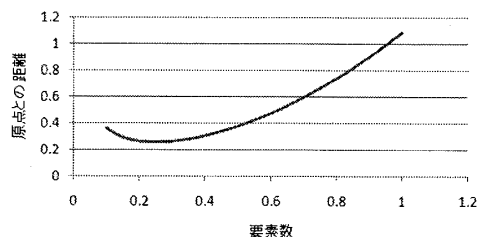


図 4: 原点との距離

分かる。よって要素数の最適な値として、ある程度探索コスト減少効果を望め、できるだけ要素数が少なくなるように、図 3 における原点に一番近い点を採用する。

図より、要素数の変化に伴うメッセージ数の変化は、

$$y = 0.2889 * x^{-0.314} \quad (1)$$

で近似することができる。この式より原点からの距離 r を、

$$r^2 = x^2 + (0.2889 * x^{-0.314})^2 \quad (2)$$

として求めたものを図 4 に示す。図より距離が最小となる要素数 0.25 が求まり、正規化に利用した最大値 101 との積を取ることで最適な要素数 25 が求まる。

5 終わりに

本研究では、オーバレイネットワークと B+木を組み合わせさせた多次元の範囲探索手法を提案し、性能を左右するパラメータである B+木の各節の要素数の最適値を求めた。今後は、最適な要素数である 25 を用いて、他手法との多次元の範囲探索コストを比較して提案手法の有効性の検証を行う。また、PlanetLab のような大規模な実環境での評価ができるテストベッドを利用した評価も行っていく。

参考文献

- [1] LiveE!プロジェクト, <http://www.live-e.org/>.
- [2] Y. Shu, B. Ooi, K. Tan, and A. Zhou, "Supporting Multi-dimensional Range Queries in Peer-to-Peer System", Proc. IEEE P2P, pp.173-180 (2005).
- [3] 金子雄, 福村真哉, 春本要, 下條真司, 西尾章治郎, "モバイル環境における端末の位置情報に基づく p2p ネットワークの提案と評価", 電子情報通信学会第 15 回データ工学ワークショップ (DEWS2004) 論文集 (2004).
- [4] J. Aspnes, and G. Shah, "Skip Graphs", ACM Transactions on Algorithms, vol 3, no 4, p.37 (2007).
- [5] A. Crainiceanu, P. Linga, J. Gehrke, and J. Shanmugasundaram, "Querying Peer-to-Peer Networks Using P-Trees", Proc. WebDB, pp.25-30 (2004).
- [6] D. Comer, "The Ubiquitous B-Tree", ACM Computing Surveys, vol 11, no 2, pp.121-138 (1979).
- [7] Overlay Weaver, <http://overlayweaver.sourceforge.net/index-j.html>.