

選択停車型を用いた旅客総旅行時間の数理的最適化

森 拓哉* 古関 隆章 (東京大学)

Mathematical optimization of passenger total travel time using select stop
Takuya Mori*, Takafumi Koseki, (The University of Tokyo)

Abstract

Stop pattern is determined for each type of train in traditional railway. In this paper, we consider every one of the stopping pattern that minimize a total travel time based on OD (origin-destination) table. We call this “select stop”. Select stop can contribute to peak cut by shifting the timing of accelerating and braking. We introduce the study of select stop calculated by mixed integer programming, the results and the problem.

キーワード：数理計画，混合整数計画法，選択停車，総旅行時間，運転整理
(Mathematibal Programming, Mixed Integer Programming, Select Stop, Total Travel Time, Rescheduling)

1. 背景と目的

鉄道の大きな特長の一つに定時運行率の高さが挙げられるが、ホームドアの設置など様々な設備の改良が講じられているにも関わらず、図 1.1.に示すように輸送障害は近年増加傾向にある。この輸送障害により乱れた列車ダイヤを調整する業務を「運転整理」と呼ぶ⁽¹⁾。

運転整理は安全上の厳しい運行ルールや設備上の都合による数多くの制約を満たしつつ、様々な運転整理手法を適切に組み合わせる非常に困難な作業である。近年、運行管理システムの導入が進み、平常時の運行管理に関しては自動化が進んでいるが、その一方で運転整理については熟練指令員の経験と勘により行われており、鉄道事業者にとって大きな負担となっている。このような背景から、計算機により運転整理案作成を自動で作成するシステムの開発が強く望まれている⁽²⁾。

筆者らは先行研究⁽³⁾⁽⁴⁾として、混合整数計画問題として運転整理の問題を解釈し、旅客損失に主眼を置いた最適な運転整理支援システムを開発してきた。一般的に運転整理では「臨時停車」は行ってもよいが「臨時通過」は行っていないという暗黙のルールが存在する⁽⁵⁾。しかし、現在 ICT (Information and Communication Technology) の発達により、利用者は列車の運行情報を手早く入手でき、また事業者側は乗客のリアルタイムの需要を把握できる可能性が見えてきた。そこで、運転整理の場面において利用者の目的地情報においては利用客の需要に応じた選択停車列車の運用が考えられる。この選択停車列車は力行と制動の

回数を減らすことにより省エネルギーに貢献することが考えられ、力行のタイミングをずらすなども考慮すればピークカットが実現できる可能性もある。運転整理と電力を組み合わせた議論は、たとえば参考文献⁽⁶⁾などがあるが、ほとんど行われていないことが現状である。そこで本論文では今まで当研究室で行われてきた混合整数計画問題で選択停車を実現するための基礎検討を行い、その結果を報告するものとする。

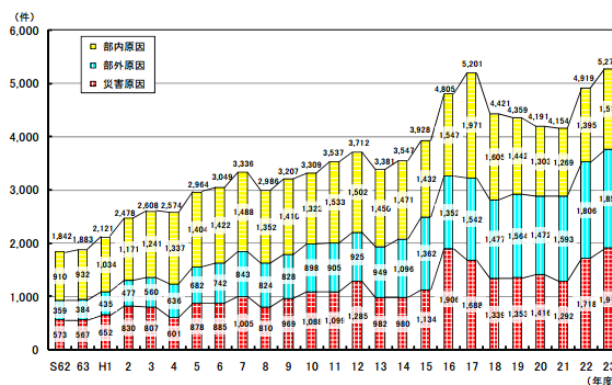


図 1.1. 日本における輸送障害の件数の推移⁽⁷⁾
Fig.1.1. The transit of the number of transport disorder in Japan

2. 選択停車の定義

郊外から都心に向かう通勤路線においては、図 2.1.のように乗客数の多い特定の駅のみ停車する優等列車を混在させ、全ての駅に停車する各停を途中駅で接続を取りながら、乗客が乗車駅から目的駅まで早く着くように工夫されているものが多い。

一方選択停車型のダイヤは複数の列車に停車駅を分担させ、1本1本の列車の高速化を図ると同時に、列車の配列を工夫することにより高密度性を確保するような運転形態を指すことが多い。その例を図 2.2.に示す。

しかし、今回この論文で扱う選択停車は、上記のように高密度を狙うものではなく、それぞれの列車を需要に応じて自由に停車させる、いわば「自由停車型」と言えるものを想定する。詳しい列車の停車条件などは次章に述べるが、このようにすることにより、列車を長時間特定の駅に停車させることなく、目的で述べたように力行のタイミングをずらすことでピークカットが実現できる可能性がある。また、運転整理の目線からは開通直後の混雑の偏在化を防ぐことができるなどのメリットが挙げられる。ただデメリットもあり、普段存在しない停車パターンで運転することになるため、旅客への細かな案内が必要とされる。ここに関しては駅構内の案内情報システムの高度化や、広く普及しつつある情報通信端末に期待し、本論文ではその前提にたつて話を進めていくこととする。

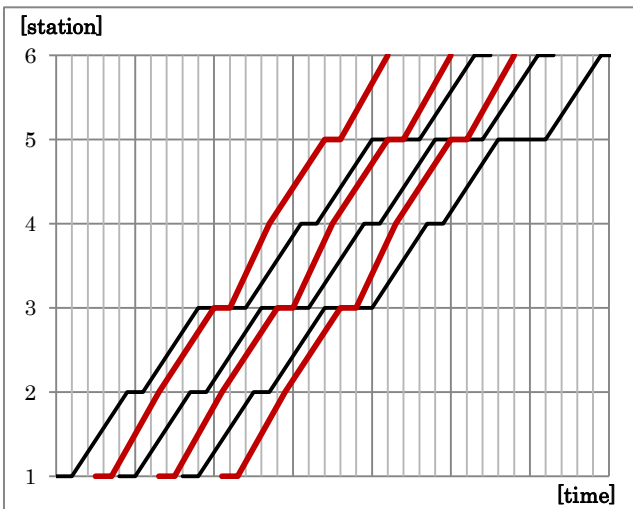


図 2.1. 緩急結合型ダイヤ

Fig2.1. A rapid-local pattern diagram

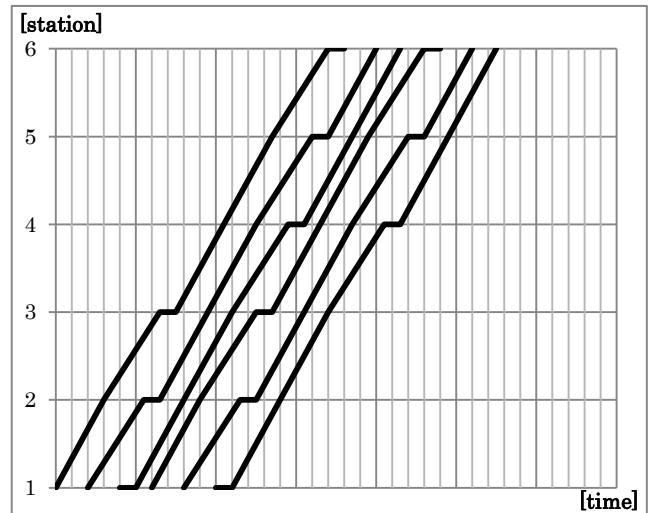


図 2.2. 選択停車型ダイヤ

Fig2.2. A select stop pattern diagram

3. 最適化問題の設定

〈3・1〉 混合整数計画法

混合整数計画法は線形計画問題の変数の一部または全部に整数条件が付加された問題である。混合整数計画問題の標準形を式(3.1)~(3.3)に示す。ここで、 Z^n は n 次元整数ベクトルの集合、 R^l は l 次元実数ベクトルの集合を示し、 x と y はそれぞれ整数変数、連続変数を表す。 c は n 次元ベクトル、 h は l 次元ベクトル、 b は m 次元ベクトルであり、 A 、 G はそれぞれ $m \times n$ 行列、 $m \times l$ 行列を表す。混合整数計画問題とは、上記のような制約条件の中で目的関数 z を最大化または最小化する実行可能解 x, y を最適解として求める問題である。

この混合整数計画問題の特徴として与えた枠組みの中で最適解が求められることがある。今回の研究では、変数の全てに整数条件を付加した整数計画問題を考える。

混合整数計画問題を実際に解く部分では、商用の数理計画ソルバ (CPLEX12.2) を用いて求解を行う⁽⁸⁾。

$$\text{目的関数} \quad z = c^T x + h^T y \quad (3.1)$$

$$\text{制約条件} \quad Ax + Gy \leq b \quad (3.2)$$

$$x \in Z^n, y \in R^l, x \geq 0, y \geq 0 \quad (3.3)$$

〈3・2〉 問題の前提

ここでは、改札などの情報を元に OD データ (旅客駅間移動データ) が得られたと仮定して、運転整理時の選択停車列車の運用を考える。まず、基礎的な検討として選択停車が再現できているかを考えるものとする。

〈3・3〉 最適化の指標

最適化の指標として重要なのが「乗客の不効用」である。乗客の不効用には

- ・所要時間

- ・混雑度
- ・乗換回数

などが考えられる。

この中でも旅客の損失の中で最も顕著である「所要時間」を評価の対象とする。その目的関数を式(3.4)に示す。この式(3.4)を最小化することを目的とする。

$$\text{minimize} \quad \sum_{o,d \in S} \sum_{k \in K} \sum_{j \in T} p_k^{o,d} t_{k,j}^{o,d} \quad (3.4)$$

$p_k^{o,d}$: 時刻 k に現れ、 o 駅から d 駅に移動する乗客の人数(人)
 $e_k^{o,d}$: 時刻 k に現れ、列車 j に乗り、 o 駅から d 駅に移動する乗客の旅行時間(分)

〈3.4〉 列車運行上の制約

列車を運行する上で、線路設備の都合満たさなければならぬ制約は以下のものとする。

- ① 各駅間の走行時間は定められた基準運転時分以上でなくてはならない
- ② 各駅での停車時間は定められた最小停車時分以上でなくてはならない
- ③ 各駅での進入・進出の際、先行列車との間隔は最小進入・進出時隔以上でなくてはならない
- ④ 列車同士の追い抜きはできない

〈3.5〉 旅客の行動仮定

旅客の経路選択に関する近似および仮定は以下の通りである。

- a) 旅客はある一定の時間ごとにまとめて駅に出現する
- b) 各旅客群は一つの経路しか選択できない
- c) 各旅客群はそれぞれの出現時刻以降に出発する列車にしか乗車できない
- d) 各旅客群の旅行時間は選択した経路の所要時間に依存する

〈3.6〉 選択停車列車を考えた定式化

前節の制約を実際に定式化した。今回は今までの優等列車とは違い停車駅を前もって決めていないため、多少変わっている部分はあるが、具体的な定式化の指針については参考文献(4)を参照されたい。

今回は、自由に列車を停車させるにあたり、新たに以下の式(3.5)で表される二値変数を設定した。この変数は列車 j が駅 s を通過する場合は1、停車する場合は0となる変数である。この変数を使って、最小停車時分、最小運転時分は以下の式(3.6)、(3.7)のように表される。

また、乗客は停車した駅でないと乗り降りできない。それを表す式が式(3.8)、式(3.9)である。

これらの式を新たに追加することにより、選択停車を実現させた。

$$pass^{s,j} \quad (3.5)$$

- ・最小停車時分

$$dep^{s,j} - arr^{s,j} \geq LD(1 - pass^{s,j}) \quad (\forall s \in S, \forall j \in T) \quad (3.6)$$

- ・最小運転時分

$$arr^{s+1,j} - dep^{s,j} \geq LR^s(1 - pass^{s+1,j}) \quad (\forall s \in S, \forall j \in T) \quad (3.7)$$

- ・旅客流動に関する式

$$\sum_{d \in S} \sum_{k \in K} z_{k,j}^{o,d} \leq M * (1 - pass^{o,j}) \quad (3.8)$$

$$\sum_{o \in S} \sum_{k \in K} z_{k,j}^{o,d} \leq M * (1 - pass^{o,j}) \quad (3.9)$$

4. ケーススタディ

〈4.1〉 路線モデル

駅数を5駅と考え、列車数は3本を考える。列車数3本のうち最初と最後の1本は全駅に止まり、選択停車を行うことで行けない駅のパターンを出さないように簡略化している。1本目と3本目の最初の駅を出発する時刻は固定する。真ん中の1本の列車に関しては自由に最初と最後の駅は必ず停車し、その他の3駅については自由に停まれるように設定する。まとめると表4.1のようになる。

表 4.1. ケース設定

Table 4.1. Case setup

The number of station	5 stations
The number of train	3 trains
Local	First, Third
Select stop	Second
Departure time at station 1	First and third : static Second : dynamic

〈4.2〉 想定する OD

選択列車が実装できたかを確認するため、表 4.2、4.3 に示す2パターンの極端な OD にて数値計算を行った。OD とは駅間の旅客移動情報のことである。

それぞれのパターンは、

- ・パターン 1 : 1 駅~3 駅~5 駅間を移動する旅客のみ
- ・パターン 2 : 1 駅~2 駅~4 駅~5 駅を移動する旅客のみである。今回 3 本の列車のうち 2 本目の列車は旅客の総旅行時間が最小になるように停車駅を決定する。つまり、明らかにパターン 1 の場合は 2 駅、4 駅は通過し、パターン 2 の場合は 3 駅を通過するはずである。

〈4.3〉 最適化計算

4.2.節の OD 表に基づいて数値計算をした結果がそれぞれ図 4.1、4.2 である。4.2.節の想定通り、パターン 1 では 2、

4 駅を通過し、パターン 2 では 3 駅を通過する結果が得られた。ここで、パターン 1 の場合 3 駅を通過することも考えられる。そこで、3 駅を通過するように式(4.1)の制約を入れたパターン 3 も用意した。その結果を図 4.3.に、目的関数の比較を表 4.4.に載せる。表 4.4.より適切に最適解を得られていることが確認できた。

表 4.2. パターン 1

Table 4.2. Pattern 1

[person]		Destination				
		1	2	3	4	5
Origin	1	0	0	1	0	1
	2	0	0	0	0	0
	3	0	0	0	0	1
	4	0	0	0	0	0
	5	0	0	0	0	0

表 4.3. パターン 2

Table 4.3. Pattern 2

[person]		Destination				
		1	2	3	4	5
Origin	1	0	1	0	1	0
	2	0	0	0	1	1
	3	0	0	0	0	0
	4	0	0	0	0	1
	5	0	0	0	0	0

表 4.4. 目的関数値比較

Table 4.4. The comparison of objective function

No. of pattern	Objective function [people·min]
Pattern 1	1321
Pattern 3	1608

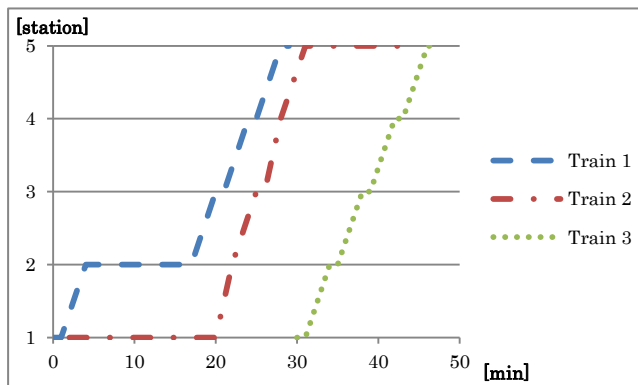


図 4.1. パターン 1 結果

Fig 4.1. The result of Pattern 1

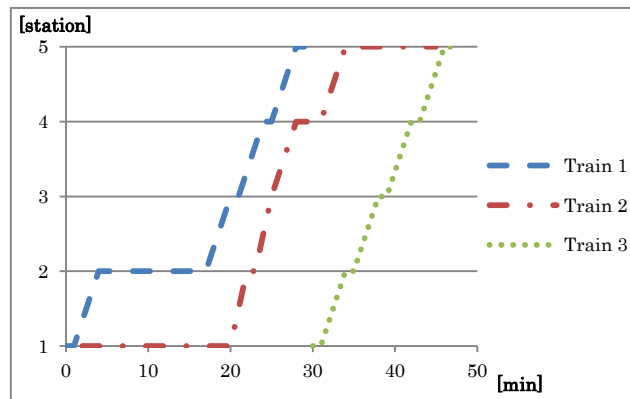


図 4.2. パターン 2 結果

Fig 4.2. The result of Pattern 2

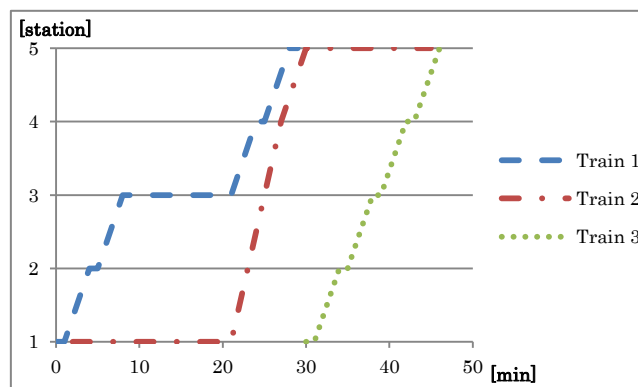


図 4.3. パターン 3 結果

Fig 4.3. The result of Pattern 3

5. 他の停車パターンとの比較

〈5・1〉 3つの停車パターン

選択停車型の優位性を示すために、以下の 3 パターンにて数値検証を行う。

パターン a: 平行ダイヤ

パターン b: 選択停車型ダイヤ

平行ダイヤとは、全て各駅停車列車によるダイヤのことを言う。今回緩急結合ダイヤは乗換あり、なしの制約が選択停車と大きく違うため考慮から外すこととする。

〈5・2〉 路線モデルと想定する OD

前提条件を表 5.1.に、想定する OD パターンを表 5.2.に示す。平行ダイヤは全て各駅停車、出発時刻は表 5.1.に倣うものとする。OD については 1 駅~3 駅~5 駅の移動客が多く、その他の駅間移動は少ないような OD を想定した。

〈5・3〉 最適化計算

数値計算後のダイヤを平行ダイヤ、選択停車型の順に図 5.1.、5.2 に示す。また目的関数の値を表 5.3.に示す。表 5.3.を見ると目的関数の値が小さくなっていることが分かる。しかし、この結果は 1 時間で打ち切った結果を載せている。

そのため最適の保証がない値となってしまう。その理由としては、選択停車を実現するにあたって計算領域が大きくなりすぎてしまっていること、変数が多くなりすぎてしまい、現実時間で計算が終わらなくなってしまうためである。

表 5.1. ケース設定
Table 5.1. Case setup

The number of station	5 stations
The number of train	5 trains
Local	First, Fifth
Select stop	Second, Third and Fourth
Departure time at station 1	First and fifth : static The others : dynamic

表 5.2. OD
Table 5.2. OD

[person]		Destination				
		1	2	3	4	5
Origin	1	0	10	100	10	100
	2	0	0	10	10	10
	3	0	0	0	10	100
	4	0	0	0	0	10
	5	0	0	0	0	0

表 5.3. 目的関数値比較

Table 5.3. The comparison of objective function

A rapid-local pattern	63,600 [person・min]
Select stop pattern	62,080 [person・min]

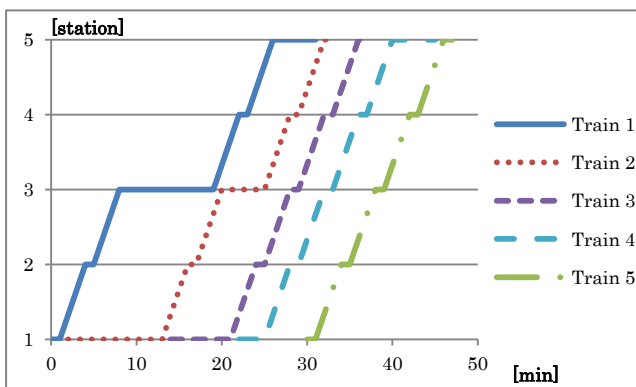


図 5.1. 平行ダイヤ型の結果

Fig 5.1. The result of rapid-local pattern

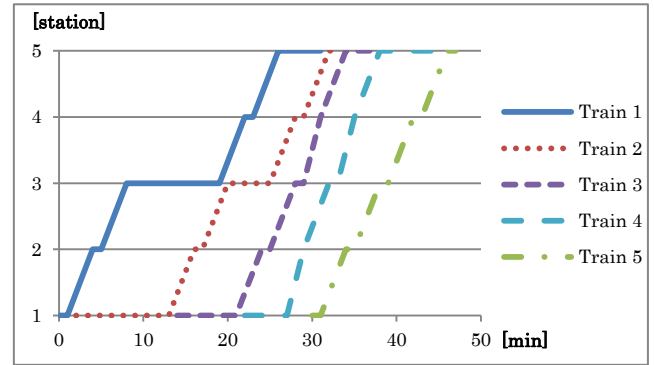


図 5.2. 選択停車型の結果

Fig 5.2. The result of select stop pattern

6. まとめ

今回、選択列車を考慮した運転整理の最適化の実現に向けた最初の検討を行った。定式化を行い列車本数が少ないケースにおいては想定通りの結果が得られ、定式化は正しく行われたものとする。しかし、列車本数が増えると途端に計算が現実時間では終わらなくなることが見られた。このシミュレーションを回すにあたって使用した計算機環境は「6コア、CPU3.20GHz、メモリ 16GB」である。

この問題を解くにあたり以下の二つのアプローチが考えられる。

- ①スーパーコンピュータなどさらなる高性能な計算機の使用
 - ②計算・変数領域を小さくする
- ①の計算機環境で数値計算を行いつつ、最終的には②の計算領域を小さくする方向でより複雑な路線の問題にも適用できるようにしていくことを今後の課題としたい。

文 献

- (1) 富井規雄：「列車ダイヤのひみつー一定時運行のしくみ」, 成山堂書店 (2005)
- (2) (財)鉄道総合技術研究所 運転システム研究室：「鉄道のスケジューリングアルゴリズム」, NTS (2005)
- (3) K.Chigusa,K.Sato,T.Koseki: "A Passenger-Oriented Optimization of Train Rescheduling Based on Mixed Integer Programming", IEEJ Transactions on Industry Applications Vol.132 No.2, pp170-177 (2012)
千種健二, 佐藤圭介, 古関隆章：「混合整数計画法に基づく列車運行乱れ時の旅客損失に主眼を置いた運転整理最適化」, 電気学会論文誌.D, 産業応用部門誌, pp170-177 (2012)
- (4) 森拓哉, 古関隆章：「混合整数計画法を用いた方向別複数線の運転整理支援」, 交通・電気鉄道研究会, TER-13-008 (2013)
- (5) 「鉄道ダイヤ回復の技術」, 電気学会・鉄道における運行計画・運行管理業務高度化に関する調査専門委員会 (2010)
- (6) 渡邊翔一郎, 森拓哉, 古関隆章：「省エネルギー運行管理のための運転法の比較分析と列車群電力可視化」, 交通・電気鉄道研究会, TER-13-017 (2013)
- (7) 国土交通省 鉄道局ホームページ：「鉄軌道輸送の安全に関わる情報 (平成 23 年度)」
<http://www.mlit.go.jp/common/000229069.pdf>
- (8) IBM ILOG CPLEX