卒 業 論 文

時間枠及び乗車時間ペナルティ付き
乗合タクシー問題に対する局所探索法

051600141　　竹田陽

名古屋大学情報文化学部自然情報学科

数理学専攻

2020年1月

時間枠及び乗車時間ペナルティ付き
乗合タクシー問題に対する局所探索法

051600141　　竹田 陽

概 要

乗合タクシー問題 (Diai-a-ride problem) とは, 異なる移動需要を持つ顧客を同じ車両で同時に輸送する際に, 効率の良い車両の割り当てとスケジューリングを考える問題である. 乗合タクシー問題では, 顧客の乗降点と到着時刻に対する時間枠制約，乗車時間に対する制約が与えられたときに, 車両の運用にかかるコストおよび顧客の不満度合いを最小にすることを目的とする.

乗合タクシー問題の制約としては, 車両の容量資源の最大容量を超えて乗車することはできない (容量制約), 車両が回るルートの大きさ (最大ルート距離)，顧客が車両に乗っている時間の長さ (最大乗車時間) が与えられる.

本研究では，時間枠と乗車時間に対する制約を凸なペナルティ関数として与えて，その問題に対して局所探索を行った.

Local search algorithm for Dial-a-ride problem with convex time window penalty

051400141 Kiyoshi Takeda

Abstract

The Dial-a-Ride Problem (DARP) consists of designing vehicle routes and schedules for multiple users who specify pickup and delivery requests between origins and destination. The aim is to design a set of minimun cost vehicle route while accommodating all requests. Side constraints include vechile capacity, route duration, maximum ride time, etc. DARP arises in share taxi service or transporting people in health care service.

A number of papers about DARP has been published. most of the previous research deal with time windows and maximum ride time as hard constraints.

In this study, we consider with the DARP where time windows and maximum ride time are soft constraints. These soft constraints make the DARP more generic. We propose iterated local search algorithm to find a solution to the problem.

目 次

第1章　はじめに

近年, 乗合タクシーサービスは新しい移動手段として需要が増加してきている．また, 老人や介護が必要な人を自宅などからヘルスケアセンターなどまで輸送するようなサービスも, 高齢化に伴い需要が増加してきている．これらのサービスの特徴としては, 利用者はリクエストとして出発地と到着地, それぞれに対して場所と時間枠を指定することができる．本研究では, これらのようなサービスにおいて利用者の満足度を考慮しつつサービスの実行にかかるコストを最小化することを考える．このような問題は乗合タクシー問題 (Dial-a-Ride problem, DARP) と呼ばれる．

乗合タクシー問題に対しては，多くの研究がなされている．多くの先行研究では, 利用者の最大乗車時間とリクエストの乗車時間と降車時間に対しての時間枠をハード制約として与えている．本研究では, 時間枠及び乗車時間をペナルティ関数で与えてソフト制約とする．こうすることで, 時間枠より少しの遅延は許容できる場合など, 様々なケースを柔軟に考慮することができるようになり, 先行研究より汎用的な問題とすることができる．

3 章では pickup and delivery problem(PDP) を紹介し，乗合タクシー問題 (DARP) について詳しく説明する．

静的 DARP に関して, 先行研究において様々な手法が提案されている．Jaw らは, この問題に対して近似解法としてルートに挿入した時の目的関数値の増加が最小になるようなリクエストを選択してルートに挿入していく連続挿入法を提案した [2]．Cordeau らは, あるルートからリクエストをひとつ取り除き, 別のルートに挿入する際にタブーサーチ探索を用いる手法を提案した [1].

第2章　問題定義

この章では，まず pickup and delivery problem(PDP) について説明し．そのあとに乗合タクシー問題 (DARP) について詳しく説明する．

2.1　Pickup and delivery problem

Pick and Delivery problem (PDP) は, 荷物を出発地まで目的地まで輸送する問題である．PDP では, 荷物を受け取る地点 (pickup) と荷物を配送する地点 (delivery) のペアからなる n 個のリクエストと m 台のが与えられたときに以下の制約を満たしつつ与えられたコストを最小化するルートを求める問題である．

1. 与えられた n 個のリクエスト全ての地点を訪問する．
2. 全ての車両はデポから出発してデポに帰る．
3. リクエストでペアになっている出発地と目的地は同じ車両が訪問する．
4. それぞれのリクエストにおいて, 必ず出発地を訪問した後に目的地を訪問する．

ここでのリクエストとは，荷物の輸送要求のことを指す．与えられるコストとしては各頂点間の距離や時間などがある．よく制約として考えられるものは, リクエストの訪問点を決められた時間内に訪れなければならない時間枠制約や車両の容量を制限した容量制約などがある．また, 車両の種類が複数あり，車両によって最大容量が異なる多資源制約や，複数のデポを考慮する問題など，様々な拡張が考えられている．

2.2　乗合タクシー問題 (Dial-a-ride problem)

乗合タクシー問題 (Dial-a-ride problem,DARP) は, PDP を人の輸送に特化した問題である、DARP は PDP と違い人を輸送するため, 車両に乗っている乗車時間が長いと利用者の不満がたまってしまう．そこで乗車時間やリクエストの訪問時間のずれなどで評価される不満度を考慮する必要がある．

DARP は, 使用する車両数を 1, リクエストにおける出発地を全てデポとする巡回セールスマン問題 (traveling salesman problem, TSP) ととらえることができる．TSP は NP 困難 [3] であることが知られているため, DARP も NP 困難である．

本研究では, 利用者が乗車時刻, 降車時刻, 乗車時間のそれぞれに対して希望を持つことを考える．それぞれの希望はそれぞれ連続区分線形凸関数のペナルティ関数で表される．乗降時刻にこのように制約を与えることで, 少しの遅延を許容するなど表現が可能になる．また, 乗車時間に応じてペナルティをかけることができるので, 不満度を柔軟に表現することが可能になる．したがって, DARP をより汎用的に解くことが可能となる．

事前に全てのリクエストがわかっている問題を静的 DARP, リクエストが全てはわかっておらず問題を解く過程でリクエストが次々と与えられる問題を動的 DARP といい, 本研究では静的 DARP を考える.

第3章　定式化

第4章　提案手法

提案手法を書く際には，

1. まず手法の大まかなアイデアや全体像を言葉で簡潔に説明し，
2. 次に手法の構成要素のおのおのの詳細を述べ，
3. 最後にそれらの構成要素をどのように組み合わせて全体の枠組みが構成されているのかを示す

というような順序で書くと分かりやすいと思います．その際，各構成要素を，サブルーチンのように入力として何を受け取って何を返すのかを記述した手続きとして名前をつけてまとめておき，最後に全体の枠組みを示す際にそれらを利用してアルゴリズムを記述すると書きやすいと思います．

このような手続きを疑似コードとしてまとめるのに `algorithmic` が便利です（`algorithmicx` も便利のようです）．Algorithm 1 に例を示します．

Algorithm 1 Ford-Fulkerson

Require: グラフ $G = (V, E)$，始点 $s \in V$, 終点 $t \in V$, および各辺 $e \in E$ の容量 u_e.
Ensure: s から t への最大フロー.

```
for e = 1 to |E| do
    x_e := 0 とする.
end for
残余ネットワーク G_x を作成する.
while G_x にフロー追加路が存在する do
    フロー追加路に沿って x にフローを追加する.
    残余ネットワーク G_x を更新する.
end while
フロー x を出力して終了.
```

なお，手続きの入出力を表す「`\REQUIRE`」と「`\ENSURE`」を使うと，通常はそれぞれ「**Require:**」と「**Ensure:**」のように表示されますが，これらを上の例のように「**Input:**」と「**Output:**」に変更するための記述が本 LaTeX ファイルのプリアンブル[1]にあります．

手続きをまとめるのに以下のようなスタイルを使うこともあります．なお，Step 番号を `description` 環境で手動で書いても同様のスタイルを実現できますが，Step を追加したり削除したりしたときに番号の相互参照で失敗しないよう，自動的に番号を振る方法の方がよいと思います．以下の例では `enumerate` 環境を利用していますが，プリアンブルに`\usepackage{enumerate}`が必要です．

[1] `\documentclass` と`\begin{document}`の間の部分．

Algorithm Ford-Fulkerson

Input: グラフ $G = (V, E)$ ...（略）

Output: s から t への最大フロー.

Step 1. すべての $e \in E$ に対し $x_e := 0$ とする.

Step 2. 残余ネットワーク G_x を作成する.

Step 3. G_x にフロー追加路が存在しなければ，フロー x を出力して終了.

Step 4. フロー追加路に沿って x にフローを追加.

Step 5. G_x を更新したのち Step 3 に戻る.

どのようなスタイルを使う方が書きやすいか，分かりやすいかは，アルゴリズムによっても変わりますし，好みもあると思いますが，アルゴリズムの反復構造は `algorithmic` の方が分かりやすいように思います.

疑似コードを読まなくてもアルゴリズムのアイデアが大体分かるように本文の文章中に説明を書きましょう．疑似コードはアルゴリズムを正確に記述するためのものなので，複雑になりがちです．従って，よほど内容に興味がある読者でなければ疑似コードまで読みたいとは思いません．読者が疑似コードを読まなくてもアイデアの概要が本文から分かるように書いてなければ，とても読みづらい論文になってしまいます.

第5章　提案手法2

第6章　計算実験

実験結果を示す際には，計算環境（実験に用いた計算機の CPU やメモリー，実装に用いた言語）を明記しましょう．計算結果を表示するのに用いる図や表を表示する例を図 6.1 と表 6.1 に示しておきます．また，今回は 2 段組ではありませんが, 2 段組みの原稿において幅の広い表を表示する例を表 6.2 に示しておきます（図も同様に`\begin{figure*}`のようにすればよい）.

図 6.1: 図の表示例

表 6.1: 表の表示例

問題例	最良値	計算時間（秒）
c05100	123	10.1
c10100	456	15.2
c20100	789	20.3

表 6.2: 2 段組みスタイルにおいて幅の広い表を表示する例

	既存手法		提案手法	
問題例	最良値	計算時間（秒）	最良値	計算時間（秒）
c05100	123	10.1	111	10.0
c10100	456	15.2	432	15.0
c20100	789	20.3	765	20.0

第7章　まとめ

まとめは著者から読者への締めくくりの言葉です．すなわち，「この論文のポイントは結局何だったのか」を端的に読者に示す大切な部分です．必ず書きましょう[1]．結局何をしてどうなったのかということを最後にもう一度手短にまとめて述べます．この節で新しいこと（つまりこれまでの節で書いて来なかったこと）を書いてはいけません．

通常の論文等ではこのように得られた成果をまとめた結論を書いて終わるのですが，研究の進捗状況を報告する普段の発表では，結論を書くことは難しいかもしれません．また，既に一定の成果が得られている場合でも，卒論・修論の締切が差し迫った時期を除き，卒論・修論に向けてさらに研究を進める予定であると思います．そのような場合には，この節のタイトルを例えば「まとめと今後の研究計画」などとして，まず現在までに得られている成果をまとめたのち，今後の研究計画を簡潔に書いてください．

[1] レター（ページ数の少ない速報的な論文（e.g., Operations Research Letters, Information Processing Letters））などの短いものではまとめの節を書かないように指示されることもあり，そのような場合を除く．

謝辞

お世話になった先生方に言葉では伝えきれないほどの感謝の念を伝えましょう．本研究の遂行にあたり, 熱心な指導と助言を頂きました柳浦睦憲教授に深く感謝の意を表します．提案手法の検討やその有用性において活発に議論を頂きました, 橋本英樹准教授, 胡艷楠氏, 呉偉氏に大変お世話になりました．深くお礼申し上げます．日々の研究室生活においては柳浦研究室の皆様にお世話になりました．夜を徹して行った麻雀ではその最適戦略の見極めるために大変有意義なものでした．卒業旅行を楽しみにしています．皆様のおかげで有意義な研究活動に勤しむことができました．深くお礼申し上げます．

参考文献

[1] Jean-François Cordeau and Gilbert Laporte. A Tabu search heuristic for the static multi-vehicle dial-a-ride problem. Transportation Research Part B: Methodological. 37. 579-594. 2003.

[2] Jang-Jei Jaw, Amedeo R Odoni, Harilaos N Psaraftis and Nigel H.M.Wilson A heuristic algorithm for the multi-vehicle advance request dial-a-ride problem with time windows. Transportation Research Part B: Methodological. Vol.20, No.3, pp. 243-245, 1986.

[3] 柳浦睦憲, 茨木俊秀, 組合せ最適化 メタ戦略を中心として, 朝倉書店, 2001