

分散センサデータの閲覧のためのインクリメンタルなデータ提供方式

白石 陽[†] 安西 祐一郎[†]

地理的に分散したセンサから得られる多種多様なセンサデータは、生活環境の状態を把握するうえで非常に有用である。しかしながら、複数のセンサネットワークから時系列のセンサデータを収集し、統合処理を行う場合には、その結果をユーザに提示するまでの応答時間の増大が問題となる。本論文では、広域に分散したセンサデータを閲覧するためのインクリメンタルなデータ提供方式を提案する。本方式は、仲介エージェントとセンサネットワークのラッパーとなるサーバエージェントから構成され、これらのエージェントの連携により、クエリの実行と検索結果の転送を制御する。検索結果は、ユーザの指定する空間分割および時区間分割のためのパラメータに基づき分割され、インクリメンタルに提供される。クライアントは、受信した検索結果に基づいて逐次的に統合処理を行い、センサデータ統合の途中結果をユーザに提示することができる。

An Incremental Query Answering Mechanism for Browsing Distributed Sensor Data

YOH SHIRAISHI[†] and YUICHIRO ANZAI[†]

Various kinds of sensor data such as weather and traffic information supplied by heterogeneous sensor networks are useful for location-oriented application. Each sensor network is a data resource that can supply on-line and temporal sensor data. We propose a retrieval mechanism with incremental data providing for browsing widely distributed sensor data. Our retrieval system consists of a mediator agent and multiple server agents. Server agents manage time-series sensor data from sensor networks. This system decomposes a query from a client into sub-queries based on spatial and temporal constraints. Each server agent transmits incrementally the retrieval result for each decomposed region at each decomposed temporal interval. The mediator agent controls the query execution and the transmission of these results from server agents. The browsing client can incrementally integrate the received sensor data based on location and temporal information.

1. ま え が き

近年、気象情報や渋滞情報などのセンサデータをオンラインで公開するシステムやサービス^{1)~3)}が多数出現しており、今後、多種多様なセンサデータをネットワークを介して利用できる環境が整備されていくと考える。本論文では、これらのセンサデータを提供するデータリソースを“センサネットワーク”と呼び、広域ネットワークに接続された異種のセンサデータを提供する地理的に分散したセンサネットワークの集合を“異種センサネットワーク環境”と呼ぶ。

異種センサネットワークから提供される多種多様なセンサデータは、ユーザの生活している環境の状態を表しているため、モバイル情報システムや地理情報シ

ステムなどの位置情報システムにとって有用な情報となる。たとえば、天気予報や渋滞情報、観光地の混雑状況などを複合的に利用したナビゲーションシステム、気象情報や大気の状態に基づく生活環境のモニタリングを行うシステムなどの応用が考えられる。このようなセンサデータを利用するシステムにとっては、最新のリアルタイムのデータやある時刻のスナップショットだけでなく、指定した時区間の時系列データも有用であり、センサデータが時間属性を持つという点を考慮することが重要である。さらに、位置情報システムでの利用という観点からは、センサデータが地理的な位置に関連付けられるという空間データとしての側面も重要である。地理的な位置情報に基づいて、収集したセンサデータを地図データや他のセンサデータと統合することができれば、より効果的な情報提供が実現できると考える。

しかしながら、広域ネットワークを介して広範囲に

[†] 慶應義塾大学理工学部情報工学科

Department of Information and Computer Science, Faculty of Science and Technology, Keio University

設置された複数のセンサネットワークから時系列データを収集する際には、データを取得するまでの検索コストや統合結果を提示するまでの応答時間が増大し、センサデータを閲覧するうえでの問題になると考える。

この問題に対する1つの方策として、処理の中間結果を利用して、逐次的に情報提供を行うアプローチがある。たとえば、部分的な検索結果を生成できる問合せ処理システムや途中結果を利用したオンライン集約処理に関する研究などが行われている^{4),5)}。そのほか、人工知能やリアルタイムシステムの分野においても、同様のアプローチが見られる^{6)~10)}。筆者らは、空間分割に基づくインクリメンタルな空間データ提供手法を提案し、実装している¹¹⁾。位置情報に基づくセンサデータ統合を行う場合には、この空間分割に基づく逐次的なデータ提供が有用であると考えられる。しかし一方で、時間情報を考慮していないため、時系列データの集約などの時系列データ統合やセンサデータの時間変化の閲覧に対応する枠組みとしては不十分である。広範囲にわたる時系列のセンサデータを効果的に閲覧するためには、位置情報や領域に加えて、時間属性を考慮した逐次的なデータ提供が効果的であると考えられる。

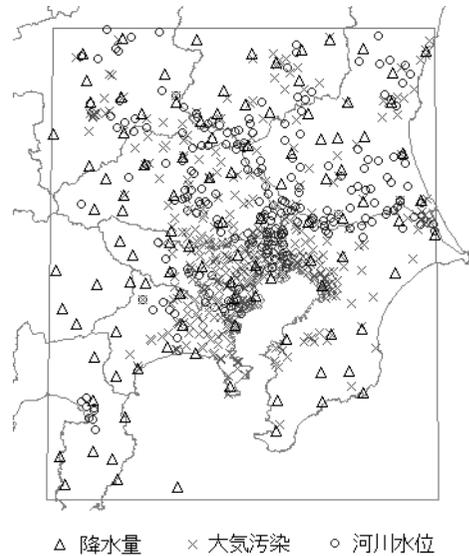
そこで本論文では、文献11)で提案した手法を、異種センサネットワーク環境におけるセンサデータ検索に適用できるように拡張し、分散センサデータを閲覧するためのインクリメンタルなデータ提供方式を提案する。本方式は、仲介エージェントとセンサネットワークのラッパーとなるサーバエージェントから構成され、クエリに指定した問合せ領域と時区間を分割し、その分割に基づく順序で検索結果をインクリメンタルに提供する。これにより、位置情報に基づくセンサデータ統合の結果を、空間分割に従って逐次的に提示するだけでなく、時間順序を考慮して提示することが可能となる。

以下、2章でセンサデータ検索に対する我々のアプローチを述べ、3章でインクリメンタルなセンサデータ提供方式について説明する。4章で、実験結果に基づき提案手法の有用性と応用可能性を検討する。最後に5章で結論を述べる。

2. アプローチ

2.1 異種センサネットワーク環境

現在、気象情報や交通情報を提供するためのWeb情報サービスが多数存在し、様々な種類のセンサデータをオンラインで利用するための環境が整備されつつある。たとえば「大気汚染物質広域監視システム」³⁾では、NO、NO₂、SO₂、OXなどの大気汚染物質に



△ 降水量 × 大気汚染 ○ 河川水位
図1 異種センサネットワーク環境

Fig. 1 Heterogeneous sensor network environment.

関する測定結果が公開されており、「川の防災情報」¹⁾では、河川に関する水位、水質、雨量などの情報を閲覧できるようになっている。図1に、これらのセンサノードの分布状況を示す。

図1の矩形領域には、411個の大気汚染観測ノード³⁾、305個の河川水位観測ノード¹⁾、105個のアメダスの降水量観測ノードが含まれており、密度や分布の異なるセンサネットワークが混在していることが分かる。なお、大気汚染観測ノードと河川水位観測ノードに関しては、各データ提供サイトで測定値とは別に観測点情報が公開されており、その中に、観測点の位置情報として住所が記載されている。本論文では、これらのノードの分布を表示するために、CSVアドレスマッチングサービス¹²⁾を利用して、住所情報から各観測点に対する経度緯度座標を求めている。

本論文では、図1のように、異種のセンサデータを提供するセンサネットワークが混在する環境（“異種センサネットワーク環境”）を想定する。異種センサネットワーク環境では、各センサネットワークの設置されている領域が、地理的に離れて存在する場合もあれば、重なり合うこともある。また、あるセンサネットワークの構成ノード数が多い場合には、データの管理や検索という点で、地域ごとに別々のセンサネットワークとして構成することも考えうる。このような環境では、ユーザは、ネットワークを介して多種のセンサデータを利用でき、ユーザの指定した領域に対して複数のセンサネットワークからセンサデータが提供さ

れることになる。

2.2 インクリメンタルなセンサデータの提供

ネットワークを介して空間データを検索する際の応答時間の問題に対応するために、筆者らは、“anytime algorithm”^{6),7)}に基づくインクリメンタルな空間データ提供手法を提案している¹¹⁾。anytime algorithmは、時間とともに処理品質を単調に増加させ、処理の中間結果を出力できる性質を持つ。文献11)では、空間データ検索の処理品質をデータの収集率として定義し、空間データを管理するサーバがクライアントに対して逐次的に検索の中間結果であるデータを提供することで処理品質を向上させている。具体的には、問合せ領域を分割し、各分割領域ごとにデータ探索を行い、その結果を逐次的にクライアントに提供する。領域分割のパラメータを指定することにより、ユーザが収集したい領域から順に空間データを取得し、領域ごとに受信データの表示や統合などの処理をすることができる。

異種センサネットワーク環境においても、広範囲にわたる時系列センサデータを収集し、その統合結果を表示する場合に、応答時間の増大が予想されるため、空間検索の場合と同様に逐次的なデータ提供が有効であると考えられる。位置情報システムでのセンサデータの利用を想定すると、領域や位置情報を考慮することによって効果的なデータ提供が期待できる。たとえば、観光中の移動ユーザにとって、近隣の交通機関の運行状況、道路の渋滞情報、観光地の混雑状況に関するセンサデータは、自分の位置、目的地あるいは移動経路に近い領域から優先的に取得したい情報である。また、統合結果がユーザの興味のある範囲から順番に表示されれば、ユーザは、その表示情報に基づいて、表示範囲の拡大や新たな領域に対するデータ要求を行うことが可能である。途中結果が“領域”という意味のあるまとまりで提供されることにより、インタラクティブにセンサデータを閲覧することができると考える。

そこで、本論文では、異種センサネットワークから時系列センサデータを収集する際の応答時間の増大に対処するために、逐次データ提供という点に着目し、文献11)で提案した空間分割に基づく手法を、次の方針に従って、センサデータ検索に適用する。

- 異種センサネットワーク環境では、単一のサーバからデータを取得するのではなく、異種のセンサデータを提供する複数のサーバからデータを収集できる検索システムが必要である。さらに、各サーバの管理するセンサネットワークの設置領域やノード分布が異なるため、その異種性を吸収

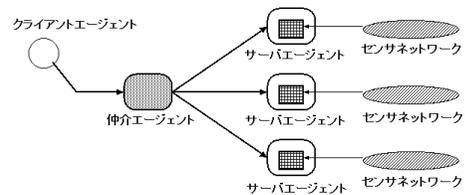


図2 システム構成

Fig.2 System overview.

する必要がある。本論文では、仲介エージェント (mediator) とラッパー (wrapper) から構成される連邦データベースシステムとして検索機構を設計し、シームレスなアクセスを実現するために位置指向検索をサポートさせる。そして、空間分割の概念を分散データ検索に拡張する。

- 各サーバは、空間データではなく、位置属性を持つ時系列データを管理する。センサデータのような時系列データに対して統合処理を行い、その結果を表示する場合には、時間順序が重要になると考えられる。そこで、空間情報だけでなく、時間情報も考慮して検索結果を分割する。具体的には、空間分割に加えて、時区間分割を導入し、これらの分割を指定するパラメータを導入することで、センサデータ統合の結果の提示順序を制御する。

3. センサデータ検索システム

3.1 システム構成

本論文では、文献13)で提案したセンサデータ検索のための連邦データベースシステムを拡張し、インクリメンタルなセンサデータ提供システムを実現する。本検索機構の構成を図2に示す。

3.1.1 サーバエージェント

センサネットワークはデータの種類ごとに構成されるものとし、サーバエージェントは、各センサネットワークのラッパーとして機能する。サーバエージェントは、対応するセンサネットワークの各センサノードの位置情報を保持するためのテーブル (location table) と、各ノードから提供されるセンサデータを時系列として管理するための時系列データテーブルを持つ。時系列データテーブルは、表1に示す属性を持つ。

3.1.2 仲介エージェント

仲介エージェントは、クライアントからのクエリを適切なサーバに転送し、各サーバからの検索結果をクライアントに転送する役割を持つ。仲介エージェントは、各サーバエージェントの提供するセンサネットワークに関する情報を管理し、この情報に基づいて、

表 1 サーバエージェントの持つ時系列データテーブル

Table 1 Data table for time series from a sensor network.

フィールド名	説明
NODE_ID	センサノードの識別子
DATA_TYPE	ノードの提供するデータの種類
VALUE	センサの出力値
UPDATE_TIME	センサデータの更新時刻

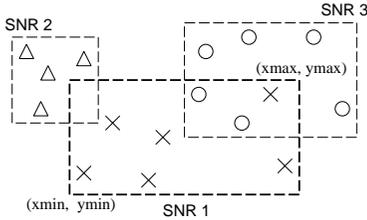


図 3 センサネットワークの位置情報
Fig. 3 Spatial information of sensor networks.

クエリに指定された条件を満たすデータを提供しうるサーバエージェントを探し出す。

センサネットワークに関する情報として、次の 2 つの属性情報が、仲介エージェントによって保持される。:data-type 属性は、サーバエージェントが提供するデータの種類の示す。:area 属性は、サーバエージェントが管理するセンサネットワークの設置場所に関する情報(以下、センサネットワークの位置情報と呼ぶ)を表す。

本論文では、センサネットワークの位置情報(SNR: Sensor Network Region)を、各センサネットワークを構成するすべてのセンサノードを含む領域として定義する。本検索システムにおいては、センサノードの位置を 2 次元座標 (x, y) で表現し、SNR を、式 (1) のように、点 (x_{min}, y_{min}) と点 (x_{max}, y_{max}) によって定まる矩形領域として表現する(図 3)。

$$SNR = \langle x_{min}, y_{min}, x_{max}, y_{max} \rangle \quad (1)$$

3.2 クエリの表現と分割

クライアントは、センサデータを収集するために、式 (2) で表現されるクエリを要求する。

$$Q = \langle A, DT, TI, SC, TC \rangle \quad (2)$$

DT は、クライアントが要求するセンサデータの種別から成る集合である。TI = $[t_{min}, t_{max}]$ は、下限値 (t_{min}) および上限値 (t_{max}) によって定まる時区間表現である。A には、2 次元の矩形や円などの領域表現が記述される。これらのパラメータを指定することにより、ある領域の任意の時区間の時系列センサデータを取得できる。

SC (Spatial Constraints: 空間制約) は問合せ領域 (A) を分割するためのパラメータであり、TC

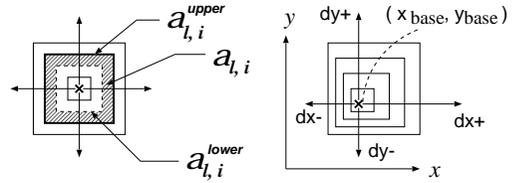


図 4 問合せ領域の分割と空間制約
Fig. 4 Query region decomposition and spatial constraints (SC).

(Temporal Constraints: 時間制約) は時区間 (TI) を分割するためのパラメータである。空間制約 (SC) によって分割された領域を順序付け、時間制約 (TC) によって分割された時区間を順序付けることで、検索結果の転送順序が決定される。TC と SC を指定することで、クエリ Q はサブクエリに分割される。

$$Q = \langle q_{1,1}, q_{1,2}, \dots, q_{2,1}, \dots \rangle \quad (3)$$

$$q_{l,i} = \langle a_{l,i}, DT, TSI \rangle \quad (4)$$

$a_{l,i}$ はサブクエリ $q_{l,i}$ に対する問合せ領域であり、 TSI は $q_{l,i}$ に対する時区間パラメータである。

まず、検索結果の空間分割に関しては、基本的に、筆者らが空間データ検索に利用した分割方法¹¹⁾に従う。

$a_{l,i}$ は、式 (6) の関係を満たす 2 つの領域表現 ($a_{l,i}^{upper}, a_{l,i}^{lower}$) の差分領域として、式 (5) のように表現される(図 4: 左図)。

$$a_{l,i} = \begin{cases} a_{l,i}^{upper} - a_{l,i}^{lower} & (i > 1) \\ a_{l,i}^{upper} & (i = 1) \end{cases} \quad (5)$$

$$a_{l,i}^{lower} \subset a_{l,i}^{upper}, a_{l,i}^{lower} = a_{l,i-1}^{upper} \quad (6)$$

問合せ領域の分割は、その領域表現に依存するが、我々はすでに空間データ検索に対する以下の 2 種類の分割方法¹¹⁾を実装しており、センサデータ検索に対しても同様の分割方法を用いる。

- 問合せ領域が円であるとき、円の中心を基準点として半径に沿って、その領域を分割する。円の中心座標と半径方向の分割幅を空間制約として指定する。
- 問合せ領域が矩形であるとき、分割の基準点と各座標軸方向の拡大幅を指定する(図 4: 右図)。

空間制約 (SC) として、分割の基準点とデータ収集領域の拡大方向を指定することで、ユーザの注目している領域から優先的にデータを取得することが可能となる。

次に、時区間分割についても、式 (7)、式 (8) に示すように、2 つの時区間 ($TSI_i^{upper}, TSI_i^{lower}$) の差分として分割時区間 TSI_i を定義する(図 5)。

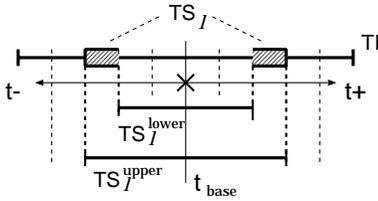


図5 時間制約に基づく区間分割

Fig. 5 Temporal interval decomposition based on temporal constraints (TC).

$$TS_l = \begin{cases} TS_l^{upper} - TS_l^{lower} & (l > 1) \\ TS_l^{upper} & (l = 1) \end{cases} \quad (7)$$

$$TS_l^{lower} \subset TS_l^{upper}, TS_l^{lower} = TS_{l-1}^{upper} \quad (8)$$

時間制約 (TC) として, 図5に示すように, 分割の基準点 (t_{base}) と各時間軸方向の拡大幅 (t_+ , t_-) を指定し, 式 (9) を用いて時区間 TS_l^{upper} を計算する.

$$TS_l^{upper} = [t_{base} + t_- \times l, t_{base} + t_+ \times l] \quad (9) \\ (t_+ \geq 0, t_- \leq 0)$$

時区間分割を用いる場合には, 優先的に情報を取得したい時刻を t_{base} として指定し, 情報の提供順序を決定するために, 拡大幅 t_+ , t_- を指定する. たとえば, 基準時刻 t_{base} として, $TI.t_{min}$ を指定し, 拡大幅として, $t_+ (> 0)$, $t_- = 0$ を指定することで, 時刻の古いデータから新しいデータという時間順序で検索結果を提示していくことができる. $t_- < 0$, $t_+ = 0$ を指定した場合には, 現在から過去にさかのぼった逆方向の情報提示が可能である. また, 基準時刻 t_{base} が必ずしもクエリに指定した時区間 TI の始点や終点とは限らないので, そのような場合には, t_+ と t_- の両方を指定する必要がある. この場合, ある時間帯を中心とした両方向への情報提供になる.

なお, 本論文では, センサデータを閲覧する際の, 時間情報を考慮することが重要であることをふまえ, 空間制約 (SC) と時間制約 (TC) の両方が指定された場合には, 検索結果は基本的に時区間分割に基づく順序で提供され, 各時区間の中で空間分割が利用されるものとする. これにより, センサデータの統合結果の空間的な分布を, 時間順序に従って閲覧することが可能となる. 空間制約 (SC) を指定しない場合には, 時区間分割に基づくデータ提供が行われ, 時間制約 (TC) を指定しない場合には, 空間分割のみが行われる.

3.3 空間制約と時間制約を考慮した検索品質の定義

本論文では, 空間制約 (SC) と時間制約 (TC) を考慮して, センサデータ検索の品質 ($q_{tc,sc}$) を式 (10)

のように定義する.

$$q_{tc,sc}(t) = \sum_{l=1}^m \sum_{i=1}^j \frac{N_{l,i}(t)}{N_{all}} \quad (10)$$

$N_{l,i}(t)$ は時刻 t における分割領域 $a_{l,i}$ に対する収集データ数である. N_{all} はそのクエリに対して収集される総データ数である. m, j に関して, TS_m は, 時区間制約を満たして検索結果が収集されている最新の時区間を示し, $a_{l,j}$ は, TS_l に対して, 空間制約を満たして検索結果が収集されている最新 (最大) の分割領域を示す. つまり, $TS_l (1 \leq l \leq m-1)$ の検索結果の収集が完了し, TS_m について $a_{m,i} (1 \leq i \leq j-1)$ の検索結果の収集が完了していることを示す. $q_{tc,sc}$ は, 時間制約と空間制約を満たしている領域のデータ収集率によって計算され, それらの制約に基づいて検索結果が転送される場合には単調に増加するが, そうでない場合には変化しない.

検索結果品質 $q_{tc,sc}$ を向上させるためには, 時間制約と空間制約を満たすように, センサデータの位置情報と時間情報を考慮して, 各サーバからの検索結果を同期して転送させる必要がある.

3.4 仲介エージェントによる検索結果の転送制御

本節では, $q_{tc,sc}$ を向上させるために, 仲介エージェントがサーバへのクエリの要求とサーバからの検索結果の転送を制御する方式 (本論文では, buf 方式と呼ぶ) について説明する.

3.4.1 処理手順

検索システムは, クライアントからのクエリ Q に対して, 次の手順に従って動作する.

- (1) まず, 時間制約 (TC) と空間制約 (SC) に基づいて, Q をサブクエリ $q_{l,i}$ に分割し, サブクエリのリストを作成する. リスト中のサブクエリは, 時区間分割に従って順序付けられ, 各時区間ごとに空間分割に従って順序付けられる. このとき, 検索結果を保持するためのバッファも同様の順序を持つように作成する.
- (2) 次に, $queue_size$ 個のサブクエリを, 該当するサーバエージェントにまとめて要求する. サブクエリの転送先のサーバの選択については, 3.4.2 項で述べる.
- (3) 各サーバからの検索結果を各時区間 TS_l に対する各領域 $a_{l,i}$ ごとにバッファに保持し, バッファの順序に従って検索結果をクライアントに転送する. $a_{l,i}$ に対する検索結果の転送を完了した後に, $a_{l,i+1}$ の検索結果の転送を開始し, サブクエリ $q_{l,i+queue_size}$ を該当するサーバに要求する.

k を分割領域数とするとき、 $i = k$ の場合には、次の時区間 (TS_{i+1}) の分割領域 $a_{i+1,1}$ の検索結果を転送する。 $(i + queue_size) = k$ の場合には、サブクエリ $q_{i+1,1}$ を要求する。

- (4) サブクエリ $q_{l,i}$ を受信したサーバは、各サブクエリに対する探索結果を分割時区間と分割領域の順序に従って逐次的に仲介エージェントに転送する。サーバでの探索については、3.4.3 項で述べる。

時間制約と空間制約を考慮したクエリの実行と検索結果転送の制御により、複数のサーバエージェントからの検索結果を、時区間ごと、領域ごとに同期させてクライアントが収集することが可能となる。さらに、サブクエリの先行要求と検索結果のバッファリングにより、複数のサーバエージェントからのデータ収集を効率的に行うことができる。

3.4.2 サーバエージェントの選択

サーバエージェントを選択するために、仲介エージェントは、まず、 $q_{l,i}$ に指定されたデータの種類 (DT) と各センサネットワークのデータの種類の ($:data-type$) を比較し、指定されたデータを提供できるセンサネットワーク s を探し出す。そして、そのセンサネットワーク s の位置情報 SNR_s と $q_{l,i}$ の問合せ領域 $a_{l,i}$ とが重なっているかどうか、すなわち、 $a_{l,i} \cap SNR_s \neq \phi$ が成り立つかどうか調べる。 $C_1 \equiv (a_{l,i}^{upper} \cap SNR_s \neq \phi)$ および $C_2 \equiv (SNR_s \subseteq a_{l,i}^{lower})$ という 2 つの条件を評価し、 C_1 が真、 C_2 が偽である場合に、センサネットワーク s を管理するサーバが選択される。

3.4.3 サーバエージェントによるデータ探索

サブクエリを受信したサーバエージェントは、location table を参照し、分割領域 ($a_{l,i}$) に含まれるセンサノードを探し出し、分割領域ごとにノード識別子 ($NODE_ID$) のリスト (ID_LIST) を生成する。センサノード v の位置 p_v について、条件 ($p_v \subseteq a_{l,i}$)、すなわち、式 (11) の条件が成り立つとき、ノード v を ID_LIST に加える。

$$(p_v \subseteq a_{l,i}^{upper}) \wedge \neg(p_v \subseteq a_{l,i}^{lower}) \quad (11)$$

そして、サーバエージェントは、 ID_LIST の要素をキーにして、時系列データテーブルを探索し、クエリの条件を満たすセンサデータを探し出す。テーブル中のセンサデータの更新時刻 t について、時刻 t が時区間 TS_i に含まれるかどうかを判断するために、式 (12) の条件が成り立つかどうかを調べる。

$$(t \subseteq TS_i^{upper}) \wedge \neg(t \subseteq TS_i^{lower}) \quad (12)$$

サーバエージェントによる探索結果は、時間属性

```
<!ELEMENT sensor-data (data-item)*>
<!ELEMENT data-item
      (id,type,location,time-series)>
<!ELEMENT id (#PCDATA)>
<!ELEMENT type (#PCDATA)>
<!ELEMENT location (x,y)>
<!ELEMENT x (#PCDATA)>
<!ELEMENT y (#PCDATA)>
<!ELEMENT time-series (temporal-item)+>
<!ELEMENT temporal-item (value,time)>
<!ELEMENT value (#PCDATA)>
<!ELEMENT time (#PCDATA)>
```

図 6 センサデータの XML 表現のための DTD

Fig. 6 XML expression for sensor data.

を持つデータ (temporal data) の系列、すなわち時系列データ (time series) として表現され、XML (eXtensible Markup Language) 文字列として仲介エージェントに転送される。仲介エージェントは、各サーバエージェントからの検索結果を、別々のパケットとしてクライアントエージェントに転送する。検索結果の XML 表現のための DTD (Document Type Definition) を図 6 に示す。

各検索結果は、 $\langle sensor_data \rangle$ タグによって囲まれ、 $\langle data_item \rangle$ タグの中に各ノードに対する時系列データ ($\langle time_series \rangle$) が記述される。 $\langle location \rangle$ タグによって各センサノードの位置情報が記述され、この位置情報に基づいてクライアントでの統合処理が行われる。

4. 実験および評価

4.1 実装

本検索機構を Java 言語 (JDK1.2) を用いて実装した。各エージェントは複数のスレッドによって実装され、検索結果の転送には TCP のコネクションを利用する。サーバエージェントは、起動時に読み込んだセンサノードの位置情報から location table を生成する。センサデータ検索時には、あらかじめ生成した時系列データを蓄積したファイルを参照する。検索システムを構成するエージェントは、表 2 に示すホスト (Sun Ultra WorkStation, Solaris OS) 上で動作させた。CA はクライアント、MA は仲介エージェント、SA はサーバエージェントである。

実験用のサンプルデータとして、気象庁提供の気象データ CD-ROM「アメダス観測年報 (2000 年)」(気象業務支援センター発行) に収録されている時別値データを利用した。この気象データ CD-ROM には、

表 2 測定環境
Table 2 Host specifications.

	CPU	メモリ	エージェント
host 1	440 MHz	1,024 MB	CA
host 2	333 MHz	384 MB	MA, SA
host 3	900 MHz (×2)	2 GB	SA

気温、降水量、風速などに関する 1 年分の時系列データが、月ごとに、観測点別にファイルに保存されている。各データファイルは 1 時間単位で順番に並んだデータレコードから構成される。

4.2, 4.4 節の実験では、アメダスの観測点をセンサノードとし、気温データ (temp) あるいは降水量データ (prec) を提供するセンサネットワークを想定する。サーバエージェントは、センサノードごとに時系列データを管理し、アメダスの観測点情報に記録されている経緯度座標に基づき location table を生成する。

なお、以降の実験では、2 点の座標で表現される矩形領域を問合せ範囲としてクエリに指定し、その矩形領域を、基準点の座標 (x_{base}, y_{base}) と各軸方向の分割幅 (dx_+, dy_+, dx_-, dy_-) によって表現される空間制約 (SC) を用いて分割する (図 4)。問合せ領域と空間制約は、経緯度座標系で指定し、単位は、度とする。また、クエリ (Q) に指定した時区間の長さを $t_{q,len}$ と表現する。

4.2 気象データを用いた実験

本節では、アメダスの気温ノードを防災気象情報サービス²⁾の区分に従い、北海道、東北、関東、中部、近畿、中国、九州地方という 7 つのグループに分割し、各グループに含まれるセンサノードから各センサネットワークが構成されると仮定する。各センサネットワークは、それぞれ 1 つのサーバエージェントによって管理され、7 つのサーバエージェントから日本全国の気温データが提供されるものとする。各サーバの管理する領域 (すなわち、各センサネットワークの SNR) と気温ノード分布を図 7 に示す。

4.2.1 検索結果品質の変化

まず、図 7 に示す分散センサネットワークを管理する複数のサーバエージェントに、仲介エージェントを介して、クエリを要求し、空間制約を指定したときの検索結果品質の変化について調べる。検索結果品質の変化は、クライアントエージェントが検索結果を受信した時刻 (t_{wait}) を記録し、各時刻に対して式 (10) で定義した $q_{tc,sc}$ を計算することによって求めることができる。

日本全国の気温分布を表示するためのクエリ ($A_{jp} = \langle 123, 24, 146, 46 \rangle$, $DT = \{temp\}$, $t_{q,len} = 20$, SC

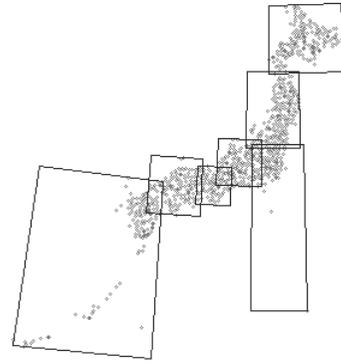


図 7 気温データを提供する分散センサネットワーク
Fig. 7 Multiple sensor networks for temperature data.

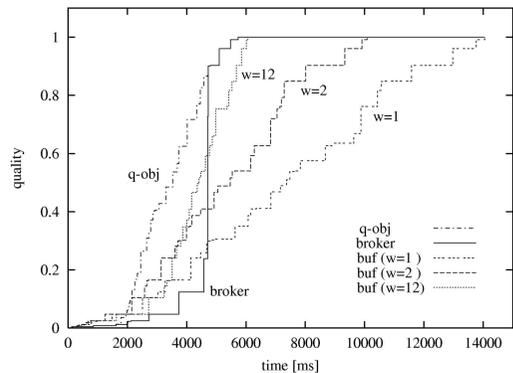


図 8 気温データを提供する分散センサネットワークに対する検索結果品質の変化

Fig. 8 The improvement of retrieval quality for temperature sensor networks.

$= \{(x_{base}, y_{base}) = (123, 24), dx_+ = dy_+ = 1\}$) を要求したときの検索結果品質 ($q_{tc,sc}$) の変化を図 8 に示す。図 8 の横軸は t_{wait} 、縦軸は $q_{tc,sc}$ である。

図 8 の buf は、3.4 節で説明した提案手法の検索応答を示す。w は queue_size である。図 8 より、 $w = 1, 2, 12$ いずれの場合も、検索結果品質が単調に増加していることが分かる。 $w = 1$ の場合には、 a_{i+1} に対するサブクエリの要求を、 a_i に対するすべての結果を受信するまで待つため、その待ち時間によるオーバーヘッドが大きい。 $w = 2, 12$ の場合は、各サーバにサブクエリを先行要求することで、サブクエリの実行開始時刻を早めることができるため、同期のための待ち時間を短縮し、総応答時間を短縮することができる。 $w = 12$ の場合には、探索コストの大きい領域のデータを先行要求しているため、 $w = 2$ の場合よりも総応答時間が短縮されている。

図 8 の broker は、各サーバがクエリの分割を行い、仲介エージェントがサーバからのサブクエリの実行

結果をそのままクライアントに転送する方式である．図 8 の $q\text{-obj}$ は，broker 方式を用いたときの単なるデータアイテムの収集率を示し， q_{obj} と broker 方式の $q_{tc,sc}$ の差は，空間制約を満たしていないデータアイテムの数に相当する．図 8 では，broker 方式を用いた場合に，データ収集の後半のある時刻に結果品質が急激に上昇している．このことは，空間制約を満たしていないデータアイテムが未処理のままクライアントで保持され，その時刻に大量のデータの処理を開始する必要があることを示す．一方，データ収集の初期の段階では，broker 方式と比較して buf 方式の検索結果品質が高く，これは初期の段階で収集される空間制約を満たしたデータアイテム数が broker 方式より buf 方式が多いことを示す．したがって，クライアントが空間制約に指定した順序で領域ごとに逐次的な統合処理を行いたい場合には，buf 方式を用いると受信データを効率良く処理できることが分かる．

4.2.2 位置情報に基づくセンサデータ統合

次に，位置情報に基づくセンサデータ統合の一例として，時系列データの集約演算の結果をメッシュ統合する場合を取り上げる．この例では，クライアントは，検索結果である時系列データを取得すると，センサノードごとに平均値，最大値，最小値などを算出するための集約演算を実行する．そして，各ノードに対する時系列データの集約演算の結果をメッシュの目(セル)ごとに統合し，その結果を表示する．

ここでは，日本全土の平均気温分布を統合結果として表示するために，前項の実験と同様のクエリを，クライアントが要求するものとする．

図 9 は，収集した気温データの集約結果である平均気温について，これらの集約結果の平均値を各セルごとに算出し，メッシュデータとして描画したものである．図 10 は，buf 方式を用いた場合の途中結果に基づくメッシュ統合の例を示す．メッシュは，第 1 次地域区画¹⁴⁾に相当する．

提案手法 (buf 方式) を用いた場合，図 10 → 図 9 のように，基準点 (123, 24) に近い領域から順番に逐次的なメッシュ統合を実現できる．これは，各分割領域に対する検索結果を受信するたびに，その領域と関連するセルのデータ収集の完了を判別できるためである．一方，各サーバがそれぞれ独立に返答する broker 方式では，各サーバからの検索結果は順番に取得できるが，複数のサーバからの検索結果を領域ごとに同期させながら収集することができない．したがって，検索結果受信時に，データ収集を完了した領域を判別できないため，途中結果に基づく逐次統合ができず，全

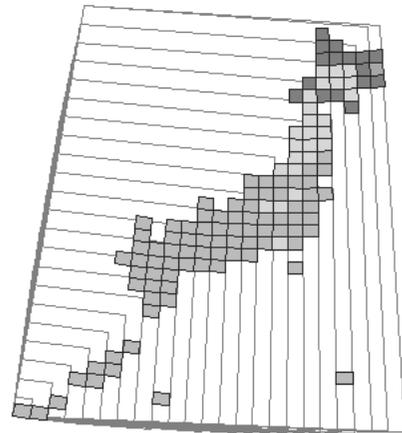


図 9 平均気温のメッシュ統合例 (buf 方式)
Fig. 9 An example of mesh integration ("buf" method).

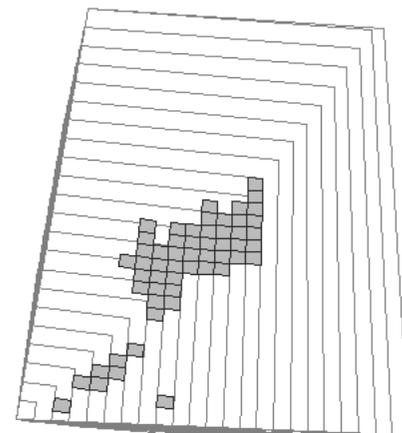


図 10 途中結果に基づく平均気温のメッシュ統合例 (buf 方式)
Fig. 10 An example of mesh integration based on intermediate results ("buf" method).

データの収集を完了するまで統合結果を表示することができない．結果として，各ノードに対する時系列の長さによっては，ユーザに対する応答時間の増大をもたらす．

データ収集を完了した領域を判別できることは，センサデータを空間的に集約する場合に有用であると考えられる．メッシュ統合に限らず，図 11 のように，地図データなどの空間データとの統合を行う場合にも，同様の議論ができる．分割領域ごとに検索結果を受信するたびに，その領域と統合対象であるポリゴンデータなどの空間データとの包含関係を調べることにより，受信したセンサデータを空間データと関連付け，その集約処理を完了させることができる．我々は，センサデータの空間集約アルゴリズムの実装も行っており，文献 15) の中で，ポリゴンデータとの空間統合および

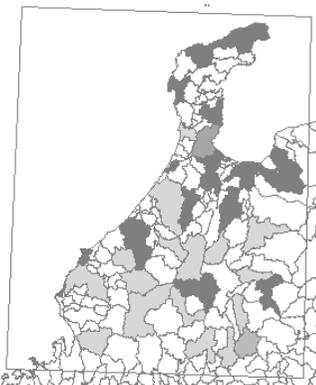


図 11 空間データ統合に基づく気温分布

Fig. 11 A result of spatial data integration for temperature data.

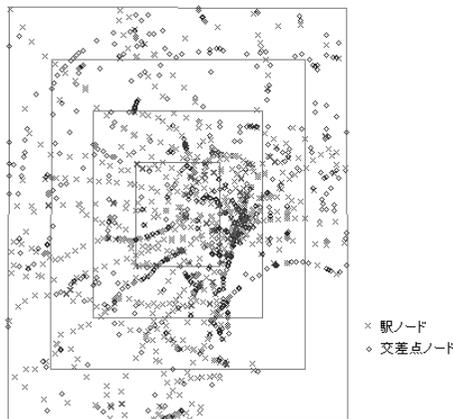


図 12 異種センサネットワークのノード分布の表示

Fig. 12 A distribution map of heterogeneous sensor nodes.

表 3 センサノード数

Table 3 The number of sensor nodes.

問合せ領域	駅ノード	交差点ノード	降水量ノード
A_{tokyo}	672	1009	8

空間補間に基づくメッシュ統合の手法について説明している。

4.3 異種センサデータ統合

本節では、異種センサネットワークを想定した環境での実験を行う。具体的には、国土数値情報¹⁶⁾の鉄道データから抽出した駅ノード、同じく道路データから抽出した交差点ノードを、利用客数や交通量などを出力するセンサノードと仮定する。本実験で想定する異種センサネットワークを構成するセンサノード数を表 3 に示す。 $A_{tokyo} = (139.5, 35.49, 139.9, 35.89)$ である。

本節の実験では、前節の実験で利用した気象センサネットワークの例よりも、ノード数が多く、密度の高い異種のセンサネットワークが互いに重なり合う状況を想定している。図 12 に、 A_{tokyo} に含まれるセンサノードの分布を示す。駅ノードは 1 個のサーバによって、交差点ノードについては県別に 7 個のサーバによって管理されているものとする。ここでは、これら合計 8 個のサーバに対して、東京近郊 (A_{tokyo}) のセンサデータを要求し、空間制約 ($SC = \{(x_{base}, y_{base}) = (139.7, 35.69), dx_+ = dy_+ = 0.05, dx_- = dy_- = -0.05\}$) を指定した場合について調べた。図 12, 図 13 は、それぞれ、最終的な結果であるノード分布, 途中結果に基づくノード分布を示す。

図 12 と 図 13 は、同一の問合せ領域 (A_{tokyo}) に関する表示結果であり、図 13 → 図 12 のように、基

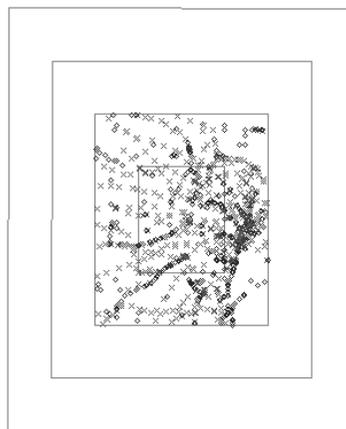


図 13 異種センサネットワークのノード分布の逐次表示 (“buf”方式)

Fig. 13 A distribution map of heterogeneous sensor nodes (“buf” method).

準点である A_{tokyo} の中心 (139.7, 35.69) に近い領域から、逐次的にノード分布が表示されることが分かる。

一方、図 14 に、broker 方式を用いた場合の同一のクエリに対する途中結果に基づくノード分布を示す。

この例では、各センサネットワークのノード分布は分割領域ごとに同期して表示されず、各サーバごとに基準点に近い領域から表示される。これは、各サーバの管理するセンサネットワークの配置やノード密度、さらにはサーバでの領域分割や検索応答が異なり、それらのサーバがそれぞれ検索結果を返すためである。結果として、ユーザの要求した順序で検索結果を収集できるとは限らない。

以上より、空間分割に基づく本手法は、配置 (ノード分布) やノード密度の異なるセンサネットワークが

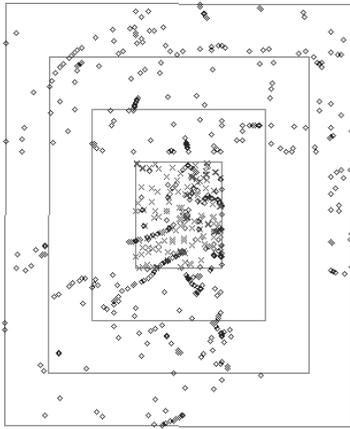


図 14 異種センサネットワークのノード分布の逐次表示 (broker 方式)

Fig. 14 A distribution map of heterogeneous sensor nodes ("broker" method).

混在する環境においても、空間制約を満たすように複数のサーバ(センサネットワーク)からのデータを領域ごとに同期させながら収集することができる。このことは、位置情報に基づく異種センサデータ統合をインクリメンタルに行ううえで有用であると考えられる。

4.4 時区間分割に基づく時系列データ統合

本節では、時区間分割に基づく時系列データ統合の例を扱う。実験データとして、アメダス気象データ CD-ROM に収録されている降水量データを用いた。

本節の実験では、時間制約 (TC) として、順方向の分割幅 (t_+) を指定し、クエリ (Q) に指定した時区間 (TI) の下限値 ($TI.t_{min}$) を基準時刻 (t_{base}) とする。時間制約の単位は、時とする。

4.4.1 時系列集約結果のインクリメンタルな結合

$A_{tp} = \langle 138.35803, 34.53578, 141.06116, 37.58538 \rangle$ の範囲の 120 時間分 ($TI = [t_{base}, t_{base} + 120]$) の降水量データを要求し、時間制約として $TC = \{t_+ = 12\}$ を指定したときの時系列データ統合の結果を図 15, 図 16, 図 17 に示す。ここでは、時系列データ統合として、積算降水量の分布の表示を扱う。クライアントは、各時区間のデータを受信するごとに、各時区間内の各ノードの降水量データを積算し、それまでの積算値に加算し、その統合結果を表示していく。このとき、各分割時区間 TS_i に関して、 $TS_i^{upper} = [t_{base}, t_{base} + 12 \times i]$ と表現され、たとえば、 $i = 5$ のとき、 $TS_5 = (TS_5^{upper} - TS_5^{lower}) = ([t_{base}, t_{base} + 60] - [t_{base}, t_{base} + 48])$ となる。

図 15, 図 16, 図 17 は、それぞれ TS_1, TS_5, TS_{10}

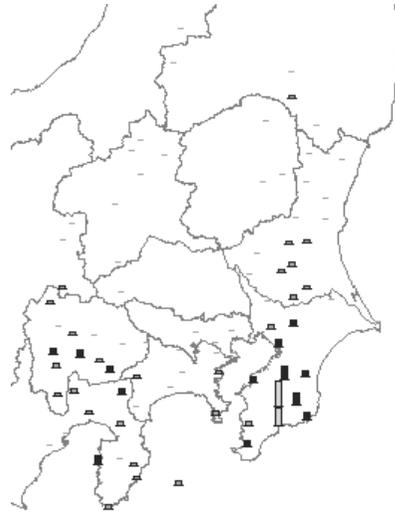


図 15 途中結果に基づく積算降水量分布 (TS_1 受信時)

Fig. 15 The total precipitation based on the intermediate results (at TS_1).

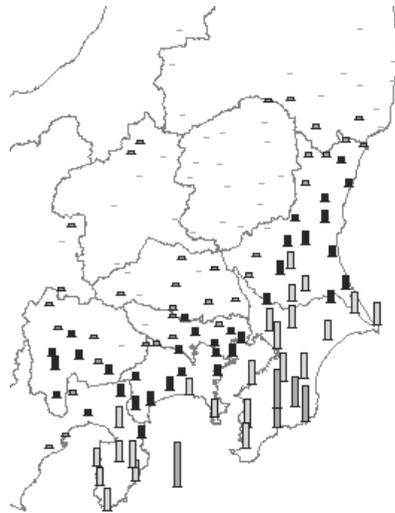


図 16 途中結果に基づく積算降水量分布 (TS_5 受信時)

Fig. 16 The total precipitation based on the intermediate results (at TS_5).

に対する途中結果に基づく積算降水量分布を示し、図 17 は最終的な 120 時間分 (すなわち 5 日分) の積算降水量分布を示すことになる。なお、図中の棒グラフは各ノードの積算降水量を示し、積算値を高さとして表現し、積算値を階級ごとに色分けしている

時区間分割を用いない場合には、図 17 が結果として提示されるのみであるが、120 時間分のデータを収集するため、その統合結果を表示するまでの応答時間が増大し、その間、何の情報も提供されない。それに対して、時区間分割を用いた場合には、時系列集約の

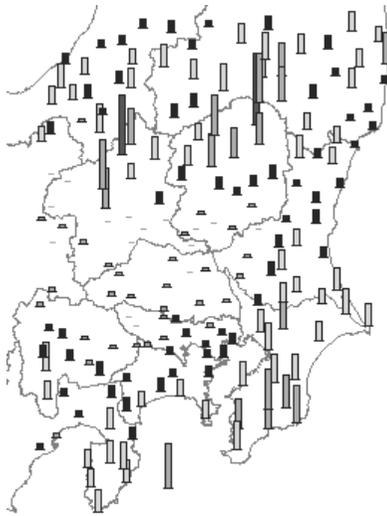


図 17 積算降水量の分布 (TS_{10} 受信後)

Fig. 17 A distribution of the total precipitation (after TS_{10}).



図 18 時区間 TS_8 に対する積算降水量分布

Fig. 18 The total precipitation based on the intermediate results (for TS_8).

途中結果を図 15 → 図 16 → 図 17 のように、インクリメンタルに提示することができる。さらに、“時区間” という意味のあるまとまりで、時間順序に従って、統合結果を表示しているため、時系列データの集約の過程を観察することもできる。たとえば、図 17 より、5 日間の雨の降水状況を把握することができ、最終的な結果としては、図の下部（南部）と上部（北部）で降水量が多いことが分かるが、図 15 と図 16 より、最初の 60 時間では南部の降水量が多く、後半の 60 時間で北部の降水量が多いことが分かる。

時系列データ収集時の応答時間が増大する場合に、検索結果を分割する基準として、空間分割を用いるか、時区間分割を用いるか、あるいは、その両方を用いるかを決定することは効果的な閲覧を実現するうえで重要である。本項で扱ったような途中結果の時間変化も含めて閲覧したい場合には、指定した全時区間の集約結果が領域ごとに表示される空間分割ではなく、時系列集約の途中結果を空間的な分布として表示できる時区間分割を用いることが適していると考えられる。

4.4.2 時系列集約結果の時間変化

4.4.1 項の実験が各時区間ごとの集約結果を結合していたのに対して、本節の実験では、日本全土 (A_{jp}) の 60 時間分の降水量データを要求し、時間制約として $TC = \{t_+ = 6\}$ を指定し、各時区間ごとの積算降水量の分布を表示させる。時区間 TS_8 , TS_9 , TS_{10} に対する積算降水量の分布を図 18, 図 19, 図 20 に示す。

この例では、時区間ごとにインクリメンタルに提供



図 19 時区間 TS_9 に対する積算降水量分布

Fig. 19 The total precipitation based on the intermediate results (for TS_9).

される検索結果に基づいて、それぞれの時区間での降水量分布を表示していく。したがって、図 18 → 図 19 → 図 20 のように、クエリに指定した時区間 (TI) の降水量の変化を時間順序に従って観察することができる。図 18 までは降雨地域が九州地方から東の方向に向かって徐々に拡大していく傾向が見られたが、その後、図 19, 図 20 に示すように、降雨地域が減少していく、すなわち指定した時区間の最後の 18 時間で、雨が上っていく様子が観察される。このように、時区間分割に基づいて提供されるセンサデータを利用して、時間順序に従って、ある範囲のセンサデータの変化だ



図 20 時区間 TS_{10} に対する積算降水量分布

Fig. 20 The total precipitation based on the intermediate results (for TS_{10}).

けでなく、降雨地域の移動や拡大状況などセンサデータの分布範囲が変化する様子も含めて観察することができる。したがって、時区間分割に基づく本手法は、センサデータ統合の結果をダイナミックに提示し、効果的なセンサデータプレゼンテーションを実現するうえで有用であると考えられる。

4.5 考 察

4.5.1 時区間分割に基づくプレゼンテーション

4.4.1, 4.4.2 項の例では、時間制約として t_+ を指定した場合を取り上げている。4.4.1 項の例では、各時区間の統合処理の完了ごとに、それまでの集約結果（積算降水量）のみを保存することで、図に示したような逐次的な表示を実現できる。4.4.2 項の例では、各時区間ごとに、その都度、集約結果を表示できるため、計算結果を保持する必要がない。逆に、各時区間に対する統合結果を保持しておくことで、受信したデータをすべて保持しなくても、すべての統合結果を表示した後で、ある時区間に対する統合結果を再度表示することができる。クライアントが、これらの途中結果を保持するのに十分な記憶容量を持っていれば、データサーバに対して、新たにデータ要求をしなくても、集約結果を再生することができる。さらに、センサデータを受信するごとに、ある時間幅（window）のセンサデータを保持し、その window に含まれる時系列データを統合していくことで、sliding window による時系列集約処理も実現可能である。たとえば、1 時間おきに過去 6 時間の降水量分布を連続して閲覧したい場合には、クライアント側で、すべての受信データを保

持しなくても、window を 1 時間単位で移動させながら、window に含まれる最新の 6 時間分のデータをつねに保持するようにすればよい。

t_- を指定することにより、現在に近い時刻から逆順序に過去にさかのぼって、時区間ごとの集約結果をアニメーションとして表示することができる。 t_- を指定した場合は、通常的时间軸と逆方向の情報提供になるが、 t_+ の場合と同様に、各時区間ごとに、集約結果を記憶すれば、任意の時区間に対する処理結果を参照することが可能である。

$t_+ (> 0)$, $t_- (< 0)$ の両方を指定した場合には、基準時刻 t_{base} を中心とした時間軸の両方向に広がる逐次結果表示となる。たとえば、1 日の最高気温や平均気温の分布を表示したい場合に、時区間分割のパラメータとして、 $t_{base} = 12$ 時、 $t_+ = 3$ 時間、 $t_- = -3$ 時間を指定する場合を考えてみる。この場合、正午を中心として、 $[12 - 3 * i, 12 + 3 * i] (i = 1, 2, 3, 4)$ という時区間単位で、統合の途中結果が表示される。一般に、気温は、夜間より昼間の方が高いため、最高気温分布を知りたい場合には、データ収集の初期の段階で、最終的な結果を閲覧できる可能性がある。平均気温分布を知りたい場合には、途中結果の変化の度合いから、気温変化の度合いを把握することができると考えられる。変化が少なければ、昼夜の気温差が小さいことになる。そのほか、台風接近時の降水量や風速の変化、通勤時間の前後での列車の混雑度や道路の渋滞状況を知りたい場合にも、 t_+ , t_- の両方を指定することが有用であると考えられる。

以上より、時区間分割に基づくデータ提供という枠組みの中で、受信データや途中結果の処理方法、時区間分割のパラメータの指定方法などを変えることで、多様なアプリケーション記述が可能であることが分かる。

4.5.2 空間分割と時区間分割の組合せ

4.2, 4.3 節では、空間分割のみを利用した場合について、4.4.1, 4.4.2 項では、時区間分割のみを用いた場合について、それぞれ説明している。ここでは、空間分割と時区間分割の両方を利用する場合について考察する。

提案手法は、時区間分割の中で空間分割を適用することができるため、時区間分割だけでは応答時間の増大に対応できない場合に、空間分割を併用することが考えられる。たとえば、4.4.2 項の実験では、6 時間ごとに検索結果をまとめて収集しているが、より長い時区間を単位としてデータを要求する場合には、応答時間の増大が予想される。そこで、空間分割を利用す

ることで、各時区間に対する検索結果を領域ごとに逐次的に提供することが可能となる。このとき、各時区間ごとに、統合結果の表示領域が拡大していく様子が観察される。

また、本論文の枠組みの中では、クエリに指定する問合せ範囲(A)と時区間(TI)は、統合処理が終了するまで変化しないと仮定しているが、これらのクエリパラメータをダイナミックに変化させることで、移動オブジェクトに対する検索をサポートできる可能性がある。さらに、現在の枠組みでは、時区間分割と空間分割は交互に順番に適用されるが、検索結果の転送順序を、より細かく柔軟に指定できるようになれば、多様なプレゼンテーションが可能になると考える。

たとえば、今後、車や人や生物など様々な移動オブジェクトに対してセンサが取り付けられ、それらのセンサからの出力データを利用できるようになれば、それらのセンサデータを活用したアプリケーションの構築が期待できる。車の移動量、人の混雑度、生物の生態などを調査するといったアプリケーションでは、移動するセンサからの出力データを収集するために、問合せ領域の位置や大きさなどを時間とともに動的に変化させられるようなクエリが必要である。そのような場合には、オブジェクトの移動に沿って時区間分割を行い、動的に変化する問合せ範囲に対して空間分割を適用することが考えられる。

実際に、この応用例を実現するためには、現在の枠組みをさらに発展させる必要があるが、多数の移動オブジェクトからのセンサデータを閲覧する場合には、このような時区間分割と空間分割を組み合わせることが効果的であると考えられる。

4.6 関連研究

4.6.1 時空間データベース技術との比較

本論文では、空間属性を持つ複数のセンサデータを統合する際のデータ提供方式について提案している。空間データの統合と表示に関する技術として、空間データベースの分野では、空間結合(spatial join)に関する研究が多数行われている^{17)~21)}。spatial joinでは、何らかの手段を用いて比較対象となる空間データを絞り込むための処理(filter処理)が行われ、候補として残った空間データに対して交差や包含などの空間関係(spatial relation)が調べられる。このfilter処理の異なる手法として、空間インデックスに基づく手法^{17),18)}や区画分割に基づく手法^{19)~21)}などが提案されている。空間インデックスに基づく手法^{17),18)}では、filter処理に、空間インデックスの各ノードの最小包圍矩形(Minimum Bounding Rectangle: MBR)を

利用する。一方、区画分割に基づく手法^{19)~21)}では、データの分布する空間を複数の区画(partition)に分割し、各区画単位で2つのデータセット内の空間データオブジェクトどうしを比較することで、filter処理を行っている。本論文で提案している手法では、センサデータを分割領域ごとに処理することで実現しているとも考えられ、その観点では、spatial joinに関する研究と類似しているといえる。

しかしながら、既存のspatial joinの手法の多くは、統合処理の途中結果を提供することを考慮していない。途中結果を逐次的に表示することは可能であるが、filter処理のためのデータ構造に依存することになる。たとえば、区画分割に基づく手法^{19)~21)}では、区画単位で表示され、文献17),18)の方法では、空間インデックスの構造に依存した順序で表示されるはずである。したがって、これらの手法では、必ずしも、ユーザの理解しやすい順序で統合結果を提示できるとは限らない。それに対して、本論文で提案している手法は、途中結果に意味を持たせるために、ユーザが指定した空間分割のパラメータに基づいて統合結果を表示するため、ユーザに対する視覚的効果も大きく、情報提示技術として有用であると考えられる。

時間データベース(temporal database)の分野では、時間属性を持ったデータ(temporal data)のための結合演算(temporal join)や集約演算(temporal aggregation)に関する研究が多数行われている。temporal joinに関する研究の中で、本論文の提案手法と関連性が高いものとして、区画分割に基づく手法があげられる^{22),23)}。これらの手法では、時間軸をpartitionと呼ばれる区画に分割し、各データセットから取り出したtemporal dataをpartitionに割り当てる。そして、各partitionごとに、別々のデータセットに属するデータどうしに対して結合演算を行うことで、全体の処理の効率化を図っている。また、temporal aggregationに関しても、同様に、区画分割に基づく手法が提案されている²⁴⁾。この手法は、時間軸をbucketと呼ばれる時区間に分割し、temporal dataを複数のbucketに関連付け、各bucketごとに処理を行う。

これらのtemporal join/aggregationに関する手法と本論文での提案手法は、統合対象であるtemporal dataをネットワークを介して取得するという違いはあるが、複数の時間データベースから取得したデータに対して、分割時区間ごとに統合処理を行っている点で、類似している。しかしながら、既存のtemporal join/aggregationの手法の多くは、結合演算や集約演

算の効率化を目的としており、各 partition ごとの計算結果または全体の計算結果を算出することに重点が置かれている。本論文では、むしろ、各時区間ごとの処理や計算結果の出力だけではなく、各時区間に対する統合処理の結果を、どのように表示していくかという統合結果の提供順序に着目している。提案手法では、ユーザが時間制約 TC を指定することで、ユーザにとって直感的で理解しやすい順序で、統合の途中結果を出力できる。したがって、時間属性を持つセンサデータの効果的なプレゼンテーションを実現するための統合技術として、既存の temporal join/aggregation 技術に比べて、提案手法が有用であると考えられる。

途中結果を利用するアプローチとしては、リアルタイムデータベースの分野で“approximate query processing”という手法が提案されている^{9),10)}。この手法は、リアルタイムシステムの分野で提唱された“imprecise computation”⁸⁾という“anytime algorithm”^{6),7)}と同様の概念に基づいており、時間が制限される状況において関係データベースに対する問合せの結果出力を逐次的に行うことを目的としている。文献 10)では、センサデータのような時間属性を持つデータ(temporal data)を扱えるモデルへの拡張が行われており、逐次的な結果出力の方法について説明しているが、その出力結果の処理方法までは言及していない。それに対して、本論文では、空間的かつ時間的なセンサデータ統合のための枠組みとして逐次的なデータ提供方式を提案している。空間分割と時区間分割のパラメータを指定することで、どのように検索結果を分割するかだけでなく、位置情報と時間情報を考慮して、センサデータ統合の結果を、どのように提示するかを制御することができる。また、本論文では、異種センサネットワーク環境を想定しているため、関係データベースシステム内の問合せ処理ではなく、連邦データベースの構成要素である仲介エージェントとサーバエージェントとの連携方法に焦点を当てている点も異なる。

4.6.2 センサデータストリーム技術との比較

データベースの分野では、近年、センサデータベースやセンサデータストリームに関する研究がさかんに行われている^{25),26)}。特に、センサからのストリームデータを対象とした研究がさかんであり、環境モニタリングなどのリアルタイムアプリケーションを構築するうえで重要である。本論文で提案している手法は、複数のデータリソースからのセンサデータを、時間情報と空間情報を考慮して、クライアントが処理しやすいように整列化している点で、センサデータストリー

ムに関する研究との類似点も見られる。空間制約や時間制約を満たす順序でセンサデータが提供されれば、データの受信ごとに逐次的に統合処理ができるが、そうでない場合には、複数のサーバからのデータを保持するためのオーバヘッドが大きくなることもある。

また、空間分割と時区間分割のパラメータを指定することにより、センサデータ統合の結果を表示する順序を制御できるという点では、マルチメディアデータのストリーミングや QoS 制御のための手法との類似性を指摘することもできる。提案手法は、センサデータを持つ空間属性と時間属性を扱っているという点が特徴的であるが、ネットワーク環境における品質制御手法を参考にしながら、より効果的な分散センサデータのプレゼンテーションを実現したいと考えている。

5. む す び

本論文では、位置情報システムが異種センサネットワークを利用するための要素技術として、インクリメンタルなセンサデータ提供方式を提案した。位置情報と時間情報の両方を考慮した逐次的なデータ提供は、時系列のセンサデータを収集し、統合する際の応答時間の増大に対処するだけでなく、効果的なセンサデータの閲覧を実現するうえでの情報提示(プレゼンテーション)技術として有用であることが、実験を通して示唆された。空間分割に基づくデータ提供は、位置情報に基づいて逐次的にセンサデータ統合を行う場合に、時区間分割に基づくデータ提供は、時間順序を考慮して時系列データ統合の結果を表示する場合に、それぞれ効果的である。また、空間分割と時区間分割に基づくアプローチは、クライアントが時間順に領域ごとに逐次的に統合処理をできるというだけでなく、ユーザがパラメータを指定することによって、その処理結果の提示順序を制御できるという点で視覚的な効果がある。これらのことから、提案手法により、インタラクティブかつ柔軟なセンサデータ閲覧を実現できると考えられる。本手法の応用として、今後、センサデータと空間データを利用した環境モニタリングやパーソナルナビゲーションのための位置情報システムを構築していきたいと考えている。

参 考 文 献

- 1) 国土交通省：川の防災情報。
<http://www.river.go.jp>.
- 2) 日本気象協会：防災気象情報サービス。
<http://www.tenki.or.jp/>
- 3) 環境省：大気汚染物質広域監視システム。

- <http://w-soramame.nies.go.jp/>
- 4) Shanmugasundaram, J., Tufte, K., DeWitt, D.J., Naughton, J.F. and Maier, D.: Architecting a Network Query Engine for Producing Partial Results, *Proc. 3rd International Workshop on the Web and Databases*, pp.17–22 (2000).
 - 5) Hellerstein, J.M., Haas, P.J. and Wang, H.J.: Online Aggregation, *Proc. ACM SIGMOD Conference*, pp.171–182 (1997).
 - 6) Boddy, M. and Dean, T.: Solving Time-Dependent Planning Problems, *Proc. 11th International Joint Conference on Artificial Intelligence*, pp.979–984 (1989).
 - 7) Zilberstein, S. and Russel, S.: Optimal composition of real-time systems, *Artificial Intelligence*, Vol.82, No.1-2, pp.181–213 (1996).
 - 8) Lin, K.-J., Natarajan, S., Liu, J.W.-S. and Krauskopf, T.: Concord: A System of Imprecise Computation, *Proc. 11th Annual International Computer Software and Applications Conference*, pp.75–81, IEEE (1987).
 - 9) Vrbsky, S.V. and Liu, J.W.S.: APPROXIMATE — A Query Processor That Produces Monotonically Improving Approximate Answers, *IEEE Trans. Knowledge and Data Engineering*, Vol.5, No.6, pp.1056–1068 (1993).
 - 10) Vrbsky, S.V.: A data model for approximate query processing of real-time databases, *Data & Knowledge Engineering*, Vol.21, pp.79–102 (1997).
 - 11) 白石 陽, 安西祐一郎: 空間検索のためのインクリメンタルなデータ提供手法, *情報処理学会論文誌: データベース*, Vol.42, No.SIG 15 (TOD 12), pp.15–29 (2001).
 - 12) 東京大学空間情報科学研究センター: CSV アドレスマッチングサービス. <http://spat.csis.u-tokyo.ac.jp/cgi-bin/geocode.cgi>
 - 13) Shiraishi, Y. and Anzai, Y.: Sensor Data Retrieval from Distributed Sensor Networks, *Proc. 2000 International Conference on Parallel and Distributed Processing Techniques and Applications*, pp.1501–1507 (2000).
 - 14) 日本地図センター: 数値地図ユーザーズガイド (第2版補訂版).
 - 15) 白石 陽, 安西祐一郎: インタラクティブなセンサデータ閲覧のためのオンライン空間集約手法, 第14回データ工学ワークショップ(DEWS2003) 論文集, 電子情報通信学会 (2003).
 - 16) 国土交通省: 国土数値情報ダウンロードサービス. <http://nlftp.mlit.go.jp/ksj/>
 - 17) Huang, Y.-W., Jing, N. and Rundensteiner, E.A.: Spatial Joins Using R-trees: Breadth-First Traversal with Global Optimization, *Proc. 23th International Conference on Very Large Data Bases*, pp.396–405 (1997).
 - 18) Lo, M.-L. and Ravishankar, C.V.: Spatial Joins Using Seeded Trees, *Proc. ACM SIGMOD Conference*, pp.209–220 (1994).
 - 19) Patel, J.M. and DeWitt, D.J.: Partition Based Spatial-Merge Join, *Proc. ACM SIGMOD Conference*, pp.259–270 (1996).
 - 20) Lo, M.-L. and Ravishankar, C.V.: Spatial Hash-Joins, *Proc. ACM SIGMOD Conference*, pp.247–258 (1996).
 - 21) Koudas, N. and Sevcik, K.C.: Size Separation Spatial Join, *Proc. ACM SIGMOD Conference*, pp.324–335 (1997).
 - 22) Soo, M.D., Snodgrass, R.T. and Jensen, C.S.: Efficient Evaluation of the Valid-Time Natural Join, *Proc. 10th International Conference on Data Engineering*, pp.282–292 (1994).
 - 23) Son, D. and Elmasri, R.: Efficient Temporal Join Processing Using Time Index, *Proc. 8th International Conference on Scientific and Statistical Database Management*, pp.252–261 (1996).
 - 24) Moon, B., Fernando, I., Lopez, V. and Immanuel, V.: Scalable Algorithms for Large Temporal Aggregation, *Proc. 16th International Conference on Data Engineering*, pp.145–154 (2000).
 - 25) Bonnet, P., Gehrke, J. and Seshadri, P.: Towards Sensor Database Systems, *Proc. 2nd International Conference on Mobile Data Management*, pp.3–14 (2001).
 - 26) Madden, S. and Franklin, M.J.: Fjording the Stream: An Architecture for Queries Over Streaming Sensor Data, *Proc. 18th International Conference on Data Engineering*, pp.555–566 (2002).

(平成 15 年 3 月 25 日受付)

(平成 15 年 7 月 9 日採録)

(担当編集委員 中谷 多哉子)



白石 陽（正会員）

平成 6 年慶應義塾大学理工学部電気工学科卒業。平成 8 年同大学大学院計算機科学専攻修士課程修了。平成 14 年同大学院計算機科学専攻博士課程単位取得退学。現在、慶應義塾大学理工学部情報工学科特別研究助手。空間データベース、地理情報システムに関する研究に従事。



安西祐一郎（正会員）

昭和 49 年慶應義塾大学大学院博士課程修了。昭和 63 年より慶應義塾大学理工学部教授、平成 5 年より理工学部長。平成 13 年より慶應義塾長。この間昭和 56 年～57 年カーネギーメロン大学客員助教授。計算機科学、認知情報処理過程の研究に従事。工学博士。電子情報通信学会、日本認知科学会、ACM、IEEE 等会員。
