

静的均衡配分による 横浜市の渋滞状況の検証と 対策



神戸大学チーム

藤原 龍・川上 航・福田 和輝・河瀬 理貴

背景・方針 (Background・Objective)



国道16号(保土ヶ谷バイパス)が
全国一位の交通量(約10万9千台/12h)

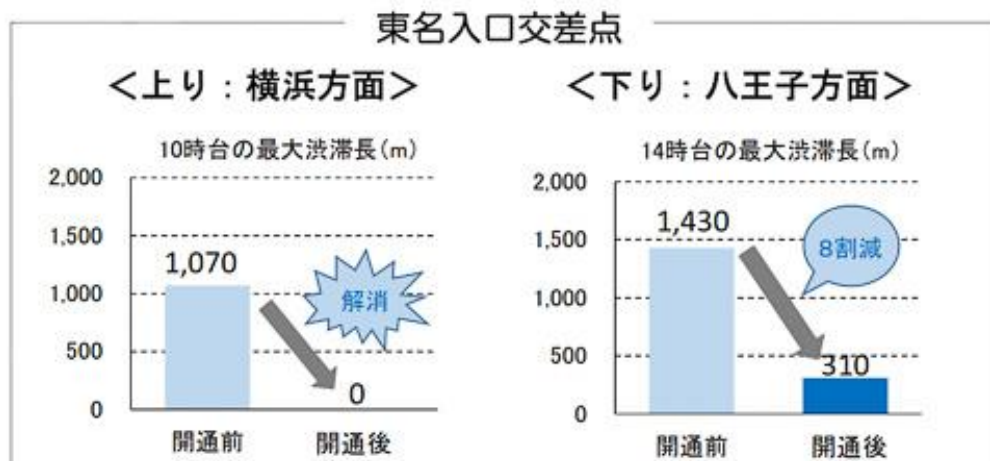
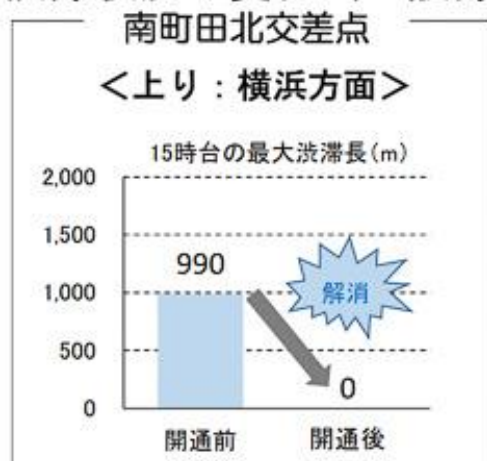
→ 慢性的な交通渋滞
Chronic traffic jam

立体化
Steric structure



背景・方針 (Background・Objective)

○渋滞状況の変化（一般部）



※昼間12時間での最大渋滞長（一般部）を表示

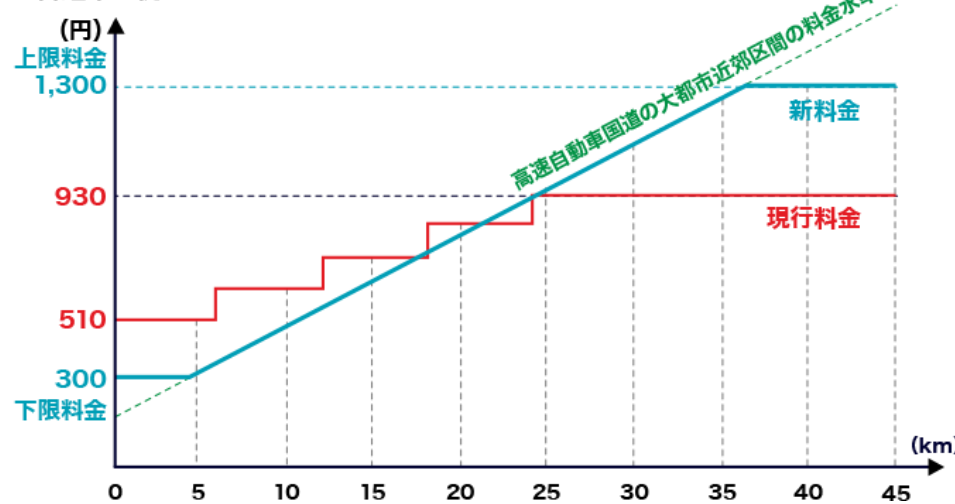
バイパスの渋滞は軽減したが、周りの状況の変化はどうか？

- **交通量配分計算**を行い検証する

配分計算(Assingnment・Calculation)

- 横浜市の交通状況を検証するため**Frank Wolfe法**を用い配分計算を行う。
- 再現性の向上の為、横浜市を走る首都高速道路の料金表を元に**料金を考慮した経路選択モデル**を構築する。

▼普通車の例



2016年度首都高速道路新料金プラン
<http://www.shutoko.jp/fee/fee-info/capitalarea-new/mex-new/>

BPR関数(BPR・Function)

首都高の料金表を元にBPR関数に**料金項**を追加。料金を考慮したBPR関数を作成する。

$$t(x) = t_0 \left\{ 1 + \alpha \left(\frac{x}{C} \right)^\beta \right\} + \varepsilon \frac{\tau L}{\eta}$$

- $t(x)$: link trip time function(hour)
- t_0 : free flow trip time(hour)
- α, β : parameter
- C : capacity of link(veh/hour)
- x : vehicle of link(veh/hour)
- τ : motorway fee(yen/km)
- L : length of link(km)
- η : time of value (yen/hour)
- ε : binary parameter
(1: motorway, 0: not motorway)

事前課題における問題点(problem)

●膨大なリンク数による計算コスト

- ・実リンク数 379000
- ・実ノード数 133490




●配分計算アルゴリズム

- ・バイナリヒープダイクストラの実装
- ・One to manyによる経路探索

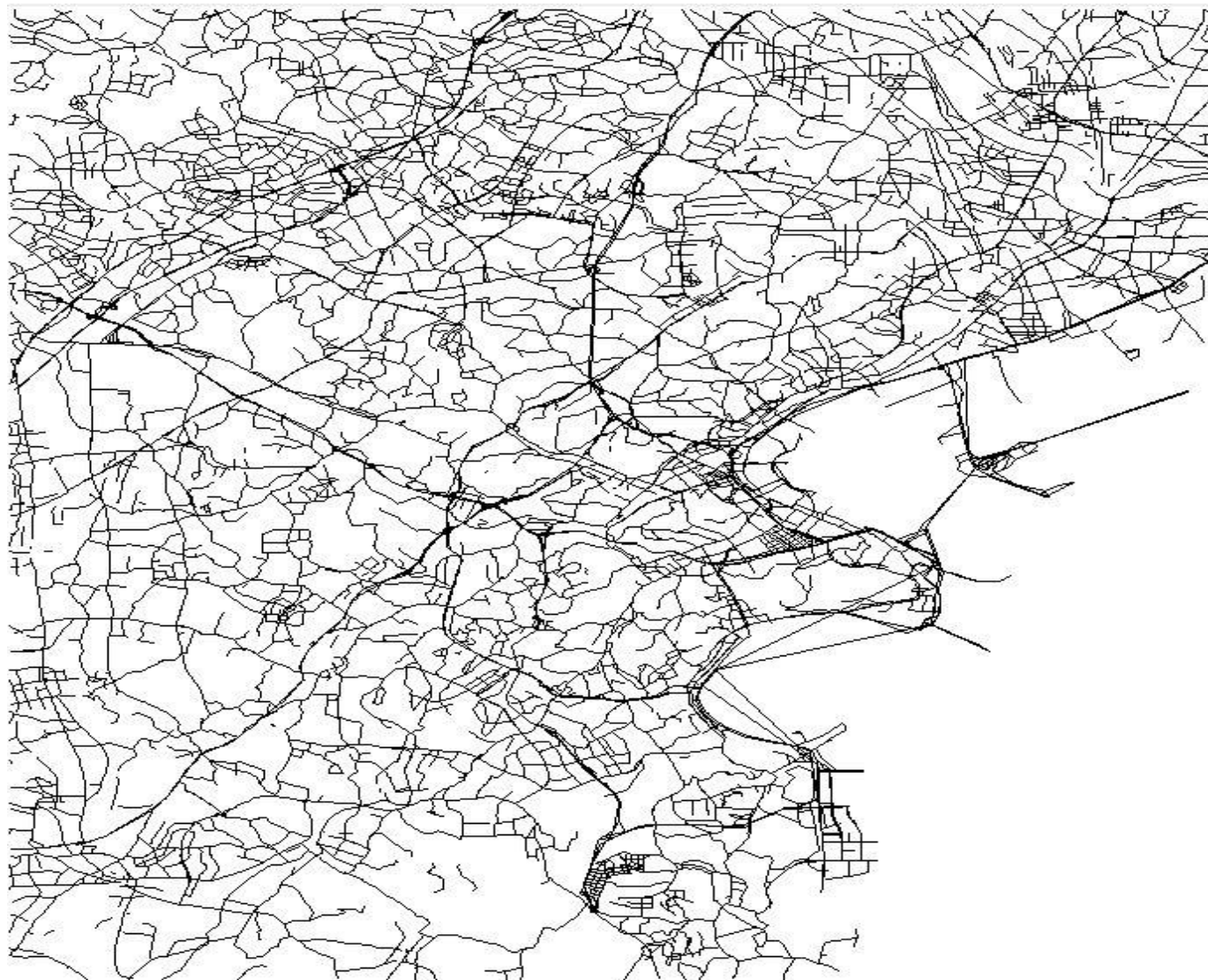
ネットワークデータの可視化(network data)



ネットワークデータの整理(network data)

道路種別	説明	写真
unclassified	国の道路体系で最も重要度の低い、通過するための道路です。	
residential	居住地間を接続する機能がなく、住宅へのアクセスとして機能する道路です。	
service	個人の住宅、キャンプ場、オフィス地域、駐車場などの中にある、アクセスのための道路です。	

ネットワークデータの整理(network data)



ネットワークデータの整理(network data)

OD交通量 ゾーン数	平成20年東京都市圏PT調査 229
ネットワーク	OpenStreetMap ノード数:32565 リンク数:71010
観測断面	道路交通センサス(平成22年全国 道路交通情勢調査)
時間評価値	1200円/hour(国土交通省より)

道路交通データの概要

配分計算アルゴリズム(algorithm)

バイナリヒープダイクストラ

- ダイクストラ法は任意の始点に繋がるノードから、**ループを使用して**距離が最も短いノードが選ばれる。

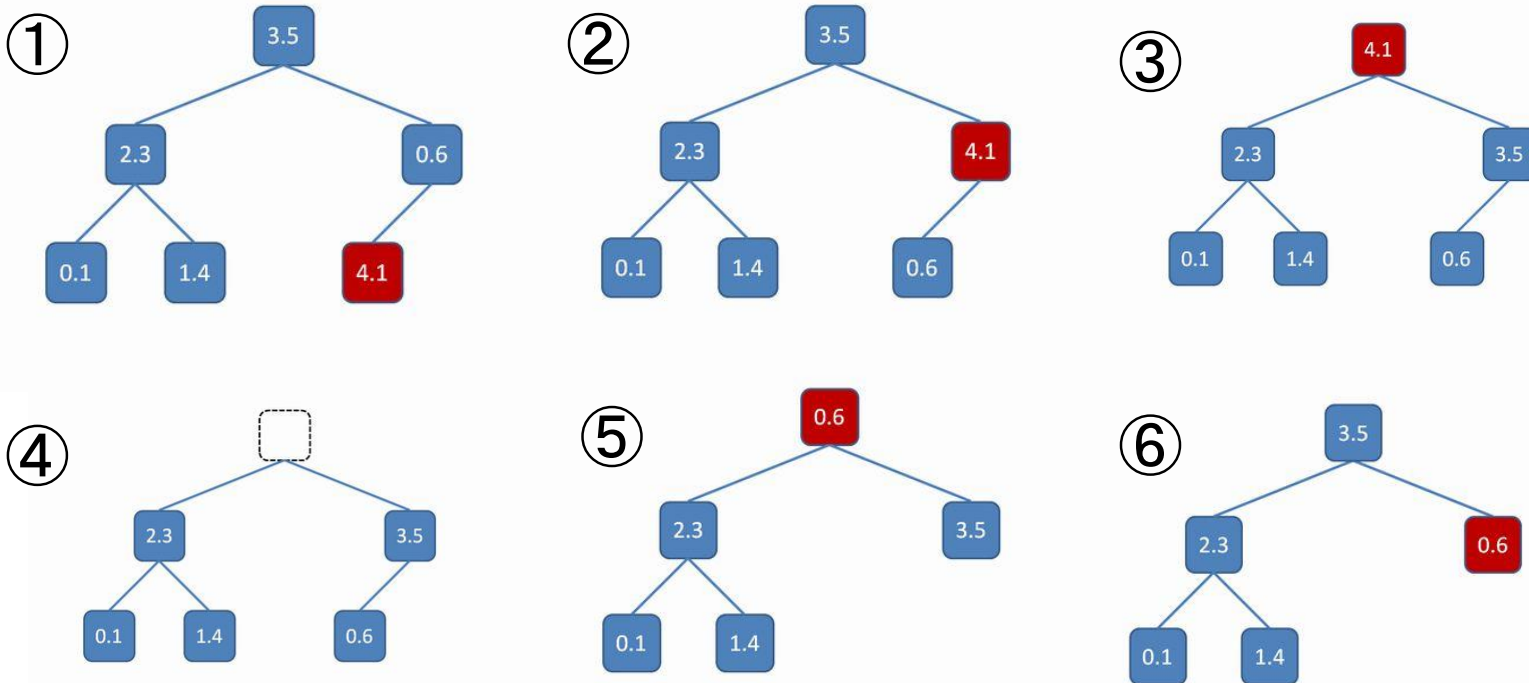
各ノードへの最短距離の更新、次のノードの抜き出しの高速化



- ヒープというデータ構造**を用いて、最小値をひと手間で取り出すことができる。
- バイナリヒープは親ノードよりも子ノードのほうが大きい
か等しい。

配分計算アルゴリズム(algorithm)

バイナリヒープダイクストラ



配分計算アルゴリズム(algorithm)

One to Manyによる経路探索

many to many

一回の計算で一つのodの最短経路のみを求める

one to many

一回の計算で複数のノードまでの最短経路を求める。

配分計算アルゴリズム(algorithm)

計算速度

バイナリヒープダイクストラ

$$O(n^2) \longrightarrow O(n \log n)$$

one to many

$$O(mn \log n) \longrightarrow O(m \log n)$$

m :リンク数 n :ノード数

配分計算結果(output)

- SiouxFallsなら配分計算はできたが...

横浜のネットワークでは計算できなかった...

なぜ回らなかった?

1. 経路選択で先行ポインタを辿れない事がある。
→新しく実装したdijkstra法中のヒープのソート操作にミスがあった?
2. ネットワークを縮小した事によりODが途切れた?
3. Frank Wolfe法の収束条件を厳しくすると回らない
→Frank wolfeの収束条件が厳しすぎた?

参考資料

● OD表の整理

横浜市は小ゾーン区分に
横浜市以外は大ゾーンの
セントロイドに集約



参考資料

● BPR関数パラメータ

配分交通量の予測に用いたパラメータ

	α	β
都市間高速道路	0.54	3.30
都市間高速道路	0.40	2.80
幹線多車線道路	0.54	2.40
幹線2車線道路	0.44	3.10
準幹線多車線道路	0.41	2.20
準幹線2車線道路	0.49	2.40

参考文献：松井寛、山田周治（1988）、道路交通センサデータに基づくBPR関数の設定

参考資料

● RMS誤差



$$\text{RMS link error} = \frac{1}{N} \sum_{l \in E_r} \frac{\sqrt{(x_h^l - x_r^l)^2}}{x_r^l}$$

x_h^l : リンクの 配分交通量

x_r^l : 交通センサス一般交通量調査による観測断面交通量

E_r : 観測実施リンクの集合