

# 経路選択仮定に依存しない 特定コミュニティへの流入交通捕捉リンクデザイン問題

Inbound Traffic Capturing Link Design Problem for a Specific Community Without Route Selection Assumptions

北海道大学工学部環境社会工学科 ○学生員 佐瀬瑠璃 (Ruri Sase)  
北海道大学大学院工学研究院 正員 杉浦聡志 (Satoshi Sugiura)

## 1. はじめに

主要な交通問題として、交通混雑や車両の排気ガスによる大気汚染が挙げられる。このような交通問題に対処する交通施策として、交通流の観測や制御が広く実施されている。交通流の観測はその実態把握を目的として行われ、また、交通流の制御は、都心部の混雑緩和を目的とした通行料課金による交通量抑制が代表的である。これらの交通施策は、道路ネットワーク上の全てのリンクで実施可能であればよいが、それはコストの面から非現実的であり、行政境界や環状道路といった地理的に容易に区域を設定できる地点で実施されることが多い<sup>1)</sup>。しかしながら、このような決定方法は観測・制御において最適な配置であるとは限らず、最小コストで所定の目的を達成する最適な実施位置検討の重要性は明らかである。

以上の社会的要請を背景に、道路ネットワーク上で交通流の観測や制御を効率的に行う位置を検討する研究が積極的に展開されてきた。交通流の観測位置に関する研究としてはトラフィックセンサー配置問題が挙げられ、例えば Yang et al.<sup>2)</sup> は、OD フローの効率的な観測を可能にするスクリーンラインの位置決定問題を提案した。また、交通流の制御位置は、例えばコードプライシングの最適な実施位置を決定する研究として、Zhang and Yang<sup>3)</sup> が料金水準と課金位置の同時最適化モデルを提案した。

ある特定の都心部や生活圏といった、道路ネットワーク中の特定領域（以下、コミュニティと呼ぶ。）に外部から流入する交通を効率的に捕捉する方策は、都心部のコードプライシングや疫病の封鎖施策など、応用の幅が大きいと考えられる。しかしながら、これまでの交通流の観測位置決定問題や制御位置決定問題は、対象のネットワーク全体の交通流を観測、制御の目標とするものが多く、指定した目的地へ流入する交通流を捕捉対象とするものは見当たらない。なお、ここでの交通流の捕捉とは、交通流の観測施策、制御施策いずれも実施可能性がある状況を指す。

Mazur et al.<sup>4)</sup> は、交通流の観測問題と制御問題は、施策の実施前後における需要変動を陽に考慮する必要性の有無に主な違いがあると指摘している。そこで、本研究では、交通流の観測位置問題、制御位置問題のいずれに対しても適用可能性を担保した、需要変動に左右されない流入交通捕捉位置の決定を目的とし、指定したコミュニティ外地点を経由して対象コミュニティに流入する全経路の交通流を捕捉するリンクを設定する。そうすることで、コミュニティ外地点を経由しコミュニティ内の地点を目的地とする交通流の迂回行動による需要変動を考慮する必要性を排除

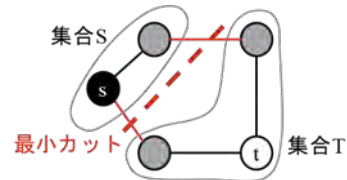


図-1 最小カットの図解

できる。

以下に本問題の特徴を挙げる。

1. 道路ネットワークの接続関係に基づくアプローチから問題を構築することで、配分交通量等を問題の入力情報に使用せず、利用者の経路選択に関する仮定に依存しない解を出力可能である。
2. 流入交通捕捉リンクをコミュニティ外に設置することで、流入交通の捕捉対象とするコミュニティのサイズに影響を及ぼさない解の出力が可能である。

## 2. 提案手法

### 2.1 概説

以下に提案手法の概説を記す。まず、本手法におけるコミュニティとは、「捕捉対象とする流入交通の目的地にあたるノードと、そのノード間を相互に接続するために必要なリンクの集合」と定義する。ただし、コミュニティに属するノードは、分析者が事前に指定する。また、流入交通補足リンクとは、「コミュニティ内ノードの相互接続性を確保しながら、コミュニティへの流入交通を捕捉するために必要な、最小数のリンク集合」のことを指すこととする。この流入交通捕捉リンクは、コミュニティ外に任意に指定したノードから、コミュニティ内ノードに対する最小カットを求解することで求められる。グラフ理論におけるカットとは、グラフ  $G = (V, E)$  内のある2点  $s, t \in V$  に関して頂点集合  $s \in S, t \in T, S + T = V$  に分離するようなリンク集合  $\forall (u, v) \in E | u \in S, v \in T$  のことである。このカットに含まれる全リンクの容量の合計が最小のものが、最小カットと呼ばれる。これは図-1のように表現される。本手法では、全リンクの容量を1としたネットワークにおいて最小カットの容量を求めることで、最小カットに含まれるリンクの本数を求める。

本稿では、最小カット数を導出する問題を定式化し、ラグランジュ緩和問題、ラグランジュ双対問題を用いた効率的な求解アルゴリズムを提供する。また、提案手法をテストネットワークに適用した結果を示す。

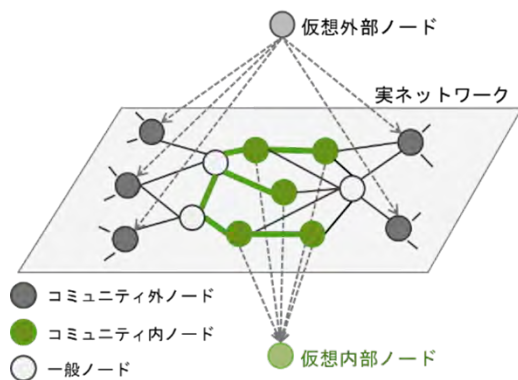


図-2 本手法における One-to-One ネットワーク

## 2.2 最小カット数算出のための One-to-One ネットワーク

杉浦<sup>5)</sup>は、道路ネットワーク上で算出した最小カット数がある地点の孤立脆弱性の指標とする道路網分析手法を提案した。この杉浦<sup>5)</sup>の提案手法では、複数の支援拠点と受援拠点を同時に考慮する目的から、仮想ノードを設定した One-to-One ネットワークにおいて最小カット数を算出している。本問題も、コミュニティ外に指定するノードとコミュニティ内ノードはそれぞれ複数想定され、それら全てを同時に考慮した最小カットを求める必要がある。そこで、杉浦<sup>5)</sup>と同様に、コミュニティ外ノード集合、コミュニティ内ノード集合を、それぞれを代替する仮想的な点（以下、それぞれ、仮想外部ノード、仮想内部ノードと呼ぶ。）に置き換える。そして、真のコミュニティ外ノードは仮想外部ノードと、真のコミュニティ内ノードは仮想内部ノードと、それぞれ仮想リンクにより接続する。ただし、仮想リンクは容量を十分大きくし、最小カット数を左右しないよう設定する。以上の解説を図示したものが図-2であり、本手法ではこのような拡張ネットワークにおける最小カットを求めることとする。

## 2.3 記号の定義

以下に定式化上使用する記号を示す。

$N$	ネットワーク中のノード集合。
$A$	ネットワーク中のリンク集合。
$\Psi$	特定コミュニティに含まれるノード集合。
$s, t$	それぞれ最大流問題のフローの起点、終点ノード。ただし $s, t \in N$ である。
$y$	最大流問題の解変数。
$x_a$	リンク $a$ の最大流問題のフロー。ただし $a \in A$ である。
$\theta_a$	リンク $a$ の補助フロー。ただし $a \in A$ である。
$n(\omega)$	コミュニティ内のノード。このうち特定のノードを $n(\omega)$ で表す。ただし $n(\omega) \in \Psi$ である。

## 2.4 最小カット数算出のための定式化

最小カット数の算出には、最大流問題による解法が一般的である。最大流問題とはグラフ中のある2点間に流すことのできるフローの最大量を求める問題であり、最大流最小カット定理によって、最大流の値は最小カットの容量

に一致することが証明されている。

ただし、本手法では、最小カットはコミュニティ内ノード間の相互接続性を確保した解を得る必要がある。そこで、最大流問題に対し、最小カットによって分離されるサブグラフにおけるコミュニティ内ノードが同一連結成分内に含まれることを保証する制約を加え、次のような混合整数線形計画問題として定式化する。

### 主問題 P

$$\max_{x, y, \theta} y \quad (1)$$

subject to

$$\sum_{a \in In(n)} x_a - \sum_{a \in Out(n)} x_a = \begin{cases} -y & \text{if } n = s \\ y & \text{if } n = t \\ 0 & \text{otherwise} \end{cases} \quad \forall n \in N \quad (2)$$

$$\sum_{a \in In(n)} \theta_a - \sum_{a \in Out(n)} \theta_a = \begin{cases} -|\Psi| + 1 & \text{if } n = n(\omega) \\ 1 & \text{if } n \in \Psi \\ 0 & \text{otherwise} \end{cases} \quad \forall n \in N \quad (3)$$

$$\theta_a \leq (|\Psi| - 1)(1 - x_a) \quad \forall a \in A \quad (4)$$

$$x_a \in \{0, 1\}, y \in \{0, 1, 2, \dots\}, \theta_a \in \{0, 1, 2, \dots\} \quad (5)$$

以下、これらの式の説明を加える。式(1)は、本問題の目的関数であり、未知変数  $y$  は流入交通捕捉リンクの本数にあたる。式(2)は、最大流問題のフロー保存制約である。ただし、本問題におけるフローの起点  $n = s$ 、終点  $n = t$  はそれぞれ、図-2における仮想外部ノード、仮想内部ノードに対応する。式(3)は、コミュニティ内のノードが同一連結成分内に含まれることを保証するために導入した補助フローのフロー保存制約であり、別途解説を加える。式(4)は、式(2)と式(3)で用いる変数を関係づけるための式である。これは最大流問題の未知変数  $x_a$  が  $\{0, 1\}$  変数であることを利用したもので、最大流問題のフロー  $x_a$  と補助フロー  $\theta_a$  を同一のリンクに流下させないための制約である。具体的には、 $x_a = 1$  のときは  $\theta_a \leq 0$  となりこのリンク  $a$  に補助フローは流せず、 $x_a = 0$  のときは  $\theta_a \leq (|\Psi| - 1)$  となりこのリンク  $a$  には最大で  $(|\Psi| - 1)$  単位の補助フローが流せることを表現するものである。

ここで、式(3)について解説を加える。この制約は、最小カットにより分離されたサブグラフにおいて、コミュニティ内ノードが同一連結成分内に含まれることを保証するものである。コミュニティ内の任意のノード  $n = n(\omega)$  を補助フローの起点として設定し、 $(|\Psi| - 1)$  単位の補助フローを流す。この補助フローは任意のノードを通過するが、コミュニティ内ノード  $n = n(\omega)$  を通過するときのみ、1単位ずつ消費される。補助フローがコミュニティ内の全ノードを通過したとき、起点から流下させた補助フローが全て消費されている状態であり、補助フローによってコミュニティ内ノードは同一連結成分として接続され

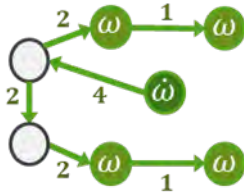


図-3 補助フロー流下の図解

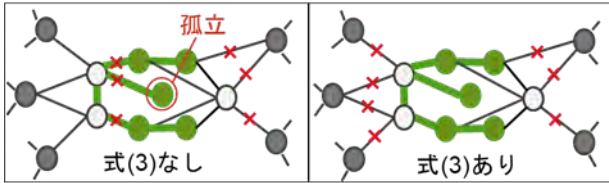


図-4 コミュニティ内ノードが同一連結成分内にあることを保証する制約の有無による解の比較

る。この条件を図-3で例示する。ただし、図中の数字は、各リンクに流下する補助フローの単位数である。

この式(3)による制約を入れない場合は、先の杉浦<sup>5)</sup>にて定式化された問題の解と一致する。本手法による解との比較を図解したものが図-4である。図-4の左のグラフがこの制約がない場合、右のグラフが制約がある場合の解である。左側のグラフは、中央のノードが他のノードに対する接続を失っていることから、本手法におけるコミュニティの条件を満たさない解となっていることを示している。

2.5 ラグランジュ緩和による演算効率化

本問題の出力である流入交通捕捉リンクは、指定したコミュニティ外ノードを通過しコミュニティ内ノードに向かう全経路を捕捉することを目指している。ゆえに、子細なリンクも含むネットワークデータを用いた計算が望ましく、それに伴い計算コストも膨大になることが想定される。そこで、本節では主問題Pの演算の効率化を図る緩和問題を考える。

主問題Pは、変数  $x_a$  が整数制約をもつ混合整数線形計画問題である。一般に混合整数線形計画問題は線形計画問題と比べ求解が非常に難しく、計算負荷が大きい大規模なネットワークでは解の探索が困難になる。ただし、主問題Pの式(2)、(3)は、いずれも左辺の係数行列が完全ユニモジュラ行列である。このような係数行列をもつ制約式は、式中の変数の整数制約を実数緩和しても、整数解が得られることが証明されている。そこで、主問題Pの式(4)を、非負のラグランジュ乗数  $\lambda$  を用いて、

$$\sum_{a \in A} \{\lambda_a (|\Psi| - 1)(1 - x_a) - \theta_a\} \quad (6)$$

のようにラグランジュ緩和する。この式(6)を、主問題の式(1)に組み込むことで、次のラグランジュ緩和問題RPを得る。

ラグランジュ緩和問題 RP

$$\max_{x,y,\theta} y + \sum_{a \in A} \{\lambda_a (|\Psi| - 1)(1 - x_a) - \theta_a\} \quad (7)$$

subject to

$$\sum_{a \in In(n)} x_a - \sum_{a \in Out(n)} x_a = \begin{cases} -y & \text{if } n = s \\ y & \text{if } n = t \\ 0 & \text{otherwise} \end{cases} \quad \forall n \in N \quad (8)$$

$$\sum_{a \in In(n)} \theta_a - \sum_{a \in Out(n)} \theta_a = \begin{cases} -|\Psi| + 1 & \text{if } n = n(\omega) \\ 1 & \text{if } n \in \psi \\ 0 & \text{otherwise} \end{cases} \quad \forall n \in N \quad (9)$$

$$x_a \geq 0, \theta_a \geq 0, \lambda_a \geq 0 \quad \forall a \in A \quad (10)$$

主問題Pは最大化問題であることから、この緩和問題RPでは主問題Pの厳密解を上回る解が得られる。そこで、緩和問題RPの最適解  $y^*, x_a^*, \theta_a^*$  を用いて、目的関数値を最小にする  $\lambda$  を探索する問題、すなわち、

$$\min_{\lambda} y^* + \sum_{a \in A} \{\lambda_a (|\Psi| - 1)(1 - x_a^*) - \theta_a^*\} \quad (11)$$

を考える。これが主問題Pに対するラグランジュ双対問題DPである。

ラグランジュ双対問題 DP

$$\min_{\lambda} y^* + \sum_{a \in A} \{\lambda_a (|\Psi| - 1)(1 - x_a^*) - \theta_a^*\} \quad (12)$$

$$x_a \geq 0, \theta_a \geq 0, \lambda_a \geq 0 \quad \forall a \in A \quad (13)$$

ラグランジュ双対問題の解を探索するためのヒューリスティックなアルゴリズムとして、劣勾配法がよく知られている。この劣勾配法の基本的なアルゴリズムを以下に示す。ただし、劣勾配を  $\partial_{ag}(\lambda)$  ( $a \in A$ ) で表す。

- Step1.  $\lambda$  を任意の非負定数からなる、実リンクの本数を次元にもつベクトルとする。また、繰り返し回数を  $i = 1$  とする。
- Step2.  $\lambda$  に対応するラグランジュ緩和問題の最適解  $x^*, \theta^*$  の組を求め、劣勾配  $\partial_{ag}(\lambda) = (|\Psi| - 1)(1 - x_a^*) - \theta_a^*$  を計算する。
- Step3. すべての  $a \in A$  に関して  $\frac{|\partial_{ag}(\lambda)|}{i} < \epsilon$  ならば、そのときの  $\lambda$  を出力して終了する。そうでないならば、 $\lambda_a = \max\{0, \lambda_a - \partial_{ag}(\lambda)/i\}$  を代入する。 $i = i + 1$  とし、Step2へ戻る。

2.6 最小カット集合の列挙方法

以上で求めた最小カット数をもとに、最小カット集合に含まれるリンクを特定する。最小カット列挙のためにはこれまでに様々なアルゴリズムが提案されているが、本研究では、Provan and Shier<sup>6)</sup>のアルゴリズムを採用す

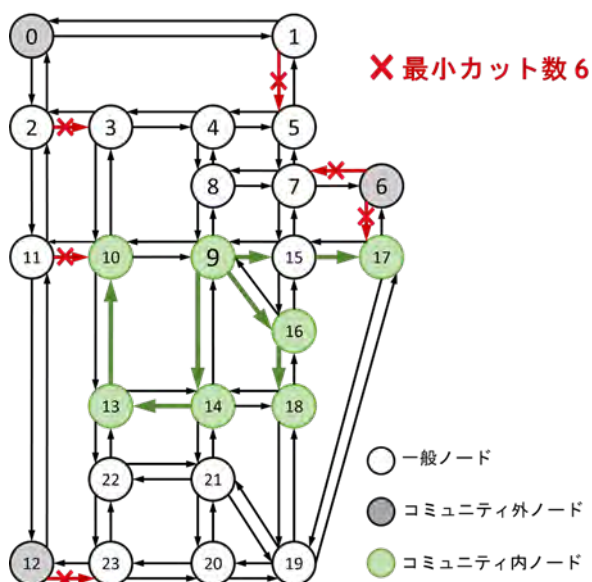


図-5 本問題の出力例

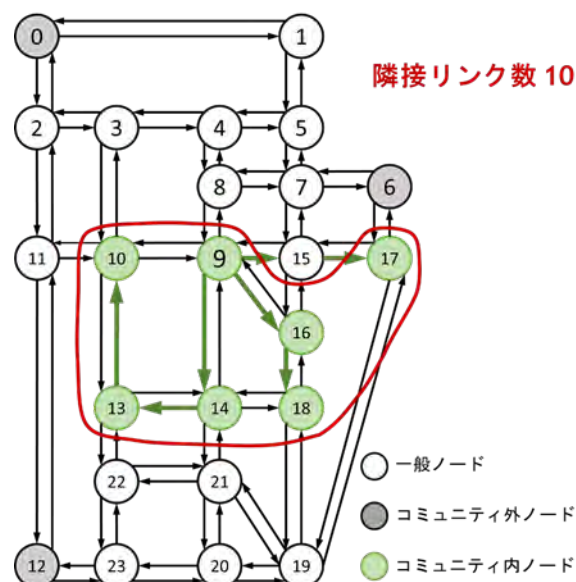


図-6 コミュニティ内ノードに対する隣接リンクを全列挙したコードン

る。この方法は高速にすべての最小カットを列挙する事が可能である。実装には python-igraph のメソッドである、all-st-mincuts を利用する。また、最小カット集合に含まれるリンクの組み合わせは複数想定されるため、実道路ネットワーク上で適用する場合など最小カット集合を一意に決定したい場合には、最小カットを全列挙したのちに、各リンクの車線数の多少を考慮した決定方法等が考えられる。次章の計算結果においては、以上の方法により特定した最小カット集合の一例を挙げるものとする。

### 3. テストネットワークでの試算

ここでは、Sioux Falls ネットワーク（ノード数 24、リンク数 76）を試算対象ネットワークとする。ただし、通行方向を考慮した最小カット集合を特定するため、有向グラフとして扱う。また、全リンクを均質なものとし、容量を 1 とする。ノード番号 0, 6, 12（灰色）を本問題におけるフローの起点となるコミュニティ外ノード、ノード番号 9, 10, 13, 14, 16, 17, 18（緑色）をコミュニティ内ノードとして指定し、全起点ノードから全終点ノード間の最小カット集合を特定した例を、図-5 に示す。なお、図-5 における最小カットが、本問題における流入交通捕捉リンクにあたる。また、本問題の有用性を示すため、参考として、コミュニティ内ノードに隣接するリンクを全列挙しコミュニティに対して閉じたコードンを形成した場合を図-6 に示す。

図-5 の最小カットは、対象コミュニティに隣接するリンクを全て囲むように形成されたコードン（図-6）と比べて少数のリンクが指定され、効率的な状態だといえる。また、図-5 中の最小カットにより全体ネットワークから分離されたサブグラフ内において、コミュニティ内の全ノードが補助フローにより同一連結成分内にあることが保証されていることから、全ノード間が相互に接続可能である。以上から本問題の条件を満たす解が得られたと確認できる。

### 4. まとめと今後の課題

本稿では、道路ネットワーク上の特定コミュニティに流入する交通を捕捉するリンク集合を決定する問題を構築した。また、定式化された問題の計算負荷低減のため、制約式をラグランジュ緩和することで、効率的な演算方法を検討した。

今後は、ラグランジュ双対問題の解の収束性向上のため共役劣勾配法の適用を予定している。また、異なる形状のネットワークでも演算を行い解の一般性を確認するとともに、実道路ネットワークに対して本問題の適用を行う。

### 参考文献

- 1) 東京都環境局. ロードプライシング検討委員会報告書, 2022/12/9 閲覧. <https://www.kankyo.metro.tokyo.lg.jp/vehicle/management/price/report.html>.
- 2) Hai Yang, L Gan, and Wilson Hon Chung Tang. Determining cordons and screen lines for origin-destination trip studies. *Proceedings of the Eastern Asia Society for Transportation Studies*, p. 85, 2001.
- 3) Xiaoning Zhang and Hai Yang. The optimal cordon-based network congestion pricing problem. *Transportation Research Part B: Methodological*, Vol. 38, No. 6, pp. 517–537, 2004.
- 4) Xavier Mazur, Marco Rinaldi, Richard D Connors, and Francesco Viti. A topological approach for identifying pricing controller locations to ensure controllability of transportation networks. *EURO Journal on Transportation and Logistics*, Vol. 11, p. 100078, 2022.
- 5) 杉浦聡志. 辺連結度と最小カットセットを用いた道路網脆弱性分析. 土木計画学研究・講演集, Vol. 57, , 2018.
- 6) J. Scott Provan and Douglas R Shier. A paradigm for listing (s, t)-cuts in graphs. *Algorithmica*, Vol. 15, No. 4, pp. 351–372, 1996.