

Liao, F., Arentze, T., Timmermans, H. (2013).

Incorporating space-time constraints and activity-travel time profiles in a multi-state supernetwork approach to individual activity-travel scheduling,

Transportation Research Part B, Vol.55, pp.41-58.

- 人の1日の行動を予測し分析するActivity (-Scheduling) model はさまざまな手法で研究が行われてきた.
 - しかし1日の活動は, その多様さ (時間的・空間的) から, 選択肢集合の列挙, 詳細な行動表現という点で課題が残されている.
1. ネットワーク (最適化) アプローチでの表現
 2. 時空間制約を考慮した活動列生成法の提案
 3. (これまでに考慮されてこなかった時間依存の) 現実的行動のモデル化

1. Multi-state supernetwork



2. space-time constraints & 3. time-dependent profiles

Activity-based modeling

+1

1) 確率効用最大化：アクティビティパターン選択モデル

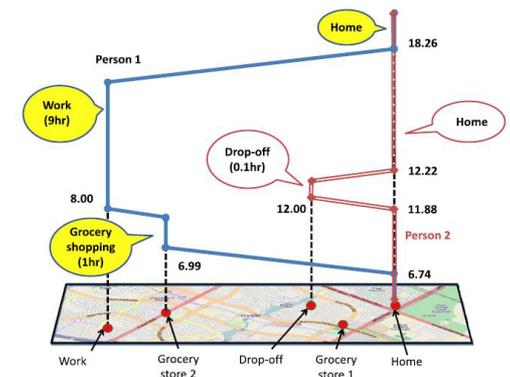
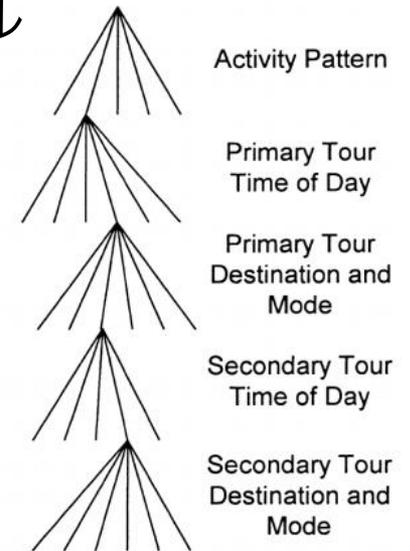
- Recker (1995)
- Fujii et al. (1998)
- Bowman and Ben-Akiva (2001)

2) 時間配分モデル：活動の割り振り

- ALBATROSS – Arentze and Timmermans (2004)
- TASHA – Miller and Roorda (2003)
- ADAPTS – Auld and Mohammadian (2009)

3) 活動列生成・最適化計算

- HAPP - Kang and Recker (2013)
- SimAGENT - Bhat et al. (2012)



Node

実際の場合 (P) を表し, 手段 (M) × 活動 (S) の状態分のノードが用意される.

場所(Place) × 手段(Mode) × 活動(activity State)

3 Links

1. Travel links (移動)

同じアクティビティの実行状態における異なるノード (場所) への遷移.

2. Transition links (手段変更)

同じアクティビティの実行状態, 同じ場所における手段の変更.

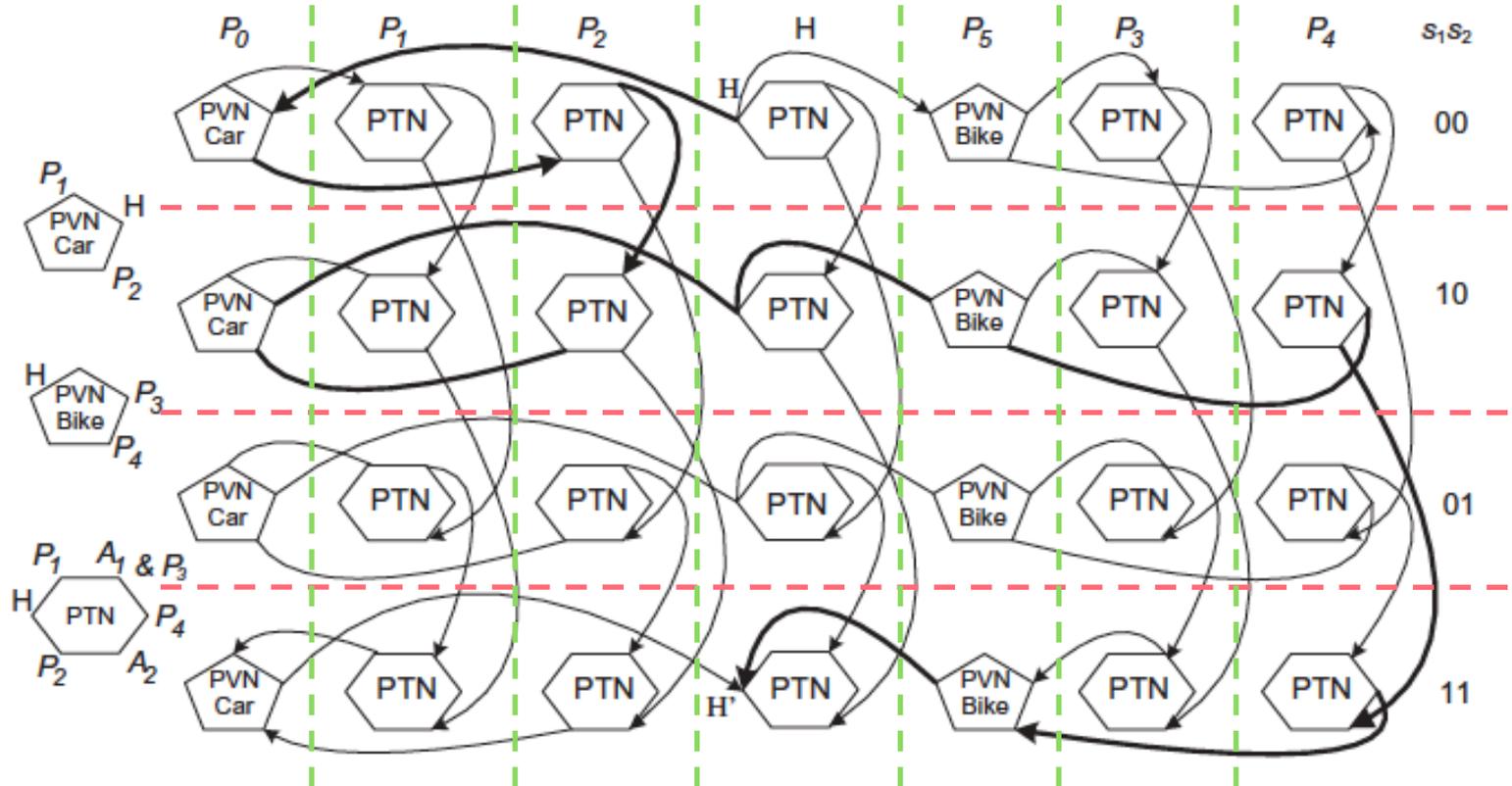
3. Transaction links (活動の実行)

同じ場所における異なるアクティビティの実行状態への遷移.

Multi-state supernetwork

3

場所 + 交通手段変更



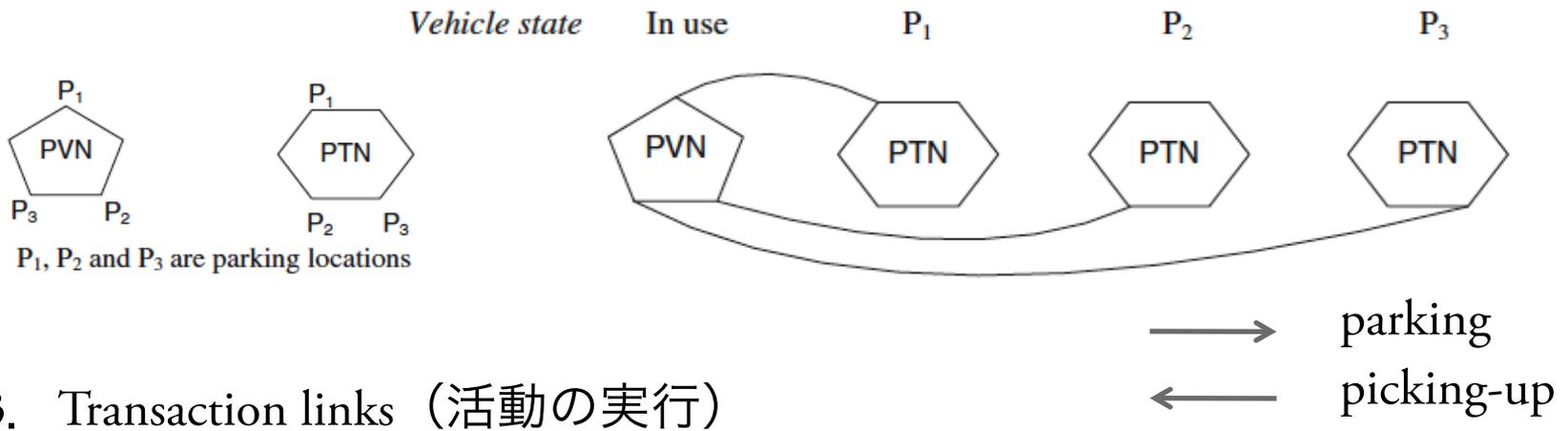
アクティビティの実行状態

- 手段： Private Vehicle Networks (PVNs) + Public Transport Networks (PTNs)
- 活動： 実行状態の組み合わせ分（各活動について 0 or 1）存在.

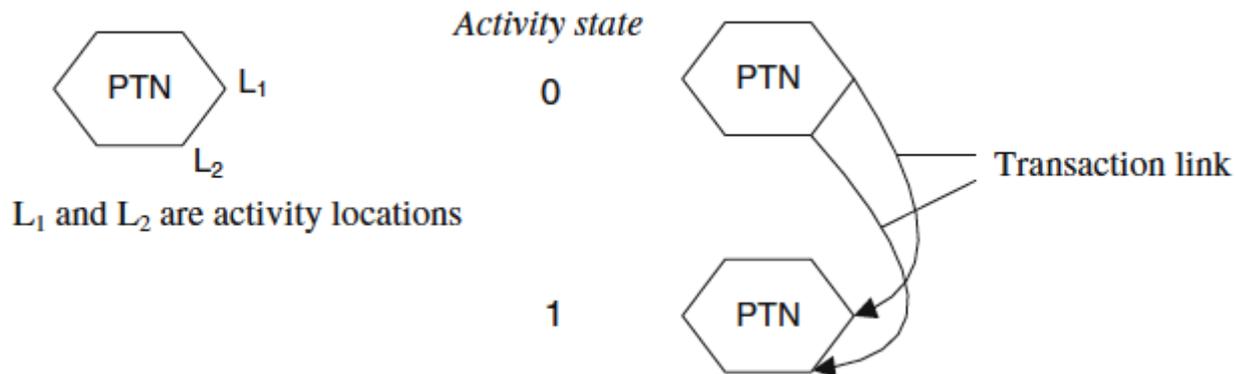
Multi-state supernetwork

1. Travel links (移動)
2. Transition links (手段変更)

※Liao et al., 2013b

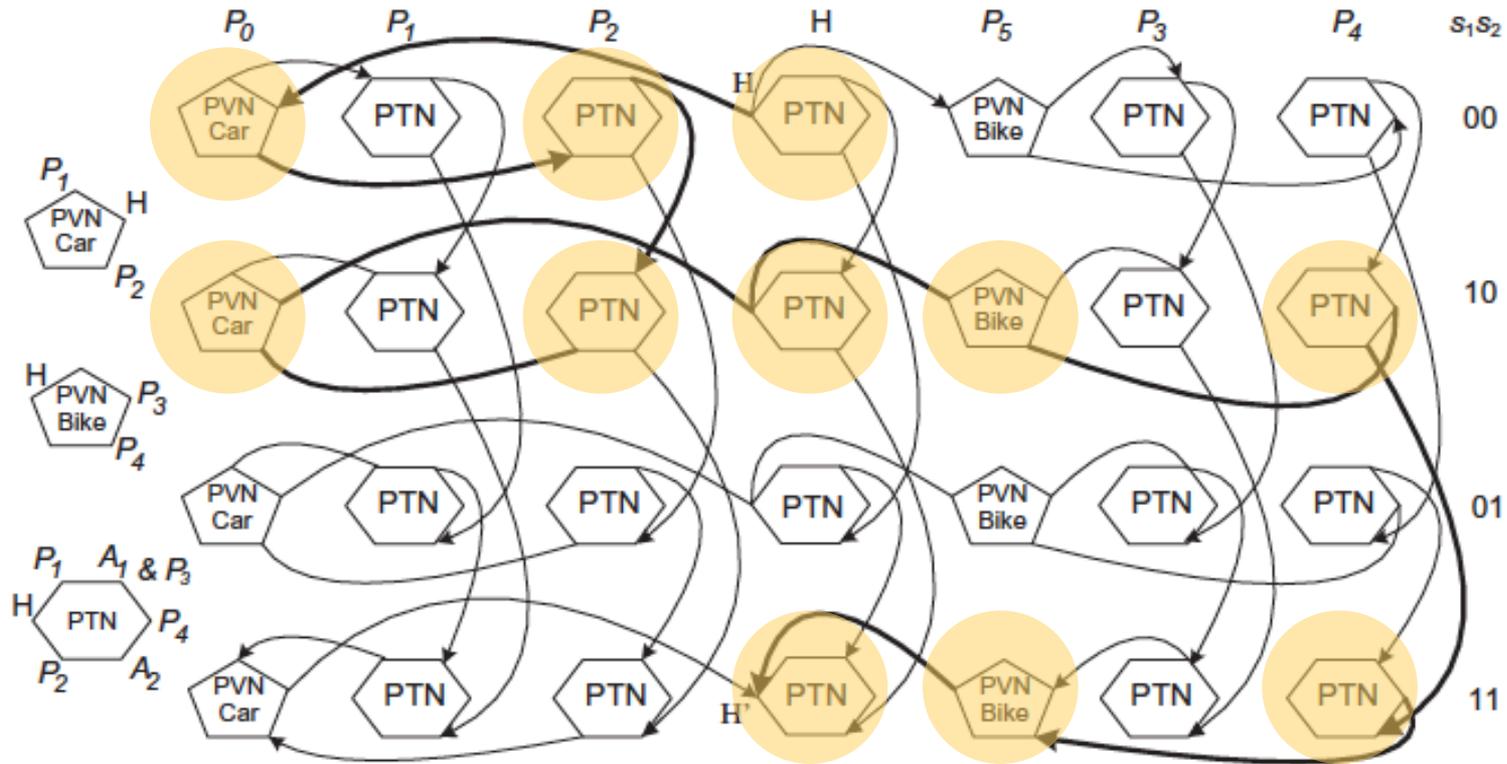


3. Transaction links (活動の実行)



Multi-state supernetwork

5



・ **太い実線**は1つの活動パターンを示している。

Home → Car (手段選択) → P2で活動 s_1 実行 → 再びCar → Home (一時帰宅)
→ Bike → P4で活動 s_2 実行 → Bike → Home (帰宅・活動終了)

各リンクコスト（個人 i , 活動状態 s , 手段 m の際のリンク l のコスト）

$$disU_{ism} = \beta_{ism} \times X_{ism} + \epsilon_{ism} \quad (1)$$

説明変数ベクトル

不効用が最小な経路（Linkの連なり = 活動パターン）を選択.

1) space-time constraints

アクティビティパタンの列挙では、選択肢数が膨大になるものの、尤もらしい選択肢集合を任意に設定する（subset: N_{δ}^A ）のは強すぎる仮定.

→ 目的地（駐車場所）選択に時空間制約を考慮して選択肢集合 N_{δ}^A を絞る.

2) time-dependency

1日のスケジュールリングを考慮するため、旅行時間の記述だけでは不十分.

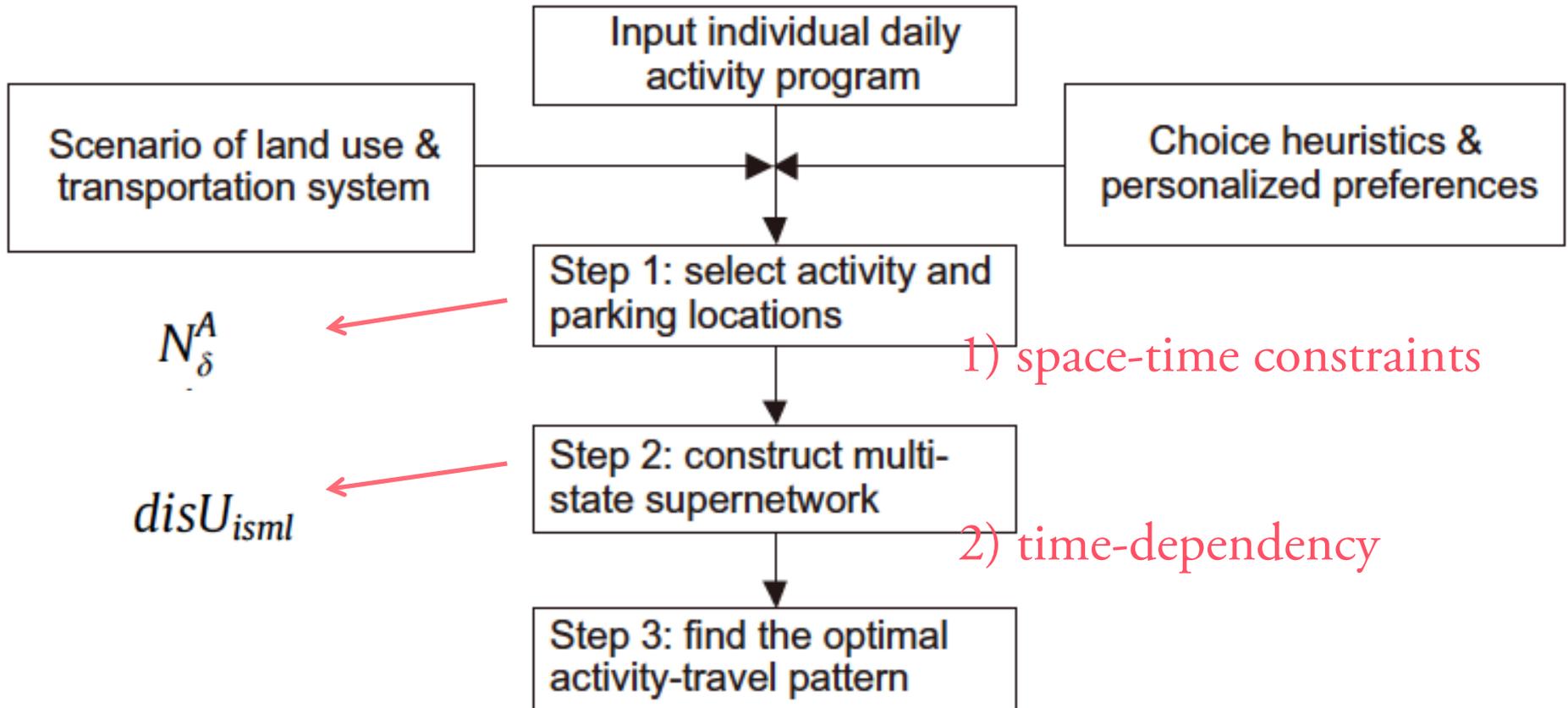
→ 活動時間・駐車時間・探索時間を同時にモデル化して時間依存性を考慮.

duration/search-time disutility profiles

3 step Approach

7

AP(Activity Program)の仮定



1. 個人は、少なくとも1つの活動 α (最大3つ) を自宅外で行なう。外にでる際の交通手段 (PV) は徒歩・自転車・自動車の3種類。
2. 活動を行なうために自宅を出る際、選択するのは1つの交通手段。
3. 個人はPTを利用する際、Parking (of PV) を行なう。また、最終的には全ての活動を終えた状態ですべての個人・PVが自宅に戻っていないなければならない。
4. 活動には、決められた場所で必ず行わなければならないもの (fixed) $c_0(\alpha)$ と、活動場所を選択できるもの (flexible) $c_j(\alpha)$ がある。
5. それぞれの活動には、個人の希望最小活動時間 $D_i(\alpha)$ がある。実際に行われる活動時間は、その理想値よりも小さくはない。また、場所ごとに time-window $[u_{c_j(\alpha)}, v_{c_j(\alpha)}]$ (営業時間のようなもの) が存在する。
6. 時空間制約のみによって決定される活動列 (Systemic) と、個人の選好が考慮された活動列 (Personal) の2種類を考える。

Step1 : Selecting locations

■活動列生成の考え方：活動順序

個人が複数の活動を行なう場合、その順序は時空間制約から定まる。

- 2つの活動： α, γ
- time-window： $[u_{L(\alpha)}, v_{L(\alpha)}], [u_{L(\gamma)}, v_{L(\gamma)}]$
(この時間内に行わなければならないということ)
- 希望最小時間： $D_i(\alpha), D_i(\gamma)$

if $v_{L(\alpha)} \leq u_{L(\gamma)}$ OR $v_{L(\alpha)} - D_i(\alpha) \leq u_{L(\gamma)} + D_i(\gamma)$, α is before γ

【例】

活動1 (弟を幼稚園へ送る) : [8:30 am, 9:30 am], 2min (minimum)

活動2 (研究室へ行く) : [8:00 am, 10:00 pm], 8hour (minimum)

どう考えても活動1→活動2の順で行われる。

Step1 : Selecting locations

10

■ 活動列生成の考え方：活動場所

fixedな活動ペア α, γ に対し, flexibleな活動 δ が加わる時.

- time-window : $[u_{c_0(\alpha)}, v_{c_0(\alpha)}], [u_{c_0(\gamma)}, v_{c_0(\gamma)}], [u_{c_j(\delta)}, v_{c_j(\delta)}] (j \geq 1)$
- 希望最小時間 : $D_i(\alpha), D_i(\gamma), D_i(\delta)$
- 旅行時間 : $tt_{c_0(\alpha)c_i(\delta)}, tt_{c_j(\delta)c_0(\gamma)}$

活動場所 $c_j(\delta)$ は次の時空間制約を充たさなくてはならない.

$$u_{c_0(\alpha)} + D_i(\alpha) + T_{i\alpha} + tt_{c_0(\alpha)c_j(\delta)} + \sum_{\delta' \in AP} (D(\delta') + T_{i\delta'}) \leq v_{c_j(\delta)} - D_i(\delta) - T_{i\delta} \quad (2)$$

$$u_{c_j(\delta)} + D_i(\delta) + T_{i\delta} + tt_{c_j(\delta)c_0(\gamma)} + \sum_{\delta' \in AP} (D(\delta') + T_{i\delta'}) \leq v_{c_0(\gamma)} - D_i(\gamma) - T_{i\gamma} \quad (3)$$

T_{i*} : 活動時間不確実性の閾値 (許容範囲)

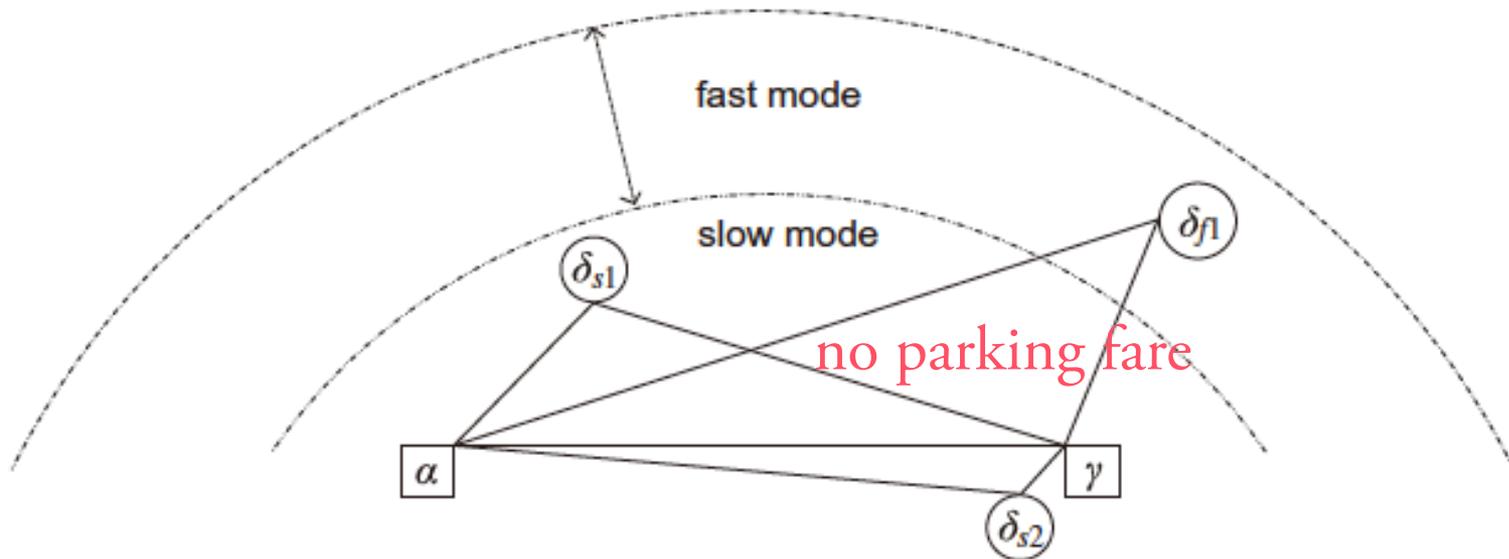
(2), (3) を充たすなかで, 不効用が最小の $c_j(\delta)$ が optimal location となる.

Step1 : Selecting locations

11

■ 活動列生成の考え方：活動場所

- ・ 交通手段を速度によって2種類考える： m_f, m_s (速度 $s_f > s_s$)



- ・ 距離 (ユークリッド距離) $E(\cdot, \cdot)$ と速度に依存.
- 実行可能解は $c_0(\alpha), c_0(\gamma)$ を焦点とする楕円で描かれる.

Step1 : Selecting locations

12

■ 活動場所の不効用

$$disU_{ic_j(\delta)} = disU_{ic_j(\delta)}^{CA} + disU_{imc_j(\delta)}^{PK} + disU_{imc_j(\delta)}^T \quad (4)$$

- 1) CA : 活動実行で得られる効用 (式1)
- 2) PK : 駐車に関する不効用 (式1)
- 3) T : 旅行時間コスト

$$disU_{imc_j(\delta)}^T = \beta_{it} \times [E(c_0(\alpha), c_j(\delta)) + E(c_j(\delta), c_0(\gamma)) - E(c_0(\alpha), c_0(\gamma))]/s_m \quad (5)$$

→ 式(4), (5)からすべての実行可能場所の不効用に順位がつく.

→ そのうち何番目まで選択肢集合とするかは, パラメータ N_δ^A を設定する.
(このパラメータの値で最適解の 正確性 ↔ 計算負荷 を調整)

$$disU_{isml} = \beta_{ism} \times X_{isml} + \epsilon_{isml} \quad (1)$$

Step1 : Selecting locations

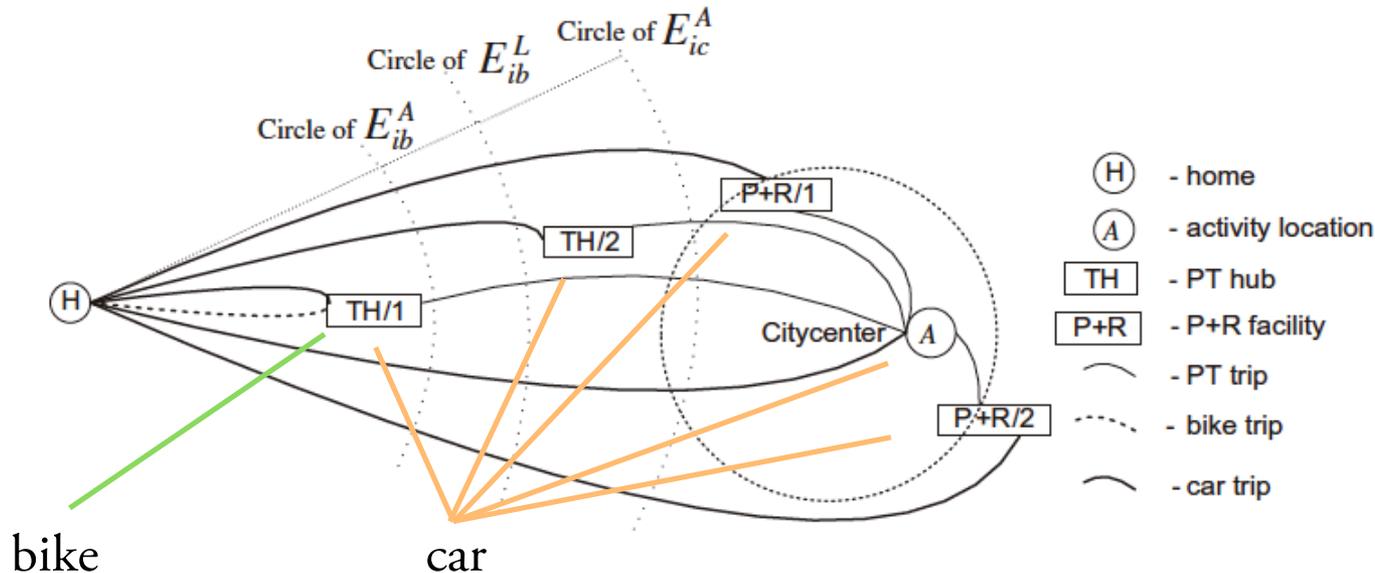
■ PV: Parking location choice

- 1) access directly
- 2) park at transport hubs (TH)
- 3) P+R facilities (P+Rs)

これより内側にある場合は必ず直接アクセス

・ 手段 p (car & bike) に Home を中心とした許容(A)・限界(L)移動距離を設定.

satisfying $E_{ip}^A < E_{ip}^L$ and $E_{ic}^L = +\infty$.



Step 1 : Selecting locations

■ Parking location choice : 実行可能解

駐車場所 P_k は次の時空間制約を充たす.

$$u_H + tt_{HP_k} + tt_{P_k C_j(\alpha)} \leq v_{C_j(\alpha)} - D_i(\alpha) - T_{i\alpha} \quad (6)$$

- (1) if P_k is a TH inside the circle E_{ip}^A , it covers the activity locations outside E_{ip}^A ;
- (2) if P_k is a TH inside the circle E_{ip}^L , it covers the activity locations outside E_{ip}^L ;
- (3) if P_k is a P + R facility, it covers the activity locations inside its hosted city center.

駐車場所 P_k の不効用は,

$$disU_{ipP_k} = disU_{ipP_k}^P + disU_{ipP_k}^T + disU_{imP_k}^T \quad (7)$$

- 1) 駐車時間に関する不効用
- 2) 駐車場までの旅行時間の不効用
- 3) 駐車場がカバーする目的地への平均コスト??

→目的地選択と同様に, パラメータ N^P を設定して選択肢集合を限定.

→このLocation selecting processでネットワークが大幅に縮小される.

Step2 : Time-dependent profile

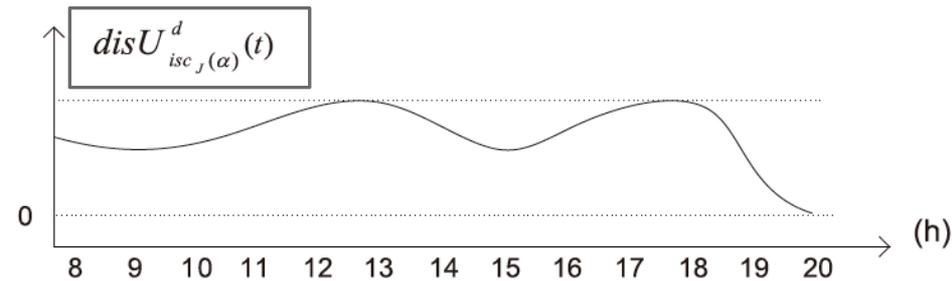
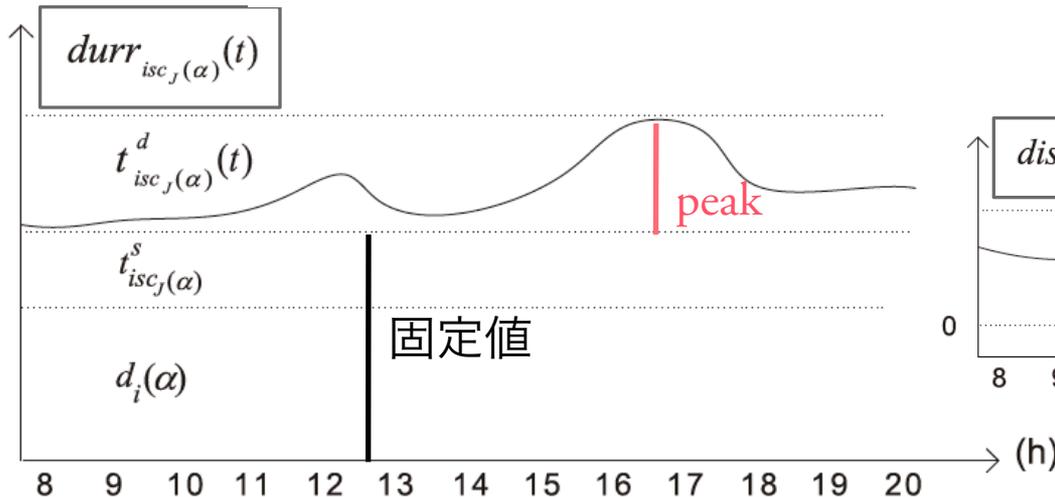
Activity participation profiles

活動では効用が得られるものの、一貫性のため不効用で表す。
 また、混雑などから希望活動時間に対する extra time の不効用を受ける。

$$durr_{isc_j(\alpha)}(t) = D_i(\alpha) + t_{isc_j(\alpha)}^s + t_{isc_j(\alpha)}^d(t) \quad (8)$$

最小活動時間 + 場所ごとに固有の extra time (自由流活動時間) 時間依存性をもつ extra time (BPR関数)

$$disU_{isc_j(\alpha)}^{CA}(t) = \beta_{is}^{CA} \times X_{isc_j(\alpha)}^{CA} + \epsilon_{isc_j(\alpha)}^{CA} + disU_{isc_j(\alpha)}^d(t) \quad (9)$$

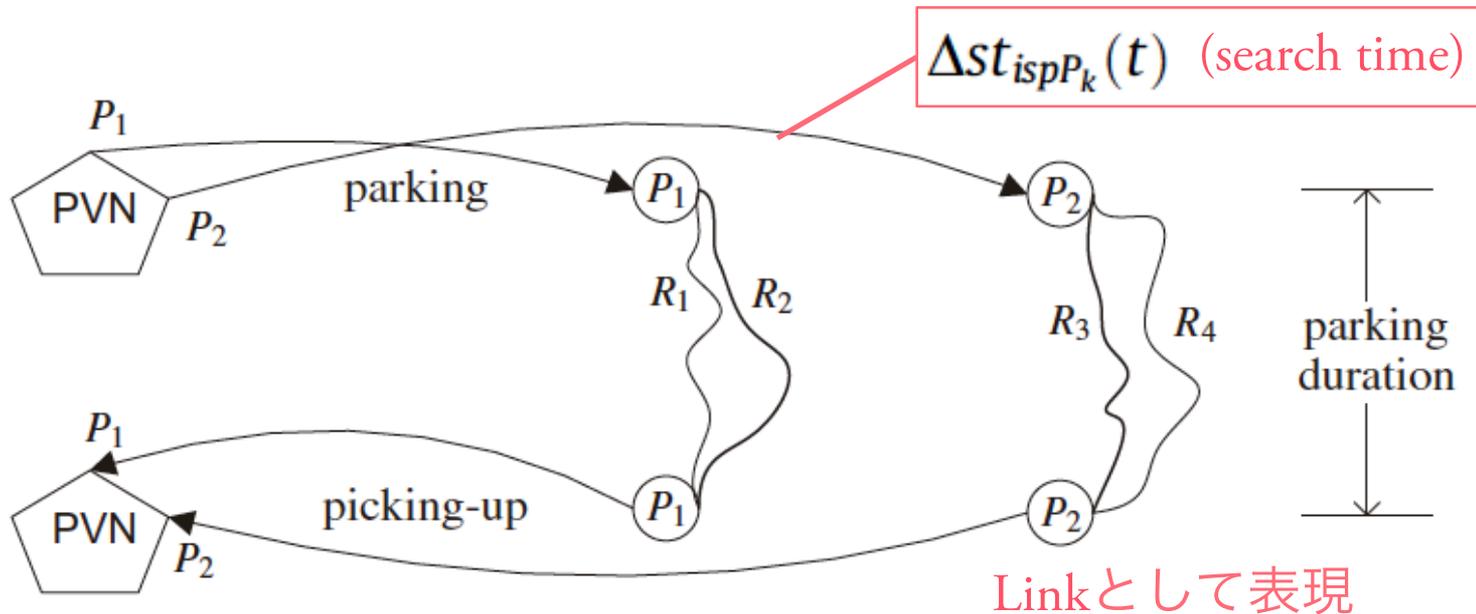


※ grocery shopping の例

Step2 : Time-dependent profile

■ Parking duration profiles

駐車の不効用についても、同様に時間依存性を考える。



$$disU_{ispP_k}^{PK}(t) = \beta_{ic} \times a_{P_k} + \beta_{is}^{PK} \times X_{ispP_k}^{PK} + \epsilon_{isc_j(\alpha)}^{CA} + disU_{ispP_k}^t(t) \quad (10)$$

Step2 : Time-dependent profile

17

■活動の実行に関するFIFO (First-In First-Out)

活動場所に到着した時間・不効用を (t_n, d_n) (time, disutility)

活動終了後の時間・不効用を $(t'_n, d'_n) = (t_n + durr_i(t_n), d_n + disU_i(t_n))$.

$t_1 \leq t_2$, $d_1 \leq d_2 - \beta_{iw} \times (t_2 - t_1)$ のとき, 以下のFIFO原則が成り立つ.

$$t_1 + durr_i(t_1) \leq t_2 + durr_i(t_2) \quad (11)$$

待ち時間分の不効用 (パラメータ β_{iw}) が追加されるので,

$$disU_i(t_1) \leq \beta_{iw} \times (t_2 - t_1) + disU_i(t_2) \quad (12)$$

$$\longrightarrow t'_1 \leq t'_2 \text{ and } d'_1 \leq d'_2 \quad (13)$$

駐車場探索にも
同様に言える.

$$durr_{isc_j(\alpha)}(t) = D_i(\alpha) + t_{isc_j(\alpha)}^s + t_{isc_j(\alpha)}^d(t) \quad (8)$$

$$disU_{isc_j(\alpha)}^{CA}(t) = \beta_{is}^{CA} \times \mathbf{X}_{isc_j(\alpha)}^{CA} + \epsilon_{isc_j(\alpha)}^{CA} + disU_{isc_j(\alpha)}^d(t) \quad (9)$$

Step3: Find optimal patterns

18

■ dominance condition

$$t_1 \leq t_2 \quad d_1 \leq d_2$$

を満たす時, (t_1, d_1) は (t_2, d_2) よりも優勢にある. $(t_1, d_1) \succeq (t_2, d_2)$
このとき, (t_2, d_2) を考慮しない.

このラベルは各ノード (n) で付けられるため, 劣勢なラベルを除いたもので B_n を作る.

最終的に, 帰宅時のラベル $B_{H'}$ の中で最適解を表現する.

$$\min\{d_{H'} + \beta_{iw} \times (t_{H'} - \min(t|t \in B_{H'}))\} \quad (14)$$

(t_i, d_i) = (arrival time, disutility)

Step3: Find optimal patterns

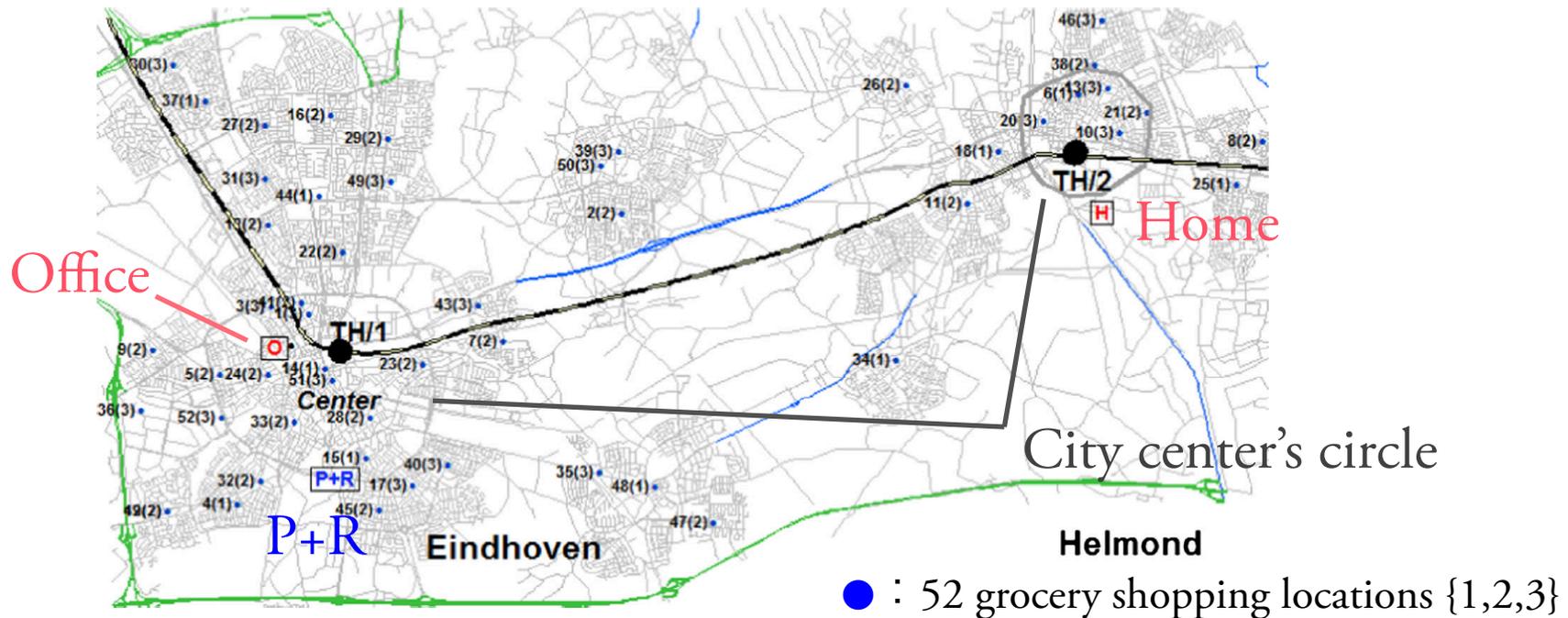
19

■ algorithm

1. input: $\langle AP, B_H, \text{personalized parameters, scenario setup} \rangle$
2. execute step 1 and step 2 in Fig. 2 to construction a supernetwork – SNK
3. initialization: $scanList = \{H\}, B_n = \emptyset$ for $n \in SNK \setminus \{H\}$
4. while $scanList \neq \emptyset$
5. choose first node n from $scanList$, and $scanList = scanList - \{n\}$
6. for each link $n \rightarrow w (w \in SNK)$
7. for each label $b_n(t, d) \in B_n$ that did not traverse $n \rightarrow w$ before
8. update $b_w(t, d)$ in terms of link type and arrival time at n
9. if $t \leq v_H$
10. merger B_w and $b_w(t, d)$ into a non-dominate set
11. end if
12. end for
13. end for
14. if B_w changes and $w \notin scanList$
15. $scanList = scanList + \{w\}$
16. end if
17. end while
18. output optimal label and backtrack the path

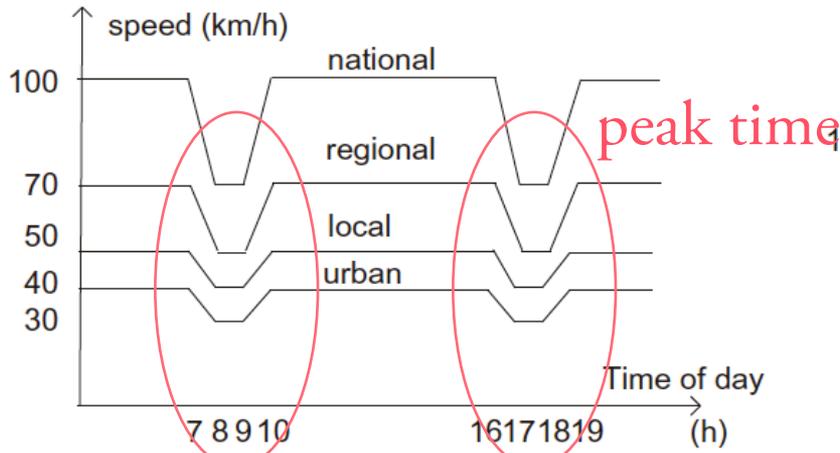
Eindhoven-Helmond corridor (28,734 nodes, 81,360 links)

- ある一日を対象に， Helmond在住の個人 i が行なう活動を考える。

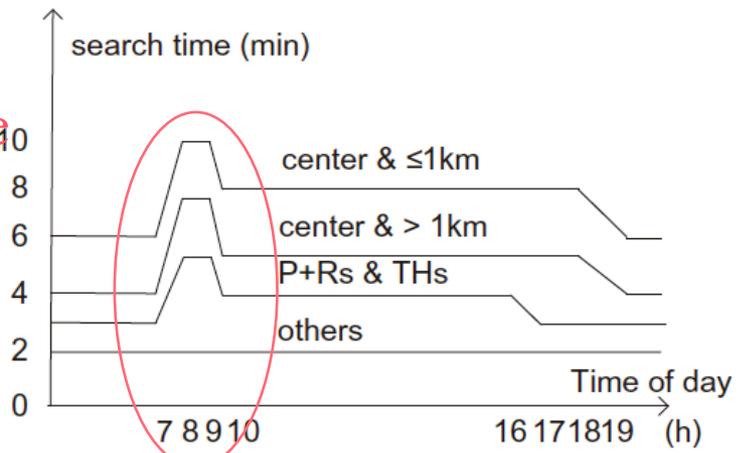


- PT : 電車 (10/12分, 30分間隔, 0.15€/km) , バス (44分, 20分間隔, 0.3€/km)
- 道路<urban, local, regional, national> : walk<5,6,0,0>, bike<14,16,17,0> (km/h), car fuel<0.18,0.16,0.12,0.1>(€/km)
- 駐車場 : <a,b>(€, €/h) = <0.8,0.18>(P+R), <1.0, 0.6>(city center), <0.5, 0.3>(1-2km)

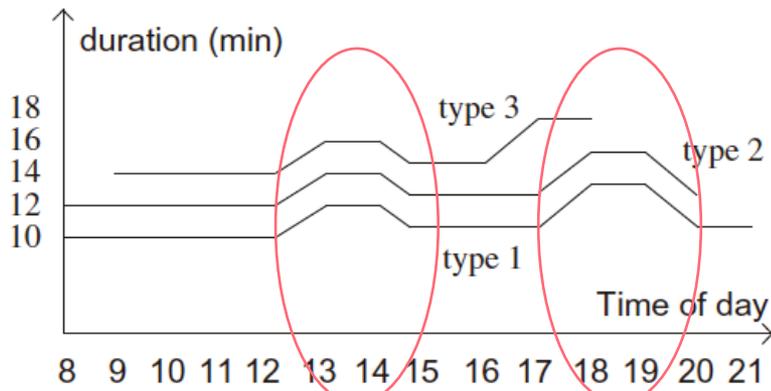
- Profiles



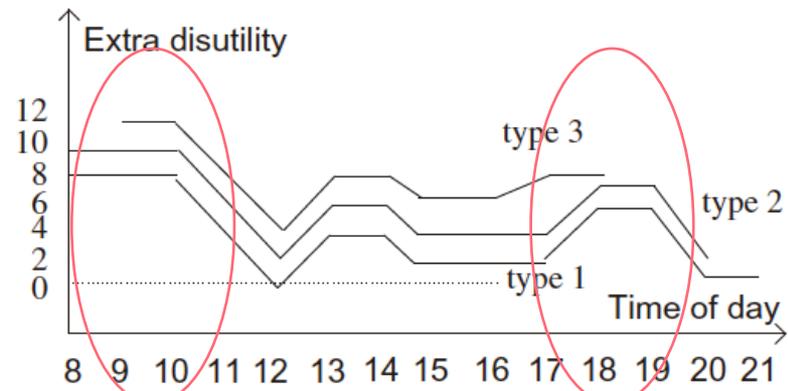
(a) car speed profiles;



(b) car parking search time profiles;



(a) duration profiles;



(b) extra disutility profiles;

Example 1

- 車を利用して仕事 (working) と日用品の買い物 (grocery shopping) を考える.
- 仕事→買い物の順

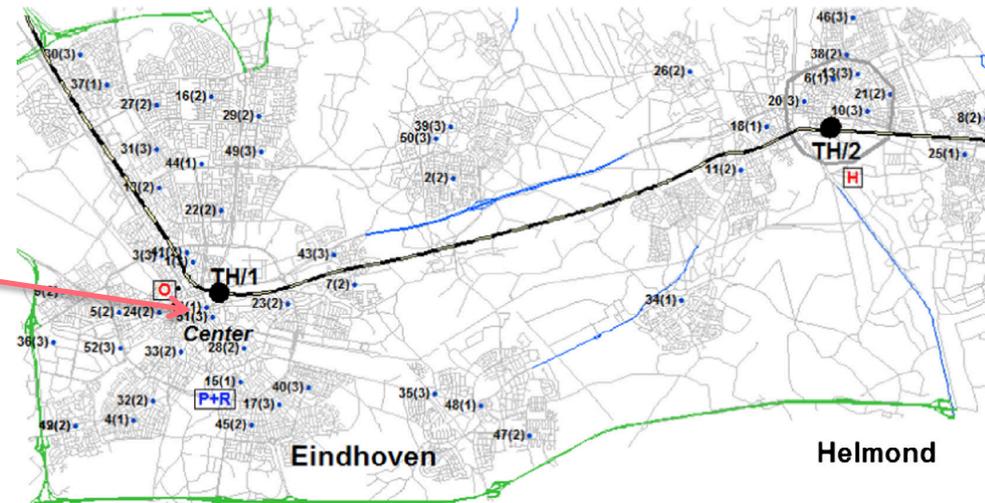
N_s^A	Supernetwork scale				Queries		Optimal label at H'	Running time (s)
	Nodes	Links			PVN	PTN		
		All	PVN	PTN				
1	72	92	41	22	97	56	(6:36 pm, 732.91)	0.29
2	105	110	51	26	119	64	(6:36 pm, 732.91)	0.37
3	144	129	62	30	157	70	(6:36 pm, 732.91)	0.48
4	189	149	74	34	184	77	(6:36 pm, 732.91)	0.62
5	240	170	87	38	226	85	(6:36 pm, 732.91)	0.78
6	297	192	101	42	259	93	(6:36 pm, 732.91)	0.90
7	360	215	116	46	296	99	(6:36 pm, 732.91)	0.99
8	429	239	132	50	348	105	(6:36 pm, 732.91)	1.12

$N=1$ でも最適値が求められた。

8つの選択枝まで絞られた

14(1)が買い物場所として選択

8:10 am に家を出て, TH/1
に車をとめて会社に歩いて
向かう行動パターン.



Example 2 : 「自転車」の選択を許容, あとはExample1と同じ設定.

N_{δ}^A	Supernetwork scale				Queries		Optimal label at H'	Running time (s)
	Nodes	Links			PVN	PTN		
		All	PVN	PTN				
1	99	119	49	29	115	68	(6:39 pm, 697.86)	0.32
2	153	147	64	35	140	78	(6:39 pm, 697.86)	0.43
3	219	177	81	41	163	86	(6:39 pm, 697.86)	0.52
4	297	209	100	47	225	95	(6:39 pm, 697.86)	0.75
5	387	247	121	56	249	107	(6:39 pm, 697.86)	0.92
6	489	287	144	65	301	117	(6:39 pm, 697.86)	1.05
7	603	329	169	74	354	129	(6:39 pm, 697.86)	1.14
8	729	373	196	83	410	137	(6:39 pm, 697.86)	1.28

Example 3 : 活動順序を (仕事→買い物), (買い物→仕事) の両方考える

N_{δ}^A	Supernetwork scale				Queries		Optimal label at H'	Running time (s)
	Nodes	Links			PVN	PTN		
		All	PVN	PTN				
1	132	222	94	58	500	222	(6:09 pm, 705.53)	1.07
2	204	277	123	70	650	258	(6:09 pm, 705.53)	1.78
4	396	401	187	100	1053	393	(6:09 pm, 705.53)	3.05
5	518	474	217	124	1366	459	(6:09 pm, 705.53)	4.17
8	972	673	339	160	2291	614	(6:09 pm, 705.53)	5.92
12	1804	977	523	220	3406	789	(6:09 pm, 705.53)	8.65
16	2596	1222	682	262	4733	935	(6:09 pm, 705.53)	11.18
19	3204	1398	798	292	7952	1294	(6:09 pm, 705.53)	16.24

1. Multi-state supernetworkを用いて個人のスケジューリング最適化問題をモデリングした.
2. 時空間制約 (space-time constraints) を入れることで選択枝集合の列挙にかかる手間 (時間) の大幅な短縮に成功した.
3. 時間依存性の活動パターンを考慮して, 詳細な現実行動のモデル化を行なった.