

需要分布に基づくロードプライシングにおける 計算量削減手法の検討

山本 規吉^{1,a)} 川上 朋也¹

概要：交通渋滞を低減させるために様々な種類のロードプライシングが提案され、各国で利用されている。ロードプライシングには現在の道路の状態に対応して課金額をリアルタイムに設定する方式もあり、道路資源の有効活用が期待されている。そのロードプライシングの中でも、事業者が需要分布に基づいて利用料金の決定と提示を事前に行う方式について、本論文では検討を行う。検討の基となる従来方式は料金決定の対象とする道路に分岐が多い場合、助長な計算によって計算量および計算時間が増加してしまう。本論文ではこの需要分布に基づく料金決定において、計算量を削減する手法を検討する。検討手法では料金決定の前処理として疎行列圧縮を用いることで、実際に通ることのない経路の計算を削減する。検討手法のシミュレーションにより、需要分布に基づく料金決定において計算量が削減されることを確認した。

1. はじめに

現在、渋滞による時間とエネルギーの浪費は世界中で多くの社会損失を生んでいる。日本だけで考えても、渋滞による損失額は11.6兆円であり、一人当たりで換算しても年間9万円の損失を生んでいることになる [1]。今後、新興国での自動車数増大により、さらなる悪化が推測される。

その渋滞問題を解決するために、ロードプライシングという手法が世界各国で研究・導入されている [2-7]。ロードプライシングは、特定の道路や地域、時間帯における自動車利用者に対して課金をすることにより、自動車利用の合理化や交通行動の転換を促し、自動車交通量の抑制を図る政策のことである。ロードプライシングには課金の対象や価格決定の方式によってさまざまな種類があり、実際の混雑状態に応じて価格を設定することで、道路の利用効率を高めることが期待できる。

本論文ではそのロードプライシングの中でも、出発地、到着地、時間帯ごとの道路利用の需要分布を事前に把握できる環境を想定し、道路事業者がその需要分布に基づいて価格の決定と提示を行うという手法 [8] について検討を行う。この手法では需要分布に基づく道路利用モデルを設計し、価格決定を最適化問題として差分進化法を用いて計算

を行う。実際に文献 [8] の中で適切な価格が計算により設定され、道路資源を有効に活用できることは確認されている。今回はその手法に対して実際の道路にも適用できるように、通行料金決定アルゴリズムに変更を加え計算量を削減することを目的とする。先行研究では道路のモデルとして比較的短いものを利用していたが、実際の道路の適用を行うためにはさらに多くの道路数を持つモデルについての計算が必要になる。道路区間の増加により、料金決定までの計算量・計算時間が増加するため、計算量削減による料金決定の時間短縮が実際の道路での適用に必要なことになる。

よって、本研究の目的としては先行研究である需要分布に基づくロードプライシング手法を実際の道路に適用することを目的とし、その手段として料金決定手法の計算量削減により実際の道路での計算時間の増加を抑える。

2. 関連研究

川上らは予想される混雑状況に応じて通行料金を設定し、利用者へ事前に提示する仕組みとして、需要分布に基づくロードプライシングを提案している [8]。

本研究ではこの提案されているロードプライシング法に対して検討を行うため、本章では先行研究である [8] から関係する内容をまとめて述べることにする。

¹ 福井大学
University of Fukui
^{a)} hb171226@u-fukui.ac.jp

2.1 想定環境

この研究では、道路事業者が事前に出発地・目的地・時間帯ごとの利用需要の分布を大まかに把握することができている環境を想定する [9]。ここでの利用需要とは道路の利用料金に対する利用者数のことを指す。また、各道路区間には谷時間あたりに走行可能な車両台数はあらかじめ定められているとする。さらに、各車両は流体モデルに従って移動するものとする [10, 11]。

この研究では各利用需要に対する利用料金の算出・出力をし、利用者へ事前に提示するシステムを想定する。出力される料金は全利用時間帯において各道路区間の可能交通容量を超過しないという制約のもと、道路事業者の総収入を最大化する。

2.2 定式化

本研究の想定環境として、隔離用需要に対する最適な利用料金を求める問題を最適化問題として表現するために、入力・出力・制約条件・目的関数に分けて定式化を行う。

2.2.1 入力

本研究における入力は、道路、利用者（車両）、需要分布の3つである。それぞれを以下に述べる。

- 道路

道路網を有効グラフ $G = (V, E)$ で表し、道路網に存在する出入口（ゲート）を $v_i \in V$ 、頂点間の道路区間（セグメント）を $e_i \in E$ とする。また、セグメント e_i を利用可能な最大数（交通容量）を c_i 、時刻 $t \in T$ における e_{in} の利用数（交通量）を e_{it} とする。ここで、 T は利用可能なすべての時間（タイムスロット）を表す。

- 利用者

各利用者は、出発ゲート $v_s \in V$ 、到着ゲート $v_g \in V$ 、出発時刻 $t_s \in T$ をもつ。

- 需要分布

出発ゲート v_i 、到着ゲート v_j によって一意に決める最短経路を $R(i, j)$ とする。また、出発時刻 t で経路 $R(i, j)$ を利用したい利用者数（需要）を N_{ijt} 台、価格 x での需要の割合を表す需要分布を確率密度関数 $f_{ijt}(x)$ とする。 $f_{ijt}(x)$ の累積分布関数を $F_{ijt}(x)$ とすると、出発時刻 t で経路を $R(i, j)$ 、価格 p_{ijt} の場合に予測される総利用者数は $(1 - F_{ijt}(p_{ijt}))N_{ijt}$ 台となる。

2.2.2 出力

本研究で出力として、利用可能なすべての経路 $R(i, j)$ 、出発時刻 t に対して価格 p_{ijt} を決定する。利用者は入力として前述の出発ゲートと到着ゲート、出発時刻を事前に指定し、出力として得られた価格に基づいて、道路を利用するかどうかを判断する。

2.2.3 制約条件

本研究では、道路のすべてのセグメントで交通容量を超えないことを想定する。そのため、制約条件として、以下のすべての時刻 t とセグメント e_i で満たす必要がある。

$$n_{it} \leq c_i \quad \forall e_i \quad \forall t \in T \quad (1)$$

2.2.4 目的関数

出発時刻 t で経路 $R(i, j)$ を利用するすべての利用者から得られる収入は、 $(1 - F_{ijt}(p_{ijt}))N_{ijt}$ となる。本研究では道路事業者の利益を最大化するため、すべての経路と出発時刻での利用者からの収入を最大化する以下を目的関数とする。

$$\max \sum_i \sum_j \sum_{t \in T} (1 - F_{ijt}(p_{ijt}))N_{ijt} \quad (2)$$

2.3 通行料金決定の流れ

今回述べた問題設定では、需要分布は経路 $R(i, j)$ と出発時刻 t ごとに必要となる。そのため最大で、各道路が片方向の場合に $V C_T$ 、双方向の場合に $2V C_T$ となる。ここで、 V はゲート数を、 T はタイムスロット数を表す。例えば、双方向で各経路の価格 p_{ijt} を $150 \leq p_{ijt} \leq 450$ の範囲かつ1単位で決定する場合、組み合わせは最大で 300 の $2V C_T$ 乗通りとなる。

本研究では、問題設定における組み合わせ最適化を行う。提案手法の流れを以下と図1に示す。

- (1) 経路と出発時刻ごとの需要分布にたいして、価格をそれぞれ仮に設定
- (2) 関係する利用需要の累積分布から、各自国の利用者数をセグメントごとに予測し、すべて交通容量を満たすかどうかを判別
- (3) 仮に設定した価格で、利用者からの総収入を計算
- (4) 各価格を変更して繰り返し、総収入が最大となる価格を求める

2.4 差分進化法

本研究では最適化手法として差分進化法をアルゴリズム内に用いている。差分進化法は Storn らによって提案された進化的アルゴリズムである [12, 13]。差分進化法は探索空間内にランダムに生成した個体から集団を構成し、各個体をほかの個体と交差させ突然変異個体を生成する。生成した突然変異個体の評価値が元の個体の評価値より優れている場合にそれらを置換することで、最適解を求める。

差分進化法は突然変異個体の生成に用いる式の違によってさまざまな形式がある。本論文では大域探索性とよく書探索性を併せ持つとされる DE/rand-to-best/1 を用いる。

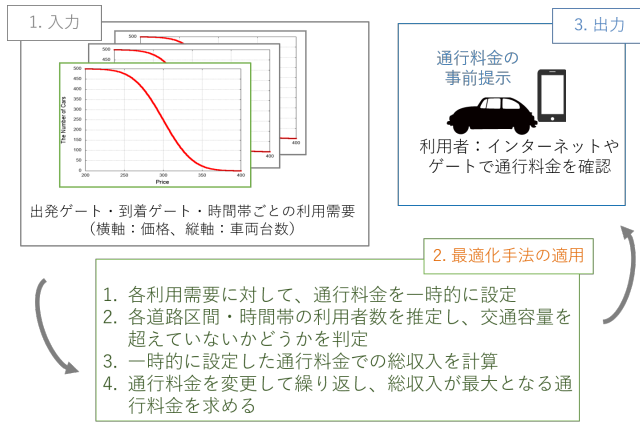


図 1 料金決定手法の流れ

以下に差分進化法における処理の流れを示す。

手順 1 世代 $G=0$ として初期個体を N_{pop} 個生成し、初期集団を構成する。構成された集団内の個体をすべて評価する。

手順 2 制約条件を満たす場合は終了する。主に世代数や関数評価回数が制約条件として用いられる。

手順 3 各個体 $x_i^{(G)}$ について以下の処理を行う。

3.1 集団よりランダムに 3 つの個体 $x_{r_1}^{(G)}, x_{r_2}^{(G)}, x_{r_3}^{(G)}$ を選択する。

3.2 以下の式より突然変異個体を生成する。

$$v_i^{(G)} = x_{r_1}^{(G)} + F \cdot (x_{best}^{(G)} - x_{r_1}^{(G)}) + F' \cdot (x_{r_2}^{(G)} - x_{r_3}^{(G)}) \quad (3)$$

3.3 $x_i^{(G)}$ と $v_i^{(G)}$ を交叉し、 $u_i^{(G)}$ を生成する。

3.4 $x_i^{(G)}$ と $u_i^{(G)}$ を比較し、評価値の優れているほうを次世代の個体 $x_i^{(G+1)}$ とする。

手順 4 手順 2 に戻る

ここで F, F' は突然変異率と呼ばれるものであり、 $0 < F, F' \leq 1$ とする

3. 計算量削減手法の検討

3.1 概要

本研究では上記で述べた需要分布 a に基づくロードプライシングに対して、実際の道路への拡張の取り組みとして計算量削減手法を提案する。ロードプライシング手法の目的は道路課金を行うことで道路資源を有効に活用するというものであり、ロードプライシング研究は最終的に実際の道路に適用し、検証を行うことでその効果を確認する必要がある。実際の道路にロードプライシング手法を活用することを考えた場合、料金決定手法によって得た最適化した料金の組み合わせを現実的な計算時間で求める必要がある。

先行研究の文献 [8] では、図 2 のようにモデルとして合

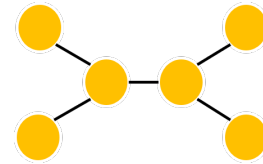


図 2 先行研究の道路モデル

流、分岐を考えたノード数 6 の道路を想定として、評価を行っている。2 章の想定環境に述べた通り、出入口を一つのゲートとして道路モデルを作成しているが、実際の道路と比べ、短縮した道路モデルであるといえる。例えば、日本の首都である首都高速道路で考えてみる [14] と、首都高速道路内の高速中央環状線だけでも 19 個の出入口があり、すべての首都高速道路にロードプライシング手法を適用するとすると先行研究の想定環境のノード数よりもかなり多くのノード数を持つ道路環境で解を求める必要があることが考えられる。道路のノード数の増加によって得られる道路区間の数は各ノードの組み合わせの数となるため、道路区間は道路数増加により、莫大に増えていくことが予想される。

提案している需要分布に基づくロードプライシング手法では各経路すべてに対して料金の選択を行い、料金の組み合わせを結果として出すものであるが、道路区間の大幅な増加によって計算を行う量が増え、最適な料金を計算する料金決定アルゴリズムの料金の出力に時間がかかり、リアルタイムでの料金の組み合わせの選択が難しくなることが想定される。

今回、料金決定アルゴリズムの改良から計算量削減を行うことで実際の道路への適用という一つの目的の達成に近づくと考えられる。

3.2 既存アルゴリズムの計算

料金決定アルゴリズムの改良という点からの計算量削減を試みるが、まず今節では既存あるアルゴリズムの計算の仕方をまとめることにする。まとめた結果から改善の余地を探す。

3 章の関連研究で述べた通り、現在利用している交通料金を決定するアルゴリズムで各経路、各時間帯から得られる道路課金の総収入の値を目的関数とし、差分進化法を用いて計算を行っている。

総収入を計算するために料金決定アルゴリズム内に各経路、各時間帯ごとの料金を格納する料金の配列を用意しており、その配列の各要素の合計値を制約条件である経路の交通容量を超えていないかを確認しながら料金の選択を行っている。

この料金の配列であるが、配列の長さはすべての経路区間の料金を調べるために (タイムスロット) × (ゲート数) × (ゲート数) 分用意している。この場合、すべての経路を

網羅することはできるが実際に道路がつながっていない、存在しない経路も配列に含むことになる。本研究では、この存在しない経路を含む配列の配列数を削減することで計算時間を削減できないかを検討することにする。

3.3 配列の要素数削減による計算量削減の検討

この節では配列の削減により同程度計算量の削減を行えるかを2章3節の料金決定アルゴリズムの計算の流れに沿いながら検討を行う。

計算の大きな流れとして、各区間、各時間帯の料金を設定する。その料金での交通容量を利用需要を用いて計算、その料金での各経路の収入、そしてその収入の合計値を評価値としてその総収入が最大になる値を差分進化法により何度も繰り返して最適な組み合わせを探すというものになっている。各経路の道路容量と収入の値はそれぞれ配列をアルゴリズム内に用意している。この2つの配列は各経路の料金の配列と同様の長さを持たせているため、料金の配列の削減は交通容量の配列と収入の配列の削減も同様に行っていることになる。

配列の削減によってまず初めに想定できる計算量削減は料金からの交通容量計算、その交通容量と料金からの経路の収入計算は計算する道路区間の数が減るため、計算量の削減を行えることが想定される。

また、制約条件として渋滞が起こらない範囲で料金設定を行うというものがあるので、交通容量の配列の各要素に対して道路モデルの設定として与えている最大交通容量を超えていないかを差分進化法の各世代で判定を行っている。前述した交通容量と収入の値の計算に関して、実際に通らない経路の計算は交通容量、収入の値のどちらとも差分進化法の世代数で変化しない0の値となる。計算に変化がないため大きな計算量削減にはつながらない可能性があるが、この制約条件の判定処理はアルゴリズム計算内で交通容量の配列の各要素を差分進化法の世代の数行う処理となっているため、配列の削減により要素数が減り、計算量の削減に貢献できるのではないかと考える。

3.4 疎行列を用いた配列の削減

今回は検討手法として疎行列を用いて経路を確保することで料金の配列の削減を試みた。

3.4.1 疎行列

まずは疎行列が何かについて説明する。

疎行列とは配列の中に0を多く含む行列のことである[15,16]。その0を多く含むという特徴から0を無視する工夫を行うことで高速に行列計算を行うことが可能になる。例えば、配列の9割の要素がゼロである大規模配列について考えると、配列内の非ゼロ要素のみの計算を行うと計算時間を10分の1まで削減できることができるとともに、行列の格納を行う際に非ゼロ要素のみを格納すると配

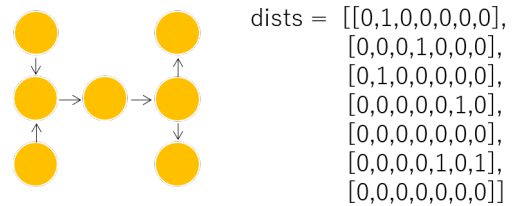


図3 道路モデルの行列表現

列の長さも10分の1まで短くすることが可能である。

しかし、行列の要素に0をほとんど含まない密行列に疎行列の計算手法を適用すると余計な計算時間がかかってしまい、単純に行列を計算するより時間がかかってしまう可能性も存在する。

3.4.2 疎行列を用いた配列の削減

本研究環境では疎行列計算を用いることによって料金決定手法のアルゴリズムの計算時間を削減できるかについて検討を行う。

本研究では道路を有向グラフとしてとらえて計算を行うことは2章にてすでに述べた。図3に示した通り、H型の片側走行のノード数7の道路モデルは図の行列のように表現することができる。この行列から実際に車が走行できる経路のみを抽出し、料金の配列を作成することで、配列の長さの削減が期待でき、そのことにより料金決定手法アルゴリズムの計算時間削減も期待をすることができるのではないかと考える。

しかし、疎行列計算は行列の要素にゼロ以外の要素を多く持つ計算に関しては元の行列のまま計算を行う場合より時間がかかってしまう可能性が存在する。その場合目的である計算量削減は達成できないだけでなく逆に計算量を増やしてしまう可能性も存在する。

3.5 本研究で想定する道路モデル

前節で述べた通り、想定する道路モデルによっては、配列の要素数削減により計算量削減は行えるが、疎行列計算による計算量増加によってアルゴリズム全体の計算量が増加してしまうケースが存在する可能性がある。

本研究では利用需要として各時間帯ごとの利用需要を入力として与えており、同じ時間帯に関してはすべての道路に関して同じ利用需要を用いて計算を行っている。実際の道路では通勤時間帯では都市部へ向かう道路が混み、帰宅時間帯では帰宅する都市部外の道路が混むなど同じ時間帯、同じ道路区間であっても向きによって利用需要が大きく変化する可能性が想定される。つまり同じ道路区間であっても別の利用需要を用いる必要があるため、道路の向きによって別の計算量アルゴリズムを検証する必要がある実際の道路への実装には求められる。そのため、本研究では想定する道路モデルとして片側走行のモデルのみを想定として考えることとする。片側走行の道路モデルでは道路を有向グ

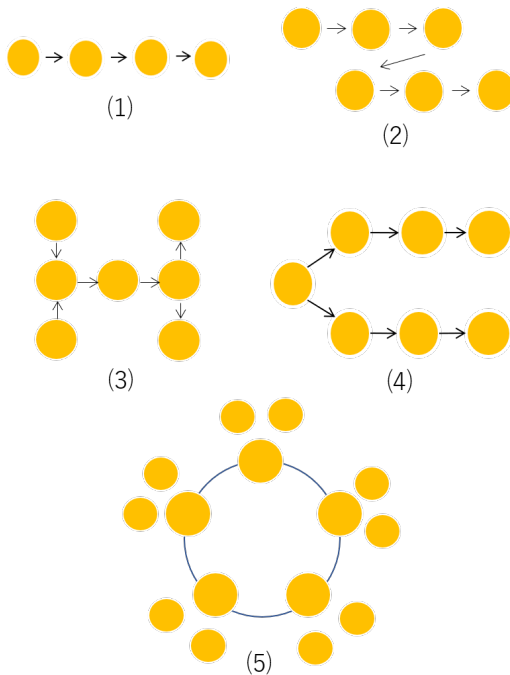


図 4 配列の要素数削減検討を行う道路モデル

表 1 配列の要素数削減結果

	(1)	(2)	(3)	(4)	(5)
配列削減前	64	144	196	196	900
配列削減後	24	60	76	48	380

ラフとして表現した際に、両側走行のモデルと比べてゼロ要素が多くなる。そのため疎行列を用いた処理の有効性がより示されることが予測できる。

3.6 道路モデルを用いた配列の削減の評価

本節では実際に道路モデルをいくつか作成し、先行研究の手法と比べ、疎行列計算を用いて配列の削減を行った場合どの程度配列を削減できるかを確認する。すべての道路モデルで時間帯を 4 と置き、配列数の確認を行う。先行研究の配列数は前述したとおり、(時間帯) × (ゲート数) × (ゲート数) であり、提案手法の配列数は (時間帯) × (実際に通る経路数) となる。実際に通る経路数は道路モデルの有効グラフに最短経路問題を適用し、得られた経路の数となる。道路モデルは図 4 に示した通りの 5 つのパターンを用意した。道路はすべて片側走行の道路を想定しており、時間帯数についてもすべて 4 を想定している。(1) はノード数 4 の直線型、(2) はノード数 4 の直線型、(3) は合流分岐を考えたノード数 7 の H 型、(4) は分岐のみを考えた V 型、(5) はノード数 15 のラウンドアバウト型である。ラウンドアバウト型はパリの凱旋門など欧州に多く見られる円形交差点であり、実際に日本でも高速道路の IC に利用されているケースがある [17] ため、モデルパターンの一つとして用意した。配列の要素数削減結果は表 1 のとおりである。

表 1 の結果からすべての道路モデルに関して、配列の要素数の削減を行えていることを確認できた。同じ直線型の (1), (2) を比較すると (1) では 64 から 24 と 62.5 % の削減, (2) では 144 から 60 と 58.3 % の削減を確認することができた。直線型ではゲート数が短いほうがより配列の要素数を削減できることがわかる。また, (3) の H 型では 196 から 76 へ 61.3 % の削減を確認することができた。直線型と比べ分岐, 合流がある道路モデルでは、実際に繋がっていない経路が増えるため、ノード数が (2) の 6 より多い 7 ゲートであるにも関わらず, (3) では (2) より多くの割合で配列の要素数削減が確認できた。この考えより, (4) は分岐のみを考える道路であるが、繋がっていないゲートの組み合わせが (3) と比べても多い。そのため 196 から 48 へ 76.5 % という (3) と比べても高い削減効果を確認することができる。また, (5) のような特殊な形の道路モデルであっても 900 から 380 へ 57.8 % の削減を確認でき、どのような道路パターンにおいても配列の要素数の削減は行うことができるといえる。

4. 実験と評価

前章でいくつかの道路モデルパターンにおいて料金の配列の効率化を行うことができたことは確認した。しかし, 3 章 3 節で述べた通り, 疎行列でない場合に粗行列計算を用いると余計に計算時間がかかってしまうケースが確認される。この章では実際に提案手法のアルゴリズムの計算時間を確認し, 疎行列を用いた計算, 配列数の削減により計算時間がどの程度変化するかを確認する。

今回求める計算時間は同じモデルで賞金決定アルゴリズムを使って料金の組み合わせを求めたとしても, 差分進化法の最適化のされ方によって変化する。そのため今回は同じモデルに 3 度計算を行い, 平均の値を実験結果の値とした。

4.1 実験環境

道路モデルは 3 種類用意し, すべての実験は同マシンの同環境で行った。今回実験を行った機器のスペックは表 2 のとおりである。

料金決定アルゴリズムで用いられる差分進化法について, 世代の最大数は 10000 とし, 条件として, 20 世代最適化の世代をも回しても評価値であるすべての区間の経路の総収入の値が変わらない場合最適化が完了できたとして料金決定を終了することとする。

道路モデルを図 5 にて示す。パターン 1 はゲート数 4 の直線型, パターン 2 はゲート数の直線型, パターン 3 はゲート数 7 の H 型を用いている。料金決定アルゴリズムとして先行研究で用いられていた, 配列を削減せずに計算を行う比較手法と, 配列の要素数を疎行列計算によって削減した検討手法の 2 つを用意し, 各道路モデルでの料金決定

表 2 計算を行った機器

CPU	Intel Celeron 3965Y (2 コア, 1.50GHz)
メモリ	4.00GB
ストレージ	SSD
OS	Windows 10 Home
使用言語	Python 3.7

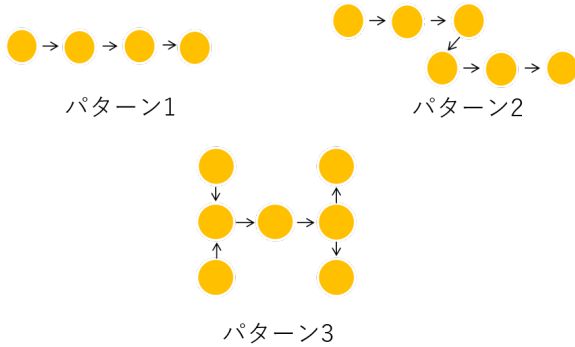


図 5 想定する道路モデル

表 3 想定する道路モデルの配列の要素数変化

	パターン 1	パターン 2	パターン 3
配列削減前	16	36	49
配列削減後	6	15	19

アルゴリズムの計算時間を求める。今回は時間帯をすべて 1 に固定して計算を行った。表 3 で示す通り、各モデルの配列の要素数はパターン 1 では比較手法の 16 個から提案手法の 6 個へ 10 個の削減、パターン 2 では 36 個から 15 個へ 21 個の削減、パターン 3 では 49 個から 19 個へ 34 個の配列数削減を行うことができています。

4.2 実験結果と評価

実験の結果を図 6 に示す。

パターン 1 では配列の要素数を削減する提案手法で 32.94[s]、削減をしない比較手法では 71.86[s] という結果になった。続いてパターン 2 では提案手法で 963.24[s]、比較手法で 1973.4[s] という結果となり、パターン 3 では提案手法で 4823.24[s]、比較手法で 7768.01[s] の結果になった。この結果より、想定したすべての道路モデルにおいて計算量削減が行うことができていますことを確認することができます。

5. おわりに

5.1 まとめ

交通渋滞による社会損失を削減し、道路資源を有効に活用するために考えられたロードプライシング手法に対して、先行研究から引き続き、利用需要が事前に得られる環境を想定して需要分布に基づくロードプライシングについて検討を行った。先行研究によって想定環境内で道路資源を有効活用し、交通渋滞を防ぐことができたことは確認で

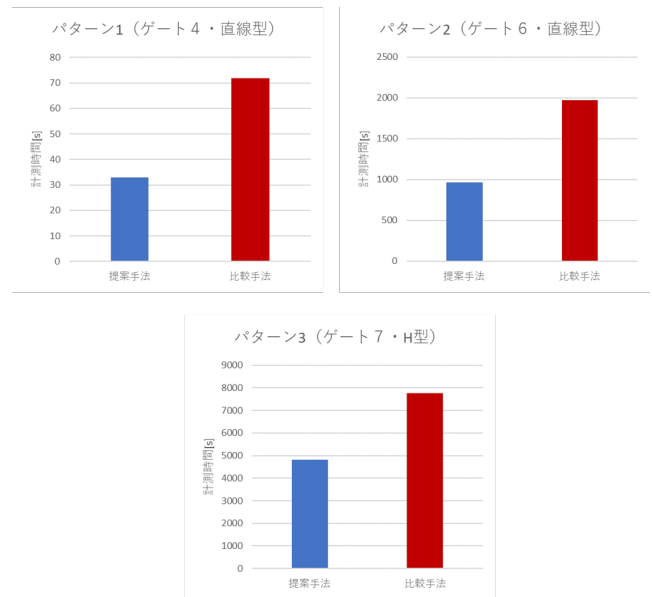


図 6 実験結果

きた。しかし、想定環境では実際の道路と比べて短い区間での計算を行っており、実際の道路への適用にはまだ至ってはいなかった。本研究ではその実際の道路への適用の一つの足掛かりとして料金決定アルゴリズムの計算量削減を検討した。リアルタイムで各経路ごとの通行料金の組み合わせを提供するためにはある程度アルゴリズムの計算速度が必要となり、その料金の組み合わせを求める料金決定アルゴリズムの計算量とともに計算時間の削減によって実際の道路への適用を行うことを目的とした。

料金決定アルゴリズムでは用いる最適化手法の一つである、差分進化法の目的関数として与えている各時間帯、各経路の総収入を計算するために、料金の配列を用意している。また、その料金の配列と事前に想定している需要分布を用いることで各時間帯、各経路の交通量を求め、その交通量を格納する配列も用意している。この交通量の配列を用いて、すべての時間帯、経路に対して交通量を超えないようにするという制約条件の確認を行っている。今回は疎行列を利用してこの料金の配列と交通量の配列の要素数を削減することで計算量削減を行った。

疎行列計算を用いることにより、配列内に存在していた実際には通らない経路の要素を削減し、配列の要素数を想定した様々な道路モデルのすべてにおいて配列の要素数の削減が行えていることを確認した。

また実際に配列の要素数削減により、計算量の削減を行ったかを配列の要素を削除する場合としない場合を、評価実験で確認した。実際に想定した道路モデルのすべてにおいて計算量の削減を確認することができ、疎行列を用いて配列の要素数を削減することは計算時間削減という料金決定アルゴリズムの効率化になったと確認できる。

5.2 今後の課題と展望

本研究の目的として、実際の道路への適用のために道路を拡張しても計算量が大幅に増加しないよう計算量の削減を行ったが、本研究の中では料金決定アルゴリズムの計算量削減を行うことができたが、実際に道路モデルを拡張して、実際の道路と同じような環境での検証を行うには至らなかった。今後の課題としては実際に、例えば首都高速道路と似た環境での検討などがあげられる。また、その規模の道路に拡張した際にも疎行列を用いた料金決定アルゴリズムの計算量削減が効果を示しているかの検証も行うべきだと考える。

また、本研究では事前に得られる需要分布として正規分布を与えているため、実際の道路に即した需要分布を用いて検証することも今後の課題の一つとして挙げられる。その際、変動的な料金設定により今までの固定料金の需要分布から分布が変化する可能性もあのため、その検討を行うことも必要になると考えられる。

本研究では計算量削減により料金決定アルゴリズムの効率化を行うことによって、先行研究としてある需要分布に基づくロードプライシング手法 [8] の実装により近づけることができた。今後の展望として、今後研究を進めていくことで、実際に渋滞が良く起こる道路区間でのリアルタイムでの料金の提示を行うロードプライシング手法を適用することにより、渋滞を減らし、渋滞による社会損失を防ぐことまでに研究成果を進めていくことを考えている。

謝辞 本研究の一部は福井大学研究育成経費の助成による成果である。

参考文献

- [1] 国土交通省道路局：渋滞の現状と施策体系。
<http://www.mlit.go.jp/road/sisaku/tdm/Top03-01-01.html> (参照 2020-10-13).
- [2] 王 曉虎：シンガポールの都市形成とプライシング政策、大阪産業大学経営論集, Vol. 16, No. 1, pp. 55–82 (2014).
- [3] 中村実男：ロンドンにおける道路混雑問題と混雑対策の展開：ロード・プライシングの導入を中心に、明大商学論叢, Vol. 100, No. 4, pp. 17–28 (2018).
- [4] 今西芳一, 内山直浩, 大瀧逸朗, 中梯 諭, 根本敏則：料金体系変更による社会的余剰への影響：首都高の距離別料金導入をケーススタディとして、計画行政, Vol. 39, No. 2, pp. 49–55 (2016).
- [5] 高木良太, 円山琢也, 溝上章志：エリア課金の最適設計問題：課金領域・レベルの決定アルゴリズムの構築と適用, 土木学会論文集 D3, Vol. 67, No. 5, pp. I.1233–I.1242 (2011).
- [6] 円山琢也：領域形状に着目した次善混雑課金の政策分析, 応用地域学研究, Vol. 20, pp. 13–22 (2016).
- [7] 井ノ口弘昭, 秋山孝正：群知能技術を用いた都市高速道路の料金設定方法の提案, 交通工学論文集, Vol. 5, No. 4, pp. A.18–A.23 (2019).
- [8] 川上朋也, 増田健一, 柴田直樹, 伊藤 実：ロードプライシングにおける需要分布に基づく料金決定手法の提案, マルチメディア, 分散, 協調とモバイル (DICOMO2017) シンポジウム論文集, pp. 934–940 (2017).

- [9] 土木学会：道路交通需要予測の理論と適用 第 I 編利用者均衡配分の適用に向けて, 土木学会 (2003).
- [10] 井上博司：連続流体モデルによる混雑したネットワーク交通流の動的シミュレーション手法, 土木学会論文集, Vol. 1997, No. 569, pp. 85–94 (1997).
- [11] 久井 守：交通流のモデリングと信号制御, 計測と制御, Vol. 41, No. 3, pp. 193–198 (2002).
- [12] Storn, R. and Price, K.: Minimizing the Real Functions of the ICEC'96 Contest by Differential Evolution, *Proceedings of the IEEE international Conference on Evolutionary Computation*, pp. 842–844 (1996).
- [13] Storn, R. and Price, K.: Differential Evolution—A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces, *Journal of Global Optimization*, Vol. 11, No. 4, pp. 341–359 (1997).
- [14] 首都高速道路株式会社：首都高道路交通状況マップ.
<http://search.shutoko-eng.jp/rmap.html> (参照 2021-02-18).
- [15] 岡田真幸, 櫻井鉄也, 寺西慶太：近似係数行列に対する疎行列用直接解法を用いた前処理, 日本応用数理学会論文誌, Vol. 17, No. 3, pp. 319–329 (2007).
- [16] 笹岡 岳, 仲渡文成, 鈴木 智, 河村 隆：疎行列圧縮格納方式を用いたモデル予測制御の演算効率化に関する研究, 第 61 回自動制御連合講演会講演論文集, 11F1, 4 pages (2018).
- [17] 国土交通省：第 1 回ラウンドアバウト検討委員会, ラウンドアバウトの現状. <http://www.mlit.go.jp/road/ir/ir-council/roundabout/pdf01/4.pdf> (参照 2021-02-18).