範囲検索 P2P による大規模センサーオーバレイネットワーク構成技術に関する研究

Shao Xun　地引昌弘　寺西裕一　西永　望

センサー資源のプロパティによる検索をサポートしたスキップグラフ（SG）によるセンサーオーバレイネットワークを提案する。インターネットの階層構造を組み込んだ SG の階層指向拡張（HSG）が、レイテンシーとトラフィックのローカリティを改善する。またノードの自律的な相互支援を実現する仮想ノード手法により、フラッシュクラウドの影響を緩和する。シミュレーションにより、この構想の効率性は明らかである。

1　はじめに

今日、我々はインターネットなしにはほとんど何もできない。インターネットは成長を続けており、そのトラフィック量は、2025 年には今日の 1 万倍を超えるだろうと予測されている。だが、インターネットにはいくつか構造的な問題がある。たとえば、冗長性機能や、ネットワークに導入される追加的な機能の受入に伴う互換性の問題などである。これらの問題が解決されなければ、インターネットはいずれ社会的なインフラとしての機能を失ってしまうだろう。こうした理由から、我々は「新世代ネットワーク」（New-Generation Network、NwGN）[1] の研究・開発を進めている。そのねらいは、50～100 年のライフスパンを持つ新しい社会的インフラとして機能するような、今日のインターネットが抱える問題から解放されたネットワークを生み出すことである。NwGN においては、1 兆個以上のオブジェクトを扱うネットワークを実現することを目指す非常に大規模な情報共有ネットワークプロジェクトが非常に重要な要素となる。

小型・高性能・低消費電力の CPU チップ、大容量・低コストのメモリ、高速モバイルネットワークという面における近年の技術発展のおかげで、近い将来、多数の自律的センサーネットワークが展開されるものと期待される。こうしたネットワークにより、社会的インフラの全般的な活用及び利便性を改善することができる。だが、現行のインターネットには、実生活に流通する膨大な数の（いくつかの試算によれば 1 兆個を超える）オブジェクトを扱うだけの余裕はない。我々は、膨大な数のデバイスを扱える大規模情報共有ネットワークのプラットホームに向けた基礎技術を開発することを目指して、自己組織型の範囲検索 P2P 技術である SG（Skip Graph、スキップグラフ）[6] に基づいたセンサーオーバレイネットワーク[2]-[5] 構造を提案した。図 1 にそのアーキテクチャーを示す。図の右側に示したものが、1 つのオーバレイノードのアーキテクチャーである。1 つのオーバレイノードには、少なくとも 3 つのコンポーネントが含まれる。センサー資源、センサーデータストレージ、コンピューティング資源である。その利用者は、機械、エンドユーザー、IoT（モノのインターネット）サービス事業者、あるいは他のセンサーオーバレイノードも考えられる。実際には、センサー資源の保有者はこのアーキテクチャーを、スマートゲートウェイ、クラウドコンピューティング、フォグコンピューティング技術[7][8] と合わせて容易に実装することができる。図の左側は、個別のセンサーオーバレイノードを統合することで開発されるセンサーオーバレイネットワークを示している。ゲートウェイ付近の数字は、センサー資源のプロパティである。図の左下側は、SG に基づいたオーバレイ基層を示している。これは構造的な範囲検索 P2P である。範囲検索は、それによってユーザーが正確なユニーク

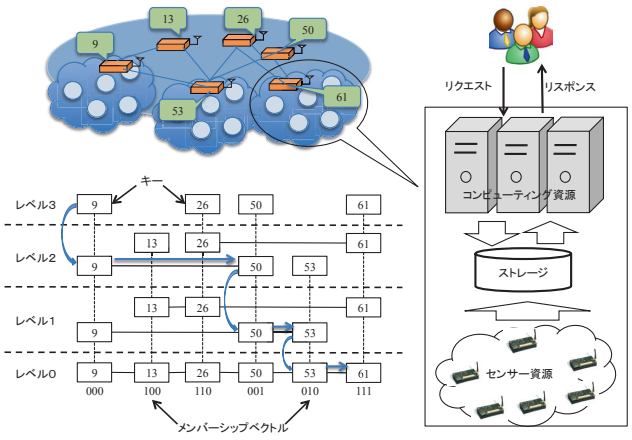


図 1　センサーオーバレイ・アーキテクチャー

（一意的）ID や IP アドレスを知らなくてもセンサー資源を発見できるようになるため、非常に重要な機能である。たとえば、ある地域で地震が発生した場合、ユーザーは「震源から 100 km 以内の地域にあるセンサー」といった検索条件により、必要とするセンサー資源を見つけることができる。SG の機能については、**2** において詳細に紹介する。

SG はセンサー資源の検索においては効率的だが、論理的トポロジーと物理的トポロジーのあいだの不一致に悩まされる。大規模な P2P システムでは、ピア間の 1 回のホップが複数の AS、国、場合によっては複数の大陸にもまたがる可能性があり、そのことがレイテンシーとローカリティを大幅に悪化させ、システムのパフォーマンスに影響を与える。実際に、インターネット内のノードは、それらがアクセスする ISP ネットワーク、それらが所在する国といった自然な階層的プロパティを持っている。こうした階層的プロパティは、精度は粗いものの、ノード間のレイテンシーを推測するために利用できる。たとえば、同一の ISP ネットワークにアクセスしているノード間のレイテンシーは小さいと考えられるのに対し、異なる ISP ネットワークにアクセスしているノード間のレイテンシーは大きくなる。したがって、同一の ISP 又は AS に属するノードにメッセージを転送すれば、レイテンシーは大幅に低下させることができる。さらに、先行研究によれば、ネットワーク接続が失敗するのは、主としてボーダゲートウェイプロトコル（BGP）の障害が原因である。結果として、同一の ISP 又は AS に属するノードは同時に接続を失敗する。したがって、ルーティングのローカリティを改善すれば、障害の切り分けにも有益である。大規模なセンサーオーバレイネットワーク構築に向けて SG を改善するため、我々は SG の階層指向拡張（hierarchy-aware extension）である HSG を提案する。HSG において我々は、インターネットの階層的プロパティを反映する「H エントリ」と称する追加のエントリを含むように SG のルーティングテーブルを拡張する。ルーティングのプロセスにおいて、H エントリはレイテンシーとトラフィックのローカリティを改善するために高い優先順位で用いられる。我々は現実に近いシナリオを用いて詳細なシミュレーションを実施した。その結果から、HSG は SG に比べ、わずかなオーバーヘッドでレイテンシー及びローカリティの双方を改善できることが分かった。

SG の本質的な問題としては、トポロジーの不一致の他に、もう 1 つ、オーバレイノードがフラッシュクラウド（flash crowd）に非常に悩まされるという点がある。フラッシュクラウドの特徴は、相対的に短い時間のあいだに、あるサービスに対するリクエストが劇的に増加することである[9]。フラッシュクラウドがセンサーオーバレイネットワークに及ぼすダメージが極めて深刻なのは、主として 2 つの理由による。第 1 の理由は、1 つのノードがハイエンドのサーバ及びクラスタほど多くのコンピューティング資源・ストレージ資源を持っていないからである。第 2 の理由は、SG のキー順序保存性である。フラッシュクラウドが発生すると、複数の隣接するノードが同時にホットスポットとなる可能性が高い。キーがセンサー資源の場所を示しているとすると、地震が発生した場合に、震源周辺のノードがすべてホットスポットになる。センサーオーバレイネットワークにおけるフラッシュクラウドを軽減するために我々が採った手法は、フラッシュクラウドが生じる場合、ホットスポットであるノード以外の大半のノードは空いているという観察に基づいている。我々のアプローチでは、あるノードがフラッシュクラウドを検知すると、そのノードは空きノードの情報を集めるために指定された TTL にサンプリングメッセージを送信する。十分な空きノードの情報が集まった後、そのノードは、条件を満たす空きノードの各々に対し、自らと同じキーを持つ仮想ノードを作るよう要請する。それから、この仮想ノードがあたかも通常のノードであるかのようにオーバレイネットワークに挿入される。キー順序保存性に従い、すべての仮想ノードはオーバレイトポロジー中でホットスポットとなっているノードの周囲に挿入される。仮想ノードはホットスポットとなっているノードと同じプロセスを走らせ、ユーザーに対して同じサービスを提供する。このアプローチにより、ホットスポットとなっているノードの検索サービス負荷だけでなく、ホットスポット近隣のノードの検索ルーティング負荷も大幅に分散させることができる。

以下、本稿は 2 つの部分に分かれている。**2** では、スキップグラフを簡単に紹介する。**3** 及び **4** は、それぞれ、HSG 及び仮想ノードに基づくフラッシュクラウド軽減手法を紹介する。最後の節で今後の取組への展望を示して締めくくりとする。

2　SG の簡単な紹介

本節では、SG を簡単に紹介する。SG はキー検索のための P2P アプリケーション用の分散データ構造である。SG における各ノードは 2 つのフィールドを持っている。キーとメンバーシップベクトルである。ノード u のメンバーシップベクトルを $m(u)$ で表すことにする、$m(u)$ の要素は有限個のアルファベットの集合 π に属する。アルファベットの大きさの逆数を p とする。すなわち、$p=\frac{1}{|\pi|}$ である。理論上は、$m(u)$ は π

上の無限語となる。しかし現実に必要となるのは、長さ O(logN) の接頭辞のみである。ここで N はノードの総数である。SG には複数の階層があり、ノードは階層が上がるごとに小さくなっていく二重のリストにグループ化される。階層は 0 から始まる。どのリストにおいても、ノードはキーのアルファベット順にソートされる。階層 0 ではすべてのノードが単一のリストに所属する。階層 1 ではノードは $\frac{1}{p}$ 個のリストに分割される、同様に、階層 i の 1 つのリストに含まれるすべてのノードは、階層 $(i+1)$ では、$\frac{1}{p}$ 個のリストに分割される。メンバーシップベクトルは、あるピアが各階層においてどのリストに属するかを決定する。$m(u)$ の長さ l の接頭辞を $m(u)\uparrow l$ で表そう。すると、$m(u)\uparrow l = m(v)\uparrow l$ であれば、2 つのノード u 及び v は階層 i において同じリストに属する。SG の検索アルゴリズムは次のようになる。各ノードでは、クエリーの値を、クエリーが到達した階層におけるそのノードのキーと比較する。そして、クエリーを左に送るか右に送るかによって決定が下される。さらに、次のノードのキーとの比較が行われる。そのキーがクエリーの値よりも大きくなければ、クエリーはそのノードに送られる。次のノードのキーがクエリーの値よりも大きければ、階層を 1 減らして、隣接するノードのキーとの比較プロセスが、条件が満たされるまで続けられる。図 1 は、現実に最も普通のケースである $p = 1/2$ であるような SG の例を示したものである。図の中で、長方形はノードを表し、長方形の中の数字はキーを表す。各ノードの下の二進数列はそのノードのメンバーシップベクトルで、その接頭辞が任意の階層においてノードをリストに分類するために用いられる。ここで、一番左側のキー 9 を持つノードから、キー 61 を検索するとしよう。61 は 9 よりも大きいから、クエリーは右に送られる。我々は検索の行先を、9 の最も上の階層 (階層 3) で探す。1 つもリンクが存在しないため、階層を 1 下げる (つまり階層 2 に下げる)。階層 2 において、右に隣接するキー 50 を持つノードのキーが参照される。50 は 61 より小さいので、クエリーは 50 に送られる。このプロセスが、キー 61 のノードに到達するまで繰り返される。

3 効果的なセンサー資源検索のための階層指向スキップグラフ (HSG)

3.1 ノードの階層構造

1 で述べたように、レイテンシーとローカリティの双方を改善するには、階層構造が効率的である。HSG における重要な問題は、どのようにクラスタを作るか、そして新たなノードをどのクラスタに追加するべきか判断することである。本研究では、インターネットの内在的な階層構造を反映するような自然な形でノードのクラスタ化を試みた。図 2 に示したのがそのシンプルな例である。「NTT 東日本」と「NTT 西日本」(日本にある 2 つの AS) が最低層のクラスタ (「葉」) である。この 2 つは更に大きなクラスタ「日本」を形成する。最上層 (「根」) では、「日本」「アメリカ」をはじめ世界のいくつかの国が「世界」という層を形成している。「NTT 東日本」に属するノードは「日本」クラスタ、「世界」クラスタにも属している。同一のクラスタに所在するノードは、物理空間においても互いに近接している。この階層構造に基づいて、我々はピア間の物理的距離を反映するよう、クラスタ上の距離を定義する。2 つのピア間のクラスタ上の距離は、それらが所在する「葉」から、最も低い共通の「親」までの最大深度として定義される。

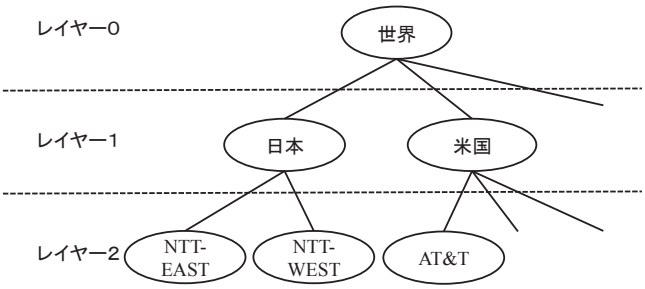


図 2　インターネットの階層構造

3.2 HSG のシステム設計

SG の階層的プロパティを反映するため、我々は SG のルーティングテーブルを拡張し、ルーティングテーブルにおける 2 種類のエントリ、すなわち「B エントリ」と「H エントリ」を含むようにした。B エントリは、SG ルーティングテーブルの基本エントリを、当該ノードから遠隔ノードまでのクラスタ上の距離を追加することによって拡張したものである。HSG ルーティングテーブルの任意の層におけるいずれの側でも、もし B エントリが空でなければ、1 つないし複数の H エントリが存在する可能性がある。H エントリによって参照される遠隔ノードは、B エントリに含まれるものよりもメッセージの転送先として優れた候補である。H エントリにおけるポインタは、クラスタ上の距離とキー値という双方の要件を満たさなければならない。この要件を図 3 で説明する。

キーが k であるようなノード u を想定する。階層 i における B エントリはキー k_i^b を持つ遠隔ノード u_i^b を、階層 $(i\text{-}1)$ における B エントリはキー k_{i-1}^b を持つ遠隔ノード u_{i-1}^b を、そして階層 i における H エントリはキー k_i^h を持つ遠隔ノード u_i^h を各々参照している。階層 i に

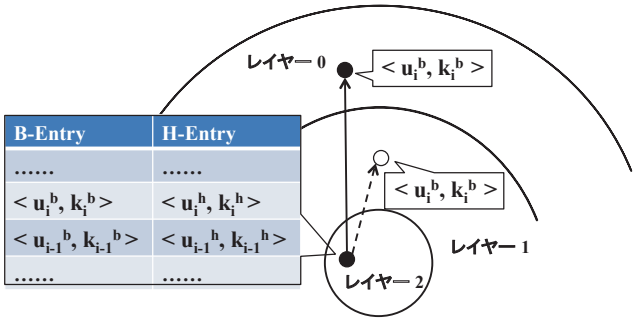


図 3　HSG におけるルーティング原理

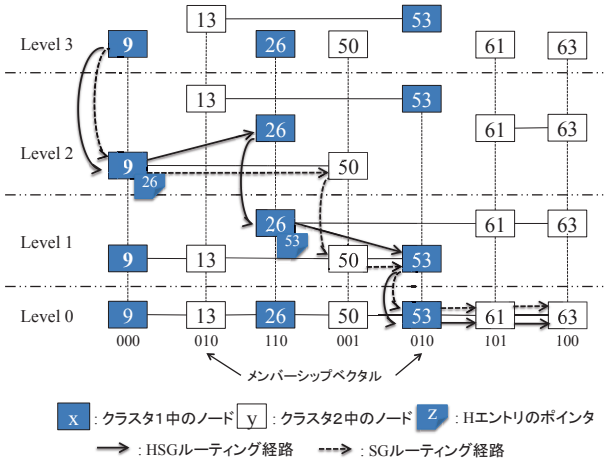


図 4　HSG におけるルーティングの例

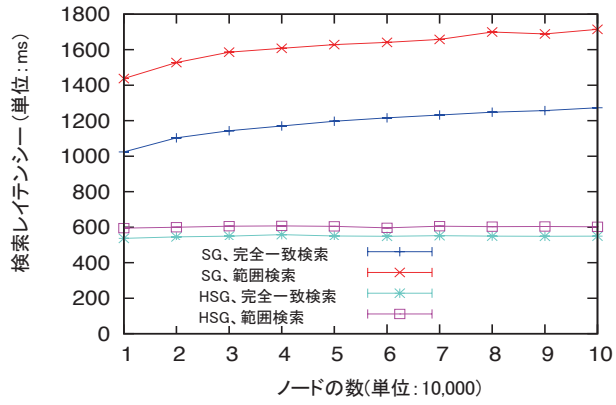


図 5　平均検索レイテンシーの比較

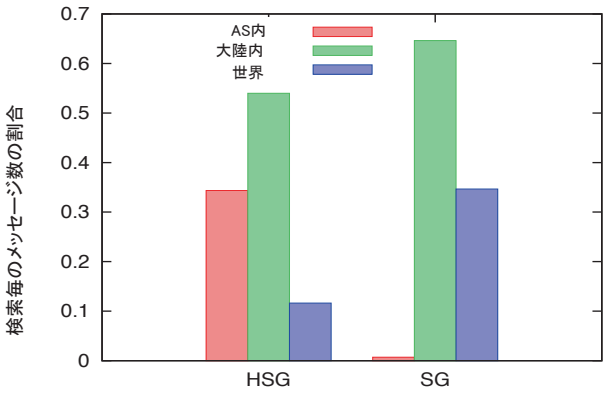


図 6　トラフィックローカリティの比較

おける H エントリは、次の 2 つの要件を満たさなければならない。1 つめは、$k_{i-1}^b < k_i^h < k_i^b$ であり、これによって、平均的な検索ホップ数が O (logN) を維持することが保証される。2 つめは、u から u_i^h までのクラスタ上の距離が u から u_i^b までの距離よりも小さくなければならない。この例では、u から u_i^h までの距離は 2 で、u から u_i^b までの距離は 3 である。$k_{i-1}^b < k_i^h < k_i^b$ であると仮定すると、H エントリのポインタは u_i^h を指示するはずである。この図では N の右側におけるルーティングテーブルのエントリのみを示している。左側は、キー値の要件が反対になったものと同じである。HSG におけるルーティングプロセスは標準的な SG と類似しているが、H エントリが B エントリよりも高い優先順位で用いられるという点が異なっている。

図 4 に検索例を示す。この図において、ノード 9 は階層 0 においてノード 13、階層 1 及び階層 2 においてノード 50 を指示するルーティングエントリを持っている。HSG では、ノード 9 は H エントリにおいてノード 26 を参照している。ノード 9 がキー 61 を検索する場合、次のホップはノード 50 ではなくノード 26 となる。

3.3　評価

インターネットのネットワーク特性を反映するため、我々は PingER[10] プロジェクトから得たデータを用いた。PingER はインターネット上のトラフィックを監視するための測定インフラを持つ複数の共同プロジェクトの 1 つである。監視用ノードは、ping コマンドを用いて遠隔ノードへの送信を開始し、反応時間を測定・記録する。シミュレーションでは、「AS」「大陸」「世界」という 3 つのレイヤーを持つ階層構造を考えた。具体的には、7 つの大陸、500 の AS、10,000〜100,000 のノードというシナリオを考えた。

初回のシミュレーションでは、ノードの数を 10,000 から 100,000 まで変えてみた。範囲検索に関しては、検索範囲は 2,000 に固定した。図 5 は、SG 及び HSG の検索レイテンシーを示したものである。

完全一致検索及び範囲検索の双方において HSG のパフォーマンスは SG のそれを大幅に上回ることが分かる。さらに、SG においてはノードの数が増えるにつれてレイテンシーが増大するのに対して、HSG におけるレイテンシー増大は無視できる。また、HSG においては完全一致検索と範囲検索の差が非常に小さいことにも注意したい。これに比べて、SG では範囲検索のレイテンシーが完全一致検索のレイテンシーよりも大幅に大きい。レイテンシーの他、HSG はトラ

フィックローカリティという点でも SG を上回っている。図 6 に見るように、SG の AS 内トラフィックレシオはほぼゼロである。これに比べて、HSG では大きなトラフィックローカリティを実現している。

4 仮想ノードによるセンサーオーバレイにおけるフラッシュクラウドの軽減[4]

このセクションでは、SG ベースのセンサーオーバレイネットワーク及びその他一部の範囲検索 P2P におけるもう 1 つの本質的な問題であるフラッシュクラウドの軽減について、我々が採った方法を紹介する。

4.1　仮想ノードに基づくメカニズム

図 7 は、仮想ノード・メカニズムの基本的アイデアを示すものである。仮想ノードの生成は 2 つのステップで行われる。第 1 に、ホットスポットとなったノードがいくつかの空きノードに助けを求めるリクエストを送信する。空きノードの容量に十分な余裕があれば、そのノードはホットスポットとなったノードと等しいキーを持つ仮想ノードを起動し、その仮想ノードを、標準的な SG 参加アルゴリズムを持った通常ノードであるかのように、元の SG に参加させる。

図 7 において、実線で描いた長方形は物理ノードを示し、点線で描いた長方形は仮想ノードを示す。ノード内の数字はキーで、カッコの中の文字は物理ノードの識別子である。物理ノード a の本来のキーは 3 である。ホットスポットとなったノード（キーは 50）からリクエストを受信すると、物理ノード a はキー 50 を持つ仮想ノードを起動し、これを通常ノードとして元の SG に参加させる。仮想ノードはすべて SG においてホットスポットノードの付近にあって、仮想ノードゾーンを形成する。

仮想ノードの効果を更に詳しく論じる前に、まず、ホットスポットノードがどうやって空きノードを見つけることができるのかを説明しよう。SG のトポロジーは、リンク間のランダムウォークが迅速に安定分布に収れんする特性を持つ拡張であることが分かっている。SG のトポロジーは正則グラフなので、この安定分布は一様分布となる。これによって、無作為抽出のための非常にシンプルなアルゴリズムが導かれる。ホットスポットノードは O（logN）の TTL を持つサンプルリクエストメッセージを送信する。ここで N はノードの総数である。パス上のすべてのノードは近隣のリンクを無作為に選択し、リクエストメッセージを転送し、TTL を減少させる。TTL が切れたノードは、サンプルを返信する。サンプルにおいては、平均（N-H）/N 個の空きノードがある。ここで H はホットスポットであるノードの数である。H<<N であれば、ホットスポットノードが十分な数の空きノードを見つけることは極めて簡単である。では、仮想ノード導入の効果について、いくつかの分析結果を示す。M が仮想ノードゾーンの大きさだとする。つまり、(M-1) 個の仮想ノードと 1 個のホットスポットノードがあるという意味である。ここで、r_i（ここで $0 \le i \le M$ とする）は、仮想ノードゾーンにおける i 番目のノードを示す。r_i は仮想ノードでもホットスポットノードでもよいことに注意する。ここでキー順序において $s < r_o$ であるような s からの検索を考えると、次の 2 つの結果が得られる。

定理 1：s からの検索が、各仮想ノード及び本来のホットスポットノードにヒットする確率は、O（1 /M）である。

定理 2：u を、キー順序において $s < u < r_0$ であり、u と r_o の距離が d であるようなノードだとする。d ～ M を考えると、s から仮想ノードゾーンまでの検索が u を経由する確率は、O（1 /（M+d））である。証明の詳細は文献 [4] に記載した。

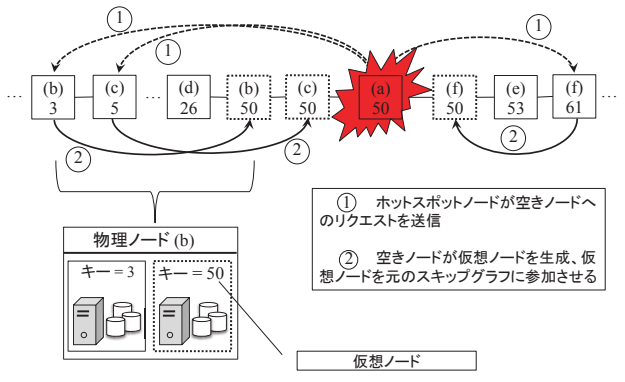


図 7　仮想ノードを用いたフラッシュクラウド軽減戦略

4.2　評価

本節において、我々は提案した方法を評価するための徹底的なシミュレーションを行った。シミュレーションでは、仮想の複製ノード及びホットスポットノードのあいだでの検索サービス負荷の分布を評価した。また、我々のアプローチによって、仮想複製ノードゾーン近くのノードに対する検索ルーティング負荷を低下させることができるか否かも研究した。SG は確率的なオーバレイネットワークなので、我々は各々 10,000 個のノードを持つ 200 の SG を生成し、結果はすべて、200 の独立した SG の平均となっている。

図 8 は、仮想ノード及び本来のノードのあいだでの負荷分布を示すものである。横軸は仮想ノードの数を、縦軸は各ノードの理想的な負荷レシオを示している。

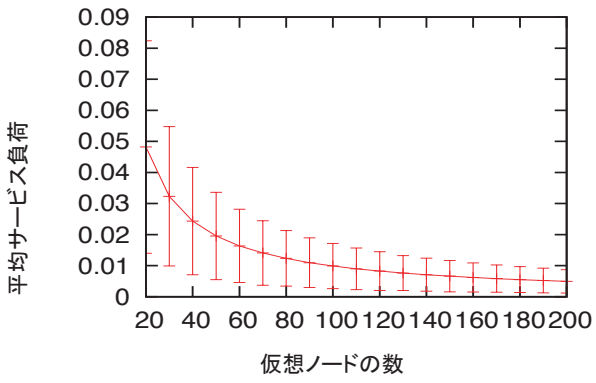


図 8　サービス負荷の分布

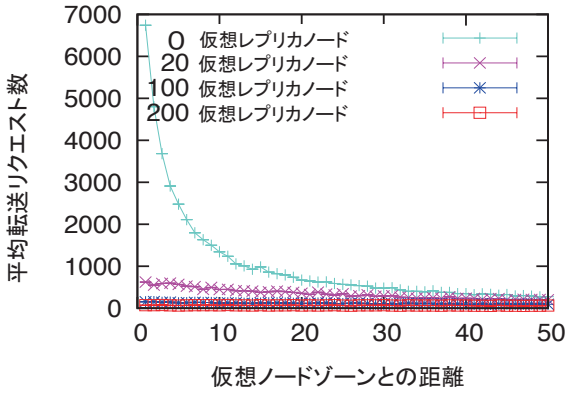


図 9　検索ルーティング負荷の軽減

たとえば、仮想ノードの数が 20 個であれば、理想的な負荷レシオは 1 /20 になるはずである。この図では、実際の負荷レシオと理想的な負荷レシオの標準誤差をエラーバーの形で示している。我々の方法が、サービス負荷の分布という点でうまく機能していることが分かる。

それから、図 9 に見るように、仮想ノードゾーン近くのノードに対する検索ルーティング負荷を調べた。

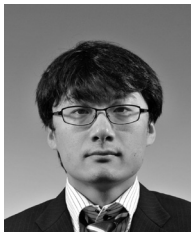
この図で、横軸はノードと仮想ノードゾーンとの距離、縦軸はノードによって転送されたリクエストの数を示している。このシミュレーションでは、無作為に選ばれたノードが仮想複製ノードゾーンに対して 10,000 件のリクエストを送信する。仮想ノードが存在しない場合、ホットスポットノードに最も近いノードは 10,000 件のリクエストのうち約 7,000 件を転送していたが、仮想ノードが 20 個あるだけでも、仮想ノードゾーンの近隣にあるノードが転送するリクエスト件数は劇的に減少している。

5　結論と今後の取組

我々は本研究において、世界中に分布するセンサー資源を統合し、効果的で複雑な検索を可能とするような、ある種の範囲検索 P2P である SG に基づいたセンサーオーバレイネットワークを提示した。世界規模での IoT システムの実現を目指す中で、我々は範囲検索 P2P の本質的かつ重大な問題に取り組んだ。すなわち、論理的トポロジーと物理的トポロジーの不一致と、検索のフラッシュクラウドである。シミュレーションでは、我々の方法によって検索のレイテンシーを 50%近く減少させることが可能であり、フラッシュクラウドの影響も効果的に軽減できることが分かった。今後、フィールド実験により我々の方法を実証していくことを計画している。

【参考文献】

1　NwGN project: http://www.nict.go.jp/en/nrh/nwgn/nwgn-randd-projects.html
2　Y. Teranishi, “PIAX: Toward a Framework for Sensor Overlay Network,” in Proc. CCNC’09, 2009
3　X. Shao, M. Jibiki, Y. Teranishi, and N. Nishinaga, “A Low Cost Hierarchy-Awareness Extension of Skip Graph for World-Wide Range Retrievals,” in Proc. COMPSAC’14, 2014
4　X. Shao, M. Jibiki, Y. Teranishi, and N. Nishinaga, “A Virtual Node-based Flash Crowds Alleviation Method for Sensor Overlay Networks,” to be presented in Proc. COMPSAC’15, 2015
5　R. Banno, S. Takeuchi, M. Takemoto, and T. Kawano, “A Distributed Topic-Based Pub/Sub Method for Exhaust Data Streams towards Scalable Event-Driven Systems,” in Proc. COMPSAC’14, 2014
6　J. Aspnes and G. Shah, “Skip Graphs,” ACM Transactions on Algorithms, Vol.3, No.4, pp.1–20, 2007
7　M. Aazam and E. N. Huh, “Smart Gateway Based Communication for Cloud of Things,” in Proc. ISSNIP’14, 2014
8　F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, “Fog Computing andIts Role in the Internet of Things,” in Proc. ACM MCC’12, 2012
9　J. Jung, B. Krishnamurthy, and M. Rabinovich, “Flash Crowds and Denial of Service Attacks: Characterization and Implications for CDNs and Web Sites,” in Proc. WWW’02, 2002
10　PingER project: http://www-iepm.slac.stanford.edu/pinger/

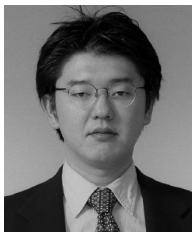


Shao Xun
ネットワーク研究本部ネットワークシステム
総合研究室研究員
博士 (情報科学)
分散コンピューティング、オーバーレイネットワーク、センサーネットワーク

地引昌弘　（じびき　まさひろ）
ネットワーク研究本部ネットワークシステム
総合研究室専門研究員
博士 (システムズ・マネジメント)
新世代ネットワーク、情報指向ネットワーク、超大規模情報流通ネットワーク

寺西裕一　（てらにし　ゆういち）

ネットワーク研究本部ネットワークシステム
総合研究室研究マネージャー
博士（工学）
ユビキタスコンピューティング、オーバレイネットワーク、マルチメディア、データベース
モバイル



西永　望　（にしなが　のぞむ）

ネットワーク研究本部ネットワークシステム
総合研究室室長
博士（工学）
新世代ネットワーク