

無線センサ DB における局所値を考慮した 分散フィルタリング結合演算手法とその評価

中村 達也[†] 中野 美由紀[‡] 喜連川 優[‡]

[†] 東京大学大学院情報理工学系研究科 〒153-8505 東京都目黒区駒場4-6-1

[‡] 東京大学生産技術研究所 〒153-8505 東京都目黒区駒場4-6-1

E-mail: [†] nakamura@tkl.iisu-tokyo.ac.jp, [‡] {miyuki,kitsure}@tkl.iisu-tokyo.ac.jp

あらまし 近年、情報化社会を支える IT 基盤技術として、無線センサネットワークシステム (WSNs) が注目されている。WSNs では、多量のセンサからのデータを効率的に、収集、整理、管理、利用するセンサ DB が必要となる。収集されるデータ量の爆発的増加に対し、センサノードは未だに、電力やバンド幅などの物理的な制限が存在している。高効率なセンサ DB の実現を目指し、結合演算処理における転送量削減や、特定のノードへの過負荷を避けるルーティングなどの研究が行われている。大規模センサ DB 上の問合せ処理効率化手法の一つであるフィルタリング機構を基に、地理的に近い、あるいは距離的に近い環境下における環境センサは似た値を持つことに着目し、よりデータ転送量を削減する分散フィルタリング機構を提案した。フィルタリング機構がセンサ全体のフィルタ条件の制御を行うため、センサ数が増えるに従い、フィルタ条件の管理コストが膨大になる。そこで、本提案手法では、距離の近いセンサのフィルタ条件はセンサグループ内で決定することにより、不要な条件転送などを減らし、センサ数が増大しても管理コストを抑える。実際の環境データを元に人工的にデータセットを作成し、センサの局所値を考慮した本手法のフィルタ更新処理、再送処理が持つ効果を、センサ数に対する特性に着目して評価する。

キーワード 無線センサネットワーク、フィルタリング、分散化、転送量削減

1. はじめに

近年、様々な場所にセンサが設置され、分析や制御などに利用されている。無線通信機能を持つ情報通信端末の小型化、低コスト化が飛躍的に進み、無線センサネットワークの研究が多数行われている。電子タグや IC チップ、無線センサノードを多数用いることで、大規模かつ高解像度のオンライン分散計測システムを構築でき、現代社会における IT 社会基盤として期待されている。すでに、IBM の smart planet プロジェクト[1]や Google の powermeter[2]等、実運用も図られている。これらの実用例に示されるように、無線センサネットワークから得られる大規模なデータを効率的かつ永続的に統合・管理・利用する技術が求められ、無線センサネットワーク上のデータベース技術への注目が高まっている。

しかし、無線センサノードの性能は向上してきているものの、大規模データベース等処理負荷の高いシステムは、既存の技術を単純に応用しても、ネットワークバンド幅や電力などの物理的制限が多く、無線センサネットワークに適用可能なデータベース技術が必要となる。中でも、データベース処理で最も負荷が高く、頻繁に用いられる join 処理の研究が近年多く報告されている。各センサノードからデータベースが置かれたベースステーションに転送され、突き合わせ(マッチング)処理が行われる。しかし、単純に全データを送信すると、転送コスト(送信、受信によるノードのリソースの消費)がかかる。そこで、join 処理の突き合わせ(マッチング)対象となるデータのみを送れば、各センサノードの転送コストを抑えられる。マッチング処理となるデータをどのように選択的に転送するか(あるいは不要なデータを転送しないか)、効率的な join 処理手法の研究が多くされている。[3,4,5,6,7]

本稿では、転送コストを削減する大規模センサ DB 上の結合演算処理効率化手法として、M Stern ら[3]の CJF(Continuous Join Filter)と

いうフィルタリング機構を基に、ノードセンサ数が爆発的に増えた場合に問題となるフィルタ条件の管理コストを抑える手法として、環境センサが持つデータの局所性に着目し、この機構を分散化させた分散フィルタリング機構を提案した。本手法のフィルタ更新処理、再送処理における WSNs 全体の転送量削減効果についての評価を行う。本稿の構成は、まず第 2 章で関連研究、第 3 章で CJF、第 4 章で分散フィルタリング機構、第 5 章で基本評価、最後にまとめと今後の課題を述べている。

2. 関連研究

センサ DB の join 処理における転送量を減らす研究が多くなされている。[4,5,6,7]

まず、H.Yu ら[4]は、Synopsis という関数を用い、各センサノード上のデータからクエリで指定されたデータストリーム情報を抽出する。抽出された情報に用いて join 処理を行い、その結果から tuple を送るべきノードを絞り込む。この手法は、対象となる tuple 属性が多くなると抽出された結果が大きくなり、適切に必要な tuple を抽出することが出来ない可能性がある。

次に M.Stern ら[5]の手法では、static な属性の join 条件を先に処理することで、送信するべきノードを減らす Base join に加え、ベースステーションにおいて、semi-join を行い、さらに対象ノードを減らして tuple の送信をする。semi-join とは join 条件に含まれる属性の値をベースステーションに送り、その属性に対してのみ join を行い、条件に合致するノードだけ、全属性を含む tuple を送るようにする操作である。

また、X.Yang ら[6]の手法では、static な attribute の value を用いて、センサノードの relation を分散化させている。分散化することで、部分的に join 処理を行えるため、転送量が抑えられている。しかし

ながら、static な attribute の value に依存してしまっているため、クエリに多くの active な attribute が関わると精度が落ちてしまう。また、大きな relation には不適合であることも問題になっている。

最後に、M.L.Yiu[7]らの手法では、Yang らの分散化手法で用いられていた static attribute value の代わりに、各センサノード間の距離をパラメータに用いて、relation を分散化することで、各センサノードの通信範囲内において、部分的に join 処理を行っている。

3. Continuous Join Filter (CJF)

M Stern ら[3]は Continuous Join Query に対して、Continuous Join Filter(以下 CJF)というフィルタリングアルゴリズムを提案している。この手法は、温度センサ等連続的にデータを送るノードに対し、データ転送範囲を指定したフィルタを各センサ上に設定することで、データの取捨選択を行い、余分な転送量を削減する。

本手法では、あるノードから転送されるデータの連続値が、他ノードの値と多くの場合においてマッチしないことに着目し、ある値幅のフィルタをノード毎に設定し、センシングした値がフィルタ内であれば、そのデータは処理に必要無いと判断、データ転送を行わないことでデータの転送量を削減する。しかし、ベースステーションが転送された全データの結合を行う際、フィルタリングされたデータを用いて結合処理しなくてはならない。その場合、ベースステーションはノードに再送要求を出し、データを転送させる。この場合、再送要求にデータ転送が加わるため、転送オーバーヘッドが生じる (collision が起きる)。ベースステーションは、全てのデータを処理したのち、その処理で用いたデータの情報からフィルタリング値の調整と更新を行い、効率良く転送量を削減できるようにフィルタを維持している。

3.1 CJF 処理の流れ

- 操作 1. 各センサはフィルタリング条件にあてはまらない場合、データをベースステーションに転送
 - 操作 2. ベースステーションはフィルタリングされたデータが、本当に join 処理に不要かどうか調べ、必要な場合は、センサに転送命令を転送(collision)
 - 操作 3. ベースステーションは、直前に処理したデータ情報と与えられたクエリを基に、最適なフィルタを計算
 - 操作 4. ベースステーションは、転送コストの比較を行い、フィルタを更新するかどうかを決定
- 操作 1.-4.が繰り返し行われる。

3.2 最適なフィルタサイズ計算

フィルタは、転送コストが最小化されるように設定される。ここで、転送コストとは、フィルタを通過した場合の予測転送コストと、操作 2(collision)が起きた場合に必要とする予測転送コストで構成される。予測転送コストを求めるために、フィルタリングされる確率、他センサノードと join する確率などが用いられる。

また、フィルタリングによって双方のノードのデータが転送されない場合、正しい join 結果を得ることができない。そこで、フィルタは転送コストを最小に抑え、且つ、フィルタ同士の衝突をさけるための条件を加え、最適なフィルタサイズを計算している。

3.3 フィルタ更新

新しいフィルタに更新するためにも、転送量が必要となるため、変化の少ない属性を持つフィルタである場合、常に更新し続けると、かえってコストが高くなってしまう。

そのため、更新の決定の際に、charge コストと blocking コストを考慮し、更新の可否を行う。

charge コストは、現フィルタのときと新フィルタのときの予測転送コスト差を表している。また、最後にフィルタを更新した時間と現在の時間の時間差を加味し、古いフィルタを長時間利用する懸念を避けている。

blocking コストは、フィルタ更新をしなかった場合に、他センサの予測転送コストがどれだけ影響を受けるかを数値化したものである。他センサへの影響を考慮することにより、あるセンサの新フィルタと他センサの現フィルタが衝突し、正しい join 結果を落とす可能性を取り除いている。

CJF では、フィルタの中心を前データ値を用いているため、次にセンシングされた属性値が大きく変化し続けるときには、フィルタリング効果が失われるだけでなく、フィルタを頻繁に更新しなければいけないという問題がある。

4. 分散フィルタリング機構

本章では、提案手法である分散フィルタリング機構 (Distributed CJF) の説明を行う。

CJF では、フィルタサイズの計算・更新をベースステーションが処理している。そのため、フィルタを更新しなければいけないセンサに対し、常にベースステーションから、新しいフィルタ情報をセンサに転送することになる。現状のセンサノード数では処理可能であるが、センサノード数が増えると更新再計算・更新コストは甚大なものとなる。そこで、本稿ではベースステーションで行っているフィルタ更新処理を分散化する手法について検討する。

提案する分散フィルタリング機構では、図1のように、位置情報、routing path 等を考慮して、センサノードをいくつかのグループに分け、グループ毎に中間ノードを設定する。各グループ内のセンサは、中間ノードにデータを転送することで、中間ノードは、グループ内のセンサデータを保持することができる。これにより、ベースステーションで処理されていたフィルタサイズの計算・更新を、各中間ノードがグループ内のセンサデータに限定することで、同様の処理が可能になる。もし、中間ノードにおいて、グループ内のセンサデータを用いて、計算したフィルタサイズが、ベースステーションにおいて、全センサデータを用いて、計算されたフィルタサイズと同じであれば、ベースステーションから更新するセンサに、新しいフィルタ情報を転送する代わりに、同様のフィルタ情報を中間ノードからセンサに転送できる。つまり、ベースステーションは中間ノードに対して、中間ノードが計算したフィルタサイズをセンサに転送しろという命令を転送するだけで済むため、更新における転送量を削減することができると考えられる。

環境センサが持つ観測データの局所性を用いれば、グループ内で各センサのフィルタ条件を管理でき、効率的な分散化を行える。ま

た、本手法の性質から、センサ数が増える程、グループ内で管理できるセンサが増え、中間ノードからベースステーションまでの負荷を抑えられるので、分散化によって効率的な転送量削減効果が得られると考えられる。

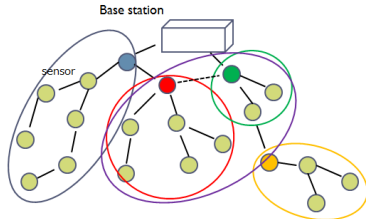


図1: 分散化におけるグルーピング例

4.1. フィルタ更新

フィルタ更新時における新しいフィルタ情報の転送を各グループの中間ノードがベースステーションの代わりに行うことで、転送量を削減できる。

加えて、グループ内だけでなく、フィルタ更新時におけるグループ間の更新頻度相関を用いることで、更なる転送量の削減が可能になる。CJF アルゴリズムでは、あるセンサのフィルタ更新が、他のセンサもフィルタ更新に影響する場合がある。このような状況は、センサ同士、もしくはグループ同士でデータの相関が多く見受けられる。そのため、フィルタ更新において、図1の点線で示されたように、相関の強いグループの中間ノード同士でデータ送受信を可能にし、一方の中間ノードのみでフィルタサイズ計算を行うことで、相関の強いグループを、局所的に一つのグループとしてみなせ、一つのグループによるフィルタ値設定と同様に転送量を削減できると考えられる。

4.2. collision

また、collision が起きた際の転送量も削減も期待できる。collision とは、クエリに合致するデータを持つセンサが、フィルタリングによって、データ転送されなかった場合に、ベースステーションがセンサに対して、再送命令を転送し、データを転送させなければいけない操作である。従来の方法では、ベースステーション内にある全センサのフィルタ情報を用いて、ベースステーションが再送の可否を判断する。これに対し、分散フィルタリング機構では、グループ内のセンサのフィルタ情報は中間ノードに保持されているため、フィルタリングされたセンサの再送の可否を、部分的に中間ノードで判断することができる。中間ノードで再送命令を転送できれば、ベースステーションから再送命令を転送する必要がなくなり、転送量を削減することができる。

5. 基本評価

分散フィルタリング機構を用いたことによるデータ転送量の削減効果について検討する。今回の実験では、Micro Stem らのCJF[3]のフィルタリングアルゴリズムを実装し、シミュレーションをした場合、どれほど転送量を削減出来るかのフィルタリング効果、及び分散フィルタリング機構を適用した場合のフィルタ更新処理、再送処理の転送量削減効果についての性能評価を行う。

5.1 シミュレーション環境

5.1.1 データセット

[13]では、LUCE データセット[14]という、環境データセットを用いて、本手法の性能評価を行っていた。しかし、LUCE データセットはセンサ数が 88 個と少なく、数千個単位などのセンサ数に対する本手法の評価をすることができない。そこで、実際の環境データを基にして、大規模な無線センサネットワークを仮定したデータセットを人工的に作成した。データセットは、グループ間の平均観測値の差が小さいデータセット1、平均観測値の差が大きいデータセット2の二つを比較することで、グループの局所性の差に対する本手法の性能比較も行う。

データセットの基として気象庁[9]で公開されている観測データを利用した。この観測データは、全国の気象台、及びアメダスを利用して、10分毎に気温、湿度、気圧風向・風速などのデータを取得している。同様に、位置情報も保持している。これらのデータと補間曲面、及び補間曲線を用いて、シミュレーションに必要なデータセットを作成した。

まず、補間する基となる観測データとして、データセット1では関東圏内、データセット2では日本全国内の25点において、2011/7/1~2011/12/31の半年間の観測データを取得した。関東圏と全国でデータセットを作成することで、二つのデータセットが持つ局所性に差をつけた。25点の物理的配置は、それぞれ図2、3に表す。この25点を仮想的に5×5のメッシュ状に配置し、補間関数としてB-splineを用いて、データの補間を行う。補間曲面を用いることで、5×5点の枠内の任意の位置における観測データの推定が可能になる。そこで、5×5点の内、外枠の16点を除く、9点の周囲に、複数のセンサを配置した場合を仮定し、全センサの観測データを推定する。本シミュレーションでは、各9点の周囲に、10,100,1000個のセンサを配置した場合を仮定した。

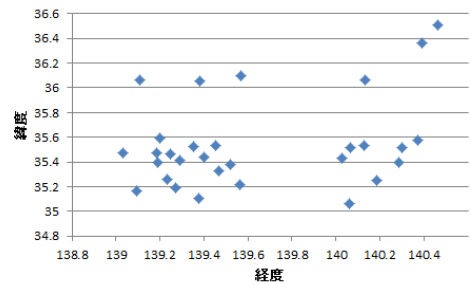


図2: 25点の物理的配置 (データセット1)

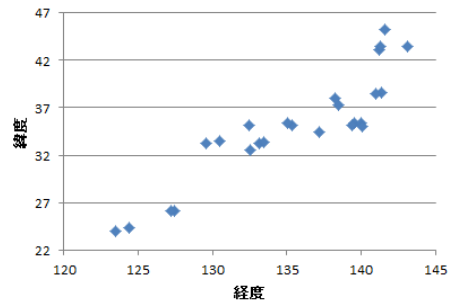


図3: 25点の物理的配置 (データセット2)

次に、各センサデータの拡張を行う。上記の補間によって、得られた推定データは10分毎の観測データである。10分毎では smarter planet プロジェクトなどで用いられる IT 社会基盤となり得るセンサデータとしては、詳細な変化が分からず、不十分であるため、B-spline 補間曲線を用いて、5秒毎の観測データに拡張する。拡張した観測データをデータセットとして用いる。

5.1.2 ルーティングとグループ

今回のデータセットでは、ベースステーションを全センサの位置情報の平均値に設定した。そして、各グループ内で最もベースステーションと位置に近いセンサを中間ノードに設定し、1hop で直接ベースステーションと通信可能とする。中間ノード以外の各センサは、属するグループの中間ノードと 1hop で通信可能とする。

本シミュレーションでは、9 点の観測点の周囲にセンサを配置した場合を想定しているため、グループ数は9で固定している。位置情報などを用いて、グループ数を変化させることで、より局所値を考慮した効率的なフィルタリングが可能になる。

5.1.3 クエリと比較対象

全てのシミュレーションにおいて、以下のクエリを用いる。

```
SELECT A.*, B.*
FROM Sensors A, Sensors B
AND |A. temperature - B. temperature| < X °C
AND A.id != B.id
SAMPLE PERIOD 5s
```

このクエリはセンサが自分以外のノードにおいて、温度差が X°C 未満を満たすノードが存在すれば、その tuple 同士を結合する操作を5秒毎に行うためのクエリである。温度差 X を変化させ、各クエリに対する性能評価も行う。

今回のシミュレーションでは比較対象として、削減率と転送量を用いている。削減率は、join 処理に不要なデータ転送量を 100% としたときの削減転送量の割合を表している。同様に、比較転送量も処理に不要なデータ転送量を表している。ここでいう、転送量は 1hop を1転送量としてカウントしている。

5.2 シミュレーション結果

フィルタリングアルゴリズム、及び分散フィルタリング機構の性能評価を行う。本シミュレーションでは、クエリに与える温度差として、0.01,0.05,0.1 の三種類を用いた。

5.2.1 CJF と提案手法の性能評価

作成した各データセットに対し、CJF と提案手法のシミュレーション評価を行う。各9点に配置するセンサ数を1000個、グループ数を9に設定している。不要なデータ転送量の比較として、三つの手法を用いている。

Non-Filter: フィルタを実装しない従来手法の場合の不要なデータ転送量

CJF: CJF を適用した場合の不要なデータ転送量

DCJF: 提案手法を適用した場合の不要なデータ転送量

5.2.1.1 データセット1

データセット1に対する性能評価結果を図4、5に示す。

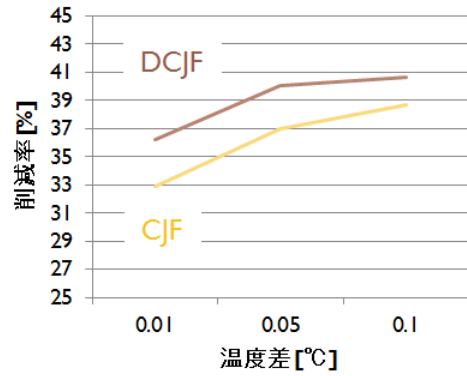


図4: 温度差に対する削減率評価 (データセット1)

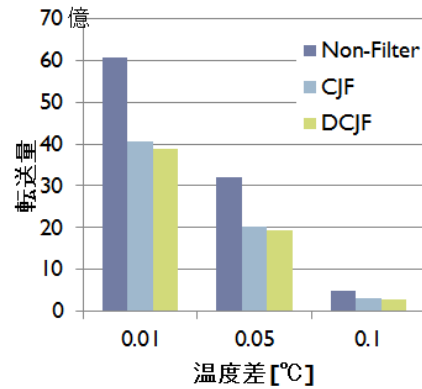


図5: 温度差に対する転送量評価 (データセット1)

性能評価結果から、提案手法を用いることで、従来手法である CJF よりも、最大 3.5% の転送量削減効果を得ることができる。また、センサ数が多いため、どのクエリでも join selectivity は高かったが、その中でも、0.01°C の結果よりも 0.1°C の方が join selectivity が高いので、CJF は 38.7%、提案手法は 40.7% と、0.01°C の場合よりも共に高い転送量削減効果を得られている。しかし、join selectivity が高ければ高いほど、常に転送され続けるようなセンサが増えるため、更新処理、再送処理が起りにくくなり、全転送量における両処理の割合が少なくなる。よって、提案手法により得られる削減できる転送量は、全転送量の比べ、少なくなってしまうと考えられる。

5.2.1.2 データセット2

同様に、データセット2に対する性能評価結果を図6、7に示す。

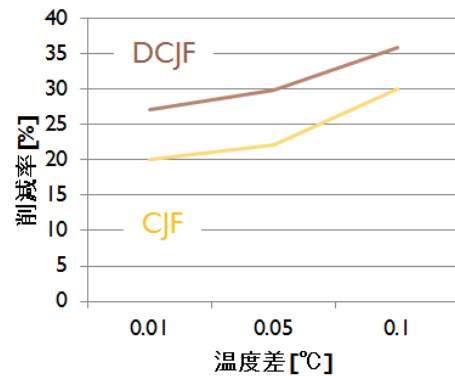


図6: 温度差に対する削減率評価 (データセット2)

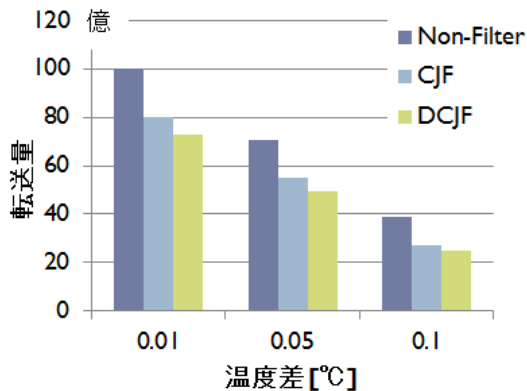


図7：温度差に対する転送量評価（データセット2）

シミュレーション結果より、提案手法を用いることで、CJFに比べ、クエリが0.01°Cの場合は約7%、0.05°Cの場合は約8%、0.1°Cの場合は約6%の転送量削減効果を得られていることが分かる。また、グループ間の差が少ないデータセット1の性能評価と比較すると、削減効果が約5%も増えている。これは、グループ間の差が大きいと、各グループがより局所化されたものになるため、提案手法の分散効果を得られているからだと考えられる。

5.2.2 分散フィルタリング機構のセンサ数に対する特性検証

データセット2に対し、センサ数を各9点10、100、1000個と変化させ、削減率の変化を検証することで、センサ数に対する特性評価を行う。クエリに用いる温度差を0.01°C、グループ数を9に設定している。センサ数を変化させ、シミュレーションを行い、特性を検証した結果を図8に示す。図の横軸はセンサ数の合計、縦軸は各手法の削減率を表す。

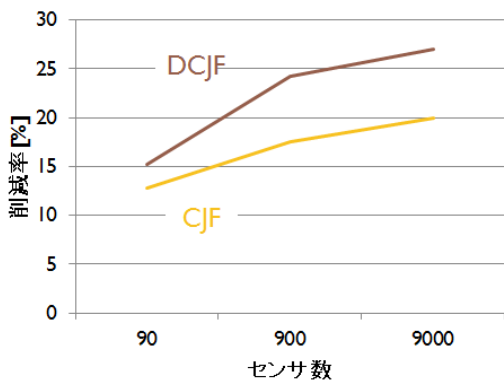


図8：センサ数に対する特性

シミュレーション結果から、センサ数が増えるにつれ、CJFと提案手法の削減効果の差が増えていることが分かる。これは、センサ数が増える程、グループ化による削減効果が得られるためであると考えられる。このことから、今後、センサ数が増え、大規模なWSNsが利用される際にも、フィルタリング機構が適用できると考えられる。

5.2.3 分散フィルタリング機構における各処理の評価

分散化フィルタリングを適用することにより、フィルタの更新処理、データの再送処理において、従来のフィルタリング機構であるCJFよりも転送量削減効果が得られる。各処理において、センサ数

に対する削減効果を検証する。各処理において、クエリに用いる温度差は0.01°C、グループ数を9に設定している。シミュレーションには、データセット2を用いている。

5.2.3.1 更新処理

提案手法を用いた場合、CJFと比較し、フィルタ更新処理に必要とする転送量の差を図9に示す。図の削減率は、CJFで更新処理に必要とする転送量を100%としたときの提案手法による削減できる転送量の割合を表す。

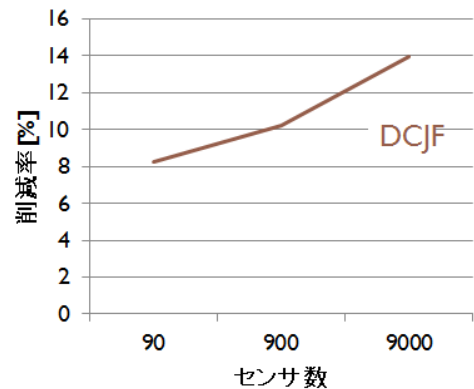


図9：センサ数に対する特性（更新処理）

更新処理も、センサ数が90のときに約8%、センサ数が9000のときに約14%と、センサ数が多い程、削減率が高くなっている。これは、性能評価と同様に、グループの局所性を利用した分散化処理を行うことができているため、図のような特性が得られたと考えられる。

5.2.3.2 再送処理

提案手法を用いた場合、CJFと比較し、データ再送処理に必要とする転送量の差を図10に示す。図の削減率は、CJFで再送処理に必要とする転送量を100%としたときの提案手法による削減できる転送量の割合を表す。

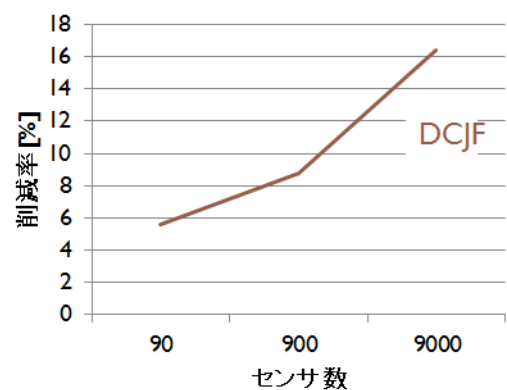


図10：センサ数に対する特性（再送処理）

データ再送処理も、センサ数が90のときに約6%、センサ数が9000のときに約16.5%と、センサ数が多い程、削減率が高くなっている。更新処理と同様に、グループの局所性を利用した分散化処理効果を得られている。一方、更新処理よりもセンサ数の増加に対する転送量削減効果は得られているが、これはデータセットやクエリに依存する部分であると考えられる。

6. まとめ

本研究では、Micro Stem らによって CJF を基に、中間ノードを用いた分散フィルタリング機構を提案した。本手法は、ベースステーションが担う処理の一部を中間ノードでも処理を可能にすることで、更新処理、再送処理における処理の効率化を行った。そして、本手法を気象庁の観測データと補間によって作成した二つのデータセットに適用し、グループ間の平均観測値に差が大きい程、効果的に局所性を利用でき、センサ数が増える程、更新処理、再送処理における転送量削減効果が得られることを検証した。

今後、位置情報などを用いて、より局所値を考慮したグルーピングを行うことで、グループ数に対する転送量削減効果を検証する。また、フィルタリングの効果は、センスされるデータ特性やクエリに依存する可能性があるため、様々なデータセット、クエリに対して、分散フィルタリング機構の挙動を検証し、提案手法の効果をさらに検証することで、その特性を明らかにし、フィルタリング機構にルーティングパス選択アルゴリズムを組み合わせることで、より効率よくデータ転送量を削減できる手法を検討したい。

参考文献

- [1] <http://www.ibm.com/smarterplanet/us/en/>
- [2] <http://www.google.com/powermeter/about>
- [3] Micro Stem, Klemens Bohm, Erik Buchmann, Processing Continuous Join Queries in Sensor Networks: a Filtering Approach, SIGMOD 2010
- [4] Hai Yu, Ee-Peng Lim, Jun Zhang, On In-network Synopsis join Processing for Sensor Networks, MDM2006
- [5] Mirco Stem, Erik Buchmann, Klemens Bohm, Towards Efficient Processing of General-Purpose Joins in Sensor Networks, ICDE2009
- [6] X. Yang, H. B. Lim, M. T. Ozsu, and K. L. Tan. In-Network Execution of Monitoring Queries in Sensor Networks. SIGMOD, 2007.
- [7] M. L. Yiu, N. Mamoulis, and S. Bakiras. Retrieval of Spatial Join Pattern Instances from Sensor Networks. SSDBM 2007.
- [8] X. Zhu, H. Gupta, and B. Tang. Join of Multiple Data Streams in Sensor Networks. IEEE TKDE, 99(1), Jan. 2009.
- [9] <http://www.jma.go.jp/jma/index.html>
- [10] Sam Madden, Joe Hellerstein, and Wei Hong, TinyDB: In-Network Query Processing in TinyOS, <http://telegraph.cs.berkeley.edu/tinydb/tinydb.pdf>
- [11] Samuel R. Madden, Michael J. Franklin, Joseph M. Hellerstein and Wei Hong, TinyDB: An Acquisitional Query Processing System for Sensor Networks, ACM Transactions on Database Systems, Vol.30, No.1, March 2005, Pages 122-173
- [12] N. Jain, D. Kit, P. Mahajan, P. Yalagandula, M. Dahlin, Y. Zhang, STAR: Self-Tuning Aggregation for Scalable Monitoring, VLDB2007
- [13] Tatsuya Nakamura, Miyuki Nakano, Masaru Kitsuregawa, An Preliminary Investigation of Distributed Filtering System in Wireless Sensor Networks System, IEICE Technical Report Vol.111
- [14] <http://sensorscope.epfl.ch/index.php/EnvironmentData>