

ダイヤ乱れ時の貨物機関車運用整理問題

佐藤 圭介[†] 福村 直登[†]

鉄道事業者は事故などにより列車ダイヤが乱れると、収束を図るためのダイヤ変更と、それに伴う車両・乗務員のスケジュール変更を行っている。本研究は貨物列車をけん引する機関車を対象に、変更ダイヤを所与として、機関車の検査周期や検査基地の作業容量などを考慮しつつスケジュール変更を行う運用整理問題を議論する。本研究はこの問題を集合分割制約を持つ整数計画問題として定式化し、集合被覆緩和と列生成法を組み合わせで解を求める。実際の列車ダイヤ・機関車運用計画に乱れ事例を適用した計算機実験の結果、満足できる質の解が許容時間内に得られたので紹介する。

Freight Train Locomotive Rescheduling Problem after Disruptions

KEISUKE SATO[†] and NAOTO FUKUMURA[†]

This study discusses a rescheduling problem of locomotive assignment to freight trains after the timetable adjustment is done in a disrupted situation. We model the problem as an integer programming one with set-partitioning constraints taking maintenance of each locomotive into account, and solve it by a combination of its set-covering relaxation and column generation. Numerical experiments using real data reveal that our method provides a locomotive rescheduling plan of satisfactory quality in acceptable time.

1. はじめに

事故などで列車ダイヤが乱れると、鉄道事業者は事態収束のため運転整理⁴⁾—運休・時刻変更などで列車ダイヤを、またそれに伴い車両運用計画、乗務員運用計画を一時変更する。車両・乗務員の個々の変更は運用変更と呼ばれるが、ここではその総体を運用整理と呼ぶ。本研究は小規模な運転整理³⁾や、乗務員運用整理²⁾での例を踏まえ、貨物列車をけん引する機関車の運用整理に最適化手法を応用する。実務制限上、旅客列車の研究⁵⁾とは異なり、変更ダイヤを所与とし、そのダイヤに対して稼働中の機関車を充当する。

所与の変更ダイヤの中には単機（機関車単体の回送）が設定可能な区間・時間帯と最大重連（同時連結）数も指定されている。また昼夜問わず長距離運行が多い貨物列車の特性により2, 3日の運用を日付を区切らず整理する。ここで課題なのが数十時間おきに行わなければならない機関車の仕業検査⁴⁾で、本研究では臨時の検査実施を認めつつも検査基地の容量を考慮し、機関車の検査周期を守るような運用整理を行う。

2. 問題設定

2.1 ネットワーク表現

まず列車ダイヤと機関車運用をネットワークで表現する。図1は計画されたダイヤ・機関車の運用と、それに乱れが生じ2レの出発が遅れる変更ダイヤである。“仕”は列車の到着と出発の間の仕業検査実施を意味する。運用整理開始時刻 h_0 と、その間是个々の機関車の運用変更を許す運用整理期間 h を定めたとき、本研究では図2のようなネットワークを構成する。

ネットワークの構成法は、まず機関車の集合を K とおき、仕業検査周期を R とする。各 $k \in K$ は前回の仕業検査時刻 PrevM.time_k と h_0 時点での所在駅などを持つ。次に運用整理期間に含まれる変更列車ダイヤを、レグと呼ばれる機関車の運用範囲の最小単位に分割し、レグ集合 I を定義する。また整理期間内で設定してもよい単機レグ集合 $\hat{I}$ も用意する。要素 $i \in I \cup \hat{I}$ は発車時刻と到着時刻 $\text{Arr.time}_i (< h_0 + h)$ 、発車駅と到着駅を持つ。各機関車には整理期間後に最初に充当される予定のレグがあるので、それを収束レグと呼び、その集合を D とおく ($|K| = |D|$)。各 $d \in D$ は出発駅と、整理期間後の計画に沿って運用される場合の次回仕業検査予定時刻 $\text{NextM.time}_d (\geq h_0 + h)$ を持つ。レグ（ないし h_0 で検査可能駅にいる機関車）

[†] 財団法人鉄道総合技術研究所
Railway Technical Research Institute

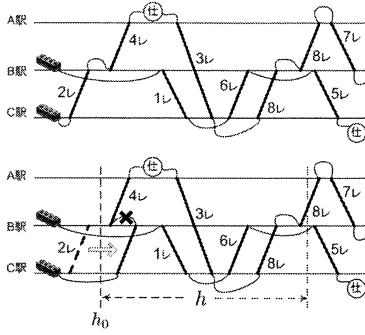


図1 ダイヤ・機関車運用計画と変更ダイヤ
Fig. 1 Timetable/locomotive scheduling plan and adjusted timetable.

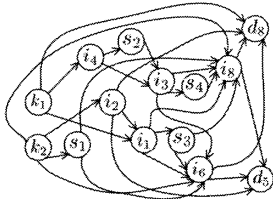


図2 ネットワーク表現
Fig. 2 Network Representation.

$v \in K \cup I \cup \hat{I}$ 後の検査の実施を s とすると, s は最も早く検査を終えられる時刻 FinishM.time_s の情報を持つ. 仕業検査実施の集合を S とおく. 以上のようにしてノード集合 $V := K \cup I \cup \hat{I} \cup D \cup S$ を構成する.

ノード対の駅と時刻関係が正しく成り立てば有向エッジをひき, その集合を E とおく. エッジのコスト関数は $c: K \times E \rightarrow \mathbb{R}_+$ とし, ノード間が元の運用計画のままのつながりであれば0を, さもなくば正の値とする. 最後に仕業検査に関連する関数 $H: K \cup S \rightarrow 2^{I \cup \hat{I} \cup D}$ を定義する. 各 $k \in K$ または $s \in S$ について,

$$\begin{aligned} H(k) &:= \{i \in I \cup \hat{I} \mid \text{PrevM.time}_k + R \geq \text{Arr.time}_i\} \\ &\quad \cup \{d \in D \mid \text{PrevM.time}_k + R \geq \text{NextM.time}_d\}, \\ H(s) &:= \{i \in I \cup \hat{I} \mid \text{FinishM.time}_s + R \geq \text{Arr.time}_i\} \\ &\quad \cup \{d \in D \mid \text{FinishM.time}_s + R \geq \text{NextM.time}_d\}. \end{aligned}$$

2.2 問題設定

ネットワーク $\mathcal{N} := (V, E, c, H)$ の機関車 $k \in K$ から D のいずれかへのパス p をノード列で表現し, パス内の機関車と検査実施ノードを, 現れる順に $s_0, \dots, s_n$ と記す ($s_0 = k$). 集合 $N_p(s_m, s_{m+1})$ を p における s_m と s_{m+1} の間にあるレグとし, $N_p(s_n)$ を最終検査後のレグとする. そして p が以下の条件を満たすとき, 有効なパスと呼ぶ:

$$H(s_m) \supseteq N_p(s_m, s_{m+1}) \quad \forall m \leq n-1,$$

$$H(s_n) \supseteq N_p(s_n).$$

機関車 k の有効なパスの集合を P^k とおく. また k のパス p のコスト c_p^k を p のエッジのコスト和とする.

単機レグ $i \in \hat{I}$ の重連上限を L_i , 同様に B を検査基地集合, T を時間枠集合として, $b \in B, t \in T$ について L_t^b を基地の時間帯の検査容量とする. 以上の入力のもとで本研究の貨物機関車運用整理問題を, $I \cup D$ の各要素を重複なく被覆し, $i \in \hat{I}$ は被覆なしに L_i まで重複を許し, かつ $b \in B$ の $t \in T$ の検査容量 L_t^b を満たすような各機関車の有効なパスの組合せのなかで, コスト和が最小のものを出力する問題とする.

3. 定式化と解法

3.1 整数計画問題としての定式化

機関車 k とその有効なパス p について, レグ i が p に含まれるなら a_{ip}^k を1, さもなくば0とおく (同様に a_{tp}^{bk} を定義). 変数 x_p^k を, k が p を選択するならば1と, さもなくば0とおき, 以下に定式化する:

$$\min. \sum_{k \in K} \sum_{p \in P^k} c_p^k x_p^k$$

$$\text{s.t.} \quad \sum_{k \in K} \sum_{p \in P^k} a_{ip}^k x_p^k = 1 \quad \forall i \in I \cup D, \quad (1)$$

$$\sum_{k \in K} \sum_{p \in P^k} a_{ip}^k x_p^k \leq L_i \quad \forall i \in \hat{I}, \quad (2)$$

$$\sum_{p \in P^k} x_p^k = 1 \quad \forall k \in K, \quad (3)$$

$$\sum_{k \in K} \sum_{p \in P^k} a_{tp}^{bk} x_p^k \leq L_t^b \quad \forall b \in B \quad \forall t \in T, \quad (4)$$

$$x_p^k \in \{0, 1\} \quad \forall k \in K \quad \forall p \in P^k.$$

(1) 式は各レグに1台の機関車を充当する集合分割制約を, (2) 式は単機レグの重連上限を意味する. 機関車は有効なパスの集合から1つ選択し ((3) 式), (4) 式で検査容量を守らせる. 運用整理には迅速さが要求されるため P^k を事前列挙せず列生成法¹⁾を用いるが, 多くの退化が予期されるため, 集合被覆緩和を利用した列生成法を提案する. 全体解法を図3に示す.

3.2 初期化と限定主問題

big- M 法により目的関数の係数が M で制約式行列の列要素が全て1である連続変数 $y \geq 0$ を追加し, 問題を実行可能にする. そして (1) 式の等号を $\geq$ にする集合被覆緩和を行う. 同時に (2) 式の右辺も ∞ とする. 部分集合 $P_\ell^k \subseteq P^k$ を P^k の代わりにし変数 x を線形緩和したものを限定主問題 (RMP_ℓ) とする.

3.3 列生成子問題

(RMP_ℓ) を解いて得られる (1)(2) 式に対する双対価格を λ_i , (3) 式に対するそれを μ^k , (4) 式に対する

Step 1: 【初期化】. $\ell := 0, P_\ell^k := \emptyset, Z_{LB} := 0$ とする. big- M 法により実行可能にする.
 (1)(2) 式を集合被覆緩和する. 緩和解除パラメータ $G \geq 0$ を与える.

Step 2: 【限定主問題 (RMP_ℓ)】. P_ℓ^k 上で線形緩和最適解を求める. $Z_\ell :=$ 線形緩和最適値 とする.

Step 3: 【列生成子問題 (SP_ℓ)】. 最短路問題に帰着し最適解を求める.
 $C_\ell^k := k$ の追加の有効なパス とする. $r_\ell^k := k$ の最短路値 とする.

Step 4: 【下界更新と緩和解除】. $Z_{LB} := \max\{Z_{LB}, Z_\ell + \sum_{k \in K} r_\ell^k\}$ とする.
 もし $G \times (Z_\ell - Z_{LB}) \geq -\sum_{k \in K} r_\ell^k$ であれば
 $\tilde{I}_\ell := (RMP_\ell)$ の解で (1)(2) 式の元の形を違反している制約式の集合 とし, 緩和解除する.

Step 5: 【反復終了判定】.
 もし $Z_\ell > Z_{LB}$ または $\tilde{I}_\ell \neq \emptyset$ ならば, $\ell := \ell + 1, P_\ell^k := P_{\ell-1}^k \cup C_\ell^k$ として Step 2 へ.
 ではなく, もし $Z_\ell = M$ ならば, 整数実行不可能として終了する.

Step 6: 【整数解探索問題 (IP)】. (RMP_ℓ) の制約式をすべて緩和解除し分枝切除法で整数解を求める.
 $Z^{IP} :=$ 整数解の目的関数値 とし, $Z^{IP} < M$ ならば解を出力して終了する.

Step 7: 【実行可能性判定問題 (BP)】. 分枝価格法²⁾ で整数実行可能性を判定する.

図 3 集合被覆緩和を利用した解法
 Fig. 3 Solution method using set-covering relaxation.

ものを ν_i^b とおくと, 列生成子問題 (SP_ℓ) は次のようになる. 各 $k \in K$ について,

$$\begin{aligned} \text{find} \quad & p \in P^k \\ \text{such that} \quad & c_p^k - \sum_{i \in I \cup \hat{I} \cup D} a_{ip}^k \lambda_i \\ & - \mu^k - \sum_{b \in B} \sum_{t \in T} a_{tp}^{bk} \nu_t^b < 0. \end{aligned}$$

上記を満たす有効なパスの探索のため, $\mathcal{N}$ に以下の手順を施し拡大ネットワーク $\mathcal{N}^*$ を構築する.

- Step A: 各レグ $i \in I \cup \hat{I} \cup D$ に対して, 着時刻順位を格納するインデックス Ind_i を与える.
- Step B: 集合 $I \cup \hat{I} \cup D$ から Arr.time_i ($i \in D$ の場合は NextM.time_i) の早い順に要素 i を取り出し, Ind_i に 1 から順の値を付与する.
- Step C: ネットワーク $\mathcal{N}$ を $(|I \cup \hat{I} \cup D| - |H(k)| + 1)$ 個に複製し, $\mathcal{N}^1, \dots, \mathcal{N}^{|I \cup \hat{I} \cup D|}$ と整数値で添え字付けする. 各複製ネットワークを $\mathcal{N}^j = (V^j, E^j, c^j, H^j)$ と表記する.
- Step D: 各 $\mathcal{N}^j$ のエッジ $(s^j, v^j) \in E^j \cap (S^j \times V^j)$ を, $(s^j, v^j | H^j(s^j)|)$ につけ加える.
- Step E: 各 $\mathcal{N}^j$ について, $\text{Ind}_{ij} > j$ であるノード ij へと入るエッジを削除する.

例として図 4, 図 5 を示す. 機関車 $k \in K$ について $\mathcal{N}^*$ で $k^{|H(k)|} \in K^{|H(k)|}$ から $\bigcup_{j' \in \{|H(k)|, \dots, |I \cup \hat{I} \cup D|\}} D^{j'}$ の要素へのパスは有効なパスとなり, 有効なパスは $\mathcal{N}^*$ 上のパスで表現できる. よって (SP_ℓ) は双対価格を $\mathcal{N}^*$ のコストに付加すれば最短路問題に帰着でき, 有向非巡回グラフに対するダイクストラ法で各 k あたり独立に $O(|I \cup \hat{I} \cup D||E|)$ で求まる (特に $h \leq R$ ならば

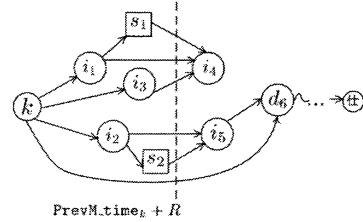


図 4 ネットワーク $\mathcal{N}$
 Fig. 4 Network $\mathcal{N}$.

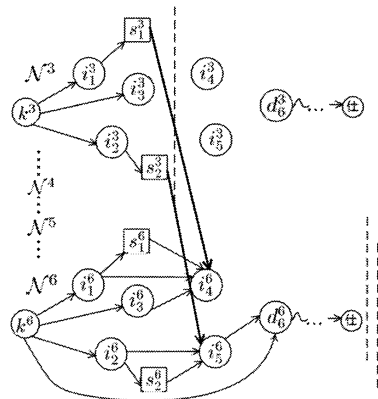


図 5 拡大ネットワーク $\mathcal{N}^*$
 Fig. 5 Extended network $\mathcal{N}^*$.

任意の $s \in S$ で $|H(s)| \geq |I \cup \hat{I}|$ なので $O(|K||E|)$.
 双対価格値を付加した $\mathcal{N}^*$ にダイクストラ法を適用して得られる, コスト値が負の有効なパスを集合 C_ℓ^k に追加する. また k の最短路値を r_ℓ^k とおく.

表 1 実験結果と計算時間 (10 回の平均)
Table 1 Results and solution time (average of 10 trials.)

No.	期間 (h)	変更台数	gap(%)	ℓ	$\sum P_\ell^k $	計算時間 (s)	(RMP_ℓ)	(SP_ℓ)	(IP)	最大計算時間 (s)
1	48	6.0	0.00	8.3	9595.2	2.7	0.9	1.8	0.0	3.5
1	60	6.0	0.00	7.1	9585.9	3.7	0.9	2.7	0.0	4.1
4	48	21.0	0.00	19.4	10874.6	6.9	3.4	3.5	0.0	10.1
4	60	20.0	0.00	28.9	14620.8	20.4	10.4	10.0	0.0	25.3
4	72	21.0	0.00	39.6	17997.6	50.6	26.5	24.1	0.0	71.7
5	48	45.0	0.25	24.6	12460.5	13.0	7.4	4.2	1.4	16.5
5	60	43.5	2.00	29.4	15763.7	29.6	15.3	9.9	4.3	34.7
5	72	44.7	0.80	43.9	20411.3	68.8	37.8	26.0	5.0	78.4

表 2 集合被覆緩和なしでの計算時間 (10 回の平均)
Table 2 Solution time without set-covering relaxation (average of 10 trials.)

No.	期間 (h)	ℓ	$\sum P_\ell^k $	計算時間 (s)
4	48	26.4	17068.8	18.8
4	60	45.4	27991.1	76.5
4	72	85.1	49856.4	309.6

3.4 下界更新と緩和解除, 反復終了判定

双対価格 λ_i, ν_ℓ^k を乗数とみれば最短路問題の解は (1)(2)(4) 式をラグランジュ緩和した問題の解である。これらの乗数と μ^k は線形計画問題の双対最適解であり, 相補性条件より $r_\ell^k \leq 0$ が成り立つ。ラグランジュ緩和問題の最適値は $Z_\ell + \sum_{k \in K} r_\ell^k$ と一致し¹⁾, 最短路値を元の問題の下界として採用できる。ここで G に依存して, (1)(2) 式で元の形の制約を違反しているレグ集合 $\tilde{I}_\ell$ を抽出し, 元に戻す。

式 $r_\ell^k = 0$ と $C_\ell^k = \emptyset$ は等価なため, Z_ℓ と Z_{LB} が一致しなければ Step 2 へ戻る。集合 $\tilde{I}_\ell$ が非空の場合も制約式系が変化したため同様にする。

4. 計算機実験と今後の課題

実際の貨物列車ダイヤ・機関車運用計画に遅れや運用が発生した事例を適用し, 提案手法を検証した。結果が表 1 で, 運用計画が変更される台数, MIP ギャップ, 反復回数, 総生成列数, 実計算時間と内訳を示している。試行により解らが異なるのは (RMP_ℓ) で双対最適解が一意ではないことによる。整理期間を長くすると台数が増えているが整理内容としてはシンプルになり, 計算時間も 2, 3 分という実務要請を満たしている。集合被覆緩和なしの結果も表 2 に示す。

対象は日本最大の運行頻度を持つ地域の機関車 142 (= $|K|$) 台の担当レグとし, 検査周期は車両形式により 72 ないし 96 時間とした。表 3 に事例の概要を, 運用整理期間別の問題サイズを表 4 に示す。運用矛盾台数とは, 変更ダイヤのもとでは所定の計画通りに運用できない機関車の数を表す。環境は

表 3 ダイヤ乱れ事例
Table 3 Disruption cases.

No.	遅延数	平均遅延 (h)	運用矛盾台数
1	0	12	3
4	0	15	10
5	2	50	24

表 4 問題のサイズ (3 事例の平均)
Table 4 Problem size (average of 3 cases.)

整理期間 (h)	$ I \cup D $	$ V $	$ E $
48	758.7	1373.0	27817.7
60	937.7	1656.0	40683.7
72	1110.7	1943.0	57269.0

Core 2 Quad 2.66GHz, 3.5 GB RAM, WinXP で, JavaSE 6 で実装し (RMP_ℓ) と (IP) に CPLEX 11.0 を使用, $G = 3.0$ とした ((RMP_ℓ) と (IP) は 1 コアで, (SP_ℓ) は 4 コアで計算)。

今後の研究として, 運用整理が実行不可能な場合の列車の優先度に応じた機関車割り当て問題がある。機関車・乗務員の統合的な運用整理も大きな課題である。

参考文献

- Desaulniers, G. et al.(eds.): *Column Generation*, Springer (2005).
- Rezanova, N. J. and Ryan, D. M.: The train driver recovery problem — a set partitioning based model and solution method, *Proceedings of RailHannover2007* (2007).
- Walker, C. G. et al.: Simultaneous disruption recovery of a train timetable and crew roster in real time, *Computers and Operations Research*, Vol. 32, No. 8, pp. 2077–2094 (2005).
- 富井規雄: 列車ダイヤのひみつ — 定時運行のしくみ, 成山堂書店 (2005).
- 富井規雄ほか: 利用者の不満を最小にする列車運転整理アルゴリズム, 情報処理学会論文誌: 数理モデル化と応用, Vol. 46, No. SIG 2(TOM 11), pp. 26–38 (2005).