

オンデマンドバスのためのリアルタイムスケジューリング アルゴリズムとシミュレーションによるその評価

2006年10月に道路運送法が改正され、定期定路線ではない乗合旅客運送に関する規制緩和が始まると、これまで認可が困難であったオンデマンドバスが目されるようになった。発生する乗客について時刻や経路を弾力的に設定しながら運送する新しいオンデマンドバスに対する高効率運行への期待は大きいですが、時間指定ができないことから時間制約のある移動の需要には対応できないという技術的問題がある。本稿では、従来とは異なった時間指定の出来るオンデマンドバスのアルゴリズム開発について述べ、シミュレーション実験によりその性能を評価した。

キーワード **オンデマンドバス, 時間的制約のある需要, シミュレーション, 改正道路運送法**

大和裕幸
YAMATO, Hiroyuki

工博 東京大学大学院新領域創成科学研究科人間環境学専攻教授

坪内孝太
TSUBOUCHI, Kota

修(環境) 東京大学大学院新領域創成科学研究科人間環境学専攻博士課程

稗方和夫
HIEKATA, Kazuo

修(工) 東京大学大学院工学系研究科環境海洋工学専攻助教

1—はじめに

2006年10月に道路運送法が改正され、定期定路線ではない乗合旅客運送に関する規制緩和がなされる¹⁾と、オンデマンドバスのように乗客の移動需要に合わせて弾力的に運行する交通機関が法制度的には可能となったが、技術的課題が残っている。

オンデマンドバスとは、タクシーにみられる個の需要に応じた運行機能とバスにみられる大量輸送の機能を併せ持った予約をして利用する新しい交通機関である。決められた時刻に決められた経路を運行する路線バスのように、乗客のいないバス停まで迂回をして移動が長くかかるという不便さは生じない。一方で、バスのように似通った移動の乗客があれば乗り合わせるといった効率的な運行が実現できる。

我が国でも数多くのオンデマンドバスが運行されている。初期の事例としては高知県四万十市や福島県南相馬市、最近では2007年3月1日から京都府福知山市におけるオンデマンドバスが挙げられる。

しかし、実用化されている事例の多くでは費やすコストの割に利用者が少ないといった問題がある。東北地方の自治体の例をあげると1回の移動あたり平均6,000円を超える補助をしている(初期投資の減価償却分は考慮していない)ケースもある²⁾。

そこで、本稿は現行のオンデマンドバスの調査を通じ、

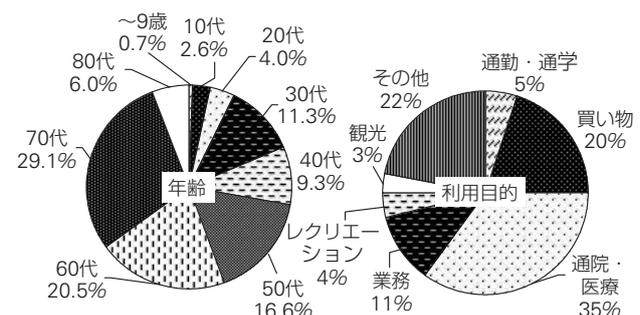
利用時間の指定ができないために需要の多い通勤などの利用に適さないという問題点に注目する。さらに、問題の解消に役立つ時間指定のできるスケジューリングアルゴリズムを開発し、千葉県柏市北部をモデルとしたシミュレーション実験によりその性能を検証することを目的とする。

2—現行のオンデマンドバス

2.1 現行のオンデマンドバスの利用実績

現行のオンデマンドバスの利用実績を調べると、主に高齢者や主婦の通院や買い物といった時間的な制約のない目的に利用されていることが分かる。オンデマンドバスが通勤や通学などに使われている事例は少ない。

図—1は高知県四万十市(旧:中村市)を走る「なかむらまちバス」と呼ばれるオンデマンドバスの利用者構成



■図—1 中村まちバス利用者構成

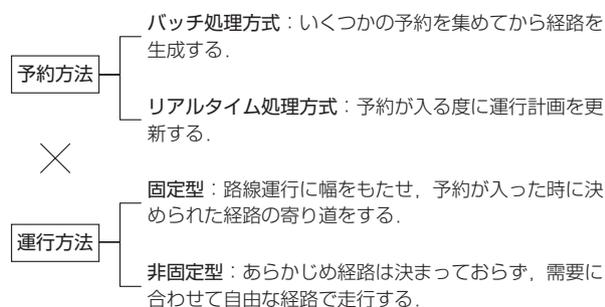
であるが、50歳以上が7割以上をしめ、主に買い物や通院、レクリエーションといった目的に利用されている³⁾。

2.2 現行のオンデマンドバスの問題点

ここでは現行のオンデマンドバスを分類し、分類された各タイプのオンデマンドバスにおける問題点について指摘する。

2.2.1 オンデマンドバスの分類

オンデマンドバスとは定期定路線ではなく乗客の需要に合わせて弾力的に移動する交通機関の事を指すが、走り方、車両サイズ、経路と時間の自由度、乗り換えの有無などの観点からいくつかに分けることが出来る⁴⁾。本稿では野田らの分類⁵⁾に従い、図一2に示す走り方と予約方法によって定義される4種類に分けた。



■図一2 オンデマンドバスの分類

2.2.2 各タイプの評価

現行のオンデマンドバスの問題点であった利用者の偏りという観点から各タイプを評価する。

まず、予約方法にバッチ処理を採用しているシステムでは、予約した時点ではまだ正確な運行経路や時間が決まっておらず、発着時刻の見積もりができない。そのため、通勤や通学といった時間的に制約のある移動に利用されるには不向きで、現行のオンデマンドバスに見られる利用状況が限界となる。

次に、時間・経路が固定され寄り道を行うタイプのオンデマンドバスについても同様のことが言える。寄り道の長さにもよるが、迂回にかかる時間が比較的大きい場合には、寄り道に予約が入るかどうかによって運行時間が大きく変化し、定時性の問題がでる。ただ、あらかじめ運行時刻に余裕をもっている場合は、ある程度の範囲で予想できるため、バッチ処理方式と比べると利用者の偏りは小さい。しかし、時間や経路が固定されている分、路線バスとの差分が小さくフレキシブルな移動が難しい。

すなわち、利用者に偏りを作らず、乗合を発生させながら移動するオンデマンドバスという点では、リアルタイム処理の予約方式を採用し、かつ経路や時刻が始めか

ら決まっていないタイプが理想的だといえる。

しかしながら、リアルタイム処理方式でも、あらかじめ入っている予約に対して新しい予約が入った場合、バスは最初の乗客が乗車した後に迂回をして新しい予約の出発地に立ち寄る必要がある。このときに予定していた移動距離よりも長くなるため、既に決まっていた運行計画に対し遅延が発生する。その結果、乗客に示した到着予定時刻を守れないことになる。本稿では、新しい予約が入ることで迂回が生じそれまでに予約した乗客の出発時刻および到着時刻に生じるオンデマンドバス特有の遅延を、渋滞や事故が原因で発生する一般的な遅延と区別する目的で「スケジューリングによる遅延」と呼称する。

スケジューリングによる遅延は迂回の数、すなわち乗客の数に比例して大きくなる。したがって多くの予約を追加しながらも、許容されない遅延を発生しないアルゴリズムが必要である。

3—スケジューリングアルゴリズムの開発⁶⁾

3.1 オンデマンドバスシステムに求められる要件

現行の調査によりオンデマンドバスシステムに求められる3つの要件が整理された。①リアルタイム処理を採用していること、②経路・時刻非固定型であること、③スケジューリングによる遅延を生じないことの3点である。ここでは、3点の要件を満たすスケジューリングアルゴリズムの開発について示す。

3.2 On-line Dial-A-Ride Problem

オンデマンドバスが解く問題は複数の乗客のデマンド情報(どこから・どこまで・何時に移動したいかという情報)から全員の要望を満たすバスの移動経路を作成する問題である。オペレーションズ・リサーチの分野ではDARP (Dial-A-Ride Problem)と呼ばれている。

一般的にDARP (Dial-A-Ride Problem)はNP-hardと証明されている⁷⁾。NP-hardとは計算複雑性理論における概念の一つで、計算の組合せが多すぎるためにコンピュータを利用しても大きな時間や膨大なメモリ量が必要となり、実時間内には最適解を求めることができない問題を指す。計算時間の限られる実用問題の場合はしばしば近似により準最適解を短時間で求めるケースが多い。

特に、本研究のようにリアルタイム処理を採用する場合、これまでに入った予約から作られた経路を一度分解し、新しい乗客の予約情報を加えた最適な運行経路を導出するという計算プロセスを予約が入る度に行う必要がある。これをOn-line DARPという。

On-line DARPで要求される計算時間は数十秒ある

いは数秒のオーダーであり、短い時間で解を求めるには何らかの制約条件を設ける必要がある。たとえば、Jawらの挿入法⁸⁾では予約時間の早い人から順番に予約を入れるという仮想的制約から解の存在範囲を絞っている。また、Liらの論文⁹⁾などに見られる制約条件としては、計算回数に制限を設け、その制限を超えてもなお解が見つからなければ解は存在しないと仮定する。

NP-hardである問題については問題規模についても考慮する必要がある。NP-hardな問題では問題規模の増加につれて爆発的に計算時間が増えるケースが見られるためである。本問題の場合、バスの台数や乗客数およびバス停の数が増えると問題は複雑になり、多くの計算時間を要する。特にバスが複数台ある場合には、各々のバスに新しい予約を挿入するスケジューリング問題を解くことになり、バスの台数倍の計算時間がかかるようになる。

バスの台数や乗客数、あるいはバス停の数に限界がある解法は汎用的な解法とはいえない。たとえば、Fabriらの論文¹⁰⁾にも見られる方法は、バスをランダムに選び最初に可能解が見つかるまで順に計算を行うとしており、すべてのバスに対して計算は行う必要はない。そのため、この方法ではバスの数が大きくなったときにも適用できるアルゴリズムといえるが、車両数や乗客数が多い場合には同じ方向に2台の車両で移動するといった非効率な可能解を選択してしまう問題点が指摘されている。

3.3 本研究で開発したスケジューリングアルゴリズム

以上の議論を踏まえて、本研究で開発したスケジューリングアルゴリズムについて述べる。まず、解の存在範囲を合理的に制限するために、ゆとり時間 (Slack Time) という特殊な Time Window を用いた制約条件を設ける。ゆとり時間とはバスの到着イベントに設ける時間幅の制約であり、あるバス停に時刻 t_A から時刻 t_B までの間にたとえ他に予約が追加されたとしても到着しなければならないといったものである。なお、本稿で用いる発着イベントは、バスが決められたバス停に到着して乗客を乗降させ、時間になったらバス停を出発するそれぞれのイベントのことを指す。

あるバス停へバスが到着する時刻はバス停の到着イベントに設けられたゆとり時間によって制御されることから、このゆとり時間を設定することで、スケジューリングによる遅延を防ぐことが出来る。

たとえば、9:00に到着を希望し、ゆとり時間が10分の到着イベントに対しては、8:50-9:00の間に到着するように制約が設定される。そうすると、10分の範囲で早く到着することはあっても9:00を過ぎることはない。ゆとり時間は一律に決定することも、予約一つ一つに違う

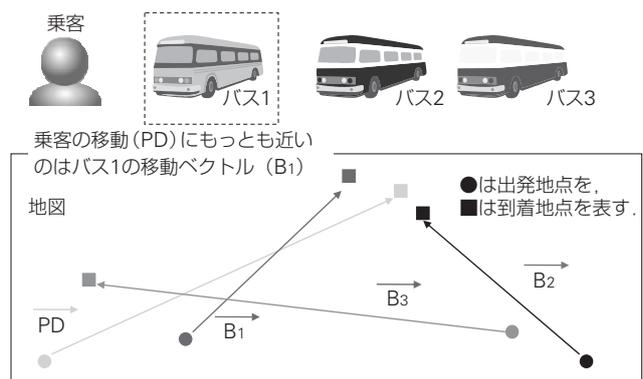
値を設定することもできる。

本稿では、このゆとり時間を導入することで、時間的制約のある需要にも対応できると考えた。Time Window を設定することで、約束した時間に対して指定したゆとり時間分だけ早着することはあっても、遅れは生じない。したがって、通勤・通学、電車への乗換のような時間的制約のある移動需要にも対応できる。

続いて、問題規模が大きくなった場合にも計算時間が爆発的に増えない計算方法が必要になる。特にバスの台数が問題規模に影響を及ぼし兼ねない。本研究では①新しい乗客を乗せるバスを決めて、②決めたバスについてスケジューリングするという二段階の計算アプローチをとる。この利点は、①バスの台数に応じて検索範囲が広がらない、②対象とするバスに入っている予約のみを考えれば良く、乗客数が増えてもバスの数だけ計算が逆に小さくなるという2点ある。以降、このアプローチに従い、より少ない計算時間で解を出すためにまず可能解の見つかる可能性が高いバスを選択する“バス選択アルゴリズム”，ゆとり時間を用いた“Time Window の設定方法”および選択したバスのみを対象としてDARPを解く“経路生成アルゴリズム”の3点について説明する。

3.3.1 バス選択アルゴリズム

バス選択アルゴリズムは対象とする予約をどのバスに配分するか決めるアルゴリズムである。具体的には、新しく入った予約情報のベクトルとその時間帯のバスの移動ベクトルを比較し、予約が成立する可能性の高いバスを選択するという方法である。図-3にはバス選択アルゴリズムの実例を示した。



■図-3 バス選択アルゴリズム図解

まず、対象とする予約の移動ベクトル \vec{PD} を以下のように定義する。

$$\vec{PD} = \vec{OD} - \vec{OP} \quad (1)$$

ここで \vec{OP} は出発地点の基準点からの位置ベクトルを、 \vec{OD} は到着地点の基準点からの位置ベクトルを示す。

次に、比較に用いるバス*i*の移動ベクトル \vec{B}_i を次のように表す。

$$\vec{B}_i = \vec{OG}_i - \vec{OS}_i \quad (2)$$

\vec{OS}_i は \vec{OP} と、 \vec{OG}_i は \vec{OD} と同時刻におけるバス*i*の位置ベクトルをそれぞれ表す。すなわち、 \vec{B}_i は、 \vec{PD} と同時刻のバス*i*の移動を切り抜いたベクトルである。

ここで、 \vec{B}_i と \vec{PD} がなす角 θ_i を式(3)から求め、その結果 $|\theta_i|$ の値が最も0に近いバスにおいて次の運行計画生成アルゴリズムを導入する。なお、 $|\theta_i|$ の最小値をとるバスが複数台ある場合には乱数により一方を選択する。

$$\theta_i = \arccos \left(\frac{\vec{PD} \cdot \vec{B}_i}{\|\vec{PD}\| \|\vec{B}_i\|} \right) \quad (3)$$

(ただし、 $-\frac{\pi}{2} \leq \theta_i \leq \frac{\pi}{2}$ とする。)

これにより、新しく入った予約のベクトルと最も角度が近いベクトルをもったバスが新しい予約を乗車させる候補として選択される。角度が近いことは、進行方向が類似していることをあらわし、最も運行計画が生成されやすいバスである可能性が高い。また、バス選択アルゴリズムに必要な計算量は各々のバスの移動ベクトルを求める程度の計算量であり、台数が増えても大きな計算時間を要しない。

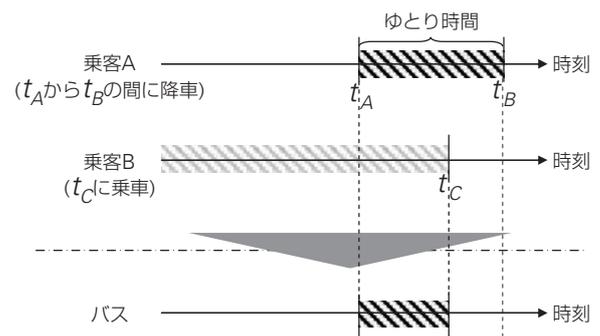
3.3.2 Time Windowの設定

本研究はゆとり時間という特殊なTime Windowを用いてスケジューリングによる遅延が生じないように工夫している。バスの発着イベントにおけるTime Windowはそのバス停で乗降する乗客のTime Windowを統合して設定される。その概要について図-4に示す。

各乗客に設定されるTime Windowは乗車と降車によって異なる。図-4の例では、同じバス停で乗客Aが降車をし、乗客Bが乗車をするケースを表している。まず、乗客Aには時刻 t_A から t_B というTime Windowが設定される。乗客が希望した到着時刻は t_B であり、したがって乗客Aはこのバス停にゆとり時間だけ早く着くことはあっても、希望した時刻よりも遅れることはない。次に乗客Bであるが、 t_C は予約をした際に通知した乗車時刻を表す。乗客Bの予約だけを考えるとバスは斜線のどこでも到着して良いが、少なくとも t_C を過ぎるまで出発してはいけないといった制約がつくことになる。

以上の要領で決まった乗客のTime Windowを総合してバスのTime Windowが設定される。すなわち、このバス停の発着イベントには「バスは t_A から t_C のうちに到着しなければならず、少なくとも t_C を過ぎるまで出発して

はいけない」といったTime Windowが制約条件として付与されることになる。



■図-4 Time Windowの設定について

3.3.3 経路生成アルゴリズム

経路生成アルゴリズムでは決められたバスに既に入っている予約を考慮して新しい予約をどこに挿入するかを決定する。アルゴリズムの概要を図-5に示す。

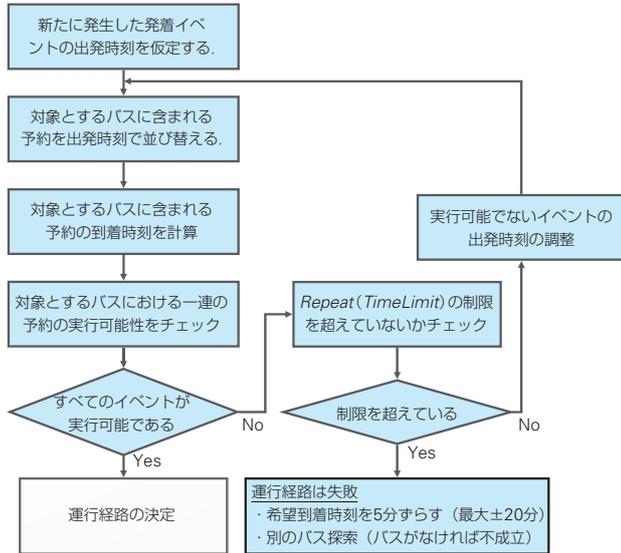
バスには既に何人かの予約が成立し、経路が生成されているとする。ここで新しい乗客が予約をし、この乗客の予約を挿入することを試みる。計算を始める前の状態として既に成立した予約で構成されている経路があり、各乗客の発着イベントにはそれぞれTime Windowが設定されている。経路生成アルゴリズムによって新しく予約をした乗客の実乗車時刻と実降車時刻とを決定する。

経路生成アルゴリズムは実乗車時刻の初期値を与えることから始まる。初期値は指定されたTime Windowの中で最も早い時間を乗車時刻として仮定し、挿入を試みる。たとえば、「地点Sを出発し、地点Tに10時に到着したい(ゆとり時間15分でST間は20分)」のような予約が入った場合、地点Tには9時45分から10時の間に到着するというTime Windowが設定され、最も早い場合は9時45分着となる。この場合、他に迂回をしない場合は地点Sを9時25分に出発すればよいと分かるため、実乗車時刻の初期値は9時25分と設定し、挿入を試みる。到着時刻によって指定する方法は前述の通りであるが、出発時刻で希望を指定する場合はこの逆の手順で考えればよい。挿入を試みたあと、生成された運行計画が実行可能かどうかをチェックし、実行可能の解が見つかった時点で計算は終了し、経路を決定とする。

ただし、スケジュールの実行可能性はバスの発着イベントが各バス停に定められたTime Windowの範囲に収まっているかにより判断する。

一方、実行可能でなければ実行可能でないイベントに対応する実乗車時刻を補正する。この修正は可能解が見つかるまで繰り返される。また、問題が大きくなったときの制約として指定された繰り返し探索回数Repeatを超え

た場合には実行可能な解が見つからなかったと仮定する。実行可能解が見つからなかった場合、まず新しい予約の希望時刻を5分ずつずらして再度検索を試みる。ただし、この変更は最大20分とする。それでも解が見つからない場合、移動ベクトルが次に近かったバスに対して同様の経路生成の計算を行う。また、これら一連の操作において指定された計算時間 $TimeLimit$ を過ぎた時点で解が見つからない場合は不成立として乗客に返す。



■図-5 経路生成アルゴリズムの概要

3.4 スケジューリングアルゴリズムのまとめ

オンデマンドバスシステムのために開発した①リアルタイム処理で予約をうけつけ、②経路・時刻が非固定で、③スケジューリングによる遅延のないアルゴリズムについて説明した。

まず、ゆとり時間を設定し解の存在範囲に制約を設け、スケジューリングによる遅延を発生しないように取り決めた。次に、バスを選択してから選択したバスのみを対象として経路生成を試みるアルゴリズムで、問題規模が大きくなった場合でも計算時間が爆発的に増加しないアルゴリズムを開発した。

4—シミュレーション実験による評価

4.1 シミュレーション実験について

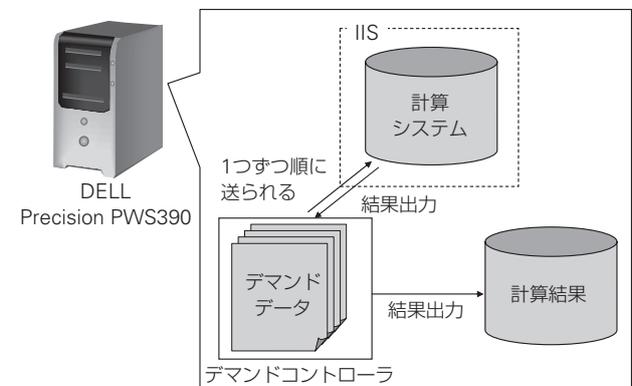
開発したアルゴリズムの性能をシミュレーション実験により評価した。ここでは、①シミュレーション実験の概要、②次に構築した顧客発生モデル、③実験で設定した変数の3点について説明する。

4.1.1 シミュレーション実験の概要

シミュレーション実験により仮想的に乗客を発生させ、

開発したスケジューリングアルゴリズムの性能を評価した。なお、本稿は開発したアルゴリズムの性能を評価することが目的であるため、予約の成立率および計算時間を評価指標とする。

シミュレーションは1台のPC (Intel® Core™2 CPU 6,600, メモリ2GB) 上で行う。Excel VBAにより順番に需要を発生させるプログラム(デマンドコントローラ)を作成し、内部の計算システムに予約情報を引き渡す。この計算システムには開発したスケジューリングアルゴリズムが実装されている。計算が終わると計算システムはスケジュール生成の成否にかかわらず、デマンドコントローラに計算が終了したことを示すシグナルを送る。それを受けたデマンドコントローラが次の予約を計算システムに引渡す。このように、連続的に仮想的な顧客を発生させて評価を行う。概要を図-6に示した。



■図-6 シミュレーションの概要

4.1.2 顧客発生モデル

シミュレーション実験のための顧客モデルを構築した。実際の都市をモデルに一定の顧客を発生させた。モデルとした都市は千葉県柏市北部地域で、路線バスの利用データを参考に顧客発生モデルを構築した。

まず、図-7に示す8つのバス停を対象区域に設置した。バス停を設置した箇所は駅や大学、病院、団地など特に移動人口が多くなる箇所を想定している。

顧客は、実際の柏市の路線バス利用調査に基づいて発生させた。路線バスの利用データからバス停毎の乗降客数は分かるが、乗客毎のOD(発着バス停の組合せ)は分かっていない。そのため、全体の利用データから時間帯毎に各バス停が発(着)地として選択される確率をそれぞれもとめ、求めた確率分布によって発生させた乗客の発着バス停を決定するという方法をとった。

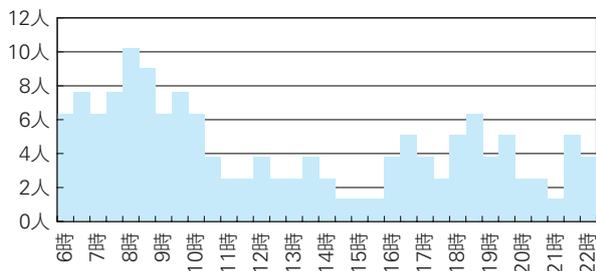
発生した顧客にはまず希望到着時刻が割り当てられる。その後、その到着時刻に応じた発着バス停の選択確率から発着バス停を決定する。希望到着時刻については、路線バスの利用データから図-8に示すような分布

を作成し、この分布になるように割り当てた。なお、発地・着地は設定した確率分布にしたがい、各々独立な事象として選択されるため実際の人の流れと異なっている。



■図一七 シミュレーション対象区域とバス停

次に、乗客数規模を中村まちバスの地域人口あたりの利用率(約0.05%)をシミュレーション実験の対象地域の人口に乗じることにより求めた。発生させる乗客規模は約160人となったが、図一八の顧客発生モデル分布と一致するように調整し160人に最も近くなる乗客数を求めたところ146人となった。



■図一八 顧客発生モデル(希望到着時刻の分布)

最後に、これらの予約が入る順番を決める。実際の利用シーンでは希望到着時刻の順序通りに予約が入ることは想定できない。146人に対し、予約を行う時刻を仮に決める。予約を行う時刻は希望到着時刻の4時間前から1時間前までの範囲でランダムに割り当てる。なお、実際の経路計算アルゴリズムには予約を行う時刻に「乗車の1時間前」のような制約は設けてはいない。乗車時刻が近い時は経路計算の結果からではなく、予約時刻が乗車時刻よりも後であるため乗車できないという不成立が生じる可能性がある。経路計算の結果ではなく、予約時刻の関係から発生する不成立はアルゴリズムの性能を評価する際に本質的ではないと判断し1時間前とした。最後に、予約を行う時刻の順序に並べ替え、最初から順に予約を発生させる。

4.1.3 シミュレーション実験で設定する変数

シミュレーション実験ではゆとり時間の長さおよび車両台数を変数として設定する。

ゆとり時間は5分、10分、15分、30分、直行移動時間の半分(TT/2と表記)の5通りで変化させる。直行移動時間とは発着地間を迂回せずに最短で移動した場合にかかる移動時間のことである。ただし、ゆとり時間の最小の値を5分とした。なお、このケースで与えた乗客146人の直行移動時間の平均は14.6分であり、TT/2における乗客のゆとり時間の平均は7.3分であった。

車両台数は1台から4台までを変化させた。車両の台数を変化させることにより、顧客の密度を変えることができる。シミュレーション実験では基本的に車両の定員は設けなかった。ただし、車両の大きさを検討するシミュレーション実験においては車両のサイズを変化させた。また、開発したアルゴリズムの性能を調べる目的のため、計算時間の上限TimeLimitは設けなかった。

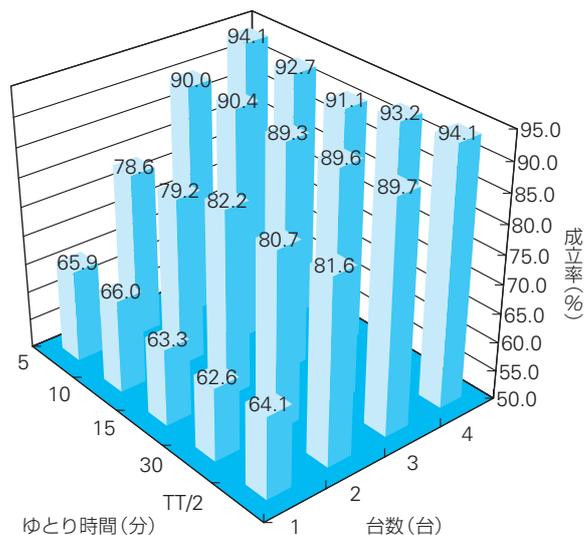
なお、本研究におけるバス選択アルゴリズムや顧客発生モデルには確率的な要素を含むため、それぞれの条件で5回のシミュレーションを繰り返し、その平均値を用いて結果の値とした。

4.2 シミュレーション実験の結果

ここでは、シミュレーション実験の結果を示す。

4.2.1 成立率の評価

本研究で示したアルゴリズムは乗客が希望する時間幅内で予約が成立しなければ予約を不成立とみなす。成立率とは不成立ではなかった予約の割合である。

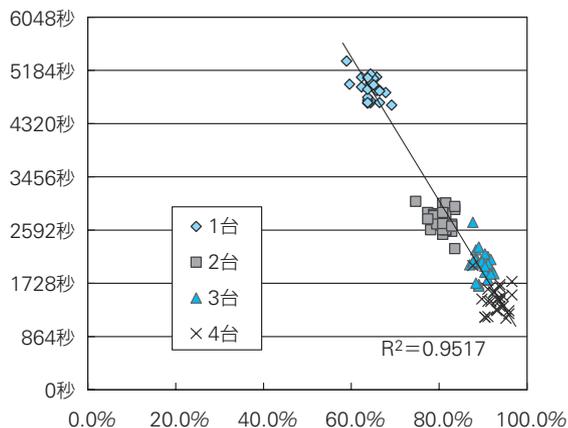


■図一九 成立率の評価

したがって、成立率が高ければより多くの乗客が乗車できたことを示す。図-9より、台数が増えれば増えるほど成立率は高くなるのが分かる。

一方、各ケースからゆとり時間を変化させることは成立率には影響を与えていないことも読み取れる。高い成立率はサービスの利便性向上に寄与するが、目標値を決めてこのようなシミュレーションを行うことで、あらかじめ必要な台数を決定することができる。

なお、台数が成立率に強い影響を与えていることを示しているのが、図-10である。図-10は図-9を作るために行った全100ケースについてゆとり時間や台数を変化させたときの成立率と計算時間の関係をプロットし、台数別に表示方法を変えたグラフである。グラフより明確に台数によってクラスタリングされていることが分かる。また、成立率と計算時間の相関は非常に強いことから、本研究で提案するアルゴリズムの特徴として成立率が向上すると計算時間も短くなるという特徴が見いだせる。

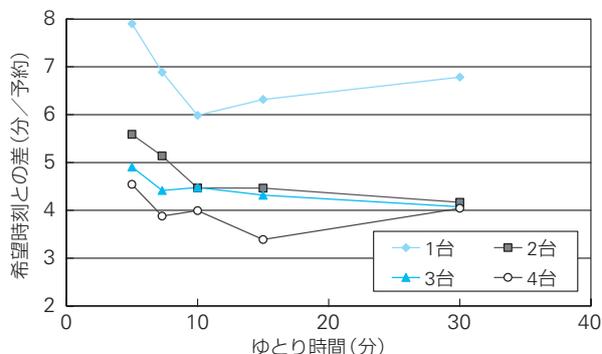


■図-10 台数別の計算時間と成立率の評価

ゆとり時間は車両台数に比べて成立率に影響を及ぼさないことは図-9および図-10より読み取れるが、多少なりとも成立した解に影響を及ぼしている。図-11は横軸にゆとり時間、縦軸に成立した1予約あたりの希望時刻との差を示したグラフである。縦軸の希望時刻との差とは、解が見つからなかった時に±20分の範囲で希望到着時刻を変えることを述べたが、その時に希望到着時刻を変化させた程度を示す。すなわち、この値が小さいほど希望到着時刻に近い時間に運行できることを示している。ただし、簡単のためゆとり時間がTT/2のケースは、その平均ゆとり時間である7.3分を横軸にとった。

図-11より、すべてのケースにおいて右下がりとなっている。したがって、各々の車両数のケースにおいてゆとり時間を大きくすると経路全体にゆとりができるため、希望到着時刻をずらすことなく経路を生成できる可能性が高まる傾向が読み取れ、ゆとり時間が大きければより

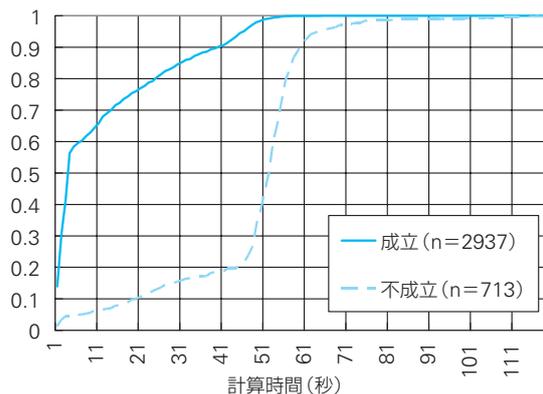
希望時刻に近い予約が成立できているといった事が分かる。また、台数を増やすことでより希望に近い到着時刻が指定できるようになっている傾向も読み取れる。なお、ゆとり時間と希望時刻との差が非単調であるのは各々のケースにおける成立率が異なっていることが原因だと推察できる。



■図-11 ゆとり時間による希望時刻との差

4.2.2 一人あたりの計算時間の評価

各々の計算にかかる計算時間を評価する。台数を2台とした時のシミュレーションの結果を成立した予約と不成立だった予約の2種類に分けた。一ケースのシミュレーションで146予約を対象とし、ゆとり時間を5種類に変化させ、さらに各々のケースに対し5回ずつシミュレーションを行ったため全部で3,650予約を処理したことになる。結果、予約のうち2,937予約が成立、713予約が不成立だった。それぞれの場合に要した計算時間を積み上げたグラフが図-12である。



■図-12 一人あたりの計算時間(累積)

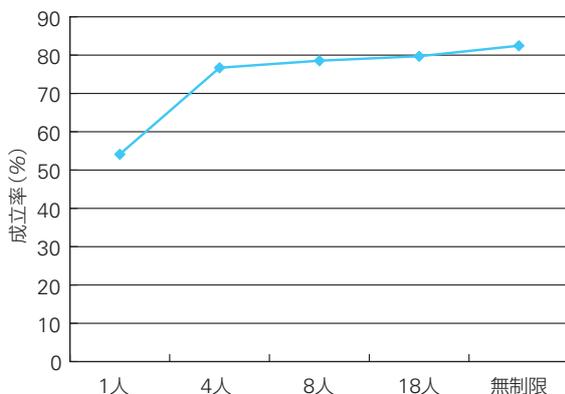
図-12より、成立した予約のうち6割は約6秒以内で、8割は25秒以内で返されることが分かる。一方、不成立の8割は解が出るまでに45秒以上を要しているのが分かる。これは、解が見つかるまで探索し続けるという本スケジューリングアルゴリズムの特徴を示しているといえる。

今回のシミュレーションでは計算時間の上限を設けなかったが、想定される需要に対してこのようなシミュレー

ションを行うことで、計算時間の上限を求めることが出来る。

4.2.3 乗り合い率の評価

最後に車両の定員を変化させたときの成立率の推移を示す。定員は、1, 4, 8, 18, 無制限と変化させた。4人はセダン型タクシーを、8人はジャンボタクシーを、18人はマイクロバスを想定している。図-13は車両が2台の時の結果である。図より4人以上のケースの場合、わずかに成立率は向上するものの、ほとんど違いは見られないことが分かる。すなわち、本ケースのシミュレーションでは4人程度の乗り合いまでが頻繁に起こっていたことが確認できる。このような方法により想定しているケースに対して最適な車両規模を想定することが出来る。



■図-13 定員を変化させたときの成立率の推移

4.3 シミュレーション実験のまとめ

本研究で開発したスケジューリングアルゴリズムの性能を評価する目的でシミュレーション実験を行った。結果、成立率の向上には台数が大きく影響しており、ゆとり時間は影響を及ぼさないことが確認できた。ただし、ゆとり時間が大きいケースでは希望した時刻をずらさずに経路を生成しやすくなることを確認できた。また、成立した予約の6割は6秒以内に返るが、不成立だった予約の8割は45秒以上かかるというスケジューリングアルゴリズムの特性を見ることが出来た。

また、台数や定員を変えて成立率や計算時間を調べることで導入規模に応じた最適な車両数、定員などといった導入設計を行えることも示した。

なお、ここで示した結果は4.1.1や4.1.2の前提下で得られる結果であることを留意する。したがって、結果の数値は参考値として考えるべきで、バス停の数や乗客数、また使用する計算機のスペックによって変化する。

5—考察

リアルタイム処理方式を用いた経路や時刻が完全に非

固定のオンデマンドバスに用いるスケジューリングアルゴリズムを開発し、成立率という目的関数の下でその評価をシミュレーション実験により行った。以下に、シミュレーション実験から得られた考察、および本アルゴリズムの実用を考える際の課題を示す。

シミュレーションの結果、ゆとり時間や台数を変化させることで解の成立率が変化する傾向を確認できた。総じて、ゆとり時間を変化させる効果と台数を増やす効果については台数を増やす効果の方が大きい事が分かった。一方で計算時間については、成立した予約のうち6割は約6秒以内で、8割は25秒以内で返され、不成立の8割は解が出るまでに45秒以上かかるといった本アルゴリズム独特の傾向を確認できた。たとえば、10秒で成立と返らなければ不成立とみなすといったアルゴリズムを加えることで、不成立率は上がるものの、より実用的な運用が可能になる。

また、車両の定員数を変えた実験からは、定員を1人→4人→8人と増やすことで成立率が増加したことから4人以下の乗り合いは頻繁に、8人程度の乗り合いはわずかに発生していたことが分かる。

以上、アルゴリズムの性能を成立率および応答時間の評価指標から評価した。その結果、これまでのオンデマンドバスとは異なり計算結果をすぐに応答し、かつ約束した乗車時刻および降車時刻を一定のTime Windowの制約条件において保証する新しい概念のオンデマンドバスは本アルゴリズムによって実現できそうと分かった。

一方、本アルゴリズムを実際のオンデマンドバス運行に適用させるためには次に示す課題があると考えられる。

まず、バス選択アルゴリズムについての課題を示す。本研究ではシミュレーション実験で行うバス停の数の少なさ、およびエリアの少なさからベクトルの偏角のみでバス選択を行ったが、より広い範囲かつ多くの台数、より密度の高いバス停の設定で実験を行う場合は偏角の他に位置や道路ネットワークといった要素を加えなければ、適切なバスを選択することは出来ない。

次に、経路生成アルゴリズムであるがより計算時間を短くするための要素として、如何に妥当な初期解を得ることができるかが挙げられる。本研究では記述の通り初期解を一律に設定したが、初期解の設定次第では計算時間や成立率のさらなる向上が期待できる。

最後に、シミュレーションの評価方法についての課題を述べる。交通システムとして総費用がどの程度小さくなるかといった交通政策的なシミュレーションを行わなければアルゴリズムの実用性については判断できない。本稿は開発したスケジューリングアルゴリズムの評価であるため一般的な成立率と計算時間という2つの指標で

評価を行ったが、実用化を考える際にはこのようなシミュレーションも重要である。

6—結論

リアルタイム処理方式を用いた経路や時刻が完全に非固定のオンデマンドバスに用いるスケジューリングアルゴリズムを開発した。開発したアルゴリズムはスケジューリングによる遅延という既存のオンデマンドバスに見られる問題点を生じないアルゴリズムである。

また、開発したアルゴリズムの性能をシミュレーション実験により評価をし、スケジューリングアルゴリズムの特性および台数や定員を調整することによって高い成立率を実現しうることを検証した。

また、本稿のシミュレーションを導入前に行うことで事前に最適な車両数や計算時間を設定することができることを示した。しかし、その際には路線バスの移動データのみではなく、地域の移動人口や大型施設の利用者などを加味した詳細な移動モデルを構築する必要がある。

今後は、開発したスケジューリングアルゴリズムを中心に据えた実システムの開発が課題となる。

謝辞：本研究は、NEDO（新エネルギー・産業技術総合開発機構）およびJST-CREST（独立行政法人 科学技術振興機構 戦略的創造研究推進事業）、KACITEC（柏の葉キャンパスシティ ITコンソーシアム）の資金をいただき

ました。なお、国土交通省総合政策局交通計画課長 佐々木良様、東京大学大学院工学系研究科都市工学専攻教授 原田昇先生からご指導をいただきました。ここに謝意を表します。

参考文献

- 1) 国土交通省自動車交通局 [2006], “道路運送法等の一部を改正する法律”。
- 2) 元田良孝 [2007], “バス問題に悩む自治体を救え! 市町村の混乱とバス110番”, 『日経グローバル』, No. 70 2007年2月19日, pp. 36-39.
- 3) 松下電器産業株式会社, 高知西南交通株式会社, 中村市, 高知県 [2000], “第4回デマンドバス実験検討委員会中山間地資料”, 『中村まちバス実験結果報告書』。
- 4) 福本雅之, 加藤博和 [2005], “適材適所となる少需要乗合交通サービス提供に関する基礎的検討”, 『第31回土木計画学研究発表会』, CD-ROM.
- 5) Itsuki Noda, Masayuki Ohta, Kasuke Shinoda, Toichiro Kumada, Hideyaki Nakashima [2003], “Evaluation of Usability of Dial-a-Ride Systems by Social Simulation”, *Proc. of Fourth International Workshop on Multi-Agent-Based Simulation*, pp. 139-152.
- 6) 大和裕幸, 稗方和夫, 坪内孝太 [2006], “オンデマンドバス～公共サービスに於けるイノベーション～”, 『オペレーションズ・リサーチ』, Vol. 51 No. 9, pp. 579-586.
- 7) Grötschel, M., D. Hauptmeier, S. O. Krumke, and J. Rambau [1999], “Simulation studies for the online dial-a-ride problem.”, *Preprint SC99-09, Konrad-Zuse-Zentrum für Informationstechnik Berlin*.
- 8) Jang-Jei Jaw, Amedeo R. Odoni, Harilaos N. Psaraftis, Nigel H. M. Wilson [1986], “A Heuristic Algorithm for the Multi-Vehicle Advance Request Dial-A-Ride Problem With Time Windows”, *Transpn. Res.-B* Vol. 20B, No. 3, pp. 243-257.
- 9) Haibing Li, Andrew Lim [2003], “Local search with annealing-like restarts to solve the VRPTW”, *European Journal of Operational research*, Vol. 150, pp. 115-127.
- 10) A. Fabiri, P. Recht [2006], “On dynamic pickup and delivery vehicle routing with several time windows and waiting times”, *Transpn. Res.-B* Vol. 40, pp. 335-350.

(原稿受付日 2007年4月16日)

A Research on Real-time Scheduling Algorithm for On-demand Bus and Evaluation by Simulation

By Hiroyuki YAMATO, Kota Tsubouchi and Kazuo Hiekata

The road transport law was amended on October 2006 and On-demand Bus service which has flexible route and flexible schedule will become more popular. In the On-demand Bus System, it is crucial to keep the arrival time of the individual customers to be punctual. This paper proposes new scheduling algorithm for this purpose with introducing slack time which regards as service time window in the bus stop. Simulation experiments for the model of Kashiwa-city were completed and the effectiveness of the algorithm was confirmed.

Key Words : *On-demand bus, time-constraint demand, simulation, amended road transport law*
