

自動車のリアルタイム位置管理方式の一検討

松倉龍之介^{†1} 村田嘉利^{†1} 佐藤永欣^{†1} 鈴木彰真^{†1}

概要：近年自動車については、全自動運転車と共に Connected car の研究開発が多く研究機関で行われている。また、その実現に向けて接続遅延を大幅に短縮する第5世代移動通信方式の研究開発も急ピッチで進められている。その結果として、全ての自動車の位置をコンピュータ上で一元管理できる環境が整いつつある。船舶や航空機については、位置や移動速度等の情報をサーバ上で準リアルタイムに一元管理することにより衝突回避や運航管理に利用するサービスが提供されている。しかしながら、船舶や飛行機と異なり自動車の数は極めて膨大であることから1つのサーバで管理するのは難しく、自動車を分散して管理する必要がある。さらに、衝突回避のためには、自車両周辺の自動車をリアルタイムに検索できなければならない。本論文では、自動車の移動にリアルタイムに追従して自車両周辺にある車両を検索できるリアルタイム分散データベースマネージメントシステムについて検討した結果について報告する。

キーワード：自動車，位置管理，衝突回避，分散管理

1. はじめに

近年自動車については、人間が運転操作を行わずに走行できる全自動運転車と共に、インターネット通信機能を付与しデータの収集や分析を行う Connected car[1]の研究開発が多く研究機関で行われている。また、その実現に向けて接続遅延を大幅に短縮する第5世代移動通信方式[2]の研究開発も急ピッチで進められている。その結果として、全ての自動車がインターネットに接続され位置や計器の故障等がコンピュータ上で管理できる環境が整いつつある。

船舶や航空機については、Flightradar24[3]やMarineTraffic[4]のように位置や移動速度、移動先等の情報を準リアルタイムに一元管理するシステムがある。航空機や船舶同士の衝突回避や運航管理、地図上への表示などのサービスが提供されている。

その一方で、船舶(全世界で12万隻)、航空機(同約3万機)と異なり自動車の数は全世界で13億台と極めて膨大であることから、位置情報のリアルタイムな一元管理の実現は難しい。全世界の車両データ(本論文では以後トラフィックデータとする)を1つのサーバで管理するのは、規模が大きくなりすぎて不可能である。すなわち、自動車を対象にすると膨大な台数を扱う必要があり集中的に管理するのは困難であり、分散して管理する必要がある。しかし、分散して管理した場合、サーバ同士の通信などが必要となり、リアルタイム性に乏しくなることが考えられる。

衝突回避や周辺車両の情報の取得といった、トラフィック管理アプリケーションによっては、自車両周辺の車両情報だけあればよい場合、必要なデータの局所性が高い。したがってアプリケーションとデータの分散管理の方法を慎重に設計すれば、自動車の台数が膨大であるにもかかわらず、十分に実用的なトラフィック管理システムを作れると考えられる。我々は、トラフィック管理システムのアプリケー

ションを、衝突回避、自車両周辺の危険な運転をする車両の発見、渋滞検出・予測、犯罪現場などを特定の時刻に通行した車両の割り出しなどと想定し、自動車トラフィックをどのように分散管理するのがよいか検討を進めてきた。

衝突回避や自車両周辺の危険な車両の発見のためには、自車両周辺の自動車をリアルタイムに入手できる必要がある。渋滞の検出や予測のためには、自車両周辺、または特定の地点の周辺の車両を準リアルタイムに入手できる必要がある。一方、犯罪現場など、特定の地点と特定の時刻に通行した全車両の洗い出しにはリアルタイム性は必要ない。

本論文では、トラフィックデータの数に応じて管理サーバを複数設けトラフィックデータを効率的に分散管理する方法と、自動車の移動にリアルタイムに追従して自車両周辺にある車両を検索する方法の2つの機能を備えたリアルタイム分散データベースマネージメントシステムについて検討する。また本論文では提案するトラフィックデータの管理手法の適応先として、車両同士の衝突回避を目的とした周辺車両の抽出を想定する。

2. 関連研究

関連研究として、航空機におけるリアルタイムな位置表示システムである Flightradar24 と MarineTraffic について述べる。

Flightradar24 は、2006年にスウェーデンの航空ファンの2人が航空機の空中衝突防止に使用されている識別子や現在位置、高度などの情報を含んだ信号である ADS-B を受信し航空機の現在位置をリアルタイム表示するシステムである。ユーザは誰でも受信した ADS-B 信号をシステムのサーバに送信することができるため、世界各国の航空ファンによってネットワークが拡大し世界中の航空機の情報を取得することができる。ADS-B はもともと、航空機の空中

^{†1} 岩手県立大学

衝突防止装置が使用する信号である。航空機は速度、方向、高度などを含んだビーコン信号を 1090MHz で送信しており、他の航空機の動向も 1090MHz で送信されてくる信号を受信することで知ることが可能である。空中衝突防止装置は他の航空機と衝突する危険があると判断すると、受信したビーコンの情報に基づき、パイロットに旋回・上昇・下降などの回避操作を指示する。

MarineTraffic は、2007 年に公開されたプロジェクトで Frightradar24 と同様に船舶の識別に使用されている識別子や現在位置、航路などの情報を含んだ信号である AIS (Automatic Identification System)の信号を受信し航空機の現在位置をリアルタイム表示するシステムである。AIS は、2002 年にテロリズムへの対処を目的として、国際海事機関の主導により船舶に導入された国際 VHF を利用した船舶を自動識別する装置である。船舶は 161MHz か 162MHz で速度、方向、航路などを含んだ海上移動業務識別コードを無線で送信しており、他の船舶の動向も 161MHz か 162MHz で送信されている信号を AIS で受信することにより知ることが可能である。

これらのシステムでは元々衝突防止用に航空機や船舶からビーコンとして送信されている信号を利用している。ところが、陸上交通には衝突防止用のビーコン信号は今のところ利用されていない。また、ビーコン信号が利用されたとしても受信可能な範囲が狭い、市街地などではビーコンが到達しない範囲があるといった問題が考えられる。このため、本論文のシステムは車両の車両がインターネット経由でトラフィックデータをサーバに送信し、車両周辺の他車両をサーバ側、または車両側で抽出するというものになっている。

3. トラフィックデータサイズと自動車トラフィック管理システムの実現方法の検討

自動車用トラフィック管理システムを実現するには、まず、全世界の自動車 1 台当たりのデータ量を想定し、1 日においてサーバに送信されるトラフィックデータの総量を想定する必要がある。データベースサーバの用途や機能により送信すべきトラフィックデータの要素に違いが生じる可能性があることにも留意が必要である。本論文ではトラフィックデータの基礎要素として、「車両固有の車両 id、車両位置の情報、トラフィックデータ送信時の時刻、移動速度、移動方向」を想定する。車両 ID、時刻が 64 ビット、緯度、経度、速度、移動方向が各 32 ビットとして、すべてのトラフィックデータをバイナリで表現してもデータだけで約 30Byte と想定できる。日本の自動車の交通量はピーク時の 7 時から 19 時の間で約 1 億が走行しており、日本の全自動車が 1 秒毎にサーバに送信した場合 7 時から 19 時の間でトラフィックデータの総量はオーバーヘッドを考慮

しなくても約 129.6TByte と想定される。また全世界のトラフィックデータを管理する場合、全世界の自動車保有数は約 13 億台であるため全車両が同時に走行し 1 秒毎にトラフィックデータを送信すると 1 日で約 3369TByte と想定される。

想定されるトラフィックデータの総量から 1 つのサーバで管理するためには、サーバの通信回線の容量が明らかに足りない。また、通信回線の容量が足りたとしても、1 台のサーバで取り扱うのはかなり困難な量である。サーバを分割したとしても、データそのものの性質や更新頻度から、従来の一般的なリレーショナルデータベースで管理するのは不向きであると思われる。当然、1 日全てのトラフィックデータを単純にすべて保存するのは、容量の問題からもデータの再利用性の問題からも難しいと考えられる。

このように、自動車交通のトラフィックデータを集中型システムで管理するのは難しい。さらに、効率的に管理するためには新しいモデルが必要と考えられる。そのためこの論文では、全世界のトラフィックデータを分散し短時間で抽出可能な管理手法を提案する。

4. トラフィックデータの分散方法

4.1 分散方法の検討

ここではトラフィックデータの分散方法について述べる。

トラフィックデータを分散させるためにはサーバをエリアごとに分割し、車両の走行地域ごとの担当サーバを決めるのが効果的な方法である。サーバを複数台用意する場合でも車両 ID などを使って分割する方法、すべてのサーバに同じデータを持たせる方法も考えられるが、自動車の移動速度や普段走行する地域が限られることを考慮すると、走行地域によるエリア分割が効果的と考えられる。エリア分割であれば周辺車両の抽出が非常に容易であるという利点がある。

走行地域に基づいて分割する場合でも、都道府県・市町村単位に分化する方法、緯度経度などに基づきグリッド上に分割する方法が考えられる。都道府県・市町村単位で分割する場合、川や山が都道府県・市町村境になっていることがおおいいため自動車がエリアをまたぐ可能性は低くなる。本論文ではトラフィックデータの分散方法として、緯度経度ベースでグリッド状にエリア分割して各エリアにサーバを設置する方法を提案する。

この方法を選択した理由として、エリア分割のサイズ調整が比較的容易で柔軟と考えられること、エリア分割なので周辺車両の抽出が容易であり、衝突防止システムなどの実現が容易と思われる点がある。自治体単位で分割すると、車両が非常に多い自治体、広大なエリアを持つが過疎化が進んでいて車両台数が少ない自治体があり、自治体単位でのみの分割ではサーバへの負荷が偏ると考えられる。

緯度経度ベースで分割する方法として緯度経度そのものを使う方法、緯度経度ベースの地点コードを用いる方法が考えられる。緯度経度ベースの地点コードには、UTM 座標、MGRS[5]、MAPCODE[6]などがある。本研究では MGRS を使用したメッシュ分割を行う。

MGRS (Military Grid Reference System)は米軍発祥のグリッドコードであり全世界の位置を最も詳細な場合には 1m 単位で示することができる。MGRS の特徴として、コードの桁数によりグリッドのサイズと地点コードの詳細さを表現することができ、緯度経度との変換が簡単であることがあげられる。MGRS は極地以外では UTM 図法、極地では UPS 図法に基づくため、極地と極地以外では異なるコード体系を持っている。以下では極地以外の MGRS コードについて述べる。

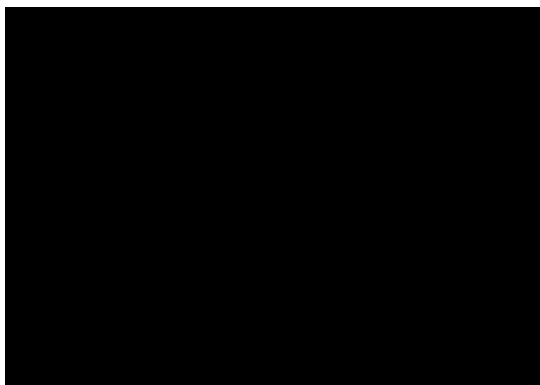


図 1 MGRS コードと構成要素

MGRS コードは 5 つのブロックに分かれている。ここでは「54SWK1169005902」という MGRS コードを例として説明する。このコードは北緯 39 度 48 分 10 秒、東経 141 度 8 分 11 秒にある、岩手県立大学の研究室を表している。先頭の 54 は UTM ゾーン番号を表す。UTM ゾーンは経度を 6 度ごとに 60 分割した数字で示されているコードである。S は緯度バンドであり、南緯 80 度から北緯 84 度まで 8 度ごとに 20 分割したアルファベットで示されているコードである。WK は 100 キロメートル格子 ID と呼ばれ、緯度バンドと UTM ゾーンで特定された領域内を、100 キロメートル四方の格子に区切り、アルファベット 2 文字で格子を特定する。アルファベット一文字目は経度方向を示し二文字目は緯度方向を示している。残りの数字列は 100 キロメートル四方の格子内の座標を表す。この数字列は二つの要素からなり、X 座標(Easting)が 11690、Y 座標(Northing)が 05902 である。これらの座標の単位はメートルであり、100 キロメートル格子が示す範囲の南西端を原点として示される。この数字列は最大 10 桁、最小 2 桁で表現され、MGRS でコードされた座標の精度を表現している。すなわち、10 桁の場合は 1 メートル単位の精度、2 桁の場合は 10 キロメートル単位の精度である。X 座標と Y 座標の精度は同じである必要がある。

図 2 に MGRS を適用した管理対象域の分割例を示す。この場合、Easting と Northing がそれぞれ 2 桁ずつ計 4 桁であるため 1000 メートル四方のメッシュである。

54sue0200	54sue0201	54sue0202
54sue0100	54sue0101	54sue0102
54sue0000	54sue0001	54sue0002

図 2 MGRS を適用した管理対象域の分割例

4.2 エリア分割のサイズ

管理対象域のエリアサイズを決定する手法として、分割数に応じた複数の項目における性能を選択し最適なエリアサイズを決定していく手法を提案する。今回は、「サーバがメモリ上で管理するトラフィックデータ」、「車両周辺の他車両抽出にかかる時間」について性能を選択しエリアサイズを決定する。なお、本論文では管理対象域を日本全国とし、トラフィックデータのサイズを 30Byte、交通車両数は 1 時間で 1000 万台と想定してエリア分割を行う。

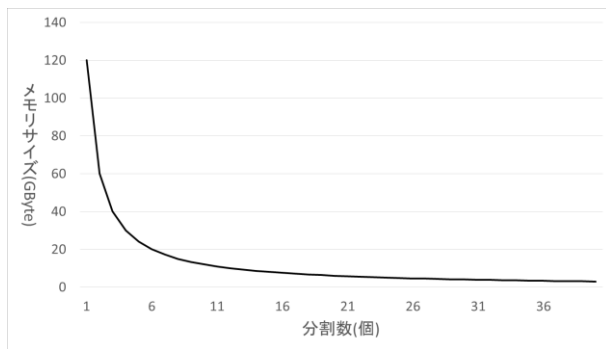
初めに、管理対象域における総トラフィックデータサイズを算出する。総トラフィックデータサイズである B[Byte] は式(1)により算出する。このとき、T[Byte]はトラフィックデータのサイズ、C[台/h]は交通台数、 t_1 [秒]はトラフィックデータの受信間隔、 t_2 [秒]はトラフィックデータの保持時間を表す。

$$B = \frac{TCt_2}{t_1} \quad (1)$$

次に、サーバが 1 時間メモリ上で管理するトラフィックデータ D[Byte]を図 3 に示す。今回はトラフィックデータの受信間隔を 1 秒、保持時間を 10 分とする。

分割数 n におけるサーバがメモリ上で管理するトラフィックデータである D(n)[Byte]は式(2)により算出する。このとき、B[Byte]は式(1)で算出した総トラフィックデータサイズを表す。

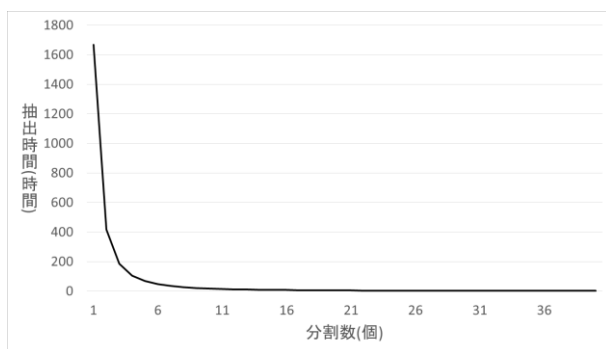
$$D(n) = \frac{B}{n} \quad (2)$$



車両周辺の他車両の最大抽出時間 $t[\text{sec}]$ を図 4 に示す。
今回はトラフィックデータ 1 個の抽出時間を 1 ミリ秒とする。

分割数 n における抽出時間 $t(n)[\text{sec}]$ は式(3)により算出する. このとき, $C[\text{台/h}]$ は交通台数, $s[\text{sec}]$ はトラフィックデータ 1 個の抽出時間を表す.

$$t(n) = \frac{Ds}{n} \quad (3)$$



これら 2 つの式より最適なエリアサイズを決定する。今回はサーバのメモリサイズを 8GByte と想定し、式(2)より管理対象域を 15 分割することで最適なエリア分割を行うことができる。

5. トラフィックデータの管理方法

次に各エリアに対応した管理サーバにおけるトラフィックデータの管理方法について述べる。従来のリレーショナルデータベースを用いてトラフィックデータを管理すると、トラフィックデータの更新や周辺車両のトラフィックデータ抽出に時間がかかりリアルタイム性が損なわれてしまう。そのため本論文では、リングバッファを用いて最新のトラフィックデータをメインメモリ上で管理し、古くなったデータを保存用にハードディスクなどのより低速な媒体に書き出す方法を提案する。このようにすることで、新しく受

けとったトラフィックデータの処理時間を短縮し、古くなってあまり使われなくなったデータを保管することができる。

5.2 リングバッファを用いたトラフィックデータの管理

車両 id	位置情報 (mgrs)	時間 (hh:mm:ss)
----------	----------------	------------------

図 5 リングバッファを用いたメモリの例

サーバが車両から送信されたトラフィックデータを受信したとき、最新データの次に受信したデータを挿入する。データが既にあった場合には上書きする。これによりあるエリアに一定期間存在していた車両のトラフィックデータが保存される。この一定期間はトラフィックデータの数、すなわち走行中の車両の台数とリングバッファのサイズで決まる。したがって、車両の台数がピークになる時間帯に、どの程度の期間のデータを保存する必要があるかに基づいてリングバッファのサイズを決める。

車両 id	位置情報 (mgrs)	時間 (hh:mm:ss)
車1	54sue0000000000	07:00:00

HDD

図 6 過去トラフィックデータの書き出し

6. 周辺車両の抽出方法

6.1 車両周辺他車両抽出

次に周辺車両の抽出方法について提案していく。周辺車両の抽出は MGRS の特性を利用して行う。例えば、車両同士の衝突回避を目的としたとき、周囲 10 メートルに存在する他車両の抽出を行うことで危険通知が可能になると考えられる。図 7 に他車両の抽出方法についての図を示す。

車両 id	位置情報 (mgrs)	時間 (hh:mm:ss)
車1	54sue0000000000	07:00:00
車2	54sue0000100001	07:00:00
車2	54sue0000100002	07:00:01
車1	54sue0000100000	07:00:01
車2	54sue0000100003	07:00:02

図 7 抽出対象車両トラフィックデータと抽出された他車両トラフィックデータ

抽出対象車両の MGRS コードが「54SUE0000100001」であるとき 10 メートル周辺の他車両を抽出するためには X 座標と Y 座標の桁数に注目する。10 メートル周辺のエリアは X, Y 座標の 1 桁目から 4 桁目が同じコードであるトラフィックデータを検索することで抽出することができる。つまり「54SUE0000X0000Y」のコードであるトラフィックデータが車 n の周辺にある他車両ということになり、図 7 の白文字のトラフィックデータが周辺に存在していた車両

ということになる。

6.2 現実世界との位置ずれ補完

走行中の車両が周辺車両を抽出するとき、抽出してきたトラフィックデータは過去のデータであるため現実世界と管理サーバ上の車両の位置にずれが生じる場合がある。



図 8 データベース上の位置と現実世界の位置ずれ

この位置ずれを解消するために、メモリ上で過去のトラフィックデータを保持し移動速度と方向から現実世界の車両位置を補完する。ある車両の過去トラフィックデータはサーバのメモリに格納されているため、周辺車両の抽出を行ったときに過去トラフィックデータの位置情報や移動速度、移動方向などを用いて現実世界の位置を推定することが可能になると考えられる。位置ずれの解消方法を図 9 に示す。

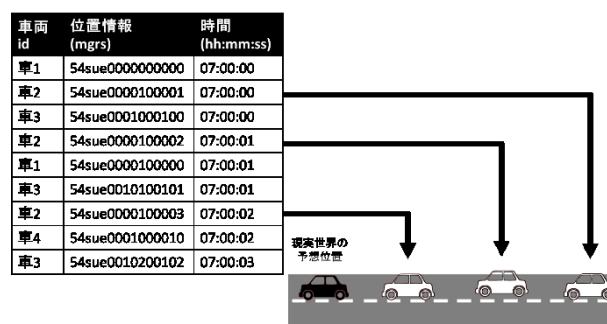


図 9 現在の車両位置の予測

図 9 では車 2 の過去データに基づき現在位置の予想を行っている。メモリには 3 つ分のトラフィックデータが格納されており、過去のコードから移動方向を予測することができる。また位置情報の他に移動速度や移動方向が加わることでより更に高精度な位置予想が可能になると考えられる。

6.3 サーバの管理エリアの分割

前項まで提案してきた抽出方法を適応したとき、4.2 項で示した式(3)より、15 分割の場合抽出にかかる最大時間は約 7 時間と想定さる。抽出にかかる時間の想定として、衝突回避を目的とした周辺車両の抽出を考慮すると 1 秒程度での抽出が望まれる。そのため更にリアルタイム性を高める手法を考案する必要がある。

想定される方法として、管理エリアを更に分割する方法が挙げられ、本論文の 5 項で提案した 1 つのサーバに 1 つ

のリングバッファではなく 1 つのサーバを複数のリングバッファで管理する方法を提案する。図 10 に管理エリアを複数のゾーンに分割しそれぞれにリングバッファを割り当てた例を示す。

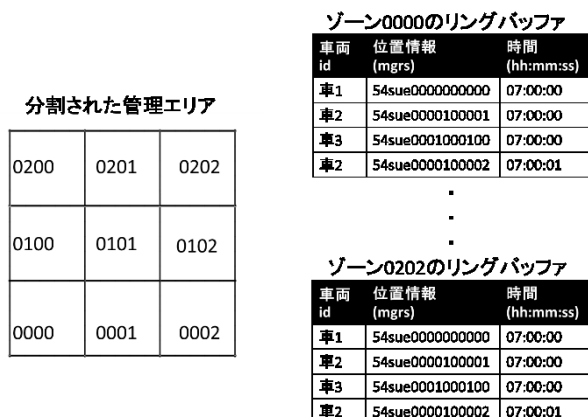


図 10 分割された管理エリアと
割り当てられたリングバッファ

複数のリングバッファでトラフィックデータを管理する場合、検索範囲が複数のリングバッファに跨る可能性が考慮されるため、図 11 のような抽出方法が考案される。

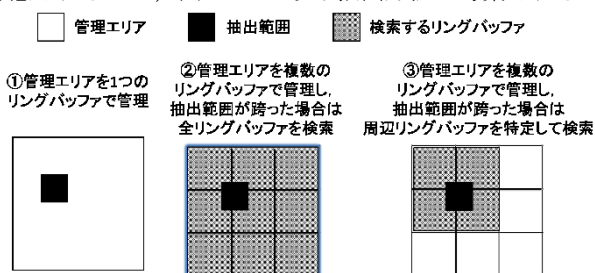


図 11 トラフィックデータの抽出方法

方式①は、前項までに述べていた管理方法で最大抽出時間は約 7 時間である。

方法②は、管理サーバを複数のリングバッファで管理し、抽出範囲が複数のリングバッファに跨ったときは車両が所属するリングバッファの周辺 8 つのリングバッファを検索する方法である。この方法では、抽出範囲が跨らない場合は方法①の抽出時間よりもリングバッファの数だけ短縮されるが、複数のリングバッファに跨った場合は跨らない場合の 9 倍ほど時間がかかることが予想される。

方法③は、管理サーバを複数のリングバッファで管理し、抽出範囲が複数のリングバッファに跨ったときは車両 MGRS コードからの周辺のリングバッファを特定して検索する方法である。この方法では、抽出範囲が跨らない場合は方法②の抽出範囲が跨らない場合と同じ抽出時間になるが、複数のリングバッファに跨った場合は抽出すべきリングバッファの選別に時間がかかるが、方法②の抽出範囲が跨る場合よりも短くなると予想される。

7. 隣接サーバ間のデータやり取り

7.1 車両が隣接サーバの管理エリア移動時の処理

トラフィックデータを複数のサーバで分散管理するとき、図 12 に示す車両とサーバの関係になることが予想される。

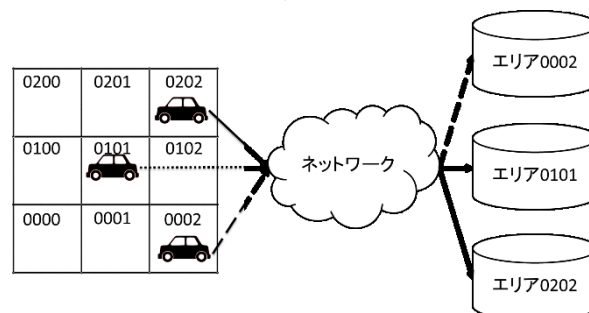


図 12 システム構成図

車両がトラフィックデータをサーバに送信するとき、GPS で取得した位置情報を MGRS のコードに変換する。これによりコードの X 座標と Y 座標の最初から 2 桁目を参照することで送信すべきサーバを特定することが可能になる。また、管理エリアから別の管理エリアに移動したときには、図 12 に示すようにサーバのネーミングされたコードと車両の MGRS コードと違いが生まれるため新しく送信すべきサーバを特定することができる。これにより、ゲートウェイサーバを設けずにトラフィックデータの振り分けを行うことが可能だと考えられる。

7.2 エリア境界線付近の車両抽出方法

全世界のトラフィックデータを複数のサーバで分散管理する場合に最も大きな問題点として、エリア境界線付近における周辺エリアの他車両抽出が難しいと考えられる。それに対して考えられる手段として 3 つ挙げられる。

ひとつ目は、セル状に分割し隣接サーバに問い合わせを行う方法である。この場合、隣接しているサーバに問い合わせをした後に各サーバで周辺車両のトラフィックデータ抽出を行う。トラフィックデータの抽出には 6.3 項で示したいずれかの抽出方法を使用する。まず初めに、抽出対象車両の MGRS コードから抽出範囲が複数のサーバに跨ることを判断し、どのサーバに跨っているのか判定する。そして判定を行った後に各サーバの抽出範囲の跨っているリングバッファからトラフィックデータの抽出を行う。

ふたつ目は、ある程度管理エリアを重複させる方法である。この場合、分割した 6.3 項で示した分割したゾーン 1 つ分を重複させて各エリアにサーバを配置する。検索方法を図 13 に示す。

エリアが重複しているゾーンに抽出対象車両が存在する場合、車両の MGRS コードからどのサーバに近いかを判定を行う。その後判別したサーバでトラフィックの抽出を行う。

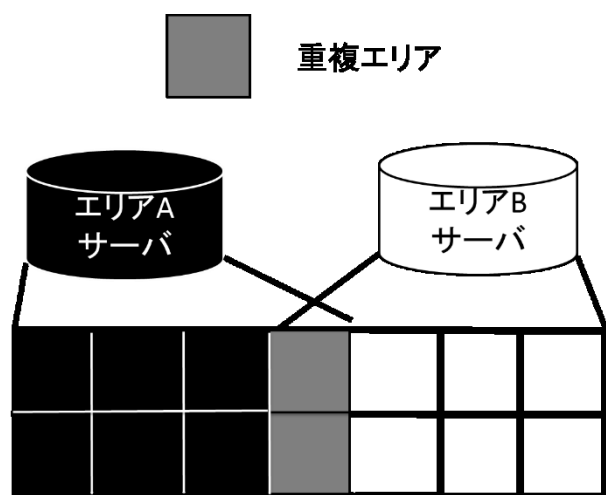


図 13 エリアを重複した場合のサーバ構成

3 つ目は、エリア境界線付近を管理するサーバを設ける方法である。この場合、エリア境界線付近に存在するトラフィックデータを格納するサーバを新たに追加する。サーバのネーミングは4.1 項で示したように MGRS コードを利用して行う。サーバ構成と検索方法を図 14 で示す。

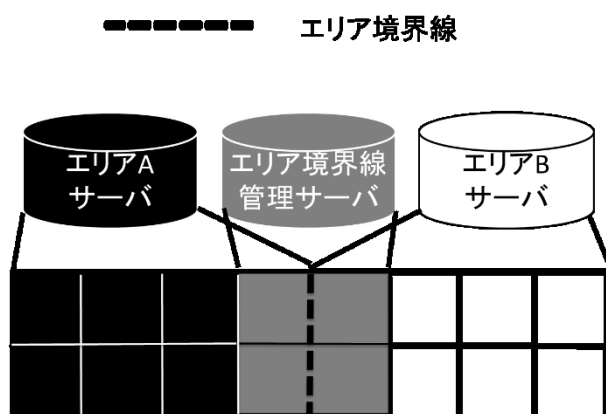


図 14 エリア境界線付近を管理するサーバを設置した場合のサーバ構成

この場合、エリア境界線付近に抽出対象車両が存在する抽出対象車両の MGRS コードから所属するエリア境界線付近を管理するサーバを特定し周辺車両の抽出を行う。

7.3 位置ずれ補完のために過去データを取得する方法

6.2 項で述べた現実世界との位置ずれの補完を別の管理対象エリアから移動直後の車両で行うとき、6.3 項で提案したいいずれかの抽出方法を用いることで過去データの抽出を行う必要がある。また、7.2 項で示したいいずれかの方法に適した車両の過去トラフィックデータを検索する方法を考案する必要がある。

8. おわりに

本論文では、自動車の移動にリアルタイムに追従して自車両周辺にある車両を検索できるリアルタイム分散データベースマネジメントシステムについて検討した結果について報告した。ここでは、主にエリアの分割とデータの分散方法、データの更新、管理の方法について述べた。今後は管理対象域の分割による適したトラフィックデータ問い合わせ方法と管理エリア内のトラフィックデータの管理方法について検討し、交通シミュレータと提案システムのプロトタイプを実装し、提案手法の有効性について検証していきたい。

参考文献

- [1] 総務省, 「Connected Car 社会の実現に向けた研究会」, http://www.soumu.go.jp/menu_news/s-news/01kiban14_02000279.html
- [2] 総務省, 「情報通信白書平成 28 年度 高度道路交通システムの推進」, <http://www.soumu.go.jp/johotsusintokei/whitepaper/ja/h28/html/na000000.html>
- [3] Flightradar24 AB, “Flight Tracker | Flightradar24 | Track Planes In Real-Time”, <http://www.flightradar24.com>
- [4] MarineTraffic.com, “MarineTraffic: Global Ship Tracking Intelligence | AIS Marine Traffic” , <https://www.marinetraffic.com>
- [5] National Geospatial-intelligence Agency, “DMA TECHNICAL MANUAL 8358.1”, <http://earth-info.nga.mil/GandG/publications/tm8358.1/tr83581b.html>
- [6] デンソー株式会社, “MAPCODE”, <https://www.denso-communications.jp/mapcode/>