

# 経路・出発時刻の選択を内生化した 動的利用者均衡配分の効率的解法

河向 隆志<sup>1</sup>・酒井 高良<sup>2</sup>・赤松 隆<sup>3</sup>

<sup>1</sup>正会員 株式会社ドーコン (〒 004-8585 札幌市厚別区厚別中央 1 条 5 丁目 4 番 1 号)  
E-mail: kawa30176672@gmail.com

<sup>2</sup>学生会員 東北大学 大学院情報科学研究科 (〒 980-8579 仙台市青葉区荒巻字青葉 6-6-06-408)  
E-mail: takara.sakai.t1@dc.tohoku.ac.jp

<sup>3</sup>正会員 東北大学教授 大学院情報科学研究科 (〒 980-8579 仙台市青葉区荒巻字青葉 6-6-06-408)  
E-mail: akamatsu@plan.civil.tohoku.ac.jp

本研究では、経路・出発時刻同時選択を内生化した動的利用者均衡 (DUE) 配分を解くためのアルゴリズムを提案する。より具体的にはまず、DUE 配分が起点出発時刻別に分解できる特性を活用し、起点出発時刻ベースの座標系のもとで均衡条件を定式化する。次に、この定式化が、簡明な数学的構造を持つ線形相補性問題 (LCP) に帰着する事実に基づき、LCP に対する大域収束的 Newton 法に基づくアルゴリズムを採用する。更に、解くべき LCP がグラフ論的構造から疎な係数行列を持つことを活用して、大規模ネットワークに対しても適用可能なアルゴリズムを開発する。提案アルゴリズムが大規模ネットワークに対しても効率的に均衡解を得られることを、数値実験によって明らかにする。

**Key Words:** 動的利用者均衡配分, 線形相補性問題, 1 起点多終点, 大域収束的 Newton 法

## 1. はじめに

時々刻々と変化する交通流に応じた交通制御手法を構築するためには、動的な交通流を明示的に扱った数理モデルが欠かせない。動的な交通流を扱ったモデルは多数提案されているが、中でも動的利用者均衡 (DUE) 配分は、車両の滞留と時間経過の概念を明示的に表現している点で、ベンチマークとして重要なモデルである。そのため、実際の大规模ネットワークに対する配分モデルとして利用されることも多く、効率的計算法の開発は欠かせない。

DUE 配分は大きく、経路のみを選択するモデル (経路選択 DUE 配分) と、経路・出発時刻を同時に選択するモデル (同時選択 DUE 配分) に大別され、両モデルのアルゴリズムは数多く提案されている。例えば、同時選択 DUE 配分については、Szeto and Lo<sup>1)</sup>, Jang et al.<sup>2)</sup>, Friesz et al.<sup>3)</sup> などが代表的である。また、経路選択 DUE 配分については Bliemer and Bovy<sup>4)</sup>, Varia and Dhingra<sup>5)</sup>, Long et al.<sup>6)</sup> などの研究が存在する。しかしながら、これらの提案アルゴリズムの多くは、経路費用関数の単調性を仮定したものを除き、大規模ネットワークに対して収束が保証されていない。ここで、経路費用関数の単調性は一般に保証できないため、結局のところ厳密解への収束保証がなされたアルゴリズムはほとんど存在しない。また、これらのアルゴリズムは、

(1) 経路別に定義された変数を使用し、(2) Euler 座標系 (絶対時刻系) で定式化しているため、経路列挙および、経路間の交通の相互作用の考慮が必要となり、数理計画理論に基づいた厳密解法開発の妨げとなっている。一方、Akamatsu<sup>7)</sup> の研究では、経路選択 DUE 配分を非線形相補性問題 (NLCP) として定式化することで、数理計画理論に基づいた効率的解法 (大域収束的 Newton 法ベースのアルゴリズム) を示している。ここでの定式化は、これまでに示した従来研究とは異なり、Lagrange 座標系 (起点出発時刻ベースの座標系) で記述し、ノード・リンク別に定義された変数を使用している。

本研究では、同時選択 DUE 配分を対象として、収束が保証され、大規模ネットワークにも適用可能な効率的アルゴリズムを提案する。これを実現するための方法として、経路選択 DUE 配分の効率的解法である Akamatsu<sup>7)</sup> を出発時刻選択も内生化したモデルにも適用できるように拡張する。まず、DUE 配分の均衡条件を Lagrange 座標系のもとで定式化する。次に、Lagrange 座標系のもとで定式化された問題が、ノード・リンク別に定義された変数を用いた、簡明な数学的構造を持つ線形相補性問題 (LCP) に帰着することを示す。そして、この事実に基づき、DUE 配分の解法として LCP に対する大域収束的 Newton 法に基づくアルゴリズムを採用する。更に、解くべき LCP は、グラフ論的構造から疎な係数行列を持つ。このことを利用して、大規

模ネットワークに対しても適用可能なアルゴリズムを開発する。その後、提案アルゴリズムの収束性及び効率性を、従来研究で提案されているアルゴリズムとの比較実験を通じて検証する。

本稿の構成は以下の通りである。2章では、対象とする交通ネットワーク並びに DUE 配分を定義する。3章では、同時選択 DUE 配分を LCP として定式化する。4章では、大域収束的 Newton 法に基づくアルゴリズムを提案する。5章では、数値実験を通して、提案アルゴリズムが効率的に均衡解を得られることを示す。

## 2. モデル

### (1) 交通ネットワークとその表現

本稿では、最も基本的な、OD ペアが 1 起点多終点であるネットワークのみを対象とする。だが、OD ペアが多起点 1 終点であるネットワークに対しても、起点に関して条件を追加すれば同様に定式化できる。このネットワークでは、起終点ノード以外のノード (通過ノード) が存在しても良いものとする。

本研究で定式化する DUE 配分は、 $L$  個の要素を持つリンク集合  $\mathcal{L}$ 、 $N+1$  個の要素を持つノード集合  $\mathcal{N}$  から構成される交通ネットワーク  $G(\mathcal{N}, \mathcal{L})$  で定義する。このうち、ノード集合は、 $N_1$  個の要素を持つ終点ノード集合  $\mathcal{N}_1$  と  $N_2$  個の要素を持つ通過ノード集合  $\mathcal{N}_2$ 、そして唯一の起点ノードに分割できる。ネットワークの接続関係は、起点に対応する行を削除した  $N \times L$  ノード・リンク接続行列  $\mathbf{A}$  によって表現する。

ネットワークを構成するリンクの通過渋滞の表現には、Point Queue モデルと First-In-First-Out (FIFO) 原則を採用する。すなわち、リンクに待ち行列が発生していない場合には、流入フローは、流入から自由移動時間  $c_{ij}$  だけ経った後で流入フローと同一の流率で流出する。一方、リンクに待ち行列が発生している場合には、リンク流入順に容量  $\mu_{ij}$  だけの流出率で流出する。

### (2) 経路・出発時刻同時選択 DUE 配分の定義

本研究では、1 起点多終点ネットワークにおける同時選択 DUE 配分を扱う。Euler 座標系 (絶対時刻  $\hat{t}$  系) において、経路  $r$  ベースの変数を用いると、同時選択 DUE 配分は、以下の要素：

- $Q_d$  : 唯一起点から終点ノード  $d$  への OD 交通需要
- $\hat{t}_w$  : 希望到着時刻
- $s(\hat{t}, \hat{t}_w)$  : スケジュール遅れ関数
- $c_r(\mathbf{f}(\hat{t}, \hat{t}_w))$  : 経路  $r$  旅行時間

を与件としたときに、DUE 状態における時々刻々の経路フロー  $f_r(\hat{t}, \hat{t}_w)$  を求める問題となる。ここで、DUE 状態とは、どの時刻においても、どの利用者も自分だ

けが経路ないしは出発時刻を変更しても自分の経験する交通費用を改善できない状態のことである。このとき、経路交通流率  $\mathbf{f}$  のもとで、利用者の経験する経路  $r$  交通費用  $C_{r,d}(\mathbf{f}(\hat{t}, \hat{t}_w))$  は、経路旅行時間とスケジュール遅れの和として表される：

$$C_{r,d}(\mathbf{f}(\hat{t}, \hat{t}_w)) = c_r(\mathbf{f}(\hat{t}, \hat{t}_w)) + s(\hat{t}_w) \geq 0 \quad (1)$$

以上より、DUE 状態は利用者の行動原理を表す均衡条件および、交通フローが満たすべき物理的条件を表すフロー保存則を用いて表すことができる：

- 利用者の行動原理

$$0 \leq f_{r,d}(\hat{t}, \hat{t}_w) \perp C_{r,d}(\mathbf{f}(\hat{t}, \hat{t}_w)) - \rho_d \geq 0 \quad (2)$$

- フロー保存則  $\forall r \in \mathcal{R}_d, \forall d \in \mathcal{D}, \forall \hat{t} \in [0, T]$ .

$$\sum_{r \in \mathcal{R}_d} \int_0^T f_{r,d}(\hat{t}, \hat{t}_w) dt = Q_d \quad \forall d \in \mathcal{D}. \quad (3)$$

ここで、 $\rho_d$  は均衡交通費用、 $\mathcal{R}_d$  は終点が  $d$  である経路の集合、 $\mathcal{D}$  は終点ノード集合である。

### (3) Lagrange 座標系と DUE 配分の時間分解特性

DUE 配分の解の収束保証のためには、経路費用関数  $C_{r,d}(\mathbf{f}(\hat{t}, \hat{t}_w))$  の数学的特性を解析する必要がある。しかし、 $C_{r,d}(\mathbf{f}(\hat{t}, \hat{t}_w))$  は時刻  $\hat{t}$  別の経路交通流率に依存するため、解析の際には対象となる経路を列挙する必要がある。しかも、一般のネットワークでは経路交通流率どうしが相互干渉している。そのため、経路費用関数  $C_{r,d}(\mathbf{f}(\hat{t}, \hat{t}_w))$  の特性を把握することは困難である。

上記の困難は、問題を Lagrange 座標系で記述することにより解決できる。この考えに基づいて定式化を進めるにあたっては、DUE 配分の基本特性に関する次の観察が有用である<sup>8)</sup>：「DUE 状態では、同一起点を同一時刻に出発した利用者は、終点までに利用される各ノードへ同一時刻に到着する。また、利用者は終点までに利用される各ノードへ出発時刻順に到着する。」この特性と FIFO 原則より、ある起点出発時刻  $t$  に対応した DUE 状態は、時刻  $t$  以前に出発したフローに関する情報のみから決定されることが分かる。

以上より、次のような、DUE の時間分解特性が得られる：「Lagrange 座標系の下では、1 起点多終点ネットワークにおける DUE 配分の均衡条件は、ノード・リンク別に定義された変数を用いて定式化することができる」。この特性により、Euler 座標系で記述した場合に比べて、より簡明な数学的構造を持つ問題として表現することができる。

### 3. DUE 配分の定式化

3.1 節で示す均衡条件は、Lagrange 座標系<sup>8)9)</sup> で定義された変数を用いて、単純な形で表現できる。以下では、起点出発時刻  $t \in [0, T]$  別に定義された変数を用いて定式化を行う。

- $q_i(t)$  : OD 交通流率
- $y_{ij}(t)$  : リンク  $(i, j)$  交通流率
- $w_{ij}(t)$  : リンク  $(i, j)$  で経験する待ち行列遅れ時間
- $\tau_i(t)$  : 最も早くノード  $i$  に到着する時刻
- $\rho_i$  : 起点からノード  $i$  までの均衡交通費用

上記の変数を要素として持つベクトルを、それぞれ  $\mathbf{q}(t), \mathbf{y}(t), \mathbf{w}(t), \boldsymbol{\tau}(t), \boldsymbol{\rho}$  とする。更に、議論を簡単にするため、利用者の希望出発時刻  $t_w$  はみな同一であり、且つ同一のスケジュール遅れ関数  $s(t)$  を持つとする。ここに、 $s(t)$  は  $t_w$  で最小値をとる、起点出発時刻に関する凸関数である。

#### (1) 均衡条件

DUE 状態は、出発時刻別に定義された変数を用いて、以下の条件が同時に成立した状態として定式化される。

##### a) 待ち行列条件

待ち行列が Point Queue モデルで仮定されているとき、均衡状態でのリンク移動時間は次式で表される。

$$c_{ij} + w_{ij}(t) = \max \left[ c_{ij} + \frac{dw_{ij}(t)}{dt} + \frac{y_{ij}(t)}{\mu_{ij}} - \frac{d\tau_i(t)}{dt}, c_{ij} \right] \quad \forall ij \in \mathcal{L}, \forall t. \quad (4)$$

さらに、式 (4) の両辺からリンク自由移動時間  $c_{ij}$  を減じると、待ち行列条件は次の相補性条件に帰着する。

$$0 \leq w_{ij}(t) \perp -y_{ij}(t) + \mu_{ij} \left( \frac{dw_{ij}(t)}{dt} + \frac{d\tau_i(t)}{dt} \right) \geq 0 \quad \forall t. \quad (5)$$

##### b) 経路選択条件

DUE 状態では、どの利用者も、自分だけが経路を変更しても自分の交通費用を改善できない。このことは、最適な出発時刻  $t$  のもとでは、経路の変更によって移動時間を改善できないことを意味する。起点出発時刻が  $t$ 、ノード  $i, j$  への到着時刻がそれぞれ  $\tau_i(t), \tau_j(t)$  であるから、リンク  $(i, j)$  が最短経路を構成するリンクであるための条件は次の相補性問題の形に帰着する。

$$0 \leq y_{ij}(t) \perp c_{ij} + w_{ij}(t) + \tau_i(t) - \tau_j(t) \geq 0 \quad \forall ij \in \mathcal{L}, \forall t. \quad (6)$$

##### c) 出発時刻選択条件

DUE 状態では、どの利用者も、自分だけが起点出発時刻を変更しても自分の交通費用を改善できない。ここで、出発時刻が  $t$  であるときの交通費用は、移動時間とスケジュール遅れ費用の和  $(\tau_i(t) - t) + s(t)$  に相当する。これは、DUE 状態で選択されている出発時刻では

時刻によらない均衡交通費用  $\rho_i$  に等しくなる。すなわち、全ての終点ノードに関して、出発時刻選択条件は次式に帰着する。

$$0 \leq q_i(t) \perp (\tau_i(t) - t) + s(t) \geq \rho_i \quad \forall i \in \mathcal{N}_1, \forall t. \quad (7)$$

##### d) 各ノードでのフロー保存則

フローが流れているリンクでは、DUE 状態では経路選択条件より  $\tau_i(t) + c_{ij} + w_{ij}(t) = \tau_j(t)$  が成立する。ここから、FIFO 条件を考慮に入れたフロー保存則は、終点ノード及び通過ノードについて、それぞれ以下の式に帰着する。

$$\sum_{k \in NO(i)} y_{ki}(t) - \sum_{l \in NI(i)} y_{il}(t) - q_i(t) = 0 \quad \forall k \in \mathcal{N}_1, \forall t. \quad (8)$$

$$\sum_{k \in NO(i)} y_{ki}(t) - \sum_{l \in NI(i)} y_{il}(t) = 0 \quad \forall k \in \mathcal{N}_2, \forall t. \quad (9)$$

ここに、 $NI(i)$  はノード  $i$  へ流入するリンクの上流側ノードの集合、 $NO(i)$  はノード  $i$  から流出するリンクの下流側ノードの集合である。

##### e) OD 間フロー保存則

終点ノードに関して、対象時間帯  $[0, T]$  を通じてノード  $i$  から発生する交通需要  $Q_i$  は、各時刻  $t$  に起点を出発する OD 交通流率の総和である。

$$Q_i = \int_0^T q_i(t) dt \quad \forall i \in \mathcal{N}_1. \quad (10)$$

#### (2) 等価な LCP としての表現

3.1 節で示した均衡条件は、ベクトル・行列表記するとそれぞれ以下のように表せる。

- 待ち行列条件

$$\mathbf{0} \leq \mathbf{w}(t) \perp -\mathbf{y}(t) + \mathbf{C} \frac{d\mathbf{w}(t)}{dt} + \mathbf{C} \mathbf{A}_+^T \frac{d\boldsymbol{\tau}(t)}{dt} \geq \mathbf{0} \quad \forall t. \quad (11)$$

- 経路選択条件

$$\mathbf{0} \leq \mathbf{y}(t) \perp \mathbf{c} + \mathbf{w}(t) + \mathbf{A}^T \boldsymbol{\tau}(t) \geq \mathbf{0} \quad \forall t. \quad (12)$$

- 出発時刻選択条件

$$\mathbf{0} \leq \mathbf{q}_1(t) \perp (\boldsymbol{\tau}_1(t) - t) + s(t) \mathbf{1}_{N_1} - \boldsymbol{\rho}_1 \geq \mathbf{0} \quad \forall t. \quad (13)$$

- 各ノードでのフロー保存則

$$\mathbf{0} \leq \boldsymbol{\tau}_1(t) \perp \mathbf{A}_1 \mathbf{y}(t) + \mathbf{q}_1(t) \geq \mathbf{0} \quad \forall t. \quad (14)$$

$$\mathbf{0} \leq \boldsymbol{\tau}_2(t) \perp \mathbf{A}_2 \mathbf{y}(t) \geq \mathbf{0} \quad \forall t. \quad (15)$$

- OD 間フロー保存則

$$\mathbf{0} \leq \boldsymbol{\rho}_1 \perp \int_0^T \mathbf{q}_1(t) dt - \mathbf{Q}_1 \geq \mathbf{0} \quad (16)$$

ここに、 $\mathbf{C}$  は対角線分がリンク容量  $\mu_{ij}$  となる  $L \times L$  対角行列、 $\mathbf{c}$  はリンク自由移動時間  $c_{ij}$  を要素に持つベクトル、 $\mathbf{Q}_1$  は終点ノード別交通需要をノード番号順に並べた  $N_1$  次元ベクトル、 $\mathbf{A}_+$  は  $\mathbf{A}$  の  $-1$  要素を  $0$  にした行列、 $\mathbf{A}_1, \mathbf{A}_2$  は、行列  $\mathbf{A}$  をそれぞれ  $\mathcal{N}_1, \mathcal{N}_2$  に属するノードに対応する行ごとに分割した、それぞれ  $N_1 \times L$

行列及び  $N_2 \times L$  行列,  $\Delta = d/dt$  は起点出発時刻  $t$  に関する 1 階微分を表す.

均衡条件 (11) – (16) は連続時間系で定式化されている. ここで, 時刻を離散近似し, 適当な行列  $M$  並びにベクトル  $b$  を用いて書き直すことにより, 次の命題 1 が導かれる:

[命題 1] 1 起点多終点ネットワークでの経路・出発時刻同時選択 DUE 配分は, 以下に示す LCP と等価である. Find  $x$  such that  $0 \leq x \perp F(x) \equiv Mx + b \geq 0$ , where  $M, b$ , and  $x$  are defined as

$$M \equiv \begin{bmatrix} & & & M_{14} & -1_K \otimes I_{N_1} \\ & & I_{KL} & M_{24} & \\ & -I_{KL} & \Delta_K \otimes C & M_{34} & \\ -M_{14}^\top & -M_{24}^\top & & & \\ \mathbf{1}_K^\top \otimes I_{N_1} & & & & \end{bmatrix}$$

$$b \equiv \begin{bmatrix} (s-t) \otimes \mathbf{1}_{N_1} \\ \mathbf{1}_K \otimes c \\ \mathbf{0} \\ \mathbf{0} \\ -Q_1 \end{bmatrix}, \quad x \equiv \begin{bmatrix} q_1 \\ y \\ w \\ \tau \\ \rho_1 \end{bmatrix} \quad (17)$$

ここに,  $M_{14} \equiv [I_{KN_1} \ O]$ ,  $M_{24} \equiv [I_K \otimes A_1^\top \ I_K \otimes A_2^\top]$ ,  $M_{34} \equiv [\Delta_K \otimes CA_{1+}^\top \ \Delta_K \otimes CA_{2+}^\top]$ .  $K$  は離散近似した時刻の総数,  $\mathbf{1}_K$  は  $K$  次元 1 ベクトル,  $I_n$  は  $n \times n$  単位行列,  $A_{1+}, A_{2+}$  は, 行列  $A_+$  を, それぞれ  $N_1, N_2$  に属するノードに対応する行ごとに分割したそれぞれ  $N_1 \times L$  行列,  $N_2 \times L$  行列,  $s$  は  $s(t)$  を時刻順に並べた  $K$  次元ベクトル,  $\otimes$  は Kronecker 積である.

経路・出発時刻同時選択 DUE 配分を LCP として定式化したことにより, 問題の構造を明快に表現することができた. しかも, LCP は数理計画分野ではよく知られた問題であり, その解法については多くの研究蓄積が存在する. 以降では, LCP を解くための効率的なアルゴリズムの一例として, 大域収束的 Newton 法に基づいたアルゴリズムを提案する.

## 4. LCP に対する大域収束的 Newton 法

### (1) アルゴリズムの基本的な考え方

本研究では, LCP を解く枠組みとして, Facchinei, Soares<sup>10)</sup> の提案した大域収束的 Newton 法に基づいたアルゴリズムを採用する. このアルゴリズムは, 広いクラスの相補性問題について大域的収束が保証され, 解への収束が高速であるという特徴を有している. 大域収束的 Newton 法で主になされる演算は, (a) 改訂ベクトルの決定, (b) 解の改訂, (c) 収束判定である. このうち, (a) の概略は以下の通りである.

いま,  $x^n$  は  $n$  回目計算にて既に得られた解とし,  $x^n$  の非零要素に対応した変数集合  $\beta$  と残りの変数集合  $\alpha$  が分かっているとす. すると, それに対応する  $x$  と  $F$  は  $x = [x_\beta, \mathbf{0}]$ ,  $F = [\mathbf{0}, F_\alpha]$  と表せる. このとき, 連立

方程式  $F_i(x_\beta, \mathbf{0}) = 0 (i \in \beta)$  を解くことによって, LCP の解を得ることができる. ここで, もし  $\nabla_\beta F_\beta(x^*)$  が逆行列を持たなければ, この方程式は大域収束的 Newton 法を用いて解くことができる. ここに,  $\nabla_\beta$  は  $x_\beta$  に関する微分演算子である.

$$[\nabla_\beta F_\beta] d_\beta^n = -F_\beta(x^n) + \nabla_\alpha F_\beta x_\alpha^n \quad (18)$$

を解いて  $n$  回目改訂ベクトル  $d_\beta^n$  を求め,  $x_\beta^{n+1} := x_\beta^n + d_\beta^n$  で解を改訂すれば相補性問題の解が得られる.

さて, 一般には, 解を得るまでは真の  $\alpha, \beta$  は判明しない. しかし, 各反復で適切な近似集合を作成し, それを利用すれば, 早い収束が期待できる. Facchinei, Soares<sup>10)</sup> では, 以下の基準に基づいて近似集合を作成している:

$$\alpha = \{i | x_i^n \leq \epsilon F_i(x^n)\}, \beta = \{i | x_i^n > \epsilon F_i(x^n)\} \quad (19)$$

ここに,  $\epsilon$  は正の定数. このとき,  $d_\alpha^n$  は  $d_\alpha^n = -x_\alpha^n$  として決定される.

更に, (b),(c) に関しては, アルゴリズムの大域的収束性を保証/強化するために, LCP に対する merit 関数  $\Psi(x)$  を利用する. merit 関数とは, ベクトル  $x^*$  が LCP の解であるとき, 関数値  $\Psi(x^*)$  が 0 をとるような非負関数である. ここでは, 以下に示す関数を, LCP に対する微分可能な merit 関数として採用する<sup>11)</sup>:

$$\Psi(x) \equiv \sum_i \psi(x_i, F_i(x))^2 \quad (20)$$

$$\text{where } \psi(x, y) = \sqrt{x^2 + y^2} - (x + y)$$

この関数を使用すれば, Step Size の調整においても凸計画問題と同様に 1 次元探索を行うことができる. これにより, 緩和法等の従来解法に比べて, アルゴリズムの大域的収束性が強化される.

以上より, 大域収束的 Newton 法アルゴリズムは以下のようにまとめられる:

#### Step.0 Initialization

$n := 0$  とする.  $(\epsilon, \bar{\rho}, \bar{p})$  の値及び初期解  $x^n$  を設定する. (但し,  $\epsilon > 0, \bar{\rho} > 0, \bar{p} > 2$ )

#### Step.1 Stopping Test

収束判定が満たされれば, 計算終了. そうでなければ, Step.2 へ.

#### Step.2 Direction Finding

- (a) 変数分割集合  $\alpha, \beta$  を, 式 (19) に基づいて決定する.
- (b) 線形連立方程式 (18) を解き,  $d_\beta^n$  を決定する. また,  $d_\alpha^n := -x_\alpha^n$  とする. その後, Step.3 へ.
- (c) もし線形連立方程式 (18) が解けないか, あるいは  $d^n$  が  $\nabla \Psi(x^n) \cdot d^n \leq -\bar{\rho} |d^n|^{\bar{p}}$  を満足しないなら,  $d^n := -\nabla \Psi(x^n)$  とする. その後, Step.3 へ.

#### Step.3 Line Search and Move

一次元探索問題:

$$\min_\gamma \{\Psi(x^n + \gamma d) | \gamma \geq 0\} \quad (21)$$

を解き、解を改訂する： $\mathbf{x}^{n+1} := \mathbf{x}^n + \gamma \mathbf{d}^n$   
 その後、 $n := n + 1$  として、Step.1 へ戻る。

(2) DUE 配分問題への応用

前節で示したアルゴリズムの効率性は、線形連立方程式 (18) を解くための計算処理に左右される。今回の DUE 配分では連立方程式 (18) は大規模であり、直接解こうとすると膨大な計算処理を要してしまう。しかし、近似集合  $\beta$  がある条件を満足する場合には、問題のグラフ論的構造を活用することにより、問題をより少ない計算回数で解くことができる。以下では、連立方程式 (18) をブロックごとに示す。その後、連立方程式を、余分な未知変数を消去することにより、未知変数  $\boldsymbol{\tau}_1(t), \boldsymbol{\rho}$  のみで表現する。

まず、DUE 配分 (17) において、関数  $F$  は線形関数であるから、 $n$  回目計算時の解  $\mathbf{x}^n$  が与件であるとき  $\nabla F(\mathbf{x}^n) = \mathbf{M}$  が成立する。このうち変数集合  $\beta$  に属する変数のみに着目すれば、式 (18) は次のように書き換えられる。

$$\mathbf{M}_{\beta\beta} \mathbf{d}_\beta^n = -\mathbf{F}_\beta(\mathbf{x}^n) + \mathbf{M}_{\beta\alpha} \mathbf{x}_\alpha^n \quad (22)$$

以降の議論では、使用する変数は変数集合  $\beta$  に属する変数のみである。そのため、表記を簡単にするために添え字の  $\beta$  を省略する。更に、連立方程式 (22) は、起点出発時刻  $t$  別に以下の方程式に分解できる。

$$\tilde{\boldsymbol{\tau}}_1(t) - \tilde{\boldsymbol{\rho}}_1 = -\mathbf{F}_1^q(t) \quad \forall t. \quad (23)$$

$$\tilde{\mathbf{w}}(t) + \mathbf{A}^T \tilde{\boldsymbol{\tau}}(t) = -\mathbf{F}^w(t) \quad \forall t. \quad (24)$$

$$-\tilde{\mathbf{y}}(t) + \mathbf{C} \Delta \tilde{\mathbf{w}}(t) + \mathbf{C} \mathbf{A}_+^T \Delta \tilde{\boldsymbol{\tau}}(t) = -\mathbf{F}^w(t) \quad \forall t. \quad (25)$$

$$\mathbf{A}_1 \tilde{\mathbf{y}}(t) + \tilde{\mathbf{q}}_1(t) = -\mathbf{F}_1^r(t) \quad \forall t. \quad (26)$$

$$\mathbf{A}_2 \tilde{\mathbf{y}}(t) = -\mathbf{F}_2^r(t) \quad \forall t. \quad (27)$$

$$\sum_t \tilde{\mathbf{q}}_1(t) dt = -\mathbf{F}_1^p \quad (28)$$

ここに、 $\tilde{\mathbf{q}}_1, \tilde{\mathbf{y}}, \tilde{\mathbf{w}}, \tilde{\boldsymbol{\tau}}, \tilde{\boldsymbol{\rho}}_1$  は  $\mathbf{d}^n$  の  $(\mathbf{q}_1, \mathbf{y}, \mathbf{w}, \boldsymbol{\tau}, \boldsymbol{\rho}_1)$  に対応する要素ベクトル、 $\mathbf{F}_1^q, \mathbf{F}^w, \mathbf{F}^r, \mathbf{F}_1^p$  は  $\mathbf{F}(\mathbf{x}^n)$  の  $(\mathbf{q}_1, \mathbf{y}, \mathbf{w}, \boldsymbol{\tau}, \boldsymbol{\rho}_1)$  に対応する要素ベクトルである。さらに、連立方程式 (17) は DUE 配分問題特有の性質により未知変数  $\tilde{\mathbf{q}}(t), \tilde{\mathbf{y}}(t), \tilde{\mathbf{w}}(t), \tilde{\boldsymbol{\tau}}_1(t), \tilde{\boldsymbol{\tau}}_2(t)$  を消去することで、より簡単な以下の方程式に帰着する<sup>1</sup>：

$$[\mathbf{A} \mathbf{C} \mathbf{A}_+^T]_1 \tilde{\boldsymbol{\rho}}_1 = \mathbf{G} \quad (29)$$

ここで、 $\mathbf{G}$  は  $\mathbf{F}(\mathbf{x})$  の要素の簡単な線形式から求められる時間依存しない定数ベクトルである。左辺  $[\mathbf{A} \mathbf{C} \mathbf{A}_+^T]_1$  は、行列  $\mathbf{A} \mathbf{C} \mathbf{A}_+^T$  の終点ノードに関する行・列のみから再構成された行列である。

改訂ベクトルの残りの要素に関しては、連立方程式 (29) の解  $\tilde{\boldsymbol{\rho}}_1$  を式 (23) - (28) に代入することにより、 $\tilde{\boldsymbol{\tau}}_1(t), \tilde{\boldsymbol{\tau}}_2(t), \tilde{\mathbf{w}}(t), \tilde{\mathbf{y}}(t), \tilde{\mathbf{q}}_1(t)$  の順に求めることがで

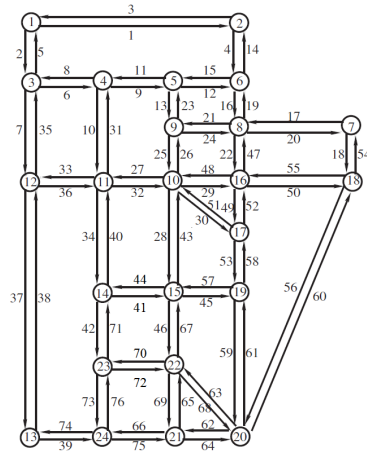


図-1 Sioux Falls ネットワーク

きる。

5. 数値実験

本章では、提案アルゴリズムと、近年 Han et al.<sup>12)</sup> により提案されたアルゴリズム (以下、Han アルゴリズムと呼ぶ) の比較実験の結果を示す。<sup>2</sup> Han アルゴリズムでは、リンクの渋滞モデルに Physical Queue(LWR モデル) を採用している。そのため、プログラム中の Forward Wave Speed と Backward Wave Speed パラメータを、それぞれ  $l_{ij}/c_{ij}$ , 0 とし、提案アルゴリズムと同じ Point Queue モデルとなるよう調整した。今回は、配分問題のテスト用ネットワークとして標準的に用いられている、Sioux Falls ネットワーク (図-1) に対する数値実験結果を示す。Sioux Falls ネットワークのノード数及びリンク数はそれぞれ 24, 76 である。今回の実験では、発生需要が多い上位 5 つのノードを代表的な起点ノードとして計算を行った。また、唯一起点を除く全てのノードを終点とした。唯一起点以外のノードが起点となっている OD 交通需要は全て 0 とした。

実験に使用した計算機は i7-3930K 型 CPU 搭載の PC (メモリ 32GB), OS は Windows 7 Pro, プログラム言語は MATLAB(R2018a) である。

数値実験は、実験 1, 2, 3 から成る。実験 1, 2 では、時間刻み、ネットワークの混雑度をそれぞれ変えて両アルゴリズムの収束までに要する反復回数を比較する。実験 3 では、両アルゴリズムの 1 反復当たりの平均 CPU 時間を計測する。最後に、これらの実験結果から、より大規模なネットワークに提案アルゴリズムを適用した場合の計算量について考察する。

<sup>1</sup> 厳密には、均衡解においてすべての相補性条件 (11) - (16) の右辺について、等式制約が満足している場合に、連立方程式 (17) を簡単に整理することができる。

<sup>2</sup> Han アルゴリズムは以下のサイトにてプログラムが公開されており、本研究でも公開プログラムを利用し実験を行った。 . . <https://github.com/DrKeHan/DTA>

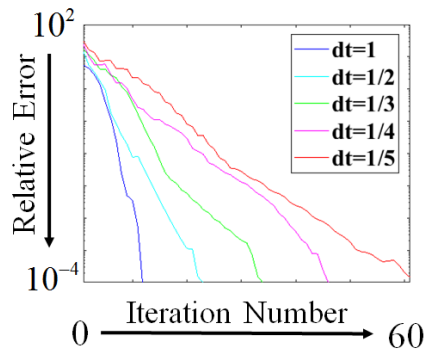


図-2 問題の規模の違いによる収束状況 (提案アルゴリズム)

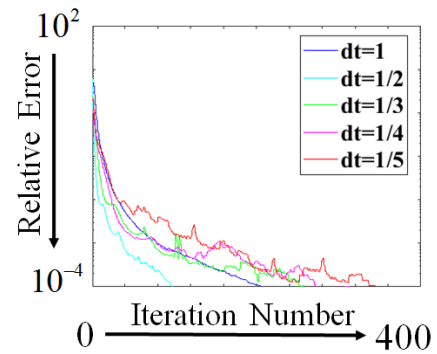


図-3 問題の規模の違いによる収束状況 (Han アルゴリズム)

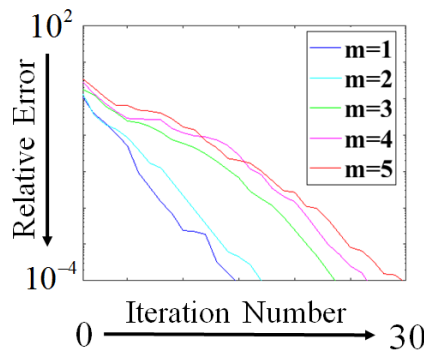


図-4 混雑度の違いによる収束状況 (提案アルゴリズム)

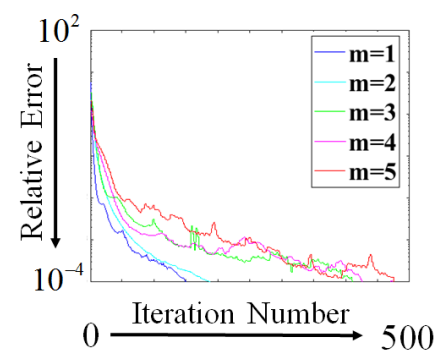


図-5 混雑度の違いによる収束状況 (Han アルゴリズム)

### (1) 実験 1: 問題の規模の違いによる収束状況の比較

本節では、問題の規模が反復回数に与える影響を、時間刻み  $dt$  を変化させて検証する。

図-2、図-3 は、DUE 配分の収束状況を、 $dt = 1, 1/2, 1/3, 1/4, 1/5$  と変化させた場合について示したグラフである。グラフの横軸は反復回数、縦軸は最早到着時刻の相対誤差  $|\tau^{n+1} - \tau^n|/|\tau^n|$  を示している。問題の規模が増加するに伴い、収束するまでに要する反復回数は増加している。そして、いずれの配分時間帯  $K$  においても、提案アルゴリズムは Han アルゴリズムの  $1/10$  程度の反復回数で収束していることが確認できる。

### (2) 実験 2: ネットワークの混雑度の違いによる収束状況の比較

静的な交通量配分では、ネットワークの混雑度が増すにつれて均衡解を得るために必要な反復回数が増加することが知られており、その関係は DUE 配分でも見られると予想される。そのため本節では、全ての OD ペアの OD 交通需要を、基準とする交通需要の  $m$  倍とし (i.e.  $\mathbf{Q} := m\mathbf{Q}, m = 1, 2, \dots$ ), 混雑度  $m$  が反復回数に与える影響について検証する。

図-4、図-5 は、DUE 配分の収束状況を、 $m = 1, \dots, 5$  と変化させた場合について示したグラフである。グラフの横軸は反復回数、縦軸は最早到着時刻の相対誤差  $|\tau^{n+1} - \tau^n|/|\tau^n|$  を示している。問題の規模が増加するに伴い、収束するまでに要する反復回数は増加している。尚、 $m$  を 5 よりも大きくした場合についても数値実験を行ったが、反復回数は急激に増加することはなかった。そして、いずれの混雑度  $m$  においても、提案アルゴリズムは Han アルゴリズムの  $1/10$  程度の反復回数で収束していることが確認できる。

### (3) 実験 3: 1 反復当たりの計算量の比較

本節では、両アルゴリズムの 1 反復当たりの計算量を比較することにより、提案アルゴリズムの効率性を検証する。提案アルゴリズムは線形連立方程式 (20) を解く処理、Han アルゴリズムは Network Loading に計算処理の大部分を要する。Han アルゴリズムの Network Loading において、加減乗除の演算回数は、経路数  $R$  と配分時間帯数  $K$  に比例する。対して、提案アルゴリズムの連立方程式の処理は、最悪でも  $O(K(N+L))$  に抑えられている。ここに、 $N$  はノード数、 $L$  はリンク

数である。一般に、ネットワークの規模が大きくなるに連れて、経路数は指数関数的に増大する。すなわち、提案アルゴリズムは、Han アルゴリズムに比べて少ない計算量で解を得られる。

このことをより具体的に示すために、今回の実験で最も計算処理を要した  $dt = 1/5, m = 5$  について、提案アルゴリズムと Han アルゴリズムの 1 反復当たりの平均 CPU 時間を計測することにより確認する。その結果はそれぞれ 1.47(s), 14.3(s) であり、提案アルゴリズムの 1 反復当たりの計算量は、Han アルゴリズムの 1/10 程度の計算量で済んでいる。

以上より、5.1 節、5.2 節の結果も踏まえると、提案アルゴリズムは Han アルゴリズムと比較して 1/100 程度の計算量で解を得られることが明らかとなった。大規模なネットワークでは、更に計算量の差は大きくなると考えられる。

## 6. おわりに

本研究では、経路・出発時刻同時選択 DUE 配分を解くための、効率的なアルゴリズムを提案した。具体的には、まず、問題を起点出発時刻ベースの座標系のもとで定式化し、簡明な数学的構造を持つ線形相補性問題 (LCP) として表現できることを示した。次に、この LCP に対して DUE 配分問題特有のグラフ論的構造を活かした大域収束的 Newton 法ベースのアルゴリズムを構築した。その後、数値実験により従来手法との比較を行い提案アルゴリズムの効率性を確認した。ここでは、比較的小規模の Sioux Falls ネットワークに対する数値実験で、計算量がおおよそ 1/100 程度に減少することを示した。さらに、提案アルゴリズムと従来手法の反復回数と 1 反復あたりの計算量の比較を通して、より大規模なネットワークにおいては、提案アルゴリズムの方が効率性が高まる可能性が大きいことを示した。

## 参考文献

- 1) Szeto W.Y., Lo, H.K., A cell-based simultaneous route and departure time choice model with elastic demand, *Transportation Research Part B*, Vol. 38, No. 7, pp. 593-612, 2004.
- 2) Jang, W., Ran, B., Choi, K., A discrete time dynamic flow model and a formulation and solution method for dynamic route choice, *Transportation Research Part B*, Vol. 39, No. 7 pp. 593-620, 2005.
- 3) Friesz, T.L., Kim, T., Kwon, C., Rigdon, M.A., Approximate network loading and dual-time-scale dynamic user equilibrium. *Transportation Research Part B* Vol. 45, No.1, pp.176-207, 2011.
- 4) Bliemer, M.C. and Bovy, P.H., Quasi-variational inequality formulation of the multiclass dynamic traffic assignment problem. *Transportation Research Part B*, Vol.37, No.6, pp.501-519, 2003.
- 5) Varia, H.R. and Dhingra, S.L., Dynamic user equilib-

- rium traffic assignment on congested multidestination network. *Journal of transportation engineering*, Vol. 130, No.2, pp.211-221, 2004.
- 6) Long, J., Huang, H.J., Gao, Z. and Szeto, W.Y., An intersection-movement-based dynamic user optimal route choice problem. *Operations Research*, Vol. 61, No.5, pp.1134-1147, 2013.
- 7) Akamatsu, T., An efficient algorithm for dynamic traffic equilibrium assignment with queues, *Transportation Science*, vol. 35, 4, pp. 389-404, 2001.
- 8) Kuwahara, M., Akamatsu, T., Dynamic equilibrium assignment with queues for a one-to-many OD pattern, *Transportation and Traffic Theory*, vol. 12, pp. 185-204, Elsevier, 1993.
- 9) Akamatsu, T., Wada, K., Hayashi, S., The corridor problem with discrete multiple bottlenecks, *Transportation Research Part B*, vol.81, pp.808-829, Elsevier, 2015.
- 10) Facchinei, F., Soares, J., Testing a new class of algorithms for nonlinear complementarity problems. A. Maugeri, F. Giannessi, eds. *Variational Inequalities and Network Equilibrium Problems*, Plenum Press, 1995.
- 11) Fisher, A., A special Newton-type optimization Method, *Optimization*, No.24, pp. 269-284, 1992.
- 12) Han, K., Eve, G., Friesz, T.L., Computing dynamic user equilibria on large-scale networks, *Networks and Spatial Economics*, Vol. 18, pp. 1-34, 2019.

(2019. 3. 10 受付)