

# シェア型交通サービスの効率的な配車計画の提案

Efficient dispatching and routing of electric vehicles for ridesharing services

中央大学大学院理工学研究科情報工学専攻  
19N8100013L 毎田 明生

**要約：**近年、都市部では都市化に伴う渋滞や環境汚染、過疎地では不採算路線の増加などの問題を解決するため、デマンドバスが注目されている。しかし実用化されている多くのデマンドバスは、予約にオペレータが介入する必要があることなどから利便性はまだまだ高いものとは言えない。また、デマンドバスはライドシェアや相乗りなどを前提としたサービスであることから既存のタクシーなどのサービスでは起こりえない遅延が起きてしまう場合がある。この現状を受けて、自由な経路を持ち、最小限の遅延で送迎が行えるようなデマンドバスのシステムが求められている。そこで本研究では現在注目を集めている自動運転車を車両とするデマンドバスの効率的な配車計画を提案する。

**キーワード:**ライドシェア, 経路最適化, 割当て最適化

## 1 序論

モータリゼーションの過度の進展は特に都市部において交通渋滞を深刻化させ、旅行時間の増加や大気汚染、地球温暖化などの問題を引き起こしている。一方で、最近ではスマートフォンの大規模な普及と通信費の減少により、Uber や Lyft 社が主導するユーザー中心の MoD (Mobility on Demand) システムが開発されている。

MoD システムでは、顧客は必要なときに最適なモビリティやサービスを利用することができる。たとえば、Uber や Lyft では、アプリを用いて好きなタイミングで配車を依頼することができるだけでなく、食べ物などの荷物の配送が可能なサービスもある。また自動運転車が普及すれば、これまで様々な理由で運転ができなかった人も気軽に移動を行えるようになる。さらに、運転する必要がなくなったことにより、車内での過ごし方が変わり新たなサービスが生まれると期待される。

そのため今後、自動運転車の普及に伴い車両数が増加していき、都市部では渋滞や環境汚染の深刻化が加速していくという予測がある。このような問題に対してライドシェアや相乗りを利用したサービスは、複数人で同じ車両を利用することで個人で移動するよりも車両数を減らすことができるため、先のような問題への解決に貢献できると考えている。

## 2 問題定義

### 2.1 On-line Dial-A-Ride Problem

デマンドバスが解く問題は複数の乗客のリクエスト (いつ・どこで・何人で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 Dial-A-Ride Problem (DARP) と呼ばれている。(図 1)

図 1 の場合、各頂点に目的地情報を持ったリクエストが発生しており、頂点  $s$  では出発地  $s$  から目的地 1 の

リクエストが、頂点 1 では出発地 1 から目的地 2 のリクエストが発生している。同様に頂点 2,3,4 でもリクエストが発生しているとする。この時、DARP の解は車両の経路であり、図 1 の場合では赤い経路が 1 つの解として挙げられる。このとき、車両は頂点  $s, 1, 3, 2, 4, 5$  の順で移動し、頂点 3 から頂点 2 の間ライドシェアが行われている。

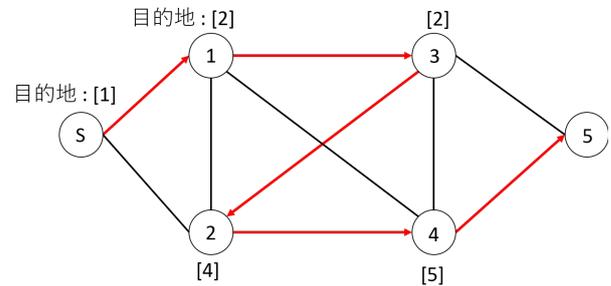


図 1 DARP

### 2.2 グラフについて

本研究では仮想空間で実験を行った。仮想空間は頂点集合を  $V$ 、辺集合を  $E$  とした  $G = (V, E)$  で表す。 $G$  は  $k \times k$  の格子状の有向グラフであり、各辺  $(x, y) \in E$  は移動距離  $d_{xy}$  をもつ。また、 $G$  の各頂点の 2 点間  $x, y$  には 2 つの辺が存在し、頂点  $x$  から頂点  $y$  に向かう辺  $(x, y)$  と頂点  $y$  から頂点  $x$  に向かう辺  $(y, x)$  が存在する。この 2 辺はどちらも同一の重みとする。この辺  $(x, y)$  と辺  $(y, x)$  は別々の辺とするため、辺  $(x, y)$  上にいる場合に辺  $(y, x)$  にリクエストが発生したとしても、その場で乗客を乗せる事はできない。

### 2.3 リクエストについて

乗客のデマンド情報 (リクエスト) について詳細を示す。リクエストは時間経過と共に発生し、発生したリクエストから順に番号  $R = \{r_1, r_2, \dots, r_N\}$  を与えていく。リクエストの発生間隔は一様ランダムに発生するため、同時刻に複数のリクエストが発生する場合もある。リクエストの数はリクエストとして認められるリクエストを合計  $N$  個発生しなければならない。

リクエスト情報の詳細について説明する。各リクエストの乗客人数はリクエストごとに異なり、最大乗車可能人数  $C^{pass}$  を超える乗客人数をもつリクエストは発生しない。リクエストの出発地点と到着地点は将来需要量グラフにもとづいて決定される。各乗客は出発到着時刻の他に遅延やズレを許容できる時間を設定する。決定されたリクエストの変更や分解などは認めない。リクエストの構成要素を表 1 に記述する。

表 1 リクエストの構成

構成要素	定義
リクエスト番号	あるリクエストの番号 $r_i (i = 1, 2, \dots, N)$
乗客人数	あるリクエストで乗ってくる乗客人数 $r_i^{pass} (\leq C^{pass})$
出発地点	あるリクエストの出発地点 $r_i^{start} (\in E)$
到着地点	あるリクエストの到着地点 $r_i^{goal} (\in E)$
出発希望時間	あるリクエストの出発希望時間 $r_i^{startTime} (nowTime < r_i^{startTime})$
到着希望時間	あるリクエストの到着希望時間 $r_i^{goalTime} (< \text{営業時間})$
出発時間枠	出発希望時間の前後で許容できる時間 $r_i^{startTW}$
到着時間枠	到着希望時間の前で許容できる時間 $r_i^{goalTW}$

## 2.4 時間枠について

時間枠について説明を行う。シェア型交通サービスはライドシェアを特徴にもつサービスであるため、新規リクエストの希望内容によっては、車両は最初の乗客が乗車した後に迂回をして新規リクエスト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に間に合わないことがある。

したがって、乗客には出発時間や到着時間に時間枠という幅を持ってもらうことで時間枠内の出発や到着を約束する。しかし新規リクエストが希望している時間枠内に到着や出発ができない場合にはそのリクエストを断るという形になるため、車両の効率的な運行が重要となってくる。時間枠の設定はリクエスト時に乗客がそれぞれ行う。出発時間枠は出発希望時刻の前後、到着時間枠は到着希望時刻の前に設定する。そのため、到着時間に関しては約束した時間に対して指定した時間枠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と考えている。

## 2.5 車両について

シェア型交通サービスの車両は自動運転車であり、自動運転 (SAE) レベル 4 とする。これは利用可能な SAE レベルの中でトップレベルのものである。加えて車両は個人では保有せず地域全体で保有しているものとする。

車両は  $m$  台あり全車共通のものとする。そのため各車両に乗れる最大乗車可能人数や移動速度、充電時間や容量に関しても共通している。送迎や充電のために車両が移動を行う際には必ず各車両は経路を持ち、その経路に従う。また乗客の乗降にかかる時間はないものとして考える。移動速度に関してはグラフの重みに対応しており、速度 1 は実験時間 1 に対してグラフの重み 1 移動することを表している。また経路を持たない車両はその場に止まる。

各車両は共通の最大燃料量  $F$  を持ち、移動を行うたび移動に応じた燃料が減少する。減少した燃料は充電によって増加させることができる。充電は充電スポットのみで行うことができ、各充電スポットはグラフ上に点在している。1 つの充電スポットで同時に充電できる台数は決まっており、最大充電台数  $P$  と呼称

する。最大充電台数は各充電スポットで共通であるため、現在使われている充電スポットを全車両は把握しており、一番近い空いている充電スポットに向かう行動をとる。

## 2.6 将来需要量グラフについて

シェア型交通サービスを効率的に運行させるにあたって、ライドシェアの成立数または成立率が重要になってくる。そのため、各車両はなるべく乗客が多い場所いわゆる繁華街や駅前などを運行経路として選ぶことでライドシェアの成立数の増加を見込むことができる。したがって、乗客の許容する範囲内で寄り道を行う経路を将来需要量グラフをもとに各車両は生成する。

将来需要量グラフは仮想空間としてのグラフ  $G$  と同じく 2 方向の有向グラフであり、各辺の重みが将来需要量となるグラフである。各辺の将来需要量の決定方法としては、将来需要が高いと見込める場所 (高需要地) をもとに決定される。高需要地とは繁華街や商業施設、駅などを想定している。この高需要地は辺上に設定され、周りの将来需要量よりも高い値で設定される。周りの辺はこの高需要地から波紋状のように将来需要量が減少する。

将来需要量の決定の詳細な流れとしては、高需要地のある 2 辺を 1 辺として、高需要地に与えられた値と等しい値を設定する。これを朝や昼、夜などの時間帯によって 2 辺に将来需要量を分けることで値を決定する。そのため、各辺の将来需要量は時刻 (実験時間) によって変化する。つぎに高需要地に接している周りの辺の値を決定する。先の 1 辺として与えた値より決められた減少量分を引いた値を周りの辺に与えていき、これもまた時間帯によって 2 辺に将来需要量を分けていく。これを繰り返し、全ての辺の将来需要量を決定する。しかし、高需要地はグラフ上に複数箇所あるので、将来需要量も高需要地の数の分決定されてしまうが、本研究では割り振られた中で一番値が高いものを各辺が採用する形とした。

## 3 リクエストの車両への割当て

本研究のアルゴリズムは大きく 2 つに分けられる。1 つは新規リクエストをどの車両に割当てるか、2 つ目は新規リクエストを含めた新しい経路の生成である。まずリクエストの車両への割当てについて説明する。

### 3.1 逐次挿入法の概要

本研究で用いた逐次挿入法という手法について説明を行う。一般にデマンドバスで最適な配分および経路探索をする問題は動的な巡回セールスマン問題と等しいものであり、完全な最適解を求めることは計算量の点で難しい。特にあらかじめリクエスト情報がわかっていない On-Line DARP では新たなリクエストを受け入れるにあたって、既存のリクエストの到着希望時刻を越えないように割当てや経路生成をする必要があるため、問題としてはより複雑になってくる。そこで本研究では On-Line なリクエストの割当てが求められる逐次挿入法を用いることとする。逐次挿入法とは各車両が持っている経路情報の全ての位置に新規リクエストの挿入を順に試行し、経路を決定する手法である。まず各車両の経路ごとに新規リクエストを挿入し、その車両で一番評価値が高くなる経路を決定する。その後各車両ごとの一番評価値が高い経路同士を比較し、一番評価値が高い車両に新規リクエストを割当てて

### 3.2 割当てについて

本節では、リクエストの車両への割当てについて述べる。リクエストの車両への割当てでは、無闇にリクエストを割当ててしまうと、ライドシェアという特徴を活かすことができず、ライドシェアを行わない割当てを行う可能性がある。したがって、本研究では空車の車両もしくは送迎中の乗客が乗っているがライドシェアできる状態の車両(乗車可能状態車両)に焦点を当て、これらの利用を活発に行えるような割当てを行う。

車両への割当てを行う際には各車両にスコアが必要になる。このスコアとは各車両の状態を表すものであり、新規リクエストが発生した際にどの車両に新規リクエストを割当てていくかを決定する数値となる。そのため、新規リクエストが発生すると各車両にリクエストの情報が行き渡り、それぞれの車両が新規リクエストを含んだ新しい経路を生成し、その経路をもとにスコアを計算する。スコアを計算するための経路生成に関しては次の節で紹介する。

本研究における割当て最適化の目的は短時間になるべく多くの乗客を送迎することであるため、以下のようにスコア  $S_{v_j}$  を計算した。表 2 に以下の式で使用している変数や記号の説明をまとめる。

$$S_{v_j} = \frac{\sum_{i \in R_{ok}} r_i^{pass}}{T_j} \quad (1)$$

表 2 変数および記号の説明

変数および記号	定義
$S_{v_j}$	車両 $v_j$ に対するスコア
$r_i^{pass}$	各リクエストの乗客人数
$R_{ok}$	車両に割当てられたリクエストの集合
$T_j$	各車両ごとの現在の時刻から空車状態になるまでの時間

式 (1) は単位時間当たりの乗客の送迎人数を表している。本研究は時間枠を超えてしまうリクエストは挿入されないため、車両に割当てられているリクエストのみを評価する。

## 4 将来需要を考慮した経路生成

将来需要を考慮した経路生成について説明を行う。各車両の経路は将来需要量によって決められていく。経路を生成するにあたって大きく 2 つの手順が存在する。1 つはリクエストの車両への割当てのスコアを決定するための初期解を構築すること、2 つ目はライドシェアの成立数を増加させるための近傍操作である。

初期解の構築は逐次挿入法を用いて行う。ある新規リクエストが発生すると逐次挿入法によって全ての車両の乗降順序に挿入される。各車両の乗降順序のなかで制約を満たす位置に新規リクエストを挿入し、その車両の乗降順序を決定する。このとき初期解は、各リクエスト同士を最短経路で結んだものである。この初期解に対して、将来需要を考慮した経路を生成し、各車両にとって将来需要量が 1 番見込める経路を提案する。

つまり、リクエストの車両への割当てで決められた乗降順序を最短経路で結んだものが初期解となり、この初期解に対して近傍操作を行い、将来需要が見込める経路を探索する。

### 4.1 制約条件

主な 2 つの制約について述べる。1 つ目は出発希望時間から作成された出発時間枠内に必ず車両はそのリクエストの出発地に到着し、出発しなければならないこと、2 つ目は到着希望時間から作成された到着時間枠内に必ず車両はそのリクエストの目的地に到着すること、この 2 つが各車両が経路決めるうえでの制約である。この 2 つの制約を満たさないリクエストはその車両には割当てず、全ての車両で満たさない場合にはそのリクエストを断る。

### 4.2 近傍操作

得られた初期解に対して近傍操作を行い、より将来需要量が見込める経路の探索を行う。各経路は将来需要量のグラフからその「経路の将来需要量」を計算する。経路の将来需要量の計算方法としてはその経路の辺に与えられた将来需要量の和である。

本研究で行った近傍操作について説明する。初期解として生成された経路順を  $route_v = \langle A, B, C, D, E \rangle$  とする。この経路  $route_v$  のどれかに接している頂点を  $p$  とし、 $p$  と接しているかつ経路  $route_v$  とも接している頂点もしくは経路  $route_v$  上の頂点を  $q$  とする。しかし  $q$  は  $p$  と接している経路上の頂点とは接してはならない。下の図 2、図 3 を用いて説明を行うと、今回は  $B$  と接している頂点を  $p$ 、 $C$  と接している頂点を  $q$  としている。この  $p$  と  $q$  がそれぞれ接している経路の頂点を  $p$ 、 $q$  と結び、結ばれた経路の頂点同士の辺または経路の一部を削除する。最後に  $p$  と  $q$  を結び新たな経路を生成する。下の図では、 $p$  と  $B$ 、 $q$  と  $C$ 、 $p$  と  $q$  を結び、 $B$  から  $C$  までの経路の一部または辺を削除する。 $q$  が複数あるときには、順に  $q$  を割当てて計算を行う。

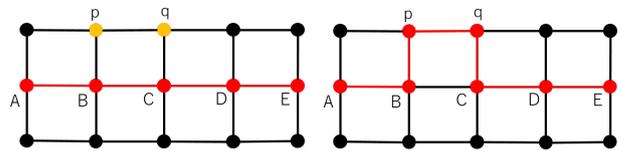


図 2 操作前

図 3 操作後

このような近傍操作を行い、制約条件を満たすものは実行可能解、満たさないものは実行不可能解とする。実行可能解となったものは、その経路の将来需要量を計算する。この操作を経路の頂点の全てに対して行い、最大の経路の将来需要量を持つ経路を探索する。全ての頂点の対して探索を終えると最大の将来需要量を持つ経路を初期解として更新し、これらの操作を繰り返す。近傍内の移動は最良移動戦略であり、即時の移動は行わない。

### 4.3 リクエストの待ち伏せ

また本研究では駅や商業施設などにあるタクシー乗り場を想定して、リクエストの待ち伏せを考慮した。待ち伏せを行える場所や台数は指定されており、車両は空車状態のみという条件付きであるが乗客の利用率の増加を見込むことが出来る。

## 5 計算機実験

### 5.1 実験準備

本実験でのグラフの頂点数は 100(10 × 10)、辺の数は 360 である。またグラフ上の各辺の重みは 1 か

ら5の間で一様ランダムに決定するが、向かい合う辺  $(x, y)$  と辺  $(y, x)$  は同一の重みとする。リクエストはグラフ上の辺に発生し、最大で400件、最小で200件発生する。リクエスト情報はリクエスト数や車両台数によって異なる。各車両の最大乗車可能人数は8人である。車両台数は6, 8, 10台を用意し各車両の移動速度は1とした。実験時間としては24時間を想定して1440を実験時間とする。

## 5.2 評価項目

本研究ではライドシェアを行わない通常のタクシーのような配車とライドシェアを行った配車、ライドシェアかつ将来需要を考慮した配車の3つを比較した。この3つを比較するため、リジェクト率と車両の平均移動距離を評価する。リジェクト率とは、全乗客に対しリジェクトされた乗客数の割合を意味する。車両の平均移動距離は全車両の移動距離を車両台数で割ることで算出する。将来需要を考慮する配車にはリクエストの待ち伏せも含まれる。ライドシェアを行わない場合の実装は車両が空車時のみ乗客が乗れるという制約に変更することで対応した。

## 5.3 リジェクト率の結果

まずリジェクト率を比較する。図4は車両台数6台、8台、10台を比べたものである。車両台数が増えるとともにリジェクト率が下がっていくことが分かる。また、どの車両台数の場合でも本研究の将来需要を考慮した手法が最も低いリジェクト率を実現しており、適切に将来需要が考慮されていることが分かる。

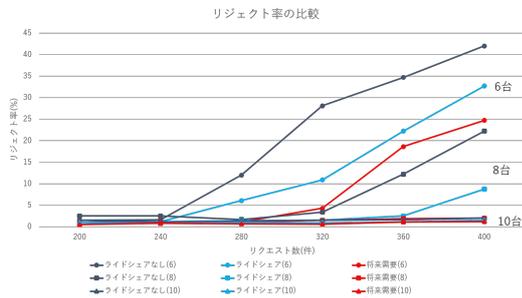


図4 リジェクト率の比較

## 5.4 車両の平均移動距離の結果

次に各車両の平均移動距離を比較する。図5に平均移動距離の結果を記す。これにおいてもすべての車両台数で本研究の将来需要を考慮した手法が最も短い移動距離を実現することができた。

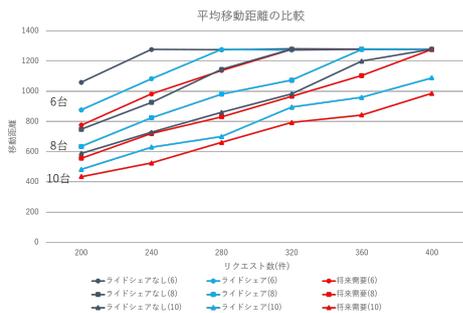


図5 平均移動距離の比較

## 6 考察

リジェクト率の考察から行う。リジェクト率はその車両台数でリクエストを補えなくなった場合に大きく増える傾向がみられる。そのためリジェクト率が抑えられているときは、そのリクエスト数に対して車両台数が十分であることがわかる。また処理したリクエスト件数に注目すると、リクエスト数400件の車両台数6台のライドシェアなしが232件、ライドシェアありが269件、将来需要を考慮したものが301件とライドシェアなしのもの比べてそれぞれ約1.15倍、約1.29倍とわずかながらライドシェアによる処理件数の増加が確認できた。リクエスト数は400で統一していることからリクエスト1件あたりの移動距離が低下したものだと考えられる。他の車両台数についても同様の傾向がみられた。車両台数10台の場合では他の車両台数と比べて横ばいの傾向があり、リクエスト数に対して車両台数が十分であったことを確認できた。

次に平均移動距離の考察を行う。全体的な傾向としてはリクエスト数に応じて平均移動距離が増加していることがわかる。移動距離1300付近で移動距離の増加が止まるような傾向がみられるが、これは車両の充電時間によるものが大きいと考えられる。それ以外の要素としてはリクエストが挿入されるタイミングが悪く、わずかな時間空車状態になっていることも考えられる。また、平均移動距離が1300付近のものリジェクト率をみると、ほとんどのものがリジェクト率が高く、車両台数に対してリクエストが多く、1日の営業時間ほとんど休むことなく移動していることが確認できる。将来需要を考慮した手法は待ち伏せも行っており、送迎後に高需要地に移動するため、平均移動距離が増加すると予測していたが、実際には低く抑えることが出来た。他の2つの手法ではリクエストが経路に挿入されていない(空車)状態では移動は行わずその場に止まるが、リクエストが発生した時には出発地に向かう必要がある。そのため、出発地に向かうまでの距離は将来需要を考慮しない2つの手法が短い、将来需要を考慮する手法のその後のライドシェアによる距離の短縮が大きい影響を与えていると考えられる。

## 7 まとめ

本研究では、自動運転車を想定したデマンドバスサービスであるシェア型交通サービスの効率的な配車計画を提案するために、将来需要を考慮する経路生成や待ち伏せを取り入れた手法やライドシェアを積極的に利用していく割当てについて検証を行った。結果として、現在のタクシーやデマンドバスサービスを想定したものよりリジェクト率と平均移動距離を減らすことができた。今後の課題としては、より現実に近いリクエスト発生モデルの作成や車両ごとではなく全体の最適化、充電スケジュールの最適化が挙げられる。

## 参考文献

- [1] 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.
- [2] 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.