

## レイヤー概念を用いたダイクストラ法によるバス経路探索アルゴリズムの作成

八代高専土木建築工学科 学生会員 高木良太  
正会員 橋本淳也

### 1. はじめに

インターネットの急速な普及により、各交通機関の情報提供手段は一段と Web サイトを重視したものとなった。中でも Web サイト上で提供される電車の乗り換え案内サービスは大変便利なものである。一方で、路線バスの Web での情報提供量は不十分なままである。現在、バス事業者の中で乗り換え案内機能を有する Web サイトを運営しているのはごく一部の状況である。その要因は、乗り換え案内システムを構築する際の技術的問題である。バス路線網の停留所数および系統数は鉄道網のそれと比較して非常に多く、バス停が近接している場合は徒歩での移動（以下、徒歩連絡）も考慮が必要である。それゆえ目的地までのルートが無数に存在することになるため、経路選択の難しさや計算量の増加が大きな問題となる。これらの問題を解決するため、経路検索アルゴリズムとして、レイヤーの概念を取り入れた改良ダイクストラ法を提案する。この方法は、路線の乗り換え案内を動的に行う際に適した特徴を有しており、検索精度や品質、実行時間のそれぞれのバランスが取れている。さらに複数経路の探索も容易に行うことができる。

### 2. アルゴリズム

#### 2.1 作成上の問題点

バス路線網の停留所数は鉄道網のそれと比較して非常に多い。また、バスの運行ルートである系統の数が非常に多いため、これらの停留所（ノード）と系統（リンク）から構成されるネットワーク（図-1）は鉄道網に比べ複雑なものとなる。複数のリンクの合流や分岐、並走や交差は至る所で発生している。また、ノード同士の位置関係は大変に重要である。鉄道路線ではあまり見られない、停留所間を徒歩で接続するという手段も常に考慮しなければならない。これらの要因により、出発地から目的地までのルートは数多く存在することになる。この無数のルートの中から制約条件を満足する到達可能なルートを探し、適切な評価方法によるフィルタリングを行い、実用的で妥当なルートのみを取り出す必要がある。

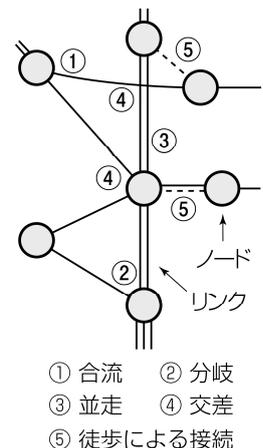


図-1 ノードとリンクの構成

#### 2.2 既存アルゴリズムの問題点

既存の経路探索アルゴリズムとしてダイクストラ法がある。これは2地点間の最短コストの経路を探し出す優れた方法であるのだが、1つの経路しか検索できない。反復計算を行うことで近似解を導く遺伝的アルゴリズム（GA）も存在する。これは複数経路の解が期待できるが、解の不確実性や計算時間が問題となる。負荷の軽さが求められる Web 上での使用を考えると GA の適用は不可である。

#### 2.3 アプローチ

本アルゴリズムは、確実に解が得られて計算時間が短いダイクストラ法をベースとする。ただし、最短コストの経路以外を導く方法が必要となる。第2経路を求める方法に、得られた最短経路を途中の分岐点で枝切りして再度ダイクストラ法を適用するという方法があるが、複数解を得るには何度も処理を実行する必要がある。また、乗り換え案内を行う場合、利用者の都合により乗り換え回数をなるべく少なくするように望まれる場合がある。このとき、乗り換え回数を制限する必要があるが、この条件下で上記の方法を適用するのは困難である。これらの問題を解決するため、本アルゴリズムでは、バス路線ネットワークを3次元へ拡張し、レイヤーを用いることでこれらの問題に対処した。

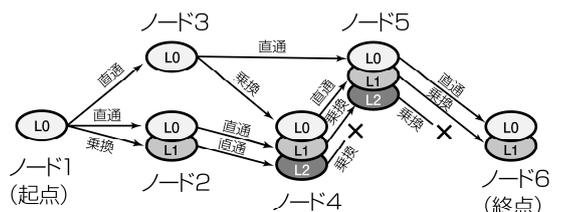


図-2 レイヤーを用いたダイクストラ法の模式図

このとき、乗り換え回数を制限する必要があるが、この条件下で上記の方法を適用するのは困難である。これらの問題を解決するため、本アルゴリズムでは、バス路線ネットワークを3次元へ拡張し、レイヤーを用いることでこれらの問題に対処した。

## 2.4 レイヤーを用いたダイクストラ法

通常のダイクストラ法では、1つのノードは1つの状態しか保持しないことになっている。この制約を変更し、複数の状態をノードに持たせることにする(図-2)。これをレイヤー化して管理し、ネットワークの一部として探索時に利用することで効率的に複数経路を探ることが可能になる。また、この方法は乗り換え案内のように乗り換え回数や徒歩による連絡回数が制限された検索条件に対しても有効に機能する。

## 2.5 徒歩連絡の考慮

既存のアルゴリズムで徒歩連絡を考慮しているものもあるが、これらの多くは徒歩での移動が可能なリンクをあらかじめ設定している。本研究ではデータ作成の手間・見落としを考慮し、バス停の座標データからバス停間距離を算出し、一定の範囲(500m程度)にあるバス停への徒歩リンクを自動作成する。この徒歩リンクは距離から移動時間を推定し、いつでも自由に移動できるものとしている。また、座標データが用意されればバス停だけでなく施設での検索なども可能となる。

## 2.6 最適経路の選択

上記の改良ダイクストラ法をバス路線網に適用することで、目的地までの複数のルートを得ることができる。このとき、時刻表データは使用していない。このアルゴリズムで得るのは経路のみ、すなわち乗り換え地点のみである。そして、この乗り換え地点に適合するルートを系統データと時刻表データから抽出し、実際に利用可能なルートが完成する(図-3)。この過程を通すことで、あらゆる起点時刻におけるルートが得られ、かつ不利経路を容易にフィルタリングすることが可能になる。

## 2.7 メリット・デメリット

レイヤーを用いたダイクストラ法は、レイヤー管理されたネットワークに対してダイクストラ法の手順を適用することで、最短経路や第N経路を探索できる。そして、定められた乗り換え数や徒歩での接続回数の制限内で最良経路を探索できる。通常のダイクストラ法では、このような乗り換え案内問題にうまく適用できない。

しかし、通常のダイクストラ法と比較して計算量は増大するというデメリットもある。網目のような密度の高いネットワークでは多大な計算量になるだろう。しかし各ノードで、レイヤーが持つルートの評価、比較を行い、早期の枝切りを徹底させることで計算量の増加を最小限に抑えることが可能である。特に、バス路線網においては枝切りが行いやすく、通常のダイクストラ法を繰り返すよりも遙かに高速に実行可能である。

## 3. 検証

実際のバス路線網を利用して計算速度と検索結果の妥当性の検証を行う。ノードとなる停留所数は約1300、リンク数は約6000となっている。1時間程度で移動可能であり、主要な路線が多く存在する停留所間では0.5sec以下で全起点時刻の検索が可能であった。Webサービス上でも実用に耐える速度だと考えられる。また、検索結果の妥当性を検証するため、任意に設定した10区間に対して、案内業務社員と一般事務社員に回答をいただいた。社員の案内結果と本アルゴリズムで検索した結果を比較すると、案内業務社員と同等以上の検索結果であることがわかった。

また、徒歩連絡を導入することで実情に近い案内がされた。時間帯によっては、選択された乗車停留所からのバス運行が少ない場合がある。このとき、より多くのバスが運行されている最寄りの停留所を自動的に探し出し、その停留所までを徒歩連絡とすることで、より実用性の高いルートを案内することが可能となった。

## 4. おわりに

レイヤーを用いた改良ダイクストラ法を使用することで、効率的に複数経路の探索が可能になった。レイヤー概念を取り入れることで、限定された乗り換え回数内で良い結果を出すことができた。それに加えて、徒歩連絡によってより実態に近い案内ができた。また、実行速度においてもWeb上で使用できるほどの高速さを確保できた。

検索結果の検証にあたり、熊本バス株式会社の皆様にご協力をいただきました。ここに謝意を表します。

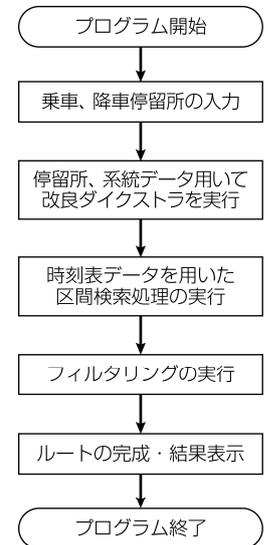


図-3 処理の流れ