

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 —

大和 裕幸, 稗方 和夫, 坪内 孝太

利用者の要求に合わせて運行するオンデマンドバスは新しい公共交通サービスとして期待されているものの、成功例はほとんどない。日本各地で運行されているが、その多くが赤字運営で地域の助成無しでは運行できない。本稿では現行のオンデマンドバスの調査から、時間的制約のある利用目的に対応できていないという問題点を指摘する。さらに、その問題点を解決する新しいアルゴリズムを用いた実用的なオンデマンドバスシステムを提案する。提案したシステムを構築し、千葉県柏市北部において実証実験を行い、乗客のアンケート結果からその有効性を示す。それにより、提案する新しいオンデマンドバスサービスが、個を対象にした公共マストランジットという新しい概念の交通手段となりうる可能性について述べる。

キーワード：オンデマンドバス, DAPTW, 個別適合, 公共サービス, アルゴリズム

1. 序論

駅などから自宅への移動手段として長年利用されてきているにも拘わらず、路線バスは、時刻が正確でない、路線がわかりにくい、などの不便な点が多く、快適な交通とはいえない。また、運行会社の採算が成り立つことも困難な事が多い[1]。一方、都市圏の拡大と過疎地域の増加から、バスのような安価でフレキシブルな交通システムの必要性はますます高まってきている。

車両技術ばかりではなく、最適化アルゴリズム、移動体通信、データベースなど最近の情報技術を用いて、新しいサービスを実現することができる。OR 技術を用いた革新的なサービスとしてのオンデマンドバスシステムを提案したい。

2. 背景と目的

オンデマンドバスは、乗客の需要に合わせてフレキシブルに運行するバスである。定刻に決まったバス停を決まった経路で巡る路線バスに対し、オンデマンドバスには時刻表や経路が決まっていない。そのため、マストランジットの個別対応運行、環境に優しい交通などといった側面から新しい公共交通手段として期待されている。

地域の交通空白地域・不便地域の解消など、地域住

やまと ひろゆき, ひえかた かずお, つぼうち こうた
東京大学 大学院新領域創成科学研究科
〒277-8563 柏市柏の葉 5-1-5

民の利便性向上のため、全国でコミュニティバスが運行されている。しかし、その多くは採算性に問題がある。もともと公共サービスの困難なところに導入されるので基本的に採算がとれないのが実状とはいえ、ランニングコストさえも賄えておらず、昨今ではコミュニティバスを廃止する地方公共団体も増え始めている。

コミュニティバスを運行している地方公共団体は、少数のバスで広大な地域を運行しカバーしようとする。そのため、バスがすべてのバス停を1巡するのに2時間もかかり、1日に4本しかスケジュールを組むことができないといった事例もある。このような理由から公共交通手段としての利便性は低く、利用者も少ない。図1に示すような典型的な悪循環に陥っているバスが多い。

一方、需要があるときにのみ稼働するオンデマンドバスでは乗り合いによる効率と経路の自由から、採算性もよく便利な末端輸送システムとなる可能性がある。しかし、現状のオンデマンドバスは時間的制約のある利用目的に対応できないというところに問題がある。

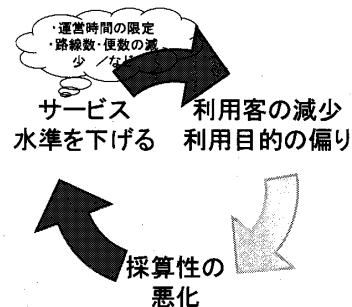


図1 地域運営バスが陥る悪循環

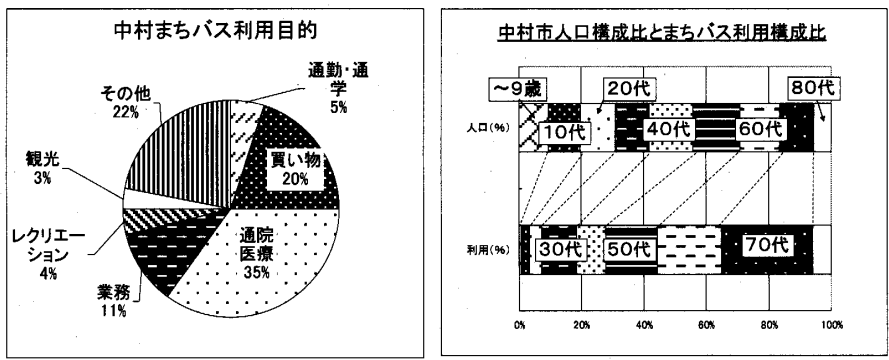


図2 中村まちバス利用状況

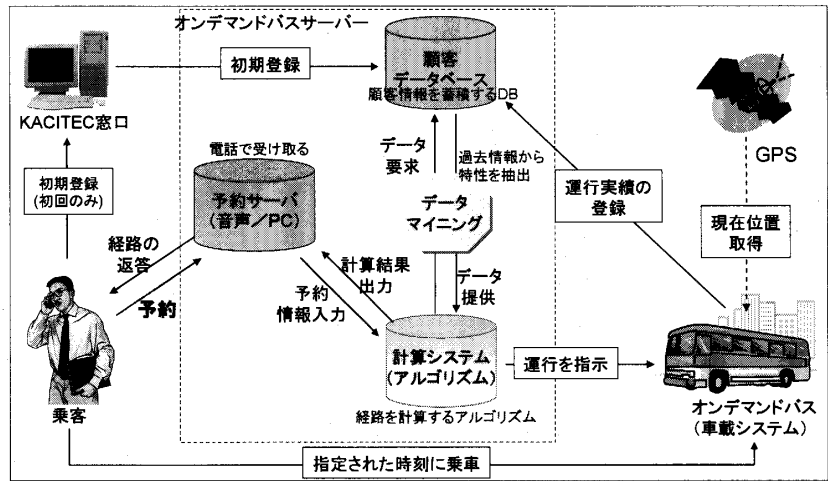


図3 システムの概要

高知県中村市（現：四万十市）では、『なかむらまちバス[2]』と呼ばれるオンデマンドバスが運行されている。乗客が電話をかけると、オペレータが対応しバスを配車するシステムである。地域住民からの期待は高い。しかし、図2からも分かるように高齢者向けの交通手段であり、利用目的も限られていることがわかる。

たしかに地方の公共交通の整備状況の悪化は高齢者に最も影響を及ぼす。高齢者に多く利用されている現行のオンデマンドバスは地方交通の利便性を向上させているといえよう。本稿では多様な需要に対応しさらに利便性を向上させる可能性について述べる。そのために、対象を拡大し幅広い年齢層、利用目的に対応できるように新しいオンデマンドバスサービスを提案する。

3. 提案するオンデマンドバスサービス

3.1 求められる機能

前節に述べたように、これまでのオンデマンドバスは、予約した段階でバスに乗る時刻・降りる時刻を予想できない。乗客がバスを予約した時点で、その予約

の後にいくつ乗り合いの予約が入るか分からないためである。

バスの乗車時刻および下車時刻を予約の時点で把握することができるという機能が必要である。しかも、オンデマンドバスの特徴でもある乗り合い機能は損なわない。このサービスを実現するために、リアルタイム処理技術と到着時刻にタイムウィンドウを設けるという考え方を導入した。

3.2 開発したシステム

当研究室で開発したシステムを図3に示す。乗客がwebや携帯電話を通じて予約サーバにアクセスする。サーバは出発バス停・到着バス停・希望到着時刻の3つの予約情報を獲得し、その情報を計算アルゴリズムサーバに引き渡す。計算アルゴリズムサーバでは、独自のアルゴリズムを用いて運行計画を更新する。そのアルゴリズムについては第4節で述べる。うまく経路が生成できた場合には、乗車バス停での待ち合わせ時刻を割り出し予約サーバに伝え、予約サーバが結果を乗客に音声で返す。このようにして利用者は予約の段階で出発地、出発時刻、到着地、到着時刻まで知ることができる。一方、バス側の車載器にも運行計画が更新

されたことを伝える。乗客は乗車バス停での乗車時刻に間に合うようにバス停に向かい、時刻になればバスが来るので乗車する。

本システムは予約サーバ、計算システム、車載システム、そして顧客データベースという4つの機能から成り立っている。計算システムについては第4節で述べる。ここでは、他の3つの機能について示す。

まず、予約サーバについて説明する。予約サーバはこれまでオペレータが行っていた顧客のデマンド情報の聞き取りを行う機能をもつ。回線制御はComputer Telephony Integration (CTI) プラットフォームにより行う。CTIプラットフォームにより回線が確立されると、顧客とシステムの間で対話が行われる。予約サーバは音声認識技術により顧客の話した内容を認識する。対話フローを実装するための技術にはVoiceXMLを利用した。また、予約サーバと計算システム間の対話についてはCGIを用いた。

次に、車載システムについて説明する。新しいデマンドが入る度に更新される運行計画の情報や顧客情報をバスとサーバの間で交換する必要がある。この通信はTCP/IP上のソケット通信で行う。車載器はFOMA端末によりインターネットに接続している。

バスに積まれた車載器には開発したソフトをタブレットPC上に組み込み、構築した。開発したソフトは操作環境のよくない運転席でも操作できるよう、操作が簡便で、ミスも少なくなるよう心懸けて作成した。

最後に顧客データベースについて説明する。顧客データベースは2つの機能を有する。顧客情報を管理する機能と、顧客の利用状況のログを保存する機能である。顧客情報の管理は汎用のリレーショナルデータベースを用いて行う。利用状況のログの保存については、大量の利用実績を保存し、顧客の個性や習慣などといった要素を抽出することを目的としている。これにより、より個に対応したサービスが可能となる。現在はログを保存している段階であるが、今後高度なデータマイニング技術を用いたログデータの利用を課題としている。

3.3 リアルタイム処理

既存のオンデマンドバスの運行計画生成には、一定時間の予約をまとめて処理するバッチ処理が用いられてきた。そのために、利用者は実際に乗車してみなければ到着時刻が分からないなどという不便さがあった。

本システムにはバッチ処理ではなく、予約の度に運行計画を更新していく処理方法であるリアルタイム処

理を用いた予約システムを導入した。

リアルタイム処理を導入するため、アルゴリズムには顧客がストレスを感じない程度の待ち時間で運行計画を返答できる速いものが必要となる。この問題を解決したアルゴリズムについては第4節で説明する。

3.4 ゆとり時間の導入

リアルタイム処理は、利便性の高い予約方法を実現するメリットを持つと同時に、大人数の予約処理を妨げるというデメリットを持つ。似たような移動の乗客を乗り合わせる機能はバスという交通手段の特長の1つであり、それを損なう処理方式は根本的に問題である。

この問題に対し、タイムウィンドウを用いた予約方法を採用することで解決を図った。乗客に希望到着時刻を「点」で指定させるのではなく、タイムウィンドウをもった「幅」で指定させることで問題の制約条件を緩めることができる。このタイムウィンドウのことを「ゆとり時間」と呼称する。このゆとり時間の値には、一般路線バスの到着遅れ限界である10分を用いた。

また、到着時刻にゆとり時間の概念を用いることで、リアルタイム処理による予約方法でも通常の乗り合わせと同等のサービスを実現することができる。例えば、9:00に到着を希望する場合には、8:50-9:00の間に到着するように乗降することになり、9:00を過ぎることはない。

4. 問題解決のためのアルゴリズム

次に、独自に開発した計算アルゴリズムについて説明する。計算アルゴリズムは、バス選択アルゴリズム・運行計画生成アルゴリズムの2段階に分けた。問題の定式化を行い2段階のアルゴリズムについて説明する。

4.1 問題の定式化

オンデマンドバスの運行計画問題は一般的にDial-A-Ride Problem with Time Window (DARPTW)と呼ばれる問題である。経路生成問題として有名な巡回セールスマン問題(TSP)と異なり、各ノードや運送する乗客ごとに割り当てられた時間などの強い制約条件を持つことが特徴である。なお、問題の定式化については、Yanら[3]やMarcoら[4]の論文を参考にした。

オンデマンドバスを利用する際に、乗客 n は予約の際に乗車バス停と降車バス停を指定する必要がある。

さらに、本稿で提案するオンデマンドバスの場合、降車バス停への希望到着時刻を指定する。

一方、バスは単数あるいは複数の予約により構成された制約条件を満たしながら移動する。この時、各バス停に課せられる制約条件は ET_n (earliest time) と LT_n (latest time) の2つの変数で表すことができる。バスはあるバス停に少なくとも時刻 ET_n までに到着しておく必要があり、少なくともそのバス停を時刻 LT_n には出発しておかねばならない。

バスは時刻 ET_n から時刻 LT_n の範囲内の時刻 AT_n (arrive time) に到着し、時刻 ST_n (set out time) に出発する。また ST_n と AT_n を用いることで、以下の式からバス停で待機する時間 WT_n (wait time) が分かる。

$$WT_n = ST_n - AT_n \quad (\max WT_n = LT_n - ET_n) \quad (1)$$

さて、 ET_n や LT_n は乗客の予約によって決まる。乗客の予約情報 D_n は次のように整理することができる。

$$D_n = \{p_n, d_n, EPT_n, LPT_n, PT_n, EDT_n, LDT_n, DT_n, Bus_n\}$$

ただし、 p_n は乗車バス停の番号を、 d_n は降車バス停の番号を示す。 Bus_n は乗客が乗車するバス番号を示す。 EPT_n は乗車バス停にバスが到着できる最も早い時刻 (earliest pick-up time) を、 LPT_n は乗車バス停にバスが到着する最大限に遅い時刻 (latest pick-up time) をしめす。すなわち、乗客 n が乗車するバス停 p_n にバスが到着する時刻 PT_n (pick-up time) について考えた場合、次の関係式が成り立つ。なお、本稿では、時間の早遅を大小関係で示す。

$$EPT_n \leq PT_n \leq LPT_n \quad (2)$$

また、降車バス停 d_n に到着する時刻に関しては同様に EDT_n (Earliest drop-off time), LDT_n (latest drop-off time), DT_n (Drop-off Time) についてその制約が記されており、以下の関係式が成り立つ。

$$EDT_n \leq DT_n \leq LDT_n \quad (3)$$

これまで、一般的な DARPTW にあてはまる式について述べた。ここからはこれらの変数間に成り立つ関係式について述べる。提案するオンデマンドバスシステムに適用する場合、以下の式が成立する。

$$PT_n = LPT_n \quad (\text{ただし、予約済の場合}) \quad (4)$$

運行計画生成の対象となる乗客の場合、 PT_n の値を導出することが目的であり、予約処理の段階では定まっていない。運行計画生成の対象となる乗客に対しては LPT_n を以下のように定義した。

$$LPT_n = LDT_n - TT(p_n, d_n) \quad (5)$$

ただし、 $TT(p_n, d_n)$ は p_n から d_n まで直接移動した際の移動時間である。つまり式(5)は乗客 n の出発地を少なくとも時刻 $LDT_n - TT(p_n, d_n)$ には出発しておかねばならないことを示す。なお、移動時間は全経路の移動時間をデータとして持っており、それにより求める。

また、 EDT_n と LDT_n には次の式(6)に示す関係式がある。式中の slacktime はゆとり時間を示す。

$$LDT_n - EDT_n = \text{SlackTime}(10 \text{ min}) \quad (6)$$

次に、バスの移動については次のモデルで示す。

$$Bus_{i,x} = \{a_{i,x}, ET_{i,x}, LT_{i,x}, AT_{i,x}, ST_{i,x}, Cap_i\} \quad (7)$$

ただし i はバスの番号を示し、 x はバス停を巡る順番を示す。すなわち、 $Bus_{i,x}$ はバス i が x 番目に停車するバス停での様子を示す。 $a_{i,x}$ はバス i が x 番目に通過するバス停の番号を、 $ET_{i,x}$ (earliest time) は $a_{i,x}$ に少なくとも到着していなければならない時刻を示す。さらに、バス i はバス停 $a_{i,x}$ を少なくとも $LT_{i,x}$ (latest time) には離れ、次のバス停 $a_{i,x+1}$ に向かわねばならない。実際に $a_{i,x}$ に到着するのは時刻 $AT_{i,x}$ で、実際に $a_{i,x}$ を出発する時刻は $ST_{i,x}$ である。これらの値は乗客の予約により定義される。

また、 $a_{i,x}$ の成分からなる行列 \mathbf{a} は例えば以下のようにっており、バス i が x 番目に通過するバス停における乗客と降客の両方を要素に含む行列である。

$$\mathbf{a} = \begin{pmatrix} p_1 & p_6 & p_3 & p_5 & d_1 & d_3 & d_5 \\ p_2 & p_4 & d_2 & d_4 & 0 & 0 & 0 \end{pmatrix}$$

この問題の目的は、この行列 \mathbf{a} および行列 \mathbf{AT} 、行列 \mathbf{ST} を求めることである。それにより、 p_{new} におけるバスとの待ち合わせ時刻を求めることができる。ただし、その際に乗客のデマンドから構成される時間の制約条件を満たす必要がある。

4.2 アルゴリズムについて

上記の問題をバス選択アルゴリズムと運行計画生成アルゴリズムという2段階で解決する。まず、バス選択アルゴリズムにより、新しく入った予約がどのバスに割り当てられると最も良い経路が見つかりやすいかを判断することができる。

次に、経路生成アルゴリズムを適用する。経路生成アルゴリズムでは、今後予約が入るであろう乗客予定者とうまく乗り合わせができ、かつ乗客が指定した時間的制約を破らないように経路を生成する。

アルゴリズムをこのように2段階に分けることで問

問題規模が変わっても計算時間が等比級数的に増えない実用性を重視した。Dial-a-rideの問題に見られる多くのアルゴリズムは、問題規模の拡大と共に計算時間が級数的に増大する。これでは、数台の小さい規模の問題を解くときには支障を来さないものの、問題規模が大きくなってしまった時に実用化に耐えうるものではなくなる。本問題の場合は特に、順番に入るデマンドが制約条件となり、解の探索範囲は絞られる。しかしそれでも即座の解法は大変であるので、工夫が必要になる。本研究では、大規模の問題に対応できるアルゴリズムを開発した。具体的には図4のアルゴリズムとなる。

ここでの工夫点は2点ある。まず1点目は、バス選択アルゴリズムにおいて、計算の負荷がかからない簡単な計算で準最適なバスを選択する。具体的にはオーダー $O(n)$ の計算量となるように工夫した。

2点目は、バスが選択された後に行う運行計画生成アルゴリズムでは、拡張性を重視したアルゴリズムを考案した。具体的には、経路への逐次挿入を改善したヒューリスティックな解法で、データベースと連動しており拡張性も高い。

以上、計算負荷のかかるヒューリスティックな解法には、バスの台数が増えても1回のみ行うという仕様にし、バスの台数が増えることによるインパクトは計算負荷の小さいバス選択アルゴリズムにのみ与えることで、問題規模の差異による計算時間増加という問題点を解決した。

4.3 バス選択アルゴリズム

バス選択アルゴリズムでは、少ない計算時間でかつ大規模の問題を解決する目的で、単純にベクトルを比較する方法を採用する。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、最適なバスを選択するという方法である。

新しく入った乗客 n の予約情報を示すベクトル \vec{A}_n を次のように表す。

$$\vec{A}_n = (d_{nx} - p_{nx}, d_{ny} - p_{ny}) \quad (8)$$

ここで d_{nx} とは、 d_n の緯度経度情報を示し、日本測地形のデータにより x 成分 (緯度)、 y 成分 (経度) を算出している。

一方で、比較に用いる i 番目のバスのベクトル \vec{B}_i を次のように表す。

$$\vec{B}_i = (a_{i,t^*x} - a_{i,s^*x}, a_{i,t^*y} - a_{i,s^*y}) \quad (9)$$

ここで、 a_{i,s^*x} は、 i 番目のバスにおける s^* 番目のバス停の x 座標であることを示す。さらに、 s^* 、 t^* はそれぞれ予約情報の乗車時刻 LPT_n と下車時刻 LDT_n に最も近いタイミングを示す。 LPT_n の値は、この時点で定まっていなくても、先述した式(5)の値を採用する。

$$LPT_n = LDT_n - TT(p_n, d_n) \quad (10)$$

ここで、以下の $\cos \theta_i$ を各々の内積から求め、その結果 $\cos \theta_i$ の値が最も大き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。

$$\cos \theta_i = \frac{\vec{A}_n \cdot \vec{B}_i}{|\vec{A}_n| |\vec{B}_i|} \quad (11)$$

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を乗車候補として選択する。角度が近いことは、進行方向が類似していることを意味し、最も運行計画が生成されやすいバス停である可能性が高い。また、ここまでの計算量のオーダーは $O(n)$ であり、バスの台数が数百台レベルに増えた場合でも大して計算時間は変わらないことが確認できる。

4.4 運行計画生成アルゴリズム

運行計画生成アルゴリズムは決められたバスに対して、何番目の予約として処理すれば良いのかを決定するアルゴリズムである。具体的には逐次挿入から得ら

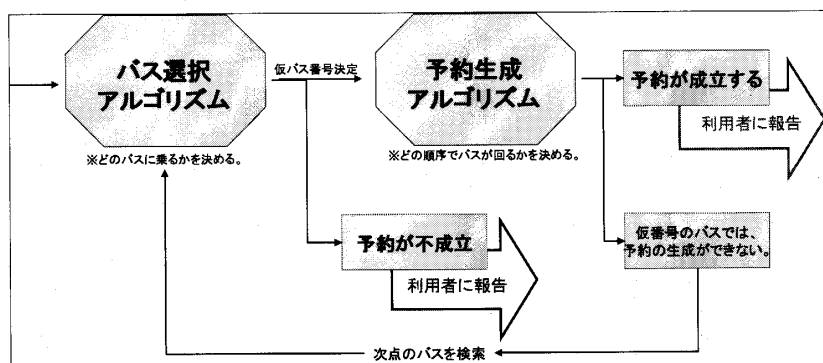


図4 アルゴリズム概要

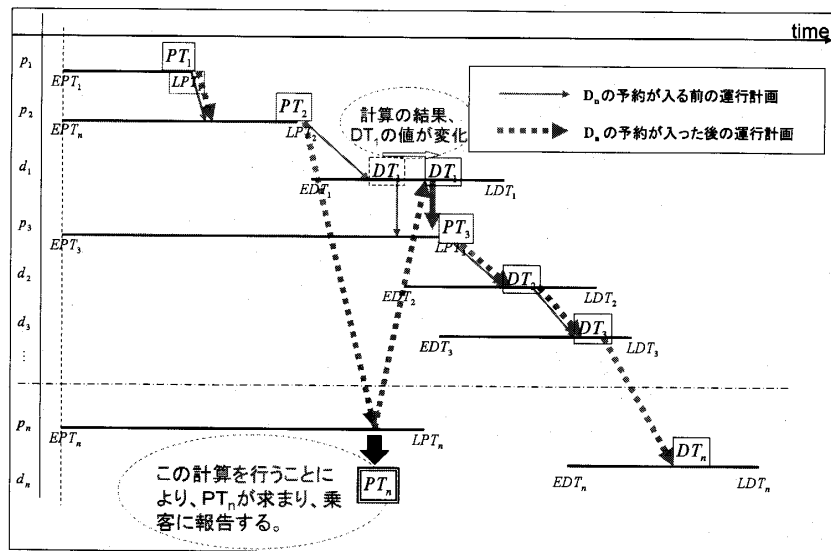


図5 運行計画生成のアルゴリズム

れた解をヒューリスティックな解法を用いて調整し、準最適化を行うという解法である。

アルゴリズムの概念を図5で示す。点線より上は乗客 n の予約 D_n が入る前のバスの移動状況を示す。 p_1 から始まり、時刻 PT_1 になれば p_2 に向かう。 p_2 には時刻 $PT_1 + TT(p_1, p_2)$ に到着する。

p_2 では $PT_2 - PT_1 - TT(p_1, p_2)$ 時間だけ待機し時刻 PT_2 になり次第、 d_1 に向かう。また、各バス停に割り当てられたタイムウィンドウは先に示す通りに定義し、それを線で表現した。

ここで、新しく D_n が入ることで、運行計画の再設計が行われる。ただ、 PT_1 から PT_{n-1} については、乗客に約束してしまった時刻であるため、固定する。

到着時刻を示す DT_1 から DT_{n-1} については、 $EDT_i \leq DT_i \leq LDT_i$ の範囲で移動できる。そして、このアルゴリズムを適用した結果、新しい運行計画が生成される。これにより、 PT_n が定まり、それを乗客にリアルタイムで伝達することができる。

5. 実証実験

5.1 実証実験の概要

この一連のシステムの有効性を示す実証実験を2006年の1月～3月の3ヶ月間、表1の概要で行った。

2006年1月の第1期では乗客を乗せず、ダミーの乗客を移動させる仮想実験を行った。2月に行った第2期の実証実験では、東京大学柏キャンパス関係者へのみ告知をし、利用して頂いた。2006年3月には、一般の乗客を対象にした実証実験を行った。第3期ではバスを2台に増やし、図6のように地域を拡大した。

表1 実証実験の概要

	第1期	第2期	第3期
目的	サービス動作検証	モニターによる利便性評価	一般客による利便性評価
期間	10日間	10日間	9日間
時間	19:30～22:30	19:30～22:30	19:30～22:30
バス台数	1台	1台	2台
地域	柏市北部	柏市北部	柏駅以北
利用客	なし	東大関係者	一般乗客
	—	55人	105人

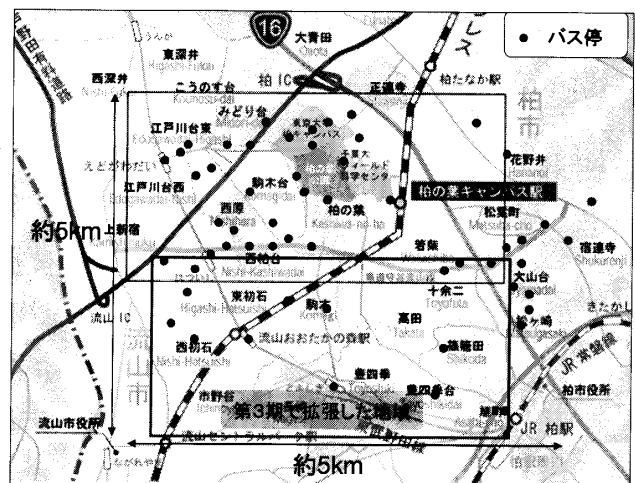


図6 実験エリア

なお、実証実験は乗客から運賃をいただかず無料で運行した。

5.2 実証実験の結果

5.2.1 予約の成立率について

ログ解析を行い、予約の成立率を調べた。結果は図7の通りである。不成立率が18%とあり、約5人に1人はバスの予約を断っていることが分かる。不成立を

減らすためには予約の密度に応じた適切なバスの台数を定義する必要がある。

5.2.2 利用者の年齢比や利用目的について

図2と図8とを比較すると、これまでとは異なる年齢構成・利用目的で利用されたことが分かる。この利用者の分布は東大関係者に広く利用を促すことで故意に設定したものである。これにより、提案するサービスがどのように受け取られるのかを検証しようとした。ただ、実験時間が夜間であるため、通勤・通学の時間的制約に対する要望は出勤時の移動よりも弱まることに留意したい。

5.2.3 便利だと思った点

乗客にこのサービスで便利だと感じた点についてアンケートを行った。図9より多くの方が時間を指定できることに価値を見いだしていることが分かる。また、オンデマンドバス特有のバス停をきめ細やかに増やすことができることについても便利だと感じている人が多かった。

5.2.4 再利用の意思

総合的に判断し、再度このサービスを利用したいかを尋ねた。図10より7割近くの乗客が再度利用する意思を示したことが分かる。このことから、本サービスが好評だったことが読み取れる。ただ、運賃が無料で行った影響も考えなければならないことに留意したい。

5.3 実証実験のまとめ

実証実験の結果、提唱する「時間的制約のある利用目的に対応した」オンデマンドバスは、好評であったことが分かる。その結果、これまでは利用者や利用目的が限られていた交通手段であるオンデマンドバスも一般的な公共交通手段として存在しうることが分かった。

6. まとめ

オンデマンドバスの技術と実証実験について筆者らが考えるシステムを例としてその可能性について論じた。

具体的には、予約サーバ、独自の運行計画アルゴリ

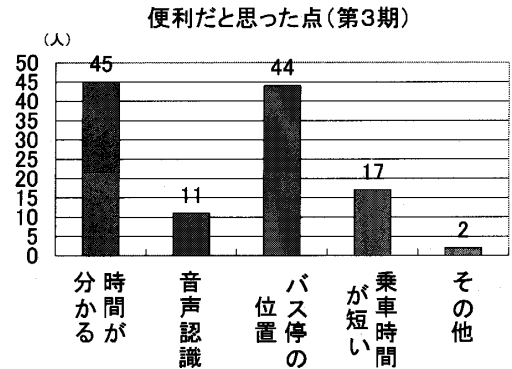


図9 便利だと思った点

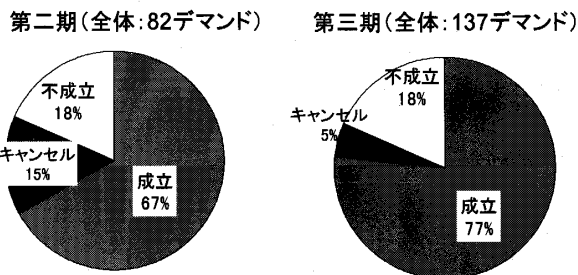


図7 デマンド成立率

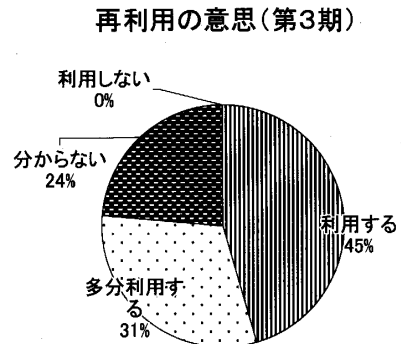


図10 再利用の意思

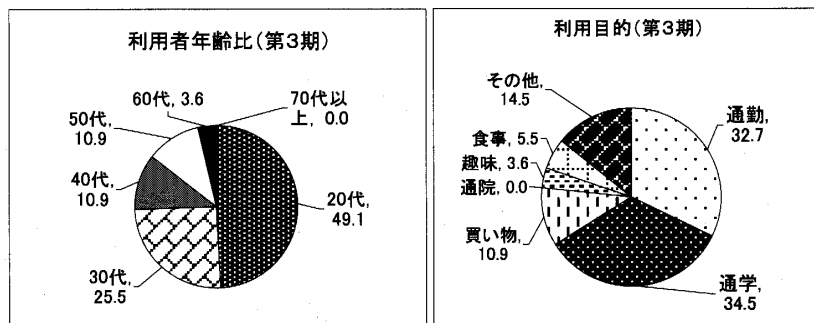


図8 利用年齢比と利用目的

ズム、車載システム、データベース技術を駆使して新しい公共サービスとしてのオンデマンドバスシステムを提案した。

実験により個へのサービスとしてのオンデマンドバスが成立しうる感触を得たが、今回の検討では交通システムの検討に不可欠な採算と便益の評価が欠けている。便益を検討しこのようなシステムの価値を適切に評価することがこういったサービスの導入を促す点で重要で、様々な都市構造に最適なビジネスモデルを定量的に考えることも事業化に先立って必要となる。

謝辞 本研究は、JST-CREST および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。システムの構築には NTT-IT, NTT ドコモのご協力をいただきました。実験をするにあたり、国交省、柏市のご助力を賜りました。バスの運行は東武バスイースト、染谷交通の二社が担当されました。ここに記して謝意を表します。

なお、本プロジェクトに関する詳細は下記の URL を参照されたい。

<http://www.nakl.t.u-tokyo.ac.jp/demandbus/index.htm>

参考文献

- [1] 国土交通省：“平成 16 年度の一般乗合バス事業（保有車両数 30 両以上）の収支状況について”
- [2] 中村市, 高知県：“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”
- [3] Shangyao Yan, Chin-Jen Chi, Ching-Hui Tang: “Inter-city bus routing and timetable setting under stochastic demands,” *Transportation Research Part A* 40 (2006) 572-586
- [4] Marco Diana, Maged M. Dessouky: A new regret insertion heuristic for solving large-scale dial-a-ride problem with time window, *Transportation Research Part B* 38 (2004) 539-557