一般社団法人 電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
IEICE Technical Report
DE2012-20 (2012-08)

確率的データストリームに対する問合せ処理手法

加藤　翔† 石川　佳治†,††

† 名古屋大学
†† 国立情報学研究所
E-mail: †kato@db.itc.nagoya-u.ac.jp, ††ishikawa@itc.nagoya-u.ac.jp

あらまし センサ機器の普及により，それらのセンシングにより取得される情報が爆発的に増えており，センサデータ問合せ処理技術が重要となってきている．センサデータの多くはデータストリームであり，リアルタイムに問い合わせされる場合もあれば，蓄積されて後で問い合わせされる場合もある．データストリームは従来のデータとは異なる性質を持つことから，より表現能力の高い問合せ言語が必要とされている．そのような背景から，SASE+ というパターンマッチング言語が提案されている [3]．SASE+ 言語は，従来の選択，結合，集約では表現できなかったクリーネ閉包の表現が可能であり，また，複数の戦略が用意されているのでアプリケーションに適した戦略を選ぶことができるなど，高い表現能力をもつパターンマッチング言語である．センサデータは，そのセンシングの過程でノイズやデータ欠損が発生することから，必ずしも正確であるとは限らないという問題もある．そのため，センサデータはしばしば統計モデルなどを用いた生データの処理が行われる．そこで本研究では，加工されたデータストリームの問合せ処理に焦点を当てる．特に，*Markovian Streams* [1], [2] で提案されているような，確率モデルに基づきセンサデータストリームを処理した結果である確率的データストリームを対象として考える．本稿では，この確率的データストリームを SASE+ 言語を拡張して処理する手法について述べる．また，データストリームはリアルタイムの問合せに焦点が当てられることが多いが，本研究では一旦蓄積して後から問合せを行う場合についても考察する．

キーワード 確率的データストリーム，SASE+，NFA^b オートマトン

Query Processing over Probabilistic Data Streams

Sho KATO† and Yoshiharu ISHIKAWA†,††

† Nagoya University
†† National Institute on Informatics
E-mail: †kato@db.itc.nagoya-u.ac.jp, ††ishikawa@itc.nagoya-u.ac.jp

Abstract Query prosessing techniques for sensor data have become important, because the sensor data have increased explosively by growth of sensor devices. Most of sensor data is represented as data streams queried in a real-time manner or a retrospective manner using the stored data streams. Richer query languages over data streams are requierd because of the nature of data streams that is diffrent from that of traditional data. Based on this background, a query language for pattern matching over data streams, called SASE+ was developed [3]. SASE+ is a richer language that contains constructs for expressing the Kleene closure and strategies for selecting relevant data from an input stream mixing relevant and irrelevant data. Since sensor data may be inaccurate because of sensing noise or lack of data, we often apply preprocessing to raw data using a statistical model and so on. Therefore, we focus on query prosessing over preprosessed data streams, particularly probabilistic data streams that are the results of preprosessing using probabilistic model, which are proposed in Markovian Streams [1], [2]. In this paper, we propose a query prosessing method over this probabilistic data stream model using extended SASE+, and also discuss query processing for stored data streams.

Key words Probabilistic data streams, SASE+, NFA^b automaton

1. まえがき

今日では，センサ機器が広く普及しており，さまざまな環境下においてセンサ情報を大量に取得できるようになった．また，ネットワーク技術の発展により，従来はその場の処理のみに利用されていたセンサ情報がネットワークを介して収集されるようになり，さらに多くのセンサ情報を手に入れることが可能となった．そのため，センシングにより得られる大量の情報に対する分析のためのセンサデータ問合せ処理技術が重要となってきている．センサデータの多くはストリーム形式で表現され，リアルタイムに問合せを行うこともあれば，蓄積して後で問合せを行う場合もある．

データストリームへの問合せにおいては，従来の RDBMS 技術の表現能力では不十分な面があることから，データストリームの問合せ表現により適したパターンマッチング言語の研究が行われている [3]．また，問合せの評価においても，選択，結合，集約といった従来手法はデータストリームに対しては効率的でなく，より適した評価モデルの研究が行われている [4]．

本研究では，自律移動型ロボットに搭載の各種センサにより取得される位置データストリームに対して問合せを行う状況を考える．センシングによって得られたデータは必ずしも正確であるとは限らず，ノイズやデータ欠損をしばしば生じるため，センシングデータには統計処理による加工が施される．そこで本研究では，加工されたデータストリームの問合せ処理に焦点を当てる．特に，*Markovian Streams* [1], [2] で提案されているような，確率モデルに基づきセンサデータストリームを処理した結果である **確率的データストリーム** を対象として考える．また，問合せ言語としてはパターンマッチング言語 SASE+ [3] を基礎とし，確率的な位置データストリームを扱うための空間述語を入れて拡張を行う．問合せ評価モデルの基礎として NFA^b オートマトン [4] を用いる．また，データストリームにおける問合せはリアルタイム処理に焦点が当てられることが多いが，本研究ではデータストリームを一旦蓄積して後から問合せを行う場合も想定する．

2. 関連研究

2.1 SASE+

本研究の問合せ言語の基礎となるパターンマッチング言語 SASE+ [3] について紹介する．データストリームに対するパターン問合せにおいては，入力ストリームから有限個であるがその個数は問わないデータを問い合わせたい場合がある．このような問合せは，従来の選択，結合，集約を用いた手法では表現できず，クリーネ閉包演算子 '+' を用いる手法で表現する必要がある．このような背景から，SASE+ では **クリーネ閉包** を含む問合せ表現を可能としている．また，**イベント選択戦略** (event selection strategy) という概念を導入することで，入力ストリームの選択に柔軟性を与えている．ここでパターン問合せの例として，株式市場のトレンド獲得を考える．具体的には，高い出来高の後に株価が上がり，その後出来高が急落した銘柄を問い合わせたい場合を考える．以下にこの問合せに対する SASE+ の記述例を示す．

```
PATTERN SEQ(Stock+ a[], Stock b)
WHERE skip_till_next_match(a[], b){
        [symbol]
  and   a[1].volume > 1000
  and   a[i].price > avg(a[i..1].price)
  and   b.volume < 80% * a[a.LEN].volume}
WITHIN 1 hour
```

入力ストリームはイベントストリームであり，個々のイベントは，イベントタイプ，属性値の集合，時刻をもつとする．PATTERN 項はシーケンスパターンを表しており，この場合は，クリーネ閉包の株式コンポーネント Stock+ と単一の株式コンポーネント Stock の 2 つから構成される．各コンポーネントに対しては変数が宣言され，イベントを参照するために用いる．クリーネ閉包コンポーネントに対しては配列変数が宣言される．WHERE 項では {} で囲まれた領域に述語を記述する．一つ目の記法 [symbol] は，処理されるイベントすべてが同じ銘柄である（すなわち，同じ ID に関するものである）ことを要求する．SASE+ においてはこれを等価テスト (equivalence test) と呼ぶ．2 つ目の a[1] の述語は，最初の出来高が 1000 より大きいことを要求する．3 つ目の a[i] の述語は，現在処理しているイベントの株価が今まで処理したイベントの株価の平均値より大きいことを要求する．今までに処理したイベントは a[..i-1] で記述される．a[i] の述語により，上昇トレンドの銘柄を獲得できる．なお，トレンドであるため単調増加である必要はない．最後の b の述語は，今処理しているイベントの出来高とその直前のイベントの出来高を比較しており，直前よりも 20% 以上落ちていることを要求する．a[a.LEN] は配列 a[] の最後の要素を指定する．WITHIN 項には時間窓を記述する．ここでは 1 時間としている．最後に，WHERE 項の中で関数として宣言されるイベント選択戦略について紹介する．イベント選択戦略は，**関連イベント** (relevant event) と **無関連イベント** (irrelevant event) の混ざった入力ストリームからどのように関連イベントを選択するかの戦略である．アプリケーションによって要求される戦略は異なる．

Strict contiguity 問合せパターンにマッチしたイベントシーケンスのイベントが連続 (contiguous) していることを要求するイベント選択戦略である．DNA シーケンスなどの文字列に対する正規式マッチングで一般的な要求である．

Partition contiguity この戦略は，ある条件に基づいてイベントを概念的に分割し，条件を満たすイベントを関連イベントとする．関連イベントにおいては，先の戦略のようにイベントが連続していることを要求する．つまり，入力ストリームからの関連イベントが，直前に処理された関連イベントの次の関連イベントであることを必要とする．SASE+では一般的に等価テスト（例えば，問合せ例の [symbol]）が分割条件として使われる．問合せ例の a[i] の述語は，株価の上昇トレンド（つまり，局所的な下降はあってもよい）の獲得を目的としているので，この戦略では柔軟性が足りない．

Skip till next match 問合せパターンにマッチしないイベ

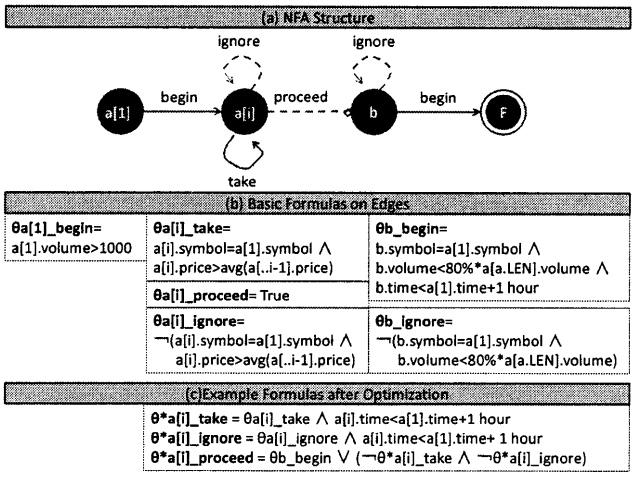

図 1　NFA^b オートマトンの例

ントをすべて無関連イベントとしてスキップする．問合せ例において戦略として用いられており，トレンド定義を満たさない値をスキップすることができる．実世界の想定として，「入力イベントの中には，ある特定のパターンへの *"semantic noise"* が存在し，パターンマッチングを続けるには無視すべきである」というものが多くある．そのような想定においてこの戦略は重要である．

Skip till any match 先の戦略をさらに緩めて，関連イベントにおいて非決定性を許す戦略である．入力ストリームからの関連イベントに対して，(1) 関連イベントとしての処理を行う．(2) 無関連イベントと同様にスキップする．という 2 つを実行する．この戦略は，本質的にはイベントサブセットにおける推移閉包を計算する．

SASE+ では，この 4 つのイベント選択戦略をアプリケーションに合わせて選ぶことが可能である．

2.2　NFA^b オートマトン

本研究の問合せ評価モデルの基礎となる NFA^b オートマトン [4] について紹介する．データストリームにおけるパターン問合せの評価は，従来の選択，結合，集約を使う手法では効率的ではないとされており，ストリームにおける効率的な評価のために提案された評価モデルがこの NFA^b オートマトンである．

NFA^b オートマトンは，非決定性有限オートマトンとマッチバッファで構成される．オートマトン A は，$A = (Q, E, \theta, q_1, F)$ で構成され，Q は状態集合，E はエッジ集合，θ はエッジをラベルする制約式 (formula)，q_1 は初期状態，F は最終状態である．マッチバッファは，問合せパターンにマッチしたイベントを格納するバッファである．NFA^b オートマトンの例として，2.1 の SASE+ 言語の記述例を変換したものを図 1 に示し，問合せ言語からのオートマトン生成手順を以下に記す．

状態 問合せ言語の `PATTERN` 項のコンポーネントから順に生成される．クリーネ閉包コンポーネントに対しては状態ペア `p[1]`, `p[i]` を生成する．クリーネ閉包でないコンポーネントに対しては *singleton* 状態を生成する．最後に，右端に最終状態を生成する．

エッジ 状態ペア `p[1]`, `p[i]` に対して，`p[1]` からは begin エッジ，`p[i]` からは take エッジと proceed エッジを延ばす．*singleton* 状態からは begin エッジを延ばす．最後に，初期状態と最終状態以外の状態から ignore エッジを延ばす．なお，take エッジと ignore エッジはループエッジである．エッジには以下の 3 つが関連付けされる．

- 制約式 θ_{q_egde}：状態遷移の制約条件
- 入力ストリームに対する動作：イベントを消費するかしないか
- マッチバッファに対する動作：イベントをマッチバッファに書き込むかどうか

エッジの制約式はパターン問合せ言語の `WHERE` 項からコンパイルされる (本稿では詳細は省略する)．図 1 のように，take エッジ，begin エッジは実線で記述する．そして，これらは入力ストリームからのイベントを消費し，そのイベントをマッチバッファに書き込む．ignore エッジは破線で記述する．これは入力ストリームからのイベントを消費し，そのイベントはマッチバッファには書き込まない．proceed エッジは特殊な ε エッジであり，入力ストリームからのイベントを消費せずに次の状態への ε 遷移を試す．ignore エッジの制約式はイベント選択戦略によって以下のように決定される．

- *Strict contiguity*：False
- *Partition contiguity*：¬ (partition condition)
- *Skip till next match*：¬ (take or begin condition)
- *Skip till any match*：True

最後に，問合せ言語の `WITHIN` 項の時間窓条件を最終状態を指すエッジの制約式に追加する．以上が問合せ言語からのオートマトン生成の素朴な手順である．エッジの制約式についてはさらなる最適化が可能である．詳細については参考文献 [4] を参照のこと．

3.　確率的データストリーム

本研究で対象とする確率的データストリームについて説明する．本稿では，自律移動型ロボットの行動分析を例として考える．各種センサ（例：レンジセンサ，ソナー）により取得されたセンシングデータはノイズを含むため，確率モデル [5] を用いてロボットの自己位置推定が行われるとする．ここでは，確率モデルとして粒子フィルタ (particle filter) を用いてデータ処理を行った結果である確率的な位置データストリームを考える．図 2 は，粒子フィルタのイメージ図である．領域はロボットに登録されている地図領域である．粒子の集合はロボットの離散的な確率分布を表しており，ロボットのある領域における存在確率は，その領域に含まれる粒子の割合によって決まる．例えば，ある領域内に粒子全体の 8 割が存在している場合，その領域にロボットが存在する確率は 0.8 である．図 2 は，ある特定の時刻のある特定のロボットについての粒子データのイメージ図であり，実際には各時刻，各ロボットに対して，それぞれ図のような粒子データが存在する．つまり，個々の位置データは，時刻，ロボット ID，粒子データをもつ．

本研究では，このような位置データが入力ストリームである

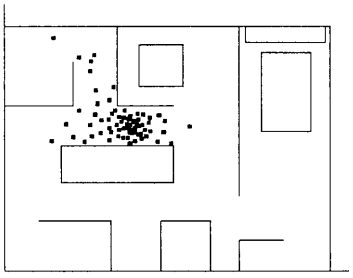

図 2　粒子フィルタによる確率的な位置データ

ことを想定し，それらの確率的データストリームに対する効率的な問合せ処理手法を開発することを目的とする．関連研究である *Markovian Streams* [1], [2] でも確率的データストリームを対象としているが，部屋や廊下といった事前に静的に区分けされた領域間の移動を考慮していた（例：「オフィス 1 から 2 へいつ移動したか確率を付与して提示せよ」）．一方，本研究では，動的に与えられる任意の領域間の移動についての問合せを行うことを目的としている．

4. 本研究について

4.1 確率的データストリームに対する SASE+ の拡張

本研究では，確率的データストリームとして確率的な位置データを対象としており，問合せの際に任意の領域を扱うことを目的としているため，以下のような空間述語を入れて SASE+ を拡張する．

- `rect(x, y, w, h)`
- `circ(x, y, r)`

`rect(x, y, w, h)` は，座標 `(x, y)` から幅 `w`，高さ `h` の矩形領域を表す．`circ(x, y, r)` は，座標 `(x, y)` から半径 `r` の円領域を表す．ここで例として，この空間述語を用いた問合せ Q を以下に示す．具体的には，領域 `rect(0,6,3,3)` から離れた領域 `rect(8,0,4,9)` に 10 分以内に移動したロボットの問合せを考える．なお，以降では，`rect(0,6,3,3)`, `rect(8,0,4,9)` をそれぞれ領域 A，領域 B と記述する．

```
PATTERN SEQ(Location a, Location b)
WHERE skip_till_next_match(a, b){
        [robot_id]
  and   overlaps(rect(0,6,3,3), a.position)
  and   overlaps(rect(8,0,4,9), b.position)
WITHIN 10 minute
```

PATTERN 項は，2 つの位置データコンポーネント Location から構成される．WHERE 項の一つ目の述語は，ロボット ID における等価テストである．2 つ目の a の述語は，粒子データと領域 A に重なりがあることを要求する．3 つ目の b の述語も同様に，粒子データと指定領域に重なりがあることを要求する．時間窓は 10 分である．なお，この問合せ Q を 2.2 で述べた NFA^b オートマトン生成手順に従って変換すると図 3 のようになる．

4.2 パターンマッチングにおける確率計算

簡単な問合せ例を示して，確率的データストリームのパターンマッチングにおける確率計算について説明する．ロボット R が給湯室にいたイベントを K とする（給湯室にいなかったイベントは ¬ K とする）．問合せ例 Q_K を「時刻 $t=1$ から $t=3$ においてイベント K が発生した」とする．時刻 $t=1$ から $t=3$ において，イベント K が発生したか否かの確率が図 4 の (a) のようであったとする．このとき，時刻 $t=1$ から $t=3$ までのシーケンスのパターンは図 4 の (b) の 8 通りある．

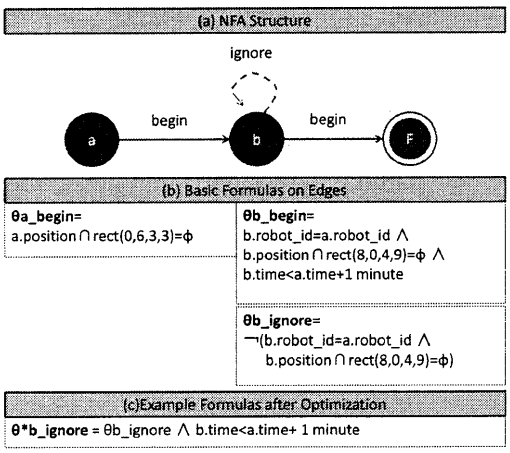

図 3　問合せ Q に対する NFA^b オートマトン

t	P(K)	P(¬K)
1	0.30	0.70
2	0.60	0.40
3	0.70	0.30

(a) Probability about event K

t=1	t=2	t=3	Pr
¬K	¬K	¬K	0.084
¬K	¬K	K	0.196
¬K	K	¬K	0.126
¬K	K	K	0.294
K	¬K	¬K	0.036
K	¬K	K	0.084
K	K	¬K	0.054
K	K	K	0.126

(b) Sequence patterns from t = 1 to t = 3

図 4　イベント K の確率とシーケンスパターン

図 4 の (b) のシーケンスパターンのうち，問合せ Q_K を満たすパターンは 2 行目から 8 行目までの 7 通りである．よって，問合せ Q_K が満たされる確率は，該当する 7 つのシーケンスの確率の和をとって 0.916 となる．このように，確率的データストリームにおいては，問合せパターンに複数のシーケンスがマッチした場合，マッチした確率の総和を計算して問合せ結果の確率とする．

次に，イベントが時間順に流れてくる実際のストリーム処理環境における確率計算についてに述べる．まず，時刻 $t=1$ においてイベント K が発生したシーケンスと発生していないシーケンスができる．前者のシーケンスは問合せパターンを満たすので，そのシーケンスの確率 0.30 を問合せパターンの確率に加える．以降ではそのシーケンスの追跡は不要である．次に，時刻 $t=2$ において，先の後者のシーケンスから時刻 $t=1$ のときと同様に 2 つのシーケンスができる．時刻 $t=1$ のときと同様に，

問合せパターンを満たすシーケンスの確率 $0.70 * 0.60 = 0.42$ を問合せパターンの確率に加える．時刻 $t = 3$ についても同様にして確率 $0.70 * 0.40 * 0.70 = 0.196$ を問合せパターンの確率に加える．結果，問合せパターンの確率 0.916 が得られる．上記の確率計算の流れを表で示したものを図 5 に示す．

t = 1	t = 2	t = 3	Pr
¬K	¬K	¬K	0.084
		K	0.196
	K		0.42
K			0.3

0.916

図 5　ストリーム環境下における確率計算

4.3　確率的データストリームにおける ignore エッジの処理

[4] においては，ignore エッジを遷移したイベントは無関連イベント（すなわち，*semantic noise*）とされ，マッチバッファに書き込まれずシーケンスに影響を与えない．しかし，確率的データストリームにおいては [4] とは異なり，ignore エッジを遷移したイベントが *semantic noise* とされないない場合がある．本章ではその点について述べる．

本来，入力ストリームからのイベントはタイムスタンプが同じであっても同時に処理されることはない．しかし，本研究においては，同時に複数のイベントが処理されることがある．これは，入力ストリームのイベントと処理されるイベントが異なることによる．具体的には，本稿で想定している入力ストリームのイベントは時刻，ロボット ID，粒子データをもつ位置データであり，処理されるイベントはその位置データが問合せ述語を満たすか否かで分割されたイベントだからである．このことから，あるイベントが ignore エッジを遷移した場合でも，同時に処理されたイベントがあり，それが begin エッジや take エッジを遷移した場合は，それは *semantic noise* でない．なぜなら，元々それらを合わせた 1 つのイベントが入力ストリームからのイベントであり，そのイベントが問合せパターンのセマンティクスに関連しているからである．よって，ignore エッジを遷移したイベントに対しては，同時に処理されたイベントがあり，それが begin エッジや take エッジを遷移した場合，それをマッチバッファに書き込む．そうでない場合は，*semantic noise* としてマッチバッファに書き込まない．

4.4　問合せ処理

データストリームが問合せシーケンスにマッチするためには，各状態において次状態への遷移が満たされる必要がある．つまり，問合せ処理においては，各状態における次状態への遷移エッジの制約式が重要である．リアルタイム処理においては，[4] の処理を基礎とする．本研究では，確率的データストリームを対象としているため，加えてシーケンスの確率計算を行う．なお，ignore エッジの処理については 4.3 で述べた通りに行う．

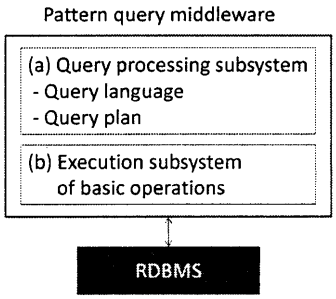

図 6　本研究の位置付け

次に，データストリームを一旦蓄積した履歴データの問合せ処理に焦点を当てる．リアルタイム処理では初期状態から順に処理を行う必要があるが，履歴データの処理においては順に行う必要はなく，絞り込みの効く部分から先に処理を進めることで処理効率を上げることができると考えられる．よって本研究では，次状態への遷移条件に着目して絞り込みの効く部分から処理を行うことで，効率的な問合せ処理を実現することを目的とする．以降でその手法について述べる．

問合せ処理においては，まず NFA^b オートマトンの各状態から proceed エッジが延びているか注目する．proceed エッジのない *singleton* 状態や，状態ペア `p[1]`，`p[i]` の内の状態 `p[1]` は，begin エッジの制約式が満たされれば次状態に遷移が可能である．一方で，proceed エッジがある状態 `p[i]` では，proceed エッジの制約式が満たされれば次状態への ε 遷移を試すが，ε 遷移であるため次状態のエッジの制約式が満たされなければ遷移は発生しない．つまり，proceed エッジのある状態 `p[i]` は，次状態への遷移を次状態のエッジ制約式に依存している．これについて本研究では，proceed エッジのある状態 `p[i]` は，問合せシーケンスにマッチするデータの探索において重要度が低いと考える．そこで，本研究の問合せ処理においては，まず，問合せシーケンスから状態 `p[i]` を取り除き，その部分を区切りとして問合せシーケンスをいくつかのサブシーケンスに分割する．次に，各サブシーケンスの処理の順番を決め，最後に，取り除かれていた状態 `p[i]` を処理キューに加えて処理を実行する．各サブシーケンスの処理順は次のように決定する．begin エッジの制約式の選択性はシーケンスにマッチするデータの絞り込み度の見積りに有効であると考えられるので，分割されたサブシーケンスの begin エッジの制約式を評価する．その評価に基づいて，各サブシーケンスの絞り込み度を見積ってそれらの処理の順番を決定する．

ここまでは問合せシーケンスの処理順の決定について述べたが，ここからは処理について述べる．本研究の位置付けを図 6 の (a) に示す．本研究では図 6 にあるように，RDBMS が下位に存在することを想定しており，RDBMS のその処理能力を積極的に活用することを考えている．なお，図 6 の (b) は，早矢仕ら [6] の研究であり，統合したフレームワークの開発を考えている．最近の RDBMS には，SQL の再帰問合せ機能を提供しているものが存在しており，従来の SQL では記述できなかったクリーネ閉包の記述が可能となっている．よって，NFA^b

オートマトンを SQL 問合せに変換することは可能だと考えている．処理としては，各サブシーケンスを SQL 問合せに展開して，その多くを RDBMS の処理に落とし込むことを考えている．

5. まとめと今後の課題

本稿では，確率的データストリームにおける問合せ処理手法について述べた．本手法は，SASE+言語に空間述語を入れて拡張し，確率的な位置データストリームに対する問合せを可能としている．また，NFA^b オートマトンが確率的データストリームを想定していないことにより発生する問題点に対して，解決のための処理手法の提案を行った．本稿では，リアルタイムデータだけでなく蓄積されたデータに対しても考察し，効率的であると考えられる処理手法の提案を行った．

今後の課題としては，提案の処理手法について実装と評価が必要である．

謝　　辞

本研究の一部は，内閣府最先端研究開発プロジェクト (FIRST) による．

文　　献

[1] J. Letchner, C. Ré, M. Balazinska, and M. Philipose. Access methods for Markovian streams. In *Proc. ICDE 2009*, pp. 246–257, 2009.

[2] C. Ré, J. Letchner, M. Balazinksa, and D. Suciu. Event queries on correlated probabilistic streams. In *Proc. ACM SIGMOD*, pp. 715–728, 2008.

[3] D. Gyllstrom, J. Agrawal, et al. On supporting kleene closure over event streams. In *Proc. ICDE 2008*, Poster, 2008.

[4] J. Agrawal, Y. Diao, D. Gyllstrom, N. Immerman. Efficient pattern matching over event streams. In *Proc. ACM SIGMOD*, pp. 147–159, 2008.

[5] S. Thrun, W. Burgard, and D. Fox. 確率ロボティクス．毎日コミュニケーションズ, 2007.

[6] 早矢仕，董，加藤，石川．移動ロボットのための確率的空間問合せシステムの構築．情報処理学会第 74 回全国大会，1T-8, 2012.