

Mitigating the impact of selfish routing: An optimal-ratio control scheme (ORCS) inspired by autonomous driving

Kenan Zhang, Yu (Marco) Nie

2025年5月26日 理論談話会#8

B4 中澤輝一

Section1, 2 研究の概要

研究の概要

- 近年自動運転車が増加
 - 車間距離短縮・自動車間の相互作用で混雑緩和
 - 自動運転車は経路も制御できる
 - システム最適になるような経路に流すことができる
-
- しかし、すべての車をリアルタイムに制御するのは通信容量的にも、計算コスト的にも現実的ではない
 - システム最適化のために望まない経路を通らされる場合がある
→システム最適と制御強度のバランスが重要

研究の概要

- 2種類の利用者が共存するネットワークを想定
- 利用者均衡 (UE) ユーザー：自身の旅行時間を最小化する

$$\min_{\mathbf{x}_1 \in \Omega_1} z_1(\mathbf{x}_1, \mathbf{x}_2) = \sum_{a \in A} \int_0^{x_{1a}} t_a(x + x_{2a}) dx,$$

- システム最適 (SO) ユーザー：総旅行時間を最小化する

$$\min_{\mathbf{x}_2 \in \Omega_2} z_2(\mathbf{x}_1, \mathbf{x}_2) = \sum_{a \in A} t_a(x_{1a} + x_{2a})(x_{1a} + x_{2a}).$$

- 総旅行時間と制御強度の合計値を最小化するように、各ODごとにSOユーザーの割合とその時の配分を決める

$$\min \quad \underbrace{\gamma \|\tilde{\mathbf{q}}\|_1}_{\text{制御強度}} + \underbrace{z_2(\mathbf{x}^*)}_{\text{総旅行時間}}$$

(SOユーザーの数)

- γ は係数（この値によって最適解は変わる）

研究の概要

年	著者	内容
1988	Harker	UEとSOの混合配分を初めて検証
1990	Van Vuren	経路案内利用者（SOに近い）と非利用者（UEに近い）の混合
1991	Van Vuren & Watling	SOとUEとSUE（UEのように行動するが、不確実性がある）
1997	Korilis	全ての利用者を制御できなくても、制御率が高ければほぼシステム最適になる
1998	Yang	旅行者情報提供システムの普及率の影響
2004	Yang & Huang	旅行者情報提供システムの最適な価格決定戦略
2007	Yang	シュタッケルベルグ競争を用いた。各割合は所与
2017	Bagloee	制御されるポテンシャルはあっても、実際になるかは自由

これまでの研究：全ODで制御割合は均一
→ODごとに変えたらどうなる？

研究の概要

- 最適割合制御スキームはネットワーク設計問題（NDP：network design problem）に直面
- NDPは、上層で変数が最適化され、下層は上層で決定した変数によって定式化されるという、2層構造になっている

上層

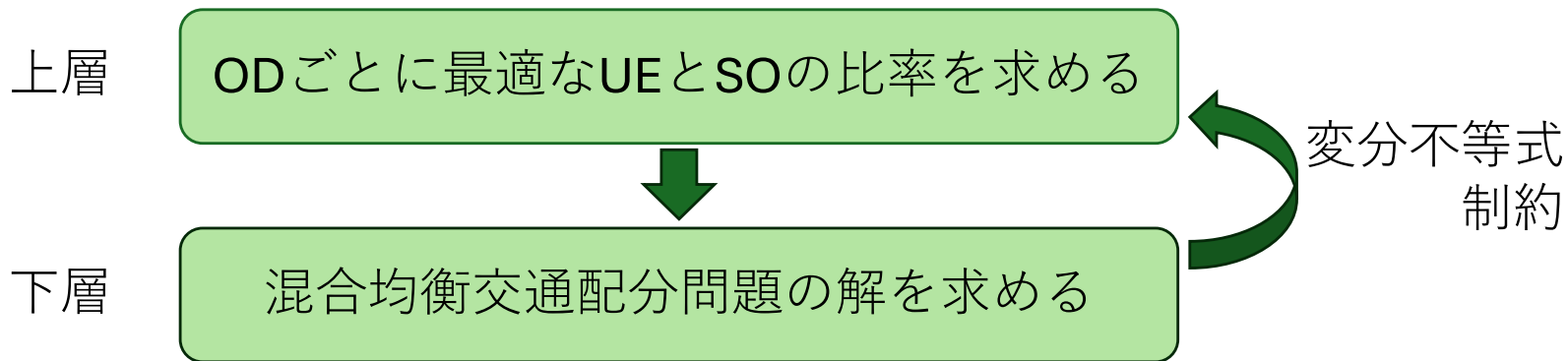
ODごとに最適なUEとSOの比率を求める



下層

混合均衡交通配分問題の解を求める

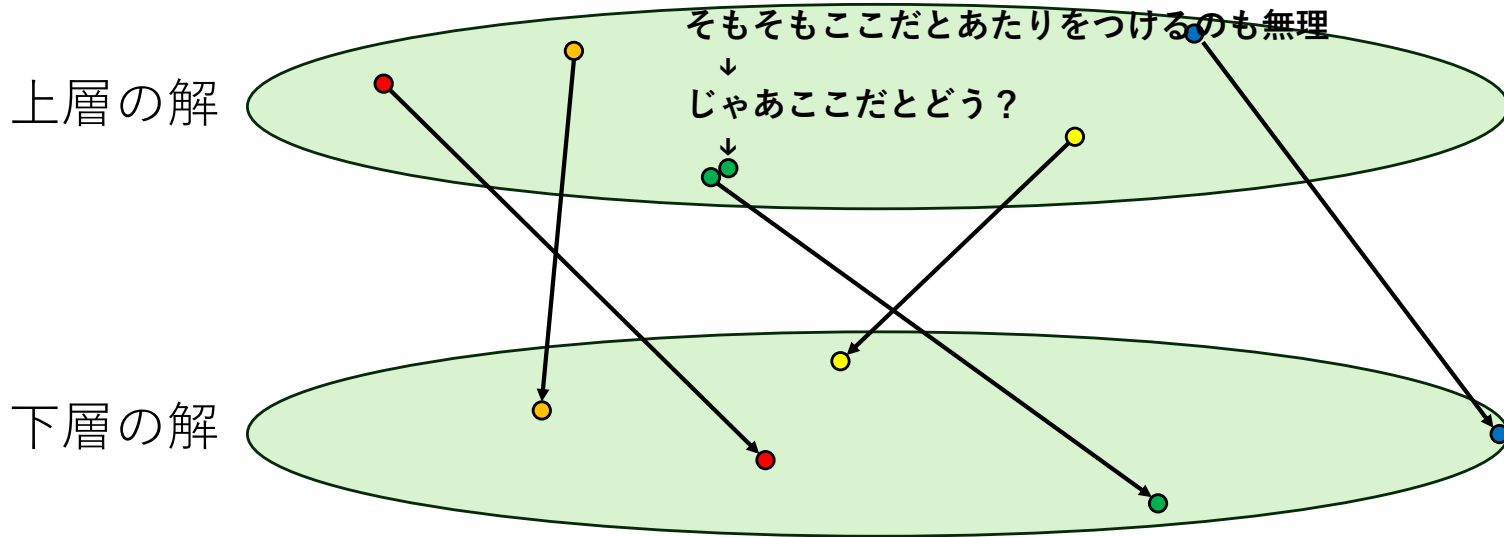
研究の概要



- NDPは通常上層に、下層の問題を均衡制約として統合した形で定式化される
- 上層と下層に明確な写像関係がない（上層でSOユーザーの割合を変えると下層がどう変化するかの関係式がなく、再計算が必要）
- 上層と下層の関係を概算するための、感度分析に基づく方法を提案

研究の概要

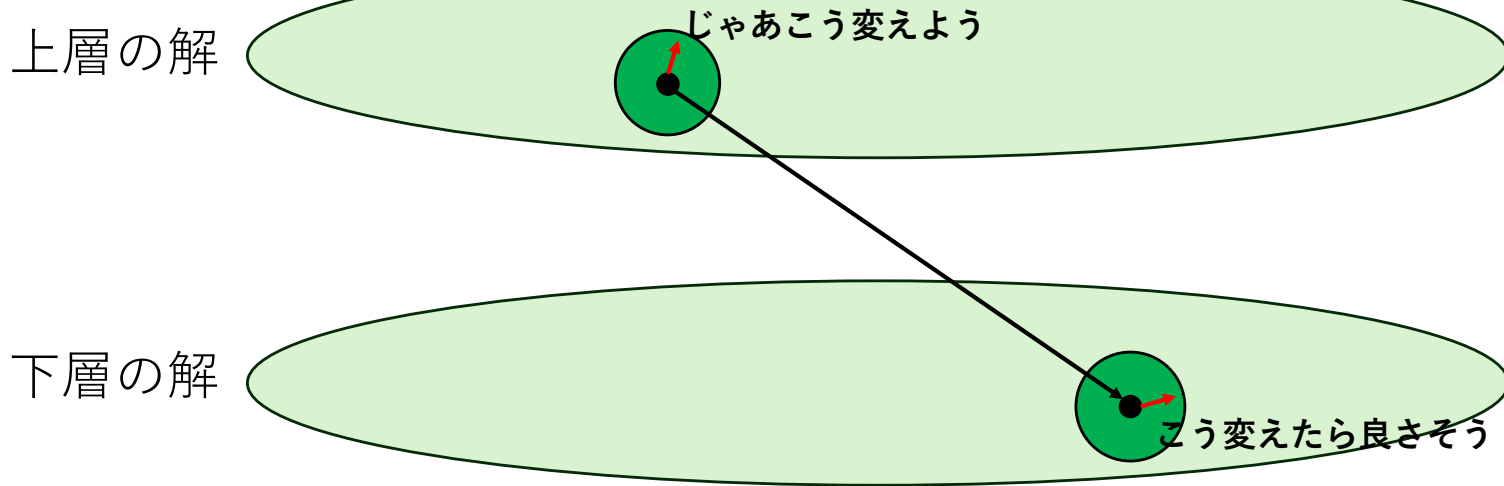
- 愚直に解くと



- 上層の一つの解に対して一つの下層の最適解
- 上層を少しでも変えると下層の変化が不明
- 勾配がわからないので、上層の解の変え方も不明

研究の概要

- 感度分析を使うと



- 近傍であれば勾配を近似して計算できる

研究の概要

- 次に、上層の問題を効率的に解くための交互方向乗数法（ADMM : alternative direction method of multiplier）を提案

$$\min \quad \gamma \|\tilde{\mathbf{q}}\|_1 + z_2(\mathbf{x}^*)$$

- 制御強度と総旅行時間を交互に更新して最適解を目指す

研究の概要

- 新規性
 - ODによって制御率を変えるORCS手法
 - 感度分析・交互方向乗数法の交通分野への活用
- 有用性
 - わずかな制御量でシステム最適にかなり近づけられる
 - 計算量もそこまで大きくない
- 信頼性
 - 数学的に正しい定式化
 - 大規模なネットワークを使った検証

Section3 マルチクラス利用者の混合均衡

上層

ODごとに最適なUEとSOの比率を求める



下層

混合均衡交通配分問題の解を求める

問題設定

上層

ODごとに最適なUEとSOの比率を求める



下層

混合均衡交通配分問題の解を求める

- Network $G(N, A)$
- N : ノード、 A : リンク
- 1 : UEに関する変数
- 2 : SOに関する変数
- D : ODのペア、 q_d : ODペア d の需要
- $\tilde{q}$: ODごとにUEをSOに変えさせた数
- x : 各リンクを通過している交通量
- t : コスト
- mt : 限界コスト
- P : パスの集合
- f : パスフロー
- Δ : リンク-パス対応行列
- c : パスコスト
- M : OD-パス対応行列

問題設定

上層

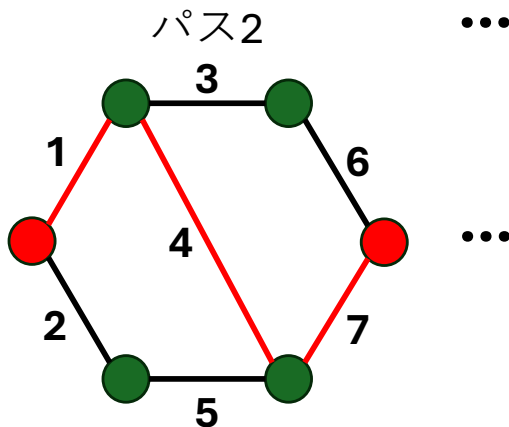
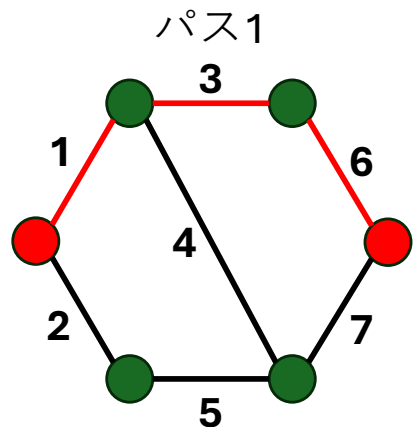
ODごとに最適なUEとSOの比率を求める

下層

混合均衡交通配分問題の解を求める

- Δ : リンク-パス対応行列

$$\Delta = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ \vdots & & & & & & \end{pmatrix}$$



...

...

問題設定

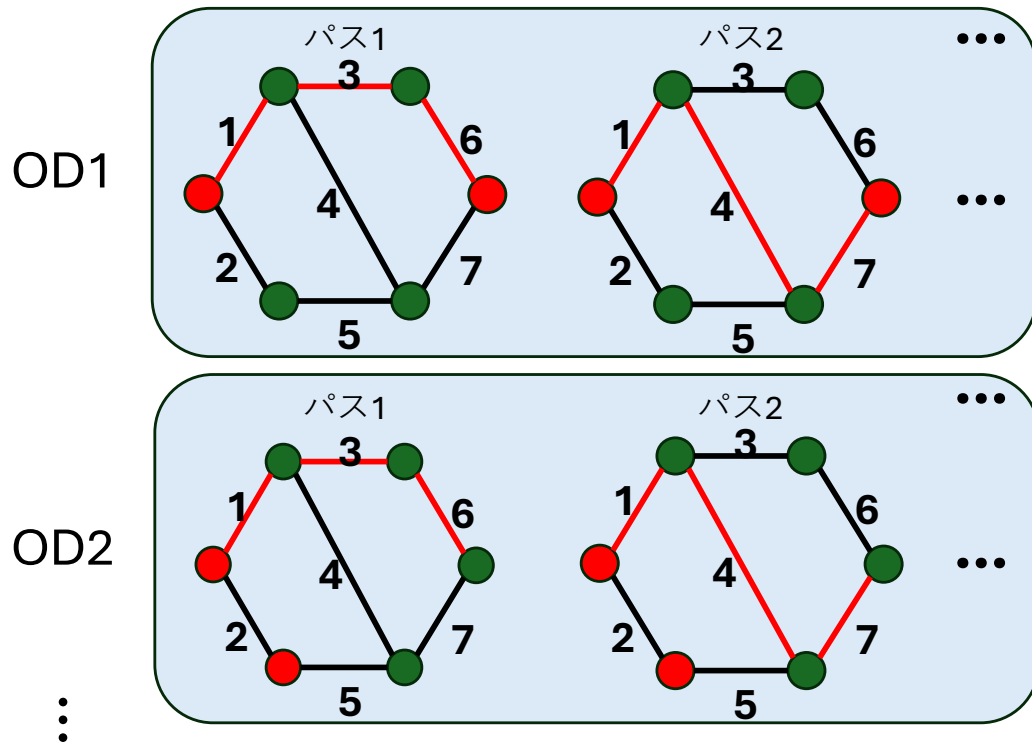
上層

ODごとに最適なUEとSOの比率を求める

下層

混合均衡交通配分問題の解を求める

- M : OD-パス対応行列



混合均衡交通配分モデル

上層

ODごとに最適なUEとSOの比率を求める



下層

混合均衡交通配分問題の解を求める

- 実行可能集合

$$\Omega_1 \triangleq \{\mathbf{x}_1 | \mathbf{x}_1 = \Delta \mathbf{f}_1, M \mathbf{f}_1 = \mathbf{q}_1; \mathbf{f}_1 \geq 0\},$$

$$\Omega_2 \triangleq \{\mathbf{x}_2 | \mathbf{x}_2 = \Delta \mathbf{f}_2, M \mathbf{f}_2 = \mathbf{q}_2; \mathbf{f}_2 \geq 0\}.$$

- 定式化

$$\min_{\mathbf{x}_1 \in \Omega_1} z_1(\mathbf{x}_1, \mathbf{x}_2) = \sum_{a \in A} \int_0^{x_{1a}} t_a(x + x_{2a}) dx, \quad \leftarrow \text{UEユーザーの均衡}$$

$$\min_{\mathbf{x}_2 \in \Omega_2} z_2(\mathbf{x}_1, \mathbf{x}_2) = \sum_{a \in A} t_a(x_{1a} + x_{2a})(x_{1a} + x_{2a}). \quad \leftarrow \text{SOユーザーの均衡}$$

混合均衡交通配分モデル

上層

ODごとに最適なUEとSOの比率を求める



下層

混合均衡交通配分問題の解を求める

- 仮定1：

$$t_a(x_a) = t_a^0 + \alpha_a(x_a)^\beta, \forall a \in A \text{ where } t_a^0 \geq 0, \alpha_a > 0, \beta > 0.$$

- 交通量が増えると所要時間は単調に増加する（下に凸）
- 解の一意性を保証するための仮定

混合均衡交通配分モデル

上層

ODごとに最適なUEとSOの比率を求める



下層

混合均衡交通配分問題の解を求める

- 次の変分不等式問題（VI：variational inequality）と同値

$$t(\mathbf{x}_1^* + \mathbf{x}_2^*) \cdot (\mathbf{x}_1 - \mathbf{x}_1^*) \geq 0, \quad \forall \mathbf{x}_1 \in \Omega_1, \quad \leftarrow \text{UEユーザーが別の選択をすると、その人の旅行時間は変わらないか伸びる}$$

$$mt(\mathbf{x}_1^* + \mathbf{x}_2^*) \cdot (\mathbf{x}_2 - \mathbf{x}_2^*) \geq 0, \quad \forall \mathbf{x}_2 \in \Omega_2. \quad \leftarrow \text{SOユーザーが別の選択をすると、総旅行時間は変わらないか伸びる}$$

- この2式は次の1式にまとめることができる

$$T(\mathbf{x}^*) \cdot (\mathbf{x} - \mathbf{x}^*) \geq 0, \quad \forall \mathbf{x} \in \Omega$$

混合交通均衡の解法アルゴリズム

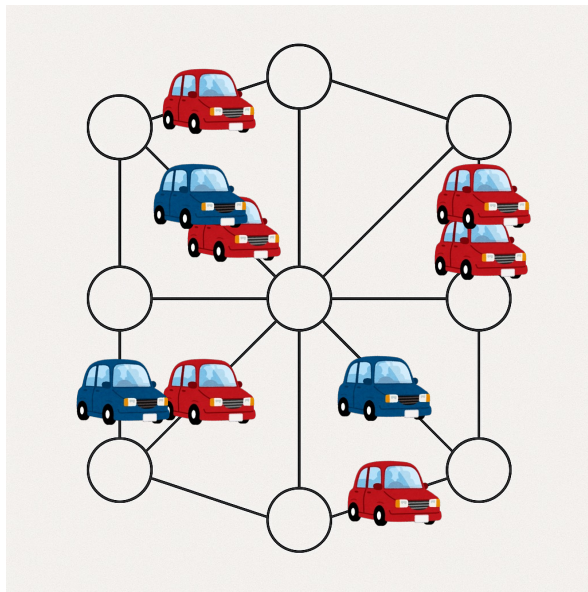
上層

ODごとに最適なUEとSOの比率を求める

下層

混合均衡交通配分問題の解を求める

- まずまっさらなネットワークでUEユーザーを均衡にする
- UEユーザーは固定した状態で、SOユーザーを均衡にする
- SOユーザーは固定した状態で、UEユーザーは再度経路選択する
- 以下、繰り返す
- 相対ギャップ（前回の解との差）がUE、SOともに一定値を下回ったら収束、計算終了
- このようにすることで、各ODの比率が決まれば、その最適な経路配分が求められる
- **これを毎回解くのは大変→勾配近似**



Section4 最適割合制御スキーム

ORCS : Optimal-ratio control scheme

上層

ODごとに最適なUEとSOの比率を求める



下層

混合均衡交通配分問題の解を求める

変分不等式
制約



MPECによる定式化と解法の詳細

上層

ODごとに最適なUEとSOの比率を求める

下層

混合均衡交通配分問題の解を求める

変分不等式
制約

- ORCS問題をMPECにより定式化すると、

$$\min \quad \gamma \|\tilde{\mathbf{q}}\|_1 + z_2(\mathbf{x}^*)$$

$$s. t. \quad \mathbf{q}^- \leq \tilde{\mathbf{q}} \leq \mathbf{q}^+$$

←制御数の変化量は下限以上上限以下

$$T(\mathbf{x}^*) \cdot (\mathbf{x} - \mathbf{x}^*) \geq 0, \forall \mathbf{x} \in \Omega(\tilde{\mathbf{q}})$$

←配分を最適解から変更すると
総所要時間は増加する

MPECによる定式化と解法の詳細

上層

ODごとに最適なUEとSOの比率を求める

下層

混合均衡交通配分問題の解を求める

変分不等式
制約

$$\min \gamma \|\tilde{\mathbf{q}}\|_1 + z_2(\mathbf{x}^*)$$

l_1 ノルム：UE→SOに変えられた台数の合計

- l_2 ノルムを使うよりもスパースな結果（多くのODペアを制御しない）を得られる

- 3台をUEからSOに変える場合

$\tilde{\mathbf{q}}$	$\ \tilde{\mathbf{q}}\ _1$	$\ \tilde{\mathbf{q}}\ _2$
(1, 1, 1)	3	$\sqrt{3}$ 好まれる
(3, 0, 0)	3	3 好まれない

- （スパースの方がいいのは実用上の理由）

MPECによる定式化と解法の詳細

上層

ODごとに最適なUEとSOの比率を求める

下層

混合均衡交通配分問題の解を求める

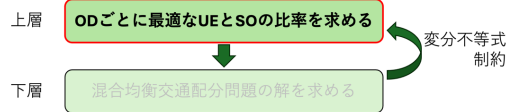
変分不等式
制約

$$\min \quad \underline{\gamma} \|\tilde{\mathbf{q}}\|_1 + z_2(\mathbf{x}^*)$$

γ : ペナルティ項の係数

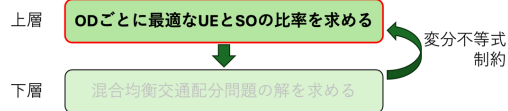
- γ が大きいと、システム最適から離れてでも制御を減らす
- γ が小さいと、多く制御してシステム最適に近づける
- 通常、試行錯誤の末に決定される

MPECによる定式化と解法の詳細



- 定理1：仮定1が満たされれば、先のMPEC問題の解は常に存在する
- 証明
- Q を $\tilde{q}$ の実行可能集合とする。リンクコスト関数が仮定1を満たせば、 Q を Ω に移す写像が存在する。それゆえ、MPEC問題は必然的に $\tilde{q}$ がリーダーによって決められ、 $\chi(\tilde{q})$ がフォロワーの反応となるようなシュタッケルベルグゲームを表す。
- HarkerとPangの系1より、MPECは最適解を持つ

MPECによる定式化と解法の詳細



- 定理1より解は存在する
- しかし、VI制約が高非線形で、MPECは凸でない
 - 複数の局所最適解があり、グローバル最適解を見つけるのは難しい
- しかし、上層の目的関数が強凸性であれば、局所最適解はグローバル最適解と非常に近い (Yang 1995)
 - 局所最適解を求める

ORCSの解法アルゴリズム

上層

ODごとに最適なUEとSOの比率を求める

下層

混合均衡交通配分問題の解を求める

変分不等式
制約

- MPEC問題を解くためには、2つの大きな障壁がある
- 1. 上層と下層の写像を表すこと（勾配近似）
- 2. 連続で微分可能ではない目的関数を持つ最適制御問題を解くこと
- 実行可能な各 q から混合均衡配分への写像は定理1より存在するが、それは閉形式ではない
- それゆえ、感度分析に基づいた近似的勾配法がこの写像を概算するために提案

ORCSの解法アルゴリズム

上層

ODごとに最適なUEとSOの比率を求める

下層

混合均衡交通配分問題の解を求める

変分不等式
制約

- MPEC問題を解くためには、2つの大きな障壁がある
- 1. 上層と下層の写像を表すこと（勾配近似）
- 2. 連続で微分可能ではない目的関数を持つ最適制御問題を解くこと
- 実行可能な各 q から混合均衡配分への写像は定理1より存在するが、それは閉形式ではない
- それゆえ、感度分析に基づいた近似的勾配法がこの写像を概算するために提案

ORCSの解法アルゴリズム

上層

ODごとに最適なUEとSOの比率を求める

変分不等式
制約

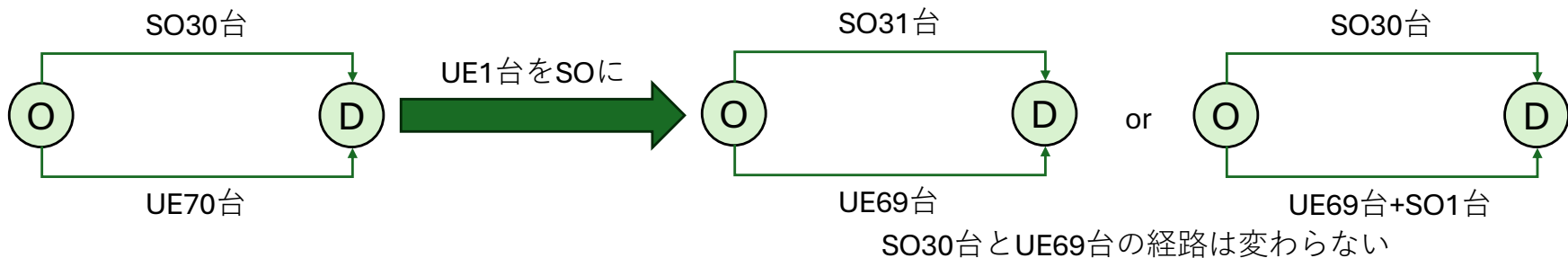
下層

混合均衡交通配分問題の解を求める

- 定理2：小さなフローシフトが起きても経路選択は変わらないのだとすれば、目的関数の第二項は次のように表せる

$$z_2(\chi(\tilde{\mathbf{q}})) \approx z_2(\mathbf{x}_1^n, \mathbf{x}_2^n) + \nabla z_2^n \cdot \tilde{\mathbf{q}},$$

- 上層で各ODの制御数を微小に変化させたとき、下層の経路選択は変更された車しか変わらない



ORCSの解法アルゴリズム

上層

ODごとに最適なUEとSOの比率を求める

変分不等式
制約

下層

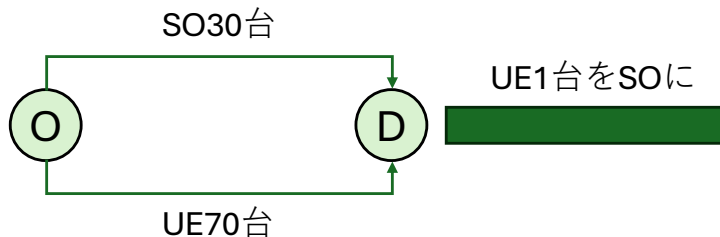
混合均衡交通配分問題の解を求める

- 定理2：小さなフローシフトが起きても経路選択は変わらないのだとすれば、目的関数の第二項は次のように表せる

$$z_2(\chi(\tilde{\mathbf{q}})) \approx z_2(\mathbf{x}_1^n, \mathbf{x}_2^n) + \nabla z_2^n \cdot \tilde{\mathbf{a}}.$$

混合交通均衡の解法アルゴリズム

- 上層で各ODの制御数を微小
更された車しか変わらない



- これを毎回解くより楽！ →

- まずまっさらなネットワークでUEユーザーを均衡にする
- UEユーザーは固定した状態で、SOユーザーを均衡にする
- SOユーザーは固定した状態で、UEユーザーは再度経路選択する
- 以下、繰り返す
- 相対ギャップ（前回の解との差）がUE、SOともに一定値を下回ったら収束、計算終了
- このようにすることで、各ODの比率が決まれば、その最適な経路配分が求められる

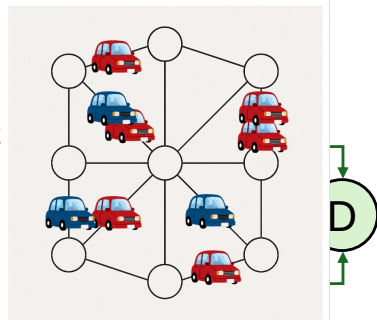
- これを毎回解くのは大変 → 勾配近似

2025/5/26

理論談話会#8

Mitigating the impact of selfish routing: An optimal-ratio control scheme (ORCS) inspired by autonomous driving

20



ORCSの解法アルゴリズム

上層

ODごとに最適なUEとSOの比率を求める

下層

混合均衡交通配分問題の解を求める

変分不等式
制約

- MPEC問題を解くためには、2つの大きな障壁がある
- 1. 上層と下層の写像を表すこと（勾配近似）
- 2. 連続で微分可能ではない目的関数を持つ最適制御問題を解くこと
- 実行可能な各 q から混合均衡配分への写像は定理1より存在するが、それは閉形式ではない
- それゆえ、感度分析に基づいた近似的勾配法がこの写像を概算するために提案

ORCSの解法アルゴリズム

上層

ODごとに最適なUEとSOの比率を求める

下層

混合均衡交通配分問題の解を求める

変分不等式
制約

- 上層の最適制御問題を解くのは、目的関数の l_1 ノルムが連続で微分可能ではないせいで難しい
- 目的関数を、繰り返すことで解くことができる2つの部分問題に分割するために交互方向乗数法 (ADMM: alternative direction method of multipliers) を導入

$$\min \quad \gamma \|\mathbf{c}\|_1 + z_2(\chi(\tilde{\mathbf{q}})),$$

$$s. t. \quad \mathbf{c} - \tilde{\mathbf{q}} = \mathbf{0},$$

ORCSの解法アルゴリズム

上層

ODごとに最適なUEとSOの比率を求める

下層

混合均衡交通配分問題の解を求める

変分不等式
制約

- ADMMを適用するため、最適制御問題を書き直す

$$\min \quad \gamma \|\mathbf{c}\|_1 + z_2(\chi(\tilde{\mathbf{q}})),$$

$$s. t. \quad \mathbf{c} - \tilde{\mathbf{q}} = \mathbf{0},$$

- $\mathbf{c}$ をわざわざ導入するのは、 $\mathbf{c}$ と $\mathbf{q}$ を交互に更新していくため
- 一時的に違う値になる
- 最終的には制約条件により、等しくなる

ORCSの解法アルゴリズム

上層

ODごとに最適なUEとSOの比率を求める

変分不等式
制約

下層

混合均衡交通配分問題の解を求める

- 拡張ラグランジュ関数（制約条件を入れる）

$$L_{\gamma,\rho}(\tilde{\mathbf{q}}, \mathbf{c}, \lambda) = \gamma \|\mathbf{c}\|_1 + z_2(\chi(\tilde{\mathbf{q}})) + \underbrace{\lambda^T (\tilde{\mathbf{q}} - \mathbf{c})}_{\text{補正項}} + \underbrace{\frac{\rho}{2} \|\tilde{\mathbf{q}} - \mathbf{c}\|_2^2}_{\text{罰則項}}$$

- 補正項（ $\tilde{q} = c$ が実現される方向への勾配を作る）
- 罰則項（ $\tilde{q} \neq c$ だと最小化したい目的関数の値が大きくなる）

元の式

$$\min \quad \gamma \|\mathbf{c}\|_1 + z_2(\chi(\tilde{\mathbf{q}})),$$

$$s. t. \quad \mathbf{c} - \tilde{\mathbf{q}} = \mathbf{0},$$

ORCSの解法アルゴリズム

上層

ODごとに最適なUEとSOの比率を求める

変分不等式
制約

下層

混合均衡交通配分問題の解を求める

• 式変形

$$L_{\gamma,\rho}(\tilde{\mathbf{q}}, \mathbf{c}, \lambda) = \gamma \|\mathbf{c}\|_1 + z_2(\chi(\tilde{\mathbf{q}})) + \lambda^T (\tilde{\mathbf{q}} - \mathbf{c}) + \frac{\rho}{2} \|\tilde{\mathbf{q}} - \mathbf{c}\|_2^2,$$

$$\lambda^T (\tilde{\mathbf{q}} - \mathbf{c}) + \frac{\rho}{2} \|\tilde{\mathbf{q}} - \mathbf{c}\|_2^2 = \frac{\rho}{2} \|(\tilde{\mathbf{q}} - \mathbf{c}) + \lambda/\rho\|_2^2 - \frac{\rho}{2} \|\lambda/\rho\|_2^2$$

$$= \frac{\rho}{2} \|\tilde{\mathbf{q}} - \mathbf{c} + \mathbf{u}\|_2^2 - \frac{\rho}{2} \|\mathbf{u}\|_2^2,$$

$$L_{\gamma,\rho}(\tilde{\mathbf{q}}, \mathbf{c}, \mathbf{u}) = \gamma \|\mathbf{c}\|_1 + z_2(\chi(\tilde{\mathbf{q}})) + \frac{\rho}{2} \|\tilde{\mathbf{q}} - \mathbf{c} + \mathbf{u}\|_2^2 - \frac{\rho}{2} \|\mathbf{u}\|_2^2.$$

ORCSの解法アルゴリズム

上層

ODごとに最適なUEとSOの比率を求める

下層

混合均衡交通配分問題の解を求める

変分不等式
制約

• 変数の更新

$$\tilde{\mathbf{q}}^{k+1} = \operatorname{argmin} z_2(\chi(\tilde{\mathbf{q}})) + \frac{\rho}{2} \|\tilde{\mathbf{q}} - \mathbf{c}^k + \mathbf{u}^k\|_2^2,$$

$$\mathbf{c}^{k+1} = \operatorname{argmin} \gamma \|\mathbf{c}\|_1 + \frac{\rho}{2} \|\tilde{\mathbf{q}}^{k+1} - \mathbf{c} + \mathbf{u}^k\|_2^2,$$

$$\mathbf{u}^{k+1} = \mathbf{u}^k + \tilde{\mathbf{q}}^{k+1} - \mathbf{c}^{k+1}.$$

$$c_d^{k+1} = \begin{cases} \min\{q_d^+, \tilde{q}_d^{k+1} + u_d^k - \gamma/\rho\}, & \tilde{q}_d^{k+1} + u_d^k > \gamma/\rho; \\ 0, & \tilde{q}_d^{k+1} + u_d^k \in [-\gamma/\rho, \gamma/\rho]; \\ \max\{q_d^-, \tilde{q}_d^{k+1} + u_d^k + \gamma/\rho\}, & \tilde{q}_d^{k+1} + u_d^k < -\gamma/\rho. \end{cases}$$

ORCSの解法アルゴリズム

上層

ODごとに最適なUEとSOの比率を求める

下層

混合均衡交通配分問題の解を求める

変分不等式
制約

- 上記のようになると、MPEC問題は2つの部分問題に分割でき、繰り返すことで解くことができる
- まずは初期のUEとSOの割合に基づいて下層の混合交通均衡を解く
- 続けて、上層の最適制御問題が、上層の解から下層の解への影響の近似として解かれる
- 利用者の割合が更新されると、新たな反復が始まる
- 両方の部分問題の解が収束するまでこの反復は続く

ORCSの解法アルゴリズム まとめ

- 本来はこういう構造

上層

ODごとにUEとSOの比率を決める



下層

混合均衡交通配分問題の解を求める



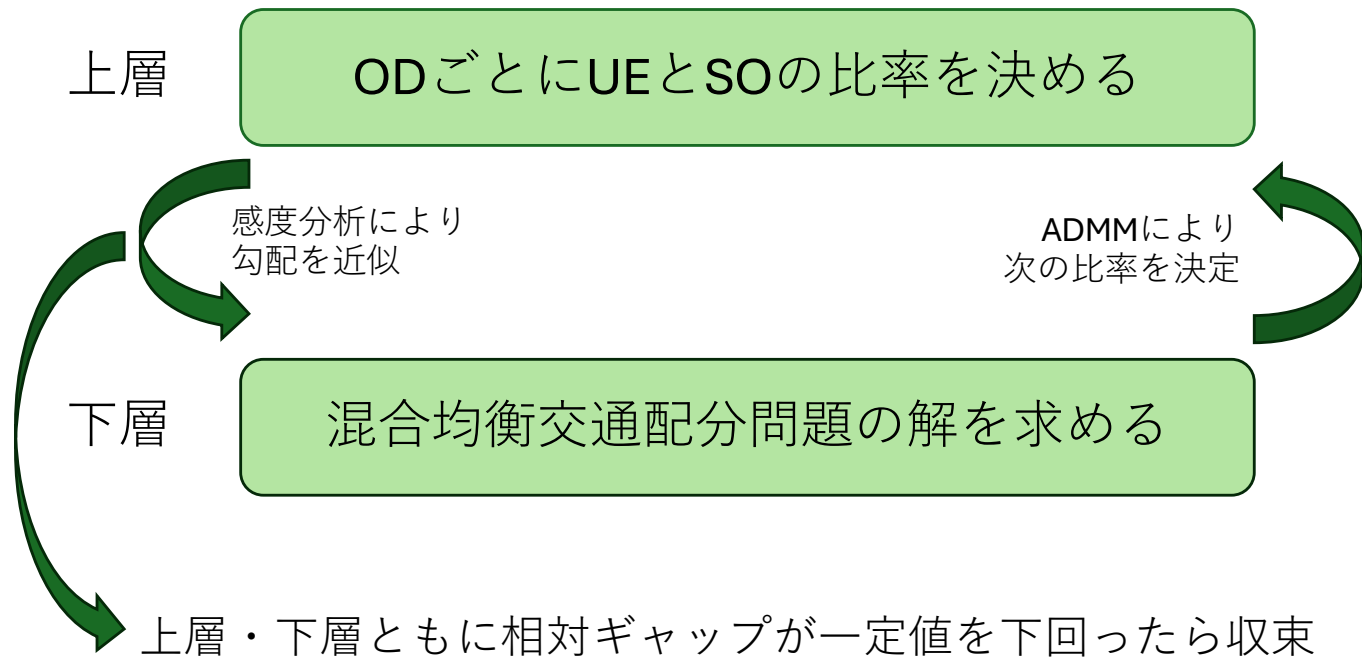
上層の比率を変えたらより良い解があるかも？

でも勾配がわからない

→変分不等式問題を解き直す

解き直すといっても、勾配がわからないので適当に変えてみるしかない

ORCSの解法アルゴリズム まとめ



Section5 数值的実験

数値の実験

- Mac OS X 64-bit workstation
- 2.7GHz intel Core i5 CPU
- 8GB 1867 MHz DDR3 RAM
- coded in MATLAB
- BPR関数

$$t_a = t_0 \left(1 + 0.15 \left(\frac{x_a}{s} \right)^4 \right),$$

- まず小さいネットワークでモデルの評価をする
- 次に大きなネットワークでアルゴリズムの計算効率と収束安定性を評価する

スーフォールズネットワーク (小さい)

- アメリカ合衆国サウスダコタ州東端に位置する都市
- GitHubレポジトリ「Transportation Network for Research」から入手
- 24ノード
- 76リンク
- 528ODペア
- 需要の小さいODが多い
- 需要が大きいODもある

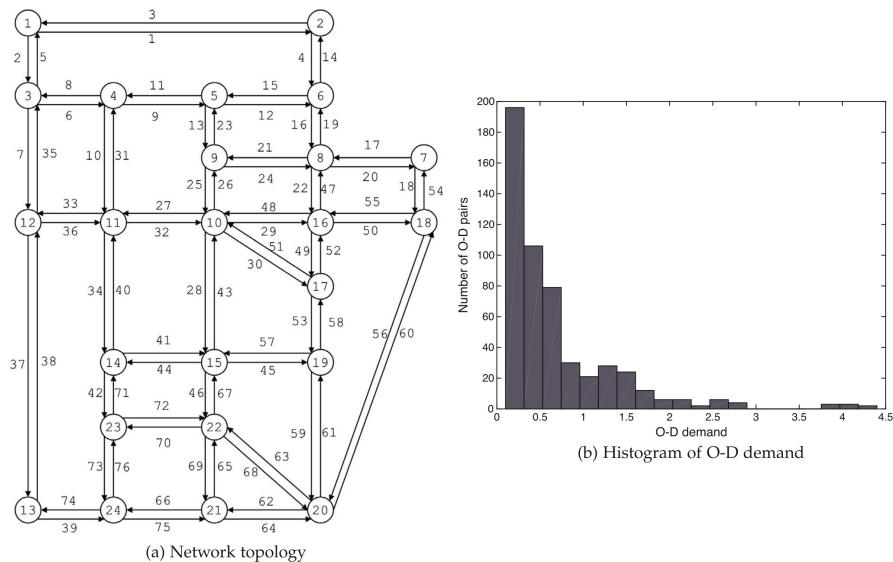


Fig. 1. Topology and demand profile of Sioux Falls network.

一様比率制御スキーム (URCS)

- URCS : Uniform-ratio control scheme
- 全てのODに対して、SOユーザーの割合を一定値にする方法
- その割合は、経路ガイドの市場普及率と解釈される

URCS

OD番号	制御率
1	10%
2	10%
3	10%
⋮	⋮

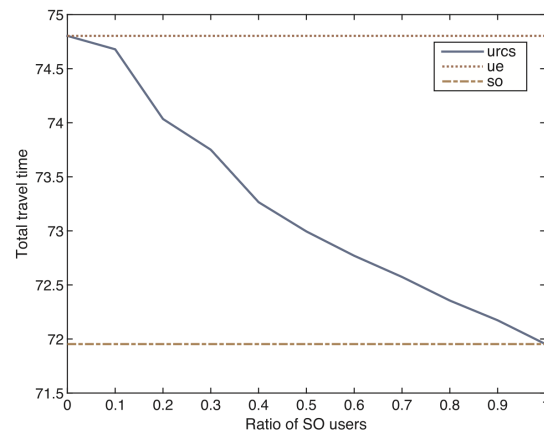
ORCS

OD番号	制御率
1	30%
2	10%
3	1%
⋮	⋮

- URCSは全ODで共通の制御率
- ORCSはODごとに制御率が異なる

一様比率制御スキーム (URCS)

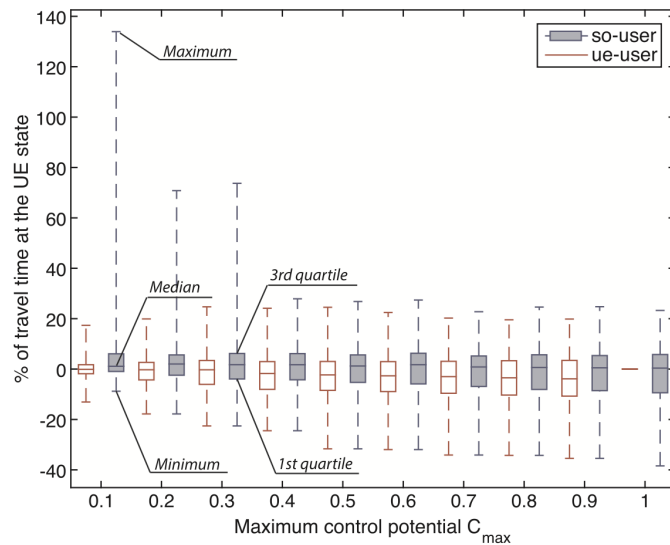
- URCSにおいては、SOユーザーの割合が増えると総旅行時間は単調に減少する



(a) Total system travel time

一様比率制御スキーム (URCS)

- 全員がUEの状態と比較した、制御率別の個人の旅行時間分布
- 制御率が小さいと、SOの方がUEよりも大きく所要時間が増加する傾向
- 最悪のケースでは、一部のSOの旅行時間が130%以上も増加
- 最悪の場合の旅行時間は、制御率が上がると急速に低下
- 40%以上がSOになると、最大の旅行時間増加はUEで20%、SOで25%程度に
- 最大の減少はSOもUEも同程度



(b) Change in travel time from the UE state

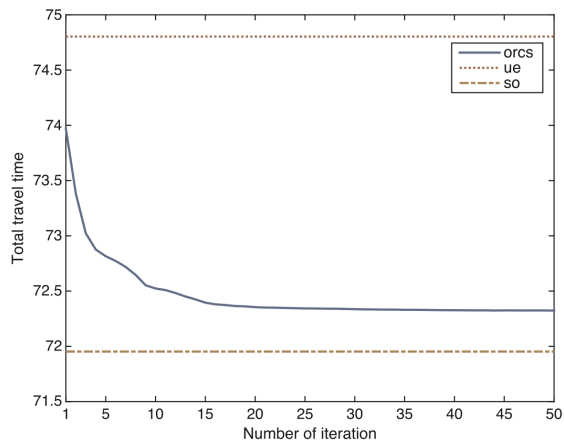
完全に制御し得るORCS

- C_{max} : 制御できる上限割合 (自動運転車の普及率)
- $C_{max} = 1$
- その気になれば全員制御することもできる
- 現実的ではないが、提唱されたモデルの最大限のパフォーマンスを評価することができる
- 制御係数 $\gamma = 0.1$

$$\min \quad \gamma \|\tilde{\mathbf{q}}\|_1 + z_2(\mathbf{x}^*)$$

完全に制御し得るORCS

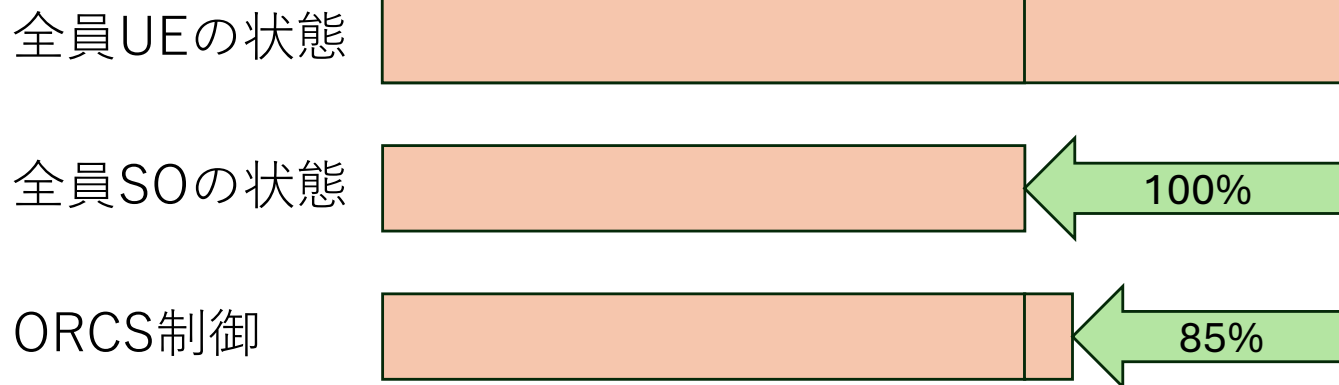
- 20回以内の反復で収束
- システム最適になった場合の総旅行時間短縮量に対して85%の旅行時間短縮を達成



(a) Total system travel time

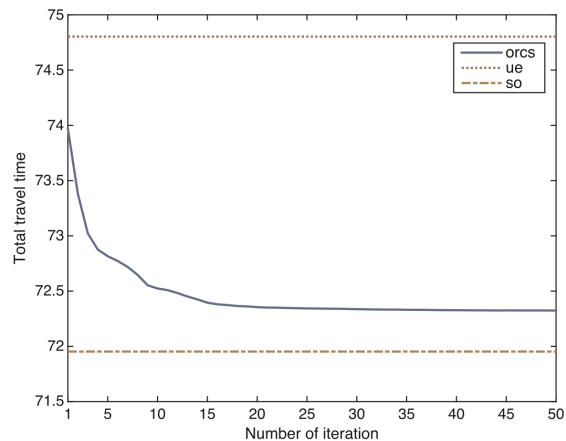
完全に制御し得るORCS

総旅行時間

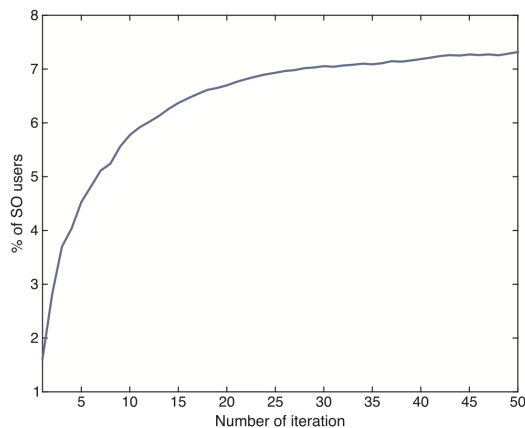


完全に制御し得るORCS

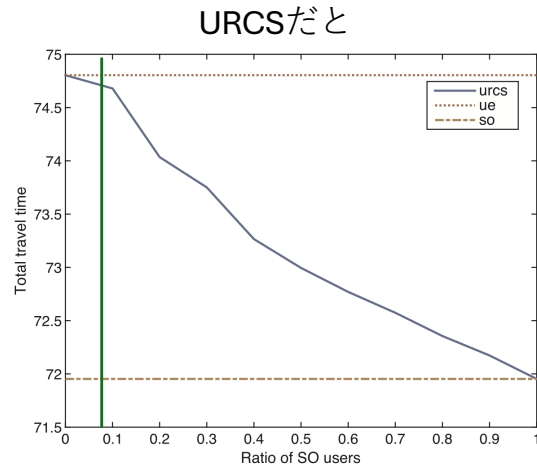
- 20回以内の反復で収束
- システム最適になった場合の総旅行時間短縮量に対して85%の旅行時間短縮を達成
- わずか7.5%以下の車しか制御していない



(a) Total system travel time



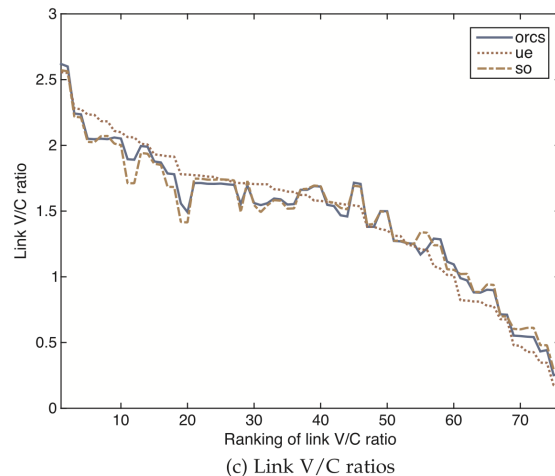
(b) Percentage of the controlled demand



(a) Total system travel time

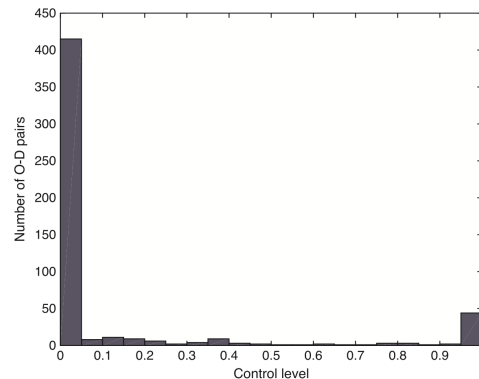
完全に制御し得るORCS

- キャパシティに対するボリューム (V/C)
- 全員がUE状態の場合のランキングで並び替え
- ORCSの場合は全員SOの場合とかなり近い
- SOとORCSは、混雑しているリンクから交通量を減らし、空いているところに増やし、バランスをよくしている



完全に制御し得るORCS

- 制御強度の分布
- 528種類中400種類以上のODペアは制御率が5%未満
- 40種類は95%以上制御
- トポロジー特性のせいでODペアによって混雑への影響の仕方が違うので不均等に分布

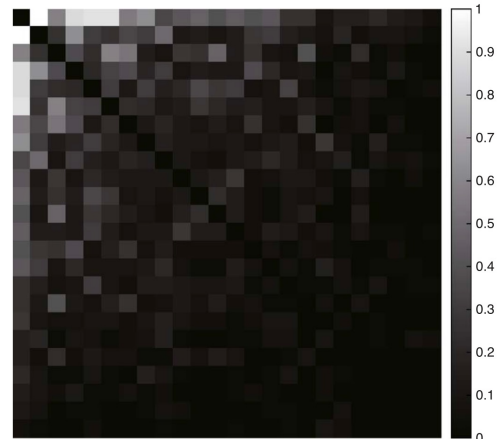


(d) Distribution of O-D control level

完全に制御し得るORCS

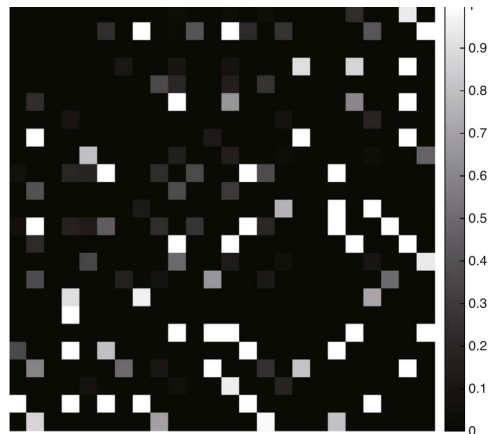
- ODを同じ順番にプロット
- 明るいほど需要や制御率が高い
- 需要の大きさと制御率に相関はない
- トポロジー特性が、各ODがどれくらい強く制御されるかを決定づけている
- l_1 ノルム定式化の影響で、スパースな解を求めるために、ORCSは需要の大きさに関係なく削減効果の大きいODを制御する傾向

需要



(e) Grey-scale map of O-D demand

制御率

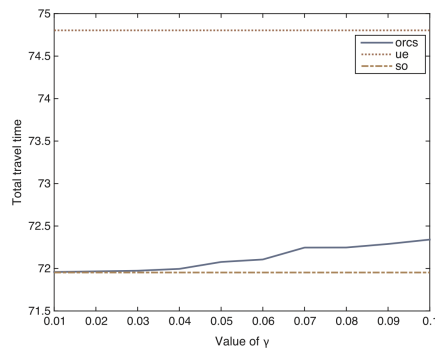


(f) Grey-scale map of O-D control level

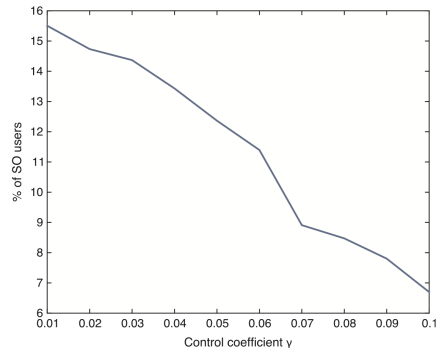
制御係数 γ の影響を分析

$$\min \gamma \|\tilde{\mathbf{q}}\|_1 + z_2(\mathbf{x}^*)$$

- γ を大きくすると、総旅行時間は増え、制御は減る
- γ を小さくすると、総旅行時間は減り、制御は増える
- $\gamma = 0.01$ の時、システム最適と比較して0.014%しか総旅行時間は長くなかった
- その時制御されていた車は15.5%



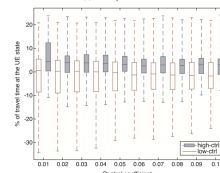
(a) Total system travel time



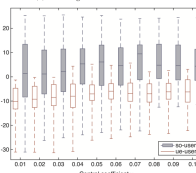
(b) Percentage of the controlled demand

制御係数 γ の影響を分析

- γ の値が異なる場合のUEとSOの個別旅行時間の分布
- 4つのグループに分類
- (1) : 95%以上が制御されているODのSOユーザー
- (2) : 5%以下しか制御されていないODのUEユーザー
- (3) : 5%~95%制御されているODのSOユーザー
- (4) : 5%~95%制御されているODのUEユーザー



(c) Comparison of travel time change (high-ctrl: SO users from O-D pairs at a high control level; low-ctrl: UE users from O-D pairs at a low control level)

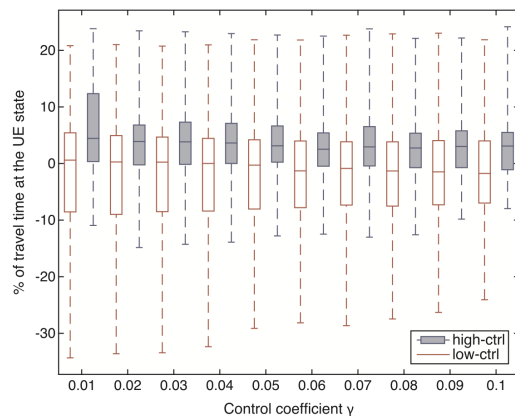


(d) Comparison of travel time change (so-user and ue-user are both in the O-D pairs at a medium control level)

制御係数 γ の影響を分析

- 1と2を比較
- グループ2の利用者の約半数が全員UEの状態と比べて短い旅行時間を享受
- グループ2の合計の削減効果は10%以下
- 75%以上のグループ1の利用者は旅行時間が増加した（最大で10%）
- γ の値が大きくなると、グループ1の損失は小さくなる傾向があり、特に最悪の場合の旅行時間は著しく小さくなった
- しかし、ごくわずかに、20%以上旅行時間が伸びた

※(1)：95%以上が制御されているODのSOユーザー
(2)：5%以下しか制御されていないODのUEユーザー



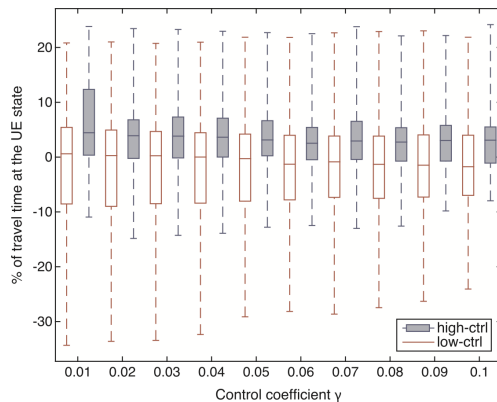
(c) Comparison of travel time change (high-ctrl: SO users from O-D pairs at a high control level; low-ctrl: UE users from O-D pairs at a low control level)

制御係数 γ の影響を分析

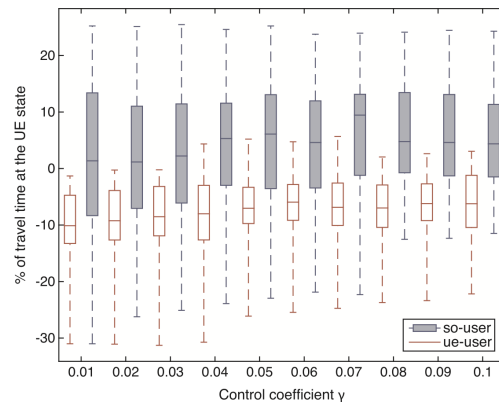
- 3と4を比較
- 3と4の差は、1と2の差よりもずっと大きい
- 3は1と似ている
- 4は他の3グループと比べて著しく大きい所要時間短縮を享受
- グループ4合計で、平均5~10%所要時間短縮

※(3) : 5%~95%制御されているODのSOユーザー

(4) : 5%~95%制御されているODのUEユーザー



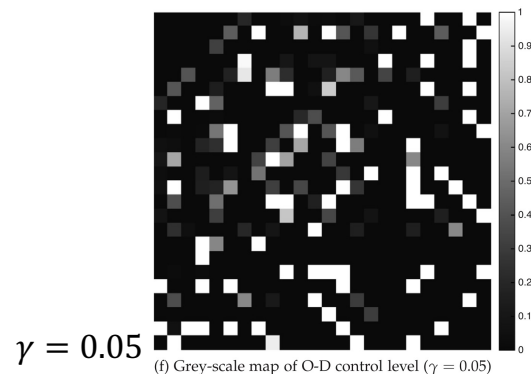
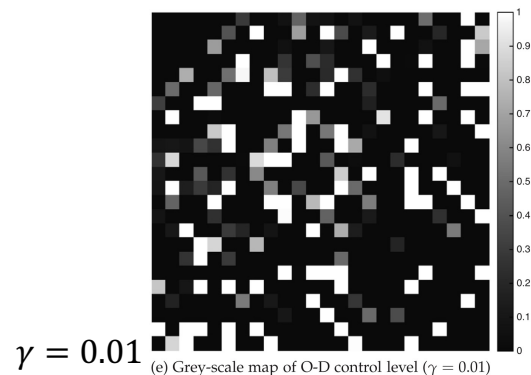
(c) Comparison of travel time change (high-ctrl: SO users from O-D pairs at a high control level; low-ctrl: UE users from O-D pairs at a low control level)



(d) Comparison of travel time change (so-user and ue-user are both in the O-D pairs at a medium control level)

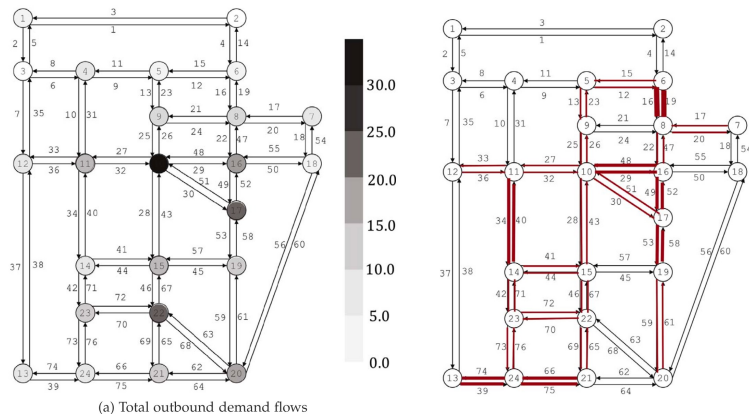
制御係数 γ の影響を分析

- γ の値が違う時の各ODの制御率
- γ が大きい時は一部のODペアに集中し、 γ が小さい時は多くのODペアを少しずつ制御する
- 制御率のパターン（制御されやすい・されにくい）は似ている
- γ が減少すると、需要の大きいODペアの制御率がより顕著に上がる
- そのようなODペアは所要時間短縮のポテンシャルが高いから



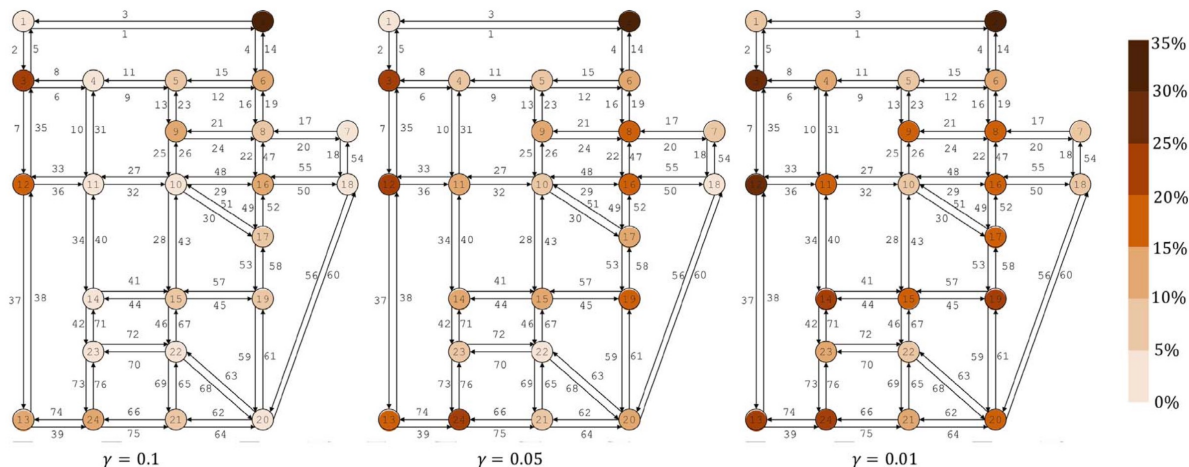
制御強度の空間的分布

- どんなODが優先される？
- 出発する需要が大きいノード：10,16,17,22
- ※スーフォールズネットワークは対称なので、出発する需要と到着する需要は等しい
- 当然、最も混雑するリンクは最も混雑するノードの近傍



制御強度の空間的分布

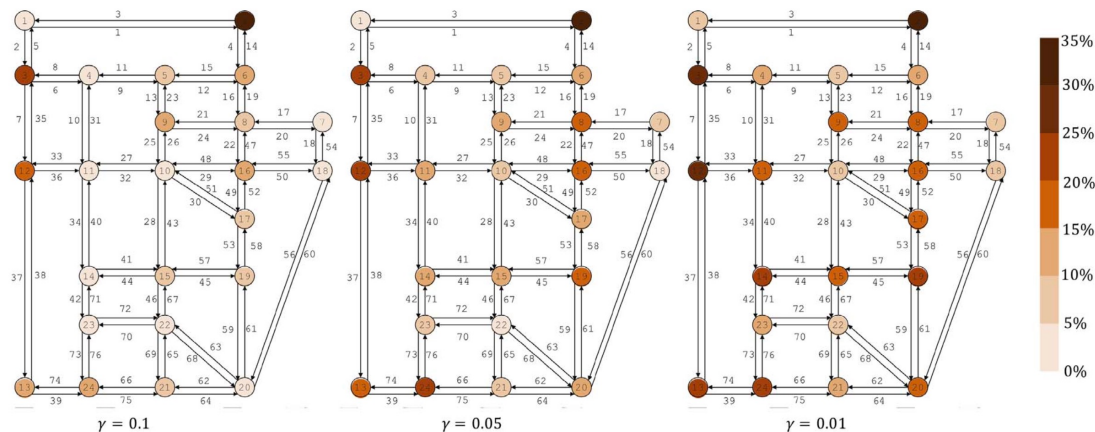
- 3種類の異なる γ でのORSC下での各ノードから出発する需要のうち制御された割合
- 最も制御されているノードは2,3,12,13
- いずれのノードも出発する需要は大きくない



(c) Ratios of controlled outbound flows

制御強度の空間的分布

- いずれのノードも、UE状態で見た時に混雑した核となるエリアの外側だが、近い場所に位置している
- これらのノードから発生する需要は混雑を緩和するために核となるエリアを避ける猶予がある
- 一方で10や16など核に含まれるノードは核を避けにくい



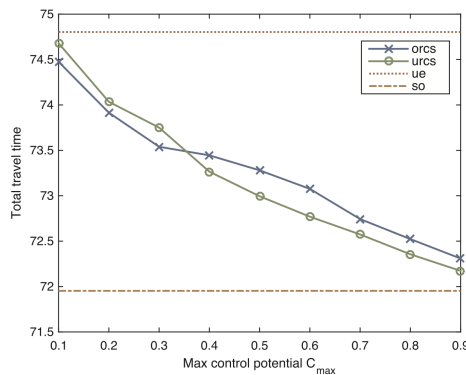
(c) Ratios of controlled outbound flows

制御上限のある場合の性能

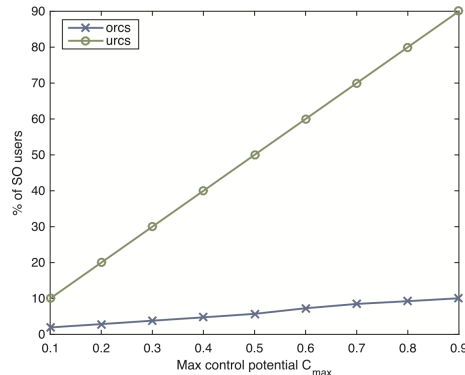
- ここまでは $C_{max} = 1$
- C_{max} の値の変化による性能の変化を見る
- C_{max} の値は市場の自動運転車の普及率と捉えられる
- → 自動運転車過渡期のORCSの性能を評価できる
- 制御係数 γ は0.05とする

制御上限のある場合の性能

- 同じ C_{max} のときのURCSとORCSの性能の違いの比較
- URCSは C_{max} までフルで制御する
- ORCSはURCSよりはるかに少ない台数でほぼ同じ性能
- $C_{max} = 0.9$ のとき、URCSに対して94.8%の短縮、11.2%の制御
- $C_{max} < 0.3$ だとORCSの方が性能が良い



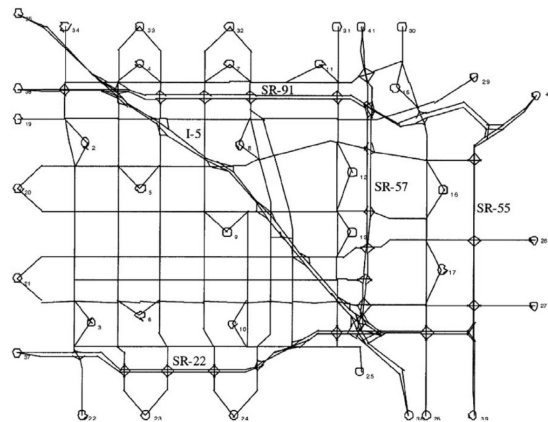
(a) Total system travel time



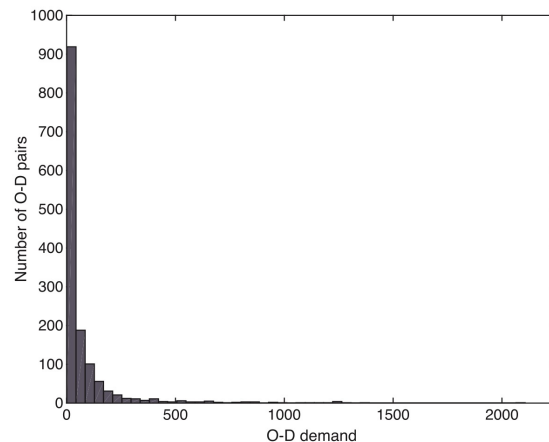
(b) Percentage of the controlled demand

大きなネットワークでの実験

- アナハイムネットワーク
- アメリカ合衆国カリフォルニア州に位置する都市
- GitHubレポジトリ「Transportation Network for Research」から入手
- 416ノード
- 914リンク
- 1406 ODペア



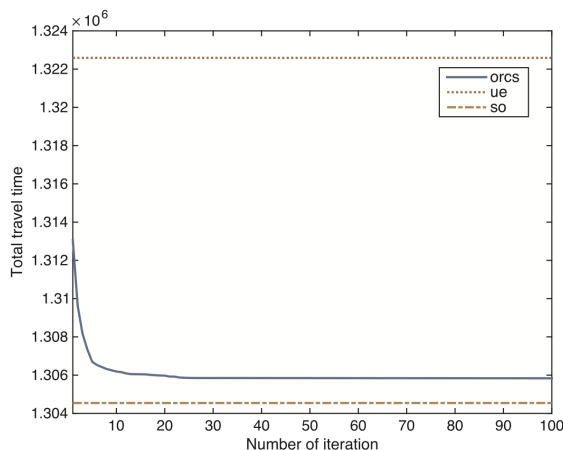
(a) Network topology



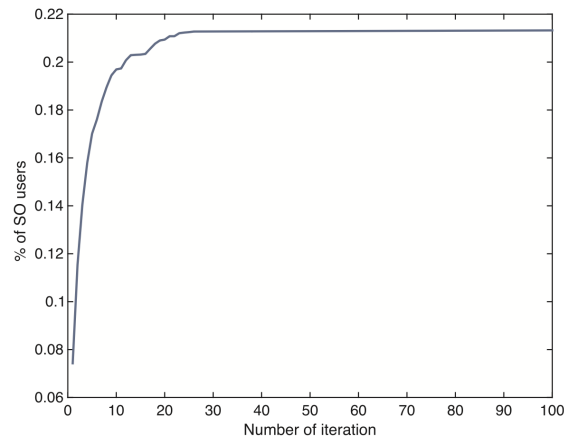
(b) Histogram of O-D demand

大きなネットワークでの実験

- $C_{max} = 1$ 、 $\gamma = 0.1$ の場合の結果
- 反復30回以内で収束
- 21.4%の車を制御
- 91.7%の総所要時間削減



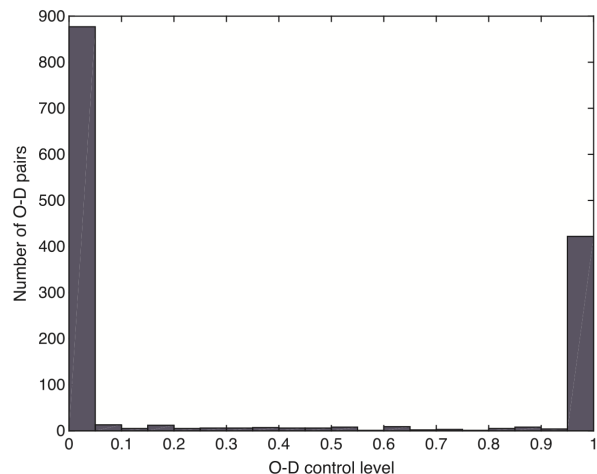
(a) Total system travel time



(b) Percentage of the controlled demand

大きなネットワークでの実験

- 約30%のODは95%以上制御されていたのに対して、60%は5%未満
- 需要量と制御率に相関はない
- →スーフォールズでの結果と同じ



(c) Distribution of O-D control level



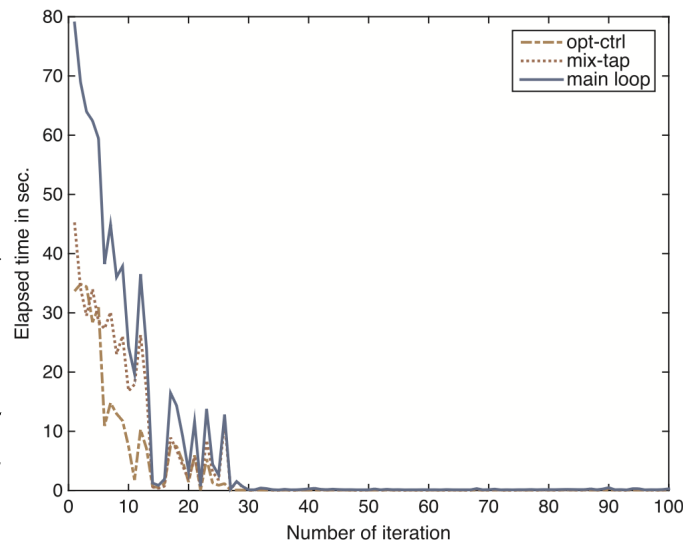
(e) Grey-scale map of O-D demand



(f) Grey-scale map of O-D control level

大きなネットワークでの実験

- アナハイムネットワークを解くのにかったCPU時間
- 上層下層どちらの部分問題を解くためにかかるCPU時間も、20回程度の反復で急速に低下し、その後は1サイクル1秒程度に
- 解が局所最適解に近づくほど、両方の部分問題は簡単に解くことができる
- 両者の解は全体のモデルが局所最適解に近づくにつれて互いに局所最適に近い初期解を提供し合うから
- 総計算時間：700秒



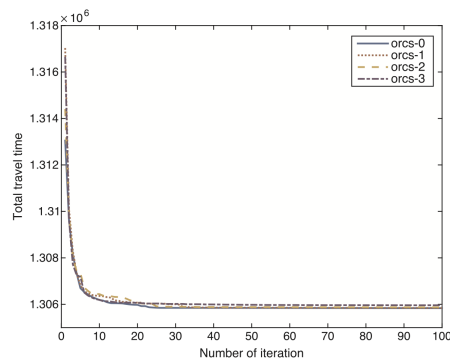
(d) Elapsed time of iterations

大きなネットワークでの実験

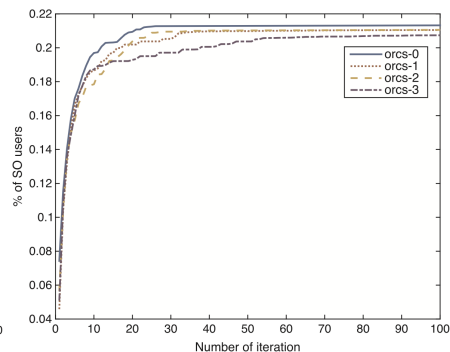
- 大規模なので、複数の局所最適解があると思われる
- 最初のSOとUEの比率を変える
- それぞれのODペアごとにSOユーザーの割合が0~0.5の範囲でランダムに決定される

大きなネットワークでの実験

- 異なる3種類の初期比率から初めて、収束後の結果
- 総旅行時間、全体の制御率、ODごとの制御率はほぼ同じ
- わずかな違いはあるが、別の似ている局所最適解だと解釈できる
- 全く異なる解から始めてもほぼ同じ結果が得られる
- 実用上は無視できる



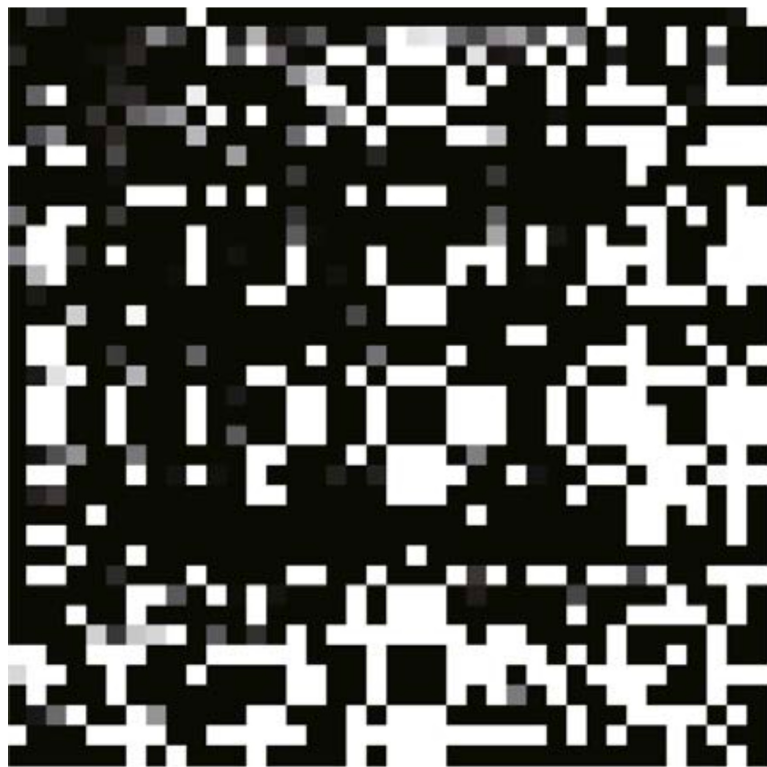
(a) Total system travel time



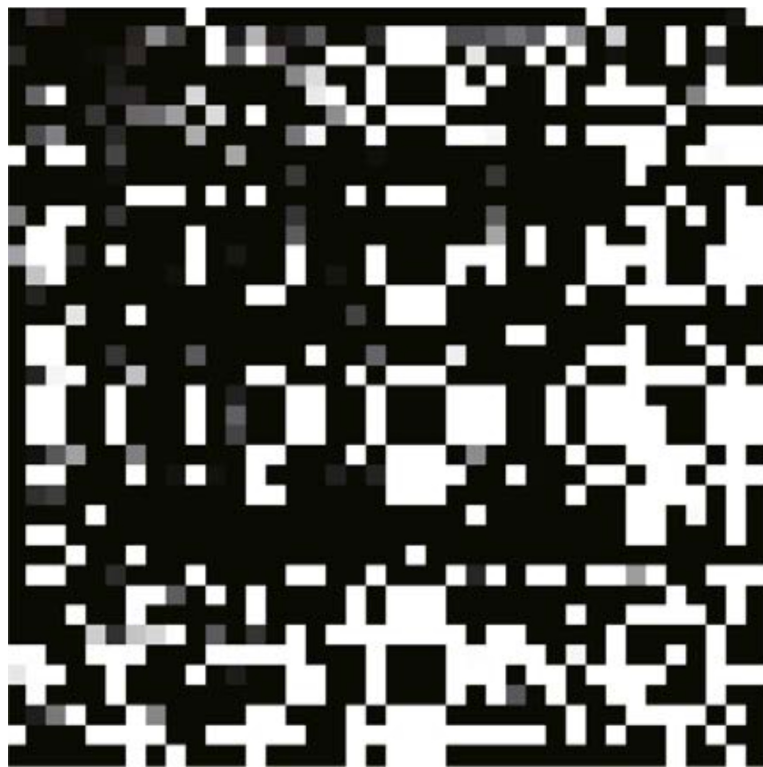
(b) Percentage of the controlled demand



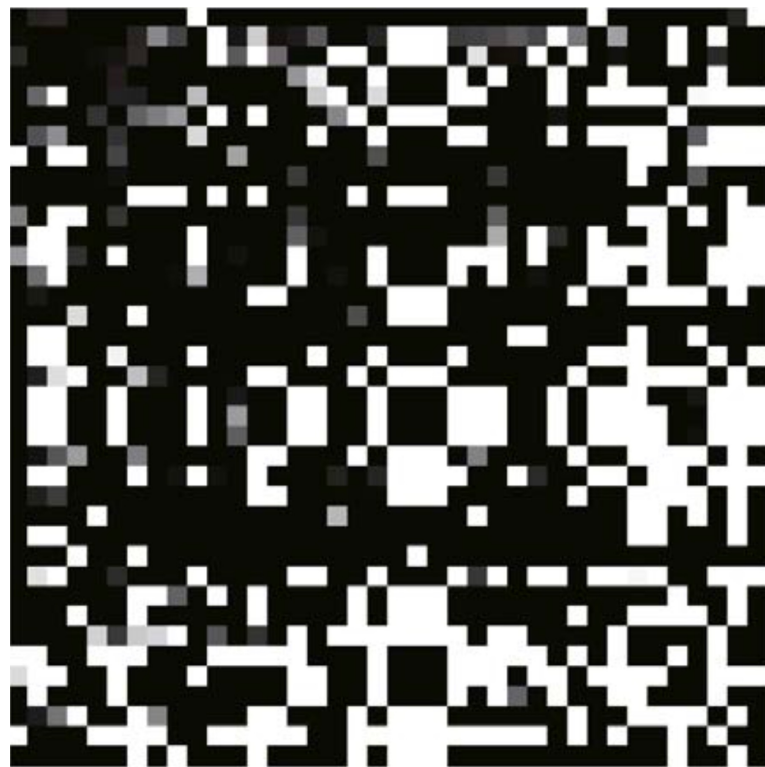
初期条件の違いの比較 1



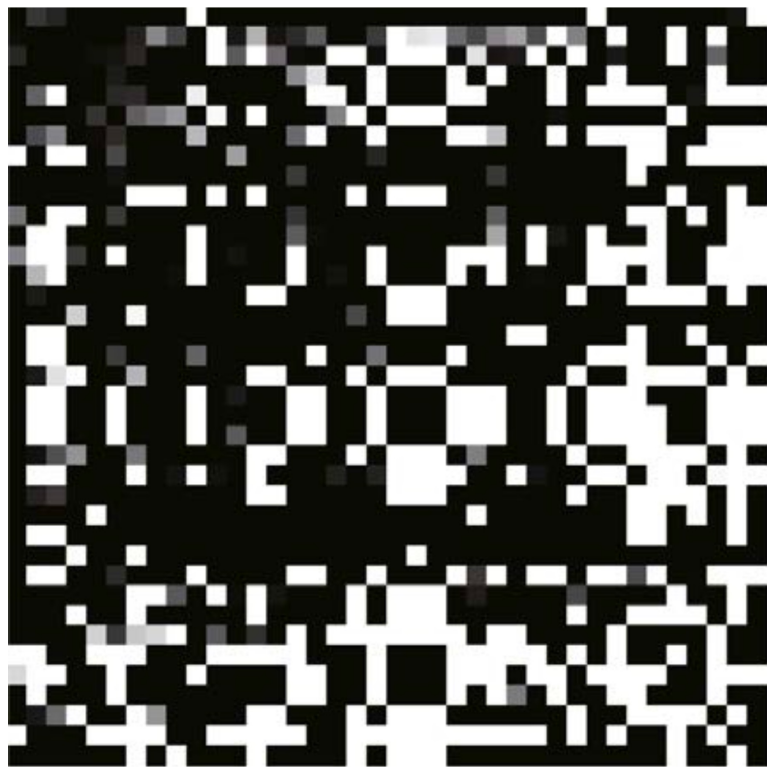
初期条件の違いの比較 2



初期条件の違いの比較 3



初期条件の違いの比較 1



Section6 結論

結論

- ORCSは2層構造
- ごく一部の車を制御するだけで、全体を大きく改善できる
- 需要量と制御率に相関はない
- 優先的に制御されるのは、混雑している近くだが外側の地点
- 感度分析に基づく勾配近似と、交互方向乗数法は計算上優れている
- 初期解が違っててもほぼ同じ局所最適解に落ち着く

結論

- 将来の研究
- 異なるトポロジカル特性を持つネットワークで試す
- UEとSOの現実との違いを調査する
- ORCSが個人の旅行時間に与える影響を考慮しながら、自動運転車の普及プロセスをモデル化する
- 異なる制御戦略に関して自動運転車の市場普及率がどう変化するかを測る

所感

- 10%前後の制御でほぼシステム最適になるのが面白いと思った
- でも自分の車がたまたま制御される側になったら嫌
- 無人の配送トラックとかなら個人としての旅行時間が伸びても不公平感はなさそう
- 現実にはたまたま特定のODに、全体の市場普及率より少ない自動運転車しかない場合もありそうだから数値設定に余裕が必要？
- システム最適と比較して0.014%：なぜここだけ比較に仕方が違う？

Appendix

Appendix1

上層

ODごとに最適なUEとSOの比率を求める



下層

混合均衡交通配分問題の解を求める

- 変分不等式の証明
- リンクコスト関数は単調増加で下凸（仮定1より）
- よって、最適化問題(3)と(4)はともに狭義凸で、(5)(6)と同値
- Ω_1 と Ω_2 は互いに素（UEでありSOでもある利用者はいない）なので、(7)式一本にまとめることができる
- 利用者の合計は決まっていて有限なので、全てのリンクフローは上に有界
- それゆえ Ω はコンパクトで閉じた集合
- Kinderlehrer and Stampacchiaの定理3.1より、(7)への解法は存在する

Appendix1

上層

ODごとに最適なUEとSOの比率を求める



下層

混合均衡交通配分問題の解を求める

- 変分不等式の証明
- 混合交通均衡状態におけるUEとSOの統合フローパターンや統合されたリンクフローの一意性は、仮定1より保証されている

Appendix2

上層

ODごとに最適なUEとSOの比率を求める

下層

混合均衡交通配分問題の解を求める

変分不等式
制約

- 定理2の証明
- V_1^n と V_2^n をそれぞれn回反復後のUEとSOの経路選択を表すパス-OD対応行列とする。 $\tilde{q}$ によるリンクフローパターンの変化は次のようになる。

$$\chi(\tilde{\mathbf{q}}) = (\mathbf{x}_1^n - \Delta V_1^n \tilde{\mathbf{q}}) + (\mathbf{x}_2^n + \Delta V_2^n \tilde{\mathbf{q}}).$$

Appendix2

上層

ODごとに最適なUEとSOの比率を求める

下層

混合均衡交通配分問題の解を求める

変分不等式
制約

- $\nabla \mathbf{z}_2^n$ は次のようになる

$$\nabla \mathbf{z}_2^n = \frac{\partial \mathbf{z}_2}{\partial \tilde{\mathbf{q}}} = m\mathbf{t}(\mathbf{x}_1^n + \mathbf{x}_2^n)^T (-\Delta V_1^n + \Delta V_2^n) = mc(\mathbf{x}_1^n + \mathbf{x}_2^n)^T (V_2^n - V_1^n),$$

- $mc(\cdot)$ はリンクフローパターンの関数としてパス限界費用を表す
- $c = \Delta^T t$
- $mc = \Delta^T m\mathbf{t}$
- $mc(\cdot) = \Delta^T m\mathbf{t}(\cdot)$

Appendix3

- 本当に相関はないのか？
- グレースケールを明るさに応じて数値化
- 需要と制御強度の相関を調べた
- 相関係数：-0.140

