駅構内入れ換え計画問題に関する研究

坂口　隆

(電気通信大学大学院電気通信学研究科情報工学専攻　現所属：㈶鉄道総合技術研究所)
指導教官　田村明久 助教授

1. はじめに

駅構内入れ換え計画とは，ターミナル駅などでの複数車両の移動手順を決める計画である．ひとつの駅の計画に数百時間を費やすこともあることから自動化が望まれている作業の一つであり，近似解法の研究が現在行われているところである．本論文では入換回数を最小化する最適化問題に関するアルゴリズムとその理論計算量について述べ，実問題へ適用した結果について考察する．

2. 駅構内入れ換え計画問題

図1は車両入換の例である．XやYの車両は到着と出発の番線が異なるので入換が必要であり，またZのように到着から出発の間に同じ番線を他が使用する場合も入換が必要となる．列車の着発時刻・番線などが与えられてこの図のような入換手順を求める計画を，駅構内入れ換え計画という．

入換計画では，複数の車両が一度に同じ番線を使用しないことや，入換する線路上で車両同士が競合しないように多くの制約がある．実際には多くの目的，評価基準が存在するが，本研究では必要な制約を満たし入換回数を最小化する問題を扱うこととし，これを駅構内入れ換え計画問題と呼ぶ．

3. 定式化

本研究では，駅構内入れ換え計画問題をSDPPとして定式化して解く方法を提案する．

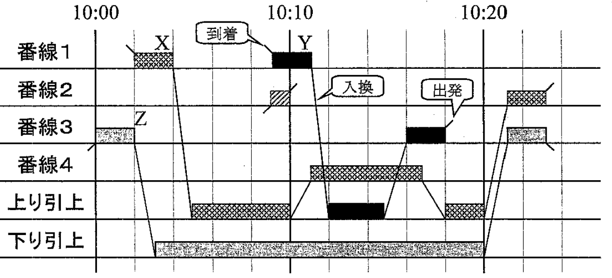

図1　車両入換の例

3.1 SDPP

頂点集合 V，辺集合 E にリンク集合 L を伴ったグラフ $G=(V,E,L)$，$L=\{(s_j,t_j)|j=1,2,\cdots,k,s_j\in V,t_j\in V\}$ をリンク付きグラフと呼び，リンク (s_j,t_j) の頂点を結ぶ k 本の頂点非共有パス $P_1,P_2,\cdots,P_k$ を求める問題を頂点非共有パス問題という．頂点非共有パスとは頂点が互いに素なパスの集合である．さらに辺の長さの和を最小化する問題を最短頂点非共有パス問題（Shortest Disjoint Path Problem）といい，ここではSDPPと略す．

これらの問題について，これまでの研究から一般の有向，無向グラフの場合の計算量はNP-完全であることが知られているが，非巡回有向グラフに限定すればC.L.Li等が $O(k|V|^{k-1}|E|)$ 時間解法を示している[1]．これをLiのアルゴリズムと呼ぶ．

3.2 SDPPへの定式化

計画上の時刻を t_0 秒間隔とし，車両が時刻 t に番線 b にいる状態を (b,t) と表す．入換車両がとることのできる状態 (b,t) の集合を頂点集合 V，可能な状態遷移 $((b,t),(b',t'))$ の集合を有向辺集合 E，入換車両 j が駅に到着した状態を $s_j=(b_j,t_j)$，駅を出発するときの状態を $d_j=(b_j',t_j')$, $L=\{(s_j,d_j)|j=1,2,\cdots,k\}$ とすることによって得られる有向グラフ $G=(V,E,L)$ 上で s_j から d_j までのパス P_j は車両 j の入換手順を表す．したがって端点の番線が同じ辺の長さを0，それ以外を1としこれに対するSDPPを解けば，同時に複数の車両が同一番線を使用しないという条件の上で入換回数が最小の入換手順が得られる．他の制約は G の変形，リンクの追加によってSDPPに取り込むことができる．

4. SDPPの解法

4.1 Liのアルゴリズムとその改善

Liのアルゴリズムでは与えられたグラフ $G=(V,E,L)$, $|L|=k$ に対して，頂点のトポロジカルオーダの一つ $\boldsymbol{\pi}$ を用いて k 次元グラフ $\bar{G}$ を構成する．$\bar{G}$ の

頂点は，$\langle v_1, v_2, \cdots, v_k \rangle$ のような k 個の G の頂点の組であり，第 j 要素は j 番目のリンクを構成するパス上の頂点に対応する．

駅構内入れ換え計画問題では k が非常に大きな値となるので，このアルゴリズムをそのまま適用しても実時間では計算できない．本研究では $\bar{G}$ の次の条件を満たす頂点によって誘導される部分グラフ（**適合部分グラフ**）を考えた．

（1） $\bar{s}$ から到達可能であること．

（2） j 番目の要素 v_j は $\boldsymbol{\pi}$ の並びにおいて s_j と t_j の間の頂点であること．

そして，適合部分グラフ上で Li のアルゴリズムと同様に最短路問題を解くアルゴリズム SDPP 1 を開発した．

4.2 リンク幅と計算量

SDPP 1 の計算量を評価するために以下で定義されるリンク幅 $W(\boldsymbol{\pi})$ という概念を導入した．ここで $\pi(v)$ は $\boldsymbol{\pi}$ の並びにおける v の順位である．

$$w(\boldsymbol{\pi}, v) \stackrel{\text{def}}{=} \{(s,t) | (s,t) \in L, \pi(s) \leq \pi(v) < \pi(t)\}$$

$$W(\boldsymbol{\pi}) \stackrel{\text{def}}{=} \max_{v \in V} |w(\boldsymbol{\pi}, v)|$$

$w(\boldsymbol{\pi}, v)$ は $\boldsymbol{\pi}$ の頂点列における v をまたぐリンクの集合であり，$W(\boldsymbol{\pi})$ はその大きさの最大値である．G に対して $\boldsymbol{\pi}$ が与えられたとき，G のリンク幅 $w = W(\boldsymbol{\pi})$ と適合部分グラフ $\tilde{G}$ の間に次の二つの定理が成立する．

定理 1　$\tilde{G}$ の頂点の数は，高々 $|V|^w$ 個である．

定理 2　$\tilde{G}$ の辺の数は，高々 $w|V|^{w-1}|E|$ 個である．

SDPP 1 の計算量は上の定理から $O(w^2|V|^{w-1}|E|)$ 時間，$O(w|V|^w)$ 領域となる．駅構内入れ換え計画問題では w の上限が駅設備によって定まるので，変数である k に比べ非常に小さな定数として評価できる．したがって SDPP 1 は同問題を多項式時間・領域で計算でき，しかも理論計算量だけでなく絶対的な計算時間，計算領域の面でも有効な解法であると言える．

5. 実問題への適用

実問題では w が 10 程度となるので，上記の結果に加え探索方法や頂点の記憶方法の改善による実計算時間・領域の縮小が必要である．そこで $\bar{s}$ から $\bar{t}$ への最短路探索に A* アルゴリズムを適用した．A* は $\bar{s}$ から $\bar{v}$ までの最短距離 $d(\bar{s}, \bar{v})$ と $\bar{v}$ から $\bar{t}$ までの最短距離 $d(\bar{v}, \bar{t})$ の推定値 $h(\bar{v})$ の和が小さな頂点 $\bar{v}$ を優先的に探索する方法で，$h(\bar{v})$ に良い下界を与えられれば最適性の保証の下で探索空間を効率的に縮小できる[2]．

$d(\bar{v}, \bar{t})$ は頂点を共有しないパス $P_{v_j t_j} (j=1 \cdots k)$ の長さの和の最小値である．したがって頂点非共有の条件を緩和した最短路長の和 $\sum_{j=1..k} d(v_j, t_j)$ は $d(\bar{v}, \bar{t})$ の下界を与える．ここではこれを $h(\bar{v})$ として定義した．

6. 理論と実用可能性の検証

w＝一定の条件の下でのリンク数と計算時間の関係を調べる実験を行い，$|V|, |E|, w$ が一定ならば計算時間は k に依らず w によって決まる上限があることを検証した．

また A* アルゴリズムなどによる改善の効果を調べる実験を行った．$\tilde{G}$ の全探索を基本とするダイクストラ法などのアルゴリズムでは w の増加に対して理論計算量が示すとおり指数的に計算時間が増加するが，これに対し A* を適用して改善したアルゴリズムでは探索頂点数を劇的に削減でき，実験的には w の増加に対しても現実的な時間（多項式時間）で計算可能であることが分かった．

最後に JR のある中規模駅の駅設備，入出車両データを使用し，仮想的な入換車両を含む厳しい条件の問題を設定して，実問題への有効性の検証を行った．計画時間 76 分，入換車両数 6 本を対象としたグラフ（$|V|=3,300$，$|E|=5,871$，$k=118$，$w=12$）に対し，数分で最適解（入換回数 22 回）を求めることが出来た．

参考文献

[1] C. L. Li, S. T. McCormick, and D. Simchi-Levi, Finding Disjoint Paths with Different Path-Costs: Complexity and Algorithms. Networks, 22, (1992), pp. 653-667.

[2] N. J. Nilsson（白井他 訳），人工知能の原理，日本コンピュータ協会，1983.