

ベイジアンネットワークを用いた確率的データストリーム処理システムの提案

Proposal of a Probabilistic Data Stream Processing System using Bayesian Networks

佐藤 亮[♥]
北川 博之[♠]

川島 英之[♠]

Ryo SATO
Hiroyuki KITAGAWA

Hideyuki KAWASHIMA

本研究では不確実な状況を表現する手法であるベイジアンネットワークを、データベース内部で扱える抽象データ型として表現し、確率的関係データと関係データの統合を実現する。さらに、確率計算の効率化も実現する。同抽象データ型はベイジアンネットワーク特有の演算子を有し、かつ提案データ型から生じる確率的関係データは関係データと統合可能である。更に問合せに応じた確率計算の省略により、計算処理時間を削減する。

This research proposes an integration of relational data and probabilistic data produced by Bayesian networks that are the techniques for expressing uncertain situations. We also propose conversion of Bayesian networks into an abstract data type that can be treated in a database, and efficient improvement of probability calculation. This abstract data type has an associated operator to generate probabilistic relational data, which can be integrated with relational data. In addition, we propose an algorithm to accelerate probabilistic calculation by omitting unnecessary propagations.

1. はじめに

RFID や GPS などの様々なセンサデバイスが使われるようになりつつある。これらのセンサデバイスからは大量のデータストリームが生成される。これを用いて、実世界で生じるイベントの検知が行われている。さらに、検知したイベントから状況を推論する研究も行われている。この推論技法の代表的なものにベイジアンネットワーク [2] がある。ベイジアンネットワークを用いて

実世界の状況を認識するシステムは多数開発されている。その例には状況推定 [1]、麻酔行為表現 [4]、スパムフィルタ等がある。これらの従来研究は時間的変動のないイベントを対象としており、時間と共に移り変わるデータストリームを対象にしたものは筆者らの知る限り存在しない。そこで本研究は、ベイジアンネットワークの対象イベントをデータストリームへと拡張し、それを処理するシステムの実現を目標とする。この目標を達成するためには以下の3つの問題に挑む必要がある。

一つ目の問題は、ベイジアンネットワークから生成される推論結果である確率的データと既存のデータとの統合が容易ではない点である。二つ目の問題は、ベイジアンネットワークには時間の概念が無いため、データストリームを処理できない点である。三つ目の問題は、ベイジアンネットワークのノード数増加と共に確率計算の処理時間が増大する点である。

そこで、本研究では以上3つの問題を対処するために、以下の3つの貢献を行う。一つ目の問題に対応する第一の貢献は、ベイジアンネットワークを関係データベースに取り込み、ベイジアンネットワークが表すイベント生起確率を確率的関係データに変換するモデルを示すことにある。これにより確率的データストリームと関係データベース内のデータとの統合が可能に成り、ベイジアンネットワークとデータベースを同時に扱うユーザの作業量を大幅に減らせる。

二つ目の問題に対応する第二の貢献は、ベイジアンネットワークにイベント生起期間の概念を導入することである。これによりベイジアンネットワークが連続的な事象を扱えるようになる。ユーザはベイジアンネットワークに対する監視期間を設定すると同時に、イベントに対して有効期間を設定する。各イベントは、有効期間経過後は未発生状態に戻される。この有効期間の設定は従来のベイジアンネットワーク研究では無視されてきた点である。

三つ目の問題に対応する第三の貢献は、検索処理の高速化である。イベントの生起確率を検索するとき、通常のベイジアンネットワークでは全イベントに対しての確率伝播処理が終了するまでは生起確率の検索ができない。それゆえベイジアンネットワークのノード数が多い場合や、生起したイベント数が多い場合には、検索処理の応答時間が劣化する。そこで本研究では従来のアルゴリズムの制約を緩和することで効率的な検索処理を実現する。

本論文の構成は以下の通りである。2節ではベイジアンネットワークとデータストリームについて概説する。3節ではベイジアンネットワークをデータベースと統合するシステムと、提案システムに対する問合せ言語について述べる。4節で提案システムにおける効率的な問合せ手法を提案し、5節では提案手法についての評価実験とその結果を示す。6節では関連研究を述べる。最後に7節では結論を述べる。

[♥] 学生会員 筑波大学大学院システム情報工学研究科 punisiro@kde.cs.tsukuba.ac.jp

[♠] 正会員 筑波大学大学院システム情報工学研究科 / 計算科学研究センター kawasima@cs.tsukuba.ac.jp

[♠] 正会員 筑波大学大学院システム情報工学研究科 / 計算科学研究センター kitagawa@cs.tsukuba.ac.jp

2. ベイジアンネットワークとデータストリーム

本節では問題の前提として、ベイジアンネットワークとデータストリームについて概説する。

2.1 ベイジアンネットワーク

ベイジアンネットワークは不確実な状況を表現するために幅広く用いられている。因果関係のあるイベントの確率変数をノードとするグラフ構造と、各ノードに割り当てられた条件付確率分布群によってベイジアンネットワークは定義される。因果関係の有無は有向グラフで表される。また、因果関係の強さを条件付確率で表す。各変数の条件付確率分布は、本研究で対象とする離散的な確率変数の場合には、親ノードと子ノードの各変数がとる具体的な値ごとに割り当てられた条件付確率を離散的な表、すなわち条件付確率表 (Conditional Probability Table(CPT)) で表現する。

図 1 にベイジアンネットワークの例を示す。このベイジアンネットワークは A, B, C, D の 4 イベントで構成されている。

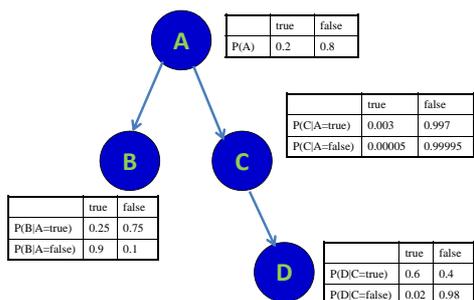


図 1 ベイジアンネットワークの例

ベイジアンネットワークの各ノードの確率値を求めるアルゴリズムは大きく分けて 2 種類存在する。それらは、近似推論アルゴリズムと厳密推論アルゴリズムである。近似推論アルゴリズムは、過去のイベント生起頻度などのサンプル数に依存する計算方法なので、サンプル数が少ないと精度が低下する。そこで、本研究では正確な確率値を求めるために、厳密推論アルゴリズムである Pearl's Message-Passing [2](以下、メッセージパッシング) を採用する。

2.2 データストリーム

近年、大量に流れ来るデータを効率的に処理する技術としてデータストリームが注目されている。データストリームに関する研究は 2002 年頃 [12] から始まった。当初は関係データ演算をストリームモデルに適用させ、処理系において性能を追求する研究が行われたが、現在では多様なアプリケーションに特化した専用

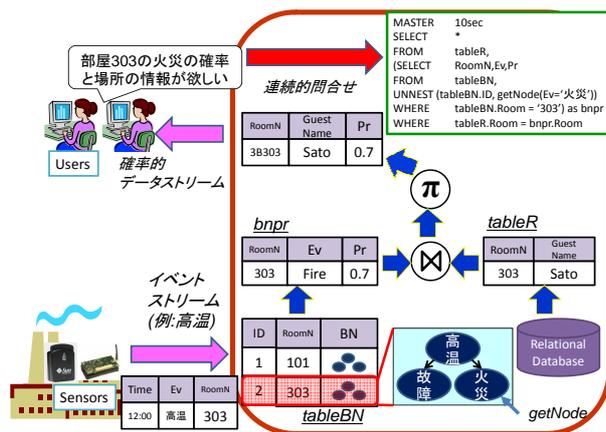


図 2 処理の流れ

問合せ言語を持つ専用システムに関する研究が多数行われている [9]。特に WaveScope[9] ではオブジェクト関係データモデルを採用し、SigSeg と名づけられた時系列データを処理するオブジェクトを導入している。SigSeg には FFT を代表とする多様な信号処理メソッドが用意されている。しかし、本研究が対象とする推論処理については WaveScope では未考慮である。

3. 確率的データと関係データの統合化

本節では、ベイジアンネットワークと関係データベースとの統合について述べる。

3.1 システムの処理の流れ

提案システムの動作を図 2 を用いてに述べる。処理の流れを述べる前に、連続的問合せとイベントストリーム対応定義について述べる (詳細は 3.3 節)。図 2 において、ユーザは「部屋 303 の火災の確率と場所の情報が欲しい」という連続的問合せを登録している。すでに済んでいるイベント対応定義により、入力イベントストリームは、3.2 節で述べるベイジアンネットワークオブジェクトを格納した BN 属性と部屋番号属性 (RoomN 属性) に関連付けられる。

それでは処理の流れを述べる。図 2 ではイベントストリームを受信している。受信したイベントストリームは tableBN 中の BN 属性と関連付けられた後、さらに RoomN 属性により関連付けが行われる。イベント中の RoomN が 303 であることから、tableBN 中の 2 行目のタプルが関連付けられる。そしてイベントには Ev 属性に「高温」が記されているため、同タプル中のベイジアンネットワークの根にあたるイベントが生起したと設定される。次に、同一ベイジアンネットワーク中の他イベントである「故障」と「火災」に関しての生起確率を、確率伝播により計算する。そして計算結果を確率的関係データの形式で出力する (図 2 中の bnpr テーブル)。bnpr テーブルは関係データである tableR と統合され、最後にユーザへ配信される。

3.2 オブジェクト関係データモデルによるベイジアンネットワークの表現

3.2.1 オブジェクト関係データモデル

ベイジアンネットワークと関係データベースを自然に統合するため、本論文ではオブジェクト関係データモデルを採用し、オブジェクトにベイジアンネットワークを格納する。ベイジアンネットワークはノードの構成や確率値や個々のノードが示すイベントの名前などの情報を有しているので、そのままでは関係データベース内で容易に扱うことができない。この問題を解決するためにオブジェクト関係データベースモデルを使用する。そのオブジェクト関係データベースのオブジェクトとして、ベイジアンネットワークの拡張データ型 (ADT) を作成する。そして、ADT として BN-Obj 型を定義し、ベイジアンネットワークのオブジェクトを格納する。BN-Obj を作成することにより、今まではデータベースに蓄えられていなかったベイジアンネットワークをデータベース内に格納可能になる。データベースにベイジアンネットワークを格納することで、関係テーブルに格納されているデータと同じようにベイジアンネットワークを扱うことができ、演算子 (選択, 射影, 結合等) を使用可能になる。

ただし、BN-Obj 型はオブジェクト型なので、ベイジアンネットワーク中のデータに対する問合せに対応できない。これを解決するために、本提案システム特有のメソッドを BN-Obj 型に用意し、BN-Obj をタプルに変換することで、問合せに対応する。この詳細は 3.2.2 節で述べる。

3.2.2 タプル生成メソッドの定義

グラフ構造である BN-Obj 型から確率値を有するタプルを抽出するためのメソッド, getNode() を本節で定義する。

getNode(条件) getNode(ノード ID or 確率値 or イベント名, 符号, 値) は検索条件としてノード ID, 確率値, イベント名を使うことができる。符号は $<$, \leq , $=$, \neq , $>$, \geq のいずれかである。たとえば、生起確率が 90% 以上の全イベントを抽出してタプルに変換したいときには、getNode(Pr \geq 0.9) と記述すれば良い。

3.2.3 ベイジアンネットワークモデルの拡張

2.1 節で定義したベイジアンネットワークのモデルには生起期間の概念が無い。そこで、ベイジアンネットワークに対する連続的問合せを実現するために、Lifespan を導入する。Lifespan はイベントの生起期間を表す。Lifespan の例を図 3 に示す。図 3 中の下向矢印はノードの生起を示し、右向矢印の長さはイベント生起期間を示す。図 3 において、時刻 t_2 においてはイベント A のみが生起している状態を表す。時刻 t_3 ではイベント A とイベント B が生起している状態を表す。そして時刻 t_4 ではイベント B が生起している状態を表す。すなわち、問合せが時刻 t_2 , t_3 , t_4 で実行されたとすると、それぞれイベント A, イベント A とイベント B, イベント B が生起しているとみなされてベイジアンネットワーク中で確率伝播が実行される。

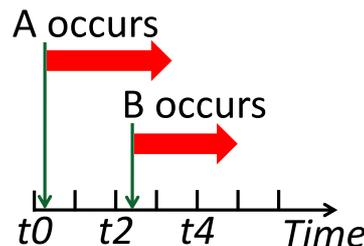


図 3 Lifespan

RoomN	Ev	Pr
103	A	0.7
103	B	0.8
103	D	0.6

図 4 問合せ例 1

3.3 問合せ言語

我々の提案モデルはオブジェクト関係データモデルであり、確率的データと関係データを統合するため、インタフェースである問合せ言語は、構文をユーザが簡単に記述できるよう SQL ライクにした。この例を図 4 に示す。

図 4 の問合せ例は、R というテーブルから、RoomN = '103' であるタプルに格納されているベイジアンネットワークに対して、Pr > 0.5 のイベントの情報を 10 秒間隔で要求している。この時 Pr > 0.5 のイベントが Ev=A, Ev=B, Ev=D だとする。この問合せ結果として図 4 の右に示す確率的データストリームがユーザに配信される。

3.3.1 各節の説明

図 4 を用いて各節の説明をする。

MASTER 節 MASTER 節では、提案システムに対する問合せを実行するタイミングを定義する。Master は時間幅のパラメータにより決定される。例えば 10 秒毎や、1 時間毎などである。Master を導入することにより、ストリームを処理可能になる。

SELECT 節 SELECT 節では、射影演算, 集約演算を記述する。

FROM 節 FROM 節は、問合せ対象とするテーブル名を記述する。BN-Obj 型に対する処理である getNode() メソッドはここで記述する。また、ベイジアンネットワークから生成されるタプルが複数存在する可能性があるため、UNNEST 操作を用いて平坦化を行うことにより、第一正規形に変換する。UNNEST 操作については SQL2003 に関する参考文献 [15] を参照されたい。

WHERE 節 WHERE 節では、FROM 節で選択したテーブルに対しての選択演算処理を記述する。

3.4 入力イベントストリームとベイジアンネットワークとの関連付け

入力イベントストリームと、ベイジアンネットワークに含まれるイベント (BN イベント) を関連付けるために、ASSOC 構文

```

ASSOC Stream.Ev WITH
SELECT BN-Obj
FROM tableR
WHERE tableR.RoomN = Stream.RoomN

```

図5 入力イベントストリームとベイジアンネットワークを関連付ける構文

を提案する。ASSOC 構文の例を図5に示す。ただし、入力イベントストリームには、Ev という名前の属性があることを前提とする。

この場合、BN イベントを同定するには、表、行、属性、そして BN イベントの名前を指定する必要がある。このうち、BN イベント以外の指定は図5中の WITH 以降で行われる。すなわち、表、行、属性の指定はそれぞれ FROM 節、WHERE 節、SELECT 節でおこなわれる。そしてイベントストリームと BN イベントの関連付は ASSOC 節で行われる。この場合、Stream.Ev と BN-Obj 内のイベントが対応付けられる。例えば Stream.Ev が「高温」であった場合、それは BN-Obj 中の「高温」に該当するイベントへと関連付けられる。

単に関連付を行うだけならば INSERT 文や UPDATE 文のように文を用いる方法もある。しかし、我々が提案するシステムにおいては、入力イベントストリームが直接処理される場合も考えられる。それゆえ、1つの入力イベントストリームに対して両方の処理を同時に実行可能にするために、ASSOC 構文を提案する。

4. 問合せ処理の効率化

この節ではベイジアンネットワークの計算コストを削減するために確率計算の効率化について述べる。通常確率計算で使用されるメッセージパッシングは Algorithm1(単純法)のように処理される。Algorithm1(単純法)では、生起している全てのイベントの更

Algorithm 1 単純法

- 1: for all occurring event do
- 2: 確率値を 1 に設定
- 3: 確率伝播を実行
- 4: end for
- 5: for all getNode do
- 6: 確率値を検索
- 7: end for

新処理終了を待たなければならない。ゆえに、イベント数やノード数が多くなるほど確率計算の処理時間は長くなる。

問合せ処理の高速化のために、メッセージパッシングの伝播処理を途中で省略し効率化を行う。通常メッセージパッシングでは、イベントが生起したノードから全てのノードに対して親ノードなら メッセージ、子ノードなら メッセージを送信し伝播を行う [2]。つまり問合せの対象外となるノードも厳密に計算をしている。そこで、計算時間を削減するために確率伝播が問合せ対

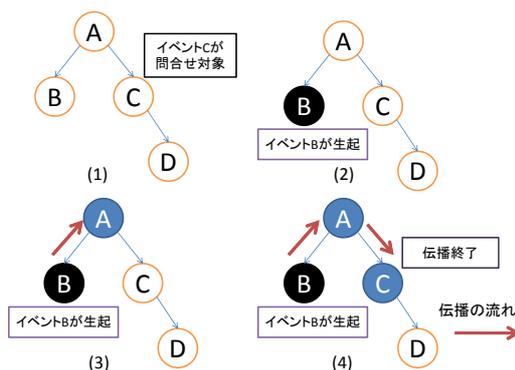


図6 アルゴリズム2:省略法の図

象となるノードまで終われば計算を打ち切る、省略法を提案する。

省略法を Algorithm 2~4 に示す。Algorithm 1 の3行目の確率伝播を開始すると Algorithm 2 の1行目に移る。まず、親ノードが存在すれば メッセージを送り、Algorithm 3 に移動する。ルートノードまで到達したら、子ノードが存在するか検索し、子ノードが存在したら Algorithm 4 に移動する。Algorithm 3 ではメッセージに基づき 値を求め、その 値と 値を使用して確率計算をする。ノードの確率計算が終わったらまた メッセージやメッセージを送る。Algorithm 4 でも同様に メッセージに基づき 値を求めて、その 値と 値を使用し確率計算をする。ノードの確率計算が終わって子ノードが存在したら Algorithm 4 を実行し伝播がされる。

このアルゴリズムは確率伝播の流れが1方向であるのに着目している。問合せ対象ノードの計算終了後、メッセージやメッセージを送信しても、問合せ対象ノードの確率値は変化しない。従って、問合せ対象ノードの確率計算終了後に、確率伝播終了の命令を挿入することで、確率伝播の省略ができ、計算コストの削減が可能になる。この手法の例を図6を用いて説明する。図6はノード Id が A, B, C, D の4ノードで構成されるベイジアンネットワークであり、(1)はその初期状態を表す。ユーザはノード C の確率値を求める問合せを登録している。この時、(2)のようにノード B が生起したとすると、(3)のようにノード A に確率が伝播する。次に、(4)のようにノード A からノード C に確率が伝播する。単純法ならば、次はノード D に伝播するが、提案手法の場合には問合せ対象のノードまで伝播しているので確率伝播を終了し、計算コストを削減する。尚、提案手法の計算精度は単純法と等しい。

5. 評価実験

ベイジアンネットワークの確率計算削減を実現する提案手法を評価する為に、単純法と省略法を Java 言語で実装した。実験には大規模ベイジアンネットワークが必要であるため、ベイジアンネットワークを生成するプログラムを作成し、使用した。作成し

Algorithm 2 省略法

```

1: for all parent do
2:   Algorithm 3(send_λ_msg_omit) を実行
3: end for
4: for all child do
5:   Algorithm 4(send_π_msg_omit) を実行
6: end for

```

Algorithm 3 省略法 (send_λ_msg_omit)

```

1: λ 値を計算
2: 確率を λ 値と π 値を用いて計算
3: if getNode searches this node then
4:   確率伝播を終了
5:   return
6: end if
7: Algorithm 2 を実行

```

Algorithm 4 省略法 (send_π_msg_omit)

```

1: π 値を計算
2: 確率を λ 値と π 値を用いて計算
3: if getNode searches this node then
4:   確率伝播を終了
5:   return
6: end if
7: for all child do
8:   Algorithm 4 を実行
9: end for

```

たベイジアンネットワークの構造は木とした。子ノードの数は乱変数で決定し、個数は 1~4 個、深さは固定しなかった。評価環境として、PC(Athlon Dual-Core 2.00GHz CPU, 1982MB RAM, Windows Vista OS) を用いた。

省略法が実際に計算コストを削減できているか実験的に評価した。問合せは図 7 の構文を用いた。図 7 の構文の内容は、3 秒ごとに部屋 103 の火災情報を要求する。条件は問合せ間隔の 3 秒ごとに全ノード数の 10%、50%、90% の確率でイベントが生起する 3 つの条件で実験を行った。例えばノード数が 300 で生起率が 90% ならば、270 のノードでイベントが生起する。実験結果を図 8 に示す。図 8 中の各記録点は 10 回の実験結果の平均値である。図 8 の横軸はベイジアンネットワークのノード数を表し、縦軸は問合せ処理時間を表す。イベント生起率が 10% の場合には、省略法は単純法よりも処理時間を最大で 56.4% 短縮し、イベント生起率が 50% の場合には、最大で 61.9% 処理時間を短縮し、イベント生起率が 90% の場合には、最大で 52.6% 処理時間を短縮した。即ち、提案手法はどの場合にも処理時間を削減できた。

```

MASTER 3sec
SELECT Ev,Pr
FROM   tableR,
      UNNEST(tableR.ID,bn.getNode(Ev=' 火災' ))
WHERE  tableR.Room = '103'

```

図 7 問合せ例 4

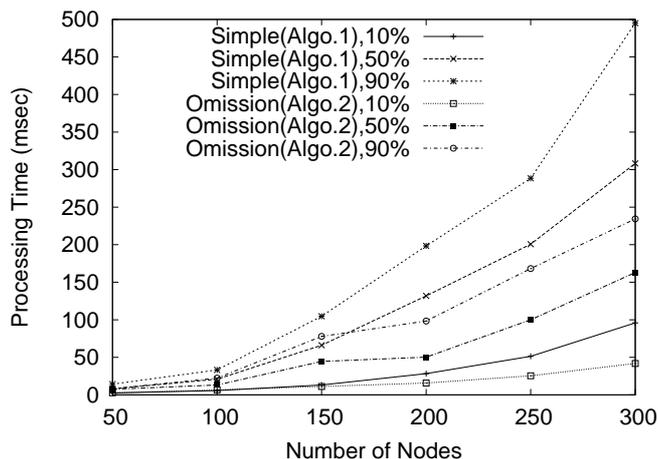


図 8 実験結果

6. 関連研究

本システムと同じように、データストリームを連続的に処理するシステムは多数存在する。代表的なものでは、STREAM [5], Aurora[6], TelegraphCQ[7], Borealis[8], WaveScope[9], PEEX[16], そして StreamSpinner[3] 等がある。これらのシステムではデータストリームに対する連続的問合せ処理を行うが、不確実な状況を表す確率的データストリームについては考慮していない。

確率的データベースに関する研究として、Trio プロジェクトの ULDB[14] が存在する。ULDB のデータモデルは従来の確率的データベース同様に、テーブル中の各タプルに確率値を付与するモデルであり、各確率値は 1 つのイベントの発生確率を表す。ULDB の新しい点は、Lineage を用いて安全な問合せ計画を実現すると同時に、確率計算の高速化を実現している点にある。安全な問合せ計画とは、正しい確率計算結果を出力する計画である [13]。本研究と ULDB の違いは 2 点ある。まず、本研究が ULDB よりも優れている点は、ベイジアンネットワークという確率計算モデルを新たにデータベースに取り入れることで関係データモデルのみを用いる ULDB よりも広いデータモデルを扱っている点にある。次に、本研究が ULDB よりも劣る点は、ULDB と異なり確率的関係データ処理に Lineage を用いない点にある。Lineage の本研究への導入については今後の課題とする。

7. 結論

実世界の状況を連続的に監視するシステムを支援するデータ処理基盤技術の開発を目標として、本論文ではベイジアンネットワークと関係データベースを統合したシステムを提案した。提案システムの新しい点は、入力イベントストリームに基づいて、それが関連する全イベントの生起確率を、確率的関係データの形式で得られることである。また、イベント生起確率を得る計算過程で無駄を省くことにより、計算時間を最大で 61.9% 削減できることを実験により確かめた。

今後の課題は、BN-Obj を格納可能なオブジェクト関係データベースシステムを構築し、それを多種センサデータを処理する基盤として適用することである。

謝辞 本研究の一部は科学研究費補助金基盤研究 (A)(# 18200005)、若手研究 (B)(# 18700096) による。

[文献]

- [1] Masaya Kadota, Hiroto Aida, Jin Nakazawa and Hideyuki Tokuda, D-JENGA: A Parallel Distributed Bayesian Inference Mechanism on Wireless Sensor Nodes. Proc. of the International Conference on Networked Sensing Systems (INSS), 2006.
- [2] Richard E. Neapolitan. Learning Bayesian Networks, Prentice Hall, 2004.
- [3] Shinichi Yamada, Yousuke Watanabe, Hiroyuki Kitagawa, and Toshiyuki Amagasa. Location-Based Information Delivery using Stream Processing Engine StreamSpinner. Proc. of the International Conference on Mobile Data Management (MDM), 2006.
- [4] 白鳥成彦, 奥出直人. ダイナミックベイジアンネットワークを用いた麻酔行為の表現. 人工知能学会全国大会, 2006
- [5] The STREAM Group. STREAM: The Stanford Stream Data Manager, IEEE Data Engineering Bulletin, 2003.
- [6] Daniel J. Abadi, Donald Carney, Ugur Cetintemel, Mitch Cherniack, Christian Convey, Sangdon Lee, Michael Stonebraker, Nesime Tatbul and Stanley B. Zdonik. Aurora: a New Model and Architecture for Data Stream Management, VLDB Journal, 2007.
- [7] Sirish Chandrasekaran, Owen Cooper, Amol Deshpande, Michael J. Franklin, Joseph M. Hellestein, Wei Hong, Sailesh Krishnamurthy, Sam Madden, Vijayshankar Raman, Fred Reiss and Mehul Shah, TelegraphCQ: Continuous Dataflow Processing for an Uncertain World, Proc. of the Conference on Innovative Data Systems Research (CIDR), 2003.
- [8] D. Abadi, Y. Ahmad, M. Balazinska, U. Cetintemel, M. Cherniack, J.-H. Hwang, W. Lindner, A. Maskey, A. Rasin, E. Ryzkina, N. Tatbul, Y. Xing, Stan Zdonik, Design of the Borealis Stream Processing Engine, Proc. of the Conference on Innovative Data Systems Research (CIDR), 2005.
- [9] Lewis Girod, Yuan Mei, Ryan Newton, Stanislav Rost, Arvind Thiagarajan, Hari Balakrishnan, and Samuel Madden, The Case for a Signal-Oriented Data Stream Management System, Proc. of the Conference on Innovative Data Systems Research (CIDR), 2007.
- [10] J. Pearl. Probabilistic Reasoning in Intelligent Systems: Network of Plausible Inference. Morgan Kaufmann, 1988.
- [11] G Piriou, et al. Recognition of Dynamic Video Contents with Global Probabilistic Models of Visual Motion, IEEE Transactions on Image Processing, 2006.
- [12] Brian Babcock, Shivnath Babu, Mayur Datar, Rajeev Motwani and Jennifer Widom, Models and Issues in Data Stream Systems. Proc. of ACM Symposium on Principles of Database Systems (PODS), 2002.
- [13] N. Dalvi and D. Suciu. Efficient Query Evaluation on Probabilistic Databases. Proc. of International Conference on Very Large Data Bases, 2004.
- [14] Omar Benjelloun, Anish Das Sarma, Alon Halevy, Jennifer Widom, ULDBs: Databases with Uncertainty and Lineage. Proc. of International Conference on Very Large Data Bases, 2006.
- [15] 土田正士, 小寺孝: SQL2003 ハンドブック. ソフト・リサーチ・センター. 2004.
- [16] Nodira Khoussainova, Magdalena Balazinska, Dan Suciu. Probabilistic RFID Data Management. UW CSE Technical Report UWCSE-07-03-01. March 1, 2007.

佐藤 亮 Ryo SATO

筑波大学大学院システム情報工学研究科在学中. 2008年筑波大学第三学群情報学類卒業. センサデータ管理に関する研究に従事. 日本データベース学会学生会員.

川島 英之 Hideyuki KAWASHIMA

2005年慶應義塾大学大学院理工学研究科開放環境科学専攻後期博士課程修了. 同年, 慶應義塾大学理工学部助手. 2007年筑波大学大学院システム情報工学研究科講師, ならびに計算科学研究センター講師. 博士(工学). センサデータ管理に関する研究に従事. 情報処理学会, ACM 各会員.

北川 博之 Hiroyuki KITAGAWA

1978年東京大学理学部物理学科卒業. 1980年同大学理学系研究科修士課程修了. 日本電気(株)勤務の後, 1988年筑波大学電子・情報工学系講師. 同助教授を経て, 現在, 筑波大学大学院システム情報工学研究科教授, ならびに計算科学研究センター教授. 理学博士(東京大学). 異種情報源統合, XMLとデータベース, データマイニング, センサデータベース, WWWデータ管理等の研究に従事. 著書「データベースシステム」(昭晃堂), 「The Unnormalized Relational Data Model」(共著, Springer-Verlag)等. 日本データベース学会理事, 情報処理学会フェロー, 電子情報通信学会フェロー, ACM, IEEE-CS, 日本ソフトウェア科学会, 各会員.