

中規模都市を対象とした集約型デマンドバスの経路計画手法の提案 A Propose of Path Planning Method for Collective Demand Bus System in Middle-Sized City

新垣 隆伍[†] 赤嶺 有平[‡] 上原 和樹[‡] 遠藤 聡志[‡] 根路 銘 もえ子[§]
Ryugo Arakaki Yuhei Akamine Kazuki Uehara Satoshi Endo Moeko Nerome

1. はじめに

現在、地方都市では自家用車が主な交通手段となっており、交通渋滞や環境汚染といった問題が発生している。これを解決する為には自家用車の利用をおさえ、公共交通機関を利用することが重要である。しかし、地方部のバスは本数が少なく不便であるため、利用客数は減少傾向にある。また、利用客が減ることで、事業者の収入が減少し運営を縮小せざるを得ない状況に追い込まれ、利用客をさらに減少させる原因となる負の連鎖を生み出している。さらに、電車やモノレールといった軌道交通は、設置コストの高さから地方都市では開発は進んでいないのが現状である。

そこで、新たな公共交通機関として「デマンドバス」が注目されており、より少ないコストで運用するための経路設計手法や複数のバスへの乗客の割当手法、シミュレーションによる経済効果の予測等、様々な研究が行われている [1, 2, 3]。デマンドバスは、利用客の数により、乗車時間や経路長が長くなる傾向がある。従って、交通需要の多い都市部においては、あまり有効でないとされてきた。実際、現在デマンドバスが運行されている地域は人口数万人程度の小規模自治体に限られている。

この問題に対し内村らは、「階層的サービス機能を有するバス・システム」を提案し、その有用性を示した [1]。しかし、最下層のデマンドバスの経路設計のみについて言及しており、システム全体を考慮した最適化は行なわれていない。

筆者らは、エージェントシミュレーションによる評価が可能な程度の詳細なモデル化を前提として、デマンドバスを含む複数交通機関を連携させた交通システムの設計を行っている [4]。

本システムにおけるデマンドバスは、乗り継ぎを前提に、始点または終点を一地点に限定したデマンドバスを用いており、筆者らは「集約型デマンドバス (GDRP: Gathering Dial a Ride Problem)」と呼んでいる。本稿では、既存のデマンドバスと比較して「旅行時間」と「バスの走行コスト」を削減する GDRP の経路設計手法を提案し、シミュレーションにより検証する。

2. デマンドバスと GDRP

デマンドバスとは、通常の路線バスの形態とは異なる乗り合いタクシーのようなバスであり、利用客の要求に応じて経路やダイヤを決定する。需要のある場所のみを通ることができ、効率の良い運行が可能になる。デマンドバスには、利用客の要求に応じて自由に経路

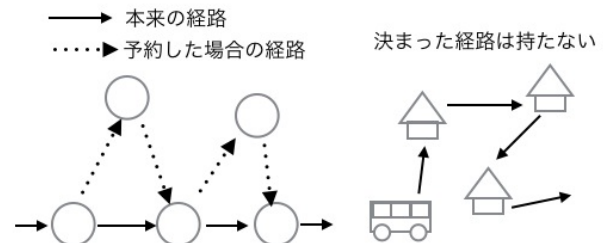


図 1: 2 種類のデマンドバスの走行経路の違い
(右: セミデマンドバス 左: フルデマンドバス)

を変更するフルデマンドバス (図 1: 右) と、普段は路線バスのように一定の経路を運行しているが、要求に応じて経路外の停留所を経由することのできるセミデマンドバス (図 1: 左) の 2 種類が存在する。筆者らが提案している集約型デマンドバスでは、要求を行なった全ての利用客を DEPOT に集約し、基幹バスや軌道交通等の大量輸送機関に乗り継いでもらうことを想定している。これにより、利用客は、徒歩で利用するには不便である駅を利用することができる。また、セミデマンドバスのように予約無しでの利用は出来ず、フルデマンドバスほど複雑な経路や計算を必要としないという特徴がある。

既存のデマンドバスの経路計画問題を DARP と呼び、集約型デマンドバスの経路計画問題を GDRP と呼ぶ。DARP は、同じ経路作成問題である巡回セールスマン問題 (TSP) や運送経路問題 (VRP) の条件に加え、乗車地点を降車地点より先に訪れる必要があり、より複雑な問題である。これは TSP, VRP と同様に、NP-hard に分類される問題であり [6]、デマンドの数が大きくなると最適解を求めるのが非常に困難である。一方、GDRP では乗・降車の一方を一地点に集約することで、乗車を先に行なうという制約がなくなるため、より広く研究がなされている TSP や VRP の解法が利用できる。また、GDRP を用いることで DARP に比べ計算の複雑性を緩和し、より単純かつ高速に経路計画や、コスト計算をすることが可能となると考えられる。

3. 提案手法

3.1. 提案手法の概要

提案手法では、バスの走行距離が短くなるような直線上、またはそれに近い経路を用いる。この場合、全ての利用客が希望乗車地点で乗車できる確率は非常に低いため、乗客が移動の際に旅行時間を最短にする戦略を取ると想定し、徒歩による移動と組み合わせるこ

[†]琉球大学大学院理工学研究科

[§]沖縄国際大学経済学部

とでバスの走行距離と全ての利用客の旅行時間を共に低減できると考えた。これにより従来のデマンドバスで問題とされる、デマンドの位置によるバスの走行経路の大きな変化、他の利用客による旅行時間の大幅な変動が軽減され、デマンド数の増加による計算コストの増加を抑えることが出来ると考えられる。

本稿では、バスの走行距離が最も短くなる直線上の経路と、利用客の歩行時間の削減を図った寄り道を行なう経路の2つの経路計画手法を提案する。また、この経路計画手法では、バッチ処理方式を採用しており、あらかじめ集めていた一定数のデマンドに対して経路計画を行なう。

直線型の経路は、既存のデマンドバスの様に複雑な経路でないため、バスの走行距離を短くすることが可能となると考えられる。また、一本の経路で対応する範囲が広いと DEPOT から遠くさがるほど徒歩による最大移動距離が増大し、大きな負担がかかる可能性が存在する。

寄り道型の経路は、直線型に比べてバスの走行距離は長くなると考えられる。しかし、寄り道を行なうことで経路とデマンドの位置との距離を縮め、利用客の徒歩での移動量を軽減することを狙っている。

この両者を比較することによって、GDRP における徒歩による移動の重要性を確認するとともに、曲線と直線のどちらが GDRP の運行に適しているかを判断する。

3.2. クラスタリングと経路生成

提案手法は、まず一台のバスが対応するデマンドを決定する為にクラスタリングを行い、その後、各クラスターで経路を生成する (Cluster-first, Route-second)。

バスの走行距離が短い経路を考えた時、経路は直線に近いほど良いと考えられる。しかし、直線上の経路のみでは全ての利用者の希望乗降地点を迎えることは出来ず、利用者の徒歩による移動距離が増大する。一方で、利用者の移動距離を短縮するとバスの走行距離が増大する、トレードオフの関係になっている。

本論では、主にバスの走行距離の短縮に着目し、直線上の経路を用いたとき、全ての利用客の総移動時間が最短となるよう、利用客の徒歩での移動距離が最短となるようなクラスタリングを行なった。

そのアルゴリズムについては、K-means を参考にし、以下のように処理を行なった。

1. 各デマンドを無作為にクラスターに割り当てる
2. 各クラスターにおいて全て出発点との総距離が小さくなるような近似直線を求める。(K-means におけるクラスターの中心に相当し、各デマンドの位置における方向ベクトルの平均より求める)
3. 全てのデマンドを、2 で求めた直線への距離が最も近いクラスターへ再割り当てする
4. 2 と 3 の動作を変化が無くなるまで繰り返し、クラスターに属するデマンドをそのバスの乗客とする

そして、クラスタリングの結果をもとにバスが通る経路の生成を行なう。直線上の経路を走行するバスは、全ての利用客の位置に対して、近似直線とデマンドの距離の平均が小さくなるような経路で走行する。そこで、クラスタリングの段階で計算した近似直線をバスの経路とする。デマンドバスはこの直線上の経路を往復することで利用客を運送する。その際に利用客の旅行時間を短くするため、バスは DEPOT から最も遠い地点で乗車する利用客から順番に乗車させ、DEPOT の方向に走行する。(図 2)

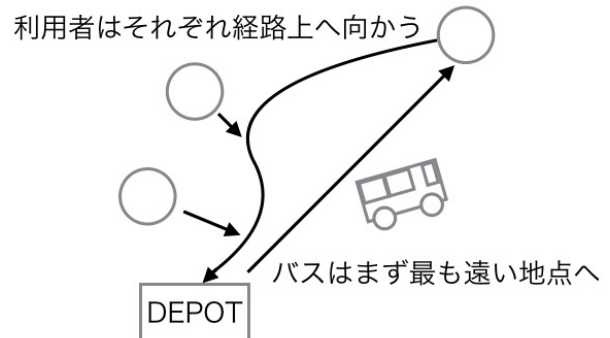


図 2: 走行例

寄り道を行なうバスの経路は、クラスタリングが終わった状態で、DEPOT から近い順にソートされた各デマンドの位置を結ぶ線を曲線で近似することによって求める。近似処理は以下の様に行なった。

1. 各クラスターの全てのデマンドを DEPOT に近い順にソートする
2. 各デマンドの間をいくつかの制御点に分割する
3. 各制御点の位置を前後の制御点の位置の平均に更新する
4. 3 を k 繰り返す (k を以下曲線係数と呼ぶ)

本稿での実験は、全て格子状のマップを用いたため、ユークリッド空間上の経路をマップ上の経路に変換する必要がある。そのため、格子状のマップ中で最も近似曲線に近い経路を選び出すように以下のように処理を行なった。

1. DEPOT を着目ノードとする
2. 着目ノードから近似曲線に最も近いノードを選択し更新する (ただし既に通過したノードを除く)
3. マップの端につく、もしくは近似曲線の端に達するまで2を繰り返したら終了する
4. 着目ノードになったノードを順番に辿ったものをバスの経路とする

上記処理をすることで、全利用客の総旅行時間が小さくなるような経路を作成できる。実際に作成した経路の例を図 3 に示す。右が寄り道をするように経路計画をした際の経路、左は寄り道をせず、極力直線に走行するタイプの経路である。DEPOT は左下に位置しており、大きい点は利用客の出発点を示している。また、クラスタは色で示されており、太い線はバスの経路である。例では 3 台のバスを利用しており、デマンドと同様に色によって対応するクラスタを表す。

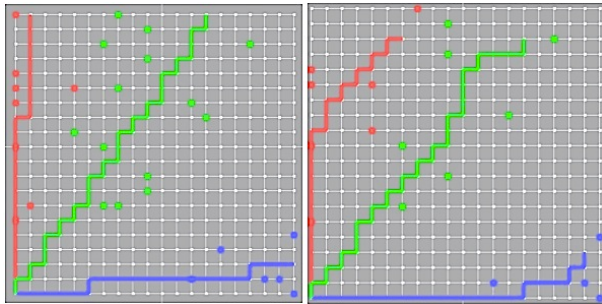


図 3: 生成経路例 (右:寄り道型 左:直線型)

4. 実験

4.1. 実験概要

提案手法の有効性を確認するため、利用客の平均旅行時間とバスの合計走行距離の 2 つの観点から、GDRP に対して 3 種類の経路計画の比較を行なった。そこで、格子サイズ及び、発生するデマンド数を変え実験を行った。比較する手法は、以下の通りである。このとき全ての手法は前述したクラスタリングを行なっている。

1. 直線上の経路を設計し、その地点まで利用客に徒歩で移動させる経路
2. 利用客の徒歩の移動距離を減らすようバスに寄り道をさせる経路
3. Greedy 法により算出した経路に 2opt 法を適用し、利用客の徒歩での移動がない経路

手法 3 は TSP 等で利用される既存手法の一つであり、全ての出発点を巡回し、DEPOT へ輸送することを目的とする。そのため、クラスタリングを除けば単純な TSP であり、TSP を解く際に有効とされる、Greedy Algorithm と 2opt 法の組み合わせを用いて経路を生成する [5]。これを提案手法である手法 1, 手法 2 の各コストと比較することで、徒歩を組み合わせることの有効性を示す。

4.2. 評価値

実験では、利用客の平均旅行時間とバスの合計走行距離を比較する。利用客の平均旅行時間は、各利用客が出発してから DEPOT に到着するまでの旅行時間を算出し、その平均をとったものである。手法 1, 2 の場合、バスでの移動以外に徒歩での移動時間が含まれる。こ

の場合、徒歩とバスの速度の関係を考え、バスの移動時間を基準とし、徒歩の移動時間はその 4 倍¹⁾とした。

また、発生させるデマンドの起点は、全ノードから無作為に選んだノードとし、終点は DEPOT とした。DEPOT は、マップの左下隅に設定した。

手法 1 では、バスの片道の経路のみを生成しており、実際には利用客を送迎する為にその経路を往復する必要がある。よって、合計走行距離は作成した経路の長さの 2 倍とする。また、手法 2 も同様に片道のみを生成しているが、乗客を乗せない移動時に寄り道をする必要はなく、最初に乗車する利用客の乗車地点に到達するまでは最短経路を通るものとする。手法 3 は、バスが通過した距離の総和である。

4.3. シミュレーション設定

使用する仮想都市のマップは、都市全体を正方形、街路は格子状である。また、全ての交差点間は等しく 300m とした。実験は、 20×20 , 50×50 , 100×100 の都市の規模の違う 3 種類のマップを用い、デマンドの発生数はそれぞれ 30, 50, 100 とした。このとき、デマンドはバッチ処理方式で、全ての交差点上からランダムに発生する。

また、バスは全て 3 台の運行とし、速度は一律 20km/h、徒歩の速度は 5km/h と設定した。

4.4. 実験結果

図 4 は、シミュレーションを都市の規模ごとに 10 回ずつ行ない、バスの合計走行距離の平均をグラフに示した結果である。また、図 5 は同様に利用者の平均旅行時間をグラフで示している。

まず、提案手法と既存手法の比較を行なう。合計走行距離を見ると、既存手法は都市の規模に関わらず提案手法である手法 1, 手法 2 より高コストであり、都市の規模が大きくなることでより差が開いている。利用客の平均旅行時間を見ると、格子サイズが小さい時には、徒歩の移動が存在しない既存手法が最も短い、都市の規模が大きくなることによって、手法 1, 手法 2 が 10% 以上短くなっている。

次に、提案手法である手法 1 と手法 2 について着目する。最も小さい規模の都市では、手法 2 が手法 1 に比べ、わずかながらどちらの評価についても勝っている。しかし、都市の規模が大きくなると、合計走行距離は、手法 2の方がおよそ 10%ほど長く走行している。しかし、平均旅行時間は手法 2の方が約 7%ほど短い距離で移動できている。

4.5. 考察

都市の規模が小さい場合、提案手法は利用客の平均旅行時間において、比較的多くのコストがかかってしまう。これは小規模都市では、既存手法の経路にあまり複雑さが無く、徒歩での移動コストがバスの移動コストに比べて大きく影響しているからだと考えられる。

都市の規模が大きい場合、既存手法と比べ、平均旅行時間を 10% 以上短くすることができた。これは、都市の規模が大きい、または、使用するバスの台数が少ない等の状況では、既存手法は複雑な経路となり、効

¹⁾一般的なバスの速度は 15-20km/h、徒歩の速度は 4.8km/h とされている

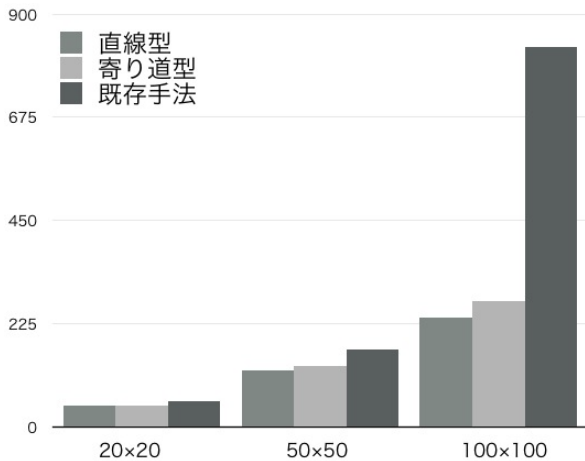


図 4: 各都市での合計走行距離の比較 (km)

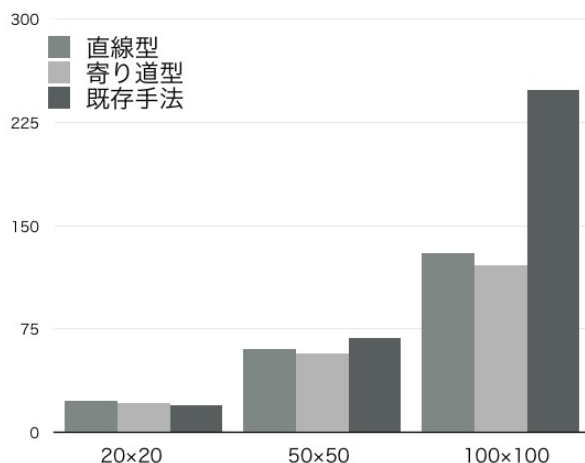


図 5: 各都市での平均旅行時間の比較 (分)

率的に輸送することができず、大きく時間をかけてしまうからだと考えられる。また、既存手法は、全ての利用客の乗車地点を巡回するため、経路が長くなる傾向がある。よって、利用客の位置により旅行時間が変化するという特徴が強く現れ、平均旅行時間にも影響を及ぼしているものと推測される。

提案手法は、利用客に徒歩で移動してもらうことにより、バスに乗車するまでの利用客の待機時間を並列化し、他の利用客の位置によって旅行時間が大きく左右されることはないことが実験により明らかとなった。よって、格子サイズやデマンド数に関わらず、比較的安定したコストで運用できると考えられる。ただし、小規模な都市の場合、合計走行距離は提案手法の方が短いですが、平均旅行時間に注目すると、既存手法を利用の方が効率的に輸送できている。このことから、提案手法は、全ての利用客の出発点を巡回するには複雑となる広い地域、またはデマンドの発生場所が一様に分布している場合、特に有効な手法だと言える。また、直線的な動きではなく、寄り道を取り入れることでさら

に平均旅行時間の削減を可能にするが、寄り道分の合計走行時間が大きくなる。平均的な旅行時間の短縮が出来ていることから、GDRP における徒歩の移動量は少なからず旅行時間に影響を与えていて、最適な経路を導くために様々な経路計画を試す必要がある。また、曲線を用いる場合でも、バスの走行距離と利用者の徒歩での移動距離の双方を小さくする為に適切な曲線係数を求めるシステムが必要である。

5. おわりに

本稿では、集約型デマンドバスにおける経路計画アルゴリズムの提案を行い、既存手法を含む 3 手法で比較を行なった。結果、集約型デマンドバスの経路生成問題において、提案手法は、規模の大きい都市ほど旅行時間やバスの走行時間の短縮が可能であることを明らかにした。最適な曲線係数を導くアルゴリズムの開発は必要であるが、寄り道を行うことで更なる旅行時間の短縮の可能性を示した。

また、今後は格子状道路網ではなく、放射状道路網や実在する道路網を用いた実験を行なう必要がある。

今回の実験では時間制約について考慮していないが、筆者らが提案しているデマンドバスシステムのアルゴリズム [4] を用いることで、希望到着時刻を反映することが可能だと考えられる。

参考文献

- [1] 内村圭一, 斉藤隆司, Takahashi Hiro, “公共交通サービスにおける Dial-a-Ride 問題”, 電子情報通信学会論文誌. A, 基礎・境界 J81-A(4), 599-606, (1998)
- [2] 原野貴裕, 石川孝, “協調型デマンドバスの有効性について”, 情報処理学会研究報告. MBL, [モバイルコンピューティングとユビキタス通信研究会研究報告] 2004(114), 135-139, (2004)
- [3] 小柴 等, 野田 五十樹, 山下 倫央, “実都市を対象としたシミュレーションによるデマンドバス評価”, 人工知能学会全国大会論文集 (CD-ROM) 27th ROMBUNNO.1D4-3(2013)
- [4] 上原 和樹, 赤嶺 有平, 遠藤 聡志, 根路銘 もえ子, “クラスタリングとデマンドバスを用いた大規模通勤システムについての検討”, 情報処理学会研究報告. ITS, [高度交通システム] 2014-ITS-56(3), 1-7, (2014)
- [5] 村野 剛教, 松本 直樹, “巡回セールスマン問題の近似解法の比較”, 電子情報通信学会技術研究報告. NLP, 非線形問題 102(625), 1-6, (2003)
- [6] J.-F. Cordeau and G. Laporte, “The dial-a-ride problem: Models and algorithms,” Annals of Operations Research, vol. 153, no. 1, (2007)