

フルデマンド型交通における迂回度低減のための 将来需要を考慮した配車アルゴリズムの提案

若園裕太¹ 白石陽²

概要：近年，MaaS の普及に伴い，乗合を用いた交通手段であるフルデマンド型交通が注目されている．フルデマンド型交通の問題の1つとして，新たなデマンドが発生し乗合が起こる際に，走行予定の経路を変更し迂回する必要があることが挙げられる．迂回が発生することで，既に乗車しているユーザの乗車時間や新たに乗車するユーザの乗車までの待ち時間が増加し，利便性が低下することが考えられる．そこで，新たなデマンド発生時の迂回度を低減するための配車アルゴリズムを構築することで，利便性の向上を図る．本研究では，フルデマンド型交通の1つである SAVS (Smart Access Vehicle Service) を対象として，将来需要を考慮した配車アルゴリズムを提案する．具体的には，SAVS の配車アルゴリズムである逐次最適挿入法を拡張し，各車両が保持する乗降地点リストに将来需要を考慮した経由地点を付与する．提案アルゴリズムの評価を行うために，逐次最適挿入法と提案アルゴリズムでシミュレーション実験を行った．実験結果に基づいて，ユーザの待ち時間や乗車時間，車両の乗車人数の変化，乗合率など複数の観点から，提案手法の有用性について分析を行った．

Proposal of a Vehicle Dispatching Algorithm Considering Future Demand for Reducing the Degree of Detours in Full-demand Responsive Transport

YUTA WAKAZONO¹ YOH SHIRAISHI²

1. はじめに

現在，情報通信技術の発展に伴い，都市部や過疎地域などの交通問題を解決するために，MaaS (Mobility as a Service) の普及が進んでいる．MaaS とは，あらゆる交通手段を統合し，1つのモビリティサービスとして提供する概念である[1]．MaaS が普及することで，独立した交通手段にアクセスするのではなく，1つにパッケージ化されたシームレスな交通手段にアクセスが可能になる．シームレスな交通手段が提供されることで，人々は独立した交通手段に逐一アクセスする必要がなくなり，より快適に移動できる世の中が実現することが期待される．

MaaS の導入にあたり，特に乗合を用いた交通手段であるデマンド型交通が注目されている．デマンド型交通とは，ユーザが送信するデマンドに応じてスケジューリングを行い，他のユーザとの乗合を前提に運行を行う公共交通である[2], [3]．デマンドとは，配車リクエストのことを指し，ユーザが指定する乗車地点や降車地点，乗車時刻，降車締切時刻，乗車人数などの要素から成る．デマンド型交通は，運行の柔軟性からいくつかの方式に分類することができる．

その中でも本研究では，フルデマンド方式のデマンド型交通に注目する．フルデマンド方式のデマンド型交通(以降，フルデマンド型交通)は，デマンド型交通の運行方式の中で最も柔軟性の高い運行方式である．停留所や走行経路，運行時刻などの制約は決まっておらず，任意の地点から任意の地点へドアツードアに移動することができる．そのため，タクシーと同様の運行の柔軟性を期待できる．タクシーと同様の運行の柔軟性を持つことから，運賃が高額になることが懸念される．しかし，ユーザが乗合を許容することで，バス以上タクシー以下の運賃でサービスを提供することが期待できる．タクシーと同様の柔軟性を持った公共交通を低運賃で利用可能になることで，様々な人々がより移動しやすくなる世の中を実現できると考えられる．

しかし，フルデマンド型交通にはいくつかの問題が挙げられる．その問題の1つとして，新たなデマンドが発生し乗合が発生することで，走行予定の経路を変更し迂回する必要があることが挙げられる．走行予定の経路から迂回することで，既に乗車しているユーザの乗車時間が増加することや，新たに乗車するユーザの乗車までの待ち時間が増加することで，結果として，ユーザの利便性を低下させてしまうことが考えられる．そこで，将来的なデマンドの発生状況(以降，将来需要)を考慮し，あらかじめデマンドが発生しやすい地点を経由するように配車を行うことで，

¹ 公立はこだて未来大学大学院 システム情報科学研究科
Graduate School of Systems Information Science, Future University Hakodate

² 公立はこだて未来大学 システム情報科学部
School of Systems Information Science, Future University Hakodate

デマンドを効率的に処理できるようになると考える。よって、本研究の目的をフルデマンド型交通における迂回度低減のための将来需要を考慮した配車アルゴリズムの構築とする。

本研究では、フルデマンド型交通の 1 つである SAVS (Smart Access Vehicle Service) に注目する。SAVS とは、ユーザが送信するデマンドに応じて、車両を完全自動でリアルタイムに配車するサービスである[4]。SAVS には、大きく 2 つの特徴がある。1 つ目は、デマンドの事前予約を必要とせず、随時受け付けることが可能である点である。これは、SAVS のアルゴリズムである逐次最適挿入法と配車手続きを完全自動で行えるシステムにより成り立っている。逐次最適挿入法とは、ユーザから送信されたデマンドに応じて、発生したデマンドと既に割り当てられているデマンドの降車締切時刻を満たす車両の中で、最もコストの低い車両にデマンドを割り当てるアルゴリズムである[5]。このアルゴリズムにより、少ない計算量で準最適な結果を算出することで、発生するデマンドへのリアルタイムな応答が可能になっている。デマンドの事前予約を必要とせず、いつでもデマンドを受け付けることができることで、ユーザの利便性の向上が期待できる。2 つ目は、各ユーザが自由に指定する乗降地点、降車締切時刻、乗車人数の制約下で配車を行っている点である。特に、ユーザが降車締切時刻を自由に指定できることは、サービスのユーザ層を拡大することにつながる。SAVS は、ユーザの指定する降車締切時刻によって、同じ乗降地点でも様々な対応パターンが存在する。そのため、ユーザは時間的観点や料金的観点から各々のニーズに合わせたサービスの利用が可能になる。これらの理由から、SAVS は今後普及していく公共交通サービスとして有望であると考えられる。

本研究では、将来需要を考慮した配車アルゴリズムを構築するために、SAVS の配車アルゴリズムである逐次最適挿入法を拡張する。逐次最適挿入法を用いて割り当て車両を決定し、割り当てられた車両の位置と各デマンドの乗降地点間に将来需要を考慮した経由地点を付与する。また、将来需要の高い地点への車両の供給過多を抑えるために、将来需要の高いいくつかのエリアに車両を分散させる。さらに、経由地点が付与されることで、既に乗車しているユーザの乗車時間を増加させないために、経由地点の探索範囲に制限を設ける。

2. 関連研究

関連研究として、まず 2.1 節では、デマンド型交通の分類について述べる。次に 2.2 節では、フルデマンド型交通の研究事例について述べる。そして、2.3 節ではフルデマンド型交通の配車最適化に関する研究について述べる。

2.1 デマンド型交通の分類

デマンド型交通は、運行方法や運行に使用する車両サイズ、運行の柔軟性など、複数の観点からいくつかの方式に分類することができる。例えば、デマンド型交通を運行の柔軟性から見た場合、定路線方式、迂回路方式、フレックスルート方式、フルデマンド方式の 4 つの方式に分類される。これら 4 つの運行方式を図 1 に示す。

柔軟性	方式	停留所	経路
低	A. 定路線方式 	有	固定
	B. 迂回路方式 	有	固定
	C. フレックスルート方式 	有	自由
高	D. フルデマンド方式 	無	自由

図 1 デマンド型交通の分類

1 つ目の定路線方式 (図 1 の A) は、走行経路、停留所ともに決まっており、最も柔軟性の低い運行方式である。路線バスやコミュニティバスに近い運行方式であり、予約があった場合にのみ運行を行う。定路線方式の利点は、乗車率が著しく低い状態を減らすことができる点である。

2 つ目の迂回路方式 (図 1 の B) は、走行経路、停留所ともに決まっており、さらに事前にデマンドがあった場合には、走行経路を延長する運行方式である。迂回路方式の利点は、限定的な需要しかないことで公共交通が運行していないエリアにも対応することができる点である。

3 つ目のフレックスルート方式 (図 1 の C) は、停留所のみ決まっており、デマンドに応じて自由に経路を走行する運行方式である。フレックスルート方式の利点は、移動時間の短さである。予約のあった停留所のみを通ることで、走行距離を短縮することができる。

4 つ目のフルデマンド方式 (図 1 の D) は、走行経路、停留所ともに決まっていない最も柔軟性の高い運行方式である。フルデマンド方式の利点は、タクシーと同様の運行の柔軟性を持ちながらタクシーより安い運賃で利用できることである。ユーザが乗合を許容することで、タクシーより安い賃金でドアツードアの移動を提供することができる。

デマンド型交通はこれら 4 つの運行方式から、目的や地域、ニーズに合わせた運行方式が導入されている。

2.2 フルデマンド型交通の研究事例

本研究で注目するフルデマンド型交通は、SAVS や MONET、コンビニクルなどを筆頭に各地域への導入や実証実験が行われている。また、フルデマンド型交通における様々な問題に着目した研究が盛んに行われている。まず、都市部での運用効率向上や迂回度低減を目的とし、都市部での将来需要を考慮した配車を行う研究[7], [8], [9]がある。

次に、フルデマンド型交通の実際の導入に向けて料金モデルの検討を行った研究[6]や空き車両の待機地点を最適化する研究[10]がある。また、フルデマンド型交通の問題点の1つとして、過疎地域での採算を得ることが難しいことが挙げられる。そこで、過疎地域での導入に向けた研究[11], [12]も行われている。さらに、女性専用車両などの制約条件を用いた際の影響を検証する研究[13], [14]なども行われている。フルデマンド型交通は、実社会への本格的な導入に向けて研究が盛んに行われている公共交通である。

2.3 フルデマンド型交通の配車最適化に関する研究

都市部での運行を対象とし、様々な観点や目的から配車最適化を行う研究[5], [7], [8], [9]がある。

野田ら[5]は、動的なフルデマンド型交通に求められる重要な指標である計算時間の短縮化を目的とした配車の準最適化を行っている。手法として、逐次最適挿入法というオークション形式のアルゴリズムを提案している。具体的には、ユーザから送信されたデマンドに応じて、送信されたデマンドと既に割り当てられているデマンドの降車締切時刻を満たす車両の中で、デマンドを割り当てた場合のコストを算出する。そして、割り当て時のコストが最も低い車両にデマンドを割り当てるアルゴリズムである。吉塚ら[15]は、逐次最適挿入法の計算速度の性能を検証するために全数探索法と逐次最適挿入法による計算時間を比較する実験を行った。実験結果から逐次最適挿入法の計算時間は全数探索法に比べ大幅に短いことが証明されている。さらに、逐次最適挿入法により発生した迂回は、全数探索法の精度から大きく劣ることがないという結果も示されている。逐次最適挿入法は、短い計算時間で準最適な結果を返すことに特化したアルゴリズムであり、さらに準最適化の精緻化を行う余地があることが考えられる。

Alonso-Mora ら[7]は、乗車するユーザ数を最大化することを目的とした配車最適化を行っている。手法として、将来需要を予測するモデルを構築し、デマンド発生時に予測モデルから仮想デマンドを発生させている。そして、仮想デマンドと実際のデマンドをつなぐ経路を生成することで将来需要を考慮している。この手法では、仮想デマンドに類似したデマンドが実際に発生しなかった場合に冗長な迂回が発生する可能性がある。そのため、迂回による実際のユーザへの悪影響を考慮する必要があると考える。

大社ら[8]は、乗車するユーザ数を増加させながら、車両の冗長な移動を減らすことを目的とした経路最適化を行っている。手法として、デマンドが割り当てられた車両に降車締切時刻を満たす範囲で、可能な限り将来需要量を満たす経路を生成する手法を提案している。ここで、将来需要量とは、ある時間に予測されるデマンドの発生数を指す。この手法では、将来需要量の高い経路に車両が偏る可能性がある。そこで、走行している車両台数に応じて将来需要

に対して車両を分散させることで、より効率の良い運行ができると考えられる。

Lin ら[9]は、運行者側の利益を最大化することを目的として、配車最適化を行っている。手法として、移動統計に基づき、将来需要を考慮した経路を生成するアルゴリズムを提案している。この手法では、運行者側の収益向上を目的にしているため、将来需要と共に期待できる収益を優先指標としたアルゴリズムになっている。そのため、ユーザの利便性が低下する可能性がある。

3. 提案手法

本研究では、フルデマンド型交通の1つである SAVS を対象として、SAVS の配車アルゴリズムである逐次最適挿入法を拡張して将来需要を考慮した配車アルゴリズムの構築を行う。3.1 節では、拡張元のアルゴリズムである逐次最適挿入法について述べる。そして3.2 節, 3.3 節では、逐次最適挿入法を拡張した提案アルゴリズムについて述べる。

3.1 逐次最適挿入法

本研究では、SAVS の配車アルゴリズムである逐次最適挿入法を拡張し、将来需要を考慮した経由地点を付与するアルゴリズムの構築を行う。そこでまずは、逐次最適挿入法(図2)による配車までの手順を以下に示す[5]。

STEP-1 新たにデマンドを受けた時に、そのデマンドを各車両に提示する。

STEP-2 各車両は、現在保持している乗降地点リストに受けたデマンドの乗降地点を任意に挿入する。全乗降地点を経由した場合のデマンドの達成予定時刻の遅延時間をデマンド挿入前の達成予定時刻と比較し、遅延の総和を算出する。そして新たなデマンドの達成予定時刻を算出し、達成予定時刻と遅延時間の総和を挿入後のコストとする。

STEP-3 乗降地点の挿入箇所、全てのペアの組み合わせについて STEP-2 の処理を実行する。全ての乗降地点の組み合わせの中で、最も移動コストが小さいものを受け入れる候補とする。ここで、新たなデマンドが挿入されることで間に合わないデマンドが1つでも存在する場合は除外される。

STEP-4 全てのペアの候補の中で、移動コストが最小になるペアを持つ車両に対しデマンドを割り当てる。

各車両間での乗り換えやシステム側からの乗降地点の変更は行わないものとする。また、乗車後の目的地の変更も考慮しないものとする。

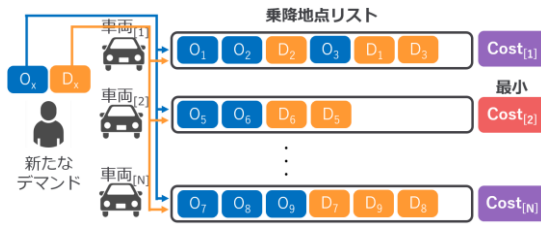


図 2 逐次最適挿入法の概要図

逐次最適挿入法は、短い計算時間で準最適な結果を出力することができるため、フルデマンド型交通において非常に有効なアルゴリズムであると考えられる。そのため、本研究では逐次最適挿入法を拡張元のアルゴリズムとする。

3.2 逐次最適挿入法の拡張方針

逐次最適挿入法の拡張方針を以下に示す。

- 既に乗車しているユーザの乗車時間を大きく増加させない制約下で将来需要を考慮する。
- 割り当てられた車両の位置と乗降地点リストの要素間に将来需要を考慮した経由地点を付与する。
- 将来需要の高い地点への車両の供給過多を抑えるために、将来需要の高い上位のいくつかのエリアに車両を分散させる。

3.3 逐次最適挿入法を拡張したアルゴリズム

逐次最適挿入法を拡張した配車アルゴリズムによる配車までの手順（図 3）を以下に述べる。事前準備として、運行するエリアを複数のグリッドに分割しておき、各時刻での将来需要を算出しておく。本稿では、将来需要データを過去のデマンドの発生数に基づくものとする。

- STEP-1 逐次最適挿入法により、割り当てられた車両の割り当て結果（割り当て車両 id, 乗降地点リスト, 乗降地点の通過予定時刻）を受け取る。
- STEP-2 デマンド割り当て決定時の車両位置と乗降地点リスト間の各要素間の位置、通過予定時刻を参照する。
- STEP-3 各要素を含むグリッドを参照し、通過予定時刻から考慮する将来需要を決定する。
- STEP-4 各要素が含まれるグリッドから成る矩形領域を経由地点決定のための探索範囲とする。矩形領域において将来需要が高い上位 3 グリッドを抽出する。この際、各要素が含まれるグリッドは含まないものとする。
- STEP-5 抽出された上位 3 グリッドからランダムに経由するグリッドを決定する。
- STEP-6 経由グリッドを通過しかつ、経由グリッドを付与した際の各要素を最短で通過できる経路を算出する。
- STEP-7 STEP-2～STEP-6 を全ての要素間に対して行う。

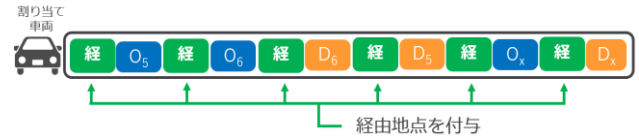


図 3 提案アルゴリズムの概要図

また、以下に示す条件時には、経由地点を付与しない。

- A) デマンド割り当て時に、割り当てられた車両に既に一定数以上、デマンドが割り当てられている場合。
- B) 経由地点の要素間の距離が 1km 以下の場合。
- C) 各要素が含まれるグリッドのどちらかの軸の座標位置が一致する場合。

まず A) に関して、既に複数のデマンドを割り当てられている車両には、経由地点を付与することで、迂回を発生させても乗合が発生することが期待できない。その理由は、デマンドにはユーザが指定する降車締切時刻があるからである。逐次最適挿入法は、降車締切時刻という制約を 1 つのデマンドでも満たさない場合には乗合が発生しない。そこで、既に一定以上のデマンドが割り当てられている車両には、経由地点を付与しない。

次に B) に関して、要素間の距離が近い場合、経由地点の付与に関わらず、乗合が発生した際の乗車時間の差は短いことが考えられる。そのため、提案アルゴリズムでは、要素間の距離が近い場合には経由地点の付与を行わない。

そして C) に関して、本稿の提案手法では、既に乗車しているユーザの乗車時間が大きく増加しないように経由地点を付与する方針である。各要素が含まれるグリッドのどちらかの座標が一致した場合、経路が変更されない限り要素間をつなぐ全てのグリッドを通過することになる。そのため、経由地点を付与する必要がない。

4. 実験

本章では、まず 4.1 節で実験目的について述べ、4.2 節では、実験環境について述べる。4.3 節では評価指標について述べ、4.4 節では実験結果とその分析について述べる。

4.1 実験目的

提案アルゴリズムによる有効性の検証を目的として、提案アルゴリズムと逐次最適挿入に基づいたシミュレーション実験を行う。また、3.3 節で言及した A) の割り当てられているデマンド数が 2 件より多い場合と 3 件より多い場合の 2 つケースで経由地点を付与しない条件で実験する。以後、割り当てられるデマンド数が 2 件より多い場合に経由地点を付与しないアルゴリズムを提案手法(2)、割り当てられるデマンド数が 3 件より多い場合に経由地点を付与しないアルゴリズムを提案手法(3)と表現する。

4.2 実験環境

本実験では、SUMO (Simulation of Urban Mobility)を用いてシミュレーション実験を行った。SUMO とは、オープンソースの道路交通シミュレータである[16]。SUMO には、外部プログラムとの連携を可能にする TraCI が備わっている。TraCI を用いることで、Python などのプログラミング言語で記述したプログラムと連携し、シミュレーション中の車両位置や通過予定経路などの情報を取得することで、車両挙動を動的に変更した。実験に用いる道路ネットワークは NetEdit, 逐次最適挿入法と提案アルゴリズムは Python を用いて構築した。

実験に用いた道路ネットワークを図 4 に示す。道路ネットワークは 5km 四方であり、交差点間の 1 辺の長さは 500m である。ノード数は 121, エッジ数は 440 である。図 4 中表示されている青色の逆三角形のマークは、車両待機地点を指す。シミュレーション開始時からデマンドが割り当てられていない車両は、車両待機地点に待機し続ける。また、デマンドを全て処理し空車になった車両は、次のデマンドが割り当てられるまでは、最後にユーザを降車させた地点に待機し続けるものとする。

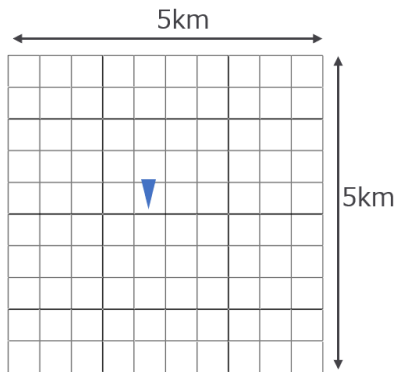


図 4 道路ネットワーク

本実験のシミュレーションのパラメータを表 1 に示す。本実験では、道路ネットワークを 1km 四方の 25 個のグリッドに分割し、グリッドごとに異なる発生確率を設定することでデマンドを発生させた。また、最初の実験として問題の複雑化を避けるために、降車締切時刻はデマンド発生から 5,000 秒、各デマンドで一度に予約できる人数は全て 1 名と単純なパラメータ設定にした。デマンドの発生パターンを図 5 に示す。図 5 中の数字は各グリッドのデマンド発生確率を示す。1,800 秒ごとにデマンドの発生パターンをパターン A, パターン B, パターン C の順に変動させ、デマンドの発生しやすいエリアを変化させた。また、1km 未満の距離のデマンドは発生しないものとした。

表 1 シミュレーションのパラメータ

パラメータ	値
シミュレーション時間	10,000 秒
発生デマンド	900 件
パターン A	300 件/[400~2,200 秒]
パターン B	300 件/[2,200~4,000 秒]
パターン C	300 件/[4,000~5,800 秒]
デマンドの距離	1km ~ 10km
降車締切時刻	デマンド発生から 5,000 秒
デマンド予約人数	1 人
車両台数	50, 75, 100 台
車両定員	4 人
最高速度	30km/h

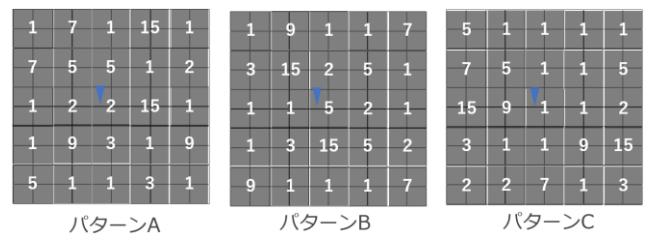


図 5 デマンドの発生パターン

4.3 評価指標

提案手法による実験結果への影響を評価するために複数の観点から実験結果を分析する。本節では、観点と各観点の定義について説明する。まず観点として、デマンド受理率や乗車までの待ち時間、乗車時間、達成時間、乗車人数の変化件数、走行距離から評価を行う。デマンド受理率は、デマンドを受け付けることができた割合を示し、受け付けることができたデマンド数を発生したデマンド数で割ることで算出している。待ち時間は、デマンドが発生してから実際にユーザが乗車するまでの時間を示す。乗車時間は、ユーザが乗車してから目的地に到着し降車するまでの時間を示す。達成時間は、各デマンドの待ち時間と乗車時間の和を示す。乗車人数の変化件数は、ユーザが乗車する時に車両の乗車人数が何人に変化するかの件数を示す。走行距離は、車両がユーザを乗車または、降車させるために走行した距離を示す。

4.4 実験結果

4.4.1 統計結果に関する分析

まず、各手法、各車両台数のデマンド受理率を表 2 に示す。表中の逐次は逐次最適挿入法、提案(2)は提案手法(2)、提案(3)は提案手法(3)を示す。表 2 から車両台数が 75 台、100 台の場合には全てのデマンドに対応することができているが、車両台数が 50 台の場合には、デマンド受理率が 7 割弱の数字を示しており、デマンドに対して車両を十分に

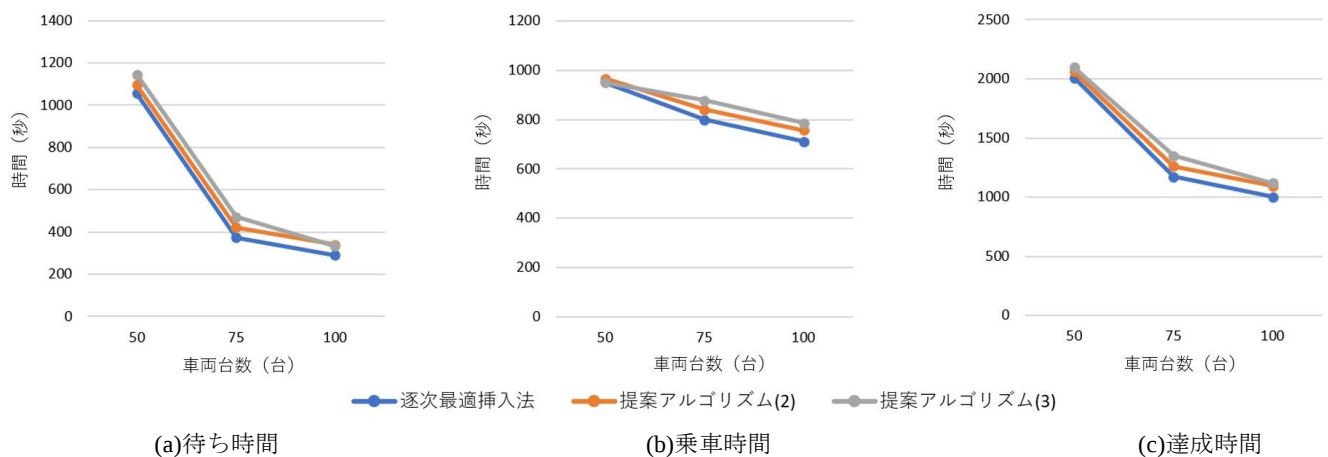


図 6 シミュレーション実験による各手法の統計データ

供給できていないことがわかる。また、車両台数が 50 台の場合では、各アルゴリズムにおいてデマンド受理率に大きな違いがないことが分かる。

表 2 各車両台数のデマンド受理率

手法	50 台	75 台	100 台
逐次	67.7%	100%	100%
提案(2)	66.7%	100%	100%
提案(3)	66.0%	100%	100%

次に待ち時間、乗車時間、達成時間の各平均を図 6 に示す。図 6 (a) が待ち時間、図 6 (b) が乗車時間、図 6 (c) が達成時間を示す。横軸は車両台数を示し、縦軸は時間を示す。図 6 (a) の待ち時間のグラフから、平均待ち時間は、逐次最適挿入法が最も短いですが各手法であまり差異がないことが分かる。図 6 (b) のグラフから、乗車時間も待ち時間と同様に逐次最適挿入法が最も短いことが分かる。また、車両台数が 75 台、100 台の時は 50 台の時に比べ乗車時間の差が大きくなっていることが分かる。図 6 (c) の達成時間から、特に車両台数が 75 台の時の逐次最適挿入法と提案手法の差が大きくなっていることが分かる。

4.4.2 シミュレーション中の乗車人数の変化に関する分析

デマンドに車両を十分に供給できている車両台数 75 台と 100 台のシミュレーション中の乗車人数の変化を表したものを図 7 に示す。横軸はシミュレーション中の時刻、縦軸は乗車人数ごとの割合を示している。乗車人数 0 人（未配車）は、車両にユーザが 1 人も乗っていない状態かつ、デマンドが 1 つも割り当てられていない車両を示す。乗車人数 0 人（配車）は、車両にユーザが 1 人も乗っていない状態かつ、デマンドが 1 つ以上割り当てられており、ユーザを乗車させるために移動している車両を示す。図 7 より、提案手法では車両台数 75 台、100 台の時どちらも同様に乗車人数が 2 人以上の割合が逐次最適挿入法に比べ多いこと

が分かる。これは提案手法により、車両を事前に将来需要の高いエリアに車両を誘導しておいたことで、乗合が発生しやすくなっていたことが考えられる。

4.4.3 各乗車人数の変化に関する分析

実際に乗合発生数を分析するために、車両台数が 75 台と 100 台における各乗車人数変化の件数を表したものを表 3、表 4 に示す。まず、表 3 から乗車人数(0→1)の件数は、逐次最適挿入法が最も多く提案手法の方が 20 件から 30 件ほど少ないことが分かる。乗車人数(1→2)の件数は、逐次最適挿入法と提案手法で大きな差異はなく、乗車人数(2→3)、乗車人数(3→4)の件数が逐次最適挿入法に比べ増加していることが分かる。次に表 4 から、車両台数が 100 台の場合には、75 台の時に比べ乗車人数(0→1)の件数が逐次最適挿入法と提案手法で差がさらに広がりさらに乗合件数が増えていることが分かる。また、車両台数 100 台の場合でも 75 台の実験結果と同様に乗車人数(2→3)、乗車人数(3→4)の件数が増加していることが分かる。逐次最適挿入法と提案手法で乗車人数変化の件数が違うことから、乗車人数の変化数ごとで待ち時間に違いがあることが考えられる。

表 3 車両 75 台における各乗車人数変化の件数

手法	乗車人数 (0→1)	乗車人数 (1→2)	乗車人数 (2→3)	乗車人数 (3→4)
逐次	255 件	366 件	213 件	46 件
提案(2)	236 件	361 件	240 件	63 件
提案(3)	223 件	357 件	244 件	76 件

表 4 車両 100 台における各乗車人数変化の件数

手法	乗車人数 (0→1)	乗車人数 (1→2)	乗車人数 (2→3)	乗車人数 (3→4)
逐次	442 件	357 件	93 件	8 件
提案(2)	391 件	362 件	126 件	21 件
提案(3)	388 件	348 件	139 件	25 件

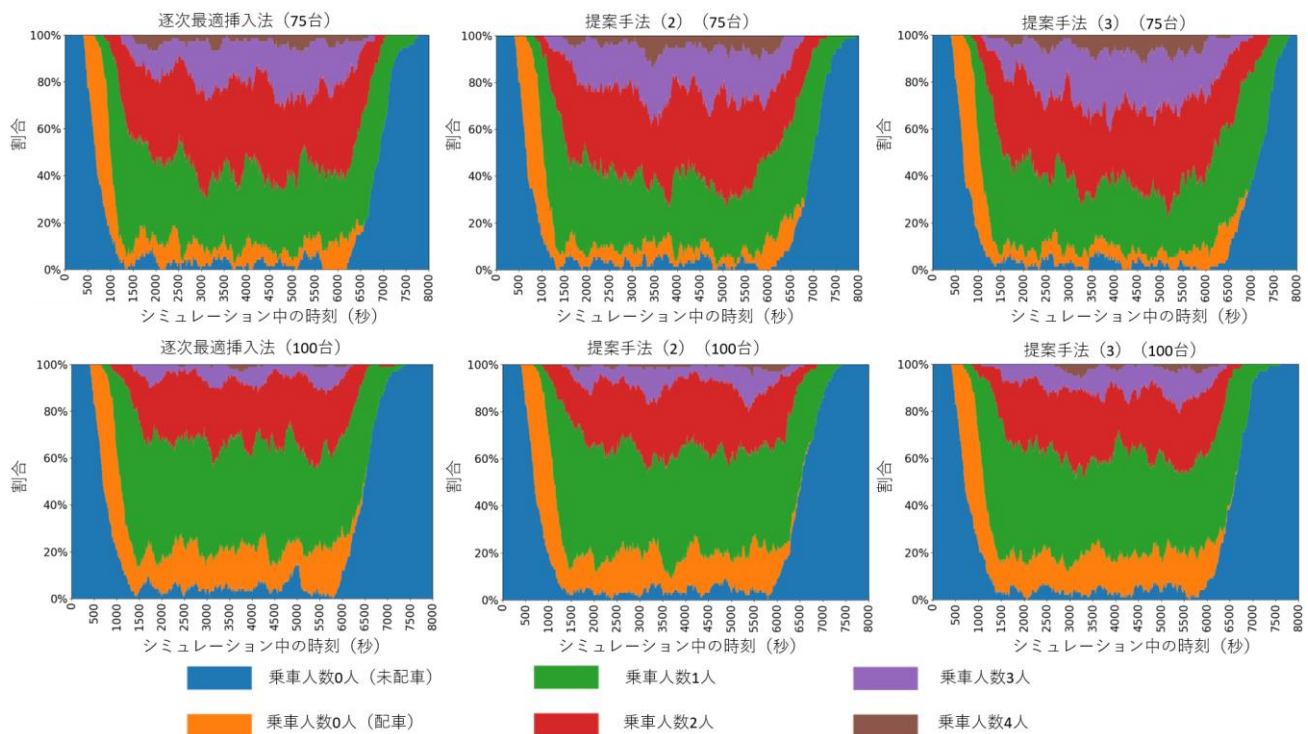


図 7 シミュレーション中の乗車人数の変化

そこで乗車人数の増加変化ごとの待ち時間を表したものを表 5, 表 6 に示す。表 5, 表 6 中の M は平均, SD は標準偏差を示す。まず, 表 5 の車両台数 75 台における各乗車人数変化の平均待ち時間から, 全体的に逐次最適挿入法の方が待ち時間が短いことが分かる。また, 提案手法(2)の乗車人数(1→2)と提案手法(3)の乗車人数の各変化数において標準偏差が逐次最適挿入法より大きいことが分かる。次に, 表 6 の車両台数 100 台における各乗車人数変化の平均待ち時間から, 乗車人数(0→1)と乗車人数(1→2)の変化時に, 待ち時間が増加しているが, 乗車人数(2→3)と乗車人数(3→4)の待ち時間の変化にはあまり差異がないことが分かる。また, 標準偏差は提案手法(2), 提案手法(3)のどちらも逐次最適挿入法に比べ大きい傾向にあることが分かる。

車両台数が 75 台, 100 台のどちらも提案手法の標準偏差が大きい傾向にある原因として, 3 つのことが考えられる。1 つ目は, 将来需要の高いエリアに車両を経由させることで, 提案手法にとって理想的な乗合が発生し円滑に乗車できたケースである。2 つ目は, 将来需要の高い地点への迂回が無駄になっているケースである。提案手法では, 将来需要の高い上位 3 グリッドからランダムにグリッドを選択している。そのため, 同じグリッドが複数回選択されると, 車両が供給過多になり, 他のエリアで発生したデマンドのユーザの待ち時間が長くなっていることが考えられる。また, 提案手法では既に乗車しているデマンドの性質や制約を考慮していないため, 乗合できない車両に迂回をさせていることが考えられる。3 つ目は, 将来需要の高いエリアを経由するために走行している過程でデマンドが割り当

てられることで, 冗長な誘導のみが発生してしまったケースである。提案手法では, 付与された経由地点を最短距離で通過するため, 乗降地点と付与された経由地点の間の将来需要は考慮していない。そのため, 経由地点に到着する前にデマンドが割り当てられることで, 提案手法による車両の誘導が有効に働いていない可能性が考えられる。

表 5 車両 75 台における各乗車人数変化の待ち時間

手法		乗車人数 (0→1)	乗車人数 (1→2)	乗車人数 (2→3)	乗車人数 (3→4)
逐次	M	460.5 秒	378.9 秒	280.2 秒	296.2 秒
	SD	289.7	278.0	241.0	292.4
提案 (2)	M	545.7 秒	403.7 秒	353.2 秒	312.5 秒
	SD	374.5	292.1	290.7	263.6
提案 (3)	M	471.8 秒	584.6 秒	396.1 秒	451.9 秒
	SD	444.0	323.6	346.6	355.7

表 6 車両 100 台における各乗車人数変化の待ち時間

手法		乗車人数 (0→1)	乗車人数 (1→2)	乗車人数 (2→3)	乗車人数 (3→4)
逐次	M	388.5 秒	202.5 秒	180.1 秒	179.0 秒
	SD	201.8	170.5	154.5	138.2
提案 (2)	M	436.5 秒	286.6 秒	212.0 秒	190.4 秒
	SD	256.9	214.0	184.1	212.1
提案 (3)	M	436.6 秒	274.4 秒	223.8 秒	157.9 秒
	SD	259.4	204.4	178.0	116.8

4.4.4 車両台数の走行距離に関する分析

車両台数が 75 台と 100 台における走行距離の違いを分析するために各手法の走行距離を表したものを表 7、表 8 に示す。表 7 から、車両台数 75 台では、逐次最適挿入法と提案手法(2)では、平均走行距離に大きな差異はないが、提案手法(3)では、平均走行距離が増加していることが分かる。また、提案手法(2)と提案手法(3)の標準偏差が大きいことが分かる。次に表 8 から、車両台数 75 台では、提案手法により平均走行距離が増加していることが分かる。標準偏差では、逐次最適挿入法の方が大きいことが分かる。

表 7 車両 75 台における走行距離

手法	平均	標準偏差
逐次	46,280.0m	3,041.1
提案(2)	46,933.3m	3,518.8
提案(3)	48,086.6m	3,762.4

表 8 車両 100 台における走行距離

手法	平均	標準偏差
逐次	41,795.0m	4,485.5
提案(2)	42,375.0m	4,243.4
提案(3)	42,655.0m	4,099.2

ここで各手法の走行距離の傾向を分析するために、車両台数 100 台における逐次最適挿入法の各車両の走行距離を図 8、車両台数 100 台における提案手法(2)の各車両の走行距離を図 9 に示す。まず、図 8 から、逐次最適挿入法では平均走行距離の値に近い走行距離の車両が多いことが分かる。次に図 9 から、逐次最適挿入法に比べ、提案手法(2)は走行距離が 40,000m 以下の車両台数が逐次最適挿入法に比べ多いことや、逐次最適挿入法には存在しない 50,000m 以上走行している車両が存在することから、ばらつきの大きいグラフになっていることが分かる。これは、4.4.3 項で言及したように提案手法により理想通りの乗合が発生した車両と冗長な迂回をした車両が存在することで、車両ごとに走行距離が大きな差異が現れたことが考えられる。

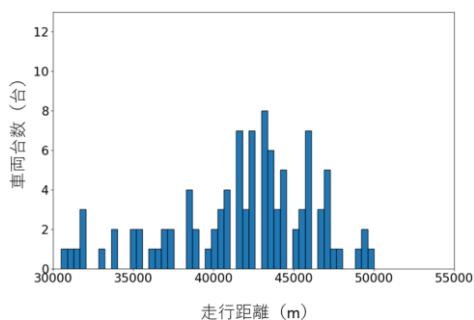


図 8 車両 100 台における逐次最適挿入法の走行距離

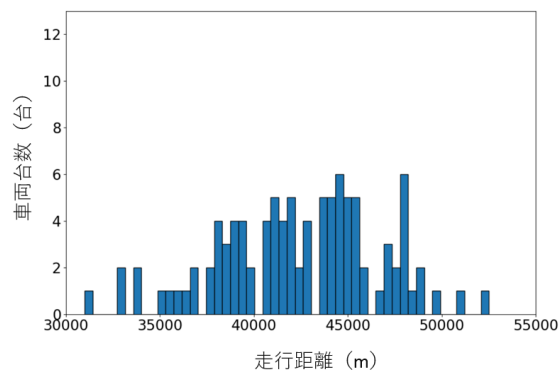


図 9 車両 100 台における提案手法(2)の走行距離

4.4.5 考察

これらの実験結果から、将来需要を考慮する際の制約の設定や既に割り当てられているデマンドの性質を考慮する必要があることが考えられる。現在の手法には特に 3 つの課題が存在することが分かった。

1 つ目に、将来需要を考慮する際に、将来需要の高い上位 3 グリッドからランダムに選択している点である。ランダムに選択されることで、一部のエリアに車両が偏ってしまうことや将来需要の低いエリアに車両が存在せずユーザの利便性低下することが考えられる。そこで、この問題を解決するために、将来需要を考慮する際に各車両が通過するグリッドを管理し、各グリッドに対して走行する車両の供給量と将来需要を合わせる必要がある。そうすることで、一部のエリアに車両が供給過多になることや全く車両が存在しないエリアが存在することを事前に防ぐことが可能になると考える。

2 つ目は、将来需要を考慮する際に、既に保持しているデマンドの特徴を考慮していない点である。現在は、将来需要をある時間単位のグリッドの発生数としている。しかし、この方法では、車両が実際に迂回し、理想通りにデマンドが発生した場合でも、迂回した車両が保持しているデマンドの降車締切時刻の制約を満たすことができない場合や他の車両にデマンドを割り当てた方がコストが小さい場合、逐次最適挿入法のアルゴリズムでは乗合が発生しない。そこで、将来需要の指標をデマンド発生数だけでなく、既に保持しているデマンドの特徴を踏まえた将来需要を考慮する必要があると考える。

3 つ目は、経路地点を付与する場合と付与しない場合の制約の設定方法にある。現在の手法では、デマンドを一定数以上保持している車両には、迂回しないような制約になっている。この制約の場合、各デマンドの降車締切時刻に対する時間的な余裕を一切無視しているという問題がある。各デマンドの降車締切時刻に対する時間的な余裕を詳細に分け、余裕のあるデマンドに対しては、経路地点を付与するアルゴリズムに改善する必要があると考える。

5. まとめ

本研究の目的は、フルデマンド型交通における迂回度低減のための将来需要を考慮した配車アルゴリズムの構築である。本論文では、逐次最適挿入法を拡張し、乗降地点リストに将来需要を考慮した経由地点を付与することで、将来需要を考慮し迂回度を低減する手法を提案した。実験結果から、提案手法により乗合数が増加することが見られたが、待ち時間や乗車時間、達成時間、走行距離に関しては改善を図ることができなかった。

今後の課題として、走行する車両の供給量と将来需要を合わせる方法や既に割り当てられているデマンドの性質の考慮方法、将来需要を考慮する際の制約の設定方法などの課題が挙げられた。これらの問題について検討しアルゴリズムの修正を行う必要があると考える。また、今回の実験環境の設定では、限定的な環境でしか実験を行っていない。デマンドの発生数や車両台数、デマンドの降車締切時刻などの実験パラメータを変えて実験、分析をする必要がある。さらに、発生した乗合の特徴や傾向の分析を行う必要があると考える。逐次最適挿入法と提案手法により発生した乗合には、特徴に違いがある可能性ある。そのため、実験結果をさらに細分化し、提案手法によるデマンドへの対応の変化を詳細に分析することで、アルゴリズムの改善を図っていく。

謝辞 本研究のシミュレーション環境の構築にあたり、公立はこだて未来大学大学の鈴木恵二研究室による支援と協力を受けている。ここに深く感謝の意を表する。本研究の一部は JSPS 科研費 JP20K11793 の助成を受けたものである。

参考文献

- [1] Sonja Heikkilä, Mobility as a Service - A Proposal for Action for the Public Administration Case Helsinki, Thesis for the degree of Master, School of Engineering, Aalto University, (2014).
- [2] 続・デマンド型交通の手引き, 国土交通省中部運輸局, https://www.tb.mlit.go.jp/hokkaido/bunyabetsu/tiikikoukyoukoutsuu/31manyaru/08zokudemandotebiki_tyuubu.pdf (最終アクセス日: 2021/05/07).
- [3] 田柳恵美子, 中島秀之, 松原仁, デマンド対応型公共交通サービスの現状と展望, 人工知能学会全国大会論文集, Vol.JSAI2013, pp.1-4 (2013).
- [4] 中島秀之, 小柴等, 佐野渉二, 落合純一, 白石陽, 平田圭二, 野田五十樹, 松原仁, Smart Access Vehicle System: フルデマンド型公共交通配車システムの実装と評価, 情報処理学会論文誌, Vol.57, No.4, pp.1290-1302 (2016).
- [5] 野田五十樹, 篠田孝祐, 太田正幸, 中島秀之, シミュレーションによるデマンドバス利便性の評価, 情報処理学会論文誌, Vol.49, No.1, pp.242-252 (2008).
- [6] 野田五十樹, 学習する利用者と進化する運行者による共進化型相乗りタクシー配車料金体系設計の試み, 情報処理学会研究報告知能システム (ICS), Vol. 2020-ICS-198, No.16, pp.1-7 (2020).
- [7] Alonso-Mora Javier, Waller Alex, and Rus Daniela, Predictive Routing for Autonomous Mobility-on-Demand Systems with Ride-Sharing, International Conference on Intelligent Robots and Systems, pp.3583-3590 (2017).
- [8] 大社綾乃, 大滝啓介, 小出智士, 西智樹, 相乗りのための将来需要を考慮した経路最適化, 情報処理学会論文誌, Vol.60, No.10, pp.1653-1661 (2019).
- [9] Lin Qiulin, Lei Den, Jingzhou Sun, and Minghua Chen, Optimal Demand-Aware Ride-Sharing Routing, IEEE Conference on Computer Communications, pp.2699-2707 (2018).
- [10] Fei Miao, Shan Lin, Sirajum Munir, John A Stankovic, Hua Huang, Desheng Zhang, Tian He, and George Pappas, Taxi Dispatch with Real-Time Sensing Data in Metropolitan Areas a Receding Horizon Control Approach. IEEE Transactions on Automation Science and Engineering, Vol.13, No.2, pp.463-478 (2016).
- [11] 政野博紀, 柴田直樹, Juntao Gao, 南和宏, 伊藤実, オンデマンドバスのための乗り換えを含むリアルタイムルートスケジューリング, 情報処理学会第24回マルチメディア通信と分散処理ワークショップ論文集, Vol.2016, pp.9-16 (2016).
- [12] 岩田聖, 鈴木恵二, 広域過疎地域に向けた SAVS のデマンド受理率に関するモデル化, 情報処理学会研究報告知能システム (ICS), Vol.2020-ICS-198, No.7, pp.1-6 (2020).
- [13] 岩田聖, 鈴木恵二, SAVS における制約を設けた乗り合い条件が与える影響の検証, 計測自動制御学会システム・情報部門学術講演会 2020 (SSI2020), pp.480-485 (2020).
- [14] 安彦久志, 松浦隆文, 属性を考慮した相乗りタクシーにおける送迎距離を制限した定式化の効果について, 情報処理学会研究報告数理モデル化と問題解決 (MPS), Vol. 2020-MPS-131, No.17, pp.1-6 (2020).
- [15] 吉塚裕生, 内田英明, 藤井秀樹, 吉村忍, ライドシェアサービス向け経路探索アルゴリズムの性能評価, 人工知能学会全国大会論文集, Vol. JSAI2018, pp.1-4 (2018).
- [16] Behrisch Michael, Bieker-Walz Laura, Erdmann Jakob, and Krajzewicz Daniel, SUMO – Simulation of Urban MObility: An Overview, International Conference on Advance in System Simulation, pp.50-60 (2011).