

# 路線バス情報サービスを充実させるためのバス路線データベースの構築 Building a database for bus routes to enhance the bus service information

朴 太龍<sup>†</sup>      伊藤 昌毅<sup>†</sup>      川村 尚生<sup>†</sup>      菅原 一孔<sup>†</sup>  
Tailong Piao<sup>†</sup>      Masaki Ito<sup>†</sup>      Takao Kawamura<sup>†</sup>      Kazunori Sugahara<sup>†</sup>  
<sup>†</sup>鳥取大学大学院 工学研究科 情報エレクトロニクス専攻

## 1 はじめに

路線バスなどの公共交通機関は自家用車などを持たない人にとっては重要な移動手段である。しかし、路線バスは実際の利用上バスに関する情報が入手困難なので利便性が高いとはいえない。そこで、私たちは、公共交通の利用促進を目的として、公共交通機関利用援助システム「バスネット」[1]を開発・運用している。バスネットは、徒歩移動を考慮した乗り換え案内を行う経路探索機能、バスの遅れを考慮した経路探索機能、バス停ごとの時刻表検索機能、印刷用時刻表作成機能などを持っており、鳥取県全域に対応している。バスネットはパソコンや携帯電話端末、鳥取駅バスターミナル、鳥取県庁等に設置している専用端末「インテリジェントバス停」から利用できる。しかし、今のバスネットのサービスは基本サービスとしてはよいが、利便性が高いとはいえない。例えば、遅れを考慮した経路探索機能の遅れ時間の精度が不十分であり、案内は主に文字であるためその情報の理解が難しい。そのため本研究では、OpenStreetMap[2]とバスロケーションシステム[3]を用いて、バス路線情報のデータベースを構築して、以上の問題を解決する。

## 2 バス路線データを利用した路線バス情報サービスの充実

バスネットにはバスロケーションシステムというバスの現在位置をGPS機能を持ったスマートフォンを用いて1分ごとに送信させてサーバ側で取得するシステムがある。そして、その情報を用いて、地図上にバスの現在位置を表示したり、遅れ時間を算出して、バスの遅れを考慮して、検索結果を出すことができる。しかし、送られてきた位置情報とバス停の位置情報のみを用いてサービスを実現しているため、その精度や表現は十分ではない。

### 2.1 バスが地図から外れて見える

バスを待っているバスネットの利用者たちにバスの現在位置を知らせるため地図上にバスの現在位置を表示している。しかし、GPSの精度が不十分なこととバスのアイコンはバスロケーションシステムから取得した位置座標をそのまま補正なしで地図に表示するので、図1のようにバスが道から外れて見えることがある。

### 2.2 遅れの精度が高くない

路線バスは道路状況によって時刻表通りに運行できないため、バスの利用者はバスがいつ来るかわからな



図 1: GPS の精度が低い

い、遅れが原因で乗り換えるべきバスや電車に乗れないかもという不安要素が存在する。そのため、私達の研究室ではバスロケーションシステムでバスの遅れを算出して、遅れを考慮した経路探索機能を開発した。現時点でバスの遅れ算出方法は以下の通りである。

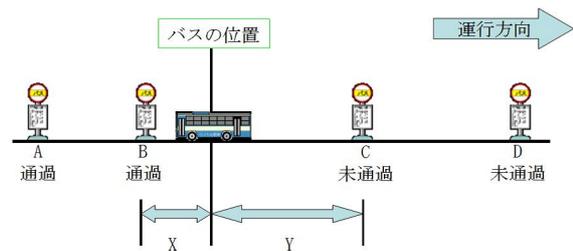


図 2: 遅れ算出

1. 最後に通ったバス停 B と次のバス停 C を探してバスの現在位置とバス停 B の距離を  $x$ 、バス停 C との距離を  $y$  とする。
2. 時刻表でバス停 B を通過時刻を  $t_x$ 、バス停 C を通過時刻を  $t_y$  とする。
3. バスが現在走行している区間における進んだ割合  $P$  を求める。

$$p = \frac{x}{x + y}$$

4. バスが現在走行中の場所の通過時刻  $t_p$  を求める.

$$t_p = t_x + (t_y - t_x)p$$

5. バスの遅れ (delay) を  $t_p$  と現在時刻で求める.

$$\text{delay} = t_p - \text{現在時刻}$$

この方法は遅れを算出するにはいい方法だが、正しい遅れ時間を算出するには前提条件がある。それは、ステップ A で説明した道路の長さ  $x$  と  $y$  が正確であることである。今のバスロケーションシステムでは、バスの現在位置とバス停 B、バス停 C との距離は直線距離を使っている。バス停 B とバス停 C の間には直線でつながる道であったら正しい結果が出るが、バス停の間はすべて直線の道でつながっているわけではないから、まわり道の場合は誤差が大きくなる。

そして、この方法で現時点の遅れ時間は算出できるが、これからどれくらい遅れるかは予測できない。遅れ予測ができないため、実際の遅れ時間の精度が不十分である。

### 2.3 情報案内サービスに親切さが足りない

バスネットの案内は主に文字であるためその情報の理解が難しい。バスロケーションシステムで、地図上には現在走行中のバスの位置を示しているが、どのような路線があり、走行中のバスがこれからどこへ進むのかという路線案内がないため、親切だとは言えない。

本稿では、バスネットにバス路線情報を追加することでより便利なサービスを実現できるようにする。バスが通る道路やバス停の正確な位置などのデータを整備し、それを利用することで前述の問題を解決する。

1. GPS の精度の問題は、バスから送られてきた位置情報を路線情報に合わせて補正することで解決できる。
2. バスの遅れ情報については、バス路線情報があれば、前述の 2.2 の  $x$  と  $y$  をもっと正確にし、遅れ時間計算の精度を上げることができる。路線情報を使えば、これからどれくらい遅れるかも予測できる。方法としては、これから通ろうとする区間を直前のバスが通過に要した時間を知ると、次のバスの通過時間がおおよそどれくらいになるか予測できる。この時間と時刻表での時間を比べると、これからどれくらい遅れるかが予測できる。
3. バスの現在位置に加えて、路線情報を地図上に表示すれば、路線とバスの関係がわかりやすくなり、より親切な情報が提供できる。

## 3 バス路線データベースの構築

本稿では前述の各種機能を実現し、編集機能にも対応できるバス路線データを作る方法について提案する。このようなバス路線データベースを構築するためには、以下の問題を解決する必要がある。

### 3.1 バス路線情報の収集方法の問題

バス路線データベースを作成するためには、バスが走る道路の位置や形状を詳しく測り、データ化しなくてはならず、手間がかかる。本研究では、OpenStreetMap から取得できる道路データとバスロケーションシステムから取得できるバスの現在位置情報ログを用いて、バス路線データベースを半自動的に構築する。

### 3.2 バス路線情報の形式や情報量の問題

地図上の表示にせよ、距離計算や遅れ予測にせよ、もちろんデータが多くて、情報も多いほうが精度が高くてよいが、データが多いほど処理にかかる時間が多くなる。そのため、どのようにバランスを取るかということと、どのようにそれぞれの処理に向けた形式のデータを用意するかということが問題となる。

バスロケーションシステムのバス現在位置の表示と地図に使うバス路線情報は地図にぴったり合わせる必要があるため、情報量が多いほうがよい。現在時刻までの遅れの算出にも正確な道路の長さを求めるため情報量が多いほうがよい。ただし、データが多すぎるとサーバ側のデータ処理が多くなるので、反応速度が落ちる。遅れ予測の時、必要なのはある区間をバスが通過するとき要した時間だけである、区間の中の細かいデータまで入ってくると計算量が増えて遅れ予測の計算が遅くなる。そのため、どうやって各方面に合うデータを作るかが問題である。

本研究では、データベースの構造を工夫し、用途に合わせて必要な情報だけを取り出せるようにする。

### 3.3 バス路線情報の編集の問題

路線情報は作った後に追加、編集、削除等の機能も作らなければならない。それで、データの構造は使用上便利だけでなく、編集にも対応しなければならない。そのためには、一つの路線を編集したとき他の路線には影響がないようにする条件と道路情報に変更があった場合、バス路線情報の方の編集ができるだけ少なくなるか、編集せずに済むようにする条件を満たさなければならない。

## 4 開発の現状

バス路線情報データベースを構築するには、まず、バスロケーションシステムのログからバスの位置情報を解析してデータベースに入れる。次に OpenStreetMap から道路情報をとってきて、処理後自分が作ったデータベースに入れる。最後に以上の二つの情報でバス路線情報データベースを構築する。

### 4.1 バスロケーションシステムのログ解析

本研究ではバスの位置情報を把握するためにバスロケーションシステムを開発した。そして、私はバス路線情報を収集するために、バスロケーションシステムのログ解析システムを開発している。バスロケーションシステムにはバスの現在位置をスマートフォンを用いて 1 分ごとに送信させて取得し、ログファイルに書き込む機能がある。このログファイルを解析すること

でどのバスがいつ、どの道路を走ったかを知る事ができる。図3はバス路線情報を収集するため、バスバスロケーションシステムのログを解析し、データベースに収めて、OpenStreetMapに表示したものである。図には15:30発十六本松(下)八丁目行のバスの位置情報を1000個表示している。今後、OpenStreetMapの道路ネットワークデータをバスの位置情報を元に切り出しデータベースに登録することで、半自動的なバス路線情報データベースの構築を可能にする。

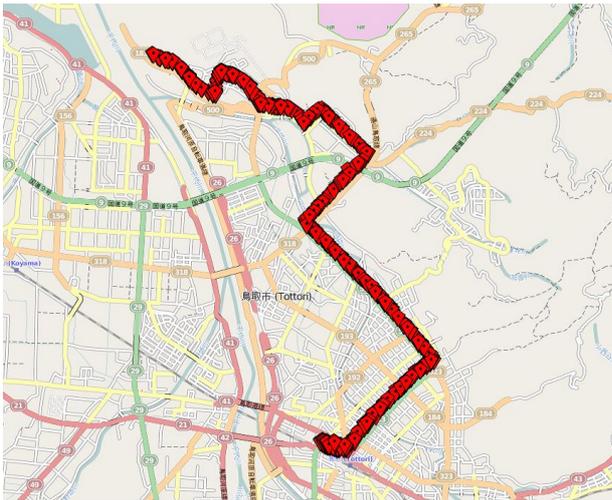


図 3: 十六本松(下)バス位置座標

#### 4.2 OpenStreetMap から道路情報を得る

OpenStreetMapは自由に利用でき、なおかつ編集機能のある世界地図を造るための共同作業プロジェクトである。バス路線情報データベースを構築するための道路情報はOpenStreetMapから取る。

OpenStreetMapには幾つかの要素があるが、ここでは基本要素のノード、ウェイ、リレーションを紹介する。ノード(node)は、OSM地図の基本的な要素である。ノードは緯度と経度から構成される、一つの地点を表したものである。ウェイ(way)は、2点以上かつ(API 0.6以降で)2000点以下のノード間を順序付きでつなぐ要素で、道路などの線状のものを表す。リレーション(relation)は、他の要素ノードやウェイ、さらには他のリレーションまでグループ化する要素である。図4はOpenStreetMapの内部様式である。この地図を構成するのはnode, way, relationである。内部様式ではnodeで位置情報を示し、nodeの集まりであるwayで道やエリアを表している。そして、relationは図には見えないが内部で道と他の道がどうやってつながっているかなどを示している。OpenStreetMapではnodeの間は直線で結んでいる。曲線は幾つかの一直線にないnodeを直線で結んだものである。

#### 4.3 バス路線データの形式

前述の問題点3.2と3.3を解決するため定義したバス路線データの形式を表すER図は図5の通りである。



図 4: JOSM 内部様式

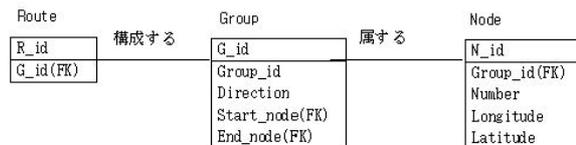


図 5: ER 図

1. Group テーブルは道路ネットワークをグラフとして表現した時、隣接ノードが二つではないノード(交差点や終点に相当する)の間をひとつの区間として Group\_id をつけて登録したものである。ひとつの区間の長さが 300m 以上になる場合は 2 つの区間に分ける。Start\_node は区間の始まりのノードの Node テーブルでの N\_id で、End\_node は区間の終わりのノードの Node テーブルでの N\_id で、Direction は Start\_node から End\_node への道か、それとも逆方向の道かを示すフラグである。方向の詳しい説明に関しては Node テーブルである。
2. Node には、区間の道路形状をさらに細かい点に分けて表現したものを、どの Group に属するかを示す Group\_id をつけて、順番を付けて入れたものである。その順番は南から北への順番である。もし、ちょうど区間の始点と終点と同じ緯度にある場合は西から東への順番に登録する。そして、Longitude と Latitude はノードの経度と緯度である。
3. Route は Group の G\_id の集合でできたバス路線情報である。

こうしてバス路線情報を二段階データ構造にすることで、問題点 3.1 と 3.3 は解決できる。

#### 1. 詳しい情報が取れる

バスロケーションシステムと地図に使う路線情報は地図にぴったり合わせるため、必要な情報は多くなる。こういう場合は Node テーブルから詳しい道路情報を取り出すことができるし、道路の長さも正確に計算できる。

#### 2. 遅れ算出に便利

現在時刻までの遅れ算出には正確な道路の長さが必要なのでデータが多いほどいいが、データが多すぎると計算時間が長くなる。それで、バランスを取らなければならない。Node テーブルには Number というフィールドがある。それを利用して、奇数/偶数番目のものだけを計算するとか二つ飛んで距離を計算するとかができる。

遅れ予測をするためには、区間の情報が必要である。バスの現在位置とバスの路線番号を知ること、すぐ Group テーブルの始点と終点を通じて、バスがどの区間にいるかが分かる。そこで、その区間を通った直前のバスが何分使ったかを見れば、遅れ予測も簡単にできる。こうすると、たくさんノードから、バスがどのノードの周りにいるかを調べるような時間のかかる計算をしなくて済む。

#### 3. バス路線情報の半自動的なつくり便利

バスの路線情報を探すとき重要なのはノードすべてを探すのではなく、バスがどの道を通るかである。それで、ここではノードを道路によってグループ分けして、Node テーブルに置き、Group テーブルにはどのグループであるかを示す Group\_id, 始点, 終点と方向を入れることで、バス路線情報を造るとき、たくさんノードを探さなくて済むようにした。

#### 4. バスの路線の編集

Route テーブルは Group の G\_id を外部キーとして持っているので、バスの路線を編集する時他のテーブルには影響がないし、細かいノードを気にしなくていい。

#### 5. 道路情報の編集

編集するノードが区間の始点と終点でない限り、Node テーブルを編集するだけでいい。

区間の始点と終点を編集する場合は編集される始点・終点の隣接する始点・終点の間のデータを再構築する。Group テーブルから元の G\_id が入っている項目を消した後、新しくできた区間情報を Group テーブルに挿入する。Route テーブルにも同じく、元の G\_id を消して新しくできた区間の G\_id を挿入する。もとの G\_id は捨てて、これから使わない。

#### 4.4 バス路線情報データベースの構築アルゴリズム

バスロケーションシステムのログにはバスの走行中の位置情報が記録されている。このログと Group テーブルを使って半自動的にバス路線情報をつくることができる。方法としては以下のとおりである。

1. 通過番号が始点番号であるバスの位置情報を探して、その辺りにある Start\_node を探し、各 Start\_node に対応する End\_node の周りでバスの位置情報があるかをチェックして、周りにバスの位置情報がある Start\_node と End\_node のペアで決まる区間の G\_id を Route テーブルに登録する。
2. 前述の End\_node を Start\_node とする区間の End\_node を探して、周りにバスの位置情報があるかをチェックして、バスの位置情報がある区間の G\_id を Route テーブルに登録する。ただし、ここでは前述のもとの Start\_node を除いて探す。
3. ステップ 2 で周りにバス路線位置情報がある End\_node があるかないかを判断して、あったらステップ 2 に移す。もし、なかったらステップ 4 に移す。
4. Start\_node の周りのバス位置情報の通過番号が終点番号かを判断する。もし、終点番号である場合は、これでバス路線探しは終わり、終点番号でない場合は、ステップ 2 で例外としたもとの Start\_node を End\_node として G\_id を Route テーブルに登録して、ステップ 2 に戻る。

#### 5 おわりに

本研究では路線バス情報サービスを充実させるためのバス路線データベースの構築手法を検討し、バス路線データの形式の定義やバスロケーションシステムのログ解析システムの開発を行った。今後、OpenStreetMap から道路の情報を切り出して処理し、前述のデータ形式でデータベースに書き込む。そして、バス路線名の登録やデータ提供用 API の開発をすすめ、システムの運用や改良を進めていく。

#### 謝辞

本研究の一部は、総務省戦略的情報通信研究開発推進制度 (SCOPE) の受託研究によって実施された。

#### 参考文献

- [1] バスネット. <http://www.ikisaki.jp/>.
- [2] Openstreetmap. <http://www.openstreetmap.org/>.
- [3] バスロケーションシステム.  
[http://www.ikisaki.jp/gps/bus\\_map\\_all.rhtml](http://www.ikisaki.jp/gps/bus_map_all.rhtml).