知 能 と 複 雑 系 128－29
（2002. 5. 24）

交通データウェアハウスにおけるOLAP

河 野 浩 之†

携帯端末やGPSから得られた位置データを，交通管制や都市計画に活用するために，多量の時空間データ処理に対するオンライン解析処理 (OLAP) が必要とされつつある．そこで，本研究では，交通状況を蓄積する交通データウェアハウスにおいて，車両数・平均速度・空間占有率などの交通管制で用いる各種交通パラメータを高速に求めるうえで有効な時空間インデックスについて考察する．そして，空間的制約を考慮したノードを生成し，各ノードが含む各種交通データの前処理演算を施す$\sum$-treeを提案する．なお，阪神高速道路において定点観測した実測値に基づくシミュレーションモデルにより生成されたデータを用いた，典型的な解析処理における計算コストの低減が$\sum$-treeによって実現されていることに基づいて，交通データウェアハウスのOLAPにおける有効性を述べる．

OLAP in Traffic Data Warehouse

HIROYUKI KAWANO†

Technologies of location service and GPS are becoming popular, and huge volume of spatial and geographic data are stored into the GIS and database systems. For traffic planning, marketing and so on, advanced spatial queries and OLAP (On-Line Analytical Processing) technologies are required, we are trying to analyze the actual data in the traffic data warehouse. In this paper, in order to execute temporal spatial queries, we store traffic data using TPR-tree and 3DR-tree, which are the data structures to analyze traffic parameters fast. We also proposed $\sum$-tree, which decreases computing costs by making nodes with spatial constraints and using prefix-sum techniques. We discuss the performance of query execution time based on the traffic model of Hanshin expressway. As the result, we make clear that $\sum$-tree is effective for OLAP in our traffic data warehouse.

1. はじめに

現在，交通管制，交通計画，都市計画などの高度化を目指して，道路網上に配備された各種情報センサーから取得した様々なデータを蓄積する交通データウェアハウス (traffic data warehouse) が整備されつつある[3]．また，PHSやGPS機能を備えた移動体端末などの普及により，長期間にわたる位置データの大規模な蓄積も可能になってきている．そこで，我々は，地理情報システム (GIS: Geographic Information System) のアプリケーション構築に必要なデータ処理技術を検討してきた[1),4)~6),8),13)]．

そこで，本稿では，長期かつ連続的に交通データを格納する交通データウェアハウスを利用した交通流解析に注目し，交通工学の研究動向を踏まえながら，その利用に必要なデータ処理技術を検討する．特に，管制や計画などの意思決定支援に有効な実時間解析技術としてのOLAP (On-Line Analytical Processing) に焦点を当てることにする[7]．そして，交通データウェアハウスにおける各種交通パラメータに対するOLAPを実現するデータ構造として$\sum$-treeを用いる．

以下，2章では交通データ処理に必要な，時空間インデックスであるTPR-treeと3DR-tree[12]を紹介する．また，各種交通パラメータを求めるための高速な領域和演算の前処理演算として，データキューブ (Data Cube)[2] の考え方を導入する．次に，3章において，空間的制約を考慮したデータ構造として$\sum$-treeを提案する．続く4章では，時間平均速度や空間平均速度など，交通計画の基礎データとなる典型的な交通パラメータ処理について触れる．加えて，提案手法の簡単な評価に基づきながら，この種の解析的問合せ処理の必要性についても触れる．そして，最後の5章が，結論と今後の課題となる．

2. 交通管制システムとデータ処理

本章では，既存の交通管制システムにおけるデータ処理の高度化について考える．次に，先進的な情報収

† 京都大学大学院情報学研究科
Graduate School of Informatics, Kyoto University

集システムから得られる大量の交通データを処理するための基礎技術として，TPR-tree や 3DR-tree などの時空間インデックスとデータキューブを用いた領域和処理について述べる．

2.1 交通管制データの高度化

これまでの交通計画では，個人の交通行動の部分的な調査データを基に，都市計画や道路計画，地域施設の配置計画，さらには，最短経路問題等，様々な交通システム設計や運用に伴う研究がなされてきた[10]．例えば，現在，路側に設置した車両検知器によってモニタリングした交通状況のデータが，交通制御や交通情報提供のベースとなっている．

もっとも，現在の交通管制システムでは，センサーの精度やデータ処理速度の制約から，処理単位時間を5分とした5分間交通量に基づいた実装を行っている．加えて，その大部分の処理データは，短期間に磁気テープへと格納されるため，交通計画を行う上で必要な解析が実現できているとも言い難い．さらに，既存の超音波センサーを用いた定点型モニタリングデータでは，車両検知器が設置された道路区間においてのみデータ収集が行われることは勿論，交通流の断面的な観測による解析となっている問題も指摘されている．

他方，交通管制データの解析作業では，多数の属性を考慮した分析も必要である．例えば，季節変動，時間帯，地域的特性，道路の特性，車種などの車両の特性，性別・年齢などのドライバーの特性が考えられ，過去の交通状況を検索するに十分な属性値を要する．したがって，多様な視点から解析するには，時空間データと多次元データの両面から考察する必要がある．

よって，都市空間に埋め込まれた各種センサーにより観測されたデータや，移動体端末から取得される各種データを効果的に蓄積し，そして，時空間データに対するリアルタイム解析を行う処理技術の必要性が高まっている．

2.2 時空間問合せと空間インデックス

本研究が目的とするシステムには，移動オブジェクトの位置データや地図データを蓄積する際に用いる空間インデックス技術，及び，効率的な位置問合せ処理を行う技術が深く関わる．

例えば，空間インデックスとして，R-tree[11] などが提案されている．しかしながら，一般に良く用いられている空間インデックスである R-tree によって，時々刻々と変化する多数の移動オブジェクトに対する位置管理を行う場合，位置情報の更新により生ずる負荷が問題となるため，何らかの解決が必要である．さらに，典型的な移動オブジェクトに関する問合せを，交通工学の視点から解釈する必要がある．

2.2.1 時空間インデックス:TPR-tree

TPR-tree(Time Parameterized R-tree) は R-tree 構造に基づくが，各ノードに対応する d 次元長方形の位置や大きさは，時間に関する関数により表現される．例えば，TPR-tree では，移動オブジェクトの現在の位置と速度を用いて，将来の位置を線形関数で近似することが行われる．すなわち，時刻 $t = t_{ref}$ において，位置 $\mathbf{x}_{ref}$，速度 $\mathbf{v}$ である移動オブジェクトの時刻 $t(t > t_{ref})$ における位置 $\mathbf{x}(t)$ が，(1) 式で表される．

$$\mathbf{x}(t) = \mathbf{x}_{ref} + \mathbf{v}(t - t_{ref}) \tag{1}$$

また，空間インデックスを生成する際には，現在の移動オブジェクトの状態だけではなく，未来においても位置が近いと考えられる移動オブジェクトを同じグループとして索引付けることが行われる．つまり，位置 $\mathbf{x}$，速度 $\mathbf{v}$ に対し，$(\mathbf{x}, \mathbf{v})$ 空間内での距離が近いものを，同じノード内へと格納する．このようなデータ構造によって，境界長方形が時間の経過と共に拡大することが抑制され，効率的な時空間インデックス構築へとつながる．

さらに，$(\mathbf{x}, \mathbf{v})$ 空間における距離定義を変える事により，位置データの更新頻度や，対象オブジェクトの速度分布などの特性変化に対応することも可能である．例えば，データの更新周期が長い時や，移動オブジェクトの速度分布の分散が大きい時は，$\mathbf{x}$ よりも $\mathbf{v}$ における距離の影響を大きくし，より遠い将来の位置を重要視するインデックスを生成する等である．

2.2.2 位置情報予測に関わる問合せ処理

交通工学の視点から，移動オブジェクトに関する以下の3種類の空間問合せを考える．なお，t_1，$t_2(t_1 < t_2)$ を時刻，R_1，R_2 を地図上の領域（長方形）とする．

(1) **定時刻問合せ** $Q_{ts} = (R_1, t_1)$: 図1に示したように，ある時刻 t において，領域 R_1 内に存在するオブジェクトを出力する．
(例) 出力されたオブジェクトの集合から交通密度や空間平均速度といった，ある瞬間の交通状況を表現する空間交通流パラメータを算定するための問合せである．

(2) **時間帯問合せ** $Q_{win} = (R_1, t_1, t_2)$: 図2に示したように，ある時間帯 t_1 から t_2 の間に領域 R_1 内に存在するオブジェクトを出力する．
(例) 空間平均速度に比べて安定的な時間平均速度などの時間交通流パラメータを算定するための問合せである．

(3) **等速移動領域問合せ** $Q_{mov} = (R_1, R_2, t_1, t_2)$:

図3に示したように，位置ベクトル $\mathbf{x}$ と時刻 t の空間 $(\mathbf{x}, t)$ 内で，(R_1, t_1) と (R_2, t_2) とを結ぶ台形内に存在するオブジェクトを出力する．(例) 進行方向と経過時間を考慮した時空間領域の交通密度から混雑状況を予測するための問合せである．

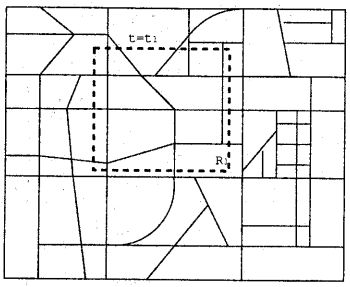

図 1 定時刻問合せの一例

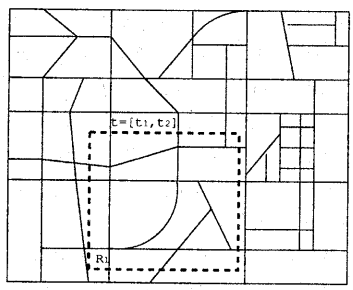

図 2 時間帯問合せの一例

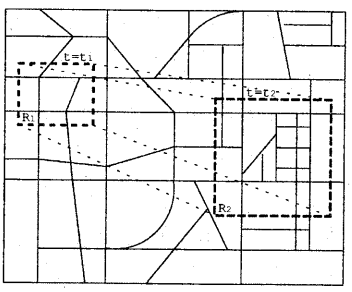

図 3 等速移動領域問合せの一例

2.2.3 時空間インデックス:3DR-tree

交通データを検索するには，平面座標系 (x, y) と時間軸 t による時空間インデックスを利用し，その検索処理速度の高速化が必要である．すなわち，オブジェクトの動きを解析処理することを考えると，3DR-tree の利用も有効であると考えられる [12]．

しかし，図4に表すような時刻 t における定時刻問い合せや，図5に表す短時間区間の問合せを処理する

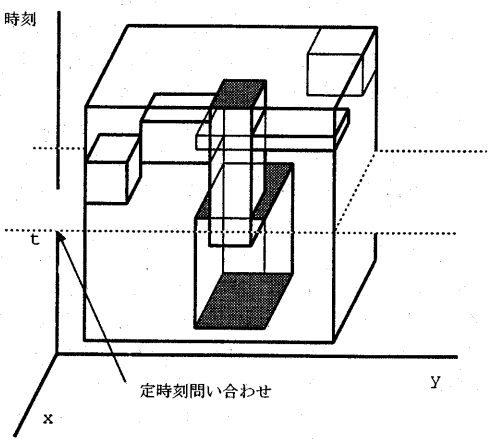

図 4 3DR-tree の例（定時刻問合せ）

場合，必要以上に大きな領域を被覆する親ノードの操作が生じるため，効率的ではないデータアクセスが生じる可能性に注意しなければならない．

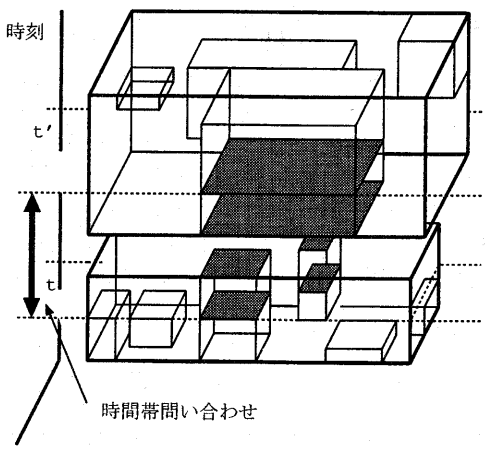

図 5 3DR-tree の例（時間帯問合せ）

例えば，図4に示す定時刻問合せに対する問合せ領域は，網掛けされたノードとなる．ここで，t 軸方向に大きく伸びたノードが含まれる場合，親ノードの覆う領域も大きくなり，アクセスノード数が増加する．これは，図5のような時間軸に幅をもつ時間帯問合せに対しても同様である．

そこで，一定時間間隔にノードを区切ることで，ノードが t 軸上に長くならない処理が必要と考えられる．よって，次章では，交通データ特有の時空間制約を考慮したノード分割を提案する．

2.3 データキューブによる領域和演算

交通データウェアハウスへの格納は追記型であり，そのデータ量が増大する一方である点が，通常のデータベース利用と異なる．また，交通計画を策定する上で，アドホックかつダイナミックな問合せをオンラインで処理する必要があり，OLAP 技術を要する．さらに，交通流の傾向，類似性，関係性など，データマイニングに関わる処理も将来必要となると思われる．

よって，これらの処理コストの大きいデータ操作を行いながら，各種交通流パラメータを求めるには，各ノード内に格納された多量のデータに対する領域和演算が必要となる．そこで，データ処理コストの削減を目標とした，データキューブなどの技術を応用することで，交通データウェアハウスに対する OLAP 技術の実現について考える．

ここで，領域和を処理する場合に用いる，元のデータ構造のセルに関する和を格納したデータキューブの性質について簡単に触れることにする [2]．まず，$D = 1, 2, \cdots, d$ をデータ構造の次元とし，n_i を次元 i の中に含まれるセル数とすると，元のデータで和を求めるには $N = \prod_{i=1}^{d} n_i$ 回のアクセスを必要とする．他方，d 次元データキューブを構成すると 2^d 個の和は $2^d - 1$ 回のアクセスで求めることができる．このように，領域和を，一定のデータアクセス回数で求まることがデータキューブの利点である．

なお，形式的にデータキューブの構成方法を示すと，$0 \leq x_i < n_i$ かつ $i \in D$ の条件で，データキューブ A は，元のデータ B を用いて，次式で表すことができる．

$$\begin{aligned} A[x_1, x_2, \cdots, x_d] &= Sum(0 : x_1, 0 : x_2, \cdots, 0 : x_d) \\ &= \sum_{i_1=0}^{x_1} \sum_{i_2=0}^{x_2} \cdots \sum_{i_d=0}^{x_d} B[i_1, i_2, \cdots, i_d] \end{aligned}$$

また，データキューブに格納される 2^d 個のパラメータの領域和は次のようになる．

$$Sum(l_1 : h_1, l_2 : h_2, \cdots, l_d : h_d) = \sum_{\forall x_j \in l_j - 1, h_j} \left[\left\{ \prod_{i=1}^{d} s(j) \right\} * A[x_1, x_2, \cdots, x_d] \right]$$

ただし，

$$\forall j \in D, s(j) = \begin{cases} 1 & (x_j = h_j) \\ -1 & (x_j = l_j - 1) \end{cases}$$

3. $\sum$-tree の提案 [6]

本章では，時空間データの特性を踏まえながら，各ノードとその下位のノードに含まれるオブジェクト数，速度の和などを効率的に処理するデータ構造である $\sum$-tree について述べる．

3.1 $\sum$-tree のデータ構造

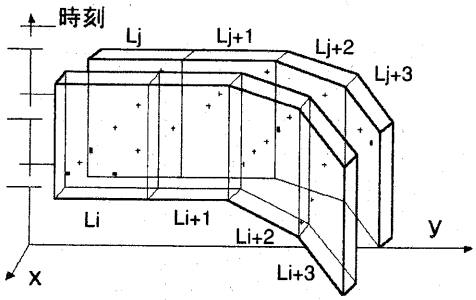

図 6　道路セグメントに基づく時空間分割

時空間データ (x, y, t) を格納するデータ構造として $\sum$-tree を定義する．まず，全オブジェクト数を n とし，葉ノードは時空間データ $(x_1, y_1, t_1), (x_2, y_2, t_2), \cdots, (x_n, y_n, t_n)$ を含む．ここで，ノード集合からなるツリー構造を作るが，オブジェクト距離の近いものでノードを構成するのではなく，オブジェクトに対する空間的制約を考慮する．

また，交通データに対する空間的制約は，オブジェクトの存在する道路車線となるので，車線内のオブジェクト距離の近いものにより上位ノードを構成する．よって，ノード $L_1, L_2, \cdots, L_N$ を構成すると，次のようなデータ格納が行われることになる．

$$\begin{cases} (x_1, y_1, t_1), (x_2, y_2, t_2), \cdots, (x_i, y_i, t_i) \subset L_1 \\ (x_{i+1}, y_{i+1}, t_{i+1}), \cdots, (x_j, y_j, t_j) \subset L_2 \\ \vdots \\ (x_{k+1}, y_{k+1}, t_{k+1}), \cdots, (x_n, y_n, t_n) \subset L_N \end{cases}$$

なお，図 6 中の点が，オブジェクトデータに対応する．

次に，$\sum$-tree に格納される属性値であるが，移動体データにおいて，(オブジェクト ID，時刻，(x, y) 座標) が，基礎属性となる．もっとも，交通流を解析するには，オブジェクトの移動速度属性なども必要となる．そこで，オブジェクト間の距離算出を行うため，t 時点から 1 観測時点前における t' 時点のデータを用いて，速度 $v_t = \frac{\sqrt{(x_t - x_{t'})^2 + (y_t - y_{t'})^2}}{t - t'}$ を与える．そして，オブジェクトの移動経路格納時に，オブジェクトが属する道路セグメントと照合しながら，$\sum$-tree のノードに格納する．このようにして，各種交通パラメータの和を各ノードに格納する．

すなわち，各オブジェクトの速度に着目すると，速

度を含めたデータ $(x_1, y_1, t_1, v_1), (x_2, y_2, t_2, v_2), \cdots, (x_n, y_n, t_n, v_n)$ が葉ノードに格納され，次式で表される速度和 $V_{L_1}, V_{L_2}, \cdots, V_{L_N}$ が上位ノードに格納される．

$$V_{L_1} = \sum_{l=1}^{i} v_l,\ V_{L_2} = \sum_{l=i+1}^{j} v_l,\ \cdots,V_{L_N} = \sum_{l=k+1}^{n} v_l$$

さらに，各ノードには，様々な交通データを演算するために，例えば，オブジェクト数 n_{ij}，オブジェクト速度和 V_{ij} が格納される．ここで，親ノードが含む子ノード数を k 個とし，$\sum$-tree に次式の値を格納する．

$$n_{(i-1)j} = \sum_{l=1+(j-1)k}^{jk} n_{il}$$

$$V_{(i-1)j} = \sum_{l=1+(j-1)k}^{jk} V_{il}$$

ところで，各ノードは立体で示されており，頂点を2つ定めることで，その立体を表すことができる．よって，立体を表す2点の座標を始点 $(x_{ijs}, y_{ijs}, t_{ijs})$，終点 $(x_{ije}, y_{ije}, t_{ije})$ とし．ノード空間を $(< x_{ijs} : y_{ijs} : t_{ijs}, x_{ije} : y_{ije} : t_{ije} >)$ で表す．子ノードの数を k 個すると，親ノードとの関係は次式で表される．

$$\begin{cases} x_{ijs} = min[x_{i(1+(j-1)k)s}, x_{i(2+(j-1)k)s}, \cdots, x_{i(jk)s}] \\ x_{ije} = max[x_{i(1+(j-1)k)e}, x_{i(2+(j-1)k)e}, \cdots, x_{i(jk)e}] \\ y_{ijs} = min[y_{i(1+(j-1)k)s}, y_{i(2+(j-1)k)s}, \cdots, y_{i(jk)s}] \\ y_{ije} = max[y_{i(1+(j-1)k)e}, y_{i(2+(j-1)k)e}, \cdots, y_{i(jk)e}] \\ t_{ijs} = min[t_{i(1+(j-1)k)s}, t_{i(2+(j-1)k)s}, \cdots, t_{i(jk)s}] \\ t_{ije} = max[t_{i(1+(j-1)k)e}, t_{i(2+(j-1)k)e}, \cdots, t_{i(jk)e}] \end{cases}$$

すなわち，図7に示すように，上位ノード L_u に，ノードの座標領域 $(x, y, t) \subset L_u$，各種交通データの和（速度の和 V_{L_u}，通過車両数 N_{L_u}）を格納する．また，上位の親ノードには，ノード領域に属する子ノードの各属性値の和を格納する．

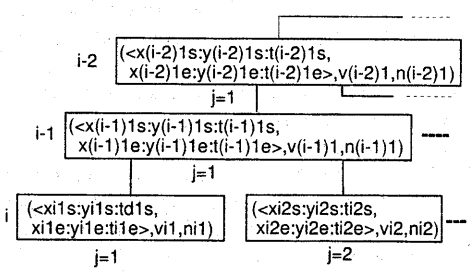

図7 $\sum$-tree のノードに対する格納値

以上のようなデータ格納を行うことで，ノード内の全オブジェクトの交通データの和は，ノードへの一回のアクセスで求めることができる．なお，同じ時空間領域に，時空間特性以外の属性値の異なる複数の $\sum$-tree を構成することを考えても良い．

もっとも，問合せコストと，データの前処理コストとのトレードオフの関係が生じることは，データキューブの作成コストの議論と同様であり[2]，今後，様々な解析処理の特性を考慮した評価が必要となる．

4. 交通パラメータと解析処理

これまでの断面交通量は，道路上の地点でしか観測できなかったが，リアルタイムでオブジェクト位置を把握できる情報収集システムを用いることで，地点ではなく，区間での交通量を把握できる．そして，$\sum$-tree を用いることで，既存のシステムで用いられている5分間交通量ではなく，交通計画に関わる各種パラメータの挙動解析がリアルタイムに実行可能になると考えられる．

つまり，現在の交通管制は，定点観測による地点速度の平均値を算出し，観測区間外で区間速度一定とみなしているため，精度の高いモデルでは無い可能性がある．よって，この種の交通データウェアハウスによる実データ解析により，より優れたモデル構築が可能になると考えられる．

加えて，この種のパラメータ処理を円滑に求めることで，渋滞時などの地点ごとに交通流が大きく異なる状況においても，より詳細なデータ収集に基づいて，時間変化により追随した渋滞判定や予測が可能になる．

このように，前章で提案した $\sum$-tree を用いることで，車両数，断面交通量，空間平均速度，交通密度の4種の交通パラメータを OLAP 技術を用いて正確に求めることは非常に重要な役割を果たす．

例えば，現在の交通管制は，定点観測により地点速度の平均値を算出し，観測区間外で区間速度一定とみなしている．しかしながら，速度には地点速度と区間速度の二つの定義があり，定点を通過する車両の瞬間的な速度が地点速度，定区間を走行するのに要した時間で区間長を除した値が区間速度である．すなわち，空間平均速度はある区間での通過したオブジェクトの区間速度の平均値を用いることで，より精度の予測ができると考えられる．なお，図8は，道路セグメントのある区間でのオブジェクトの移動と通過した距離，通過に要した時間を示しており，空間平均速度はオブジェクト数を n とすると，$\frac{1}{n}\sum_{k=1}^{n}\frac{a_k}{t_k}$ となる．

また，我々は，空間データ構造を対象とした OLAP 操作に対する研究[7]と同様，データの空間的分布に着

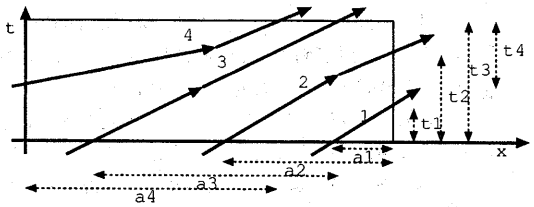

図 8 空間平均速度

目しながら性能評価を行っている．そして，文献[6]では，阪神高速道路守口線において実測された定点観測データを基に，追従モデルによるシミュレータ (MIT-SIMLab)[9] を用いて生成された交通流データに基づいた $\sum$-tree の解析処理効率を 3DR-tree による処理性能と比較し，$\sum$-tree のデータアクセスの有効性を確かめている．

5. むすび

本稿では，現在，整備されつつある交通データウェアハウスを用いて，交通計画で用いる解析処理を，どのように高度化できるかについて時空間インデックス技術と OLAP 技術を中心に検討を重ねた．そして，提案した $\sum$-tree を用いることで，各種交通パラメータを効率良く求めることができることを述べ，これらの問合せ処理を効率よく評価することで，より精度の高い交通管制・計画に役立つことを述べた．

なお，空間と時間軸に対して，どのように $\sum$-tree を構成するかによって，そのシステム性能が大きく変化すると予測される．よって，どのようなノード分割を備えた木構造が，優れた性質をもつかをの検討を今後進める必要がある．加えて，将来，一定時間で平滑化した断面交通量による現行の交通管制システムにおける交通モデルが，この種の実時間解析技術によって，どのように発展できるかは非常に興味深い点である．

謝　辞

本研究の一部は，科学研究費の成果を含む．また，マツダ研究助成金「データマイニング技術を用いたトリップデータ処理に関する研究」による支援を受けている．なお，文献[6]の共著者である，田名部淳氏（都市交通計画研究所），ならびに，岸浩史君（京都大学大学院情報学研究科）らにも感謝の意を表す．

参 考 文 献

1) 朝倉 康夫, 羽藤 英二, 大藤 武彦, 田名部 淳, “PHS による位置情報を用いた交通行動調査手法”, 土木学会論文集, No.653/IV-48, pp.95–104, 7, 2000.
2) Chun, S., Chung, C., Lee, J. Lee. S., “Dynamic Update Cube for Range-Sum Queries,” Proc. of VLDB 2001, pp.521-530, Sep. 2001.
3) (財) 阪神高速道路管理技術センター, “阪神高速道路交通管制施設に関する調査検討業務報告書,” 阪神高速道路交通管制施設に関する調査検討委員会, 2001.
4) 河野 浩之, “位置情報活用に関する基礎的考察～経路データ活用とインデックス～,” 情報処理学会研究報告, 2000-DBS-122, pp.291–298, 2000.
5) Kawano, H., “Architecture of Trip Database Systems: Spatial Index and Route Estimation Algorithm,” XIV International Conf. of Systems Science, Vol. III, pp.110-117, Poland, 2001.
6) 岸浩史，田名部淳，河野浩之, “$\sum$-Tree による時空間 OLAP 技術の交通データへの適用,” 第 12 回電子情報通信学会データ工学ワークショップ, DEWS2002, 2002.
7) Miksovsky, P., and Kouba, Z., “GOLAP-Geographical Online Analytial Processing,” Proc. of 12th Int. Conf. on Database and Expert Systems Applications (DEXA01), pp.442-449, 2001.
8) 南卓朗, 田名部淳, 河野浩之, “空間インデックスを用いた移動オブジェクト管理システムの構成と性能評価,” 情処研報 Vol.2001, No.70, DBS-125, pp.225-232, 2001.
9) MIT Intelligent Transportation Systems Program, “User’s Guide for MITSIMlab and Road Network Editor (RNE),” Aug. 2001. (http://web.mit.edu/its/products.html)
10) 日本建築学会編, “建築・都市計画のためのモデル分析の手法,” pp.71–84, pp.122–162, 井上書院, 1992.
11) Samet, H., “Spatial Data structures,” Modern Database Systems, (W. Kim, ed.), ACM Press, New York, pp. 361–385, 1995.
12) Tao, Y. and Papadias, D., “The MV3R-Tree: A Spatio-Temporal Access Method for Timestamp and Interval Queries,” Proc. of VLDB 2001, pp.431-440, Sep. 2001.
13) 都市交通計画研究所, 地域未来研究所, “移動体位置情報に基づく動的時空間データ分析技術の開発（技術開発報告書）,” 平成 10 年度新規産業創造技術開発費補助事業, 2000.