

ActScape Lab.

都市経済学ワークショップ@京都大学

2024年3月15日 16:30-18:00

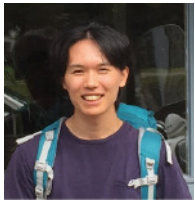
# ネットワーク上の行動モデルとその推定

大山 雄己

芝浦工業大学 土木工学科 准教授

Activity Landscape Design Lab.

[oyama@shibaura-it.ac.jp](mailto:oyama@shibaura-it.ac.jp)



**大山雄己** 博士 (工学)  
 芝浦工業大学 土木工学科 准教授

交通行動 = 「人の時空間的な移動」  
 の理論・分析手法開発を研究

- 2012 東京大学工学部都市工学科 卒業  
(ランドスケープデザイン)
- 2017 東京大学大学院都市工学専攻 学位取得  
(ネットワーク行動学)
- 2017-19 ポスドク@EPFL, Transport and Mobility Lab.  
(Discrete choice model / Network design)
- 2019-20 特任助教@東大西成総研  
(物流需要マネジメント)
- 2020- 芝浦工業大学 土木工学科  
都市環境設計研究室を主宰



▲複数都市でのビジョン策定業務・コンペ等



▲道後温泉飛鳥乃湯 中庭広場・街路基本設計 (2017.12竣工)

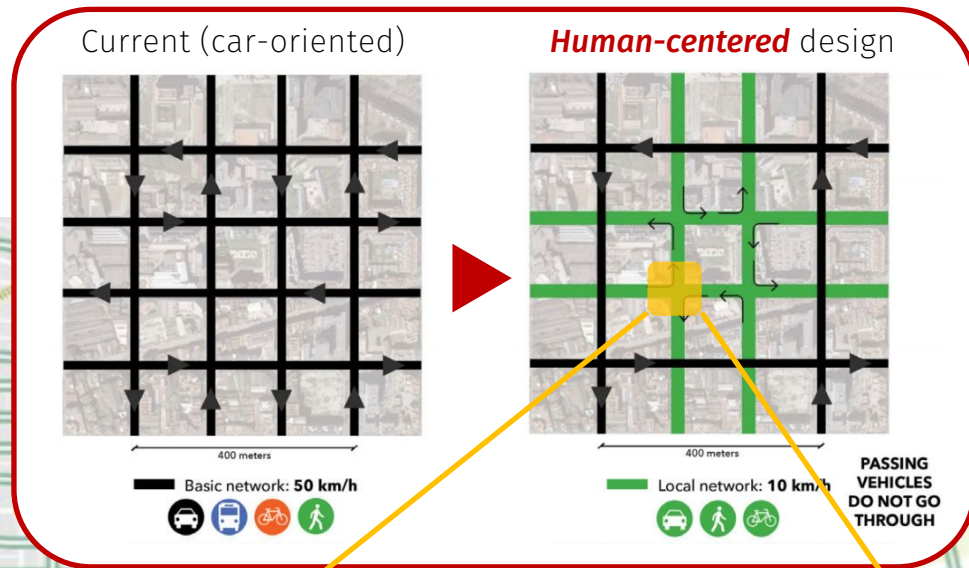


▲付知地域デザインミュージアム 広場設計 (2022.10竣工)

# “人間中心”の都市空間再設計

(例)

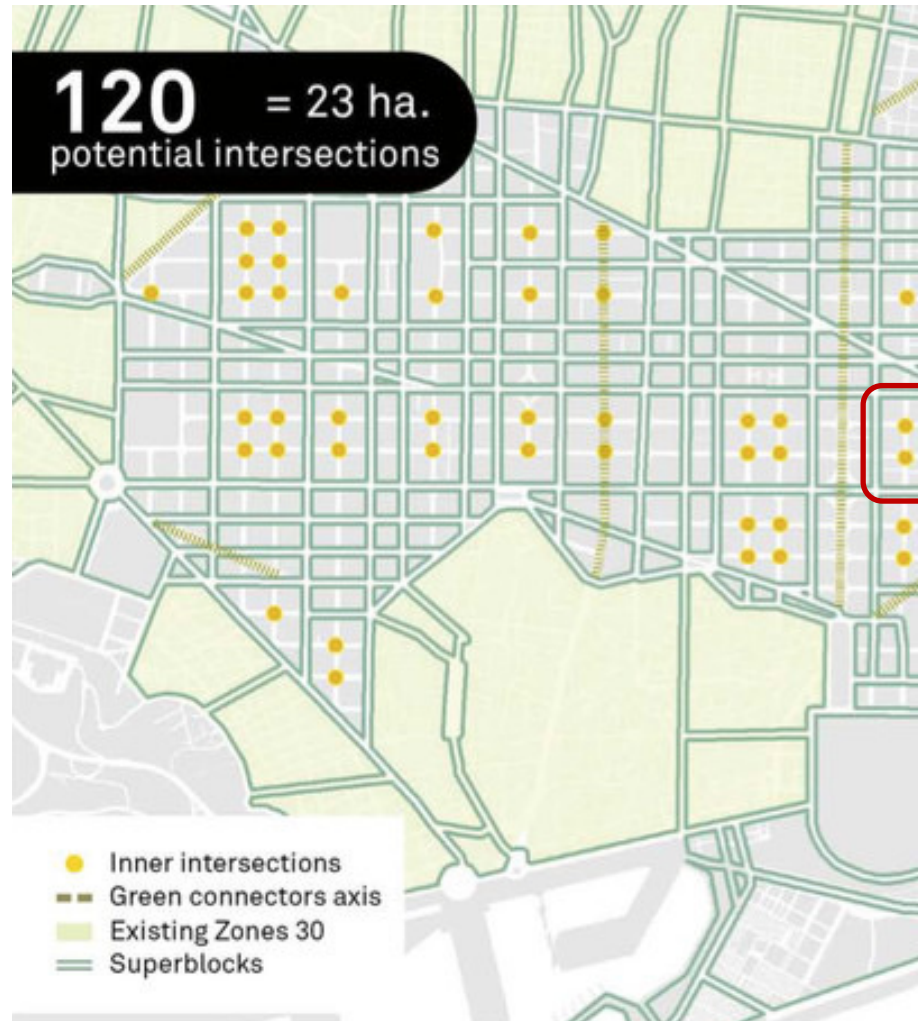
- 街路空間再編 (e.g. [Superblock](#); Google Sidewalk labs)
- 駐車場広場化 (e.g. Parklet; みんなの広場)



([City of the Future](#), by Valerio, P.)

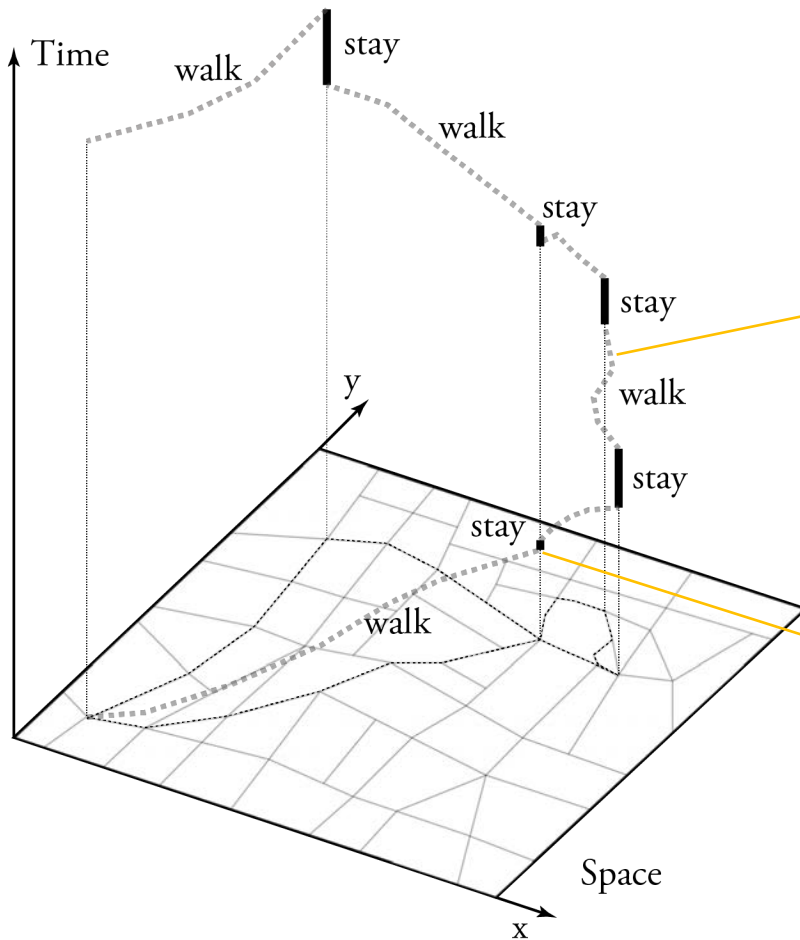


([Ajuntament Barcelona](#))





# 空間選択のシーケンスとしての回遊行動



移動の中で出会う空間・風景に反応



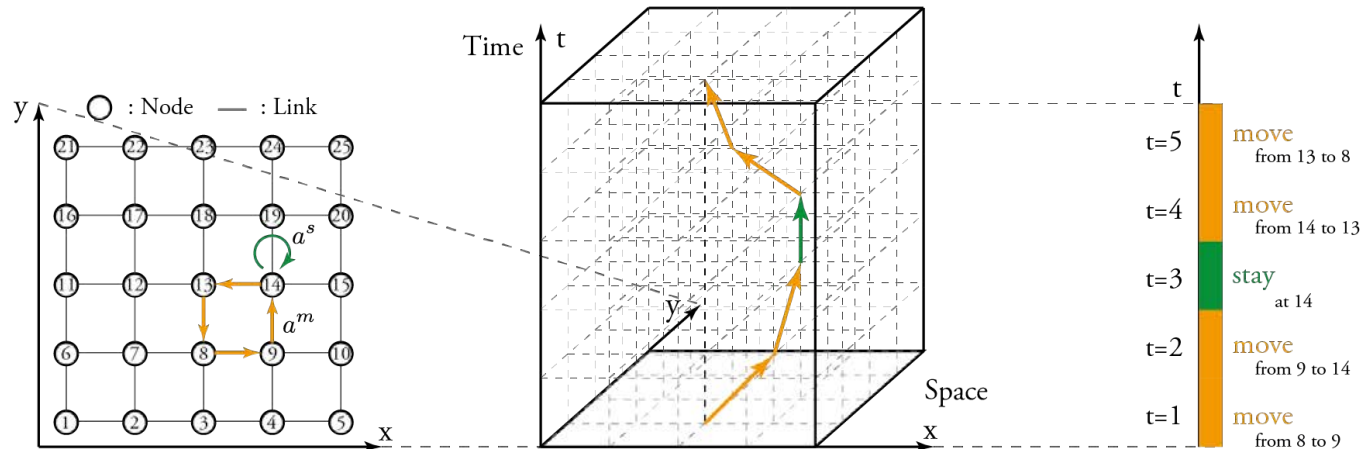
活動パターン = 時間軸上での空間(状態)選択のシーケンス

$$r = [a_1, a_2, \dots, a_J]$$

# 様々なネットワーク行動表現

Network path choice models can describe various types of behavior based on the **definition of states**.

Network	Application	State	Reference
General transportation network	Road, bicycle, walking	Link	Fosgerau+ (2013) Zimmerman+ (2017)
Multi-modal transportation network	MaaS, tourism	(Link, Mode)	De Freitas+ (2019) Tabuchi & Fukuda (2020)
Time-space network	Transit, time-dependent road NW	(Link, Time)	De Moraes Ramos+ (2020) Akamatsu+ (2023)
Activity network	Daily schedule, trip-chain	(Place, Time, Type, Stay/move)	Zimmerman et al. (2018) Oyama & Hato (2016)



Integrated choice of route, activity places and durations by activity path modeling (Oyama & Hato, 2016) 5

# 本日の内容

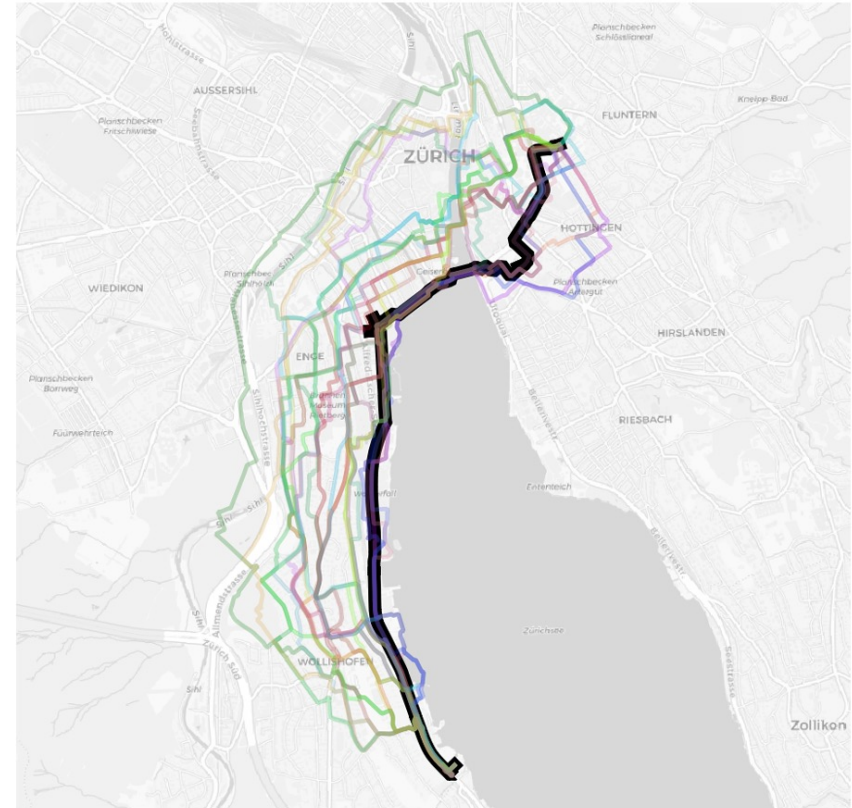
1. 交通分野における経路選択モデルの方法論的发展
2. マルコフ型経路選択モデルの推定問題とその解決
3. 局所的認知効用を考慮したモデリング

# 本日の内容

- 1. 交通分野における経路選択モデルの方法論的发展**
2. マルコフ型経路選択モデルの推定問題とその解決
3. 局所的認知効用を考慮したモデリング

# 経路選択行動モデル

- ネットワーク上である出発地から到着地まで (OD) を結ぶ経路のうち、旅行者がどの経路を選択するかをモデル化
- 交通分野においては、OD交通量のネットワークへの割当 (=交通量配分) モデルとして道路需要推計の基礎をなしている
- 政策上の主要な関心
  - 交通混雑の時空間分布予測
  - 時間価値や移動時間以外の要因 (政策変数) とのトレードオフ (Willingness-to-travel) の推計, それに基づく弾力性・政策効果評価
- ⇒ 主に**離散選択モデル**を用いた**個人の行動分析**とその**集計化による需要予測**が行われる



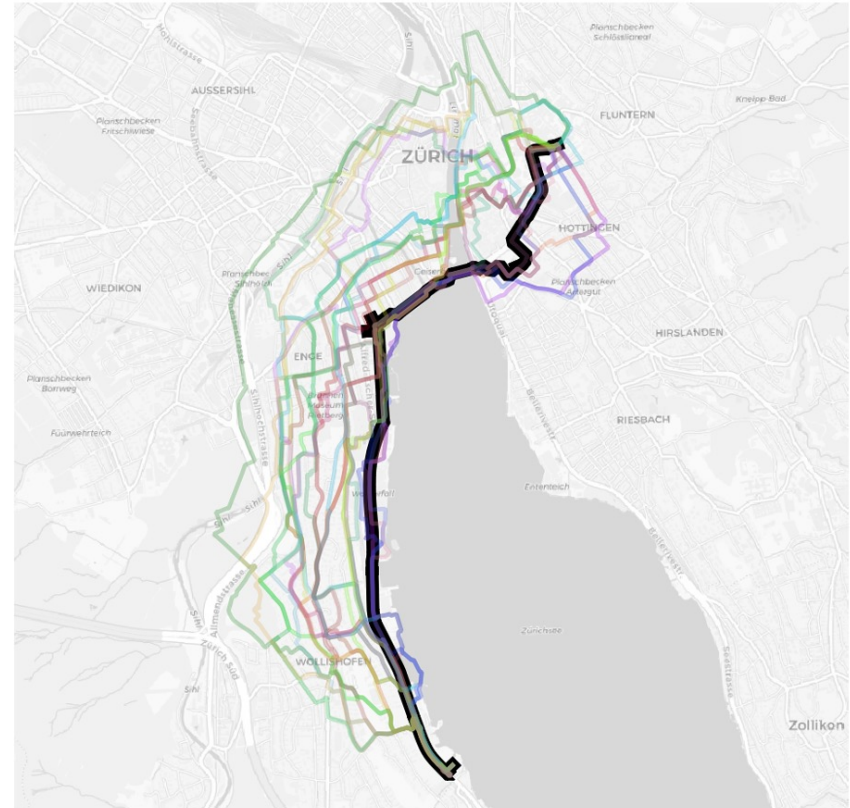
Isenschmid et al. (2022)



# 経路選択行動モデル

- 意思決定者 (旅行者)
  - トリップOD
  - 個人属性: 移動目的, 時間価値など
  - 情報の獲得: **事前 (静的)** or 逐次 (動的)
- 選択肢集合  $C_{od}$ 
  - 各経路選択肢の属性: 距離, 時間, 右左折回数, 路線特性, 沿道環境 etc
- 意思決定ルール
  - 最小費用基準 or **ランダム効用理論** or 摂動効用理論 etc
  - 選択タイミング: **事前** or 逐次

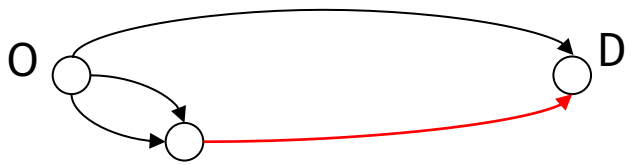
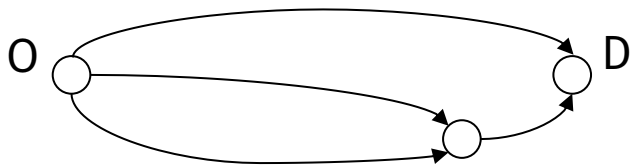
$$\max_{i \in C_{od}} U_{ni} \equiv V_{ni} + \varepsilon_{ni}$$



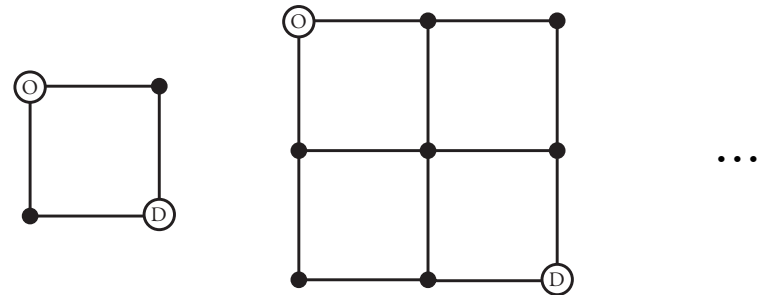
Isenschmid et al. (2022)

# 経路選択モデルのモデリング上の関心

- (1) 経路選択肢効用の相関
  - 重複率の高い経路同士で非観測の効用に相関がある
- (2) 経路選択肢集合の定義
  - ネットワーク規模に応じて候補経路数が爆発的に増加
- (3) 混雑のモデル化
  - 旅行者間の相互作用(行動の外部性)を明示的に考慮する必要がある



相関高い

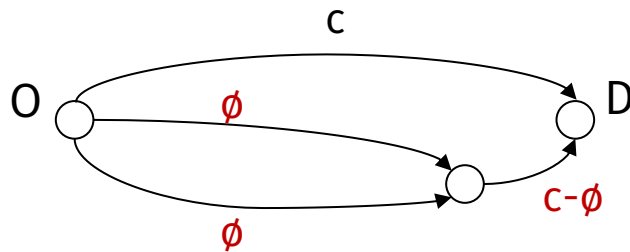


k	simple paths
1	2
2	12
3	184
4	8,512
5	1,262,816

k=10でも  
10<sup>25</sup>程度の規模

# 経路選択肢間の相関

- 基本的なモデルとして多項ロジットモデル (MNL) を想定



$$P_r = \frac{e^{-\beta c_r}}{\sum_{j \in C} e^{-\beta c_j}} = \frac{e^{-\beta c}}{3e^{-\beta c}} = \frac{1}{3}$$

$c, \phi, \beta$  に依らず等確率として予測

⇒ 特に  $\phi$  が小さい場合, **重複経路の確率を過大推計**

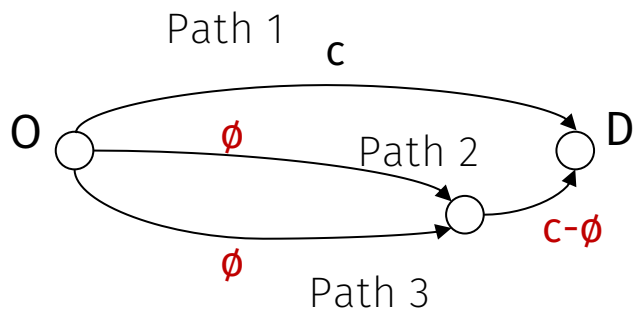
- 主に2種類のアプローチが提案
  - ペナルティ項の導入による **修正MNLモデル**
  - GEV (Generalized Extreme Value) アプローチ** (McFadden, 1978)

# 修正MNLモデル (Ben-Akiva & Bierlaire, 1999)

- 重複に伴う “**Path Size**” ペナルティ項を確定項に導入

$$V_r = -\beta_c c_r + \beta_{PS} \ln PS_r \quad \text{where} \quad PS_r = \sum_{ij \in r} \frac{c_{ij}}{c_r} \frac{1}{\sum_{r' \in C} \delta_{ij}^{r'}}$$

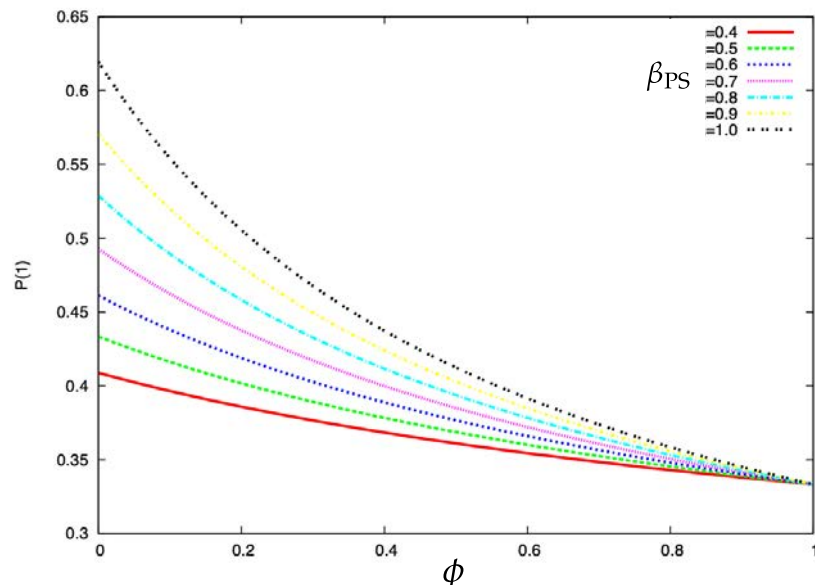
- 重複0のときにPSは最大値1をとる
- 重複長さが長く、そこで多くの経路と重複するほどPSが小さくなる = 効用が割り引かれる



$$PS_1 = \frac{c}{c} \cdot \frac{1}{1} = 1$$

$$PS_2 = PS_3 = \frac{c - \phi}{c} \cdot \frac{1}{2} + \frac{\phi}{c} \cdot \frac{1}{1} = \frac{1}{2} + \frac{\phi}{2c}$$

Bierlaire (2009)



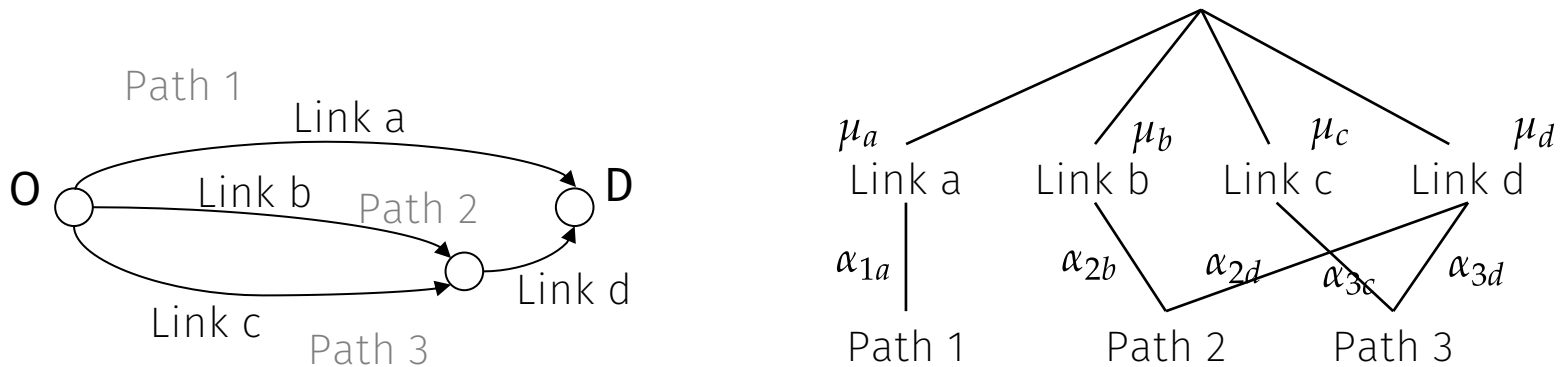


# GEVモデル (McFadden, 1978)

- 誤差相関構造を明示的にモデル化
- 経路選択モデルにおいては, Vovsha & Bekhor (1998) の Cross-Nested (Link-Nested) Logit モデルが代表的

$$P(r) = \sum_{m \in M} P(r|m)P(m) = \sum_{m \in M} \frac{(\alpha_{rm} e^{V_r})^{1/\mu_m}}{\sum_{j \in C_m} (\alpha_{jm} e^{V_j})^{1/\mu_m}} \cdot \frac{\left( \sum_{j \in C_m} (\alpha_{jm} e^{V_j})^{1/\mu_m} \right)^{\mu_m}}{\sum_{m \in M} \left( \sum_{j \in C_m} (\alpha_{jm} e^{V_j})^{1/\mu_m} \right)^{\mu_m}}$$

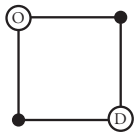
- ネスト  $m$  = 同一リンクの利用
- 各リンクへの帰属度合  $\alpha$  を重複率により表現  $\alpha_{rm} = L_m / L_r$



# 経路選択肢集合の特定

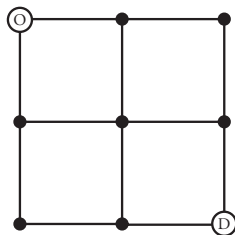
- 確率的選択モデルにおいては、選択確率 (合計1) を割り当てるための経路集合の定義が必要
- ネットワーク規模に応じて元となる候補集合が膨大になり、適切に定義することは困難
- 経路選択肢集合自体の観測も不可能
- アプローチ: **明示的** or **非明示的** な経路選択肢の列挙/生成

$k = 1$



$|\mathcal{C}| = 2$

$k = 2$



$|\mathcal{C}| = 12$

k	simple paths
1	2
2	12
3	184
4	8,512
5	1,262,816
6	575,780,564
7	789,360,053,252
8	3,266,598,486,981,640
9	41,044,208,702,632,496,804
10	1,568,758,030,464,750,013,214,100
11	182,413,291,514,248,049,241,470,885,236
12	64,528,039,343,270,018,963,357,185,158,482,118
13	69,450,664,761,521,361,664,274,701,548,907,358,996,488
14	227,449,714,676,812,739,631,826,459,327,989,863,387,613,323,440
15	2,266,745,568,862,672,746,374,567,396,713,098,934,866,324,885,408,319,028

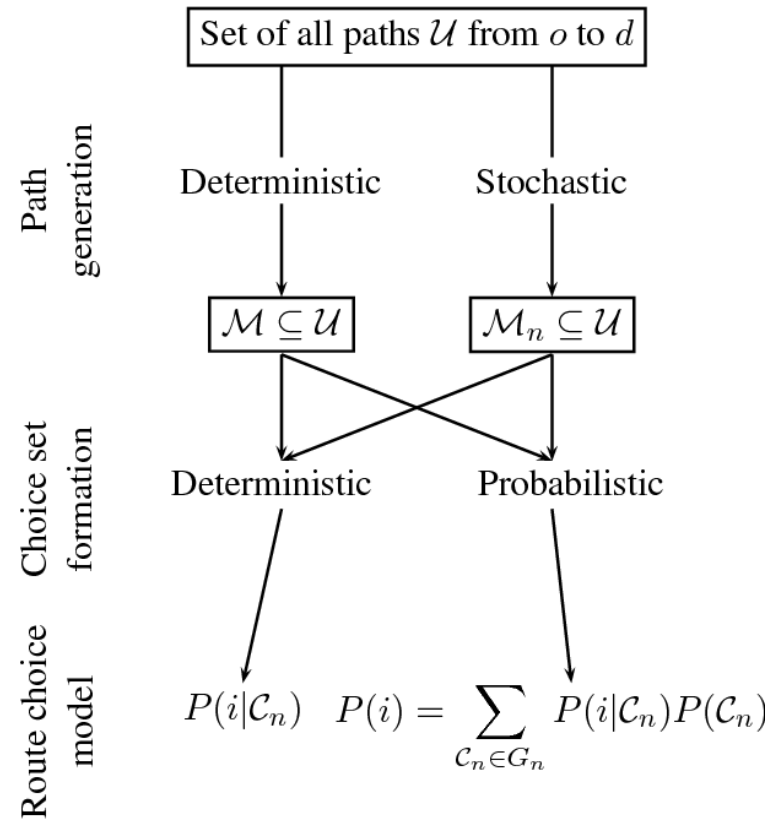
# 明示的な経路選択肢の列挙

## 決定論的経路集合の生成法

- k番目最短経路探索 (Eppstein, 1998)
- ラベリング法 (Ben-Akiva et al., 1984)
- 分枝限定法 (Hoogendoorn-Lanser, 2005)
- 幅優先探索 (Rieser-Schussler et al., 2013)

## 確率論的経路集合の生成法

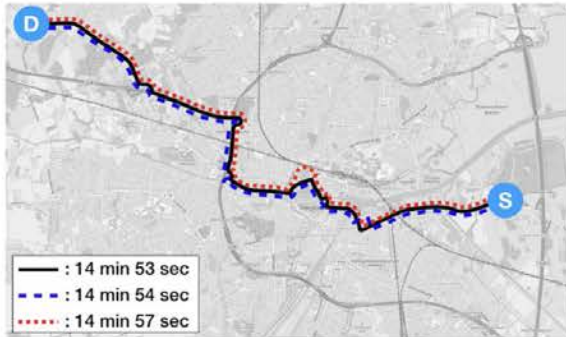
- 経路サンプリング法 (Fresinger et al., 2009; Flotterod and Bierlaire, 2013)
- 選択肢の認知確率をモデル化 (Cascetta and Papola, 2001)
- 深層学習 (VAE) を用いた方法 (Yao and Bekhor, 2022)



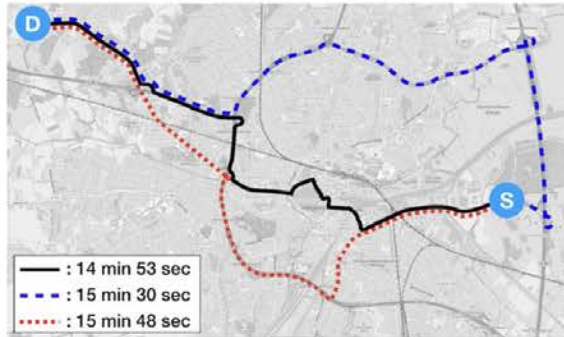
Frejinger and Bierlaire (2007)

# 明示的な経路選択肢の列挙

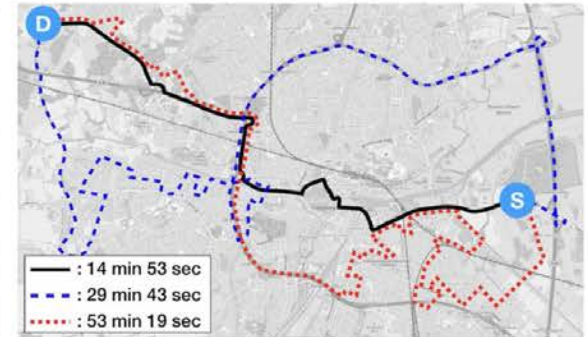
## K-Shortest Paths with Limited Overlaps (Chondrogiannis et al., 2020)



(a) K-Shortest Paths



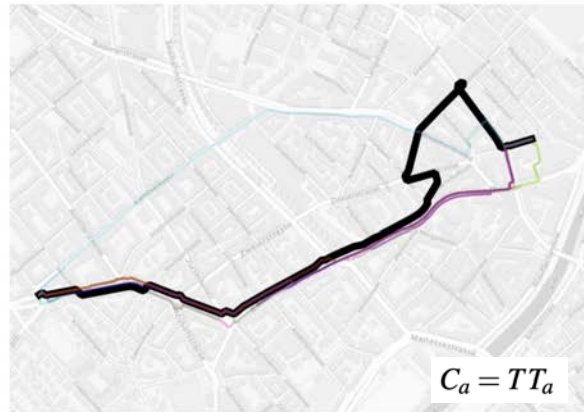
(b)  $k$ SPwLO



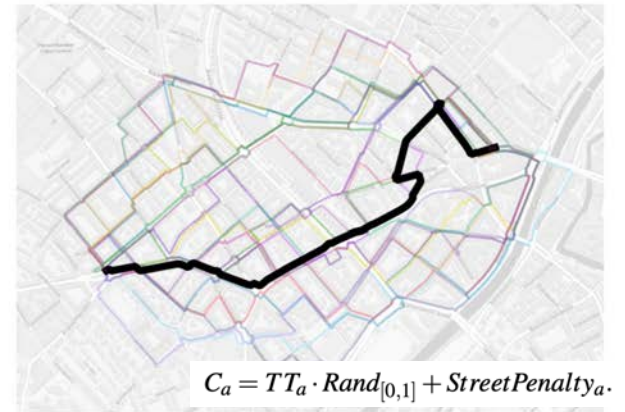
(c) Randomized search (1000 iter.)

## Breadth-first search on link elimination (Rieser-Schussler et al., 2013)

最小費用経路をベースにその一部を取り除き、代替経路を生成。コストの定義によりアルゴリズムをカスタマイズ可能。



(a) Choice set with basic cost function



(b) Choice set with extended cost function



# 明示的な経路選択肢の列挙 | 確率的アプローチ

Sampling of alternatives (Frejinger et al., 2009)

$$P(i|C_n) = \frac{q(C_n|i)P(i)}{\sum_{j \in C_n} q(C_n|j)P(j)} = \frac{e^{V_{ni} + \ln q(C_n|i)}}{\sum_{j \in C_n} e^{V_{nj} + \ln q(C_n|j)}}$$

$q(C_n|i)$  : 経路*i*が実際に選ばれた(観測された)ときの集合*C<sub>n</sub>*のサンプリング確率

Biased Random Walk による経路サンプリング

- Step 1: 下流リンクの重みを計算

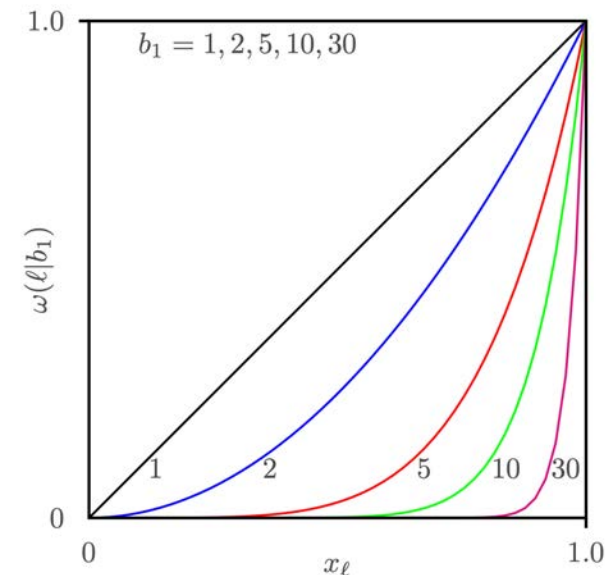
$$w(a|b_1) = 1 - (1 - x_a^{b_1}) \quad x_a = \frac{SP(a, d)}{c_a + SP(k, d)}$$

- Step 2: リンク選択確率

$$q(a|A(k), b_1) = \frac{w(a|b_1)}{\sum_{a' \in A(k)} w(a'|b_1)}$$

- Step 3: oからdに到達するまで繰り返す

$$q(i = \{o, a_1, \dots, a_l, d\}) = \prod_{l=1}^L q(a_l|A(a_{l-1}), b_1)$$



# 明示的な経路選択肢の列挙 | 確率的アプローチ

Sampling of alternatives (Frejinger et al., 2009)

1. R回の経路サンプリングを実施 (重複あり)
2. サンプリングされた経路 + 観測経路により集合  $C_n$  を形成

$$P(C_n) = \frac{R!}{\prod_j k_j!} \prod_j q(j)^{k_j} \quad k_j : \text{経路}j\text{がサンプリングされた回数}$$

経路*i*が観測された条件のもとでは

$$q(C_n|i) = \frac{R!}{(k_i - 1)! \prod_j k_j!} q(i)^{k_i - 1} \prod_j q(j)^{k_j} = P(C_n) \frac{k_i}{q(i)}$$

選択肢サンプリング (経路列挙) に対するUnbiasedな経路選択確率:

$$P(i|C_n) = \frac{e^{V_{ni} + \ln(\frac{k_i}{q(i)})}}{\sum_{j \in C_n} e^{V_{nj} + \ln(\frac{k_j}{q(j)})}}$$

# 明示的な経路選択肢の列挙 | 確率的アプローチ

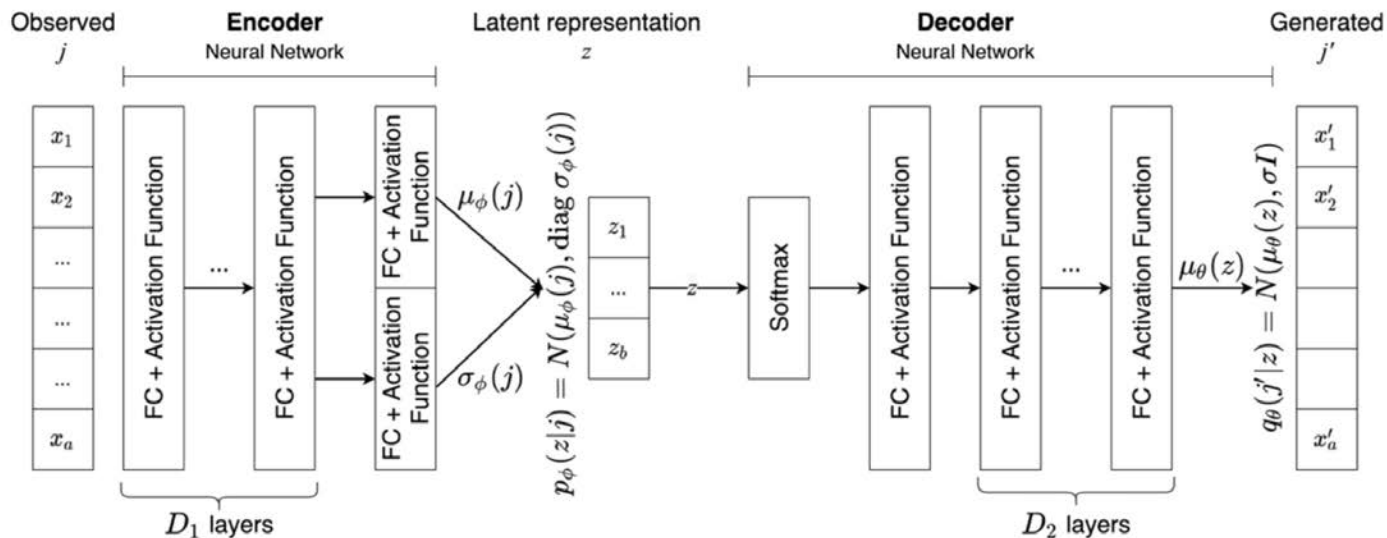
## Implicit Availability Perception (Cascetta et al., 2002)

「選択肢の認知確率」としてモデル化し導入

$$U_{ni} = V_{ni} + \ln q_n(i) + \varepsilon_{ni} \quad \tilde{q}(i) = \frac{1}{1 + e^{-\gamma x_i}} \quad x_i : \text{経路}i\text{の属性変数}$$

## Variational autoencoder approach (Yao and Bekhor, 2022)

$$\ln q_\theta(j) = \log \int \frac{p(z|j)}{p(z|j)} p(z) q_\theta(j|z) dz = \ln \mathbb{E}_{p(z|j)} \left[ \frac{p(z) q_\theta(j|z)}{p_\phi(z|j)} \right]$$

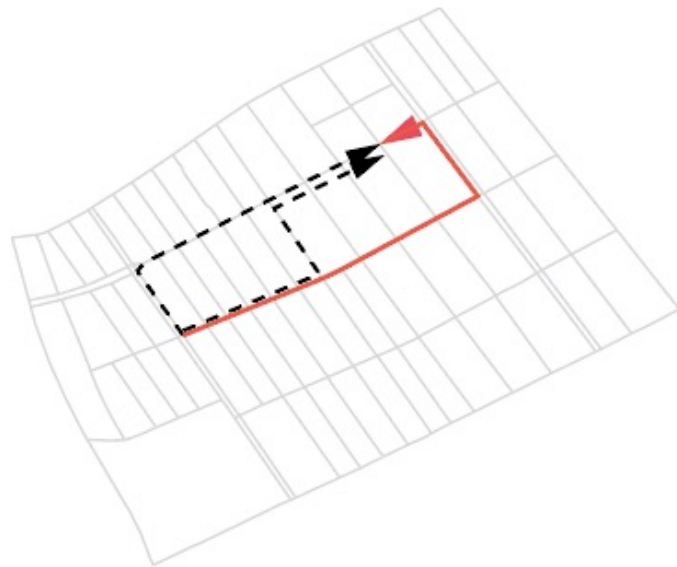


# 経路列挙の問題

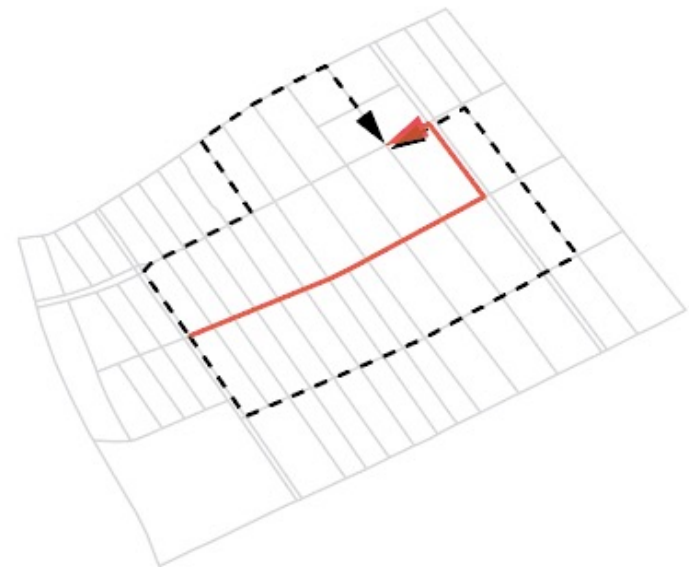
- 真の経路集合を知る（観測する）ことは不可能
- 推定の際に「観測された経路」は必ず経路選択肢集合に入れる必要がある

## 2つのInconsistency

- 推定結果が選択肢集合に依存
- 推定に用いる経路集合と予測に用いる経路集合の不一致



Distance **negatively** affects choice



Distance **positively** affects choice



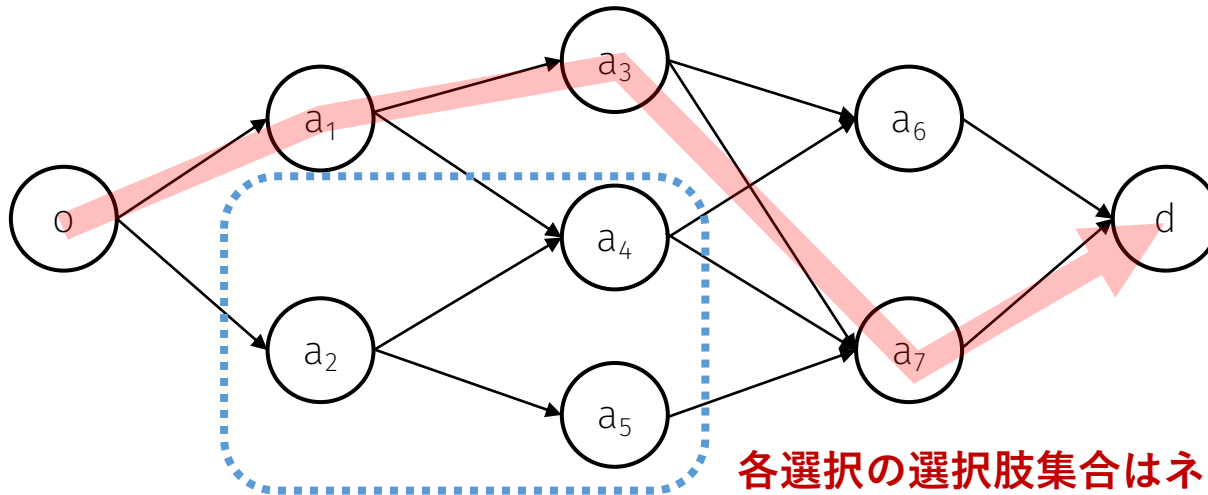
# Link-based approach

前提:

- ネットワーク (グラフ)  $G = (N, A)$
- 経路 = リンクの系列  $r = [ \underbrace{a_0}_{=0}, a_1, \dots, a_{J-1}, \underbrace{a_J}_{=d} ]$

Idea: 経路の選択を系列的なリンク選択として分解

$$P^{od}(r) = \prod_{j=1}^{J-1} p^d(a_{j+1}|a_j)$$



各選択の選択肢集合はネットワーク構造から明示的に与えられる

# Link-based approach

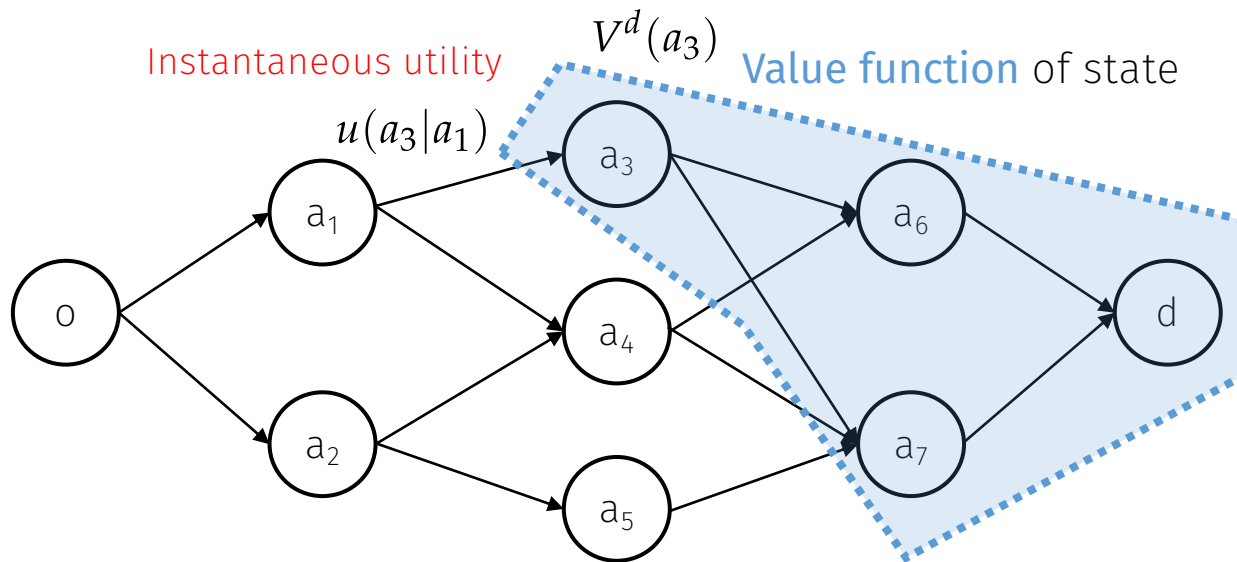
動的計画法 (DP) に基づく再帰的定式化:

- リンク選択に関する**瞬時効用**  $u(a|k)$
- 各リンクから目的地までの**価値 (最適値) 関数**  $V^d(a)$

## 最短 (最小費用) 経路問題

$$V^d(k) = \min_{a \in \mathcal{A}(k)} \{u(a|k) + V^d(a)\}$$

$$a^* = \arg \min_{a \in \mathcal{A}(k)} \{u(a|k) + V^d(a)\}$$



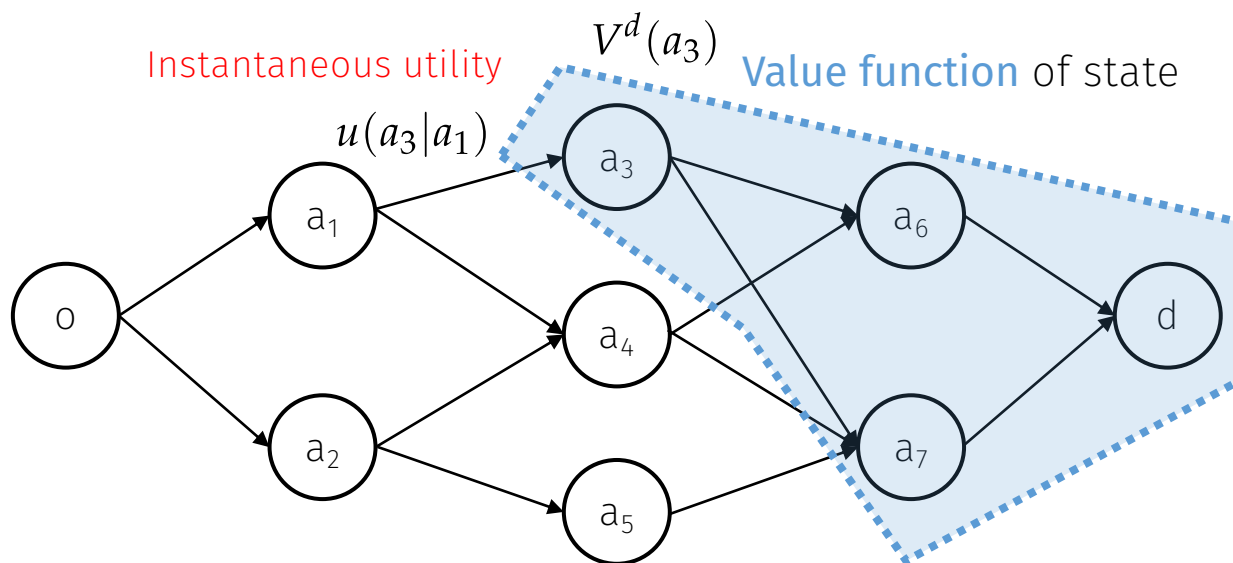
# Link-based approach

## ランダム効用最大化モデル (Dynamic discrete choice model)

$$V^d(k) = \mathbb{E} \left[ \max_{a \in \mathcal{A}(k)} \{u(a|k) + V^d(a)\} \right] \quad \text{where} \quad u(a|k) = v(a|k) + \varepsilon(a|k)$$

$$P(a|k) = \text{Prob} \left( u(a|k) + V^d(a) = \max_{a' \in \mathcal{A}(k)} \{u(a'|k) + V^d(a')\} \right)$$

※静的NW上の経路選択では、リンク選択が確定的に次の状態を決定



# Link-based approach

“Recursive” logit (RL) モデル (Akamatsu, 1996; Fosgerau et al., 2013)

誤差項  $\varepsilon$  にiidガンベル分布を仮定し，**線形方程式形へ帰着**：

$$V^d(k) = E \left[ \max_{a \in \mathcal{A}(k)} \{u(a|k) + V^d(a)\} \right] = \frac{1}{\mu} \ln \sum_{a' \in \mathcal{A}(k)} e^{\mu\{v(a'|k) + V^d(a')\}}$$
$$\Rightarrow \frac{e^{\frac{1}{\mu} V^d(k)}}{z_k} = \sum_{a \in \mathcal{A}(k)} \frac{e^{\frac{1}{\mu} \{v(a|k) + V^d(a)\}}}{M_{ka} z_a} \Rightarrow \boxed{z = \mathbf{M}z + \mathbf{b} \Leftrightarrow z = (\mathbf{I} - \mathbf{M})^{-1} \mathbf{b}}$$

選択確率は全経路を選択肢集合とする**経路選択モデルと等価**になる

$$P(a|k) = \frac{e^{\mu\{v(a|k) + V^d(a)\}}}{\sum_{a' \in \mathcal{A}(k)} e^{\mu\{v(a'|k) + V^d(a')\}}} = e^{\mu\{v(a|k) + V^d(a) - V^d(k)\}}$$
$$P(r) = \prod_{j=0}^J P(a_{j+1}|a_j) = e^{v(a_1|o) + V^d(a_1) - V^d(o)} \cdot e^{v(a_2|a_1) + V^d(a_2) - V^d(a_1)} \dots \cdot e^{v(d|a_J) + V^d(d) - V^d(a_J)}$$
$$= e^{V^d(d) - V^d(o)} e^{\sum_{j=0}^J v(a_{j+1}|a_j)} = \boxed{\frac{e^{V(r)}}{\sum_{r' \in U} e^{V(r')}}}$$

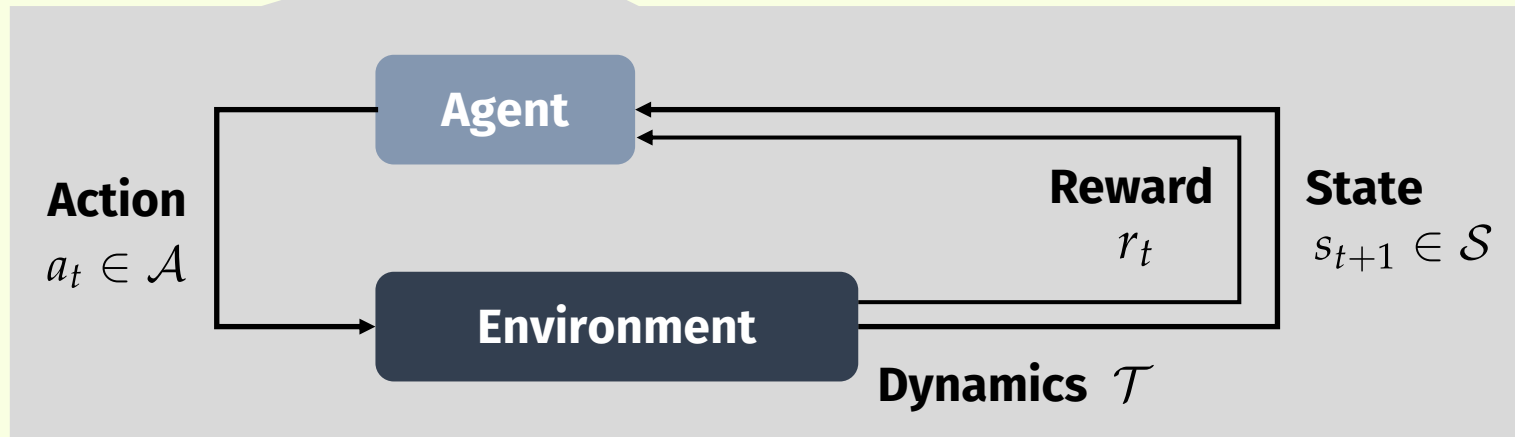
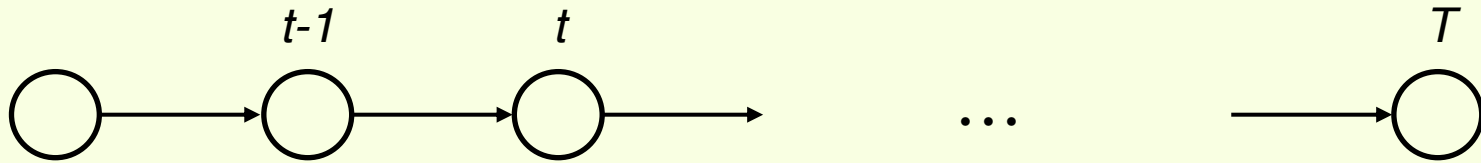
# 余談: Markov decision process (MDP)

Modeling **sequential decision-making** of an agent under uncertainty

**MDP** is defined as

$$\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{T}, r)$$

- $\mathcal{S}$  : State space
- $\mathcal{A}$  : Action space
- $\mathcal{T}$  : Dynamics (state transition)
- $r$  : Reward function



# 余談: Markov decision process (MDP)

## Agent's behavior

Decide **policy (action probability)** to maximize expected sum of rewards

$$\pi^* = \arg \max_{\pi} \mathbb{E}_{\pi} \left[ \sum_{t=0}^T \gamma^t r(s_t, a_t) \mid a_t = \pi(s_t) \right]$$

Recursive formulation:

$$V(s_t) = \max_{a_t \in \mathcal{A}(s_t)} \left\{ r(s_t, a_t) + \gamma \sum_{s_{t+1} \in \mathcal{S}} \mathcal{T}(s_{t+1} | a_t, s_t) V(s_{t+1}) \right\}$$

Different **reward specifications** lead to different MDP models (Mai & Jaillet, 2020)

- Regularized MDP  $\max_{\pi} \mathbb{E}_{\pi} \left[ \sum_{t=0}^T \gamma^t (v(s_t, a_t) + \phi(s_t)) \right]$
- **Entropy regularized MDP (Maximum causal entropy)**  $\max_{\pi} \mathbb{E}_{\pi} \left[ \sum_{t=0}^T \gamma^t (v(s_t, a_t) - \ln \pi(a_t | s_t)) \right]$
- **Dynamic discrete choice model**  $\max_{\pi} \mathbb{E}_{\pi} \left[ \sum_{t=0}^T \gamma^t (v(s_t, a_t) + \epsilon(a_t)) \right]$



# 余談: Dynamic discrete choice model (Rust, 1987)

## Connection of MDP with a **choice model**

- State  $\mathbf{s}$ : **situation** for the choice (e.g., attributes of alternatives/agents)
- Action  $\mathbf{a}$ : **choice** among alternatives
- Reward  $\mathbf{r}$ : **utility** that agent gains from choice
- Policy  $\boldsymbol{\pi}$ : **choice probability**

Under (additive) **random utility theory**:

$$\max_p \mathbb{E} \left[ \sum_{t=0}^T \gamma^t (v(a_t|s_t) + \epsilon(a_t)) \right] \quad \text{with} \quad r(a|s) = v(a|s) + \epsilon(a)$$

**Systematic & error components**

Value function (with known dynamics) is then

$$V(s_t) = \mathbb{E} \left[ \max_{a_t \in \mathcal{A}(s_t)} \left\{ v(a_t|s_t) + \epsilon(a_t) + \gamma \sum_{s_{t+1} \in \mathcal{S}} \mathcal{T}(s_{t+1}|a_t, s_t) V(s_{t+1}) \right\} \right]$$

# 余談: MDP in (static) networks

## Network route choice MDP models

- Often define **link** (node) **as state** so that action always and directly leads to the same next state (**deterministic dynamics**)
- Have **destinations** where agents terminate their actions (**episodic MDP**)

**Dynamic discrete choice model** (same as previous):

$$V(s) = \mathbb{E} \left[ \max_{a \in \mathcal{A}(s)} \left\{ v(a|s) + \epsilon(a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{T}(s'|a, s) V(s') \right\} \right]$$



**Recursive logit (route choice) model** (Fosgerau et al., 2013):

$$V(s) = \mathbb{E} \left[ \max_{a \in \mathcal{A}(s)} \{v(a|s) + \epsilon(a) + V(a)\} \right] = \ln \sum_{a \in \mathcal{A}(s)} e^{v(a|s) + V(a)} \quad (\text{i.i.d. Gumbel})$$

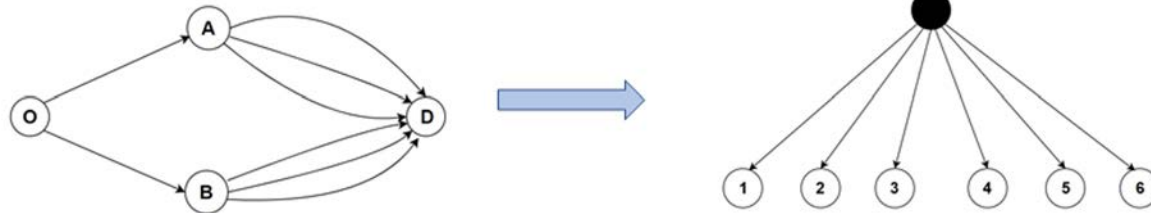
$$p(a|s) = \frac{e^{v(a|s) + V(a)}}{\sum_{a' \in \mathcal{A}(s)} e^{v(a'|s) + V(a')}} \quad (\text{action choice probability})$$

\*Equivalent to **Maximum Causal Entropy MDP** (Ziebart et al., 2008; Zierbart, 2010)

# Link-based approach

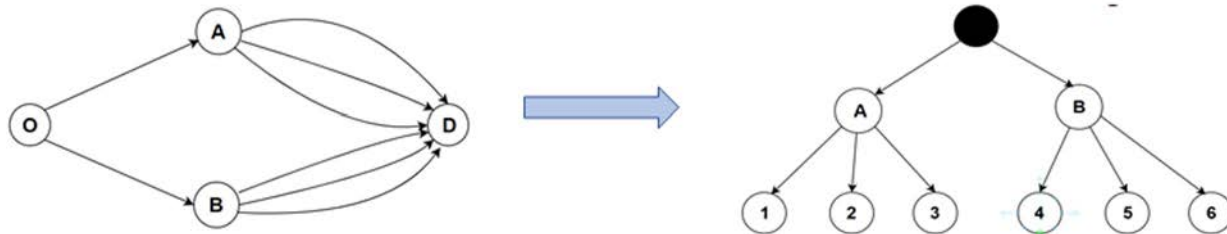
## RL モデル

$$V^d(k) = E \left[ \max_{a \in \mathcal{A}(k)} \{v(a|k) + V^d(a) + \varepsilon(a|k)\} \right]$$

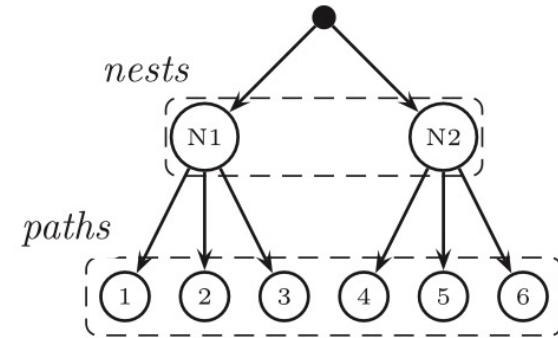
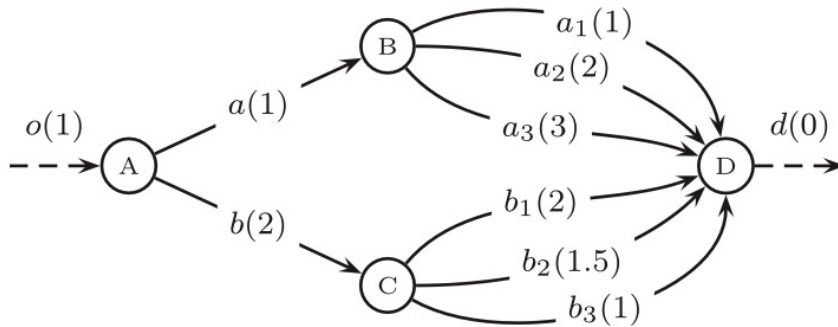
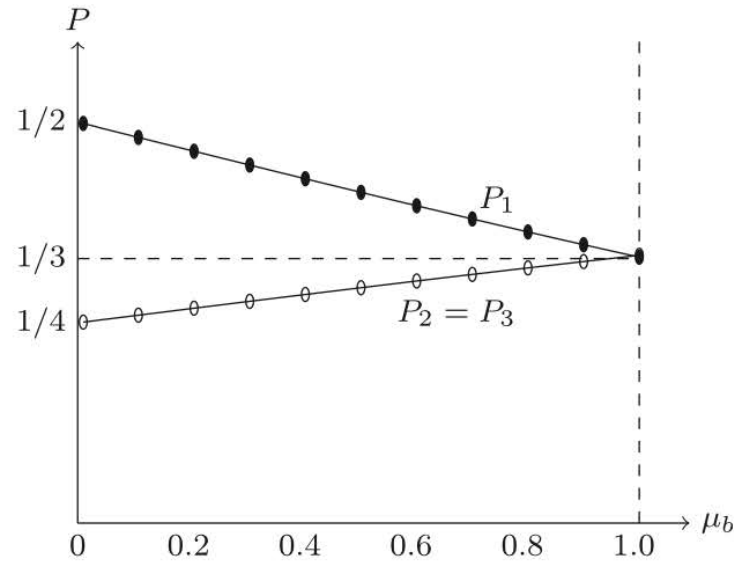
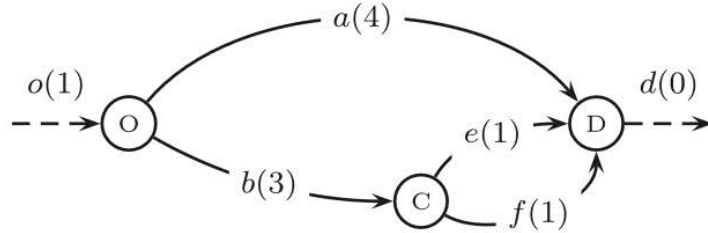


## Nested RL モデル (Mai et al., 2015)

$$V^d(k) = E \left[ \max_{a \in \mathcal{A}(k)} \{v(a|k) + V^d(a) + \mu(k)\varepsilon(a|k)\} \right]$$



# Nested RL モデル (Mai et al., 2015)

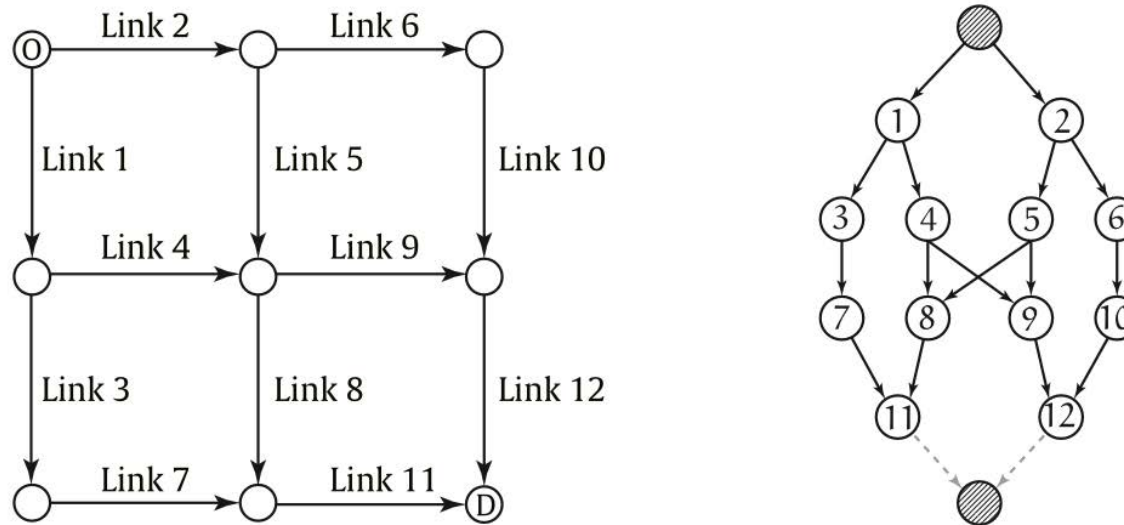


Paths	Original	Probabilities with link removed			
		$a_1$	$a_2$	$b_1$	$b_2$
1 : [o, a, a <sub>1</sub> , d]	0.54	-	0.65(+20%)	0.55(+1%)	0.56(+4%)
2 : [o, a, a <sub>2</sub> , d]	0.15	0.38(+151%)	-	0.16(+1%)	0.16(+4%)
3 : [o, a, a <sub>3</sub> , d]	0.04	0.11(+151%)	0.05(+20%)	0.05(+1%)	0.05(+4%)
4 : [o, b, b <sub>1</sub> , d]	0.02	0.05(+93%)	0.03(+15%)	-	0.03(+19%)
5 : [o, b, b <sub>2</sub> , d]	0.06	0.12(+93%)	0.07(+15%)	0.17(+6%)	-
6 : [o, b, b <sub>3</sub> , d]	0.17	0.33(+93%)	0.20(+15%)	0.18(+6%)	0.21(+19%)

# Link-based approach

## Network-GEV モデル (Oyama et al., 2022)

交通ネットワークそのものを相関を表すGEVネットワークとしてみなす



価値関数: 
$$V^d(k) = -\frac{1}{\mu} \ln \sum_{a \in \mathcal{A}(k)} \alpha_{ka}^d e^{\mu_k(v(a|k) + V^d(a))}$$

選択確率: 
$$P^d(a|k) = \frac{\alpha_{ka}^d e^{\mu_k(v(a|k) + V^d(a))}}{\sum_{a \in \mathcal{A}(k)} \alpha_{ka}^d e^{\mu_k(v(a|k) + V^d(a))}}$$

# 比較

## Path-based approach

- 重複を明示的に扱うことができる
- 「経路」ならではのNon link-additiveな属性を明示的に考慮できる
- アドホックな定義に依存しInconsistencyの問題が生じる
- 確率的な方法では計算負荷が大きい（シミュレーションが必要）

## Link-based approach

- Consistentな推定・予測
- DP理論 (MDP) に立脚し，情報科学的アプローチとも関連
- ネットワーク構造や効用サイズに依存した数値計算問題
- Link-additiveな属性しか扱うことができない



# 摂動効用 (Perturbed Utility) モデルとしての表現

## 最短 (最小費用) 経路選択問題

旅行者が**旅行費用を最小化**するように**通過リンクx**を決定する

$$\min_{x_{ij} \in \{0,1\}} \sum_{ij} c_{ij} x_{ij} \quad \text{s.t.} \quad \sum_{h \in B(i)} x_{hi} - \sum_{j \in F(i)} x_{ij} = \delta_{id} - \delta_{io}$$

## 最小”摂動”費用経路選択問題 (Fosgerau et al., 2022)

旅行者が**摂動旅行費用を最小化**するように**流率x**を選択する

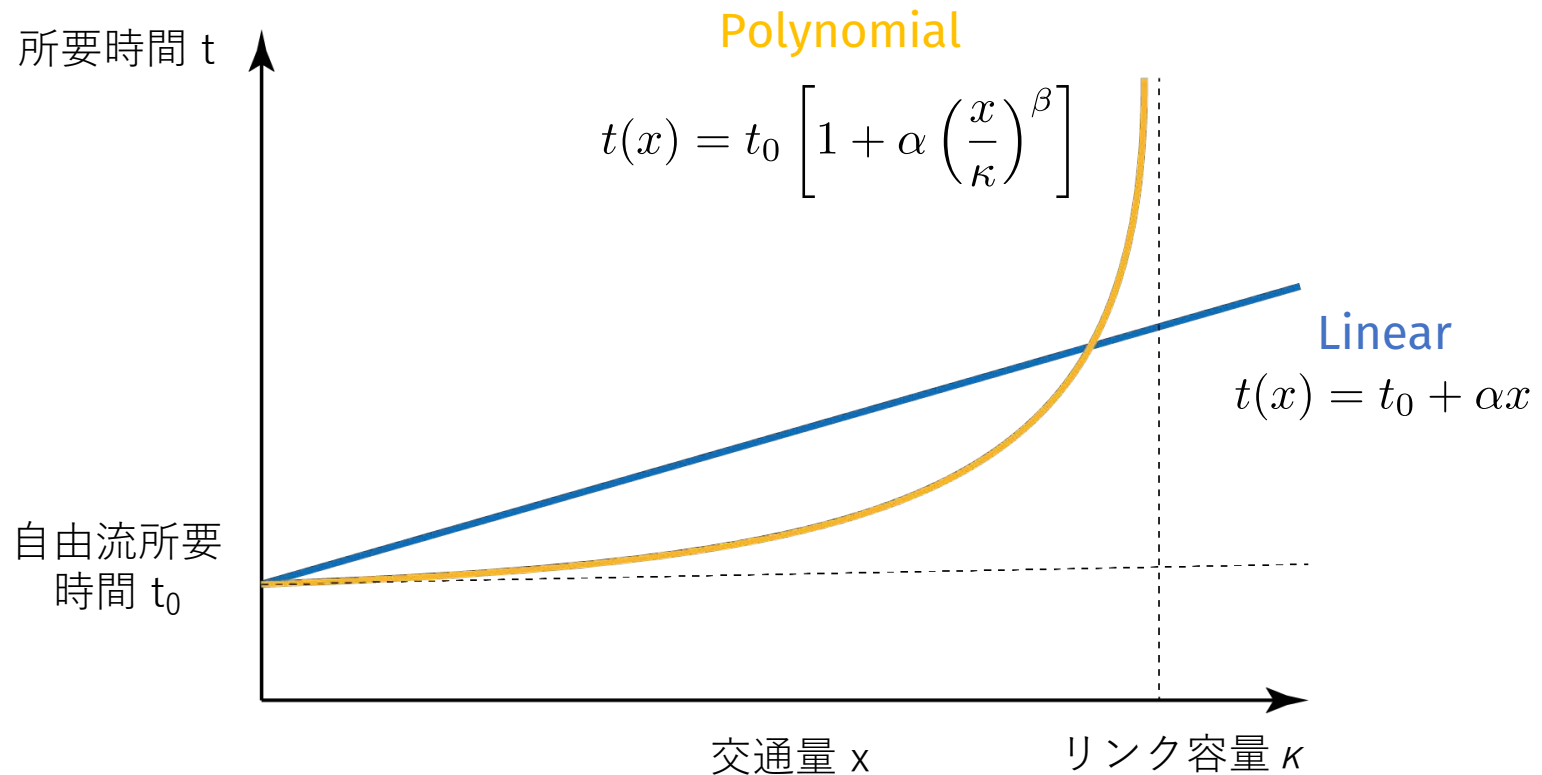
$$\min_{x_{ij} \geq 0} \sum_{ij} c_{ij} x_{ij} + F(x) \quad \text{s.t.} \quad \sum_{h \in B(i)} x_{hi} - \sum_{j \in F(i)} x_{ij} = \delta_{id} - \delta_{io}$$

※摂動関数Fがエントロピー関数のとき、ロジットモデルに一致

$$\min_{x_{ij} \geq 0} \sum_{ij} c_{ij} x_{ij} + \frac{1}{\theta} \sum_{ij} x_{ij} \ln \frac{x_{ij}}{z_i} \xrightarrow{\text{最適性条件}} p_{ij} = \frac{x_{ij}}{\sum_j x_{ij}} = e^{-\theta(c_{ij} + \mu_j - \mu_i)}$$

# 交通量配分モデル

- 交通分野において経路選択モデルは、OD交通量のネットワークへの割当 (= **交通量配分**) モデルとして道路需要推計の基礎をなしている
- 交通量配分モデルにおける渋滞 = **交通量の増加に伴う所要時間遅れ**
- リンクパフォーマンス関数  $c = t(x)$  により表現



# 交通量配分モデル

- リンクパフォーマンス関数の(一般的な)仮定
  - 交通量に対して狭義の単調増加関数
  - ヤコビ行列が対称行列(※非対称な相互作用が存在する場合, 一般に等価最適化問題が存在しない)
- **確率的交通量配分モデルの等価最適化問題**は, OD交通量 $q_{od}$ を所与とした摂動費用最小流問題として以下のように与えられる:

※このとき  $x$  はリンク交通量として求められる

$$\min_{x_{ij} \geq 0} \sum_{ij} \int_0^{x_{ij}} t_{ij}(\omega) d\omega + F(x) \quad \text{s.t.} \quad \sum_{h \in B(i)} x_{hi} - \sum_{j \in F(i)} x_{ij} = \delta_{id} \sum_0 q_{od} - \delta_{io} q_{od}$$

---

$$Ax = q$$

Network-GEV型の交通量配分モデル (Oyama et al., 2022)

$$F(x) \equiv \sum_{i \in N} \frac{1}{\theta_i} \sum_{j \in F(i)} x_{ij} \ln \frac{x_{ij}}{\alpha_{ji} z_i}$$

# 交通量配分モデルの解法

- 等価最適化問題の解析を通じて解の存在・一意性等の性質が証明可能
- 効率的な非線形最適化アルゴリズムが導出できる
  - 降下方向が反復 $m$ における所要時間 $t(x^{(m)})$ を所与としたNetwork loadingにより与えられる
- Primal アルゴリズム (未知変数: リンク交通量 $x$ )
  - **Method of Successive Average** (Sheffi and Powell, 1982)
    - ステップサイズを $1/m$ として与える
  - **Frank-Walfe / Partial Linearization methods** (Chen and Alfa, 1991; Patriksson, 1993)
    - 等価最適化問題に基づくステップサイズの最適化
- Dual アルゴリズム (未知変数: リンクコスト $c$ )
  - **Accelerated Gradient Descent (AGD)** (Oyama et al., 2022)
  - **Quasi-Newton AGD** (Fosgerau et al., 2023)
    - 双対問題が制約なし最適化 & 解空間が小さくなることに着目した効率的アルゴリズムの開発

# 交通量配分モデルの解法

## [NGEV-FD/P]

$$\min_{\mathbf{x} \geq 0} Z_P^{\text{FD}}(\mathbf{x}) \equiv C(\mathbf{X}) - \sum_{d \in \mathcal{D}} \hat{\theta}^d \cdot H^d(\mathbf{x}^d),$$

s. t.,

$$\mathbf{A}\mathbf{x}^d = \tilde{\mathbf{q}}^d,$$

$$\mathbf{X} = \sum_{d \in \mathcal{D}} \mathbf{x}^d,$$

where

$$C(\mathbf{X}) \equiv \oint_{\mathbf{X}} c(\mathbf{X}) d\mathbf{X}.$$

## [NGEV-FD/D]

$$\max_{\mathbf{c} \geq \mathbf{c}} Z_D^{\text{FD}}(\mathbf{c}) \equiv -C^*(\mathbf{c}) + \sum_{d \in \mathcal{D}} \mu^d(\mathbf{c}) \cdot \tilde{\mathbf{q}}^d,$$

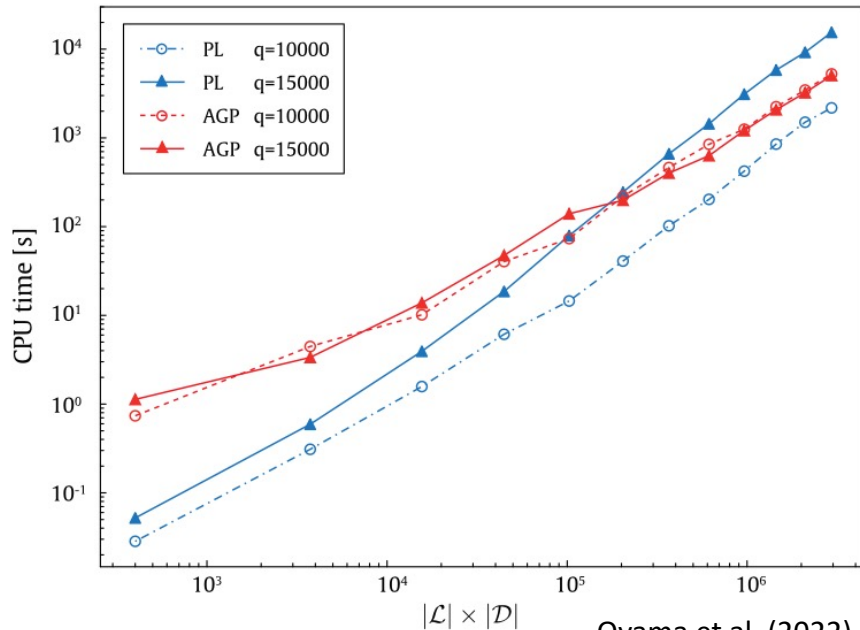
where  $C^*(\mathbf{c})$  is the conjugate dual (Legendre transform) of  $C(\mathbf{X})$ , defined as

$$C^*(\mathbf{c}) \equiv \max_{\mathbf{X}} [\mathbf{c} \cdot \mathbf{X} - C(\mathbf{X})] = \oint_{\mathbf{c}} \mathbf{c}^{-1}(\mathbf{c}) d\mathbf{c},$$

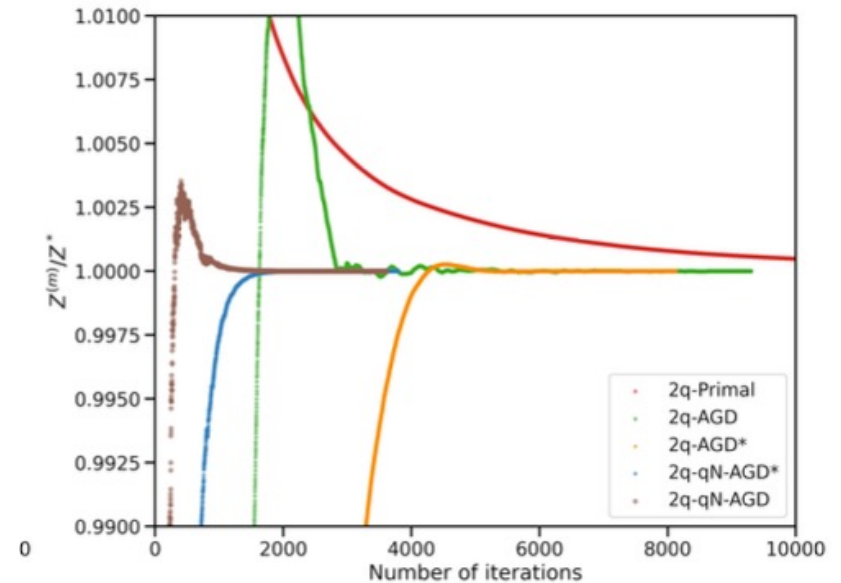
and  $\mathbf{c}^{-1}(\mathbf{c}) : \mathbb{R}_+^{|\mathcal{L}|} \rightarrow \mathbb{R}_+^{|\mathcal{L}|}$  is the inverse of the link cost function  $c(\mathbf{X})$ .

$$\nabla_{\mathbf{c}} Z_D^{\text{FD}}(\mathbf{c}) = \mathbf{X}(\mathbf{c}) - \mathbf{c}^{-1}(\mathbf{c})$$

Oyama et al. (2022)



Oyama et al. (2022)



Fosgerau et al. (2023)

# 本日の内容

1. 交通分野における経路選択モデルの方法論的发展
- 2. マルコフ型経路選択モデルの推定問題とその解決**
3. 局所的認知効用を考慮したモデリング

Yuki Oyama (2023) Capturing positive network attributes during the estimation of recursive logit models: A prism-based approach. *Transportation Research Part C: Emerging Technologies* 147: 104014



# 再起的構造を持つモデルの推定

## Nested Fixed Point (NFXP) algorithm (Rust, 1987)

### Outer algorithm

maximizes the likelihood function over the parameter space

$$\max_{\beta} LL(\beta, z(\beta)) \quad \text{e.g.} \quad \beta^{(m+1)} \leftarrow \beta^{(m)} - \mathbf{H}^{(m)} \frac{\partial LL(\beta^{(m)})}{\partial \beta}$$

### Inner algorithm

solves the value functions for each parameter value

$$z = \mathcal{T}_{\beta^{(m)}}(z) \quad z = (\mathbf{I} - \mathbf{M})^{-1} \mathbf{b} \quad \text{or} \quad z^{(l+1)} \leftarrow \mathbf{M}z^{(l)} + \mathbf{b} \\ \text{(until convergence)}$$

Inner algorithm は推定過程で探索される **全てのパラメータ候補**  $(\beta^{(0)}, \beta^{(1)}, \dots)$  **に対して解を持つ必要がある**

# 再起的構造を持つモデルの推定

The value functions are the solution of a **fixed point problem**:

$$\mathbf{z} = \mathcal{T}_\beta(\mathbf{z}) \quad \text{where} \quad \mathcal{T}_\beta(\mathbf{z}) \equiv \mathbf{M}\mathbf{z} + \mathbf{b} \quad \text{is the Bellman operator}$$

## 1. Hawkins-Simon / spectral radius condition

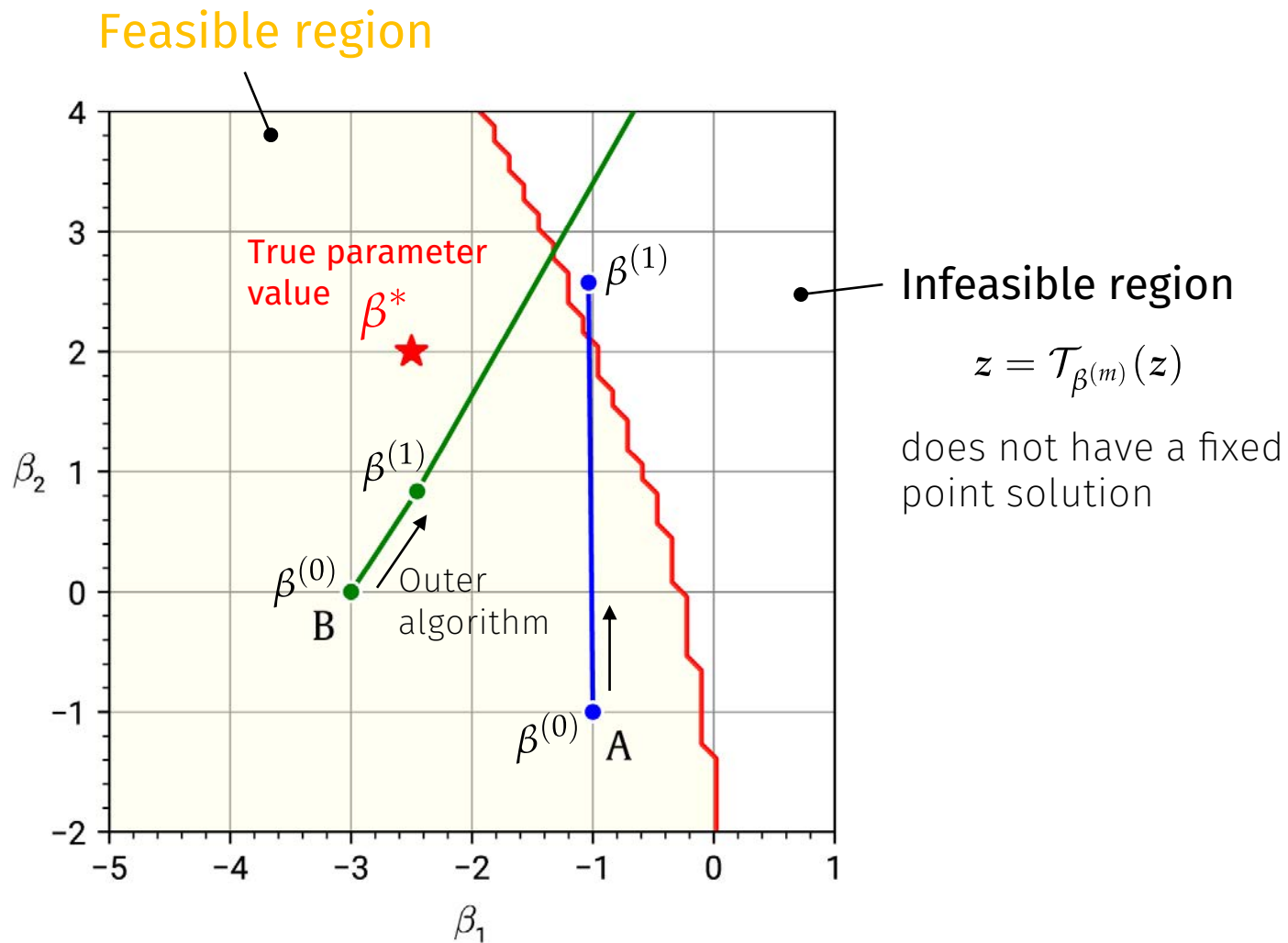
$(\mathbf{I} - \mathbf{M})$  is **invertible** when the maximum absolute of eigenvalues of  $\mathbf{M}$  is strictly less than one  $\rho(\mathbf{M}) < 1$

## 2. Utility condition (Mai and Frejinger, 2022)

$\mathcal{T}_\beta(\mathbf{z})$  is a **contraction mapping** if:

$$\sum_{a \in A(k)} M_{ka} < 1, \forall k \in A \quad \text{where} \quad M_{ka} \equiv e^{\frac{1}{\mu} v(a|k)}$$

# 再起的構造を持つモデルの推定



# 再起的構造を持つモデルの推定

1. 仮に真のパラメータ（や初期値）に対して価値関数が求解可能だとしても、**探索されるパラメータ全てがそうである保証はない**（1つでも計算可能条件を満たさなければ推定は失敗）
2. また、**真のパラメータ（未知）も計算可能条件を満たすとは限らない**

E.g., if there exist state pairs with **positive utilities**

$$M_{ka} = e^{v(a|k)} > 1, \text{ if } v(a|k) > 0$$

Then the following condition clearly does **NOT** hold:

$$\sum_{a \in A} M_{ka} < 1, \forall k \in A$$

# 再起的構造を持つモデルの推定

これまで：

計算条件を満たすための（アドホックな）効用関数の操作

- **負の効果をもつと期待される属性のみ**を導入 (e.g., 旅行時間, 右左折)
- 固定ペナルティ項の導入
- 初期パラメータ値を**十分大きな負の値**に設定



1. リンクベース経路選択モデル (RL) の**応用可能性を大きく制限**（正の効果を持ちうる属性・政策変数が評価できない）
2. これらの操作が**安定的な推定を可能にする理論的保証はない**

# 再起的構造を持つモデルの推定

価値関数の計算不能問題:

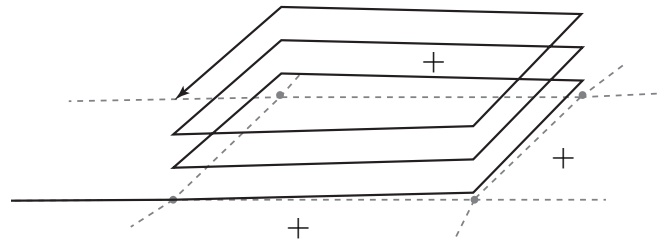
$$V^d(k) = \mathbb{E} \left[ \max_{r \in \mathcal{R}_{kd}} \{v(r) + \mu \epsilon(r)\} \right] = \ln \sum_{r \in \mathcal{R}_{kd}} e^{\frac{1}{\mu} v(r)}$$

Expectation of **ALL path utilities** connecting state  $k$  to destination  $d$

$$z = (\mathbf{I} - \mathbf{M})^{-1} \mathbf{b} = (\mathbf{I} + \mathbf{M} + \mathbf{M}^2 + \dots) \mathbf{b}$$

Power series as **implicit calculation process of the path utilities**

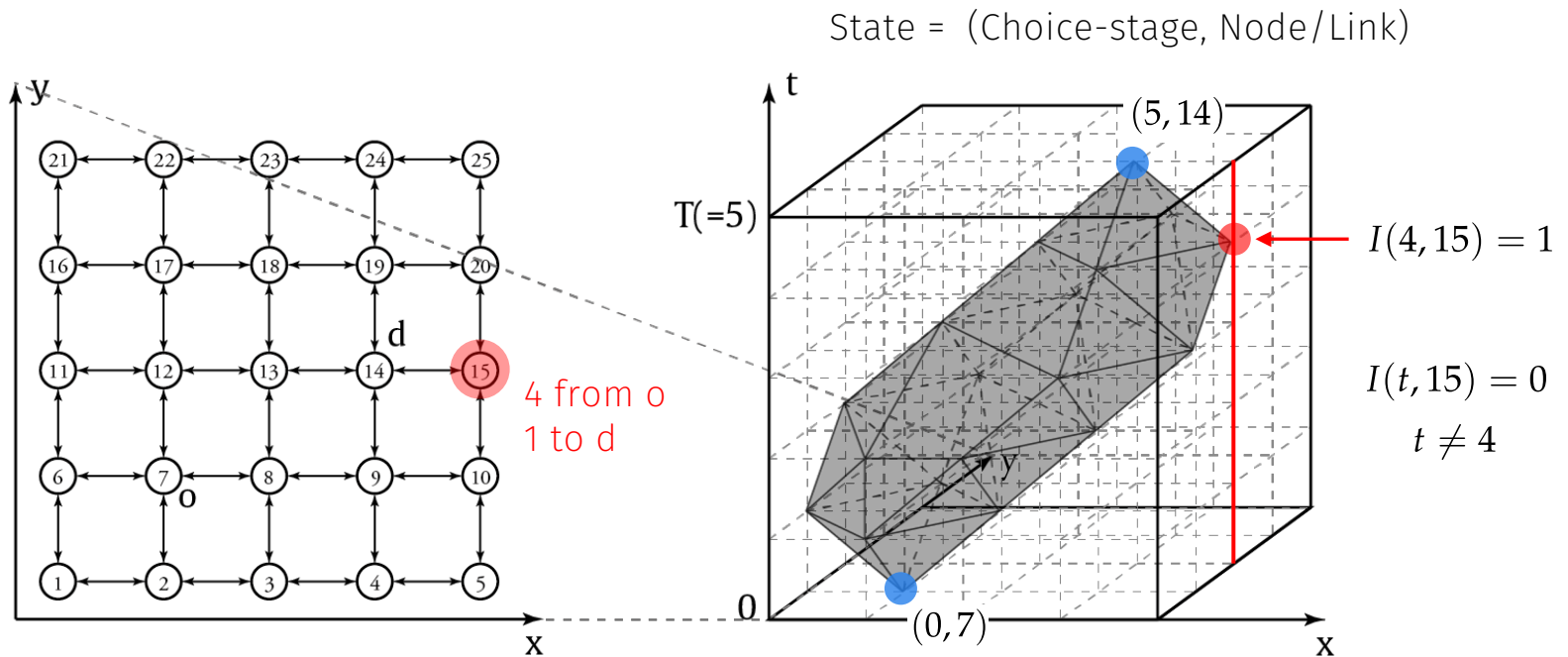
ex.) Unrealistic cycles that explosively gain utilities



**プリズム制約の導入により、行動論的に整合した形で効率的に非現実的な経路を制限し、数値計算問題を解消**



# プリズム制約を導入した時間構造化ネットワーク



1. Define a state-extended network based on **choice-stage**
2. Introduce the **choice-stage constraint**  $T$ , and evaluate **state existence conditions**  $I(t, k)$  based on the minimum number of steps from  $o$  and to  $d$
3. **States connection condition:**  $\Delta_t(a|k) = I(t, k)\delta(a|k)I(t + 1, a)$
4. The reduced set of states **forms into a prism.**

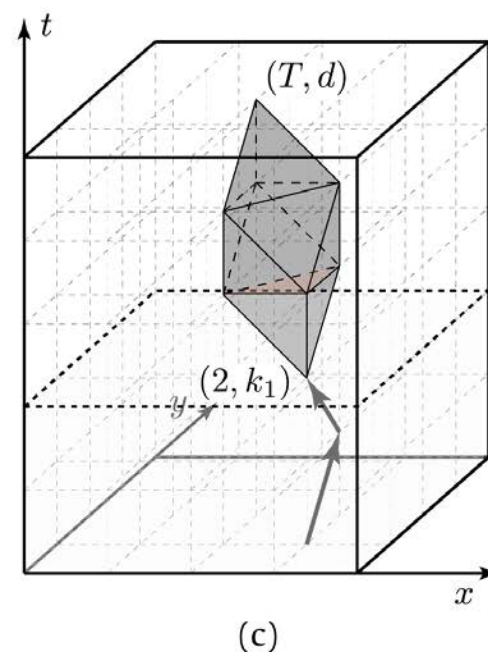
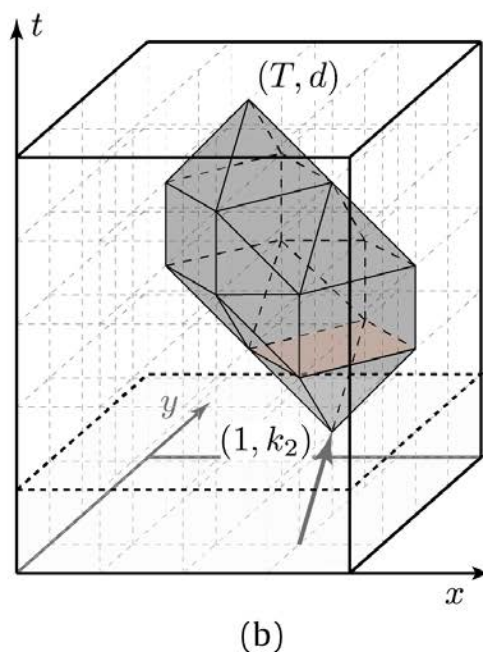
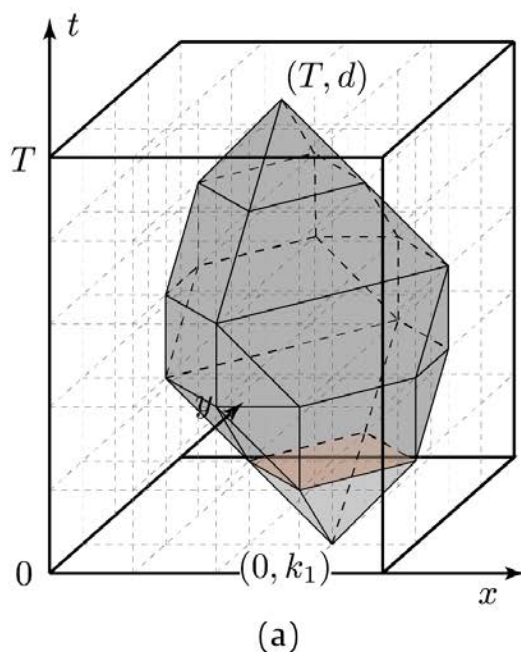
# Prism-constrained RL model

Choice-stage別に価値関数  $V(t, k)$  を再定義する:

$$z_{t,k} \equiv e^{\frac{1}{\mu} V(t,k)}$$

$$M'_{t,ka} \equiv \Delta_t(a|k) e^{\frac{1}{\mu} v(a|k)}$$

$$e^{\frac{1}{\mu} V^d(t,k)} = \sum_{a \in A(k)} \Delta_t^d(a|k) e^{\frac{1}{\mu} \{v(a|k) + V^d(t+1,a)\}} \Leftrightarrow z_t^d = \mathbf{M}'_t{}^d z_{t+1}^d + \mathbf{b}^d$$



$$V(t, k) \neq V(t', k)$$

\*ex)  $V(0, k_1) \neq V(2, k_1)$

- **各状態**（から目的地までに利用可能な全ての経路）**の評価関数**
- 上位のChoice-stageでは、目的地に効率的に向かう行動の選択確率が高くなる

# Prism-constrained RL model

T回の後ろ向き帰納法により求解が可能

- Initialize:  $t = T, z_{T,d} = 1$  and  $z_{t,k} = 0, \forall (t,k) \neq (T,d)$
- Set  $t := t - 1$  and update the value function by

$$z_t \leftarrow \mathbf{M}'_t z_{t+1} + \mathbf{b}$$

- If  $t = 0$ , finish the computation. Otherwise, go back to the previous step.

→ 効率的かつ、**パラメータ値や効用サイズによらず**  
常に価値関数が（唯一に）求められる

\*As long as  $T$  is finite,

$$z_0 = \mathbf{M}'_0 z_1 + \mathbf{b} = \mathbf{M}'_0 (\mathbf{M}'_1 z_2 + \mathbf{b}) + \mathbf{b} = \dots = (\mathbf{I} + \sum_{r=0}^{T-1} \prod_{s=0}^r \mathbf{M}'_s) \mathbf{b}.$$

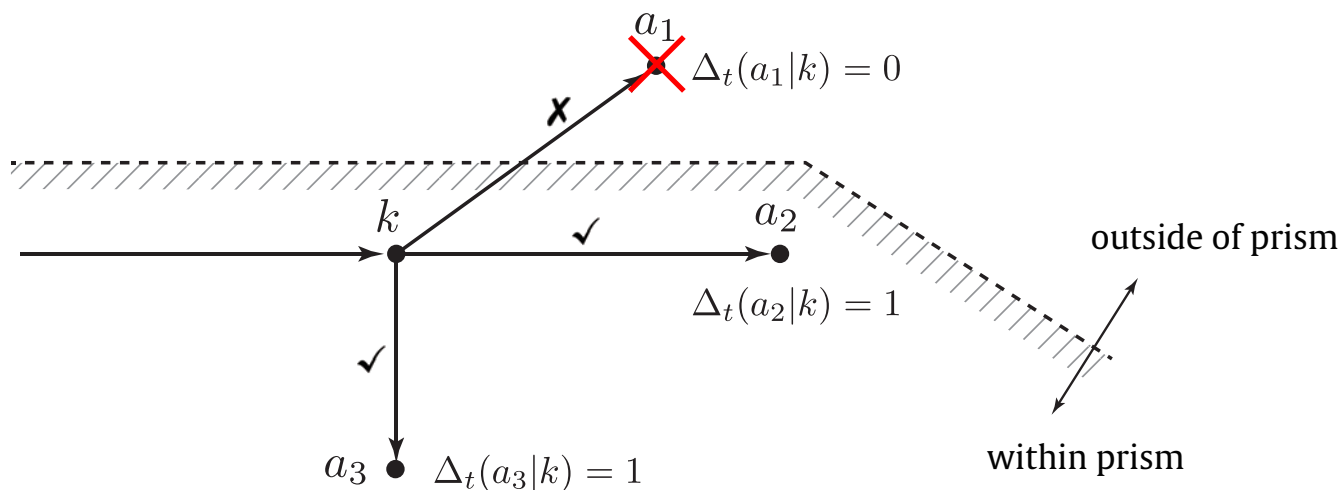
is theoretically upper bounded by a real vector because, for finite  $t$ ,

$$\prod_{s=0}^t \mathbf{M}'_s = \mathbf{M}'_0 \mathbf{M}'_1 \dots \mathbf{M}'_t \leq \mathbf{M}^t \leq \mathbf{C}$$

# Prism-constrained RL model


## Choice probability

$$p_t^d(a|k) = \frac{\Delta_t^d(a|k) e^{\frac{1}{\mu} \{v(a|k) + V^d(t+1, a)\}}}{\sum_{a' \in A(k)} \Delta_t^d(a'|k) e^{\frac{1}{\mu} \{v(a'|k) + V^d(t+1, a')\}}}$$



# Maximum likelihood estimation

## Translation of path observations

 Original:  $\sigma_n = [k_0, \dots, k_{J_n}]$

**Translated:**  $\sigma_n^* = [(0, k_0), \dots, (J_n, d_n), (J_n + 1, d_n), \dots, (T, d_n)]$

Stay at destination after the arrival

(\*Set T so that  $T \geq \max_n J_n$ )

## Likelihood function

$$\begin{aligned} LL(\beta; \sigma^*) &\equiv \log \prod_{n=1}^N P(\sigma_n^*) \\ &= \sum_{n=1}^N \sum_{t=0}^{T-1} \log p_t^{d_n}(k_{t+1}|k_t) \\ &= \frac{1}{\mu} \sum_{n=1}^N \sum_{t=0}^{T-1} \left[ v(k_{t+1}|k_t) + V^{d_n}(t+1, k_{t+1}) - V^{d_n}(t, k_t) \right] \end{aligned}$$

# 数値実験

## 検証

1. **正の効果をもつ属性が存在するとき**に，Prism-RLモデルが実際に推定可能であり，RLモデルの**真のパラメータを再現できるか？**
2. **初期値に対してどの程度影響を受けるか？**

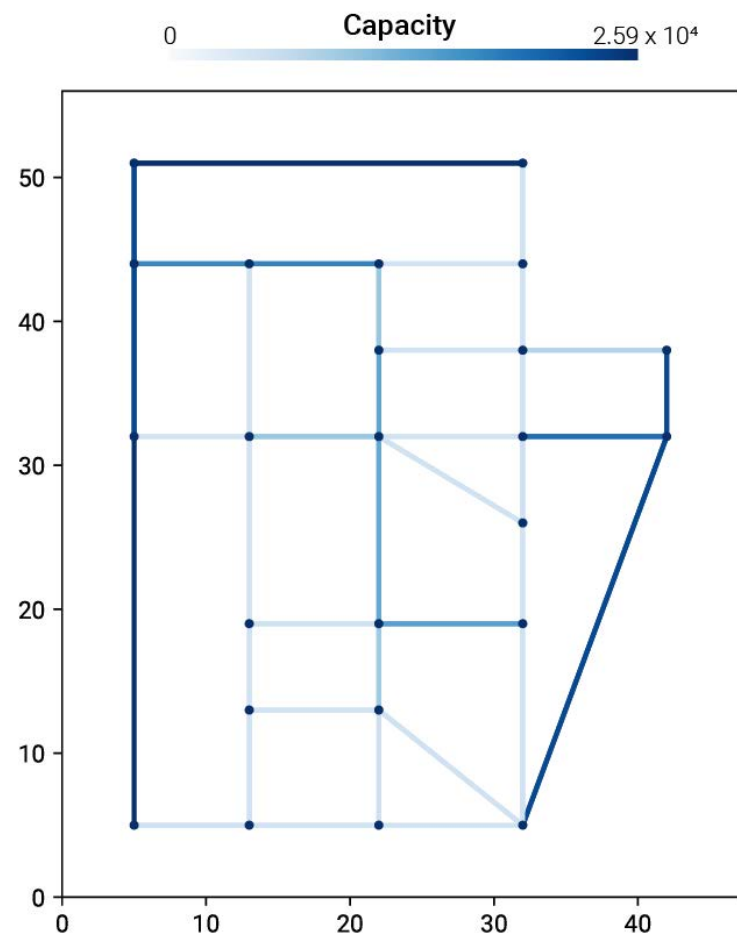
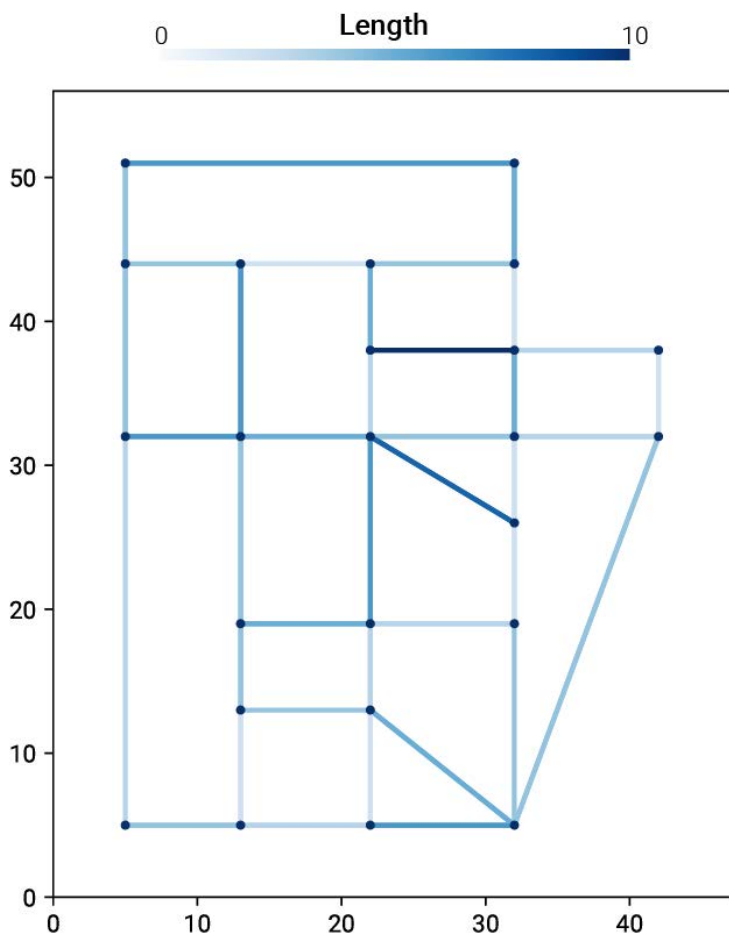
## 実験条件

- Sioux Falls ネットワーク（交通配分のベンチマークとしてよく用いられる）
- 真のパラメータを知っている状態で，（元の）**RLモデルのシミュレーションにより経路データを生成** (24 OD x 1000 samples)
- **同じデータを用いて，RLモデル・Prism-RLモデルの両方を推定**
- Prism-RLモデルの設定
  - 目的地ごとに状態ネットワークを定める
  - $T = 15$  (\*did not affect the results)

# 数値実験

$$v(a|k) = \beta_{\text{len}} \text{Length}_a + \beta_{\text{cap}} \text{Capacity}_a - 10 \text{Uturn}_{a|k}$$

$$(\beta_{\text{len}} < 0; \beta_{\text{cap}} > 0)$$



# 数值実験 | 推定結果

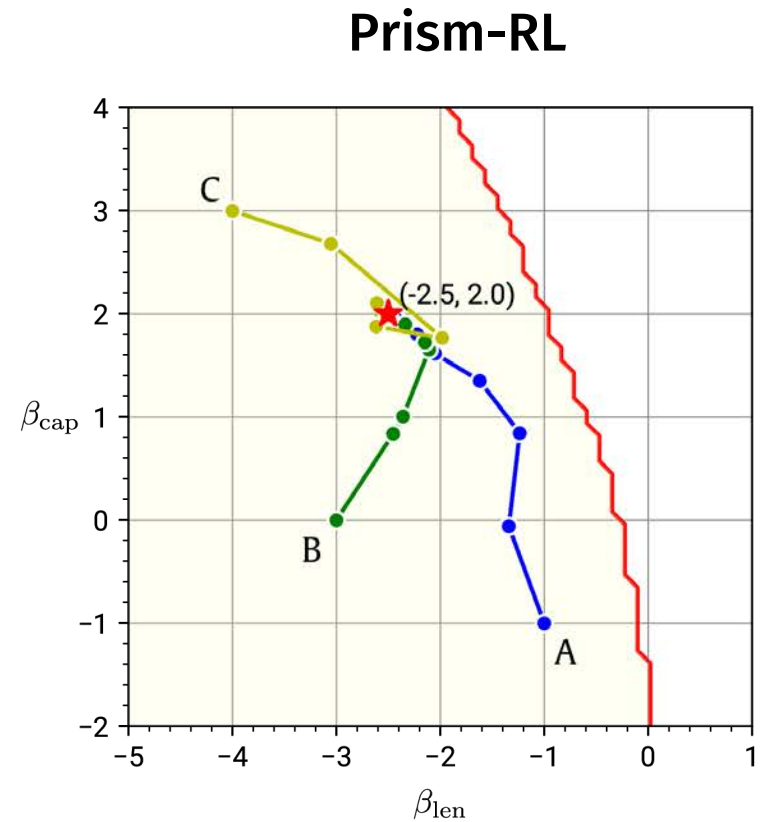
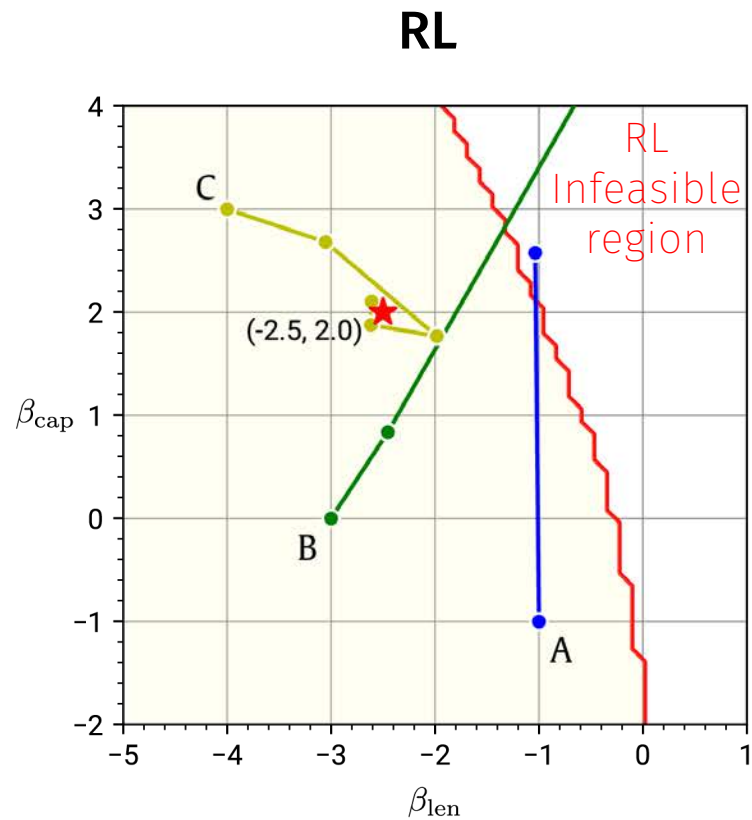
$\beta^*$	RL	Prism-RL
(-2.5, 2.0)	N/A	(-2.493**, 2.002**) 3.36s / LL=-4919.0
(-2.5, 1.5)	(-2.505**, 1.509**) 0.99s / LL=-5420.5	(-2.505**, 1.509**) 2.87s / LL=-5420.5
(-1.5, 1.5)	N/A	(-1.511**, 1.522**) 2.99s / LL=-7778.3
(-1.5, 1.0)	N/A	(-1.492**, 0.994**) 2.64s / LL=-8366.9

\* All true params are feasible solution to the RL model; estimation started with (-1, -1)

- **Prism-RL model reproduced the true values with high accuracy** even for the cases where RL model failed
- In the case where RL was successfully estimated, **the estimation results of both models were consistent**



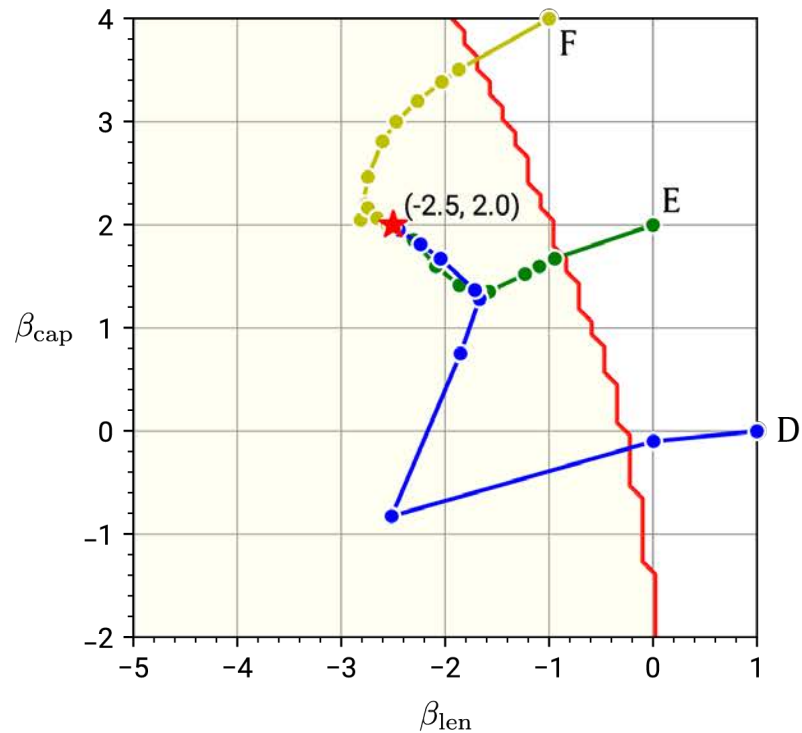
# 数値実験 | 推定におけるパラメータの探索過程



- RL model depends on starting point and often diverge during the estimation
- **Prism-RL model converges to the true value regardless of starting point** (\*update to infeasible region was not observed)

# 数値実験 | 推定におけるパラメータの探索過程

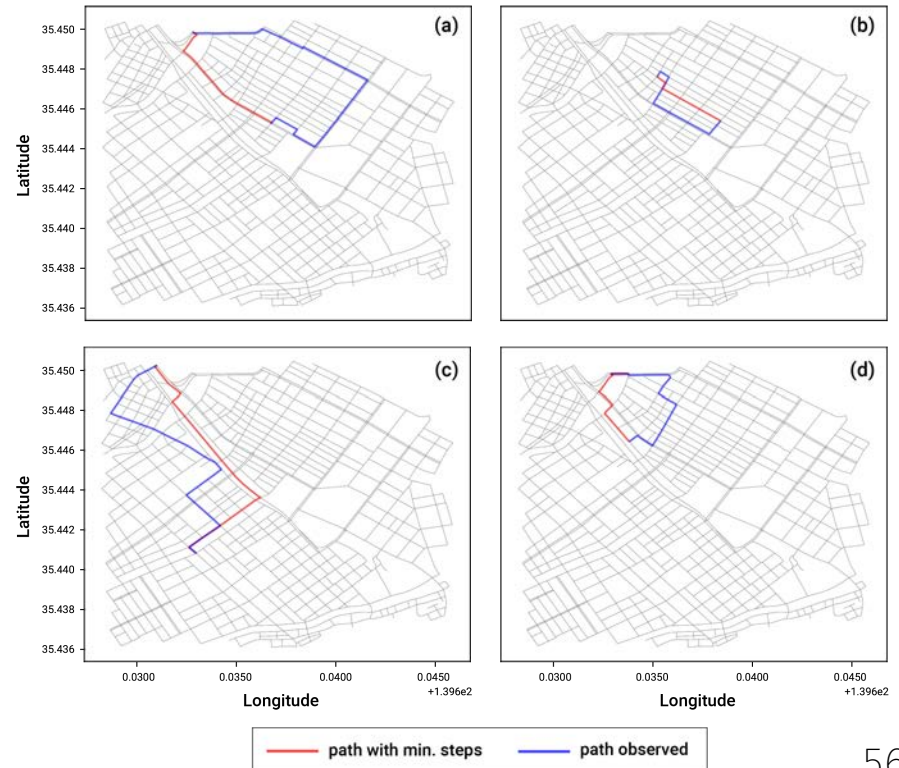
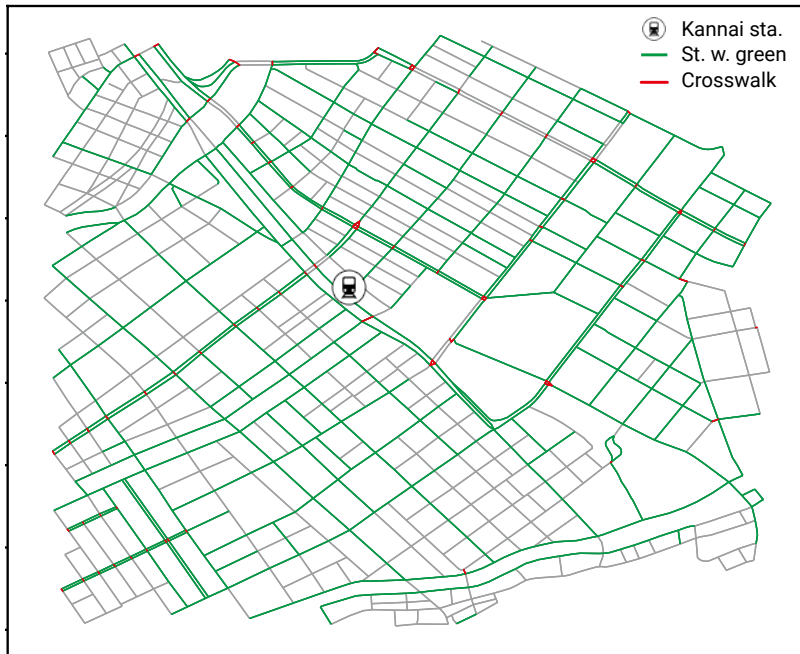
## Prism-RL



- Stably converged to the true parameter **even when starting point is outside the RL feasible region**
- No prior information on the true value is needed (you can set an initial point as you like)

# 実データへの適用

- 横浜・関内駅を中心とした1マイル四方
  - 724 Nodes, 2398 Links, 8434 Link pairs
- プロブパーソン (PP) 調査 (H30 PT) によるGPSデータ
  - 410 observed paths of 159 pedestrians, 164 destinations
  - **Diverge walking paths** including large detours



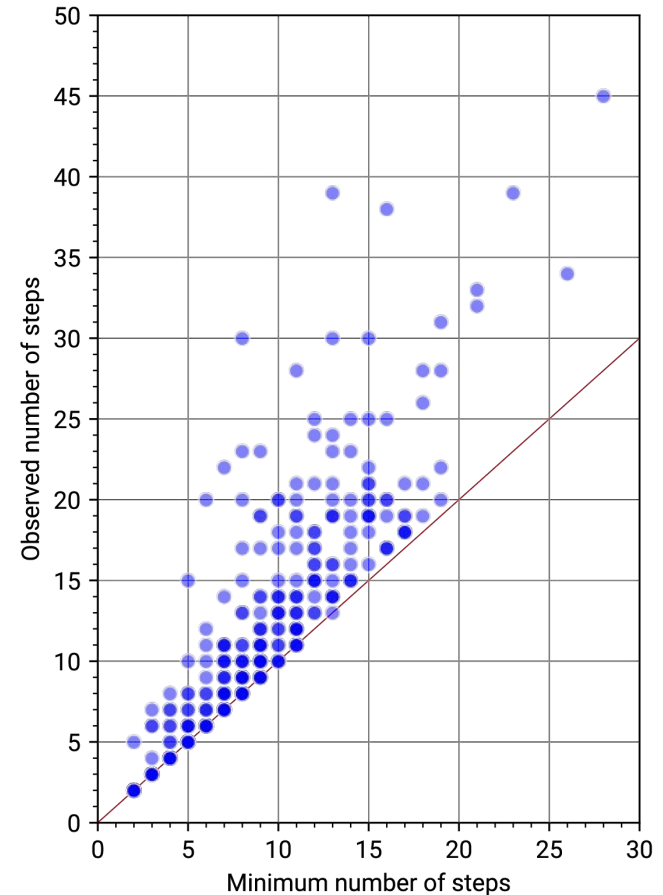
# 実データへの適用 | Choice-stage constraint $T$ の定義

count	mean	std	min	25%	50%	75%	max
410	1.24	0.39	1.00	1.00	1.08	1.33	3.75

- Define  $T$  for each  $d$  based on **observed detour rate**:

$$T_d \equiv \max_{n \in N_d} \{ \max(\underline{1.34} \times D^d(o_n), J_n) \}$$

- $N_d$  : Set of observed paths for  $d$
  - $D^d(o_n)$  : min. steps b/w observed OD pair
  - $J_n$  : observed no. of steps of  $n$
- 75 percentile value** (=1.34) to include diverse paths in the path set
  - All observations satisfy the prism constraint



# 実データへの適用 | 効用関数の設定

- Compare two different utility specifications:

$$v(a|k) = \beta_{\text{len}} \text{Length}_a + \beta_{\text{cross}} \text{Crosswalk}_a - 10 \text{Uturn}_{a|k}, \quad (\text{a})$$

$$v(a|k) = (\beta_{\text{len}} + \beta_{\text{green}} \text{Green}_a) \text{Length}_a + \beta_{\text{cross}} \text{Crosswalk}_a - 10 \text{Uturn}_{a|k}, \quad (\text{b})$$

- *Length*: length of link (m/10)
- *Crosswalk*: 1 if the link is a crosswalk and 0 otherwise
- *Green*: On-street green presence (1/0; interacted with link length)

検証：

1. 街路の緑化率は歩行者の経路選択効用に対して正の影響をもつのではないか？ Prism-RLモデルによりその効果が特定可能か？
2. 正の属性を考慮したとき， RL/Prism-RLの推定結果はどう異なるだろうか？

# 実データへの適用 | 推定結果

	RL (a)	Prism-RL (a)	RL (b)	Prism-RL (b)
$\hat{\beta}_{len}$	-0.297	-0.245	-	-0.266
std.err.	0.008	0.007	-	0.020
t-test	-38.832	-37.264	-	-13.283
$\hat{\beta}_{cross}$	-0.924	-0.774	-	-0.791
std.err.	0.075	0.171	-	0.068
t-test	-12.237	-4.517	-	-11.638
$\hat{\beta}_{green}$			-	0.049
std.err.			-	0.010
t-test			-	4.817
LL	-1772.972	-1637.484	-	-1612.894
#paths	410	410	410	410

- 負の属性のみを導入したモデル(a)では両モデルで推定結果を得た
  - Length, Crosswalk* are both negative → Pedestrians do not like paths with long distance and crosswalks
  - Prism-RL model fits better (details later)
- モデル(b)では、**Prism-RLモデルのみが推定結果を得ることができ、緑地の存在が与える正の効果を捉えることができた**
  - RL model failed with all the tested starting points
  - Adding the attribute improved goodness-of-fit (with 99% significance)

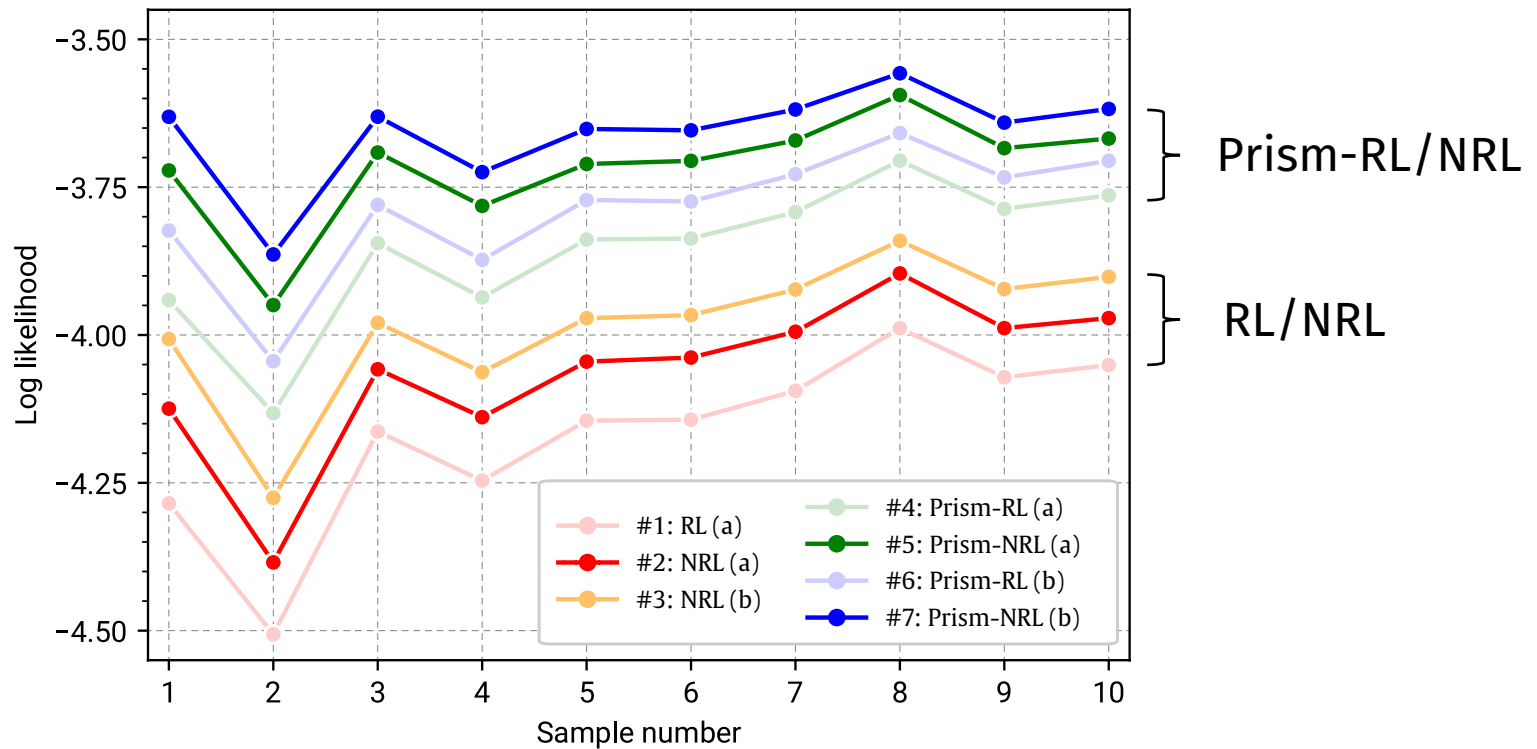
# 実データへの適用 | NRLモデルの推定結果

	NRL (a)	Prism-NRL (a)	NRL (b)	Prism-NRL (b)
$\hat{\beta}_{\text{len}}$	-0.460	-0.445	-0.485	-0.469
std.err.	0.030	0.039	0.082	0.062
t-test	-15.166	-11.304	-5.945	-7.568
$\hat{\beta}_{\text{cross}}$	-1.262	-1.206	-1.281	-1.206
std.err.	0.163	0.120	0.280	0.201
t-test	-7.728	-10.021	-4.567	-5.993
$\hat{\beta}_{\text{green}}$	-	-	0.078	0.082
std.err.	-	-	0.021	0.014
t-test	-	-	3.690	5.855
$\hat{\omega}$	0.064	0.095	0.063	0.091
std.err.	0.006	0.013	0.012	0.010
t-test	9.942	7.402	5.459	8.769
LL	-1734.622	-1587.079	-1707.068	-1565.531
#paths	410	410	410	410

\*  $\mu_k^d = e^{\omega \sqrt{\text{SP}_{kd}}}$  where  $\text{SP}_{kd}$  is the shortest path length between  $k$  and  $d$

- Prism-based approach also suited the NRL model as well
  - Same signs as RL models; Prism-NRL fits better than NRL
  - Captured correlation: scale (variance) decreases on links close to destination
- NRL could be estimated for spec. (b) but depended on starting point
  - Prism-NRL was successfully estimated with all initial points tested
  - Prism-based approach can also be viewed as a good approximation to provide a nice starting point for original RL models

# 実データへの適用 | クロス・ヴァリデーション



- Compare model performance of out-of-sample prediction
  - 10 sets of randomly split estimation and validation samples (8:2)
- Prism-RL model shows a higher prediction performance for all samples
  - **Universal set** (RL/NRL) vs **Prism-based path set** (Prism-RL/NRL)
  - Inclusion of positive attribute & nesting also improved the performance



# 実データへの適用 | Impact of $T$ (Prism-RL results)

Test **different detour rates**  $\gamma$  in  $T_d \equiv \max \left[ \max_{n \in N_d} \{ \gamma D^d(o_n), J_n \} \right]$

Small

Path set size	$\gamma$	$\hat{\beta}_{\text{len}}$	$\hat{\beta}_{\text{cross}}$	$\hat{\beta}_{\text{green}}$	$LL$
	1.25	-0.265	-0.785	0.049	-1605.104
	1.34	-0.266	-0.791	0.049	-1612.894
	1.50	-0.271	-0.796	0.050	-1632.277
	2.00	-0.284	-0.817	0.052	-1661.690

Large

\*Corresponds to RL when  $T$  goes to infinity

- The signs and scales of the estimates **remained unchanged**
- The **ratio of negative parameters systematically increased** as  $\gamma$  grew
  - Prism-RL model adjusted the parameter to keep little probability of detour/cyclic paths
- Model **fits better with smaller  $\gamma$  values** (i.e., tighter constraints)
  - Due to the **exclusion of behaviorally unrealistic paths**
  - **Trade-off with out-of-sample prediction** (choice set consistency)

# 余談: Reward inference as **inverse** problem

Forward decision-making problem: **Reinforcement Learning (RL)**

**Find optimal policy**  $p$  given reward functions  $r$

$$\max_p \mathbb{E}_p \left[ \sum_{t=0}^T \gamma^t r(a_t | s_t) \right]$$

Inverse estimation problem: **Inverse Reinforcement Learning (IRL)**

**Recover reward functions**  $r$  from expert demonstrations  $\mathcal{D}$

Maximum likelihood:

$$\max_{\theta} \mathbb{E}_{\tau \sim \mathcal{D}} [\log p_{\theta}(\tau)] = \mathbb{E}_{\tau \sim \mathcal{D}} \left[ \sum_{t=0}^T \gamma^t r_{\theta}(s_t, a_t) \right] - \log Z_{\theta}$$

where  $\mathcal{D} = \{\tau_j\}_{j=1}^N$      $\tau_j = \{(s_t^j, a_t^j)\}_{t=0}^T$

(Observed sequence of state-action pairs)

# 余談: Inverse Reinforcement Learning (IRL)

Recursive logit model / Maximum Entropy IRL

- Assuming **deterministic dynamics**
- **Linear-in-parameter** reward functions, i.e.,  $v_\theta(a|s) = \sum_l \theta_l x_{l,a|s}$

$$\max_{\theta} \mathbb{E}_{\tau \sim \mathcal{D}} [\log p_{\theta}(\tau)] = \sum_j \left( \sum_{t=0}^{T-1} v_{\theta}(s_{t+1}^j | s_t^j) - V_{\theta}(s_0^j) \right)$$

where  $p_{\theta}(a|s) = e^{v_{\theta}(a|s) + V_{\theta}(a) - V_{\theta}(s)}$

$$V_{\theta}(s) = \ln \sum_{a \in \mathcal{A}(s)} e^{v_{\theta}(a|s) + V_{\theta}(a)}$$

Solution algorithm (e.g., NXFP method; also see my previous year's lecture slides):

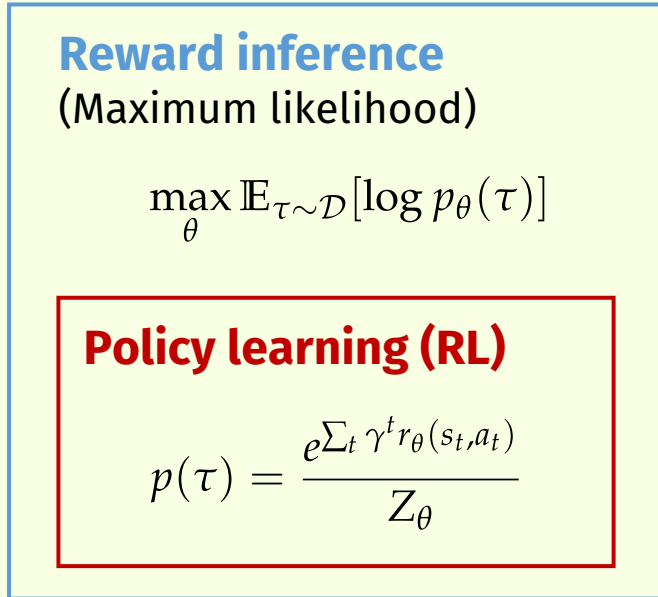
- Outer loop: **Parameter update** by non-linear optimization algorithm
  - Econometrics – newton-type methods
  - Machine learning – gradient descent methods
- Inner loop: **Compute value function** for each updated parameter
  - Recursive logit – solving the system of linear equation
  - MaxEnt IRL / nonlinear recursive models – value iteration

# 余談: Adversarial Inverse Reinforcement Learning (AIRL)

**Partition function  $Z$  for IRL is intractable** (cannot be computed) when

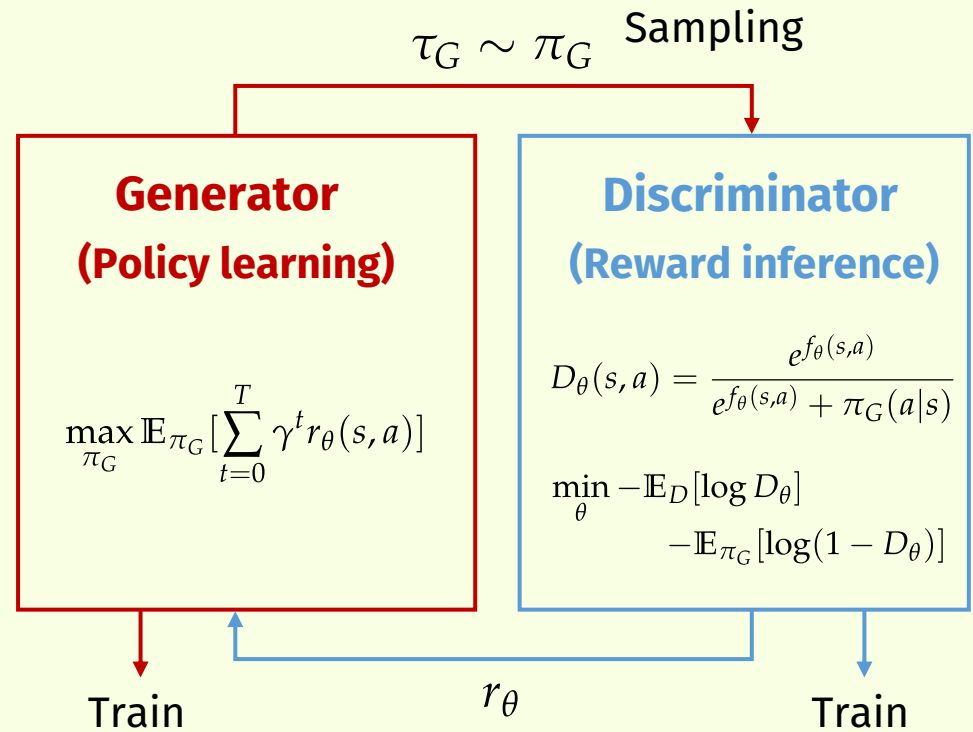
- State-action spaces are large or continuous
- Environment dynamics are stochastic and unknown

## MaxEnt IRL



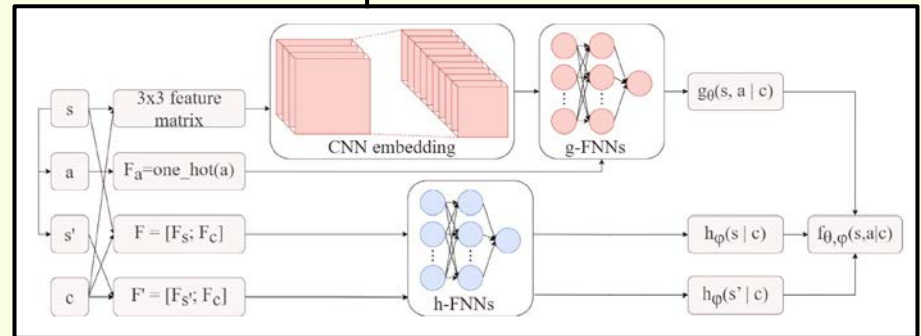
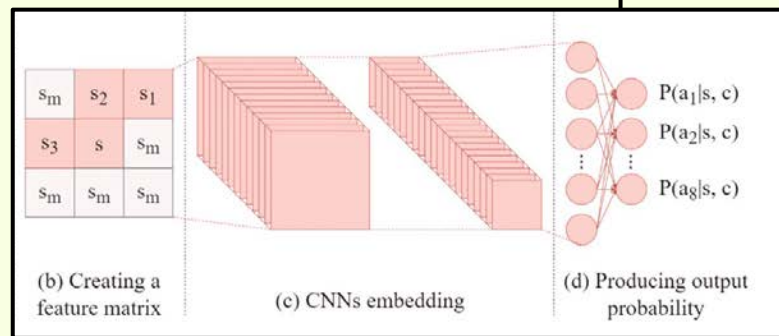
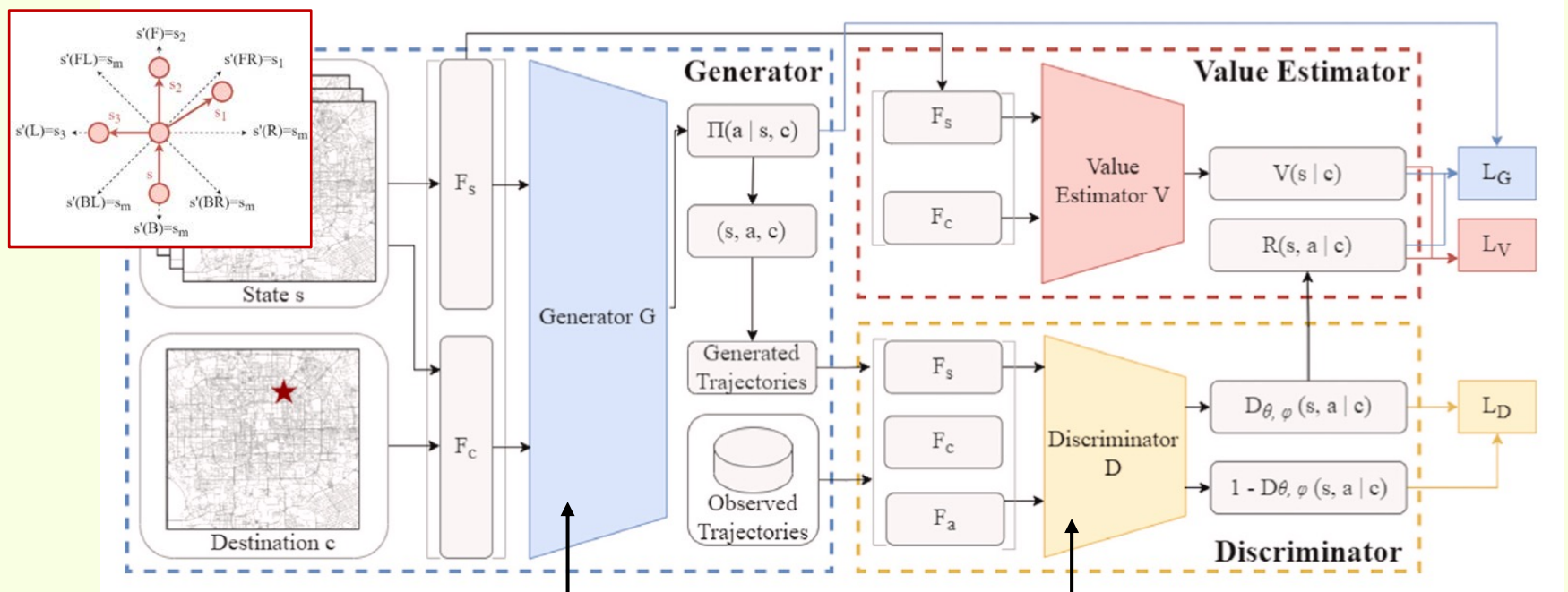
Train

## Adversarial IRL (Fu+2018)



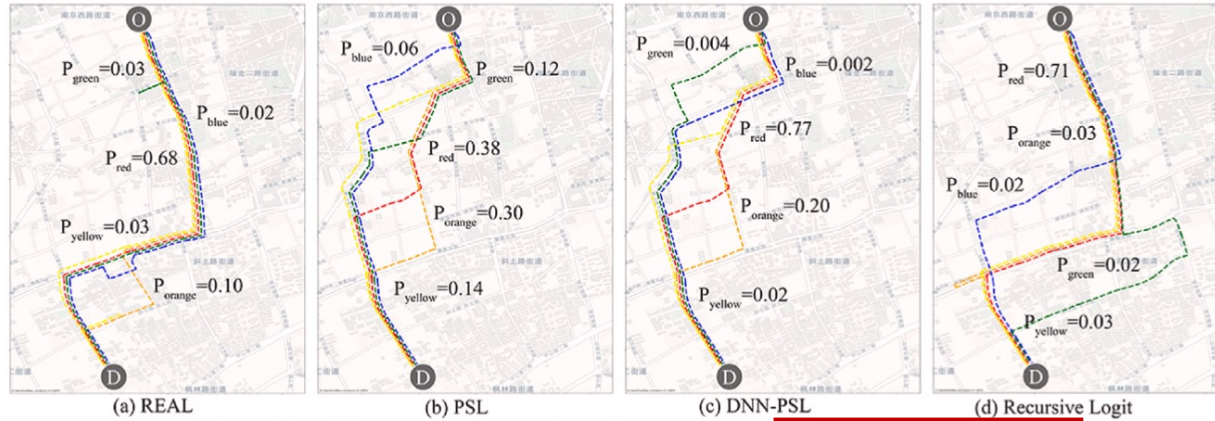
# 余談: Adversarial Inverse Reinforcement Learning (AIRL)

## Route choice application by Zhao and Liang (2023)

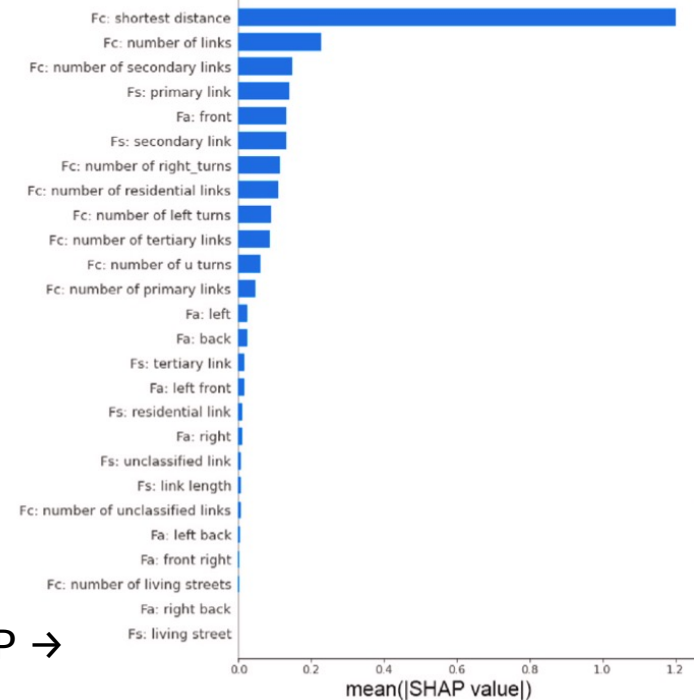
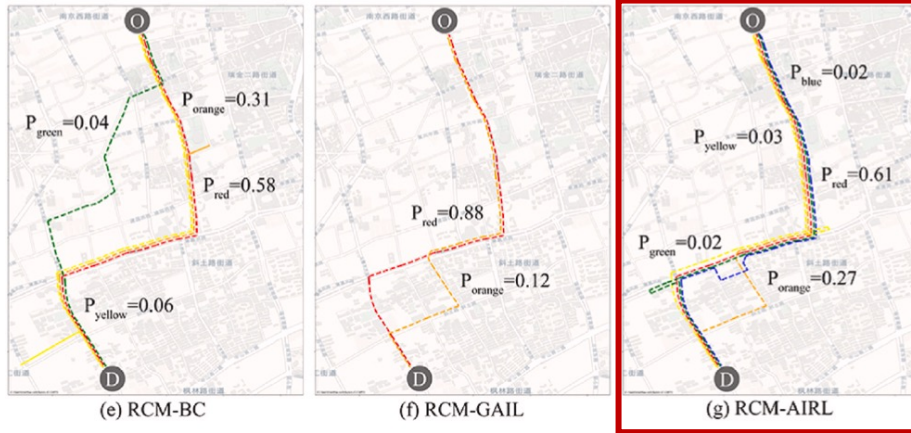


# 余談: Adversarial Inverse Reinforcement Learning (AIRL)

## Route choice application by Zhao and Liang (2023)



← Predictability



Interpretability by SHAP →

# 本日の内容

1. 交通分野における経路選択モデルの方法論的发展
2. マルコフ型経路選択モデルの推定問題とその解決
- 3. 局所的認知効用を考慮したモデリング**

Oyama, Y. (2024) Global path preference and local response: A reward decomposition approach for network path choice analysis in the presence of visually perceived attributes. Transportation Research Part A: Policy and Practice 181: 103998



# 経路選択における局所的な反応

移動の中で新たに獲得した情報を用いて人はどう行動を変化させるか？



Disrupted network with local unavailability



Visual attractiveness along a path

- (1) Local response to environment

- Change route to an available alternative / walk on visually nice street

- (2) Global path preference

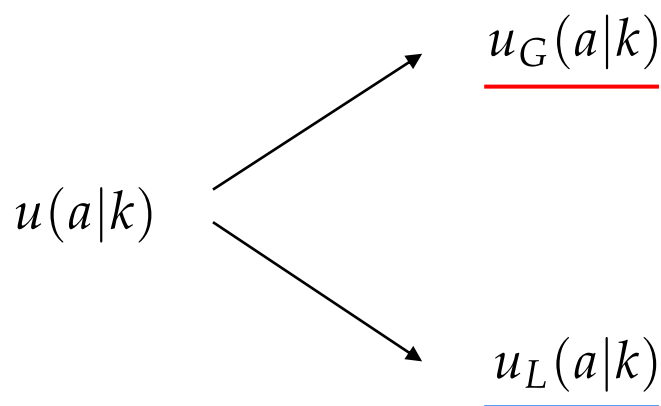
- Still want to arrive at destination as fast as possible / want to continue walking along a roofed path to destination

RQ: どの属性がどの程度大域的・局所的なメカニズムに影響を与えるか？



# The Idea:

## Decomposition of Markov reward function



**Global utility**  $v_G(a|k) + \epsilon_G(a|k)$

Perceived from any state in the network

**Local utility**  $v_L(a|k) + \epsilon_L(a|k)$

Perceived only when a traveler is in state  $k$   
(Treated as zero in other states)

$$v_G(a|k) = v(\mathbf{x}_{G,a|k}, \beta_G)$$

Function of **globally perceived attributes**

$$v_L(a|k) = v(\mathbf{x}_{L,a|k}, \beta_L)$$

Function of **locally perceived attributes**

# Reward decomposition approach

## Global value function

The expected and accumulated rewards toward destination  $d$  and represents global path preferences of traveler

$$V^d(k) \equiv \mathbb{E} \left[ \max_{r \in R_{kd}} \{u_G(r)\} \right] \quad \text{where} \quad u_G(r = \{a_1, \dots, a_J\}) = \sum_{t=1}^{J-1} \gamma^{t-1} u_G(a_{t+1}|a_t)$$

大域的認知効用のみを価値関数に導入:  $V^d(k) = V^d(x_G, \beta_G)$

In recursive form:

$$V^d(k) = \frac{1}{\mu_G} \ln \sum_{a \in A(k)} e^{\mu_G \{v_G(a|k) + \gamma V^d(a)\}} \quad \mu_G : \text{scale of } \epsilon_G(a|k)$$

# Global-local path choice model

エージェントの効用最大化行動:

1. Global value function: Represent **global path preferences**
2. Local utility: Affect path choice and leads to locally optimal behavior in each state, resulting in the **adaptation of path choice to local conditions** of the environment

選択確率:

$$p^d(a|k) = \frac{e^{\mu\{v_G(a|k) + v_L(a|k) + \gamma V^d(a)\}}}{\sum_{a' \in A(k)} e^{\mu\{v_G(a'|k) + v_L(a'|k) + \gamma V^d(a')\}}}$$

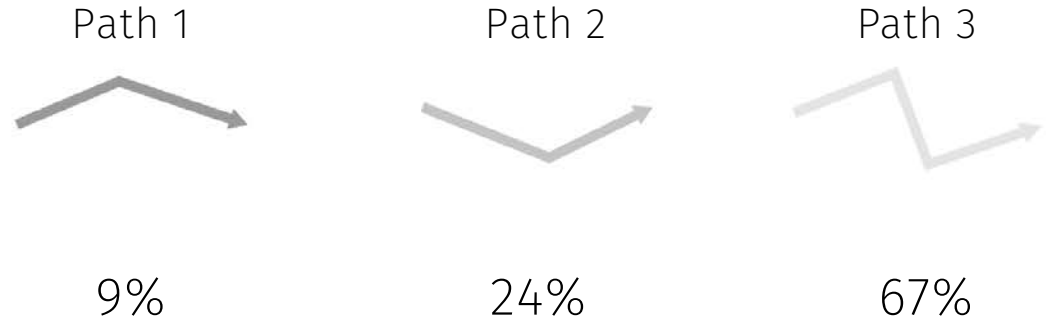
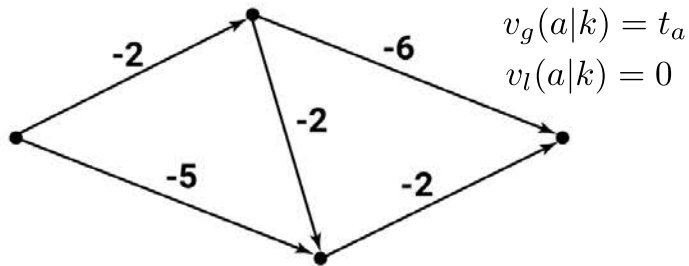
対数尤度関数:

$$LL(\boldsymbol{\theta}; \mathbf{r}) = \sum_{n=1}^N \sum_{j=1}^{J_n-1} \ln p^{d_n}(a_{j+1}|a_j) = \sum_{n=1}^N \sum_{j=1}^{J_n-1} \left\{ v(a|k) + V^d(a) - \ln \sum_{a' \in A(k)} e^{v(a'|k) + V^d(a')} \right\}$$

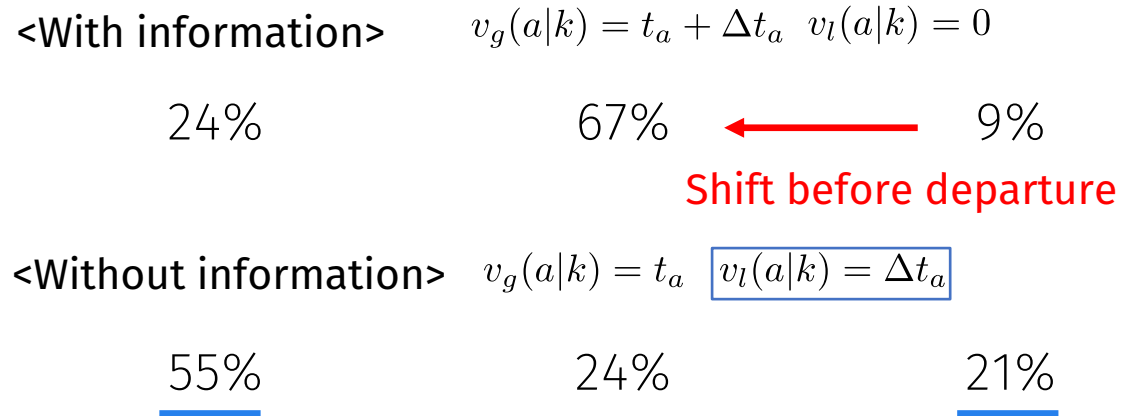
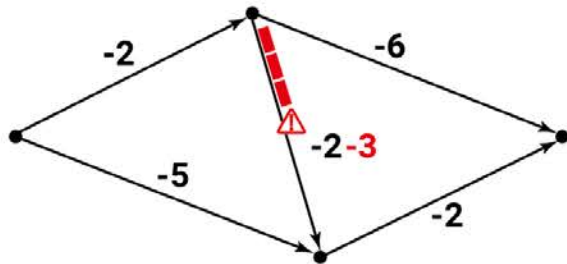
Needs **only revealed path observations** (link sequences)!

# 数値計算例（モデル）

Case 1: General path choice



Case 2&3: Unexpected event

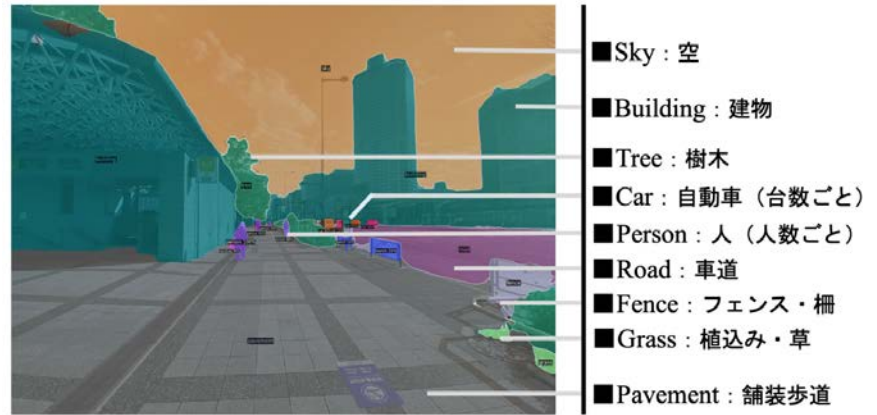
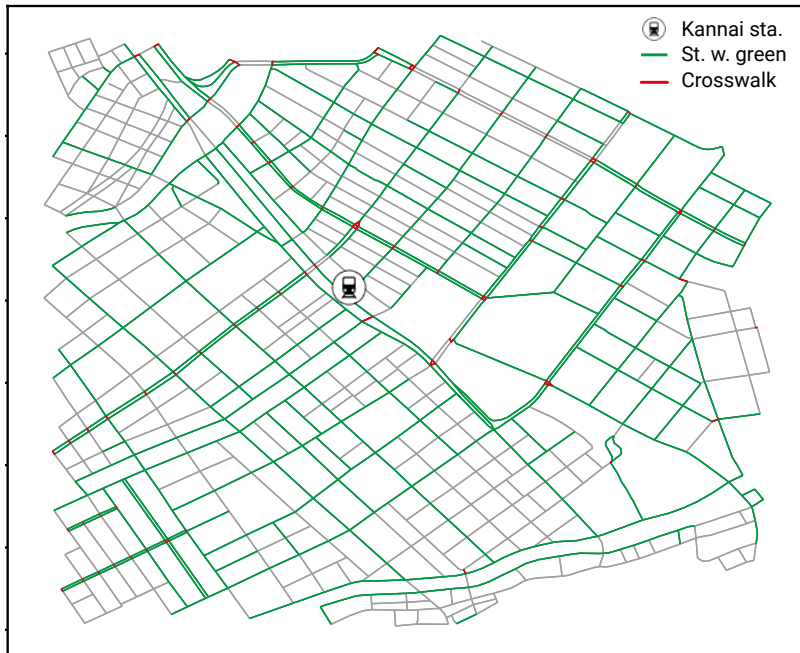


Local optimal: using globally slow routes

Flexibly capture different route choice based on **utility specifications**

# 実データへの適用

- 横浜・関内駅を中心とした1マイル四方
  - 724 Nodes, 2398 Links, 8434 Link pairs
- プロブパーソン (PP) 調査 (H30 PT) によるGPSデータ
  - 410 observed paths of 159 pedestrians, 164 destinations
  - **Diverge walking paths** including large detours



**Visual streetscape features extraction** by Semantic Segmentation

- **Green View Index (GVI)** : Average pixel ratio of “vegetation” from Google Street View images

# 実データへの適用 | クロス・ヴァリデーション

**仮説:** GVIのような**視覚的な街路の質に対して歩行者は局所的に認知・反応し**，経路選択行動を変化させているのではないか

Compare three specifications:

$$\begin{aligned} \text{(a)} \quad & \begin{cases} v_G(a|k) &= (\beta_{\text{len}}^G + \beta_{\text{walk}}^G x_a^{\text{walk}} + \beta_{\text{green}}^G x_a^{\text{green}}) x_a^{\text{len}} + \beta_{\text{cross}}^G x_a^{\text{cross}} - 20x_{a|k}^{\text{uturn}} \\ v_L(a|k) &= 0 \end{cases} \\ \text{(b)} \quad & \begin{cases} v_G(a|k) &= (\beta_{\text{len}}^G + \beta_{\text{walk}}^G x_a^{\text{walk}}) x_a^{\text{len}} + \beta_{\text{cross}}^G x_a^{\text{cross}} - 20x_{a|k}^{\text{uturn}} \\ v_L(a|k) &= \beta_{\text{green}}^L x_a^{\text{green}} x_a^{\text{len}} \end{cases} \\ \text{(c)} \quad & \begin{cases} v_G(a|k) &= (\beta_{\text{len}}^G + \beta_{\text{walk}}^G x_a^{\text{walk}} + \beta_{\text{green}}^G x_a^{\text{green}}) x_a^{\text{len}} + \beta_{\text{cross}}^G x_a^{\text{cross}} - 20x_{a|k}^{\text{uturn}} \\ v_L(a|k) &= \beta_{\text{green}}^L x_a^{\text{green}} x_a^{\text{len}} \end{cases} \end{aligned}$$

$x_a^{\text{len}}$  : Link length (m)

$x_a^{\text{green}}$  : GVI (ratio)

$x_a^{\text{uturn}}$  : Uturn dummy

$x_a^{\text{cross}}$  : Crosswalk dummy

$x_a^{\text{walk}}$  : Sidewalk width (m)

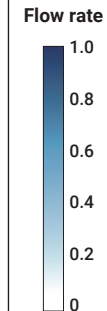
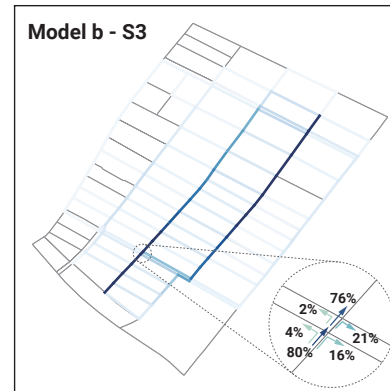
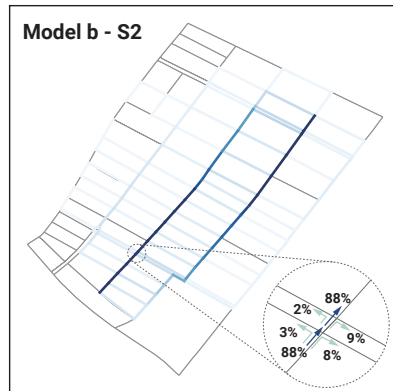
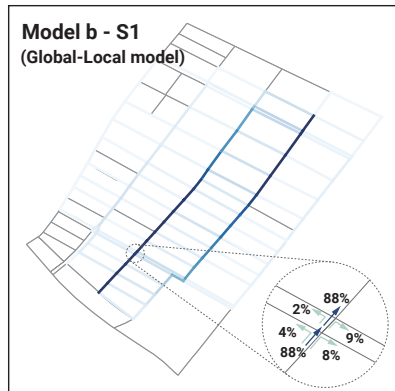
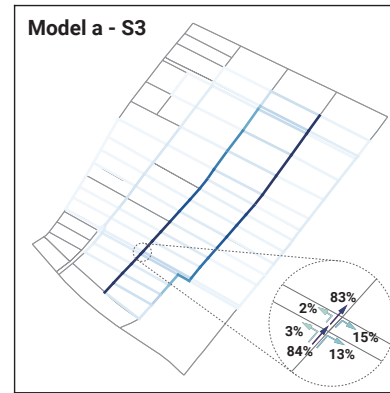
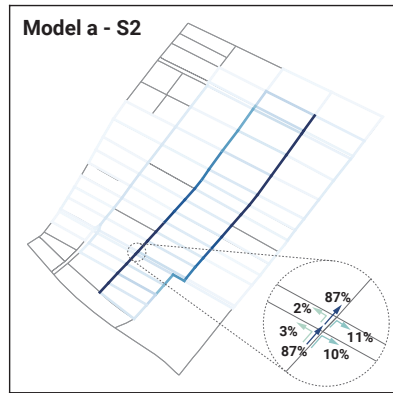
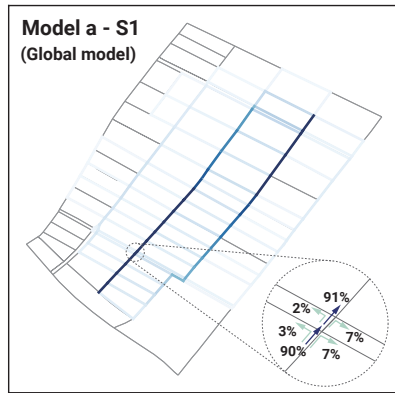
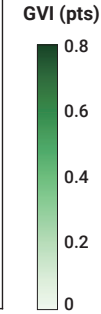
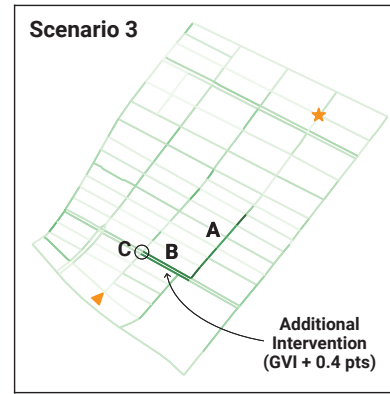
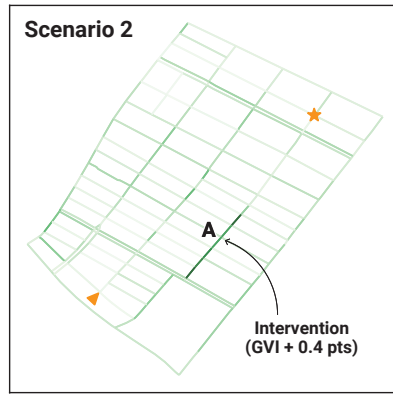
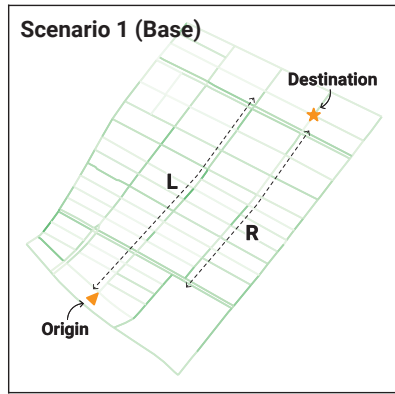
(fixed effect)

# 実データへの適用 | 推定結果

		Global model		Global-local model		Global-local model	
		Model (a)		Model (b)		Model (c)	
	Parameter	Estimate	t-stat <sup>†</sup>	Estimate	t-stat <sup>†</sup>	Estimate	t-stat <sup>†</sup>
Global	$\hat{\beta}_{len}$	-0.322	-29.65***	-0.316	-31.30***	-0.317	-27.85***
	$\hat{\beta}_{cross}$	-0.927	-16.71***	-0.886	-15.51***	-0.888	-15.34***
	$\hat{\beta}_{walk}$	0.063	6.28***	0.069	7.64***	0.067	6.49***
	$\hat{\beta}_{green}$	0.072	1.26	-	-	0.019	0.28
Local	$\hat{\beta}_{green}$	-	-	0.139	3.04***	0.132	2.42**
Path observations			410	410	410	410	410
Log-likelihood			-1701.2	-1697.3	-1697.3	-1697.3	-1697.3
$\bar{\rho}^2$			0.6897	0.6904	0.6904	0.6902	0.6902
AIC			3410.4	3402.6	3402.6	3404.6	3404.6
Validation LL			-3.981	-3.973	-3.973	-3.977	-3.977

- 歩行者は大域的な選好として、距離が短く、交差点を渡る回数が少なく、歩道幅員の広い経路を選んでいる。GVIは歩行行動に正の影響を持つ。
- モデルbは im-sample/out-of-sample 両方に対してモデルaよりも高い適合度を示しており、歩行者は（どちらかといえば）**GVIを局所的に認知し、経路選択行動を変化させている**と言える
- これはモデルcの結果からも支持される

# 実データへの適用 | シミュレーション結果

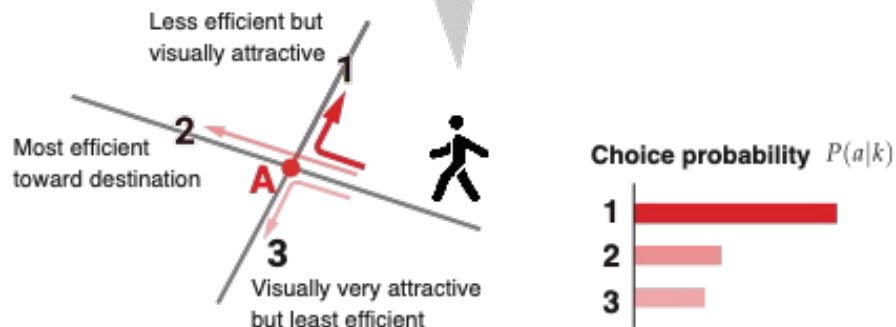
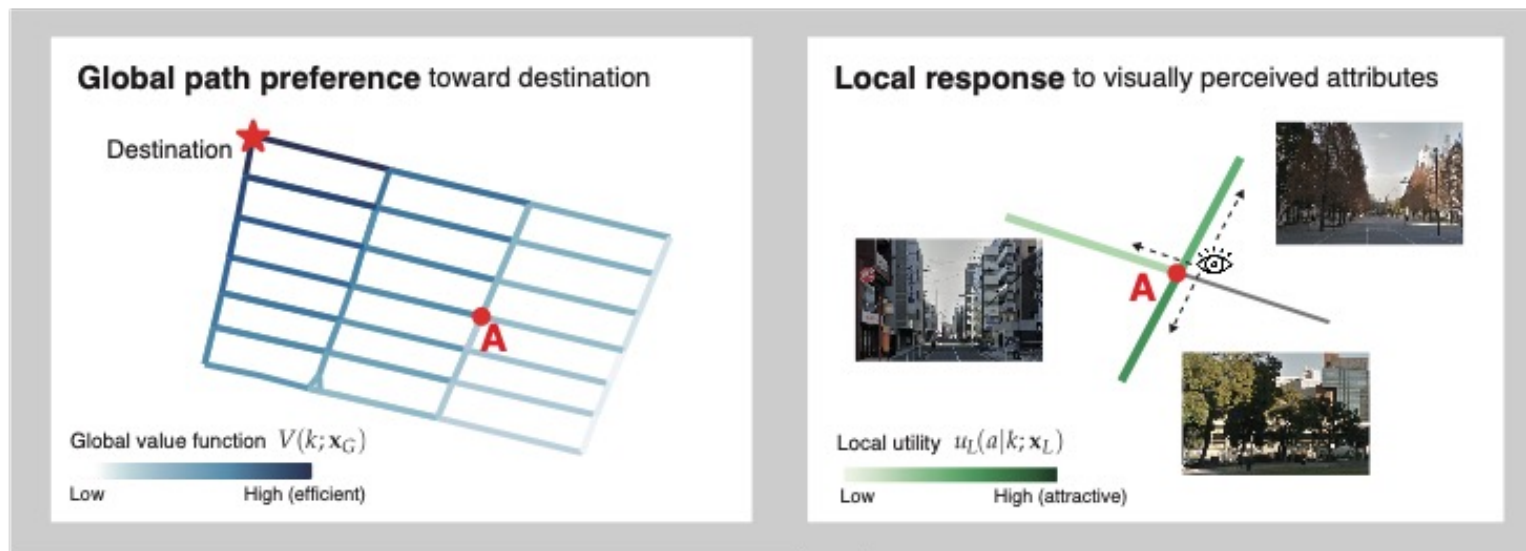


- 通りAのみに対する緑地量増加施策は、モデルbでは効果がない
- 通りBと合わせて導入することで効果を発揮
- → 政策変数が局所的に認知されているとき、**普段歩かれる通りから直接視認できる街路に対して政策を行う**ことで効果を発揮・上昇できる



# Global-local path choice model

- アドホックなルールを必要としないシンプルなモデリング
- 変数単位で大域的・局所的効用に対する異なる影響を分析可能
- 実際に観測された経路のみで推定（実証分析）が可能



## おわりに

- 主に交通分野におけるネットワーク行動モデルの発展や、発表者の近年の取り組みについて紹介
- ネットワークの定義しだいで様々な展開・応用が可能な “**Network Behavioral Science**”
  - 交通 (Random Utility Route Choice; Traffic Assignment)
  - 経済 (Perturbed Utility Theory; Potential Game Theory)
  - 情報 (Inverse Reinforcement Learning; Randomized Shortest Paths)
- 今後の展開
  - 実証的関心：モデルの空間/時間/主体的移転性
  - 技術的関心：多様なデータの観測誤差・欠損への対応（観測モデル）と予測モデルへの統合
  - 方法論的関心：Non link-additiveな属性の取り込みなど

ご清聴ありがとうございました