

# 道路の接続性を保障したテレマティクスサービスのための 地図差分更新方式

浅原 彰 規<sup>†1</sup> 谷 崎 正 明<sup>†1</sup>  
嶋 田 茂<sup>†1</sup> 森 岡 道 雄<sup>†2</sup>

ハードディスク搭載カーナビゲーションシステムおよびテレマティクスサービスの普及にともない、携帯電話を介した地図情報更新のニーズが急速に高まっており、これに対応するため、変更差分のみ配信する地図差分更新方式が検討されている。ところが、全国分の地図を対象とすると差分データサイズであってもなお大きく、実サービスの運用は困難であった。そこで我々は、配信データ量の削減のため、部分的に地図を更新するスポット更新サービスを導入する。そしてこのサービスを実現するため、道路の接続性を保持した部分的地図更新を可能とする接続保証方式を提案する。本方式を実地図データを用い評価したところ、20 km 四方の地図が 1 分以内で更新可能であることが判明した。これにより本方式を用いた地図配信サービスが実現可能な見込みを得た。

## Differential Map-update Method Ensuring Road-network Connection for Telematics Service Systems

AKINORI ASAHARA,<sup>†1</sup> MASAOKI TANIZAKI,<sup>†1</sup> SHIGERU SHIMADA<sup>†1</sup>  
and MICHIO MORIOKA<sup>†2</sup>

Needs for updating maps of a car-navigation system through a cellular-phone are increasing with popularization of car-navigation systems loading Hard Disk Drive and telematics services. Then, differential-update methods to distribute only differences of maps are suggested. Although, even if we use with these methods, distribution size to update a map of all Japanese area is too much to start actual service. Then, we introduce Spot-Update Service which is a partially updating service of maps to reduce distribution size. For this service, we propose a method for partially updating of maps with keeping connections between roads. In our experiment, a map of almost 20 km<sup>2</sup> area were updated in less than 1 min with our method. Thus, updating services through cellular-phones became possible by our method.

### 1. はじめに

HDD を搭載したカーナビゲーションシステム (HDD カーナビ) の普及とともに、携帯電話などの通信を用い最新の交通情報などを取得できるテレマティクスサービスの利用者もまた急速に増加している。実際、2006 年 8 月にはホンダのテレマティクスサービス「インターナビ<sup>12)</sup>」の利用者数が 40 万人、2006 年 6 月末時点の各自動車メーカーのテレマティクスサービス利用者数は、トヨタ自動車の「G-Book Alpha<sup>6)</sup>」が

32 万人、日産自動車「カーウィングス<sup>11)</sup>」が 23 万人と、計 100 万人規模の市場へと成長しており、さらなる発展が見込まれる。

これらテレマティクスサービスの中心的機能となるのが、渋滞情報や店舗情報など、運転者の行動支援のための最新地図情報の提供サービスである。なかでも、地図データの更新サービスに対する利用者のニーズの高さが市場調査<sup>7)</sup>により明らかになっている。このように、いつでもどこでも必要なときにカーナビの地図を更新したい、というニーズは非常に高い。しかし、現在実用化されているサービスは十分このニーズに答えていない。実際、前述のホンダ「インターナビ」の地図更新サービスは単にカーナビの HDD をディーラに持ち込み書き換えるサービスであり、地図がいつでも更新できるサービスではない。また、パイオニア「Air ナビ<sup>3)</sup>」のようにカーナビ側は地図データを持た

†1 株式会社日立製作所中央研究所  
Central Research Laboratory, Hitachi, Ltd.

†2 株式会社日立製作所オートモティブシステムグループ CIS 事業部  
Car Information Systems Division, Automotive Systems, Hitachi, Ltd.

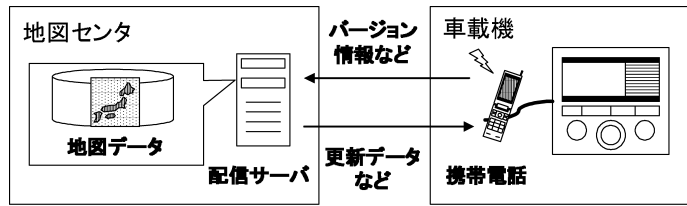


図 1 テレマティクス地図更新システムの構成  
Fig. 1 Telematics map update systems.

ず、必要範囲の地図データを随時取得するサービスも実用化されているが、この方式ではトンネル内などの通信が不可能な環境では使用できないうえ、通信データの量がユーザの費用負担が非常に大きくなるという問題がある。

そこで、道路の追加・削除など地図の変化情報である差分データのみを配信することにより、ごく小さな通信データ量でカーナビの地図を更新する方式（地図差分更新方式）が研究されている<sup>5),9)</sup>。また、組み込み環境向けの DBMS を導入することによりファイルにデータを格納するよりも容易にデータを管理できるため、地図差分更新に適した DBMS の管理方式も研究されている<sup>10)</sup>。これらを用いたテレマティクス地図差分更新システムの概要を図 1 に示す。本地図更新システムは、カーナビの車載機と地図センタによって構成される。車載機は内蔵の HDD に地図データを持っており、携帯電話などの無線通信にて地図センタから最新の地図データを受け取る。地図センタでは地図データベンダより定期的にリリースされる地図データを管理しており、リリースごとに地図データ間の差分データを生成する。この差分データは車載機に応じた形式に変換され配信される。

ところが、現状のテレマティクスには通信コスト、通信時間の面で大量のデータを送受信するのが困難であるという問題がある。たとえば、運転中にカーナビの地図が現実と合わないと感じた際に、短時間で安価に地図更新できるサービスとしてダウンロードが 1 分以内、通信費が 200 円程度で地図を更新できるサービスを考える。ここで 2007 年時点の標準的な携帯電話パケット通信として実効速度が 150 kbps、通信費は 0.02 円/パケット（1 パケット 100 byte 程度）を想定すると、1 分では 1 MB 程度しか転送できない。カーナビで用いられる地図データは 10 GB 以上にもなるため、従来のように最新地図データを配信しただけではテレマティクス地図更新サービスは実現できない。そこで、我々は差分更新方式による配信データの削減を検討した。しかし、カーナビ製品で現在使用されて

いる地図データの差分を抽出したところ、差分データのデータサイズは日本全国で年間 63.7 MB（150 kbps でダウンロード時間 56 分）、ダウンロード後の更新時間は 54 分にもなることが判明した。このような更新を運転中に実行することは、現実的ではない。また、運転時以外に更新する方法であっても、自動車を無人にしたままエンジンをかけておくことは危険であり、かといってエンジンを切った状態で動作させると自動車のバッテリーがあがってしまうため、ユーザが 1 時間近くわたって車内で更新完了を待たなければならず、実用的ではない。このように、差分更新であっても日本全国を 1 度に更新しては実用的なサービスとはならない。

そこで、本システムでは自車位置周辺のみ更新するなど、更新領域を限定することにより小さな配信データで地図を更新するスポット更新サービスを導入する。このサービスには、以下の要件を満たす方式が必要である。

1. 地図を部分的に更新しても周辺の地図となるべく不整合を起こさないこと
2. 通信データサイズが十分小さく、地図更新処理に時間がかからないこと

ここでいう不整合とは、地図のある部分を更新したとき、その領域の周辺の地図はまだ更新されていないために、更新領域の境界で地図が不具合を起こすことを指す。たとえば、建物など大きさを持った地物が更新領域内外の境界上にあるとき、建物の半分のみが更新されることがありうる。また、新規鉄道が敷設された場合、鉄道線の先の駅舎も同時に追加しなければならないなど、種別間での不整合もある。これらの地図の不整合を回避する方法としては、たとえば車載機側で周辺と整合がとれるように地図データを書き換えるアプローチが考えられるが、この方法では車載機で多くの処理が必要となるため、更新処理に時間がかかってしまう。また、上述の「Air ナビ<sup>3)</sup>」や文献 4) などのように、アプリケーションが必要とした時点でその範囲のデータをダウンロードする方法も考えられ

るが、トンネル内などで通信が不可能になるとアプリケーションの動作もできなくなるという問題があった。

この不整合は、地図のある部分の更新の影響が他の部分にまで広がってしまうことに起因する。しかし、たとえば東京で道路が建設されても、そこから十分遠方にある北海道の地図に変更は加わらないなど、更新領域から十分遠方は更新による影響は受けないと考えられる。そのため、更新の影響を受ける範囲外まで更新領域を拡大して更新することにより不整合を回避することができると考えられる。ただし、通信データサイズを減らし、更新処理時間を小さく抑えるには、この更新領域の拡大はできるだけ小さくする必要がある。

経路探索や案内誘導などのナビゲーション機能に用いられる道路ネットワークの場合、道路ネットワークの不整合は接続先道路が未更新であることに起因している。そこで、本論文ではこの不整合の回避を道路の参照整合性の維持に帰着させることにより、地図の整合性を保ったまま部分的に地図を更新できる接続保証方式を提案する。この方式では地図センタ側において接続性を保証できるように更新データを生成しておき、それを車載機の要求に適合するように配信することができる。そのため、車載機で接続性を保つ処理を実行する必要がなく、地図データが更新できるため、車載機の処理負荷は比較的小さいと考えられる。また、提案方式の有効性を確認するため、実験により配信データサイズと更新の波及範囲について評価した。

## 2. 地図差分更新技術の背景と課題

### 2.1 地図差分更新の定義

地図データとは、現実に存在する道路などの地物を記述したデータである。地図データは、地物 1 つを意味するオブジェクトで構成されており、オブジェクトには道路地図や背景（水系など）、地点情報（POI; Point Of Interest）など多くの種類がある。したがって地図データを構成するオブジェクトが同一であれば、地図データは同一であるといえる。このような地図データはある時点での測量などの結果に基づき生成される。現実の地物には建設工事などにより変更が加えられるため、生成された時点によってオブジェクトの構成は異なる。このようにオブジェクト構成に変化があることを地図が更新されたという。そこで、これらの地図データに対し、初回生成の地図を 1 とし、以降は 1 ずつ増えるように対応付けた通し番号  $i$  をバージョンと定義する。以下ではバージョン  $i$  で生成されたオブジェクトを  $m_i$ 、それらの集合で定義される地図を  $M_i$  と表記する。

地図の更新とは、地図  $M_i$  に基づいて作成された地図に処理を加え、 $i < j$  となる地図  $M_j$  に基づいて作成された地図と同じオブジェクト構成とすることで、地図の差分更新とは、これらのオブジェクトに対する変化情報だけを用いて地図を更新することを指す。オブジェクトは、道路の種別などの属性値や施設の位置を示す座標値など、地物の状態を表現するための値を保持している。以下では、これらの値列をオブジェクト値と呼ぶ。これにより、オブジェクトの変化とは、オブジェクト値の変化に帰着される。

オブジェクト値の変化を判定するためには、同一の地物を表現するオブジェクトであるかをバージョンによらず判定できる識別子が必要である。この識別子としてバージョンによらずオブジェクトを一意的に識別可能な値であるオブジェクト ID を定義する。オブジェクトは、オブジェクト ID とオブジェクト値の組合せによって表現される。そこで、以下ではオブジェクトを 0 番目の成分をオブジェクト ID、以降にオブジェクト値を並べたベクトルを用いて表記する。すなわち、バージョン  $i$ 、オブジェクト ID  $m^0$ 、オブジェクト値  $\hat{m}_i$  のオブジェクトは  $m_i = (m^0, \hat{m}_i) \in M_i$  と表記される。また、あるオブジェクトが他のオブジェクトを参照する際には、参照先のオブジェクト ID をオブジェクト値の一部として含むものとする。

オブジェクト間の差分は insert (追加), delete (削除), update (更新) で表現される。これらの処理を表現するため、オブジェクト値に未追加または削除済みを示す 0 ベクトル値を導入する。これにより、オブジェクトの更新  $m_i \rightarrow m_{i+1}$  を定義でき、

$$\text{insert: } m_i = (m^0, 0) \rightarrow m_{i+1} = (m^0, \hat{m}_{i+1})$$

$$\text{delete: } m_i = (m^0, \hat{m}_i) \rightarrow m_{i+1} = (m^0, 0)$$

$$\text{update: } m_i = (m^0, \hat{m}_i) \rightarrow m_{i+1} = (m^0, \hat{m}_{i+1})$$

と記述できる。これにより、地図更新を写像  $f: M_i \rightarrow M_{i+1}$ ,  $f(m_i) = m_i + \Delta m_i$  と定義できる。ここで、 $\Delta m_i = (m_{i+1} - m_i)$  は、 $m_i$  のオブジェクト値の差分であり、オブジェクト値の変換を意味する。以下では、 $\Delta m_i \neq 0$  を差分オブジェクトと呼ぶ。

### 2.2 スポット地図更新サービスの課題

スポット更新サービスは、更新対象となる地域内の地図  $A_i^u \subset M_i$  を差分更新するサービスである。ここで、地域を限定した地図の部分集合  $A_i^u$  を更新エリアと呼び、更新エリアを識別する更新エリア ID $_{\mu}$  を付与する。 $m_i \in A_i^u$  を満たす  $\Delta m_i$  のみを配信した例を図 2 に示す。図 2(a) の地図が次バージョンでは図 2(b) となった場合に、更新エリア A に限定して地図を更新した結果が図 2(c)、更新エリア B に限定

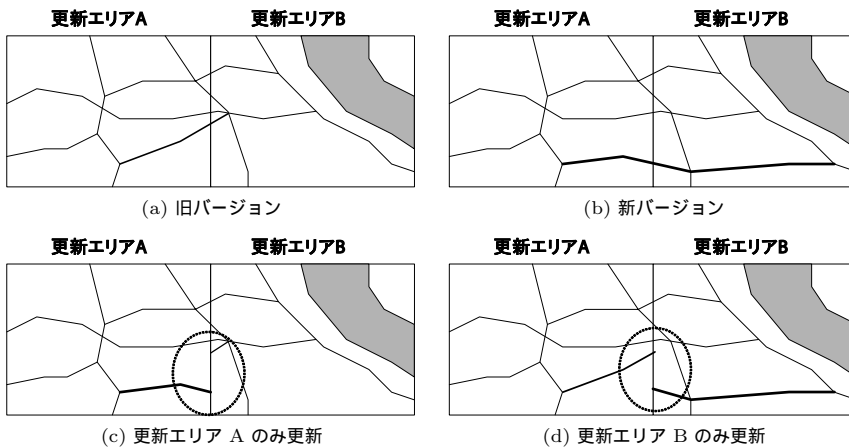


図 2 部分的地図更新による道路の切断

Fig. 2 Disconnection from partial map update.

し地図を更新した結果が図 2 (d) である。これらは新規道路が途中までしか追加されないなど、いずれも道路の接続が保たれない。経路探索アルゴリズムでは接続していない道路間には行き来できずと探索するため、このデータを用いると隣接エリアへの経路がまったくない、主要道路が袋小路になり通行できないなどの不具合を発生する。また、道路を更新するときには、誘導データなど道路に関連づけられているデータも同時に更新しなければ、参照先がないなどの不整合を生じる。このように地図の不整合は経路探索や誘導アプリケーションなど多様なナビゲーション機能に影響を及ぼす。そのため、スポット更新サービスではこのような道路ネットワークの切断は避けつつ、十分小さなデータサイズで更新が可能でなければならない。たとえば  $A_i^u$  の更新により道路  $R$  が切断される場合、道路  $R$  の接続先データを含む隣接エリア  $A_j^v$  も同時に更新することにより道路  $R$  の接続を保つことができる。ところが、この  $A_j^v$  の更新にともない他の道路  $R'$  が切断される場合、さらに道路  $R'$  の接続先更新エリア  $A_k^w$  の更新も必要となる。このように、単に更新エリア単位で地図を更新したのでは、道路の接続を維持するために次々に更新を波及させることになり、配信すべきデータ量が大きくなりすぎる。そこで本研究では、地図の更新を差分オブジェクト単位で管理することにより、接続を保つための波及領域の拡大を最小限とすることを旨とする。

### 3. 更新エレメントを用いた接続保障方式

#### 3.1 ネットワークの接続性とその定義

カーナビの経路探索や誘導機能は道路間の接続情報をもとにナビゲーションを行っている。この接続情報

は、交差点および属性変化点を意味するノード  $n_i$  とノード間の道路区間を意味するリンク  $l_i$  によって記述される。ノードは必ずリンクに接続しており、どのリンクとも接続しないようなノードは存在しない。これらのデータはパーセルと呼ばれる矩形単位で管理されており、リンクはパーセルの境界をまたがることはなく必ず境界とリンクの交点はノードになる。したがって、2つの異なるパーセル  $P, P'$  に対し、リンク  $l_i$  は必ず  $l_i \in P \Rightarrow l_i \notin P'$  を満たし、 $P, P'$  の境界にあるノード  $n_i$  は  $n_i \in P \cap P'$  となる。以下では、パーセルにはリンクとその両端ノードが格納されており、パーセル境界のノードは2つのパーセルに重複して存在し、ノード ID を通じてパーセル境界のノードが同一であることを判定できるようになっているものとする。以下ではこのパーセルを一意に識別できる識別子をパーセル ID と呼ぶ。

今回は、更新エリアとしてこのパーセルを用いる。これにより、更新エリア内の道路地図データ更新とは、パーセルに含まれるリンクおよびその両端ノードを更新することと定義でき、ノードに対する更新はリンクに対する更新に付随するものと見なせる。以下に示す例では、簡単のためにノードに対する更新の記載を省略するが、実際には各リンクの両端ノードに対する更新も行われるものとする。以上により、本論文ではパーセル単位で道路ネットワークを更新しても接続性を保証できる地図更新方式を検討する。

図 3 にリンクに対する差分オブジェクトを用いた更新の例を示す。この例は新たに交差点ができる場合の更新について、地図内のすべての差分オブジェクトを適用しなかった結果である。例 1 では、新たに追加された道路が旧来の道路と接続していない。例 2 では実

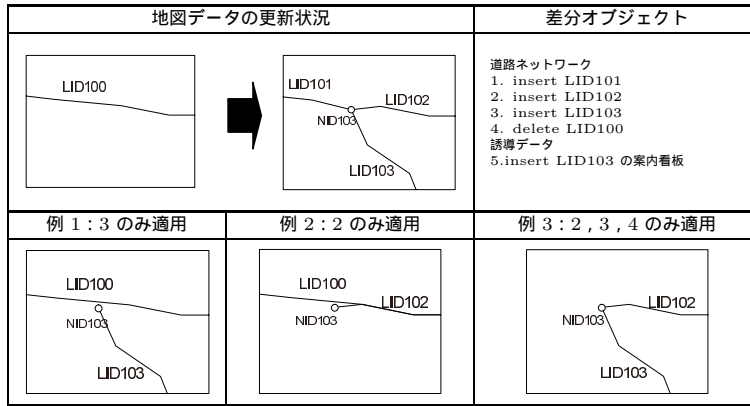


図 3 差分オブジェクト単位更新の例  
 Fig. 3 Sample of map updating by difference objects.

態としては同一のはずの道路が旧来のデータに重なっている．例 3 では，旧来の LID100 の道路が途中で切断されている．また，各例において LID103 に関する誘導データが追加されていない．例 2 の場合，追加された誘導データに関連づけられたリンクが存在しない．例 1 や例 3 の場合，地図データと異なる誘導がなされてしまう．このように，道路間が適切に接続しかつ誘導情報などの関連データも関連づけるように，更新を波及させなければならない．

これらの不整合は，あるオブジェクトのバージョンとそれが参照している他のオブジェクトのバージョンとが異なっていることに起因している．これを記述するため，バージョン  $i$  のあるオブジェクト  $a_i$  が参照する他のオブジェクト ID  $x^0$  の集合を参照オブジェクト  $\text{ref}(a_i)$  と定義する．ここでいう参照とは，リンクが保持しているその両端のノード ID を保持している，またはノードがそのノードに接続するリンクのリンク ID を保持しているという状態を指す．このとき，地図  $M$  が  $i < j$  となる 2 つのバージョン  $i, j$  のオブジェクトを含むとき，

$$\forall a^0 \in \text{ref}(m_i \in M) \Rightarrow a_i = (a^0, \hat{a}_i) = (a^0, \hat{a}_j) = a_j \in M \quad (1)$$

を満たすならば，その地図  $M$  は正則である，と定義する．また，地図データが正則でないことを「矛盾している」と呼び，これを不整合と見なす．

3.2 更新エレメントとその定義

各バージョンの地図データに矛盾がなければ，2バージョン間にあるすべての差分オブジェクトを適用した結果は矛盾しない．すなわち，単体で実行すると地図に矛盾を発生させる差分オブジェクトであっても，複数の差分オブジェクトをまとめて適用することにより

地図データの矛盾を避けられる．地図データの矛盾はオブジェクトとその参照オブジェクトとの間のバージョン差異によって発生するため，差分オブジェクト  $\Delta m_i$  の参照先  $\text{ref}(\Delta m_i) = \text{ref}(m_{i+1}) \cap \text{ref}(m_i)$  に関する差分オブジェクトを同時に適用することにより，オブジェクトの差分更新にともなう地図の矛盾を避けることができる．この方式を接続保証方式と呼ぶ．

この接続保証方式では，更新が更新エリア外まで波及するため，周辺エリアから更新の波及を受けることがある．そのため，従来の地図データに一意に番号をつけるだけのバージョン管理では正しく管理できない．そこで，接続を保証できるように差分オブジェクトを集めた集合に全バージョン間で一意の ID をつけ，適用済みとなる差分オブジェクトの集合の ID を記録することによりバージョンを管理する．そこで，以下の条件 1, 2 を満たす差分オブジェクトの集合  $E_i^\xi$  を，バージョン  $i$  からバージョン  $i + 1$  への更新エレメント ID  $\xi$  の更新エレメントと定義し，これを用いてバージョンを管理する．

1.  $\forall \Delta m_i \in E_i^\xi, \Delta m_i \neq 0$
  2.  $\exists b^0 \in \text{ref}(\Delta a_i \in E_i^\xi), \Delta b_i \neq 0 \Rightarrow \Delta b_i \in E_i^\xi$
- (2)

更新エレメント  $E_i^\xi$  に含まれるすべての差分オブジェクトを地図  $M_i$  に適用した結果得られる地図は正則である．たとえば，道路ネットワークの更新では，図 4 に示すように処理対象のリンクがつながっているもの，およびこれらのリンクの ID を参照している誘導情報など差分オブジェクトを集めたものが，更新エレメントとなる．したがって，この更新エレメントを条件を満たしながらも要素数が最小となるように生成すれば，最小限の差分オブジェクトの適用で道路の

バージョン	ver.1	ver.2	ver.3	ver.3のみ適用
地図データ				
更新内容	追加: L101 ~ L105	追加: L201 ~ L205 削除: L103	追加: L301 ~ L306 削除: L203	追加: L101 ~ L105 追加: L301 ~ L306 削除: L203

図 5 依存関係を持つ更新エレメントの例  
Fig. 5 Example of update-element with dependency.

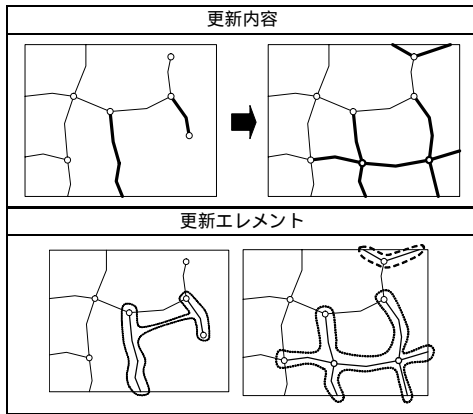


図 4 更新エレメントの例  
Fig. 4 Examples of update elements.

接続を保ちながら地図が更新できる。

3.3 複数バージョンの差分更新

次に、この2バージョン間の差分オブジェクトによって構成される更新エレメントを用いて、3バージョン以上の地図更新を行う方法を検討する。バージョン  $i, i+1, i+2$  の3つのバージョンが存在する場合、更新エレメントは  $i \rightarrow i+1, i+1 \rightarrow i+2$  の2種類が存在することになる。このとき、 $i+1 \rightarrow i+2$  の更新エレメントを適用するために  $i \rightarrow i+1$  の更新エレメントが適用済みでなければならないことがある。図5にその例を示す。この例では、バージョン3における更新内容をバージョン1のデータに適用すると破線部の道路間が適切に接続しない、点線部の誘導情報と地図データが合わず、矛盾している。

この例のように、バージョン3における更新エレメントを適用する前に、バージョン2の更新エレメントを適用しなければならないことがある。ある更新エレメント  $E_i$  の適用前に  $i > j$  となる更新エレメント  $E'_j$  を適用しなければ地図が矛盾することを更新エレメントが依存する、と定義する。以下では、 $E_i$  の依

存する更新エレメントの集合を  $rel(E_i)$  と記述する。 $E_i$  の依存する更新エレメント  $E'_j \in rel(E_i)$  は以下の1または2のいずれかの条件を満たす。

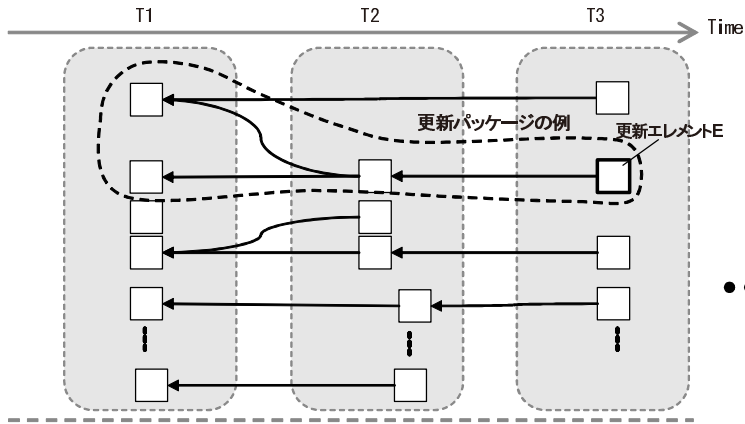
$$\begin{aligned}
 1. & \exists \Delta e_j \in E'_j, e^0 \in ref(m_i \in E_i) \\
 2. & \exists \Delta e_j \in E'_j, \Delta e_i \in E_i
 \end{aligned}
 \tag{3}$$

地図に更新エレメントを矛盾なく適用するには、その更新エレメントの依存する更新エレメントのうちまだ適用していないものだけを適用すればよい。これを依存関係の解決という。そこで、構成するオブジェクトのバージョンがすべて  $i$  以下の地図  $M$  に対し、 $\mu$  番目のパーセル  $P^\mu$  を更新してバージョン  $i$  とするための依存関係を解決した更新エレメントの集合を更新パッケージ  $C_i^\mu$  と定義する。すなわち、更新パッケージ  $C_i^\mu$  は以下の条件1, 2を満たす。

$$\begin{aligned}
 1. & \forall m_k \in P_k^\mu (k \leq i), \Delta m_k \in E_k \Rightarrow E_k \in C_i^\mu \\
 2. & \forall E \in C, rel(E) \subset C_i^\mu
 \end{aligned}
 \tag{4}$$

車載機の適用済み更新エレメントの集合を  $T$  とすると、 $C_i^\mu \cap T$  に含まれる更新エレメントをすべて適用することにより、車載機の地図  $M$  の  $\mu$  番目のパーセルがバージョン  $i$  に更新され、 $P_i^\mu \subset M$  となる。

更新パッケージの概念図を図6に示す。図中の四角形は更新エレメント、T1~T3は地図データのバージョン、更新エレメント間の依存関係は矢印でそれぞれ表記されており、過去の更新エレメントに対して依存関係がある様子を表している。たとえば、車載機の地図データに対し、T1でリリースされた更新エレメントがすべて適用済みである例を考えると、そのデータに太線四角で書かれた更新エレメント  $E$  を適用するには点線で囲まれた4つの更新エレメントを適用しなければならない。したがって、この4つの更新エレメントの更新エレメントIDをまとめたものが、この例における更新パッケージとなる。



凡例：図中□は「更新エレメント」1更新周期間に発生した変更データを表す。図中の→は更新エレメント間の依存関係を示す。

図 6 更新パッケージの概念図

Fig. 6 Picture of update package.

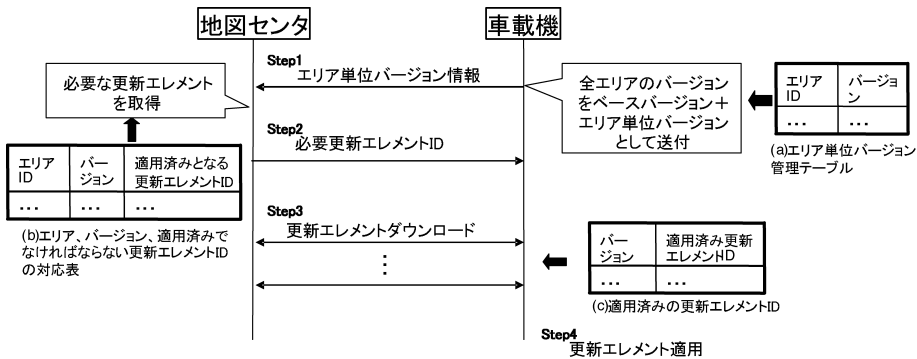


図 7 更新エレメントの取得手順

Fig. 7 Steps to download update-elements.

3.4 更新エレメントを用いたバージョン管理方式  
 車載機でパーセル ID  $\mu$  のパーセルをバージョン  $i_\mu$  に更新したとき、パーセル ID  $\mu$  のパーセル単位バージョンが  $i_\mu$  である、と定義する。このときの更新パッケージを  $C_{i_\mu}^\mu$  とすると、車載機の適用済み更新エレメントの集合は  $T = \bigcup C_{i_\mu}^\mu$  と一意に定まる。したがって、パーセル ID  $\mu$  と最終更新時点のバージョン  $i_\mu$  のみを送付することにより、地図センタで地図更新に必要な更新エレメントだけを選び出すことができる。以下ではこれを用いて更新エレメントによるバージョン管理を行う。

図 7 に、地図更新を行う処理の手順を示す。これらの手順では、車載機を持つ地図データのバージョンと、配布すべき更新データのバージョンを比較することにより、車載機が適用しなければならない更新エレメント ID の一覧を作成する。この方法としては、地

図センタから更新対象の更新エリアを更新するのに必要な更新エレメント一覧を送付し、車載機にて保持している適用済みの更新エレメント ID と比較する方法が考えられる。しかし、この方法では車載機に負荷がかかり、更新エレメント ID の送付のために送信データの量が多くなるという問題がある。そこで、パーセル単位にバージョンを記録することにより、この選別を簡易化する。

まず Step 1 では図 7(a) のように車載機から更新対象とするエリア ID と地図のバージョンを送付する。このバージョンとしては、各パーセルを更新した時点での最新の地図センタにおけるバージョン番号を用いる。なお、初期出荷時などはこれら全パーセルのバージョンが均一になるため、この均一のバージョンとエリア単位更新が行われたパーセルのバージョンのみを地図センタに送信することにより、送付するバージョン情報のサイズを小さくすることができる。

表 1 本実験で用いた差分データ

Table 1 Difference data format for this experiment.

カラム種別	リンク ID	始点ノード ID	終点ノード ID	道路種別などの属性値	形状データ
サイズ	4 byte	4 byte	4 byte	12 byte	可変 (4 byte/補間点)

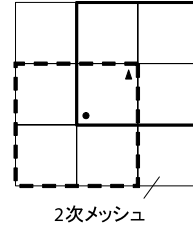
Step 2 では、地図センタに送付されてきたパーセル単位バージョンを用い、車載機が必要とする更新エレメント ID の一覧を取得する。地図センタでは、図 7(b) のように、あるパーセルを更新した際に適用済みでなければならない更新エレメント ID の一覧をパーセル単位バージョンごとに保持している。車載機から送付されてきたパーセル単位バージョンを用いて図 7(b) の対応表をひくことにより、車載機が適用済みであるはずの更新エレメント ID の一覧が取得できる。したがって、これに含まれる更新エレメント ID を更新パッケージ中の更新エレメント ID から除外することにより、車載機に必要な更新エレメント ID を選別できる。

車載機は、Step 3 で地図センタから返送された更新エレメント ID の一覧を参照しながら更新エレメントの実体をダウンロードする。この際、図 7(c) のように適用済み更新エレメント ID を管理していれば、パーセル単位更新以外によって適用された更新エレメントがあったとしても、これを除外することができる。Step 4 ではダウンロードした更新エレメントを適用することにより地図データの更新が完了する。この手順により、更新エレメントを用いても通信量を小さくしたまま地図データを更新できる。

#### 4. 提案方式の評価実験

本研究の目的は、地域を限定して地図を更新することにより地図の配信データ量を削減することである。ただし、従来の更新対象地域のデータを差し替える方式では、道路が矛盾するため、提案方式では矛盾が発生しない範囲まで更新領域を広げている。そのため、差分データの量は更新領域を広げた分だけ従来よりも大きくなることが予測される。したがって、本方式の有効性を検証するには、更新エレメントの大きさや数が携帯電話やインターネットによる配信において実用的な範囲になっているかどうかを実測しなければならない。

そこで今回、カーナビ製品で使用されている日本全国の地図データの、2005 年 1 月から 2005 年 5 月までの 4 カ月間の差分データを抽出し、その差分データから更新エレメントを生成してデータ量や更新処理にかかる時間を評価した。この地図データは、JIS<sup>1)</sup> で



2次メッシュ

図 8 本評価の更新範囲

Fig. 8 Area to update for evaluation.

規格としても採用されている KIWI フォーマットを基本としたもので、2次メッシュ<sup>8)</sup> (第2次地域区画<sup>2)</sup>) を縦横 4 分割、計 16 分割した単位をパーセルとし、日本全国をおよそ 6 万パーセルに分割して管理している。今回は、この地図データをもとに KIWI で規定されている道路リンクデータに変換し、それぞれ固有の ID を付与することにより、提案方式を適用可能なフォーマットに変換した。この処理を上記 2 つのデータに対しそれぞれ行い、2 データ間の差分を抽出した。なお、今回のデータサイズ計測に用いた差分データは、CSV テキストファイルを GZIP 圧縮したものを用いた。表 1 にリンク 1 レコードのフォーマット概要を示す。このリンクデータのデータ長は、リンクの形状によって異なるが、全データの平均値は 112.0 byte であった。

提案方式では、更新エリアはパーセルとしての性質を満たしていれば任意の大きさとする事ができるが、複数の更新エリアを同時に更新することにより任意に拡大することができるため、更新エリアは小さな方が望ましい。ただし、更新エリアの数が多くなりすぎると、車載機から地図センタへ送付しなければならないバージョン情報の数が増大する。今回の実験で用いたデータの場合、最小の管理単位のパーセルが 6 万パーセルあるため、バージョンが 2 byte で表記されると仮定すると 120 KB 近くまでなってしまう。

そこで今回の評価では、2次メッシュ 1 つに相当する 16 パーセルを更新エリアとし、現在地周辺の 4 つの更新エリアを組み合わせ更新することを想定した。この場合、送付すべきデータ量は 10 KB 程度にまで減少させることができる。また、地図更新を行った時点での現在地が更新領域の境界とならないよう、最も近い 2次メッシュ 4 枚を更新している。図 8 にこの更



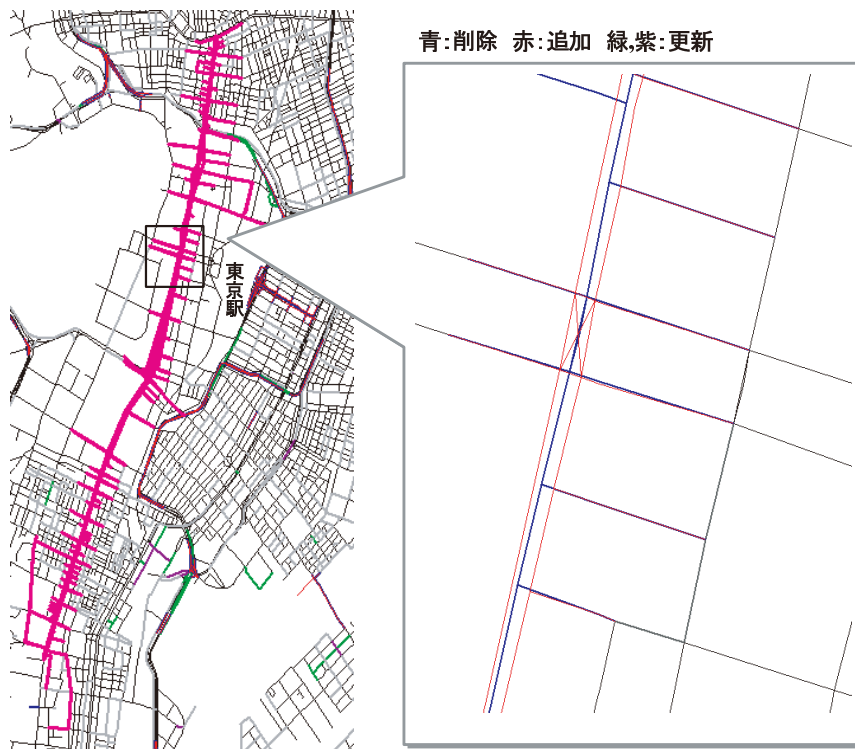


図 9 更新エレメントの例

Fig. 9 Example of an update-element.

新範囲を示す。現在地が三角の記号で示される位置であれば太い実線を、円で示される位置の場合には太い破線をそれぞれ更新する。これにより、現在地から更新パーセル外までは必ず2次メッシュの短辺長÷2の距離、およそ5kmあることが保証される。更新する範囲がたとえば経路沿いなどの線分に基づくものであるなどした場合でも、1点に対する更新を組み合わせることによって対応できるため、今回の実験の更新単位は基本的な更新の例であると考えられる。

本評価では以上の条件に基づき、提案方式である更新エレメント単位での配信と接続性を保証せず単純にパーセル内を更新する接続なしの方式、および2章で説明した道路が切断しなくなるまで更新範囲を拡大する従来方式の3つについて、ダウンロードすべきデータ量およびその時間とダウンロード後の処理時間を見積もった。

#### 4.1 更新エレメントの例

提案方式は、ある道路を更新する際に周辺道路も書き換えなければ道路が寸断されてしまうという問題を同時に周囲の道路を更新することによって回避する方式である。このような周辺道路の更新によって寸断を回避できる更新エレメントの例として、図9に東京

駅前の二条線化にともなう更新エレメントを示す。図中の黒線は変更のなかった道路、赤線は追加された道路、青線は削除された道路、緑、紫の線は形状や属性に変更のあった道路を示す。この図中で明るい紫で示された部分が1つの更新エレメントとなっている。この更新エレメントは、従来は1つの道路データによって表現されていた東京駅前の道路が、中央分離帯を挟む2本の道路として表現されるようになったという更新である。右拡大図から分かるように、この更新エレメントでは変更のあった道路だけでなく、周辺道路がこの二条線に接続するように更新されている。この例のように、本方式により道路の更新時、周辺道路との接続を保つ単位での更新が可能である。

#### 4.2 配信データサイズに関する考察

今回の実験で生成された更新エレメントを用い、今回想定するスポット更新サービス（隣接する2次メッシュ4枚）でのありうるケースすべての配信データ量を計算した。これを用い、横軸を配信データ量、縦軸をその配信データ量となるケースの相対度数としたグラフを図10(a)に示す。接続なしの場合、更新対象のパーセル内のデータしか更新しないため、大半は100KB未満のデータサイズであり、400KBを超える

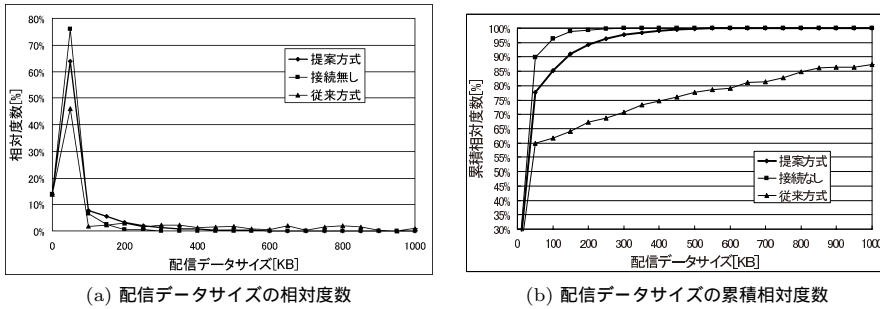


図 10 2 次メッシュ 4 枚を更新する場合の配信データサイズの相対度数分布

Fig. 10 Distribution of data size for updating 4 meshes.

表 2 地図更新の処理ステップと処理時間

Table 2 Map updating steps with processing time.

手順	処理内容	処理性能 [ms/parcel]
Step 1	パーセルデータを取得	69.04
Step 2	読み出したデータに対し、追加・削除・更新を適用	49.83
Step 3	パーセルデータを DBMS に受け渡し更新	46.20
	合計処理時間	167.40

ような更新データは存在しなかった。他方、提案方式や従来方式では更新対象のパーセル外のデータも配信されるため、全体的に配信データ量が大きく、400 KB 以上の大きな更新データとなるケースも多数あった。ただし提案手法は、従来手法に比べ更新データとなるケースの頻度が小さく、250 KB を超えるような極端に大きな更新エレメントはごく希にしか存在しなかった。実際、図 10 (b) に示したこの相対度数の累積値（累積相対度数）から、提案手法ではおよそ 95% の確率で配信サイズが 250 KB 以下となるのに比べ、従来手法では 1 MB を超えるような場合も多数あることが分かる。なお、従来手法の累積相対分布が 95% となるのは、配信データサイズが 4,400 KB のときであった。

#### 4.3 更新処理時間に関する考察

本方式を用いた場合の更新時間は、ダウンロード時間と更新処理時間の和によって見積もることができる。ダウンロード時間は現在想定されている携帯電話通信の通信性能 150 kbps を用いることにより、接続なしでは  $100 \text{ [KB]} \div 150 \text{ [kbit/s]} = 5 \text{ [s]}$ 、提案方式では  $250 \text{ [KB]} \div 150 \text{ [kbit/s]} = 13 \text{ [s]}$ 、従来方式では  $4,400 \text{ [KB]} \div 150 \text{ [kbit/s]} = 235 \text{ [s]}$ 、となると予測できる。

現在、本システムで用いる地図フォーマットは、道路データをパーセル単位で DBMS の 1 レコードとして格納する方式を用いている。したがって、地図データの更新処理はパーセルバイナリデータの取得、取得したバイナリデータの更新、更新済みバイナリデータ

による RDBMS の更新、の 3 つの手順に沿って行われる。この手順にかかる処理時間を東京駅周辺の 2.5 次メッシュ（各 1 MB 程度）400 枚に対し測定したところ、平均的には表 2 のようになった。

これを用い、本方式の更新処理時間を見積もる。本実験データを用い、2 次メッシュ 4 枚単位の更新を行う場合の更新対象パーセル数の相対度数をとったグラフを図 11 (a)、累積相対度数のグラフを図 11 (b) にそれぞれ示す。このグラフによると、接続なしでは更新対象パーセル数が 50 未満となるケースが 95% を占めており、更新の波及がないため、64 パーセル以上になることはない。一方、提案方式や従来方式では更新対象パーセル数が 64 以上になり、従来手法では 2,300 パーセル以下、提案方式では 100 パーセル以下が 95% を占めている。これらのケースの平均的な車載機更新処理時間は、接続なしの場合はおおむね  $167.40 \text{ [ms/parcel]} \times 30 \text{ [parcel]} = 5.02 \text{ [s]}$  以下、従来方式ではおおむね  $167.40 \text{ [ms/parcel]} \times 2,300 \text{ [parcel]} = 385.02 \text{ [s]}$  以下、提案方式ではおおむね  $167.40 \text{ [ms/parcel]} \times 100 \text{ [parcel]} = 16.74 \text{ [s]}$  以下であると予測される。

以上の評価結果より、4 カ月分の差分を 2 次メッシュ 4 枚の道路ネットワークを更新する場合には、これらの時間を合計し、接続なしでは 10 [s] 程度、従来方式では 620 [s] 程度提案方式では 29 [s] 程度かかることが分かる。

2 次メッシュは 1 辺が 10 km 程度あるため、今回の

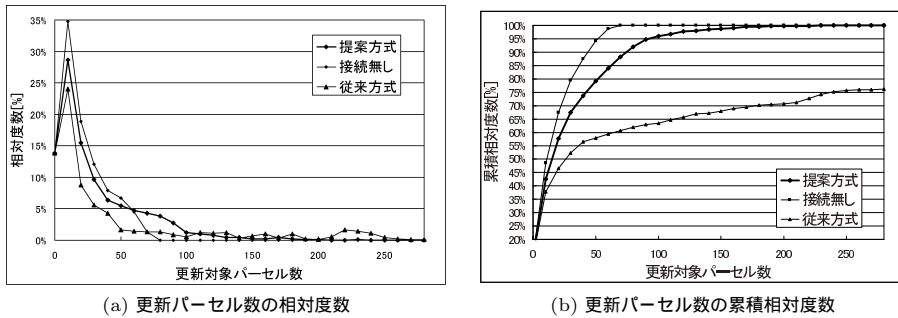


図 11 更新パーセル数の相対度数分布

Fig. 11 Distribution of mesh number to update.

更新範囲外までは最短で 5 km ある。すなわち、更新範囲外に出るまでには、国内一般道の制限速度 60 km/h で走行すれば 5 分、高速道路の制限速度 100 km/h で走行すれば 3 分かかると考えられる。接続なしで更新した場合、この範囲の地図を 3 分未満で更新できるが、道路の接続は保たれない。つねに通信が可能であれば、必要になるつど更新範囲を拡大すればよいが、車載機は通信不可能になることがあるため、それでは十分ではない。通信不可能なのが短時間であれば、その分遠方の地図まで更新しておく方法も考えられるが、通信不可能な期間がどの程度になるかは不確定である。これを確実に道路が接続するように準備しようとしたのが従来方式であるが、従来方式によると 10 分以上の時間がかかってしまう。一方、提案方式を用いると、接続なしに比べると時間やデータサイズが大きいものの、3 分を十分下回っており、走行中も十分な広範囲を更新可能である。提案方式では、更新ごとに接続が保たれているため、更新後に通信が不可能になったとしても道路の接続が保証できる。

実サービスとしては、背景地図（行政界や水系など）や POI（Point Of Interest; 店舗などの地点情報）など道路ネットワーク以外にも更新が要求されるデータがあるが、これらのデータは道路ネットワークと異なり、接続を保障しなくても誘導機能には不具合をきたさないと予測される。以上により、本方式によって携帯電話による道路地図更新サービスの実現可能性が確認できた。

## 5. おわりに

### 5.1 まとめ

本研究では、携帯電話の通信を用いたカーナビの地図データ更新サービスの実現に向け、地図を現在地付近などに限定して更新するスポット更新サービスを実現するために道路ネットワークの接続を保つ更新単位

である更新エレメントを事前に生成し ID で管理する接続保障方式を提案した。本方式を評価するため、地図データの差分を抽出し、更新エレメントを生成し、性能見積りを行った。この結果、20 km 四方の 4 カ月分の差分データが、ほとんどのケースで配信データサイズは 250 KB 以下、更新にかかる時間は 1 分程度に収まることが分かった。これにより、携帯電話で地図データを更新するサービスを実現できる見通しを得た。

### 5.2 今後の課題

本方式は、道路ネットワークに関し現在地付近のみに限定して地図を更新するサービスである。今回、直接接続している道路のみの整合性を取り扱ったが、もっと遠方のデータを必要とするアプリケーションにとっては、ほかの方法で整合性を保たなければならないことがありうる。どのように整合性を保つべきかはアプリケーションの実装によって異なるため、ナビゲーション機能が想定外の不具合を起こす可能性がある。しかし、とりうるすべての組合せでナビゲーション機能を検証することはできない。このような、部分的に更新したデータに関する品質管理の方法は、今後の課題である。

## 参考文献

- 1) 日本工業規格自動車—カーナビゲーションシステム用地図データ格納フォーマット, JIS D 0810.
- 2) 日本工業規格地域メッシュコード, JIS X 0410.
- 3) Pioneer: Air agent.com.  
<http://www.air-agent.com/>
- 4) Sakamoto, A., Ohsawa, Y. and Akai, S.: A proposal on data renewal method for car navigation system, *Proc. ISPRS2000* (2000).
- 5) Tanizaki, M., Maruyama, K. and Shimada, S.: Acceleration technique of snale-shaped regions retrieval methods for telematics navigation service system, *21st International Conference on Data Engineering (ICDE2005)* (2005).

- 6) トヨタ自動車：G-book.com.  
<http://g-book.com/>
- 7) (株) 矢野経済研究所：2004年カーナビ市場調査結果(2004).
- 8) 総務省統計局：地域メッシュ統計.  
<http://www.stat.go.jp/english/data/mesh/>
- 9) 大島雄太, 櫻井政幸, 桑原 浩, 根岸幸生, 川崎洋, 大沢 裕：暗示的位相構造を考慮した差分空間データの配信方式, 情報処理学会研究報告データベースシステム, Vol.2003, No.71, pp.295-302 (2003).
- 10) 伊藤大輔, 谷崎正明, 嶋田 茂：地図データの更新に対応した組み込みデータベースのデータ配置方式, 情報処理学会研究報告データベースシステム, Vol.2006, No.78, pp.139-144 (2006).
- 11) 日産自動車：Carwings.  
<http://drive.nissan-carwings.com/WEB/>
- 12) 本田技研工業株式会社：internavi premium club. <http://www.premium-club.jp/>

(平成 19 年 4 月 2 日受付)

(平成 19 年 10 月 2 日採録)



浅原 彰規

平成 14 年北海道大学理学部物理学  
学科卒業。平成 16 年北海道大学  
大学院理学研究科物理学専攻修士課程  
修了。同年(株)日立製作所入社。  
以来, 中央研究所にて空間情報シ

ステムの研究に従事。電子情報通信学会員。



谷崎 正明(正会員)

平成 5 年神戸大学工学部計測工学  
科卒業。平成 7 年同大学大学院工学  
研究科計測工学専攻修士課程修了。  
同年(株)日立製作所入社。中央研  
究所にて空間情報システム, モバイ  
ルデータベース等の研究開発に従事。現在, 日立製作  
所主任研究員。



嶋田 茂(正会員)

昭和 50 年名古屋工業大学大学院  
生産機械修士課程修了。昭和 50 年  
(株)日立製作所入社。以来, 中央  
研究所にて, パターン認識による地  
図・図面入力システム, 空間情報シ  
ステムの研究に従事。平成 9 年東京大学大学院工学系  
研究科より学位取得, 博士(工学)。平成 19 年より首  
都大学東京産業技術大学院大学教授。電子情報通信学  
会, IEEE, ACM 各会員。



森岡 道雄(正会員)

昭和 57 年九州大学工学部電子工  
学科卒業。昭和 59 年同大学大学院  
修士課程修了。同年(株)日立製作  
所入社。日立研究所にて計算機ハー  
ドウェアアーキテクチャおよびナビ  
ゲーションの研究に従事。平成 16 年九州大学工学博  
士。平成 19 年より同社情報・通信グループ経営戦略  
室新事業インキュベーション本部新事業推進部担当部  
長。平成 2 年カーネギーメロン大学客員研究員。平成  
6 年 IEEE ICCD 論文賞。