

# Network-oriented household activity pattern problem for system optimization

Jiangtao Liu, Jee Eun Kang, Xuesong Zhou, Ram Pendyala,  
Transportation Research Part C: Vol.94, pp. 250-269, 2017

---

B4 清水大暉

# 目次

---

1. はじめに
2. 扱う各ケースの設定
3. 定式化
4. 数値計算
5. まとめ

# 1. はじめに

---

## 背景

- ・ マルチモードの移動提案・ユーザーのスケジューリングを行うモバイルアプリの登場

ex)Google Map, Yahoo!乗換案内, 各種リマインダー・カレンダーアプリ

- ・ UberやLyftなど配車サービス, 自動運転車の登場

⇒ 旅行者が自らの活動を賢くスケジュール可能に  
世帯レベルの活動のより最適なプランニングが可能に

- ・ Recker(1995)を端緒とするHAPP(household activity pattern problem)に焦点を当てる

### HAPP

- ・ 利用可能な車両数
- ・ 活動予定(場所・時間)
- ・ ライドシェアオプション

の制約に基づき, 所定の活動を完遂する際の目的関数を最適化するように世帯の1日の活動パターン(最適経路)を決定するモデル

# 1. はじめに

---

## 既往の研究-HAPPの定式化

**Recker (1995, 2001), Recker et al. (2001), and Gan and Recker (2008)**

・ PDPTW(Pickup and Delivery Problem with Time Windows)における従来の混合線形整数問題に基づき, HAPPにおいては非常に多くの線形整数制約を要していた

**Chow and Recker (2012) and Kang and Recker (2013)**

・ より現実に即した制約条件, より大きなデータを扱うアルゴリズムを提案

**Liao et al. (2013)**

・ 多次元ネットワーク構造(移動リンク, 状態遷移リンク, 活動遷移リンク)における個人レベルの活動スケジューリング問題を提案

HAPPを数学的に厳密なモデルとして定式化するためには

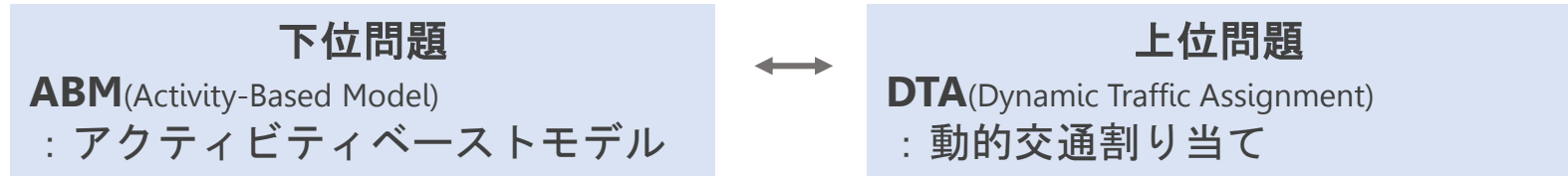
—特に大規模かつマルチモーダルな交通ネットワークにおいてライドシェア問題や世帯メンバー間の相互作用(協同)をも扱う場合には—

世帯メンバー・車両・多種の活動の3者間が絡み合う複雑な制約を

完全に考慮することがとても困難(extremely challenging)だった

# 1. はじめに

## 既往の研究-ABMとDTAの統合



HAPPにおいては、人間の活動-移動選択と、  
厳しい容量制約かつ混雑したネットワークとの相互作用を  
より正確に記述するため、ABMとDTAの統合が必要

### Kang et al. (2013)

- ・世帯の活動パターンとインフラの変化の間の相互作用を考慮したネットワークデザイン問題

### Fu et al. (2016)

- ・世帯内の相互作用をマルコフ決定過程と考え、道路混雑による影響を移動時間の関数と考える

# 1. はじめに

---

## この論文の狙い

- ①HAPPを、効率的な計算アルゴリズムを持つ、離散化された空間-時間ネットワーク(時空間ネットワーク)、或いは高次の空間-時間-状態ネットワーク(時空状ネットワーク)における多数の『時間依存・状態依存の経路探索問題』と考える
- ②活動生成と活動スケジューリングに対する交通混雑の影響を捉えるため、HAPPの特殊な2つのケースを、厳しい交通容量制約を持つ多世帯のシステム最適(本論文では目的関数：世帯の総移動コストを最小化)な活動スケジュール問題として改めて定式化する



従来のモデルの制約を正確にネットワークに表現し変換するために、  
FSM(Finite State Machines)をネットワークに  
どう事前に構築する(prebuild)かが重要  
→活動時間制約や車両選択制約などの制約式を削減

FSM

：有限オートマトン、有限状態機械。  
有限個の状態と遷移と動作の組み合わせからなる数学的に抽象化された「ふるまいのモデル」

## 2. 扱うHAPPのケース

---

### 扱うケース

- ・ ケースA, B
- ・ それらの拡張のケースC(ほぼ触れない)
- ・ より一般的なケースD(ほぼ触れない)

### 入力(所与とするもの)

- ・ 活動生成に使用される世帯のメンバー数
- ・ 実交通ネットワーク
- ・ 特定の時間窓と効用値を持つ様々な種類の活動(義務的/半義務的/選択的)
- ・ 車両群
- ・ 世帯の各メンバーへの活動/車両割当て

## 2. 扱うHAPPのケース

---

### ケースA

- ・ 義務的&選択的活動を含む複数車両・複数人の車両ルーティング問題

(Recker(1995)のケースIVに類似)

- (1) 複数車両は世帯のメンバーによって共有される
- (2) 一部の活動は世帯の任意のメンバーによって行われ、残りの活動は特定のメンバーによって行われなければならない
- (3) 特定のメンバーは特定の義務的活動または選択的活動を行う
- (4) 一部のメンバーは活動を行わない、すなわち使用されない車両がある可能性がある



容量制約なし・1世帯

1台に1人

(世帯メンバーが各自車両を選び、各々の活動を行う)



## 2. 扱うHAPPのケース

---

### ケースB

- ・ 義務的&選択的活動を含む複数車両・複数人のライドシェアリング問題

(Recker(1995)の特殊な場合のケースVに類似)

- (1) ライドシェアリングパターン(乗る世帯メンバー, ドライバーになる世帯メンバー)が所与
- (2) 一部の活動は世帯の任意のメンバーによって行われ, 残りの活動は特定のメンバーによって行われなければならない
- (3) 特定のメンバーは特定の義務的活動または選択的活動を行う



容量制約なし・1世帯

1台に複数人

(世帯メンバーが1つの車両に乗り, 送/迎をドライバーが行うことでメンバーの活動を行う)

## 2. 扱うHAPPのケース

---

### ケースC(ほぼ触れない)

- ・ 実交通ネットワークにおける混雑を捉えるため厳しい容量制約を考えた, ケースA・Bの拡張版
- ・ 世帯の活動パターンに対する時間依存のリンク移動時間(=時間変動する道路混雑)の影響が見れるように

→時間変動する交通状況下におけるシステム最適な多世帯活動スケジューリング問題



容量制約あり・多世帯  
世帯単位

### ケースD(ほぼ触れない)

- ・ 各世帯が最適な活動パターンを選択するような, 世帯レベルの動的均衡問題
- ・ 各世帯が車両選択, 交通モード選択, ライドシェアリング選択を同時に行う(=所与としない)
- ・ "In our future study, Case D will be further examined to study possible dynamic household-level equilibriums with household activity interactions."

# 2. 扱うHAPPのケース

## 従来のABM&DTA統合モデルのデータフローとの比較

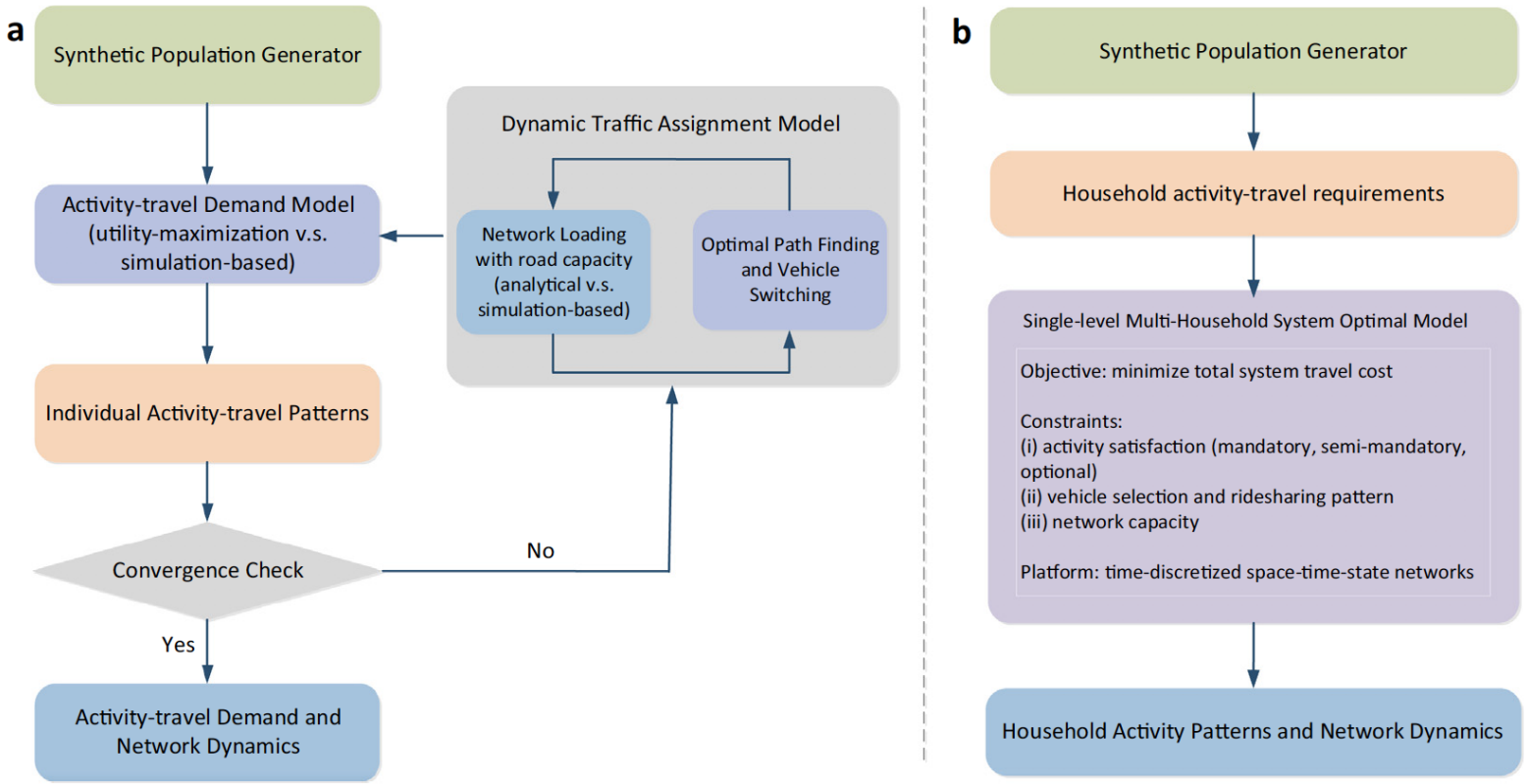


Fig. 1. (a) Existing integration framework of ABM and DTA; (b) Proposed modelling framework of Case C.

# 2. 扱うHAPPのケース

## 従来のABM&DTA統合モデルのデータフローとの比較

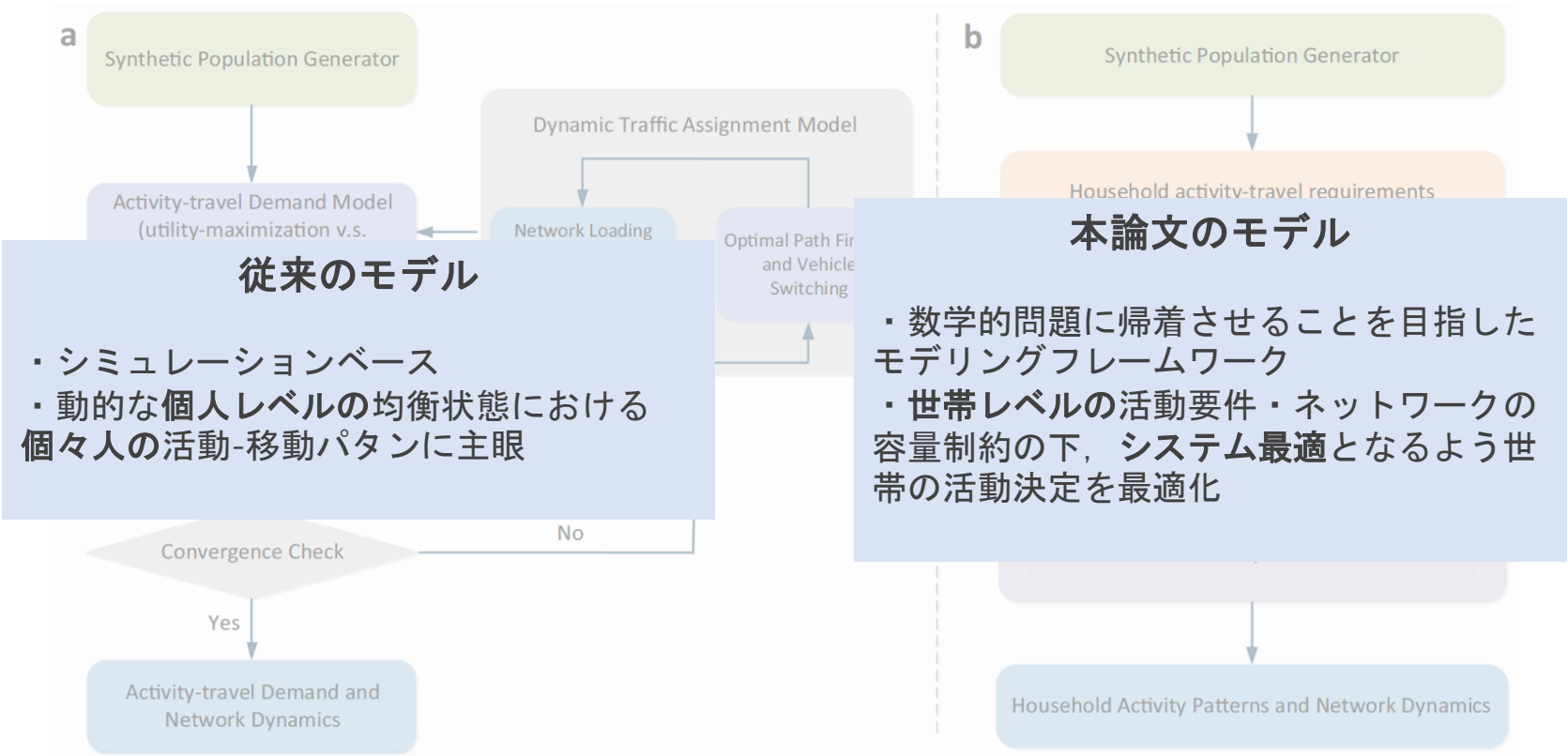


Fig. 1. (a) Existing integration framework of ABM and DTA; (b) Proposed modelling framework of Case C.

## 2.1. ケースA | ネットワーク構築と概念図

---

### 設定

- $p_1, p_2$  : 世帯メンバー
- $v_1, v_2$  : 利用可能な車両
- $a_1, a_2$  : 義務的活動(必ず行う)
- $p_1$  →  $\{v_1, v_2\}, \{a_1, a_2\}$ が利用可能
- $p_2$  →  $\{v_2\}, \{a_1\}$ のみ利用可能

→ これらを組み込んだネットワークを作成

# 2.1. ケースA | ネットワーク構築と概念図

設定	
• $p_1, p_2$	: 世帯メンバー
• $v_1, v_2$	: 利用可能な車両
• $a_1, a_2$	: 義務的活動(必ず行う)
• $p_1$ が利用可能	$\rightarrow \{v_1, v_2\}, \{a_1, a_2\}$
• $p_2$ が利用可能	$\rightarrow \{v_2\}, \{a_1\}$ のみ利用可能

## ネットワーク | 時空間ネットワーク(二次元)

楕円部 : 車両選択リンク(D : super destination nodeも一つの選択肢)

☆個人毎の選択可能な車両・車両選好, リンク毎の所要時間・コストをネットワークに反映

☆道路混雑はリンク毎の所要時間として所与 (⇔ケースC)

☆活動は開始ノードと終了ノード間の活動リンクで表現

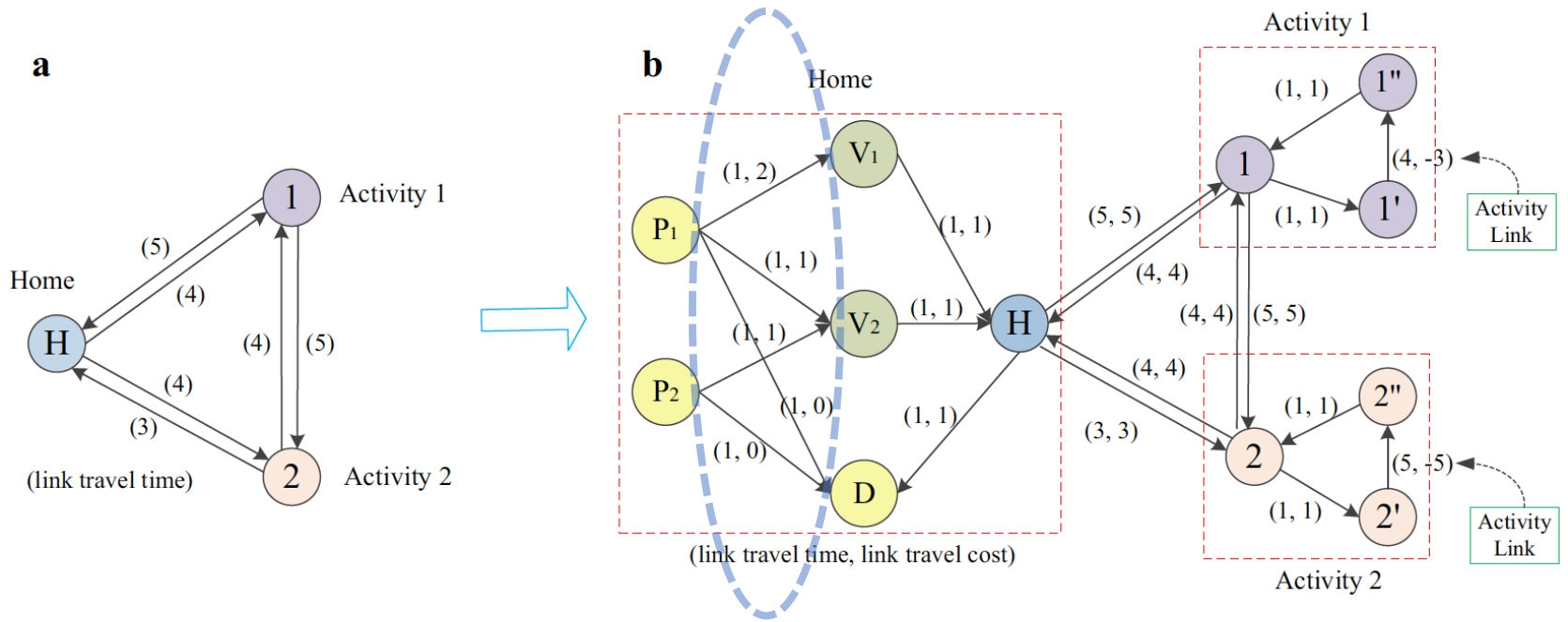


Fig. 2. (a) The physical network; (b) the corresponding modified network.

# 2.1. ケースA | ネットワーク構築と概念図

## 可能な時空間プリズム

- ・ 離散化された32の時間単位
- ・ 各々の活動には特定の開始時間窓(可能な開始時間の幅)

さっきのネットワークでは移動コストのみ設定されていたが...

- ・ Home(P1~D)での待ちコスト : 0, Activity(1,2)での待ちコスト :  $1/time$

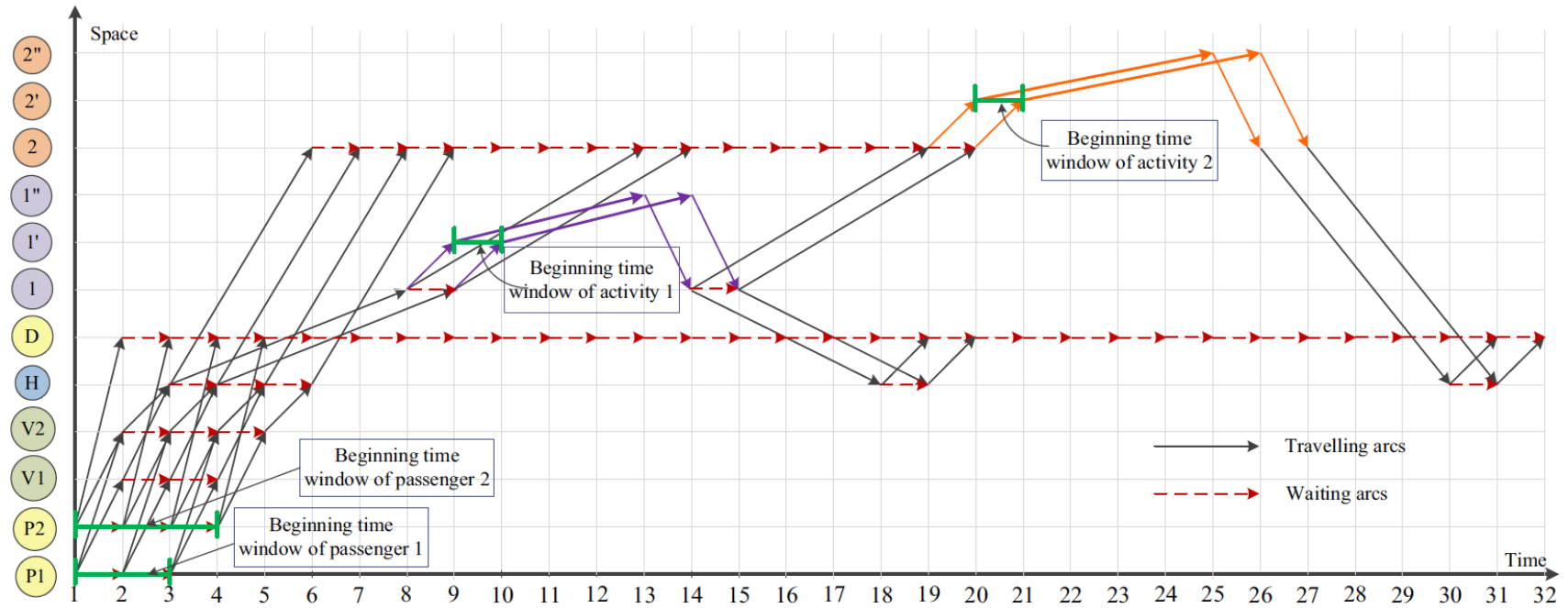


Fig. 3. Feasible searching region in the space-time network.

# 2.1. ケースA | ネットワーク構築と概念図

## 可能な時空間プリズム

時間窓と活動要件を満足するこの時空間プリズム内において、  
 世帯の全メンバーの総移動コストを最小化する経路を探せば良い  
 = 最小コスト経路探索問題

・ Home(P1~D)での待ちコスト：0, Activity(1,2)での待ちコスト：1/time

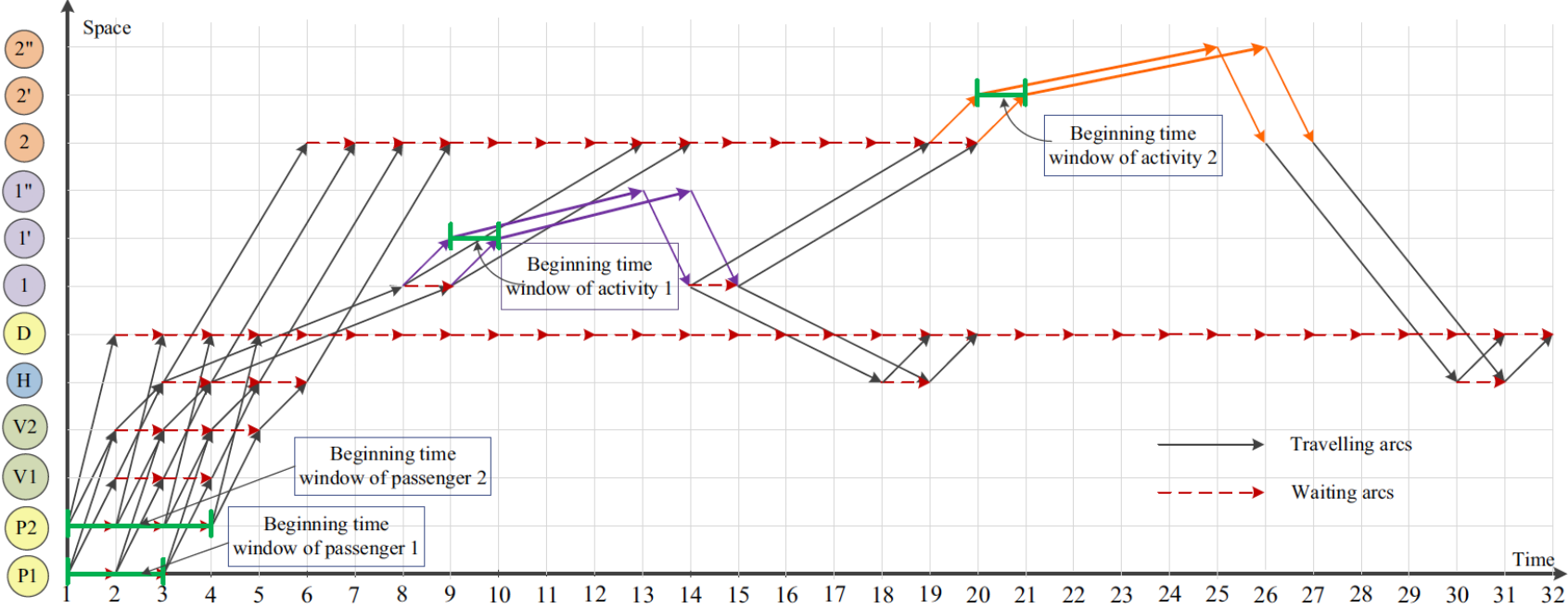


Fig. 3. Feasible searching region in the space-time network.



## 2.2. ケースB | ネットワーク構築と概念図

### 設定

☆可能なライドシェアリングパターン(pick-up/drop-offの時間窓・場所)は所与

実際はこの“世帯メンバー間の調整”が、世帯レベルのライドシェア問題のモデル化には難題

☆所定の開始時間窓内に $p_1$  (D : Driver)が $p_2$  (P : Passenger)を活動場所に降ろし、所定の終了時間窓内に $p_1$ が $p_2$ を活動場所で拾う

- ・  $p_1, p_2$  : 世帯メンバー
- ・  $v_1$  : 利用可能な車両...1台のみ, 選択済
- ・  $a_1$  :  $p_1$ の義務的活動(必ず行う)
- ・  $a_2, a_3$  :  $p_2$ の義務的活動(必ず行う)

→これらを組み込んだネットワークを作成

→ケースAと同様の二次元ネットワークでいいのか？

→新たな次元の追加

## 2.2. ケースB | ネットワーク構築と概念図

### ネットワーク | 時空状ネットワーク(三次元)

- ・ “累積活動状態”次元の追加

ケースAの時空間ネットワーク(二次元)に加え, ライドシェアリングにおける車両の状態を記述するため, “累積活動状態”次元を追加(三次元)

0 : 活動が行われていない

1 : 活動が行われている = Pは活動場所で降ろされ, まだ拾われていない

2 : 活動が既に行われた = Pは活動場所で拾われた

各活動がこれを持ち, 時間窓内で起点から終点に行く(ケースA)のに加え, 活動が確実に行われるよう, 活動の状態も2を目指す(リンクが時間・空間・状態の6つで記述される)

→制約を事前に時空状ネットワークに組み込むことができ, 制約式を大幅に減らせる

→ $3^n$ 通り. 膨大?

→下記要因により, 可能な累積活動状態パターンは絞られる

- ・ 1人のPが複数の活動選択肢 → 結局2つ以上の活動は同時には行えない

- ・ 各活動のタイトな時間窓(=制限時間), 遷移選好(=リンクの負コスト)

→これらと近年の計算機開発により, 状態パターンの爆発は計算上心配ない

# 2.2. ケースB | ネットワーク構築と概念図

## ネットワーク | 時空状ネットワーク(三次元)

- ・ drop-off, pick-upノードの追加

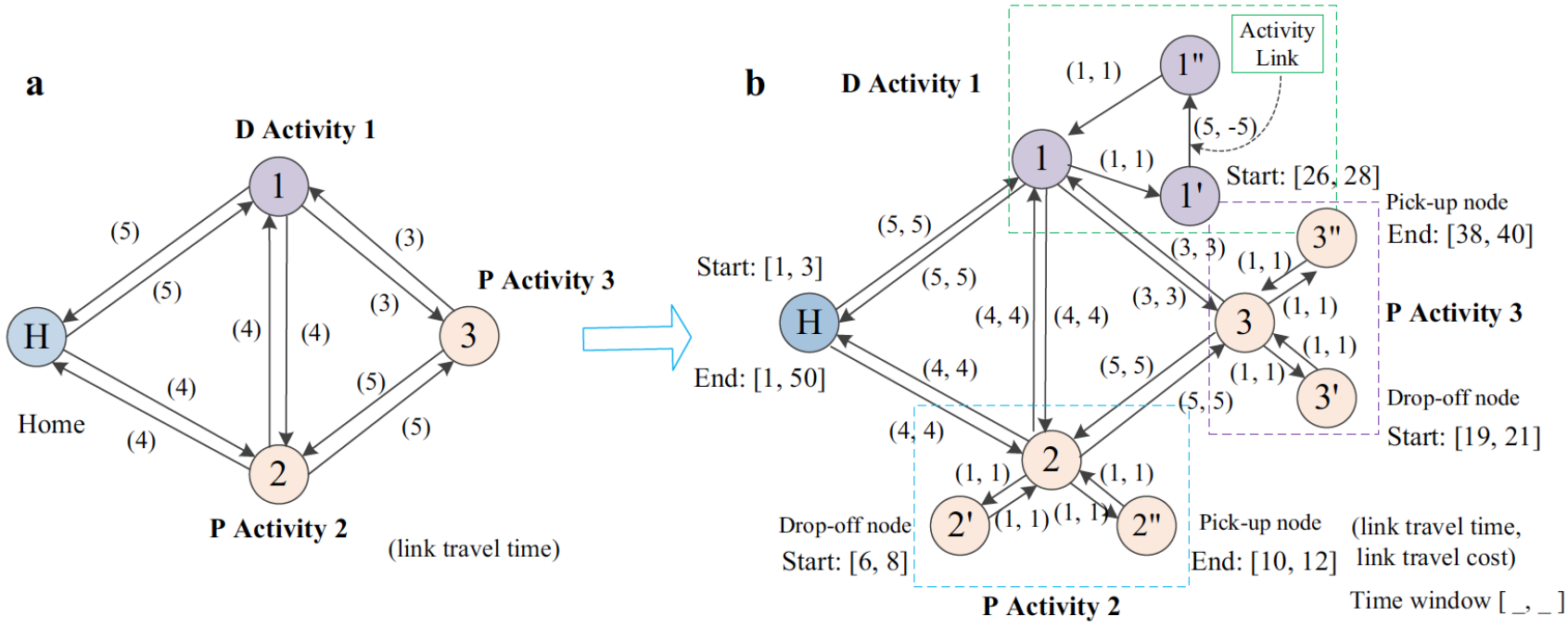


Fig. 4. (a) The physical network; (b) the corresponding modified network.

# 2.2. ケースB | ネットワーク構築と概念図

## ネットワーク | 時空状ネットワーク(三次元)

- ・ 可能な車両の状態は, 1人を $a_1$ と $a_2$ で同時に降ろす $[1,1]$ を除いた  
 $[a_2 = 0, a_3 = 0], [1,0], [2,0], [0,1], [0,2], [2,1], [1,2], [2,2]$

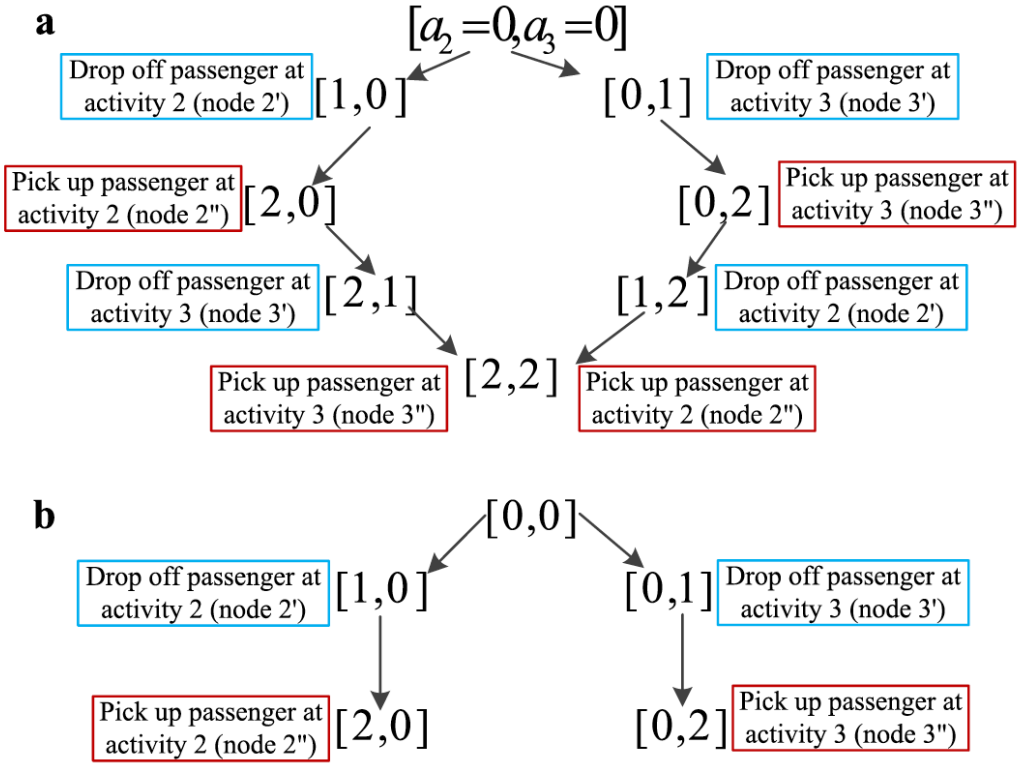


Fig. 5. (a) Both activities need to be performed; (b) exact one of two activities should be performed.

# 2.2. ケースB | ネットワーク構築と概念図

## 可能な時空状プリズム

- ・ 空間をノード2'に固定した場合の一例
- ・ ケースAと同様な待ちコストも設定

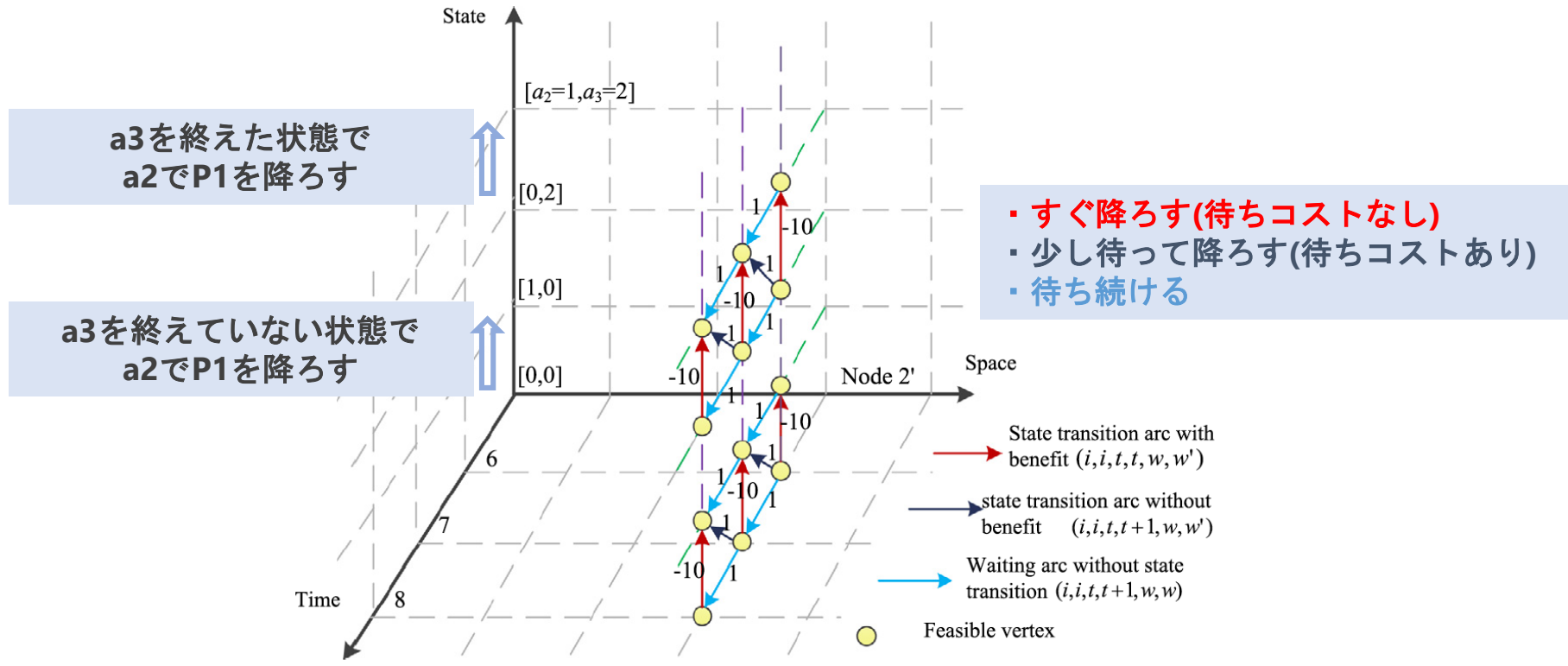


Fig. 6. Feasible arcs at node 2' in a space-time-state network.

# 2.3. ケースC | 概念図

## ケースAとケースBの拡張

・ 世帯1 :  $H1 \rightarrow A1 \rightarrow H1$ , 世帯2 :  $H2 \rightarrow A2 \rightarrow H2$

### ☆交通容量を考え、多世帯で考える

- (1) リンク容量を無視すれば、各々  $1 \rightarrow 2 \rightarrow 4 \rightarrow 2 \rightarrow 1$  を選ぶ
- (2) リンク容量を考慮すれば、世帯1が  $1 \rightarrow 2 \rightarrow 4 \rightarrow 2 \rightarrow 1$  を選べば世帯2は  $1 \rightarrow 3 \rightarrow 4 \rightarrow 3 \rightarrow 1$  を選ぶ
- (3) 世帯2の旅行時間が3から5に増えるため、時間窓がタイトなら時間窓を満たすよう出発時間を早め得る
- (4) 世帯2の時間制約が5未満なら、活動2をやめるか他の交通モードへ変更し得る

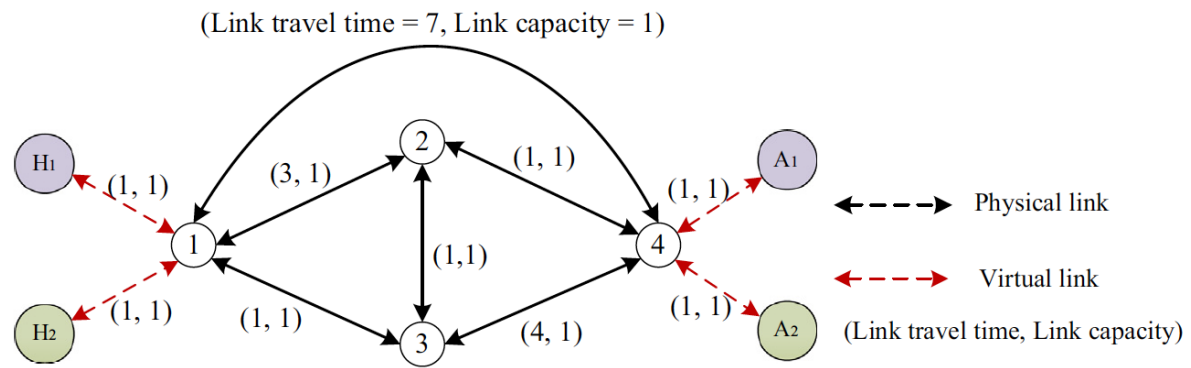


Fig. 7. A simple illustrative network for case C.

# 3.1. 定式化 | ケースA : 時空間ネットワークベースの最適化モデル

## 定式化

☆車両選択, 活動選択, 経路選択(全てリンク選択or notの二値変数)を最適化

**Model 1:**

Objective function

$$\min \sum_p \sum_{(i,j,t,s) \in E} (c_{i,j,t,s}^p \times x_{i,j,t,s}^p)$$

世帯の総移動コストを最小化

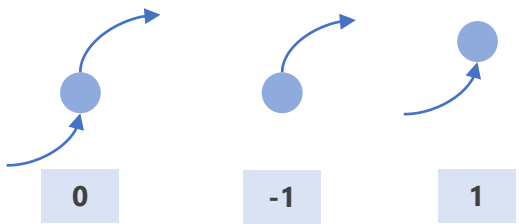
Subject to

(1) Flow balance constraint for each person:

$$\sum_{i,t:(i,j,t,s) \in E} x_{ij,t,s}^p - \sum_{i,t:(j,i,s,t) \in E} x_{ji,s,t}^p = \begin{cases} -1 & j = O(p), s = DT(p) \\ 1 & j = D(p), s = T \\ 0 & \text{otherwise} \end{cases}, \quad \forall p$$

ある時点s  
における  
ノードjへの  
全流入量

ある時点s  
における  
ノードjからの  
全流出量



$p$  : 世帯メンバー

$(i, j, t, s)$  : 時間  $t \rightarrow s$ , 空間  $i \rightarrow j$  のリンク

$E$  : リンクの集合

$c_{i,j,t,s}^p$  :  $p$  のリンク移動コスト

$x_{i,j,t,s}^p$  :  $p$  のリンク選択二値変数

$O(p)$  :  $p$  の起点ノード

$D(p)$  :  $p$  の終点ノード

$DT(p)$  :  $p$  の最速出発時刻

$T$  : 総時間

$N_v$  : 車両ノードの集合

$E(p, a_m)$  :  $p$  の義務的活動  $m$  のリンクの集合

$P_m$  : 義務的活動を持つ世帯メンバーの集合

$A_n(p)$  :  $p$  の義務的活動の候補の集合

$E(p, a_q)$  :  $p$  の選択的活動  $q$  のリンク

$P_q$  : 選択的活動を持つ世帯メンバーの集合

# 3.1. 定式化 | ケースA : 時空間ネットワークベースの最適化モデル

## 定式化

☆車両選択, 活動選択, 経路選択(全てリンク選択or notの二値変数)を最適化

(2) Vehicle selection constraint at vehicle selection node:

$$\sum_p \sum_{i,t:(i,j,t,s) \in E} x_{ij,t,s}^p \leq 1, \quad \forall j \in N_v$$

使える車両は最大1人

(3) Mandatory activity participation for one specific household member:

$$\sum_{i,t:(i,j,t,s) \in E(p,a_m)} x_{ij,t,s}^p = 1, \quad \forall p \in P_m$$

義務的活動は必ず終える

(4) Mandatory activity with multiple candidates for one household member:

$$\sum_{a_n \in A_n(p)} \sum_{i,t:(i,j,t,s) \in E(p,a_n)} x_{ij,t,s}^p = 1, \quad \forall p \in P_n$$

行える義務的活動は候補の中の1つのみ

(5) Discretionary activity for each household member

$$\sum_{i,t:(i,j,t,s) \in E(p,a_q)} x_{ij,t,s}^p \leq 1, \quad \forall p \in P_q$$

選択的活動は任意

(6) Binary variable:  $x_{(ij,t,s)}^p = \{0, 1\}$ .

$p$  : 世帯メンバー

$(i, j, t, s)$  : 時間  $t \rightarrow s$ , 空間  $i \rightarrow j$  のリンク

$E$  : リンクの集合

$c_{i,j,t,s}^p$  :  $p$  のリンク移動コスト

$x_{i,j,t,s}^p$  :  $p$  のリンク選択二値変数

$O(p)$  :  $p$  の起点ノード

$D(p)$  :  $p$  の終点ノード

$DT(p)$  :  $p$  の最速出発時刻

$T$  : 総時間

$N_v$  : 車両ノードの集合

$E(p, a_m)$  :  $p$  の義務的活動  $m$  のリンクの集合

$P_m$  : 義務的活動を持つ世帯メンバーの集合

$A_n(p)$  :  $p$  の義務的活動の候補の集合

$E(p, a_q)$  :  $p$  の選択的活動  $q$  のリンク

$P_q$  : 選択的活動を持つ世帯メンバーの集合



# 3.1. 定式化 | ケースA : 時空間ネットワークベースの最適化モデル

## Recker(1995)のCase IVとの比較

“This compact formulation enables the use of standard optimization solvers for a real-world transportation network.”

**Table 4**  
Comparison between Case IV (Recker, 1995) and our Case A.

Modelling constraints	Model R4: Case IV (Recker, 1995)	Model 1 for our Case A	Remarks
(1) Time representation	Continuous	Discretized	
(2) Network representation	Abstract physical traffic network	Time-discretized space-time physical traffic network	
(3) Objective function	Eqs. (1a)-(1f) with multiple goals	Eq. (1) with travel cost only	
(4) Coupling constraints for vehicle selection of household member	Constraints (40a)-(40b)	Embedded in the modified physical network	
(5) Vehicle spatial connectivity constraints	Constraints (2), (3), (4'), (5') and (6)	Constraints (2)-(6) in the space-time network for modelling constraints (5)-(9)	Model 1 needs to build one specific activity duration link for each activity to represent the activity process
(6) Vehicle temporal constraints	Constraints (7)-(10)		
(7) Household spatial constraints	Constraints (26)-(30)		
(8) Household temporal constraints	Constraints (31)-(33)		
(9) Illogical activity constraints	Constraints (21)-(24) and (36)-(39)		
(10) Vehicle capacity constraints	Constraints (14)-(17)	Always satisfied (solo driving pattern)	
(11) Activity time window constraints	Constraints (11)-(13) and (34)-(35)	Embedded in the space-time network	Model R4 provides a starting time window and return-home window for each activity, but in Model 1 each activity only has a starting time window and does not have the return-home window. Instead, each household member has a return-home window for his/her arrival at home
	<b>多数の制約条件</b>	<b>制約条件の少ない 線形整数問題への 帰着</b>	
(12) Travel cost/time budget constraint	Constraints (18) and (19)	Not considered but can be easily added	
(13) Variable definitional constraints	Binary and continuous variables	Binary variables only	Model R4 is a mixed integer linear programming model. Model 1 is a 0-1 integer linear programming model

# 3.2. 定式化 | ケースB : 時空状ネットワークベースの最適化モデル

## 注意

- ・ 活動の種類(義務的or選択的)により最終累積活動状態は変わり得る(下図)
- ・ フロー均衡制約を満たすため, virtual ending nodeを追加(移動時のコストは0)

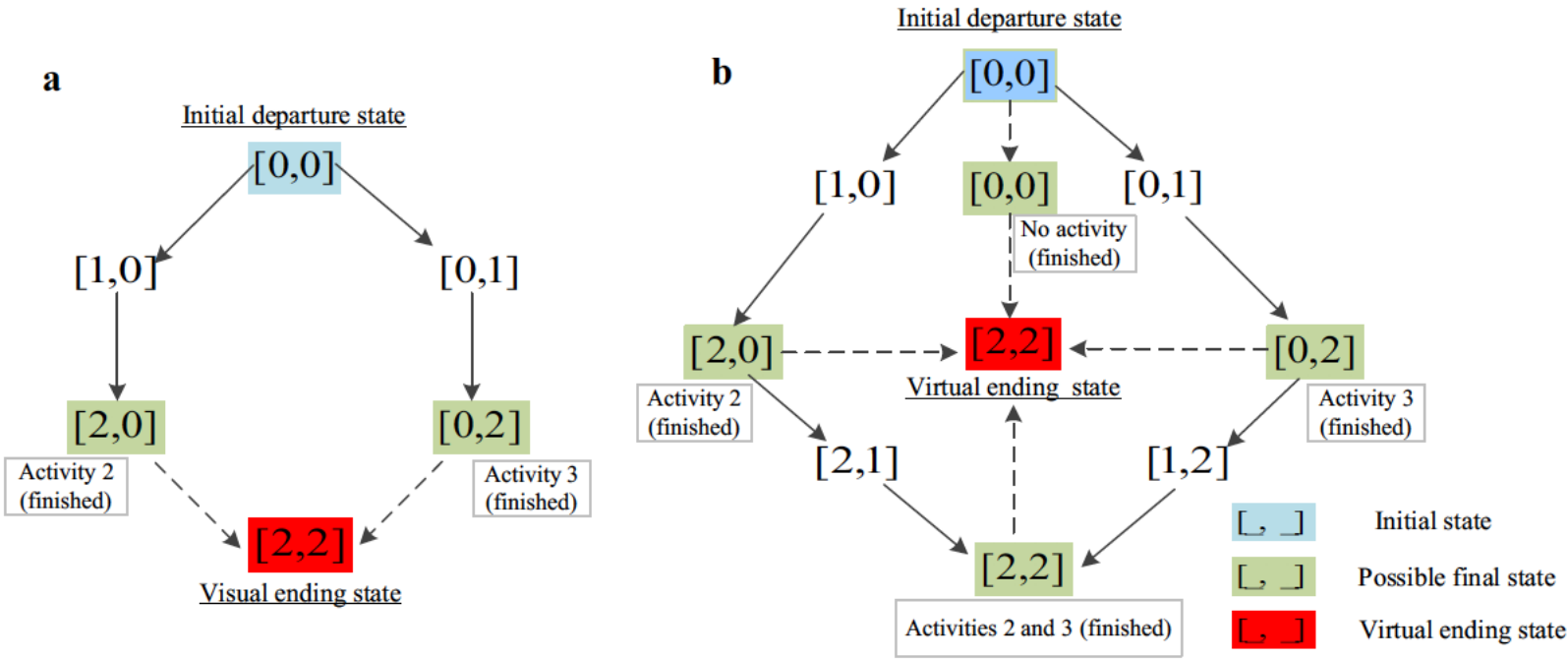


Fig. 8. (a) One of two activities should be performed; (b) two activities are optional.

# 3.2. 定式化 | ケースB : 時空状ネットワークベースの最適化モデル

## 定式化

☆三次元時空状ネットワーク : 構築済み, ライドシェアパターン : 所与

☆活動選択, 経路選択(全てリンク選択or notの二値変数)を最適化

**Model 2:**

Objective function

$$\min \sum_v \sum_{(i,j,t,s,w,w') \in E} (c_{ij,t,s,w,w'}^v \times x_{ij,t,s,w,w'}^v)$$

車両(群)の移動コスト  
 ・各世帯メンバーの活動効用を含む世帯の総コストを最小化

Subject to,

(1) Flow balance constraint for each vehicle:

$$\sum_{i,t,w:(i,j,t,s,w,w') \in E} x_{ij,t,s,w,w'}^v - \sum_{i,t,w:(j,i,s,t,w',w) \in E} x_{ij,t,s,w,w'}^v = \begin{cases} -1 & j = O(v), s = DT(v), w = [0, 0, \dots, 0] \\ 1 & j = D(v), s = T, w = [2, \dots, 2] \\ 0 & \text{otherwise} \end{cases}, \forall v$$

(2) Mandatory activity performing constraint for the driver on the activity arcs (including ride-sharing):

$$\sum_{i,t,w:(i,j,t,s,w,w') \in E(a_m)} x_{ij,t,s,w,w'}^v = 1, \forall v, \forall a \in A(v)$$

(3) Binary variable:  $x_{ij,t,s,w,w'}^v = \{0, 1\}$ .

$v$  : 車両

$(i, j, t, s, w, w')$  : 時間  $t \rightarrow s$ , 空間  $i \rightarrow j$ , 状態  $w \rightarrow w'$  のリンク

$E$  : リンクの集合

$c_{i,j,t,s,w,w'}^v$  :  $v$  のリンク移動コスト

$x_{i,j,t,s,w,w'}^v$  :  $v$  のリンク選択二値変数

$O(v)$  :  $v$  の起点ノード

$D(v)$  :  $v$  の終点ノード

$DT(v)$  :  $v$  の最速出発時刻

$T$  : 総時間

$E(v, a_m)$  :  $v$  のドライバーの義務的活動  $m$  のリンクの集合

$A(v)$  :  $v$  のドライバーの義務的活動の集合

## 3.3. 定式化 | ケースC : リンク容量制約

### 定式化

☆再掲 : ケースC...リンク容量を考慮に入れるケース

#### ・リンク容量制約

- ・ ケースAのようなperson-based=vehicle-basedの場合

$$\sum_p x_{ij,t,s}^p \leq cap_{ij,t,s}, \quad \forall (i,j,t,s) \in E$$

- ・ ケースBのようなvehicle-basedの場合

$$\sum_v x_{ij,t,s,w,w'}^v \leq cap_{ij,t,s,w,w'}, \quad \forall (i,j,t,s,w,w') \in E$$

$p$  : 世帯メンバー

$x_{i,j,t,s}^p$  :  $p$ のリンク選択二値変数

$(i,j,t,s)$  : 時間 $t \rightarrow s$ , 空間 $i \rightarrow j$ のリンク

$Cap_{i,j,t,s}$  : リンク容量

$E$  : リンクの集合

$v$  : 車両

$x_{i,j,t,s,w,w'}^v$  :  $v$ のリンク選択二値変数

$(i,j,t,s,w,w')$  : 時間 $t \rightarrow s$ , 空間 $i \rightarrow j$ , 状態 $w \rightarrow w'$ のリンク

$Cap_{i,j,t,s,w,w'}$  : リンク容量

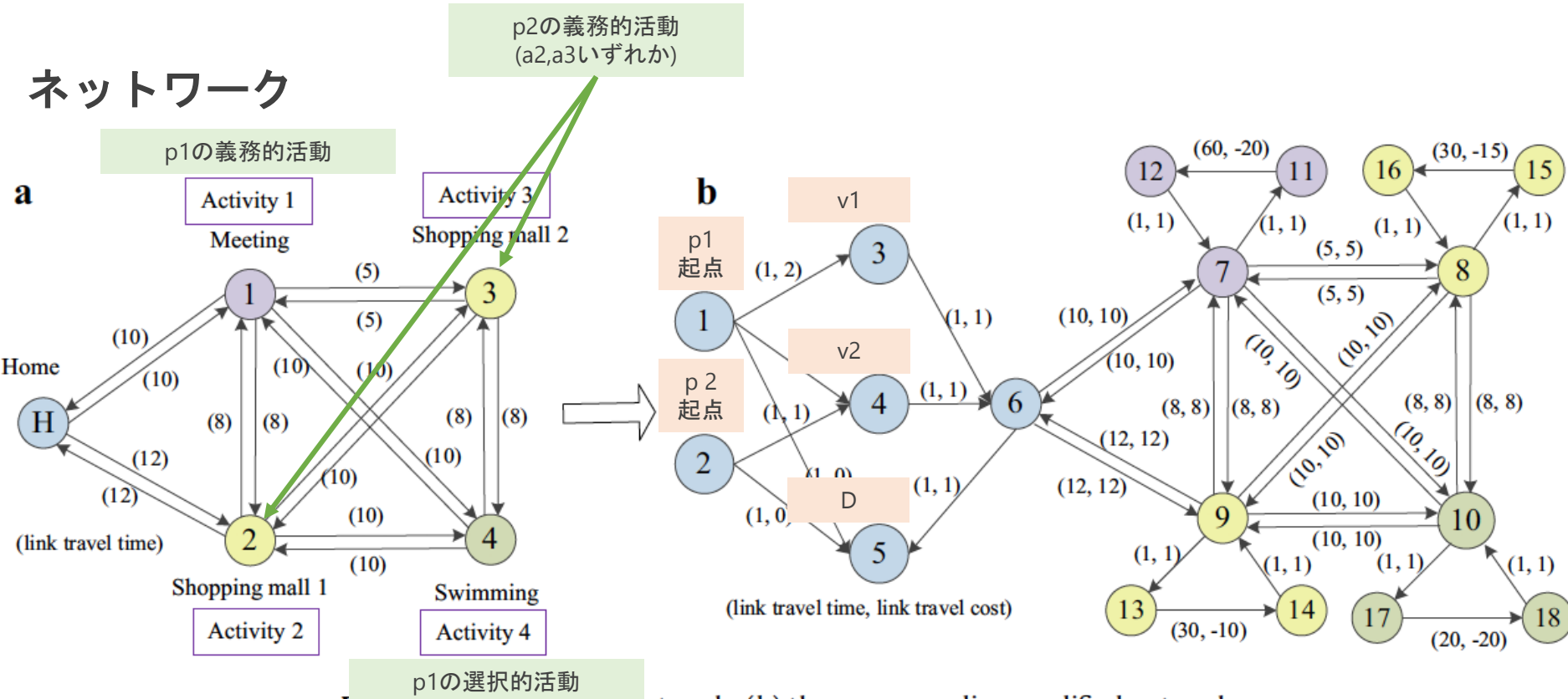
☆ドライバーの活動参加制約も時空状ネットワークに組み込めるため, ケースBは最小コスト経路探索問題となる(再掲)

→ネットワークにあらゆる制約を含めれば, あとは最小コストとなる経路探索を行えばよい

☆この最適問題を解く2つの方向性

(1)ラグランジュ緩和法 (2)キューベーストシミュレーション

# 4.1. 数値計算 | ケース A : 小規模ネットワークでの計算



## 時間窓

**Table 5**  
The specific time window for each event.

Location (node)	1	2	5	11	13	15	17
Time window	[1, 3]	[1, 3]	[1, 130]	[15, 18]	[15, 18]	[18, 20]	[86, 90]

# 4.1. 数値計算 | ケース A : 小規模ネットワークでの計算

## 計算結果

・ 使用ソフト : GAMS

時間1・時間2  
空間1・空間2  
で表されるリンクの逐次選択

**Table 6**  
The optimal solution for each household member.

Household member $p_1: x_{ij,t,s}^1 = 1$				Household member $p_2: x_{ij,t,s}^2 = 1$			
$i$	$j$	$t$	$s$	$i$	$j$	$t$	$s$
1	3	3	4	2	4	1	2
3	6	4	5	4	6	2	3
6	7	5	15	6	9	3	15
7	11	15	16	9	13	15	16
11	12	16	76	13	14	16	46
12	7	76	77	14	9	46	47
7	6	77	87	9	6	47	59
6	5	87	88	6	5	59	60



# 4.1. 数値計算 | ケース A : 小規模ネットワークでの計算

## まとめ

**時間変動する移動コスト**  
 (家-活動/活動-活動間リンクコスト)  
 ※今回の計算では時間変動しない移動コストを仮定  
**実ネットワーク**  
 可能な時間窓(時間制約に該当)  
 個々の活動の便益(負コスト)  
 (活動内リンクコスト)

これらに基づき  
 包括的に評価・選択

**最終的な活動選択**  
**最終的な経路選択**

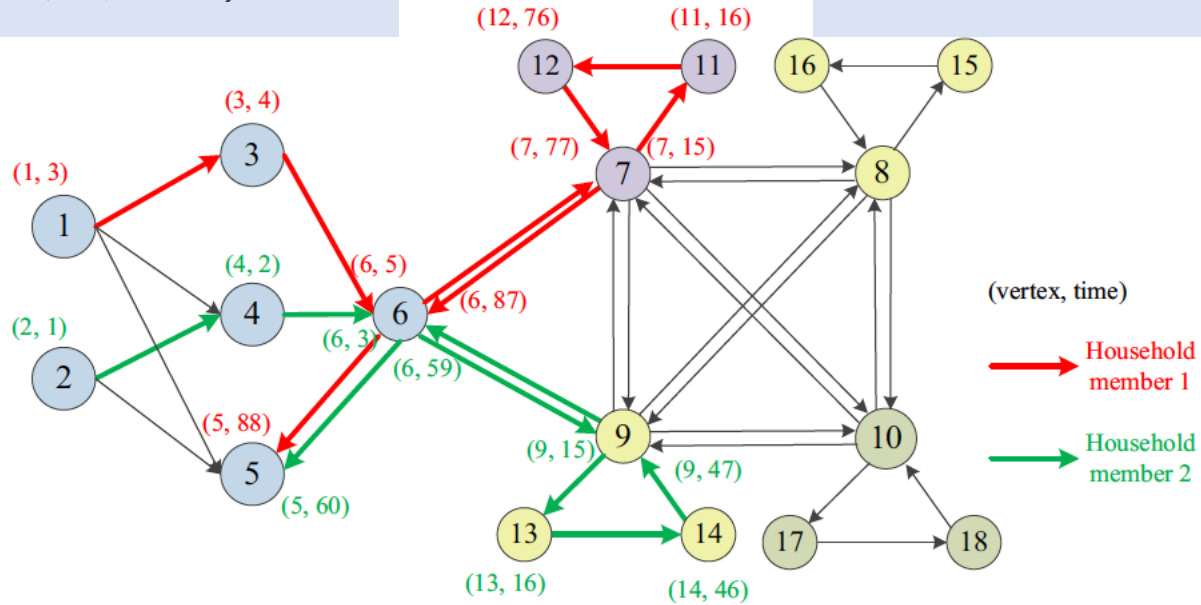


Fig. 10. The trajectories of two household members.



# 4.2. 数値計算 | ケース B : 小規模ネットワークでの計算

ネットワーク

$p_1$ (ドライバー)  
 $a_1$ は絶対する  
 $p_2$ と $p_3$ を送って拾う

$p_2$   
 $a_2$ は絶対する  
 $a_3$ はできたらしたい

$p_3$   
 $a_4$ は絶対する

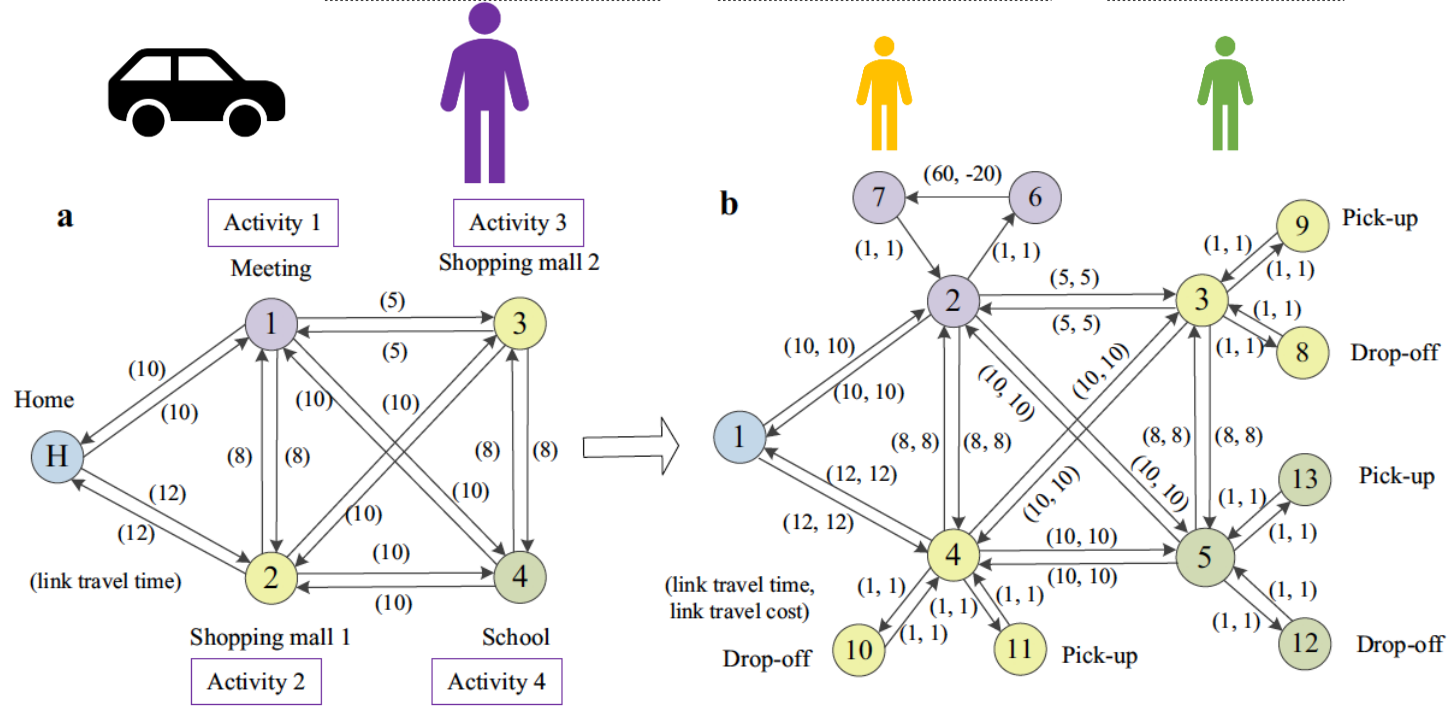


Fig. 11. (a) The physical network; (b) the corresponding modified network.

時間窓

Table 8  
 The specific time window for each event.

Location (node number)	Node 1 (departure)	Node 1 (arrival)	Node 10	Node 11	Node 12	Node 13	Node 8	Node 9	Node 6
Time window	[1, 3]	[1, 170]	[15, 16]	[114, 115]	[28, 30]	[127, 139]	[127, 129]	[137, 139]	[41, 43]

# 4.2. 数値計算 | ケース B : 小規模ネットワークでの計算

## 可能なすべての累積活動状態

- ・  $[a_2, a_3, a_4]$
- ・ 時間窓の設定から、生じ得ない活動順序を消去して行くことで、以下の8つに絞れる

**Table 7**  
Enumeration of all possible states.

State ID	State representation	State ID	State representation	State ID	State representation
1	[0, 0, 0]	9	[0, 0, 1]	17	[0, 0, 2]
2	[1, 0, 0]	10	[1, 0, 1]	18	[1, 0, 2]
3	[2, 0, 0]	11	[2, 0, 1]	19	[2, 0, 2]
4	[0, 1, 0]	12	[0, 1, 1]	20	[0, 1, 2]
5	[0, 2, 0]	13	[0, 2, 1]	21	[0, 2, 2]
6	[2, 1, 0]	14	[2, 1, 1]	22	[2, 1, 2]
7	[1, 2, 0]	15	[1, 2, 1]	23	[1, 2, 2]
8	[2, 2, 0]	16	[2, 2, 1]	24	[2, 2, 2]

## 可能な累積活動状態遷移

- ・ 先ほどと同様、起点と終点を1つにするべく virtual ending state ノードを追加

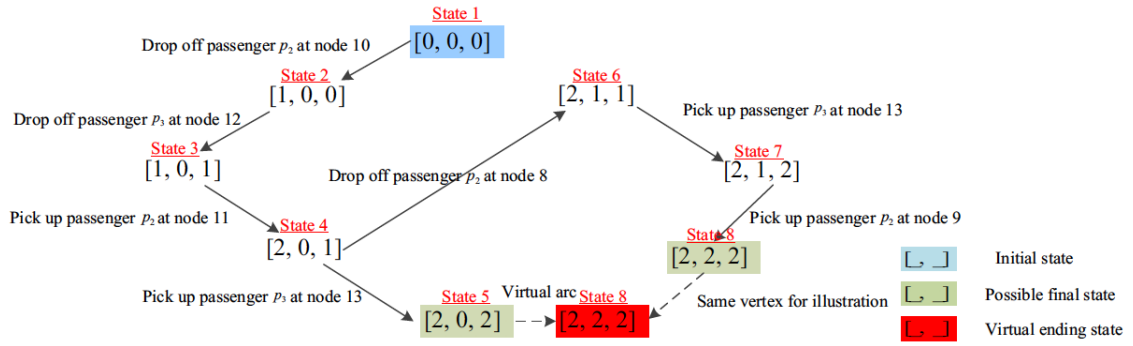


Fig. 12. The possible state transition graph.

# 4.2. 数値計算 | ケース B : 小規模ネットワークでの計算

## 計算結果

- ・  $p_2$  は  $a_3$  を行わない
- ・ 世帯の総コストは41

→  $a_3$  の活動便益(負コスト)を  $-15 - 20$  に上げると,  $p_2$  は  $a_3$  を行うようになり, 世帯の総コストは37に

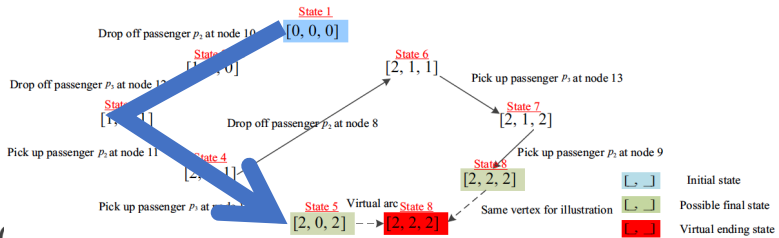


Fig. 12. The possible state transition graph.

時間1・時間2  
空間1・空間2  
状態1・状態  
で表されるリンクの逐次選択

Table 9  
The optimal solution for the household.

The only vehicle: $X_{i,j,t,s,w,w'}^1 = 1$													
$i$	$j$	$t$	$s$	$w$	$w'$	Remarks	$i$	$j$	$t$	$s$	$w$	$w'$	Remarks
1	4	3	15	1	1	Depart at home at time 3	7	2	103	104	3	3	
4	10	15	16	1	1		2	4	104	112	3	3	
10	10	16	16	1	2	State transition (passenger $p_2$ is dropped off at node 10 for activity 2)	4	4	112	113	3	3	
10	4	16	17	2	2		4	11	113	114	3	3	
4	5	17	27	2	2		11	11	114	114	3	4	State transition (passenger $p_2$ is picked up at node 11 for activity 2)
5	12	27	28	2	2		11	4	114	115	4	4	
12	12	28	28	2	3	State transition (passenger $p_3$ is dropped off at node 12 for activity 4)	4	5	115	125	4	4	
12	5	28	29	3	3		5	5	125	126	4	4	
5	2	29	39	3	3		5	13	126	127	4	4	
2	2	39	40	3	3		13	13	127	127	4	5	State transition (passenger $p_3$ is picked up at node 13 for activity 4)
2	2	40	41	3	3		13	5	127	128	5	5	
2	6	41	42	3	3		5	2	128	138	5	5	
6	6	42	43	3	3		2	1	138	148	5	5	Arrive at home at time 148
6	7	43	103	3	3	The driver $p_1$ performs activity 1	1	1	148	149	5	8	State transition (from final state to assumed final state, the virtual arc cost is 0)

## 4.3. 数値計算 | 累積活動状態を用いたラグランジュ緩和法フレームワークにおける中規模ネットワークでの計算

☆中規模ネットワークにおいて一般的なHAPPにおける累積活動状態を用いた計算効率を考察

### ネットワーク

- ・ フェニックス(アリゾナ州)
- ・ 3164リンク, 387活動場所



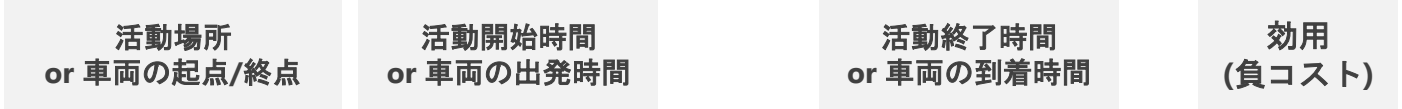
Fig. 13. One subarea of Phoenix regional transportation network.

# 4.3. 数値計算 | 累積活動状態を用いたラグランジュ緩和法フレームワークにおける中規模ネットワークでの計算

## 入力データ

☆二つの車両(ノード13)が6つの義務的活動と1つの選択的活動をどう行うか、という問題

- ・ノード13から出発しノード13へ戻る



**Table 10**  
The input data of this experiment.

agent_id	agent_type	from_node_id	to_node_id	departure_time_start	departure_time_window	arrival_time_start	arrival_time_window	base_profit	optional
1	0	23	23	30	5	40	5	150	0
2	0	24	24	10	20	70	10	133.33	0
3	0	26	26	40	10	60	5	83.33	0
4	0	25	25	20	20	80	5	150	0
5	0	39	39	70	5	90	5	133.33	0
6	0	35	35	20	5	110	5	183.33	0
7	0	38	38	35	10	120	5	133.33	1
Veh 1	1	13	13	1	1	120	1		
Veh 2	1	13	13	1	1	120	1		

## 4.3. 数値計算 | 累積活動状態を用いたラグランジュ緩和法フレームワークにおける中規模ネットワークでの計算

### 計算時間と使用メモリ

- ・ 5回のラグランジュ緩和?
- ・ もし世帯メンバーの車両/活動選好を考慮すれば, 時空状ネットワークはより削減される

**Table 11**  
CPU computation time and memory use under different number of activities.

# of activities	Maximum numbers of activity performing states	CPU time (seconds)	RAM (GB)
4	81	15.5	0.3
5	243	38.2	1.3
6	729	112.3	3.6
7	2187	337.4	11.3

## 4.4. 数値計算 | 活動と道路容量が単純化された際のシミュレーションベース フレームワークにおける大規模ネットワークでの計算

☆道路容量制約下におけるシステム最適な動的車両ルーティング問題(ケースAの特殊なケース)  
について、シミュレーションベースフレームワークにより計算を行う

### ネットワーク

- ・ ソルトレイクシティ(ユタ州)
- ・ 13,293ノード, 26,768リンク, 2,302ゾーン, 135万車両, 15時-18時
- ・ 世帯の起点, 終点, 活動場所はゾーンごととする

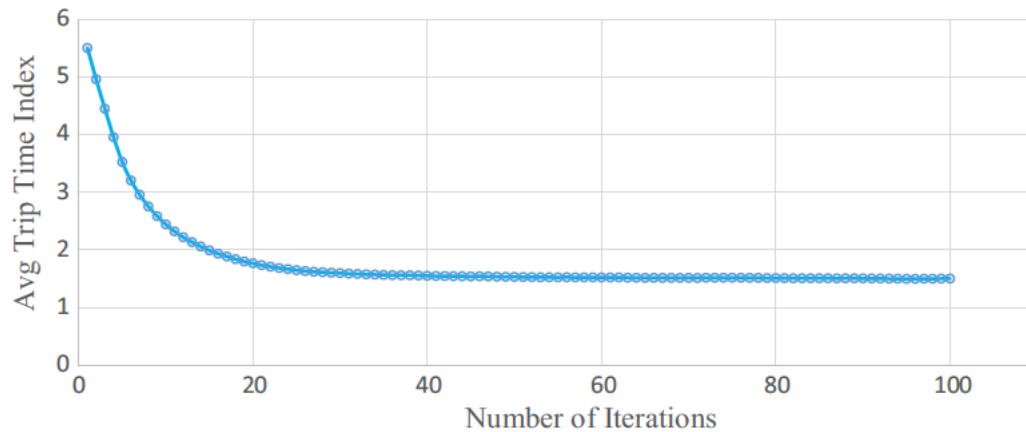


Fig. 14. Salt Lake City regional traffic network (Lu et al., 2016).

## 4.4. 数値計算 | 活動と道路容量が単純化された際のシミュレーションベース フレームワークにおける大規模ネットワークでの計算

### 反復回数と平均移動時間

- ・ 1回の反復?で"たった"1分25秒



**Fig. 15.** Average trip time index of each iteration.



# 4. 数値計算 | ケースAソースコード

```
$!title household activity scheduling problem
OPTIONS mip = CPLEX;
```

```
set i nodes /1*18/;
set t time_stamp /1*125/;
set a agent /1*2/;
set k time_window_node /1*7/;
set v vehicle /1*2/;
set m opt_act /1*1/;
```

```
alias (i, j);
alias (t, s);
```

```
-----
*connect time window nodes with the real nodes in the network based on given time window values
parameter win(t,k) /
```

```
1.1 1
2.1 1
3.1 1
1.2 2
2.2 2
3.2 2
15.3 11
16.3 11
17.3 11
18.3 11
15.4 13
16.4 13
17.4 13
18.4 13
19.5 15
19.5 15
20.5 15
86.6 17
87.6 17
88.6 17
89.6 17
90.6 17
/;
```

時間窓の幅と各ノードの対応

```
* -----physical network with arc travel time-----
```

```
parameter arcs(i,j,t,s) arc_travel_time ;
arcs(1,3,t,t+1){win(t,1)=1}=1;
arcs(1,4,t,t+1){win(t,1)=1}=1;
arcs(1,5,t,t+1){win(t,1)=1}=1;
arcs(2,4,t,t+1){win(t,2)=2}=1;
arcs(2,5,t,t+1){win(t,2)=2}=1;
arcs(3,6,t,t+1)=1;
arcs(4,6,t,t+1)=1;
arcs(6,5,t,t+1)=1;
arcs(6,7,t,t+10)=10;
arcs(7,6,t,t+10)=10;
arcs(6,9,t,t+12)=12;
arcs(9,6,t,t+12)=12;
arcs(7,9,t,t+8)=8;
arcs(9,7,t,t+8)=8;
arcs(7,8,t,t+5)=5;
arcs(8,7,t,t+5)=5;
arcs(8,9,t,t+10)=10;
arcs(9,8,t,t+10)=10;
```

各リンクとその移動時間の設定

Windows のライセンス認証  
設定を開き、Windows のライセンス認証を行ってください。

# 4. 数値計算 | ケースAソースコード

```
arcs('8','10',t,t+8)=8;
arcs('10','8',t,t+8)=8;
arcs('9','10',t,t+10)=10;
arcs('10','9',t,t+10)=10;
arcs('7','10',t,t+10)=10;
arcs('10','7',t,t+10)=10;
arcs('7','11',t,t+1)=1;
arcs('11','12',t,t+60)$(win(t,'3')= 11)=60;
arcs('12','7',t,t+1)=1;
arcs('9','13',t,t+1)=1;
arcs('13','14',t,t+30)$(win(t,'4')= 13)=30;
arcs('14','9',t,t+1)=1;
arcs('10','17',t,t+1)=1;
arcs('17','18',t,t+20)$(win(t,'6')= 17)=20;
arcs('18','10',t,t+1)=1;
arcs('8','15',t,t+1)=1;
arcs('15','16',t,t+30)$(win(t,'5')= 15)=30;
arcs('16','8',t,t+1)=1;
arcs(i,i,t,t+1)= 1;
```

\*-----arc travel cost defined by the author-----\*

```
parameter travel_cost(i,j,t,s);
travel_cost('1','3',t,t+1)$(win(t,'1')= 1)=1;
travel_cost('1','4',t,t+1)$(win(t,'1')= 1)=1;
travel_cost('1','5',t,t+1)$(win(t,'1')= 1)=0;
travel_cost('2','4',t,t+1)$(win(t,'2')= 2)=1;
travel_cost('2','5',t,t+1)$(win(t,'2')= 2)=0;
travel_cost('3','6',t,t+1)=1;
travel_cost('4','6',t,t+1)=1;
travel_cost('6','5',t,t+1)=1;
travel_cost('6','7',t,t+10)=10;
travel_cost('7','6',t,t+10)=10;
travel_cost('6','9',t,t+12)=12;
travel_cost('9','6',t,t+12)=12;
travel_cost('7','9',t,t+8)=8;
travel_cost('9','7',t,t+8)=8;
travel_cost('7','8',t,t+5)=5;
travel_cost('8','7',t,t+5)=5;
travel_cost('8','9',t,t+10)=10;
travel_cost('9','8',t,t+10)=10;
travel_cost('8','10',t,t+8)=8;
travel_cost('10','8',t,t+8)=8;
travel_cost('9','10',t,t+10)=10;
travel_cost('10','9',t,t+10)=10;
travel_cost('7','10',t,t+10)=10;
travel_cost('10','7',t,t+10)=10;
travel_cost('7','11',t,t+1)=1;
travel_cost('11','12',t,t+60)$(win(t,'3')= 11)=-20;
travel_cost('12','7',t,t+1)=1;
travel_cost('9','13',t,t+1)=1;
travel_cost('13','14',t,t+30)$(win(t,'4')= 13)=-10;
travel_cost('14','9',t,t+1)=1;
travel_cost('10','17',t,t+1)=1;
travel_cost('17','18',t,t+20)$(win(t,'6')= 17)=-20;
travel_cost('18','10',t,t+1)=1;
travel_cost('8','15',t,t+1)=1;
travel_cost('15','16',t,t+30)$(win(t,'5')= 15)=-15;
travel_cost('16','8',t,t+1)=1;
travel_cost(i,i,t,t+1)= 1;
travel_cost('1','1',t,t+1)=0;
travel_cost('2','2',t,t+1)=0;
```

各リンクとその移動コストの設定

Windows のライセンス認証  
設定を開き、Windows のライセンス認証を行ってください。

# 4. 数値計算 | ケースAソースコード

```
travel_cost('5','5',t,t+1)=0;

*-----origin, destination, and intermidate nodes of each household member-----
parameter origin(a,i,t);
origin('1','1','1')=1;
origin('2','2','1')=1;

parameter destination(a,i,t);
destination('1','5','125')=1;
destination('2','5','125')=1;

parameter intermedate(a,i,t);
intermedate(a,i,t) = (1- origin(a,i,t))* (1- destination(a,i,t));

* -----the activity type of each household member: 1 is mandatory and 2 is semi-mandatroy-----
parameter dif_act(a) /
1 1
2 2
/;

*-----nodes representing vehicles-----
parameter veh_node(v,i) /
1,3 1
2,4 1
/;

*-----the relation between optional activities and household members
parameter act_agent(m,a) /
1,1 1
/;

*-----the relation between activity links and its household member
parameter act_link(a,i,j) /
1,11,12 1
2,13,14 2
2,15,16 2
1,17,18 3
/;

variable z;

binary variables
x(a,i,j,t,s)

equations
obj
agent_on_node_origin(a,i,t) origin node agent a on node i at time t
agent_on_node_intermediate(a,i,t) intermediate node agent a on node i at time t
agent_on_node_destination(a,i,t) destination node agent a on node i at time t
mandatory_act(a) mandatory activity of agent a (assume each agent has one mandatory activity here!)
vehicle_selection(v) vehicle selection condition
semi_mand_act(a) semi-mandatory activities of agent a (one of them should be performed)
optional_act(m,a) optional activity that can be performed by multiple household members
;

obj.. z =e= sum (a,(sum((i,j,t,s)$ (arcs(i,j,t,s)>0.1), x(a,i,j,t,s)*travel_cost(i,j,t,s)))));
agent_on_node_origin(a,i,t)$ (origin(a,i,t)=1) .. sum((j,s)$ (arcs(i,j,t,s)>0.1), x(a,i,j,t,s)) =e= origin(a,i,t);
agent_on_node_destination(a,i,t)$ (destination(a,i,t)=1) .. sum((j,s)$ (arcs(j,i,s,t)>0.1), x(a,i,j,t,s)) =e= destination(a,i,t);
agent_on_node_intermediate(a,i,t)$ (intermediate(a,i,t)=1) .. sum((j,s)$ (arcs(i,j,t,s)>0.1), x(a,i,j,t,s))-sum((j,s)$ (arcs(j,i,s,t)>0.1), x(a,j,i,s,t)) =e= 0;
mandatory_act(a)$ (dif_act(a)=1) .. sum((j,i,t,s)$ (arcs(i,i,t,s)>0.1 and act_link(a,i,i)=1), x(a,i,i,t,s)) =e= 1;
optional_act(m,a) .. sum((i,j,t,s)$ (arcs(i,j,t,s)>0.1 and act_link(a,i,j)=3 and act_agent(m,a)=1), x(a,i,j,t,s)) =e= 1;

Model Household_scheduling_case_4 /all/ ;
solve Household_scheduling_case_4 using MIP minimizing z;
display x.l;
display z.l;

File output_x;
put output_x;
loop((a,i,j,t,s)$ ((x.l(a,i,j,t,s) = 1)),put @5, a,t,l, @10, i,t,l, @20, j,t,l, @30, t,t,l, @40, s,t,l/);
```

各メンバーと起点・終点ノードの対応

各メンバーと(半)義務的活動の対応

各車両の対応ノード

各メンバーと選択的活動の対応

各メンバーと活動リンクの対応

目的関数の設定など

Windows のライセンス認証  
設定を開き、Windows のライセンス認証を行ってください。

Windows のライセンス認証  
設定を開き、Windows のライセンス認証を行ってください。

## 5. まとめ

---

- ・ 複数の制約をwell-structuredな時空(状)ネットワークに組込むことで、HAPPの2つの難しいケース(個人&シェアリング)を、非常に少ない制約式で定式化した
- ・ これらは通常最適化手法、或いは時間依存/状態依存の最短経路探索アルゴリズムを用いて解ける
- ・ 数値計算により、活動の効用値が車両経路選択や世帯メンバーの活動選択に及ぼす影響を分析した
- ・ 世帯各メンバーについて最適な移動パターンが得られるよう、車両選択(ケースA)とライドシェアリング問題(ケースB)を同時に考える必要がある
- ・ こうして得られる世帯の最適な選択が全世帯に適用されれば、動的な世帯レベルでの均衡問題が解けることになる
- ・ モデル改善に向け、大規模なネットワークでのさらなる数値計算を行っている