Bayesian Demand Calibration for Dynamic Traffic Simulation

Gunnar Flötteröd, Michel Bierlaire, Kai Nagel

Transportation Science, Vol. 45 No. 4, pp541-561, 2011

2022/05/20
理論談話会 #8
M1 望月陽介

Abstract

- What

  非集計、かつ動的な交通シミュレータのcalibration手法

  =DTA(Dynamic Traffic Assignment) sim

- How
  - ベイズの枠組み
  - data：シミュレータ＋トラカン

- Result

  シミュレータへの適用、大規模なユースケースにも対応

式番号は論文に準拠、途中抜けあり

構成

1. Introduction
2. 集計予測モデルでのCalibration
   - 定式化
   - 計算実験
3. 非集計、動的シミュレータへの拡張
   - シミュレータ定義
   - 定式化
4. 非集計シミュレータでのアルゴリズム
   - 運用のためのアルゴリズム導入
   - 計算実験
5. Case Study
6. Summary & Outlook

1. Introduction

- 問題設定
- 事前知識

設定

- Micro-simulators(Calibration対象)

  今回はdisaggregate, dynamicなdemand simulatorが対象(supplyは所与)
  モデルの解像度が上がる=モデルの自由度が高まる
  →相互影響のモデル化・パラメータ決定(=calibration)の難易度が上がる

- パラメータ

  今回シミュレーションは、結果の行動を表す
  効用関数の切片など、選択確率を表現するパラメータにも対応可能(summary)

- データ

  道路上の交通量

既往研究(Calibration)

- staticな条件でのモデル
  - OD matrix estimation
  - Path flow estimation
    - 尤度最大化(Spiess 1987)、最小二乗法(Bell 1991)、エントロピー最大化(Zuylen and Willumsen 1980)

- dynamic
  - aggregate simulators(Ashok and Ben-Akiva 2000)
    →いくつもの仮定
    - 時間方向の相関関係、または一定の時間内で静的な予測
  - micro-simのcalibration(Sevcikova et al. 2007)
    - 計算可能性(パラメータと観測値間の操作について連続・微分可能・正規性など
    - 計算時間

事前知識

- demand calibrationの一手法：エントロピー関数最大化
  - 不確定性を表すエントロピーを最大にする=条件のもと偏りのない推定を行う(分布を生成)

情報量: $-\log p$（ビット）
平均情報量(エントロピー): $H = -\sum_{d \in D} p(d) \log p(d)$
  - 事象により得られる情報量の期待値

条件付きエントロピー

$$H(D|Y = y) = -\sum_{d \in D} p_{D|Y}(d|Y = y) \log p_{D|Y}(d|Y = y)$$

  - 観測$Y = y$が得られた時の、交通流Dのエントロピー

2. Aggregate Path Flow Estimation

- The prior/posterior solution
- 計算実験

問題設定

- 集計的な経路流予測
  - ODペアごとに、経路流量を予測する
  - リンク上のトラカンを観測量として利用
  - 時間軸は省略
  - 同質な旅行者を考慮

- 文字
  - N : $\#OD\ pairs$
  - d_n: $the\ largest\ possible\ \#trips\ between\ OD\ pair\ n$
  - C_n: $the\ set\ of\ available\ paths\ that\ connect\ OD\ pair\ n$
  - d_{ni}: $\#trips\ on\ path\ i\ \in C_n, where\ d_n = \sum_{i \in C_n} d_{ni}$
  - p, P: $probability\ density\ function, discrete\ probability\ function$
  - $\boldsymbol{x}$: $the\ vector\ of\ network\ conditions$
  - $\boldsymbol{y}$: $the\ traffic\ counts$

The prior solution

- 観測なしのPFE(Path Flow Estimation)

N : #OD pairs
d_n: the largest possible #trips between OD pair n
C_n: the set of available paths that connect OD pair n
d_{ni}: #trips on path $i \in C_n$, where $d_n = \sum_{i \in C_n} d_{ni}$
p, P: probability density function, discrete probability function
$\boldsymbol{x}$: the vector of network conditions
$\boldsymbol{y}$: the traffic counts

SUEを達成する経路流

$$d_{n,i} = P_n\big(i|\boldsymbol{x}(\boldsymbol{d})\big)d_n \qquad (1)$$

は、事前エントロピー

$$W(\boldsymbol{d}) = \sum_{n=1}^{N} \sum_{i \in C_n} [d_{ni} \ln P_n(i|\boldsymbol{x}(\boldsymbol{d})) - d_{ni} \ln d_{ni}] + \sum_{n=1}^{N} d_n \ln d_n$$
$$\text{s.t.} \quad d_n = \sum_{i \in C_n} d_{ni} \quad for\ all\ n = 1, 2, \dots, N \qquad (2)$$

を最大化する経路流$\boldsymbol{d}$に相当する。(証明はAppendix A)

The posterior solution

- 観測ありのPFE
  - 前ページの条件に、観測$\boldsymbol{y}$の考慮を加える

  ＝観測$\boldsymbol{y}$が妥当な割合で再生成される経路流を予測

N : #OD pairs
d_n: the largest possible #trips between OD pair n
C_n: the set of available paths that connect OD pair n
d_{ni}: #trips on path $i \in C_n$, where $d_n = \sum_{i \in C_n} d_{ni}$
p, P: probability density function, discrete probability function
$\boldsymbol{x}$: the vector of network conditions
$\boldsymbol{y}$: the traffic counts

事後エントロピー

$$W(\boldsymbol{d}|\boldsymbol{x}) = \ln p(\boldsymbol{y}|\boldsymbol{x}(\boldsymbol{d})) + W(\boldsymbol{d}) \qquad (3)$$

を最大化するような経路流$\boldsymbol{d}$を求める(Appendix B)。

$$P_n(i|\boldsymbol{x}(\boldsymbol{d}), \boldsymbol{y}) = \frac{\exp(\Lambda_{ni}+\Gamma_{ni}) P_n(i|\boldsymbol{x}(\boldsymbol{d}))}{\sum_{j \in C_n} \exp(\Lambda_{nj}+\Gamma_{nj}) P_n(j|\boldsymbol{x}(\boldsymbol{d}))}, \qquad (4)$$

$$where\ \ \Lambda_{ni} = \frac{\partial \ln p(\boldsymbol{y}|\boldsymbol{x}(\boldsymbol{d}))}{\partial d_{ni}}, \Gamma_{ni} = \sum_{m=1}^{N} \sum_{j \in C_n} \frac{d_{mj}}{P_m(j|\boldsymbol{x}(\boldsymbol{d}))} \frac{\partial P_m(j|\boldsymbol{x}(\boldsymbol{d}))}{\partial d_{ni}}$$

(5) (6)

得られた解の解釈

$$P_n(i|\boldsymbol{x}(\boldsymbol{d}),\boldsymbol{y}) = \frac{\exp(\Lambda_{ni}+\Gamma_{ni})P_n(i|\boldsymbol{x}(\boldsymbol{d}))}{\sum_{j\in C_n}\exp(\Lambda_{nj}+\Gamma_{nj})P_n(j|\boldsymbol{x}(\boldsymbol{d}))},$$

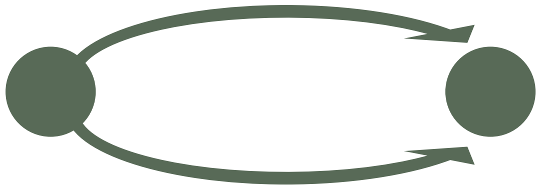
$$where\ \ \Lambda_{ni} = \frac{\partial \ln p(\boldsymbol{y}|\boldsymbol{x}(\boldsymbol{d}))}{\partial d_{ni}}, \Gamma_{ni} = \sum_{m=1}^{N}\sum_{j\in C_n}\frac{d_{mj}}{P_m(j|\boldsymbol{x}(\boldsymbol{d}))}\frac{\partial P_m(j|\boldsymbol{x}(\boldsymbol{d}))}{\partial d_{ni}}$$

- Λ_{ni} , Γ_{ni}の解釈
  - Λ_{ni}は経路流d_{ni}が(尤度)=(観測値の再生成)に与える影響を表している。
  - Γ_{ni}は経路選択がネットワーク状態に対して鈍感であると0になる。
    →計算コストと感度の問題から、後に省かれる

- 解を得るための計算
  - Global Optimとは限らない(が、多くのケースで凸性が保証されており、唯一の解)
  - 微分計算が伴う

計算実験

- シンプルなネットワークでPFEを実行



$$t(d_i) = \left(\frac{d_i}{750}\right)^2, i = 1,2 \qquad (8)$$

$$\boldsymbol{x}(\boldsymbol{d}) = \begin{pmatrix} x_1(\boldsymbol{d}) \\ x_2(\boldsymbol{d}) \\ x_3(\boldsymbol{d}) \end{pmatrix} = \begin{pmatrix} t\,(d_1) \\ t\,(d_2) \\ d_1 \end{pmatrix} \qquad (9)$$

$\boldsymbol{y}$はroute1での経路流：y_1

N : #OD pairs
d_n: the largest possible #trips between OD pair n
C_n: the set of available paths that connect OD pair n
d_{ni}: #trips on path $i \in C_n$, where $d_n = \sum_{i \in C_n} d_{ni}$
p, P: probability density function, discrete probability function
$\boldsymbol{x}$: the vector of network conditions
$\boldsymbol{y}$: the traffic counts

- 解(経路流=リンク流)は以下から求まる

$$P_n(i|\boldsymbol{x}(\boldsymbol{d})) = \frac{\exp(-t(d_1))}{\exp(-t(d_1))+\exp(-t(d_2))}, \qquad (10)$$

$$p(\boldsymbol{y}|\boldsymbol{x}(\boldsymbol{d})) \propto \exp\left[-\frac{(y_1-d_1)^2}{2\sigma_1^2}\right], \qquad (11)$$

($\sigma_1$is the standard deviation of the sensor data)

計算実験結果

- 観測値と観測精度により推定結果を可視化(Fig 1)
  - 観測がない場合と、観測による情報の両方を加味した推定に
- 観測値と観測精度により簡易アルゴリズムの精度を可視化(Fig 2)
  - 計算コストの高いΓ_{ni}を0とした場合(簡易アルゴリズム)の、exact解との差の図
  - 簡易アルゴリズムでも十分な精度であることを確認

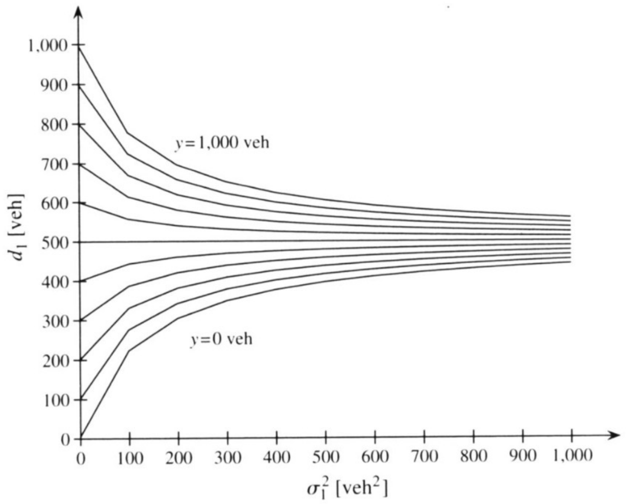


Figure 1 Calibration Results for Two Routes Example

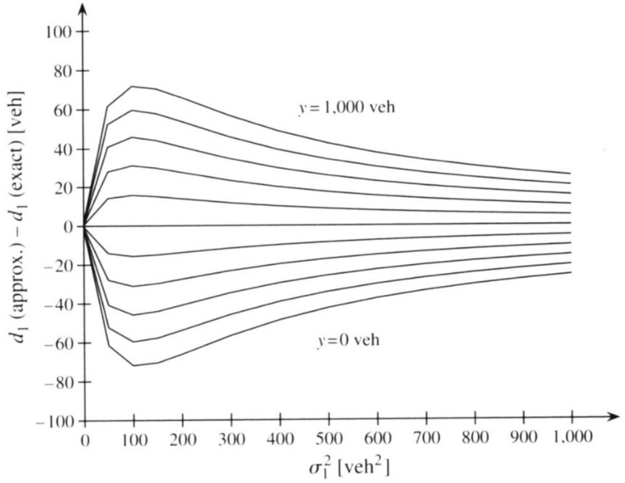


Figure 2 Bias of Simplified Calibration for Two Routes Example

3. Disaggregate Demand Calibration

- DTA simの設定(PFEとの対応)
- The prior/posterior solution

問題設定

- PFEから、問題設定を置き換える

Symbol	Macroscopic	Microscopic
$n = 1, 2, \ldots, N$	OD pairs	Agents
C_n	Routes connecting OD pair n	Plans available to agent n
$i \in C_n$	A route connecting OD pair n	A plan available to agent n
d_{ni}	#Trips on route $i \in C_n$	Stationary probability that agent n chooses plan $i \in C_n$
d_n	#Trips in OD pair n	One $\left(d_n = \sum_{i \in C_n} d_{ni}\right)$

- Supply sim
  - plan set $\{i\}$から、network condition $\boldsymbol{x}$を生成する
- Demand sim
  - agent nに対して、$P_n(i|\bar{\boldsymbol{x}})$に基づきplan iを生成する

DTA simの解法

- 需要供給が一貫する状況を得る
  (繰り返し計算)

N : #Agents
d_n: 1 for each agent n
C_n: the set of available plans for agent n
d_{ni} : Stationary probability that agent n chooses plan $i \in C_n$
p, P: probability density function, discrete probability function
$\boldsymbol{x}$: the vector of network conditions
$\boldsymbol{y}$: the traffic counts

Algorithm 1 (Iterative dynamic traffic assignment)

1. Initialize cycle counter c = 0. — free-flowなど
2. Choose initial network conditions $\boldsymbol{x}^0, \boldsymbol{x}^{-1}, \ldots$
3. Repeat for as many iterations as necessary:
   (a) Increase c by one.
   (b) Calculate expected network conditions $\overline{\boldsymbol{x}}^c$ from $\boldsymbol{x}^{c-1}, \boldsymbol{x}^{c-2}, \ldots$ — 移動平均、自己回帰など
   (c) Replanning. For $n = 1, 2, \ldots, N$, draw plan i_n^c from $P_n(i_n^c | \overline{\boldsymbol{x}}^c)$. — demand sim
   (d) Network loading. Draw network conditions $\boldsymbol{x}^c$ from $p(\boldsymbol{x}^c | \{i\}^c)$. — supply sim

→ $\boldsymbol{x}^c, \{i\}^c$が安定

The prior solution

- Stationary Distribution
  - 集計モデルの均衡状態に対応する

N : $\#Agents$
d_n: $1\ for\ each\ agent\ n$
C_n: $the\ set\ of\ available\ plans\ for\ agent\ n$
d_{ni} : Stationary probability that agent n chooses plan $i \in C_n$
p, P: $probability\ density\ function, discrete\ probability\ function$
$\boldsymbol{x}$: $the\ vector\ of\ network\ conditions$
$\boldsymbol{y}$: $the\ traffic\ counts$

$$\Pi_n(i) = P_n(i|\overline{\boldsymbol{x}}), i \in C_n, n = 1, 2, \ldots, N, \qquad (13)$$

$$\Pi(\{i\}) = \prod_{n=1}^{N} \Pi_n(i_b), \qquad (14)$$

$$\pi(\boldsymbol{x}) = p(\boldsymbol{x}|\{i\} \sim \Pi(\{i\})), \qquad (15)$$

$$\overline{\boldsymbol{x}} \approx \boldsymbol{E}\{\boldsymbol{x}|\boldsymbol{x} \sim \pi(\boldsymbol{x})\}. \qquad (16)$$

$\boldsymbol{\pi(\overline{x}^c)}$の期待値と $\boldsymbol{E\{x|x \sim \pi(x)\}}$が一致 & $\boldsymbol{\pi(\overline{x}^c)}$の裾野が狭ければ、(13)は

$$\begin{aligned} \Pi_n(i) &= \int P_n(i|\overline{\boldsymbol{x}})\pi(\overline{\boldsymbol{x}})\, d\overline{\boldsymbol{x}} \\ &\approx \int \left[P_n(i|\overline{\boldsymbol{x}}^c) + \frac{\partial P_n(i|\overline{\boldsymbol{x}}^c)}{\partial \overline{\boldsymbol{x}}^c}(\overline{\boldsymbol{x}} - \overline{\boldsymbol{x}}^c)\right]\pi(\overline{\boldsymbol{x}})\, d\overline{\boldsymbol{x}} \qquad (17),(18) \\ &= P_n(i|\overline{\boldsymbol{x}}^c) \end{aligned}$$

により、Algorithm1で得られる分布がStationary Distributionであるといえる。

Application of the Calibration

N : $\#Agents$
d_n: $1\ for\ each\ agent\ n$
C_n: $the\ set\ of\ available\ plans\ for\ agent\ n$
d_{ni} : Stationary probability that agent n chooses plan $i \in C_n$
p, P: $probability\ density\ function, discrete\ probability\ function$
$\boldsymbol{x}$: $the\ vector\ of\ network\ conditions$
$\boldsymbol{y}$: $the\ traffic\ counts$

- variationのあるDTA simで、
  PFEでの解法がそのまま適用できる理由

全てのagent nを、サイズが$1/Z$であるZ人の集合であると考える
→ Zd_{ni}はN$(Z\Pi_n(i), Z\Pi_n(i)\left(1-\Pi_n(i)\right))$に従う
→ d_{ni}はN$(\Pi_n(i), \Pi_n(i)\left(1-\Pi_n(i)\right))/Z$に従う

すると、 $Z \to \infty$においてPFEと同様の確定的な出力を得ることができる。
（network loadingにはvariationが含まれるが）

→ 集計的なモデルと、micro-simulationによる出力の差は、均衡点のnetwork conditionによる小さな差でしかない(Cascetta 1989)

The posterior solution

- PFEの解(4)~(6)と
  the prior solution(13)~(16)から、

$$\Pi_n(i|\boldsymbol{y}) = \frac{\exp(\Lambda_{ni} + \Gamma_{ni}) P_n\left(i|\overline{\boldsymbol{x}}_{|\boldsymbol{y}}\right)}{\sum_{j \in C_n} \exp(\Lambda_{nj} + \Gamma_{nj}) P_n\left(j|\overline{\boldsymbol{x}}_{|\boldsymbol{y}}\right)}, \qquad (19)$$

$$\Pi(\{i\}|\boldsymbol{y}) = \prod_{n=1}^{N} \Pi_n(i_n|\boldsymbol{y}), \qquad (20)$$

$$\pi(\boldsymbol{x}|\boldsymbol{y}) = p(\boldsymbol{x}|\{i\} \sim \Pi(\{i\})|\boldsymbol{y}), \qquad (21)$$

$$\overline{\boldsymbol{x}}_{|\boldsymbol{y}} \approx \boldsymbol{E}\{\boldsymbol{x}|\boldsymbol{x} \sim \pi(\boldsymbol{x}|\boldsymbol{y})\}. \qquad (22)$$

を得る。

N : #Agents
d_n: 1 for each agent n
C_n: the set of available plans for agent n
d_{ni} : Stationary probability that agent n chooses plan $i \in C_n$
p, P: probability density function, discrete probability function
$\boldsymbol{x}$: the vector of network conditions
$\boldsymbol{y}$: the traffic counts

The posterior solutionの解法

N : $\#Agents$
d_n: 1 $for\ each\ agent\ n$
C_n: $the\ set\ of\ available\ plans\ for\ agent\ n$
d_{ni} : Stationary probability that agent n chooses plan $i \in C_n$
p, P: $probability\ density\ function, discrete\ probability\ function$
$\boldsymbol{x}$: $the\ vector\ of\ network\ conditions$
$\boldsymbol{y}$: $the\ traffic\ counts$

Algorithm 1

1. Initialize cycle counter c = 0.
2. Choose initial network conditions $\boldsymbol{x}^0, \boldsymbol{x}^{-1}, \ldots$
3. Repeat for as many iterations as necessary:
   (a) Increase c by one.
   (b) Calculate expected network conditions $\overline{\boldsymbol{x}}^c$ from $\boldsymbol{x}^{c-1}, \boldsymbol{x}^{c-2}, \ldots$
   (c) Replanning. For $n = 1, 2, \ldots, N$, draw plan i_n^c from $P_n(i_n^c | \overline{\boldsymbol{x}}^c)$.
   (d) Network loading. Draw network conditions $\boldsymbol{x}^c$ from $p(\boldsymbol{x}^c | \{i\}^c)$.

Algorithm 2

1. Initialize the calibration and the DTA sim.
2. Repeat for as many iterations as necessary:
   (a) Calculate all Λ_{ni} & Γ_{ni}.
   (b) For all agents $n = 1, 2, \ldots, N$,draw a new plan from a choice distribution scaled by $\exp(\Lambda_{ni} + \Gamma_{ni})$
   (c) Load all agents on the network

問題 1

How to Calibrate

問題 2

How to Scale

4. Making the Framework Operational

- 計算方法とアルゴリズム
- 数値実験

Scaling項の計算

- 2 章の数値実験結果から、 Γは無視する
  - 計算負荷が高い
  - 無視しても結果が変わらない

$$\Lambda_{ni} = \frac{\partial \ln p(\boldsymbol{y}|\boldsymbol{x}(\boldsymbol{d}))}{\partial d_{ni}},$$

$$\Gamma_{ni} = \sum_{m=1}^{N} \sum_{j \in C_n} \frac{d_{mj}}{P_m(j|\boldsymbol{x}(\boldsymbol{d}))} \frac{\partial P_m(j|\boldsymbol{x}(\boldsymbol{d}))}{\partial d_{ni}}$$

- Λの計算方法を考える

$$\Lambda_{ni} = \frac{\partial \ln p(\boldsymbol{y}|\bar{\boldsymbol{x}}_{|\boldsymbol{y}})}{\partial \Pi_n(i|\boldsymbol{y})} = \left\langle \frac{\partial \ln p(\boldsymbol{y}|\bar{\boldsymbol{x}}_{|\boldsymbol{y}})}{\partial \bar{\boldsymbol{x}}_{|\boldsymbol{y}}}, \frac{\partial \bar{\boldsymbol{x}}_{|\boldsymbol{y}}}{\partial \Pi_n(i|\boldsymbol{y})} \right\rangle \quad (23)$$

第一項は、確率密度関数から計算するだけ

第二項は、リンク旅行時間が固定されている状況を仮定して計算する。

Scaling項の計算

- A proportional network下での変形

$$\Lambda_{ni} = \frac{\partial \ln p(\boldsymbol{y}|\bar{\boldsymbol{x}}_{|\boldsymbol{y}})}{\partial \Pi_n(i|\boldsymbol{y})} = \left\langle \frac{\partial \ln p(\boldsymbol{y}|\bar{\boldsymbol{x}}_{|\boldsymbol{y}})}{\partial \bar{\boldsymbol{x}}_{|\boldsymbol{y}}}, \frac{\partial \bar{\boldsymbol{x}}_{|\boldsymbol{y}}}{\partial \Pi_n(i|\boldsymbol{y})} \right\rangle, \qquad (23)$$

$$x_a(k) = \sum_{n=1}^{N} \mathbf{1}(ak \in i_n), \qquad (24)$$

Indicator Function $\mathbf{1}$, time step k

$$\bar{x}_a(k)_{|\boldsymbol{y}} = \sum_{n=1}^{N} \sum_{i \in C_n} \mathbf{1}(ak \in i_n)\Pi_n(i|\boldsymbol{y}), \qquad (25)$$

$$\Lambda_{ni} = \sum_{ak \in i} \frac{\partial \ln p(\boldsymbol{y}|\bar{\boldsymbol{x}}_{|\boldsymbol{y}})}{\partial \bar{x}_a(k)_{|\boldsymbol{y}}}. \qquad (26)$$

<example>

$$\ln p(\boldsymbol{y}|\bar{\boldsymbol{x}}_{|\boldsymbol{y}}) = const - \sum_{ak} \frac{\left(y_a(k) - \bar{x}_a(k)_{|\boldsymbol{y}}\right)^2}{2\sigma_a^2(k)}, \qquad (27)$$

$$\Lambda_{ni} = \sum_{ak \in i} \frac{y_a(k) - \bar{x}_a(k)_{|\boldsymbol{y}}}{\sigma_a^2(k)} \qquad (28)$$

$$\Lambda_{ni} = \frac{\partial \ln p(\boldsymbol{y}|\boldsymbol{x}(\boldsymbol{d}))}{\partial d_{ni}},$$

$$\Gamma_{ni} = \sum_{m=1}^{N} \sum_{j \in C_n} \frac{d_{mj}}{P_m(j|\boldsymbol{x}(\boldsymbol{d}))} \frac{\partial P_m(j|\boldsymbol{x}(\boldsymbol{d}))}{\partial d_{ni}}$$

Scalingの実施

- Rejection Sampling(Ross 2006)により、確率を実質的に補正
  - agent nに対してプランiが抽出された時、
    採択率$P_{accept,n}(i)$で採択、 $1 - P_{accept,n}(i)$で棄却する
  - 棄却率$P_{accept,n}(i)$は以下

$$P_{accept,n}(i) = \exp(\Lambda_{ni})/D_n\,, \quad (29)$$

$$D_n \geq \max_{i \in C_n} \exp(\Lambda_{ni}) \quad (30)$$

  - プランの抽出と採択決定を、採択されるまで繰り返す

→確率$P_{accept,n}(i)$でのScaling(証明はAppendix D)

DTA simでのアルゴリズム

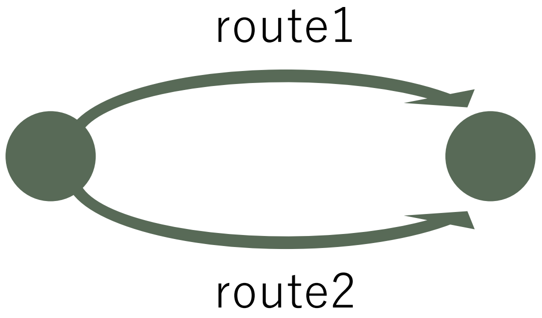
- Scalingの実装を踏まえて、Algorithm1と2を行う

Algorithm 3 (Calibration with the accept/reject estimator)

1. Initialize cycle counter $c = 0$.
2. Choose initial network conditions $\boldsymbol{x}^0, \boldsymbol{x}^{-1}, \ldots$.
3. Repeat for as many iterations as necessary:
    - (a) Increase c by one.
    - (b) Calculate expected network conditions $\overline{\boldsymbol{x}}^c_{|\boldsymbol{y}}$ from $\boldsymbol{x}^{c-1}, \boldsymbol{x}^{c-2}, \ldots$
    - (c) Replanning. For $n = 1, 2, \ldots, N$, do:
        - (i) Run the demand simulator and obtain a plan i'.
        - (ii) Calculate $\Lambda_{ni'}$ — Scaling項の計算
        - (iii) With probability $1 - P_{accept,n}(i')$, go to step 3(c(i)). — Rejection Sampling
        - (iv) Retain the first accepted draw: $i_n^c = i'$.
    - (d) Network loading. Draw $\boldsymbol{x}^c$ from $p(\boldsymbol{x}^c | \{i\}^c)$.

数値実験

- ２章と同じ設定



$$t(d_i) = \left(\frac{d_i}{750}\right)^2, i = 1,2$$

$$\boldsymbol{x}(\boldsymbol{d}) = \begin{pmatrix} x_1(\boldsymbol{d}) \\ x_2(\boldsymbol{d}) \\ x_3(\boldsymbol{d}) \end{pmatrix} = \begin{pmatrix} t\,(d_1) \\ t\,(d_2) \\ d_1 \end{pmatrix}$$

$\boldsymbol{y}$はroute1での経路流：$\boxed{y_1 = 250, \sigma_1 = 10}$

- プラン、network condition共にcalibrationの効果確認

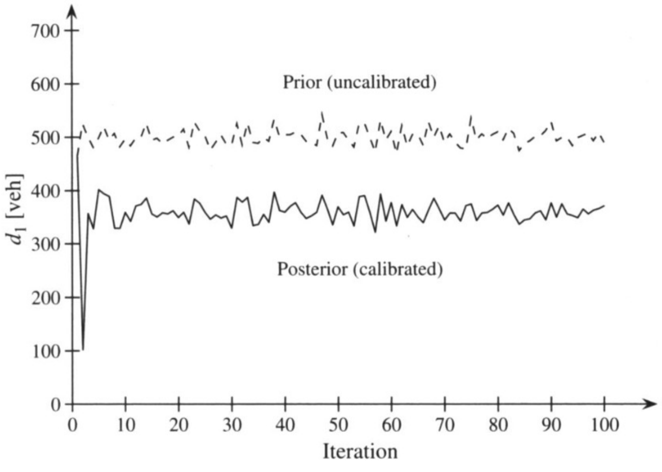


Figure 3 **Evolution of d_1 for Two Routes Example**

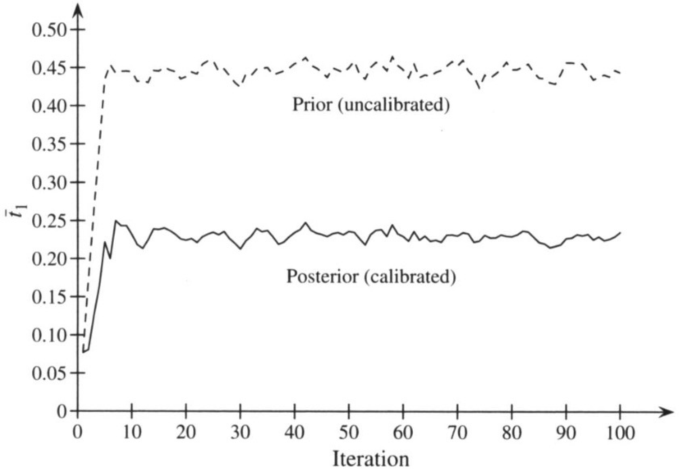


Figure 4 **Evolution of $\bar{t}_1$ for Two Routes Example**

5. Zurich Case Study

- 実データでのCalibration効果

設定

- Zurichの実ネットワーク
  - 60492 links, 24180 nodes
  - 中心30km以内を通った車両のみを考慮
  - 187484 agents(全体の10%)
  - 1時間を1タイムステップとして計算
- DTA simとcalibration
  - DTA simにはMATSim(http://www.matsim.org)を利用
  - calibrationはCadyts(http://transp-or.epfl.ch/cadyts/.)で実装
  - 500 iterationに20時間程度(64-bit, 2.67GHz, 10GB of RAM)

Figure 6 **Zurich Network**

Calibration結果

- 評価指標が、Calibrationにより低下
  - MWSE:観測分散を考慮した重みづけ誤差

$$MWSE = \left\langle \frac{(y_a(k) - x_a(k))^2}{2\sigma_a^2(k)} \right\rangle_{ak} \qquad (31)$$

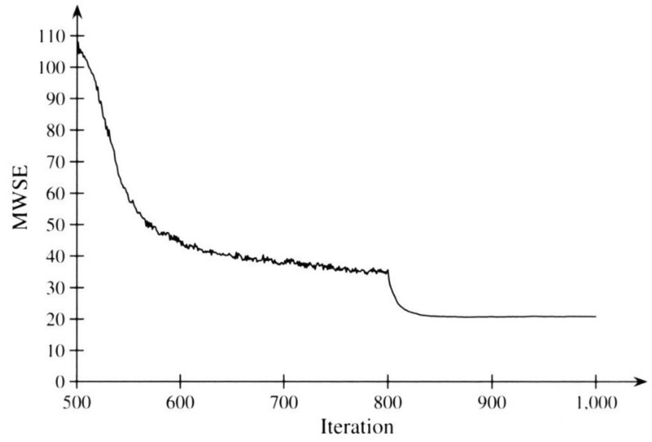


Figure 8 MWSE Using All Counting Stations

- Cross-Validationにより、観測値の再生成性も確認
  - 10データに分けて確認

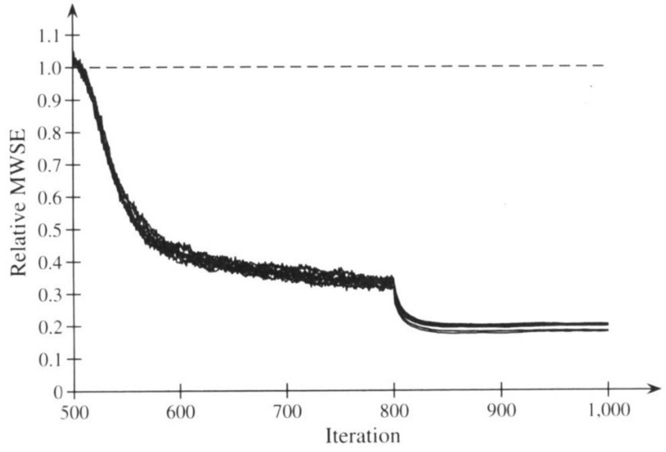


Figure 10 Validation Results—Measurement Reproduction

6. Summary and Outlook

まとめ

- 論文のまとめ
  - DTA simのcalibration
    - ベイズの設定
    - 集計データ(トラカン)を使いつつ、個人レベルのcalibration
  - 実データでの計算、softwareとしての実装
    - freeなので、皆さんも使ってみてください
- Future Works
  - demandとsupplyの両方のcalibration
  - 他種データの組み込み

参考

Gunnar Flötteröd, Michel Bierlaire, Kai Nagel. Baysian Demand Calibration for Dynamic Traffic Simulations. *Transportation Science*. Vol. 45 No. 4. pp541-561. 2011.

van Zuylen, H., L.G. Willumsen. The most likely trip matrix estimated from traffic counts. *Transportation Res. Part* B Vol. 14 No. 3. pp281-293. 1980.

https://ocw.hokudai.ac.jp/wp-content/uploads/2016/01/InformationTheory-2005-Slide-02.pdf

捕捉

- 交通でのエントロピーの考え方

ある交通量を達成する、個々のagentの選択パターン数を考える。

今回のケースでは、あるODペアnに限ると、$\frac{d_n!}{\prod_{i\in C_n} d_{ni}!}$が当てはまる。

全てのODペアに拡張すると、$P_0(\boldsymbol{d}) = \prod_{n=1}^{N} \frac{d_n!}{\prod_{i\in C_n} d_{ni}!}$と表せる。

N : #OD pairs
d_n: the largest possible #trips between OD pair n
C_n: the set of available paths that connect OD pair n
d_{ni}: #trips on path $i \in C_n$, where $d_n = \sum_{i\in C_n} d_{ni}$
p, P: probability density function, discrete probability function
$\boldsymbol{x}$: the vector of network conditions
$\boldsymbol{y}$: the traffic counts

これを最大にするような$\boldsymbol{d}$を求めるため、対数を取ってStirling's approximation : $\ln Z! \to Z \ln Z - Z$ $(for\ large\ Z)$を用いると、

$W_0(\boldsymbol{d}) = \ln P_0(\boldsymbol{d}) = \sum_{n=1}^{N} \sum_{i\in C_n} [d_n \ln d_n - d_{ni} \ln d_{ni}]$と変換できる。

d_nが所与であることを考慮して、$\sum_{n=1}^{N} \sum_{i\in C_n} -d_{ni} \ln d_{ni}$はエントロピーの形に帰着する。

捕捉

事前確率の計算

$\prod_{n=1}^{N} \frac{d_n!}{\prod_{i \in C_n} d_{ni}!}$から拡張し、事前情報$P_n(i)$を用いると、

$$P(\boldsymbol{d}) = \prod_{n=1}^{N} \left\{ \frac{d_n!}{\prod_{i \in C_n} d_{ni}!} \prod_{i \in C_n} \left(P_n(i|\boldsymbol{x}(\boldsymbol{d})) \right)^{d_{ni}} \right\}$$

N : #OD pairs
d_n: the largest possible #trips between OD pair n
C_n: the set of available paths that connect OD pair n
d_{ni}: #trips on path $i \in C_n$, where $d_n = \sum_{i \in C_n} d_{ni}$
p, P: probability density function, discrete probability function
$\boldsymbol{x}$: the vector of network conditions
$\boldsymbol{y}$: the traffic counts

事前エントロピー(←'the prior entropy function)の導出

Stirling's approximation : $\ln Z! \rightarrow Z \ln Z - Z$ ($for\ large\ Z$)を用いて、

$$W(\boldsymbol{d}) = \ln P(\boldsymbol{d})$$

$$= \sum_{n=1}^{N} \sum_{i \in C_n} [d_{ni} \ln P_n(i|\boldsymbol{x}(\boldsymbol{d})) - d_{ni} \ln d_{ni}] + \sum_{n=1}^{N} d_n \ln d_n$$

$$\text{s.t.} \quad d_n = \sum_{i \in C_n} d_{ni} \quad for\ all\ n = 1, 2, \ldots, N$$

これをラグランジュで解く(1階微分=0)と、式(1)が導出される。

捕捉

事後確率の計算

$$P(\boldsymbol{d}|\boldsymbol{y}) = \frac{P(\boldsymbol{y}|\boldsymbol{d})P(\boldsymbol{d})}{P(\boldsymbol{y})}$$
$$\propto P(\boldsymbol{y}|\boldsymbol{d})P(\boldsymbol{d})$$
$$= \int P(\boldsymbol{y}|\boldsymbol{x})P(\boldsymbol{x}|\boldsymbol{d})P(\boldsymbol{d})d\boldsymbol{x}$$

$= P(\boldsymbol{y}|\boldsymbol{x}(\boldsymbol{d}))P(\boldsymbol{d})$ ← $\boldsymbol{x}$は$\boldsymbol{d}$の関数

N : #OD pairs
d_n: the largest possible #trips between OD pair n
C_n: the set of available paths that connect OD pair n
d_{ni}: #trips on path $i \in C_n$, where $d_n = \sum_{i \in C_n} d_{ni}$
p, P: probability density function, discrete probability function
$\boldsymbol{x}$: the vector of network conditions
$\boldsymbol{y}$: the traffic counts

事後エントロピー (←'the posterior entropy function)の計算

Stirling's approximation : $\ln Z! \to Z \ln Z - Z$ (for large Z)

$$W(\boldsymbol{d}|\boldsymbol{y}) = \ln P(\boldsymbol{d}|\boldsymbol{y})$$
$$= \ln P(\boldsymbol{y}|\boldsymbol{x}(\boldsymbol{d})) + \ln P(\boldsymbol{d})$$
$$\text{s.t.}\ \ d_n = \sum_{i \in C_n} d_{ni} \ \ for\ all\ n = 1, 2, \ldots, N$$

これをラグランジュで解く(1階微分=0)と、式(4)~(6)が導出される。