

乗務員基地配置駅決定アルゴリズムの開発

加藤 怜* 佐藤 圭介* 福村 直登**

Development of an Algorithm for Allocation of a Railway Crew Depot

Satoshi KATO Keisuke SATO Naoto FUKUMURA

A railway crew depot is normally located adjacent to a station along the track. The duty of individual crew member starts from the crew depot and returns to the same, therefore, the efficiency of the crew schedule is dependent on the crew depot allocation. To analyze the best allocation, we developed an algorithm that decision of crew depot allocation and preparation of crew schedule are performed simultaneously. Applying the foregoing algorithm to the existing freight line, we are able to obtain the valid allocation and schedule plan within 24-hours.

キーワード：乗務員基地，乗務員運用計画，数理計画，施設配置・運搬経路問題，列生成法

1. はじめに

鉄道輸送において、ダイヤ改正のたびに策定される輸送計画の1つに乗務員の運用計画がある¹⁾。乗務員の出勤から退勤までの1回の勤務のことを行路と呼び、全列車に対し乗務員を割り当てた計画全体のことを乗務員運用計画と呼ぶが、鉄道事業者では無駄のない、効率のよい乗務員運用計画の作成が求められる。

乗務員基地は営業路線上のいずれかの駅に隣接して配置され、各乗務員はいずれかの基地に所属する。乗務員は、所属する乗務員基地が配置されている駅から勤務を開始し、同一の駅にて勤務を終了する必要があるが、乗務距離や乗務時間には制限があるため、乗務員基地の配置箇所により乗務員運用計画の効率は大きく変化する。よって、鉄道事業者にとって乗務員基地の配置箇所の決定は重要な問題である。本論文では、対象として貨物列車の乗務員基地配置に着目するが、貨物列車は全国にまたがる長距離の運行や夜間の運行が多くあるため、乗務員運用の効率性が乗務員基地の配置箇所に影響されやすい。

これまで、乗務員基地の配置箇所については、運用計画作成担当者により手作業で特定基地を対象とした改廃の検討は行われてきた。しかし、必要となる輸送量、すなわち列車ダイヤを最重要視した場合における基地配置箇所の全面的な見直しの検討は、乗務員運用を全て白紙状態から作成し直す必要があり、作業量の問題から困難であるため、コンピュータの活用による基地配置箇所の検討が望まれている。

乗務員基地の配置箇所の妥当性を検証するためには、

その基地配置にもとづく乗務員運用計画の妥当性についても評価する必要がある。そこで、乗務員運用計画の自動作成にも一般的に用いられる数理計画法を適用し、施設配置・運搬経路問題と呼ばれるモデルを用いることで、基地配置駅の決定とその配置にもとづく乗務員運用計画の概略案の作成を同時に行うアルゴリズムを開発した。

2章では乗務員基地と乗務員運用計画の関係について説明する。3章では問題のモデル化について、4章ではアルゴリズムの具体的な流れについて説明する。5章では実際の路線データを用いての数値実験を行い、最後に6章でまとめを述べる。

2. 問題の概要

2.1 乗務員基地と乗務員運用計画

問題の前提として、列車ダイヤが所与であるとする。全列車に対し、1人の乗務員を割り当てるのが乗務員運用計画作成の基本であるが、必ずしも1列車の全区間を1人の乗務員が運転するわけではなく、途中駅で乗務員の交代が起きることがある。よって、乗務員の勤務を考えるためには、1列車を最小単位とするのではなく、乗務員交代が可能な駅で分割したものを考えなければならない。以下、この最小単位のことを乗務と呼ぶ。すると、行路とはいくつかの乗務の集まりと定義できる。なお、乗務員は必要に応じて移動のために列車に乗車することがあり、これを便乗と呼ぶ。

各乗務員は、いずれかの乗務員基地に所属する。乗務員基地は営業路線上のいずれかの駅に隣接して配置されるが、各乗務員は所属した基地の隣接駅から勤務を開始し、同じ駅にて勤務を終了する。よって、列車ダイヤと基地配置の制約を受けながら、乗務員運用計画を作成し

* 輸送情報技術研究部 (運転システム)

** 輸送情報技術研究部

特集：輸送情報技術

なければならない。なお、乗務員の勤務は就業規則や交番作成規程に定められている各種条件を守る必要があり、行路の作成にあたってはこれらも制約条件として考慮する必要がある。

図1に列車ダイヤの例(乗務単位で示したもの)を、図2にこの列車ダイヤに対応する乗務員運用計画の例を示す。図2の例では、A駅・C駅に隣接して乗務員基地が配置されており、A駅に配置されている基地の2つの行路を、C駅に配置されている基地の3つの行路を示している。○は行路の開始を、△は行路の終了を意味しており、たとえばA1行路はA駅から乗務1を担当することから始まり、乗務13を担当することでA駅に戻って終わる行路である。これら5つの行路により、運行するべき列車の全乗務区間に乗務員を割り当てることができる。なお、C3行路は暦日をまたぐ行路であるため、この計画では1日あたり、A1、A2、C1、C2、C3行路を担当する乗務員と、前日に引き続いてC3行路を担当する乗務員の計6人が必要となる。また、乗務6はA2行路とC1行路の2つの行路が担当しているが、実際に運転を担当するのはA2行路であり、破線で示したC1行路は便乗することを意味する。

この例ではA駅・C駅が乗務員基地であることが前提となっているが、この基地配置が図1の列車ダイヤを満足することを考えたときに、望ましい配置であるかが問題となる。たとえば、A駅の基地をB駅に移転することや、A駅の基地をC駅の基地に統合することが考えられるが、この例だとC駅に統合することで、図3に示すような乗務員運用を組むことができる。乗務員基地が多いと管理費用が増加するため、一般的には基地は少ない方

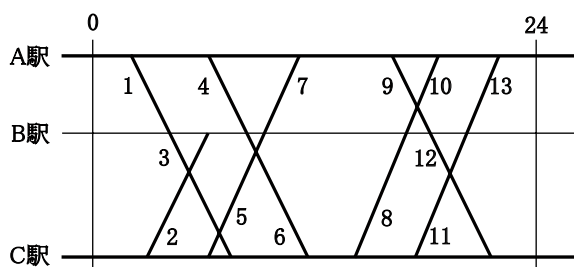


図1 列車ダイヤの例

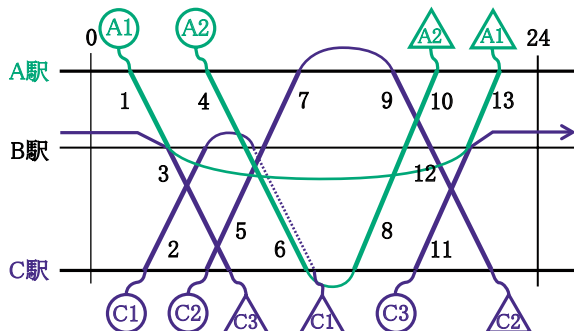


図2 乗務員運用計画の例

が望ましい。よって、図2の2基地必要となる乗務員運用計画と比較し、図3の1基地でよい計画はより好ましいと考えられる。しかし、1基地に統合したことで、暦日をまたぐ行路が増加している(1→2)。また、行路数は減少している(5→4)ものの、その分1行路あたりの拘束時間・労働時間が増加している。

乗務員基地の配置箇所妥当性を検証するためには、その基地配置にもとづく乗務員運用計画の妥当性についても評価する必要がある。よって、本研究では乗務員基地の最適な配置駅の決定のみならず、その配置にもとづく乗務員運用計画の作成まで実現することを考える。以下、この問題を基地配置問題と定義する。

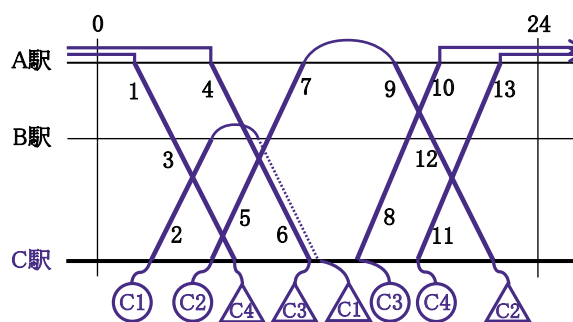


図3 C駅に統合したときの乗務員運用計画の例

2.2 基地配置の制約条件と評価指標

乗務員基地配置の制約条件として、乗務員運用計画作成のための種々の制約の中から、基地配置を左右する主要な要因として、以下に示すような行路に関する制約条件を抽出した。

- ① 所属する乗務員基地の配置駅から始まり、同一の駅で終了すること
- ② 労働時間の上限を守ること
- ③ 拘束時間の上限を守ること
- ④ 1継続乗務キロの上限を守ること
- ⑤ 1継続乗務時間の上限を守ること
- ⑥ 行先地の時間制約を守ること

一方、基地配置の評価指標に関しては、以下のような項目が考えられる。

- ① 配置基地数が少ないこと
- ② 新設・廃止する基地数が少ないこと
- ③ 必要行路数が少ないこと
- ④ 総拘束時間が少ないこと
- ⑤ 総便乗時間が少ないこと

各評価指標について、どの項目をどの程度重視するかは状況により異なると思われる、またこれらの決定には経営判断的要素も含まれるので、一意に決定してしまうことは必ずしもよくないものと思われる。そこで、各項目に重みを与え、その総和を評価関数とし、この評価値を最小にすることを考える。これにより、様々な状況に対

応できるのはもちろんのこと、重みを変えて実行することで複数の基地配置案を得ることができるので、それぞれ比較・評価することが可能である。

2.3 関連する従来研究

乗務員基地の配置箇所を検討するという研究は、日本国内はもちろんのこと、海外においてもほとんど見られない。一方、乗務員運用計画の自動作成に関する研究は欧米での航空機を対象としたものをはじめ、数多く行われている²⁾・³⁾。近年になり、国内で鉄道を対象とした研究も見られるようになった⁴⁾・⁵⁾。これらの研究に共通することは、いずれも数値計画法を適用していることである。

しかし、これらの研究は乗務員基地が所与であることが前提であり、そのもとで各基地の乗務員運用計画を自動作成するものである。よって、基地配置箇所まで検討することはできず、これらの研究成果をそのまま活用することはできない。

3. 問題のモデル化

3.1 基地配置問題へのアプローチ

2.1節で述べたように、基地配置問題は基地配置駅の決定と乗務員運用計画の作成の2つの問題を含む。よって、アプローチとして以下の2つが考えられる。

- (1) [2段階法]: 列車ダイヤをもとに、乗務員基地の配置駅を決定したのち、配置駅にもとづく乗務員運用計画を作成する。
- (2) [同時解法]: 乗務員基地の配置駅を決定する過程で、同時に乗務員運用計画を作成する。

2段階法は、まず乗務員基地の配置駅を決めてからその配置駅にもとづく乗務員運用計画を作成する、ごく自然な方法である。現在行われている手作業による検討も、このアプローチにもとづいている。コンピュータの活用を考える上では、乗務員基地の配置駅を決定してしまえば、乗務員運用計画の作成は既存の研究をそのまま活用できる利点がある。しかし、そもそも列車ダイヤから乗務員基地の配置箇所を決めることは容易ではなく、場合によっては決定した乗務員基地の配置駅からは乗務員運用計画が作成できない可能性がある。同時解法の場合、同時に扱う範囲が広く、規模が増大するという問題はあるものの、2段階法のような問題は起こりえない。よって、ここでは同時解法のアプローチをとる。

3.2 問題の数値モデル化

基地配置駅の決定と乗務員運用計画作成を同時に実現するモデルとして、本研究では施設配置・運搬経路問題 (Location-Routing Problem, LRP) と呼ばれる数値計画モ

デル⁶⁾に着目した。LRPとは、二次元平面上に施設配置の候補となりうる箇所と、ある需要をもった顧客が複数点在しており、候補の中から実際に施設を配置する箇所を決めて、同時に全顧客を訪問しかつ需要を満たせるような各配置施設からの運搬車の経路を決める問題である。このLRPにおいて、施設を乗務員基地、顧客を乗務、運搬車を乗務員、運搬経路を行路と見立てることで、基地配置問題をLRPの一種と考えることができる。

LRPの研究は、近年になり盛んに行われるようになった。数多くの研究が行われている中で、本研究は文献7)を参考に定式化を行い、列生成法によるアルゴリズムを構築する。

3.3 LRPによる定式化

LRPにもとづき、基地配置問題を定式化する。まず記号の定義を行う。以下に所与の集合および定数を示す。

- I 乗務の集合
- K 基地配置候補の集合
- P^k 基地配置候補 $k \in K$ の行路案の集合
- L_i 乗務 $i \in I$ を受け持つことができる基地配置候補の集合
- c_p^k 基地配置候補 $k \in K$ の行路案 $p \in P^k$ のコスト
- f^k 基地配置候補 $k \in K$ の配置コスト
- n^k 基地配置候補 $k \in K$ の最大行路数

行路案のコスト、基地配置のコストには、2.2節で挙げた各評価指標の重みを反映させた値を設定する。たとえば、基地配置数を最小にし、行路数も少なくする区所配置案を得たい場合には、 $c_p^k = 1$ 、 $f^k = M$ (M は十分大きな値) とすればよい。続いて、変数を示す。

- x_p^k 基地配置候補 $k \in K$ の行路案 $p \in P^k$ を採択するとき1、さもなければ0
- y^k 基地配置候補 $k \in K$ に基地を配置するとき1、さもなければ0

以上の記号を用いて、基地配置問題を以下のように定式化する。

$$\text{minimize } \sum_{k \in K} \sum_{p \in P^k} c_p^k x_p^k + \sum_{k \in K} f^k y^k \quad (1)$$

$$\text{subject to } \sum_{k \in K} \sum_{p \in P^k} a_{ip}^k x_p^k \geq 1, \forall i \in I \quad (2)$$

$$n^k y^k - \sum_{p \in P^k} x_p^k \geq 0, \forall k \in K \quad (3)$$

$$\sum_{k \in L_i} y^k \geq 1, \forall i \in I \quad (4)$$

$$x_p^k \in \{0, 1\} \quad (5)$$

$$y^k \in \{0, 1\} \quad (6)$$

(1)式は目的関数であり、2.2節で述べた各評価指標に対し重みを与えたときの、その評価関数を示す。すなわ

特集：輸送情報技術

ち、この定式化で最適化することは、評価関数値が最小となる基地配置案を求めることを意味する。(2)～(6)式は制約条件であるが、(2)式は全乗務に対し1つ以上の行路を割り当てることを示す。実際には1人が担当するが、2以上となった場合はそのうちいずれか一方の行路が担当し、他の行路は便乗する。(3)式は採択された行路の起点・終点となった配置候補には必ず基地を配置しなければならないことを意味する。(4)式はコンピュータでの処理速度を向上させるために設けた追加制約式であるが、その意味は各乗務について、受け持つことができる基地配置候補はいくつかに絞られるので、そのいずれかに基地を配置する必要があることを示す。(5)(6)式により行路採択を表す変数と基地配置を表す変数を0か1に固定する。

上記の定式化では、(2)式を不等式制約としたが、等式制約にすることも考えられる。それぞれ、数理計画モデルで集合被覆モデル、集合分割モデルに対応するが、どちらのモデルで定式化を行うかは議論が必要である⁸⁾。乗務員運用計画の厳密性を重視するのであれば、あらかじめ便乗を含めた行路を作成し、集合分割モデルで定式化を行うことが望ましい⁹⁾。しかし、集合分割モデルは実行可能解の算出が困難であり、基地配置駅の決定と同時に最適化する基地配置問題では、深刻な問題となる可能性がある。よって、(2)式のように集合被覆モデルで定式化している。

4. 最適基地配置アルゴリズム

4.1 行路作成ネットワーク

3.3節の定式化で示した問題について、実際にコンピュータに入力し解を得るためには、あらかじめ基地配置候補 k ごとに行路案集合 P^k を用意する必要がある。そこで、行路案を作成するためのネットワークを構成する。ネットワークの構成例として、図1の列車ダイヤに対するA駅の基地の行路作成ネットワークを図4に示す。始点 s ・終点 t はA駅に対応し、それぞれ行路の開始・終了を意味する。それ以外のノードは各乗務に対応している。もし乗務 i から乗務 j へ乗継が可能(継続乗務を含む)であるならば、ノード i からノード j へアークを張る。

図4のネットワークにおいて、始点から終点までの経路で、かつ2.2節で述べたような行路に関する制約条件を満たすならば、それは1つの行路案となり、その経路上のノードに対応する乗務を担当することになる。このネットワークは基地配置候補ごとに構成され、それぞれで行路を作成する。

4.2 列生成法の適用

実路線のダイヤデータを用いて行路を作成しようとする、制約条件を満たす行路案は膨大で、その数は数百

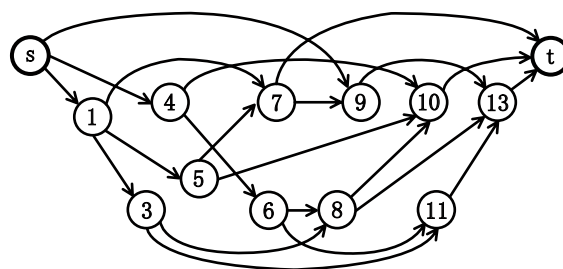


図4 行路作成ネットワーク (A駅の基地の例)

万～数千万にも及び、その中から最適解を得ることは非常に困難である。そこで、コンピュータで処理可能な数まで抑制する必要があるが、本研究では列生成法¹⁰⁾を適用することで、有効な行路のみを抽出する。列生成法は、乗務員運用計画の自動作成ではよく用いられる手法であり、近年ではダイヤ乱れ時の乗務員運用整理問題に対しても適用例が報告されている¹¹⁾。

列生成法適用の過程を示す。全行路案集合のうち一部の行路案のみを用意し、定式化の(5)、(6)式の0-1条件を0以上に緩和した問題を解く。すると、その最適解の情報として、(2)、(3)式の各制約式に対応して双対価格 $\pi_i (i \in I)$ 、 $\mu^k (k \in K)$ を算出することができる。これらの情報から、基地配置候補 k の行路案 p について、以下の \hat{c}_p^k を計算でき、新たに追加すべきか判定できる。

$$\hat{c}_p^k = c_p^k - \sum_{i \in I} a_{ip}^k \pi_i + \mu^k \quad (7)$$

この値は被約費用と呼ばれるが、もし $\hat{c}_p^k < 0$ であれば、その行路は効率的と判定され、新たに行路案集合に追加し、再度変数の条件を緩和した問題を解く。どの行路についても $\hat{c}_p^k < 0$ とならない場合、これ以上行路を追加しても意味がないと判定され、列生成を終了する。

被約費用が負である行路であれば改善は可能だが、その値が低ければ、より改善が期待できる。そこで、被約費用が最小となる行路を求めることを考える。この問題は、図4のネットワーク上での資源制約付き最短路問題に帰着することができ、ラベリングアルゴリズム¹²⁾を適用することで、効率的に最短路を求めることができる。

4.3 アルゴリズムの流れ

最適基地配置アルゴリズムの全体の流れを図5に示す。列車ダイヤの情報を取得したのち(処理[1])、図4のような行路作成ネットワークを各基地配置候補について構築する(処理[2])。続いてこのネットワーク上で、各基地配置候補についていくつかの行路を作成し、初期行路集合を構成する(処理[3])。このとき、各基地候補駅が受け持つことができる乗務を列挙することで、定式化の(4)式に相当する追加制約式を生成する(処理[4])。次に前節で述べたように、本来は0か1である変数を0以上の実数に緩和し、目的関数値が最小となる解を求める

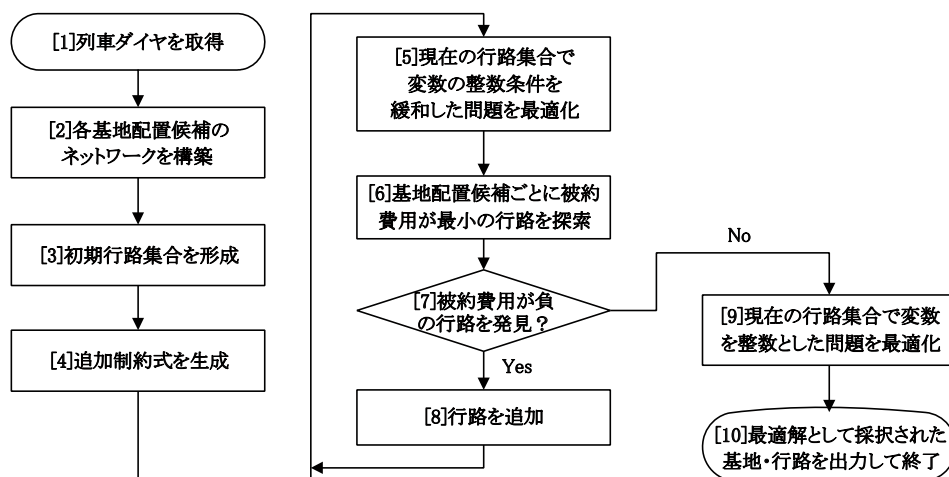


図5 最適基地配置アルゴリズムの流れ

(処理[5])。この解の情報(双対価格)を用いて、基地配置候補ごとに被約費用が最小となる行路をネットワーク上で探索する(処理[6])。その結果被約費用が負となる行路が1つでも見つかったのであれば(処理[7])、行路集合に追加する(処理[8])。そして新たな行路集合のもと、再度目的関数値が最小となる解を求める(処理[5])。この流れを繰り返し、被約費用が負である行路が1つも見つからなかったならば、各変数に0か1であるという制約を付加し、最終的な解を求める(処理[9])。この最適解において採択された乗務員基地および行路が、本アルゴリズムにより算出される基地配置案である(処理[10])。なお、数理計画問題の求解(処理[5]および処理[9])には数理計画ソフトウェアであるIBM ILOG CPLEX¹³⁾を使用する。

5. 数値実験

5.1 アルゴリズムの性能評価

実際の貨物列車の路線データ、ダイヤデータを用いて、アルゴリズムの評価試験を行った。路線図および基地配置候補駅を図6に示す。A駅~K駅の計11駅が基地配置候補駅であり、総路線長は約700km、1日の列車本数は約300本、乗務数は約1000と、貨物列車でも広範囲を対象とする。試験にはCPU Core i7 3.2GHz、メモリ3.0GBのパソコンを用いた。

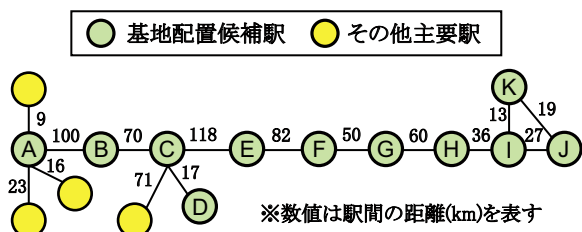


図6 路線図と基地配置候補駅

開発したアルゴリズムを適用したところ、計算時間約24時間で良質の解が算出できることを確認した。また、算出した基地配置駅(A・C・F・J駅)、乗務員運用計画についても、現状との比較や運用計画作成担当者へのヒアリング調査を通してその妥当性を確認した。

5.2 アルゴリズムの活用例

開発したアルゴリズムを用いて計算を繰り返すことで、各種設定を変更したときの、基地配置駅と乗務員運用計画の変化を検証することができる。以下では、1継続乗務キロの上限を変えた計算例を示す。

基地配置駅を検討する際には、2.2節で挙げた行路の制約条件を考慮することになるが、その中でも1継続乗務キロの上限制約は最も重要となる。なぜならば、1継続乗務キロの上限により、基地配置駅の間隔を伸ばせる可能性があり、結果として基地配置駅が変わる可能性があるからである。

表1は、1継続乗務キロの上限を200, 220, 240, 260kmと変化させたときの基地配置駅、行路数(暦日換算)、労働時間(1暦日あたりの平均値)を示している。なお、評価指標は基地配置数の最小化に最も重みをおき、行路数を少なくすることも考慮している。表1を見ると、1継続乗務キロの上限を伸ばすことで、各基地の乗務員が担当できる範囲が広まるため、基地配置駅数が減少し、また基地配置駅数が同じならば行路数が減少することがわ

表1 1継続乗務キロの上限の変更

1継続乗務キロ	基地配置駅	行路数	労働時間
200	A・C・F・J	294	6:29
220	A・C・F・J	290	6:30
240	A・C・H	293	6:38
260	A・C・I	288	6:38

特集：輸送情報技術

表2 基地配置数を変化させたときの变化

基地配置数	基地配置駅	行路数
4	A・C・F・J	290
5	A・C・E・F・J	283
6	A・C・E・F・J・K	278
7	A・C・D・E・G・J・K	277

かる。また、そのときの労働時間の変化がどの程度あるのかについても、把握することができる。

この他にも、開発したアルゴリズムでは様々な条件設定が可能である。たとえば、基地配置数に上限を設ける、あるいは基地配置数を指定することも可能である。また、実務上廃止が考えられない駅がある場合、その駅には必ず基地を配置するといった設定も可能である。これらは組合せることも可能なので、たとえば既存の基地配置駅の一部を残すことを指定しつつ、全体の基地配置数を減らすことが可能か検討を行うといったことも可能である。

計算例として基地配置数を指定した例を取り上げる。図6の路線データにおいて、1継続乗務キロの上限を220kmに固定し、基地配置駅数を4～7駅まで増加させたときの基地配置駅および行路数(暦日換算)の変化を表2に示す。基地配置駅数を増やすと行路数が減少しているが、これは各乗務について受け持つことができる基地が増加するため、必要な便乗が減ったことが要因として考えられる。しかし、行路数の減少幅は基地配置数の増加とともに小さくなるが見取れる。また、基地配置駅を見ると、基地数が少ないときに配置された駅には基地数を増やしてもほとんどの場合配置されることもわかる。このように、基地数の増減による乗務員運用計画の変化や基地配置駅との関係を、本アルゴリズムにより検証することができる。

6. まとめ

本論文では乗務員基地の配置箇所について、現在の列車ダイヤにもとづき再検討を行うために、コンピュータを用いて最適な基地配置駅を提案するアルゴリズムを開発した。モデル化にあたり、数理計画モデルの1つである施設配置・運搬経路問題に着目することで、最適な基地配置駅の提案と、その基地配置にもとづく乗務員運用計画の概略案の作成を同時に実現した。これにより、基地配置案の妥当性を乗務員運用計画を見ながら検証することができる。

数値実験では、実際の貨物列車の路線データを用いたが、広範囲を対象としても24時間以内で妥当な基地配置案を算出できることを確認した。また、基地配置駅数の指定など様々な設定が可能であるため、本アルゴリズムにより基地配置に関する多様なシミュレーションを行うことが可能になった。なお、今回は貨物列車の実路線

データを用いたが、アルゴリズムの基本的な考え方は旅客列車にも応用可能であると考えている。

今後は本研究成果を活用するために、乗務員運用の効率性についてさらに研究を進めていく所存である。

謝辞

本研究開発の実施にあたりご協力いただいた、日本貨物鉄道株式会社の重田英貴、新居優の両氏に深く感謝いたします。

文献

- 1) (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, エヌ・ティー・エス, 2005
- 2) Wedelin, D., "An algorithm for large scale 0-1 integer programming with application to airline crew scheduling," *Annals of Operations Research*, Vol.57, pp.283-301, 1995.
- 3) Caprara, A., Fischetti, M., and Toth, P., "A heuristic method for the set covering problem," *Operations Research*, Vol.47, No.5, pp.730-743, 1999.
- 4) 三浦礼, 今泉淳, 福村直登, 森戸晋: 鉄道における乗務員運用計画の集合被覆問題に対する Wedelin の解法の適用, 電気学会論文誌C, Vol.129, No.10, pp.1958-1967, 2009
- 5) 室井裕喜, 西竜志, 乾口雅弘: 鉄道乗務員スケジューリング問題に対する列生成法の収束性改善, 電気学会論文誌C, Vol.130, No.2, pp.275-283, 2010
- 6) Nagy, G. and Salhi, S., "Location-routing: Issues, models and methods," *European Journal of Operational Research*, Vol.177, pp.649-672, 2007.
- 7) Berger, R. T., Coullard, C. R., and Daskin, M. S., "Location-routing problems with distance constraints," *Transportation Science*, Vol.41, No.1, pp.29-43, 2007.
- 8) 今泉淳: 鉄道の運用計画問題に対する整数計画法によるアプローチ, オペレーションズ・リサーチ, Vol.53, No.8, pp.439-445, 2008
- 9) 坂口隆: 乗務員運用計画の集合分割モデルに対する有効行路案抽出法, 鉄道総研報告, Vol.20, No.2, pp.17-22, 2006
- 10) Wolsey, L. A., *Integer Programming*, John Wiley and Sons, New York, 1998.
- 11) 佐藤圭介, 福村直登: 貨物列車の乗務員運用整理案作成アルゴリズムの開発, 鉄道総研報告, Vol.23, No.8, pp.23-28, 2009
- 12) 植田達広, 森戸晋, 今泉淳, 福村直登: 乗務員運用計画問題の列生成子問題に対するPull型ラベリング解法と性能評価, 日本OR学会2005年秋季研究発表会アブストラクト集, pp.160-161, 2005
- 13) IBM ILOG CPLEX, <http://www-01.ibm.com/software/integration/optimization/cplex/>