Markov assignment for a pedestrian activity-based network design problem

Yuki Oyama*,
Eiji Hato, Riccardo Scarinci and Michel Bierlaire

*Department of Urban Engineering
School of Engineering, The University of Tokyo
oyama@bin.t.u-tokyo.ac.jp / yuki.oyama@epfh.ch

September 12, 2017

Outline

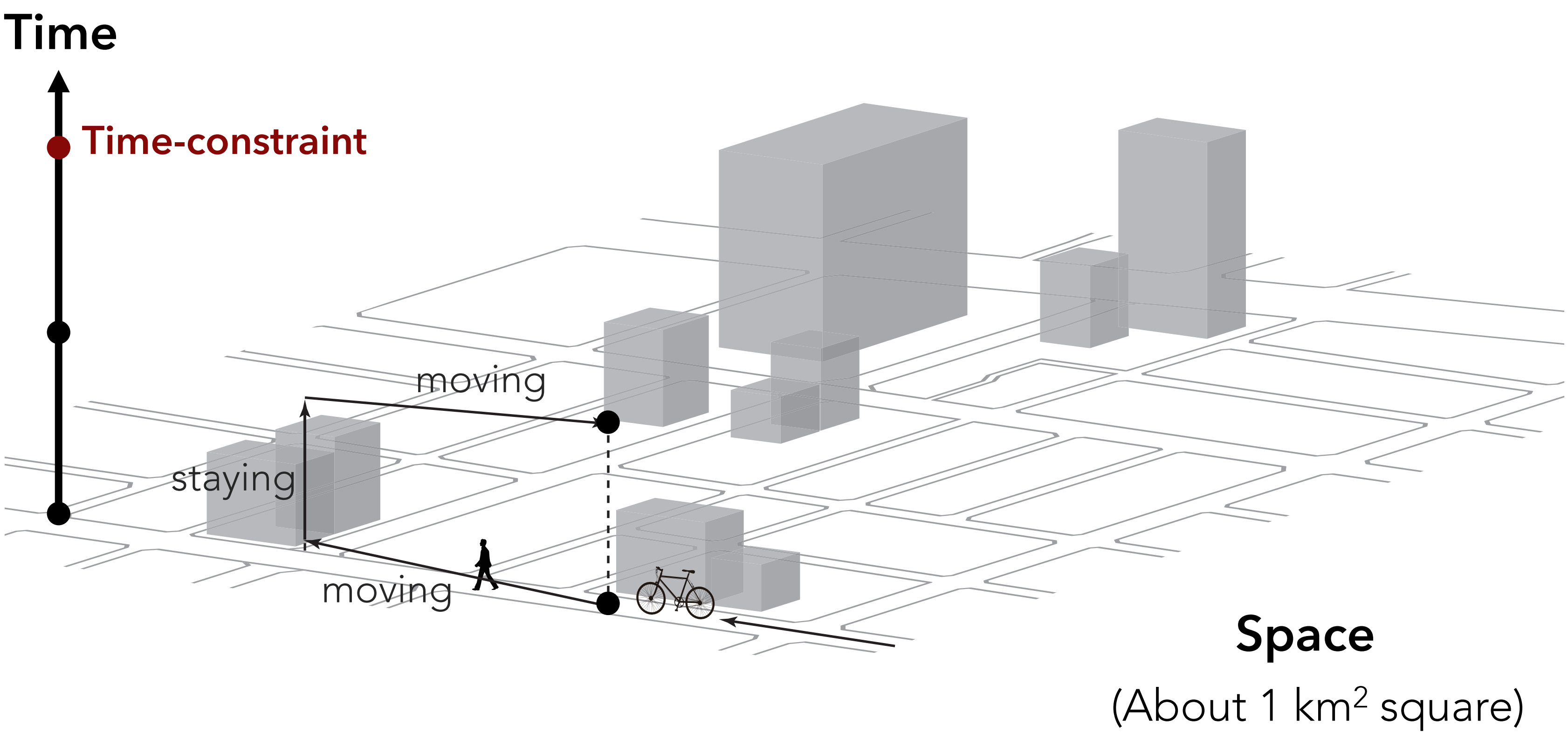
1. **Introduction**
2. **Pedestrian activity assignment**
   A) Methodology
   B) Illustrative examples
3. **Application to a network design**
4. **Conclusion**

Outline

1. **Introduction**
2. **Pedestrian activity assignment**
   - A) Methodology
   - B) Illustrative examples
3. **Application to a network design**
4. **Conclusion**

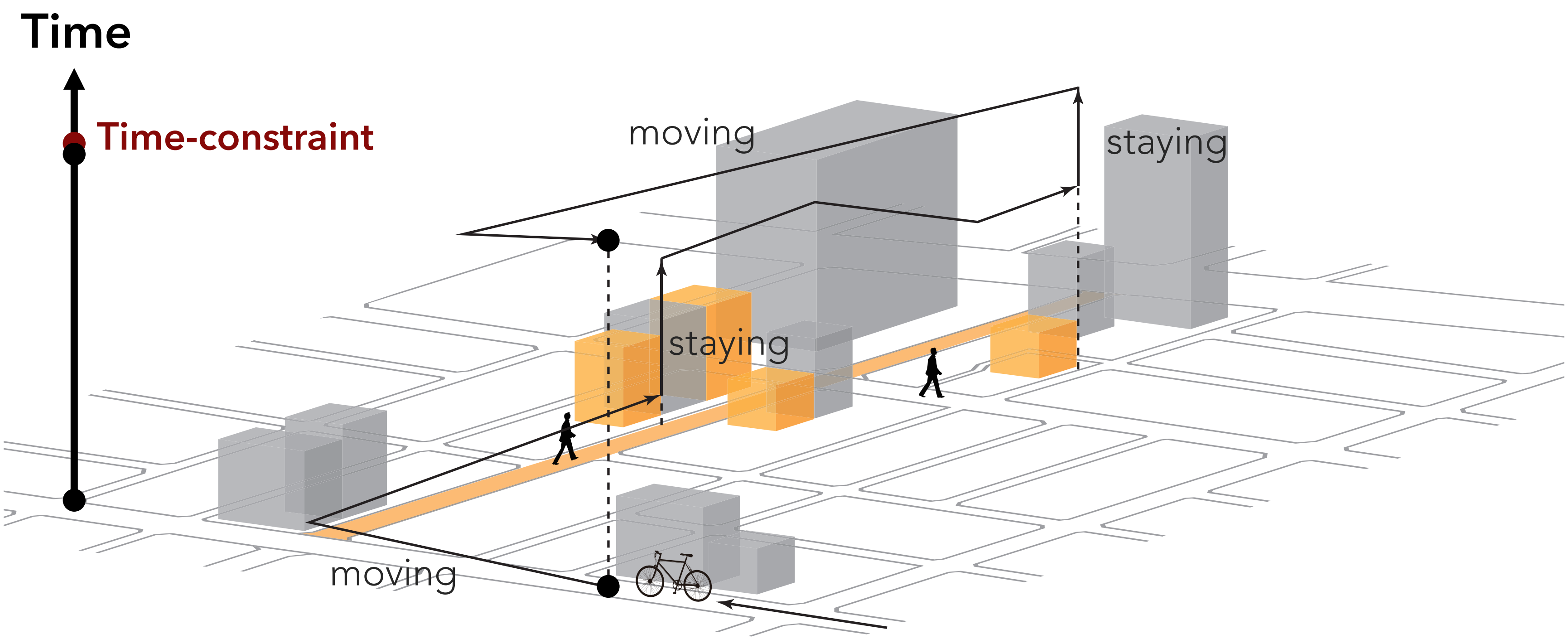
Motivation

Describing the ***sequence of travels and activities in city centers***



Motivation

Describing the ***sequence of travels and activities in city centers***



Focus of study

Activity-based network design

[Kang et al., 2013]

OD demand is not known a priori, but is the subject of responses in user itinerary choices to infrastructure improvements.

Upper level network design problem

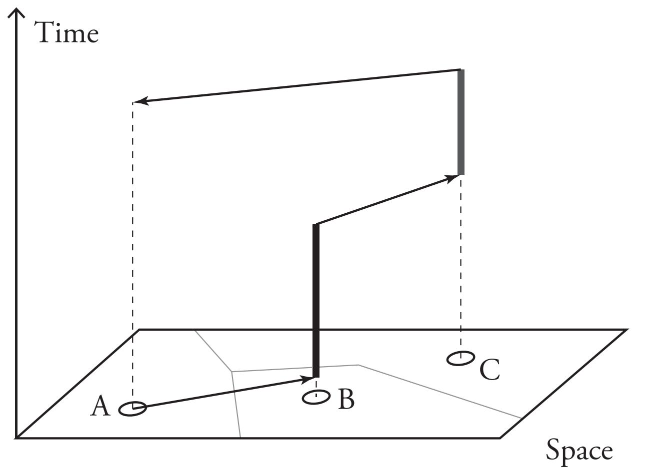
↓ ↑

Lower level activity path choice problem

Activity path choice problem

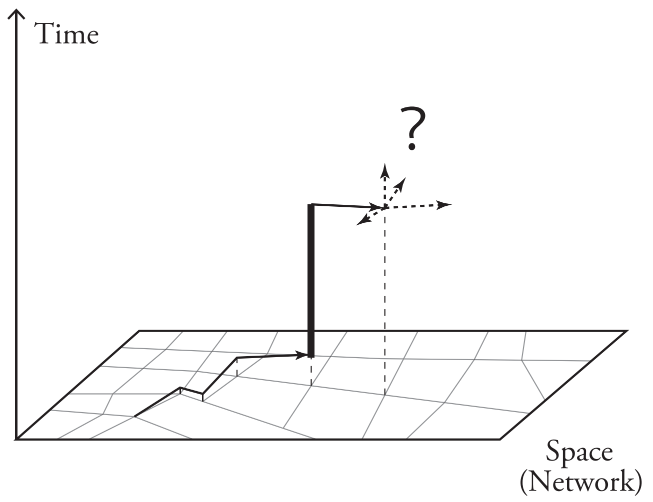
For daily household activity

- ✓ Deterministic
- ✓ Pre-trip (static)
- ✓ Low-resolution (zone-based)



For pedestrian in city centers

- ✓ Probabilistic
- ✓ Sequential (dynamic)
- ✓ High-resolution (network-based)



Contributions

1. ***Modeling pedestrian activity***
   - Combinatorial choice of route, location and duration
   - Dynamic decision making in time-space network

2. ***Algorithms for complicated computation***
   - Applying Markovian (recursive) route choice model
   - Network restriction based on the time-space prism

3. ***Application***
   - A pedestrian activity-based network design problem
   - Bi-level and multi-objective programming

Outline

1. Introduction
2. **Pedestrian activity assignment**
   A) Methodology
   B) Illustrative examples
3. Application to a network design
4. Conclusion

Framework | network description

Spatial network: includes staying node/link where activities are performed

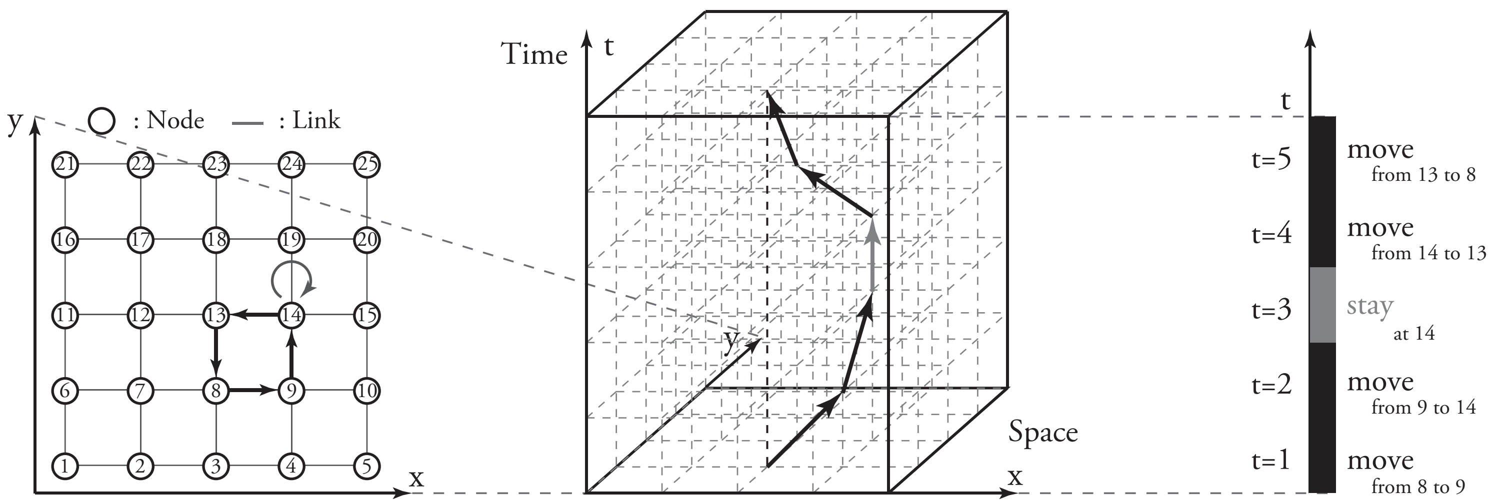
$$\mathcal{G} = (\mathcal{N}, \mathcal{A}),\ \mathcal{N} = \mathcal{N}^m \cup \mathcal{N}^s,\ \mathcal{A} = \mathcal{A}^m \cup \mathcal{A}^s$$

Discretized time: with a constant interval τ

$$t \in \{0, 1, 2, \dots, T\}$$

Activity path: the sequence of states

$$\psi = [s_0, s_1, \dots, s_T],\ \ s_t = (t, i), i \in \mathcal{N}$$



(a) (b) (c)

Framework | assumptions

1. Travelers are **homogeneous, move by only walk.** Walking speed is constant.

2. Based on **Markov decision process**, traveler's state always changes into a connected state at each discretized time *t*.

3. Traveler's decision is **restricted by time-constraint** *T*. As the result of sequential state transition from *0* to *T*, an activity path is obtained.

4. **Initial state (source, $s_0 = (0,o)$) and final state (sink, $s_T = (T,d)$) are always given.**

Network restriction

Time-space constraint

$$s_0 = (0, o), s_T = (T, d)$$

STEP1: Topological distance

$$\underline{D^o(i)}, \underline{D^d(i)}$$

STEP2: State existence condition

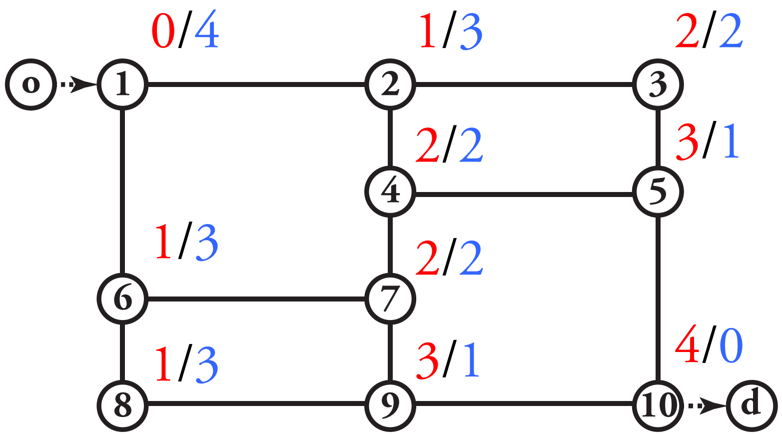
$$\mathcal{S}_t = \{i \in \mathcal{N} | I_t(i) = 1\}$$

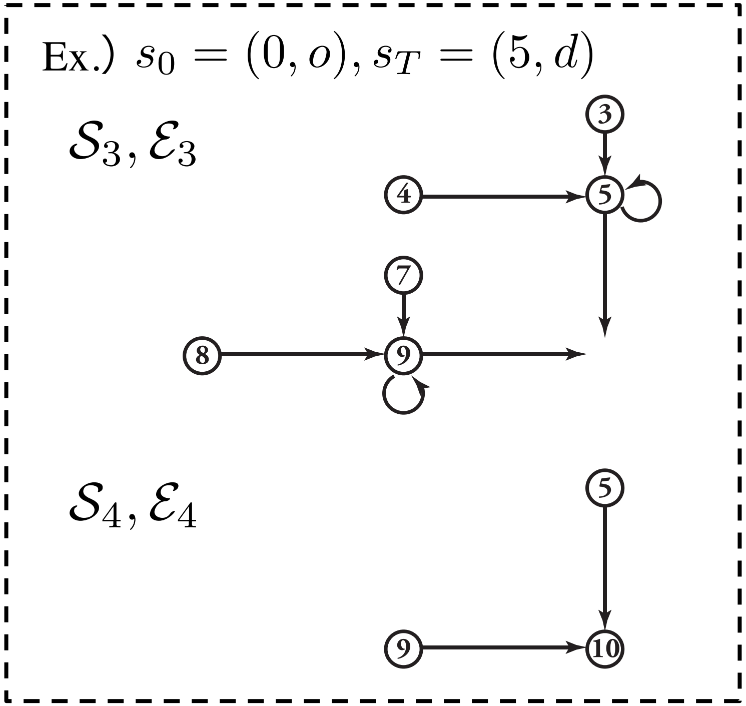
$$I_t(i) = \begin{cases} 1, & \text{if } D^o(i) \leq t, D^d(i) \leq T - t \\ 0, & \text{otherwise.} \end{cases}$$

STEP3: State connection condition

$$\mathcal{E}_t = \{(i, j) \in \mathcal{A} | \Delta_t(j|i) = 1\}$$

$$\Delta_t(j|i) = I_t(i)\delta(j|i)I_{t+1}(j)$$





Network restriction

Time-space prism

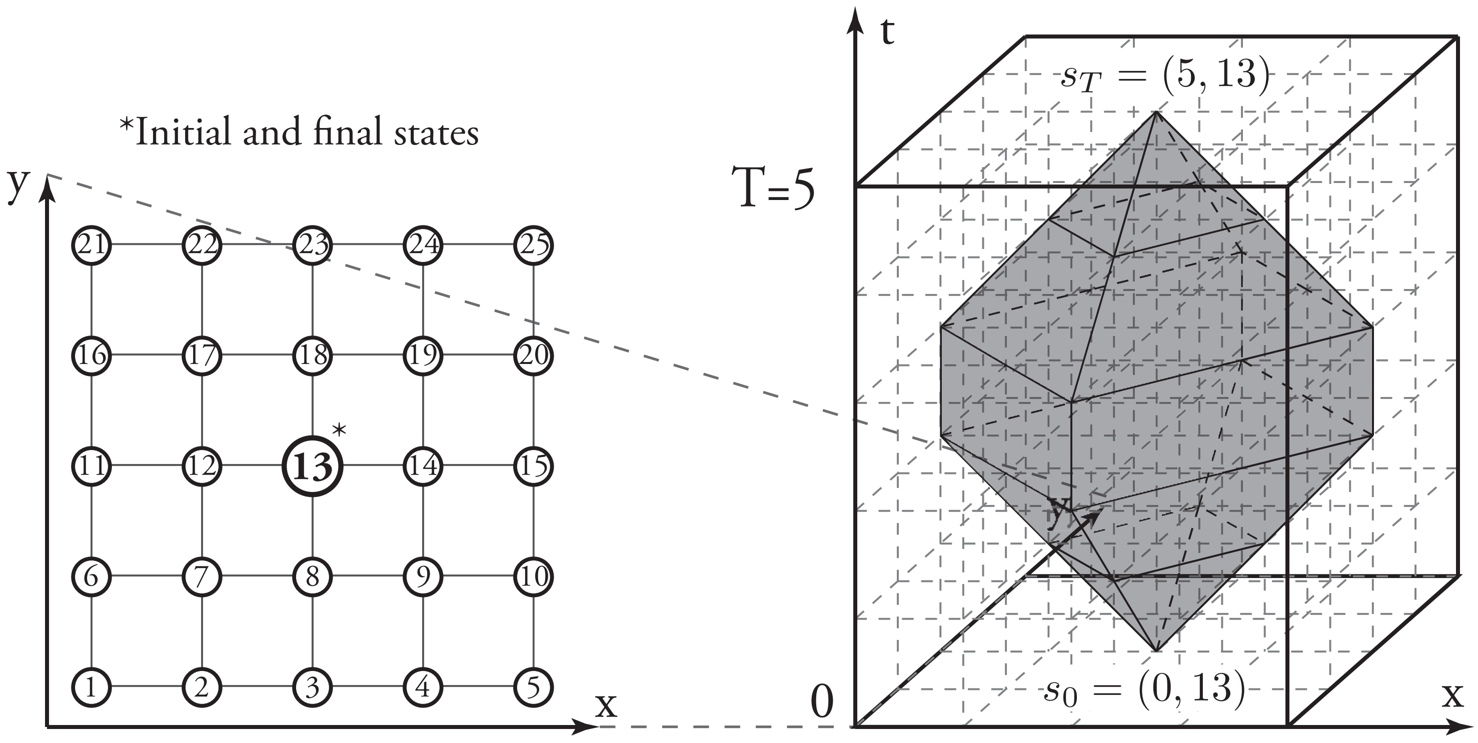


Figure: Illustrations of constrained networks by the time-space prism

Activity path choice model

Based on the ***Markov decision process***

Individual at state $s_t = i$ chooses next state $s_{t+1} = j$ that maximizes the sum of

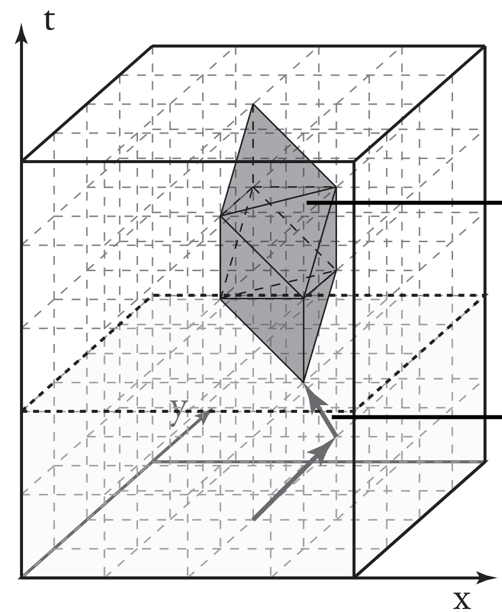
- State transition utility $\tilde{u}_{ij}^t$
- Discounted expected utility $\beta\varphi_{t+1}^{s_T}(j)$

Transition probability

$$p_t(j|i) = \frac{\Delta_t(j|i)e^{\mu\{u_{ij}^t + \beta\varphi_{t+1}(j)\}}}{\sum_{j' \in \mathcal{N}} \Delta_t(j'|i)e^{\mu\{u_{ij'}^t + \beta\varphi_{t+1}(j')\}}}$$

$\Delta_t(j|i)$: time-space prism constraint

β $(0 \leq \beta \leq 1)$: time discount factor



Maximum expected utility of the prism

$$\varphi_t(i) = \mu \ln \sum_{j \in \mathcal{N}} \Delta_t(j|i) e^{\mu\{u_{ij}^t + \beta \varphi_{t+1}(j)\}} \quad \cdots (*)$$

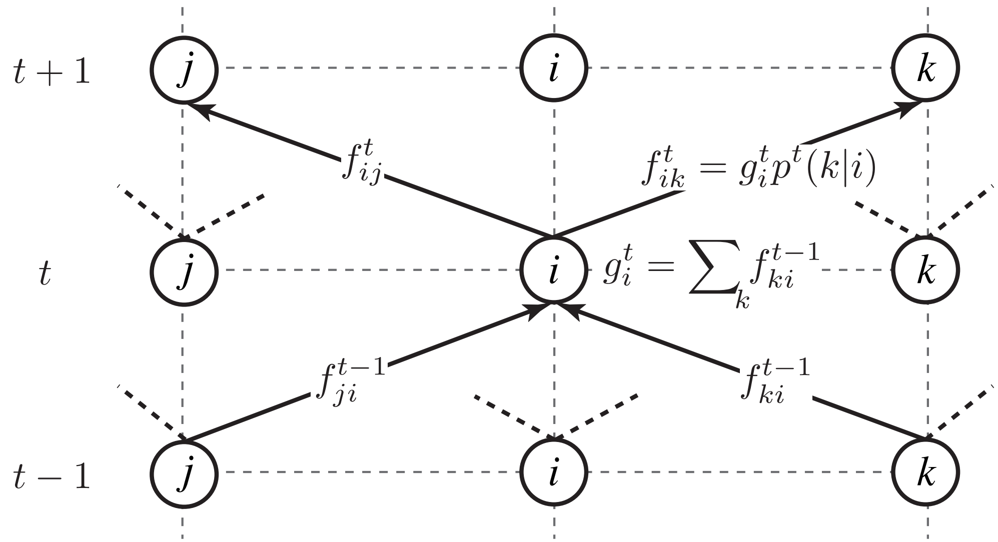
Backward induction

1. Initialize $t = T$, and $\varphi_t(i) = 1, \forall i \in \mathcal{N}$
2. Set $t = t - 1$, and calculate $\varphi_t(i)$ with Eq. (*)
3. Finish the algorithm if $t = 0$, otherwise return to Step 2.

**If the Bellman equation is non-linear, the same method can be applied.*

Network assignment

Assignment algorithm (forward)



State flow: g_i^t

Edge flow: f_{ij}^t

Input:

$$g_o^0 = Q$$

(Generating flow)

Spatial link flow:

$$x_{ij} = \sum_{t=0}^{T-1} f_{ij}^t$$

x_{ij} : flow of link (i,j)

Outline

1. Introduction
2. **Pedestrian activity assignment**
   - A) Methodology
   - B) Illustrative examples
3. Application to a network design
4. Conclusion

Model specification

Utility function:

$$\hat{u}_a^t = \underbrace{\theta_{tt} tt_a}_{\text{travel time}} + \underbrace{(\theta_w x_a^w + \theta_s x_a^s)\left(\frac{l_a}{L}\right)}_{\text{utility of moving sidewalk and shopping street}} \underbrace{+ \theta_u \int_{t\tau}^{(t+1)\tau} (x_{ij}^c + x_{ij}^d \omega) d\omega}_{\text{utility of staying}}$$

θ : a vector of coefficients

tt_a : travel time on arcs [min]

x_a^w : sidewalk width [m]

x_a^s : shopping street dummy variable

$\dot{u}_a(\omega) = x_a^c + x_a^d \omega$: deviated function of staying utility, $x_a^c > 0, x_a^d < 0$

: diminishing marginal utility

Network setting

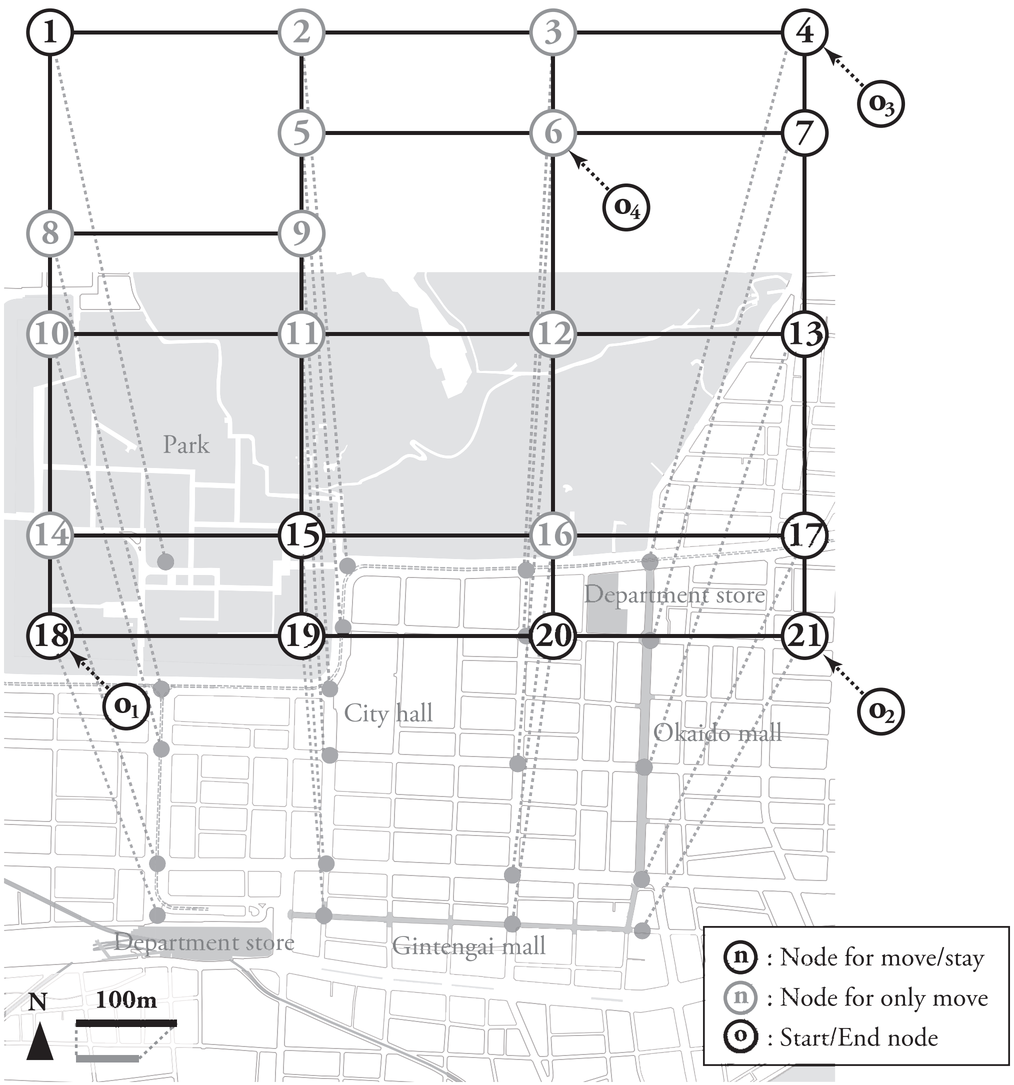


Table 4: Attributes of arcs

source	sink	L_a	n_a^{min}	n_a^{max}	x_a^s	x_a^c	x_a^d	arc
1	2	200	8	8	1	0	0	move
2	3	300	4	8	0	0	0	move
3	4	200	4	8	0	0	0	move
5	6	300	2	8	0	0	0	move
6	7	200	2	8	0	0	0	move
8	9	200	4	8	0	0	0	move
10	11	200	2	8	0	0	0	move
11	12	300	2	8	0	0	0	move
12	13	200	2	8	0	0	0	move
14	15	200	2	8	0	0	0	move
15	16	300	2	8	0	0	0	move
16	17	200	2	8	0	0	0	move
18	19	200	8	8	1	0	0	move
19	20	300	8	8	1	0	0	move
20	21	200	8	8	1	0	0	move
1	8	200	4	8	0	0	0	move
2	5	100	4	8	0	0	0	move
3	6	100	0	8	0	0	0	move
4	7	100	8	8	1	0	0	move
5	9	100	4	8	0	0	0	move
8	10	100	4	8	0	0	0	move
9	11	100	2	8	0	0	0	move
6	12	200	0	8	0	0	0	move
7	13	200	8	8	1	0	0	move
10	14	200	4	8	0	0	0	move
11	15	200	2	8	0	0	0	move
12	16	200	0	8	0	0	0	move
13	17	200	8	8	1	0	0	move
14	18	100	4	8	0	0	0	move
15	19	100	0	8	0	0	0	move
16	20	100	0	8	0	0	0	move
17	21	100	8	8	1	0	0	move
1	1	0	0	0	0	0.5	-0.005	stay
4	4	0	0	0	0	0.8	-0.015	stay
7	7	0	0	0	0	0.4	-0.003	stay
13	13	0	0	0	0	0.3	-0.001	stay
15	15	0	0	0	0	0.5	-0.006	stay
17	17	0	0	0	0	0.3	-0.001	stay
18	18	0	0	0	0	0.8	-0.015	stay
19	19	0	0	0	0	0.3	-0.001	stay
20	20	0	0	0	0	0.3	-0.001	stay
21	21	0	0	0	0	0.4	-0.002	stay

*All arcs are bidirectional and paired arc have same attributes with each other

Result | activity path choices

$$T = 90[\text{min}]$$
$$\tau = 1.5[\text{min}]$$

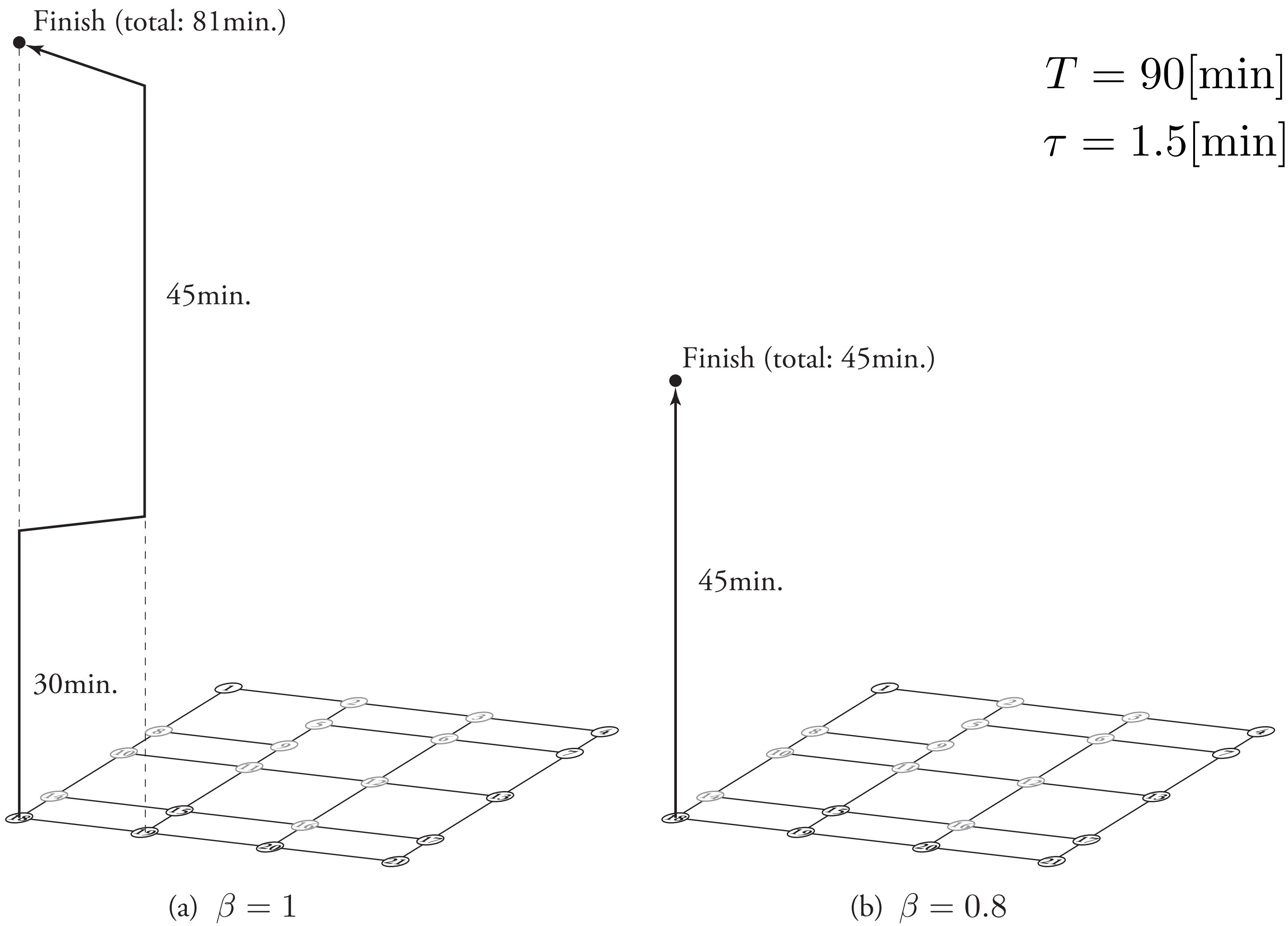


Figure: The most frequent paths departing from node 18 with deferent discount rates

Result | assignment results

Table: Input flow pattern of initial and final states

Pattern	s_0	s_T	Flow	Pattern	s_0	s_T	Flow
1	$(0, o_1)$	(T, o_1)	400	3	$(0, o_3)$	(T, o_3)	300
2	$(0, o_2)$	(T, o_2)	100	4	$(0, o_4)$	(T, o_4)	200

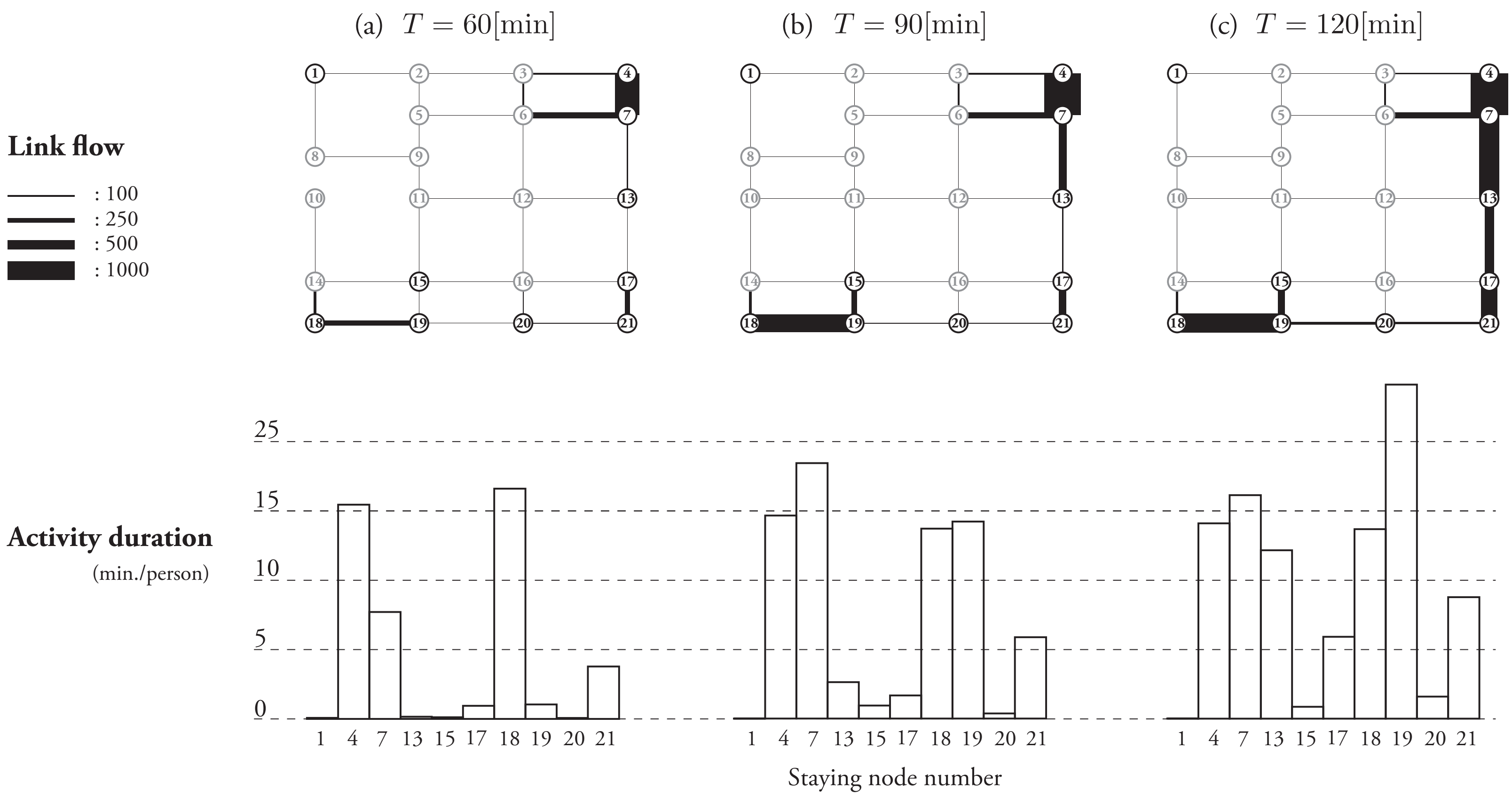


Figure: Activity assignment results with variable values of time constraints.

Outline

1. Introduction
2. Pedestrian activity assignment
   A) Methodology
   B) Illustrative examples
3. **Application to a network design**
4. Conclusion

Problem definition

Optimizing configuration of the pedestrian network

Decision variable:

$n_{ij} \in \mathbb{N}_0$: **sidewalk width on moving links** $\forall (i,j) \in \mathcal{A}^m$

s.t.,

$$n_{ij}^{\min} \leq n_{ij} \leq n_{ij}^{\max}$$

$n_{ij}^{\min}$: the minimum (current) width [m]

$n_{ij}^{\max}$: the possible maximum width [m]

Activity assignment

Decision variable: f_{ij}^{t} : **link flow at time *t*** $\forall (i,j) \in \mathcal{A}^m \cup \mathcal{A}^s, \forall t$

Multi-objective functions

1. Maximizing total duration time of district [min.]

$$\max z_1 = \sum_{(i,j)\in\mathcal{A}} \sum_t f_{ij}^t \tau$$

f_{ij}^t : edge flow

τ : time discretization unit

2. Minimizing total increases of sidewalk area [m²]

$$\min z_2 = \sum_{(i,j)\in\mathcal{A}^m} (n_{ij} - n_{ij}^{\min}) l_{ij}$$

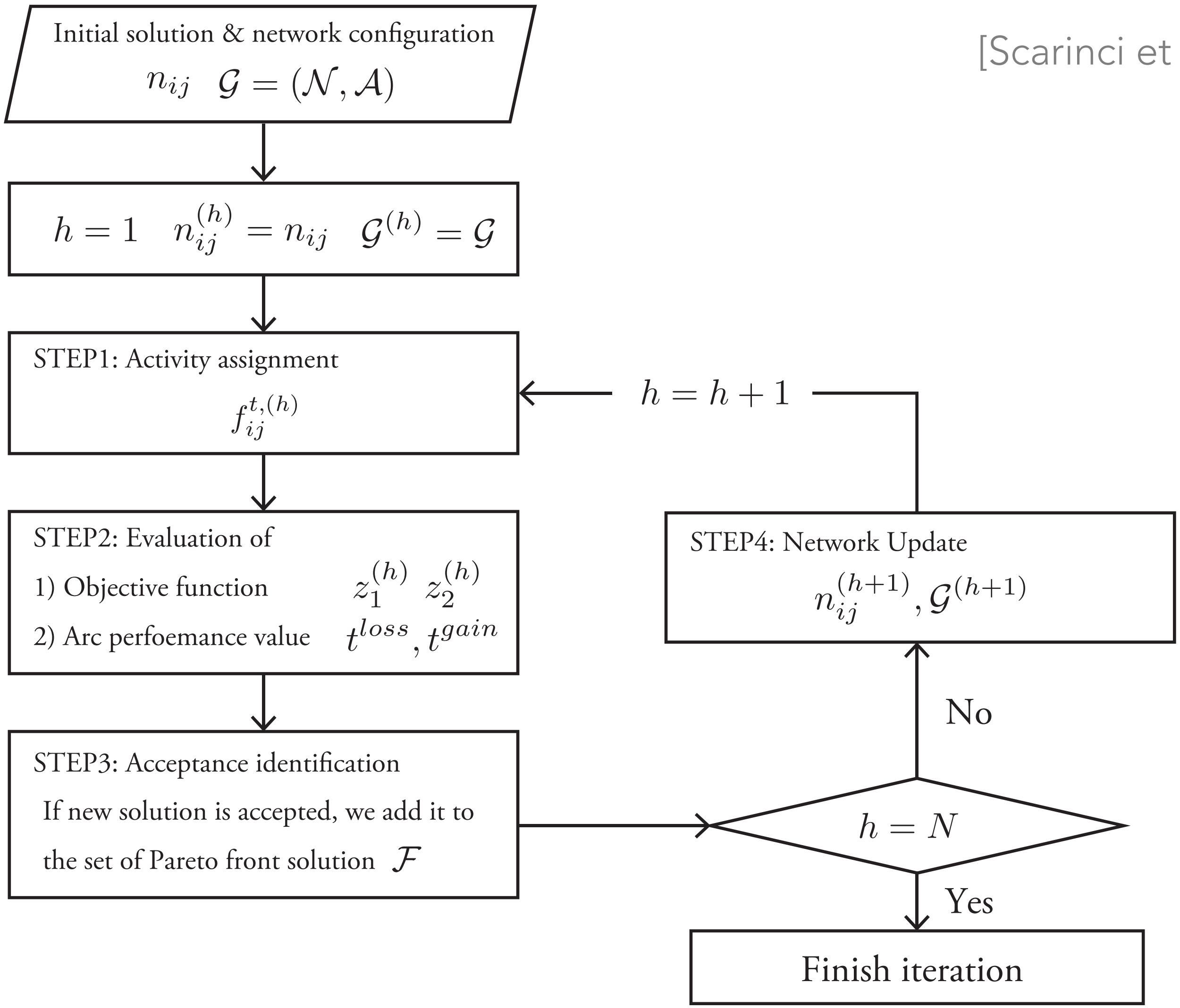
$\mathcal{A}^m$: moving link set

l_{ij} : link length

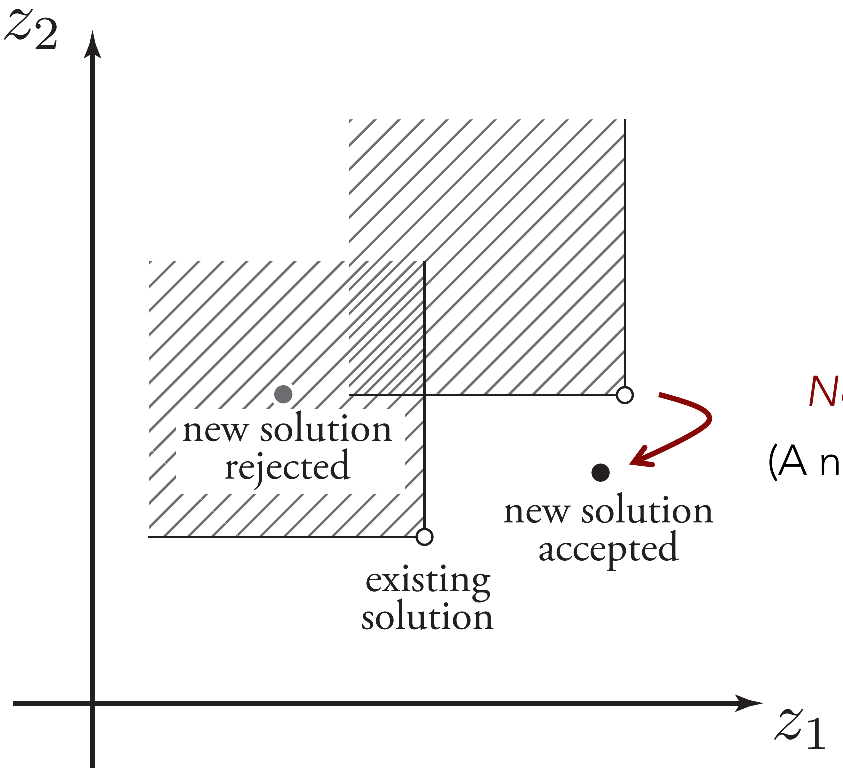
$n_{ij} - n_{ij}^{\min}$: widening width

Solution methodology

[Scarinci et al., 2017]



Acceptance criterion



Network update

(A neighborhood search)

Result | Pareto front solutions

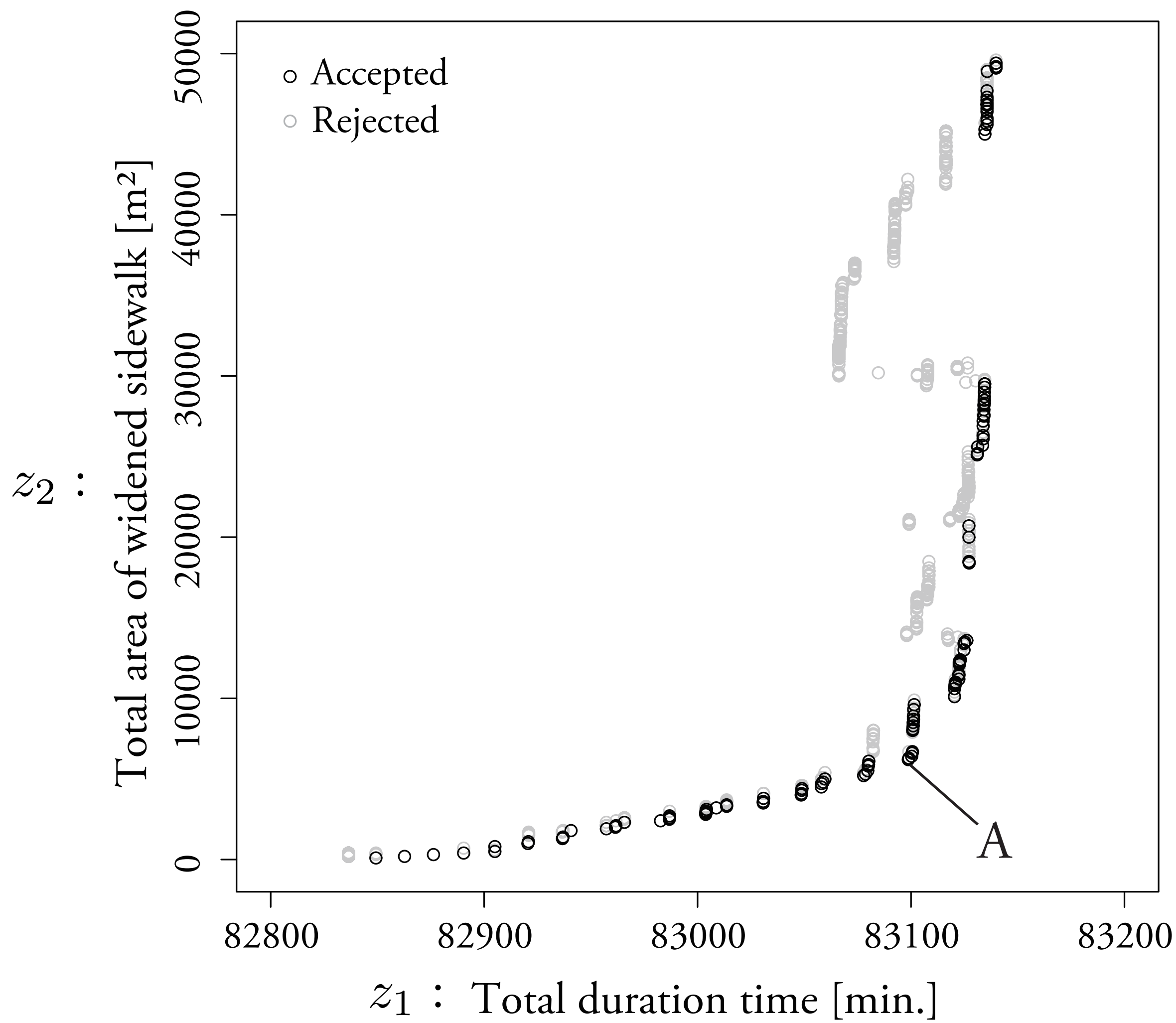


Figure: Trade-off curve between sojourn time and widened sidewalk area (CPU time: 5599.69 [s])

Result | example solution

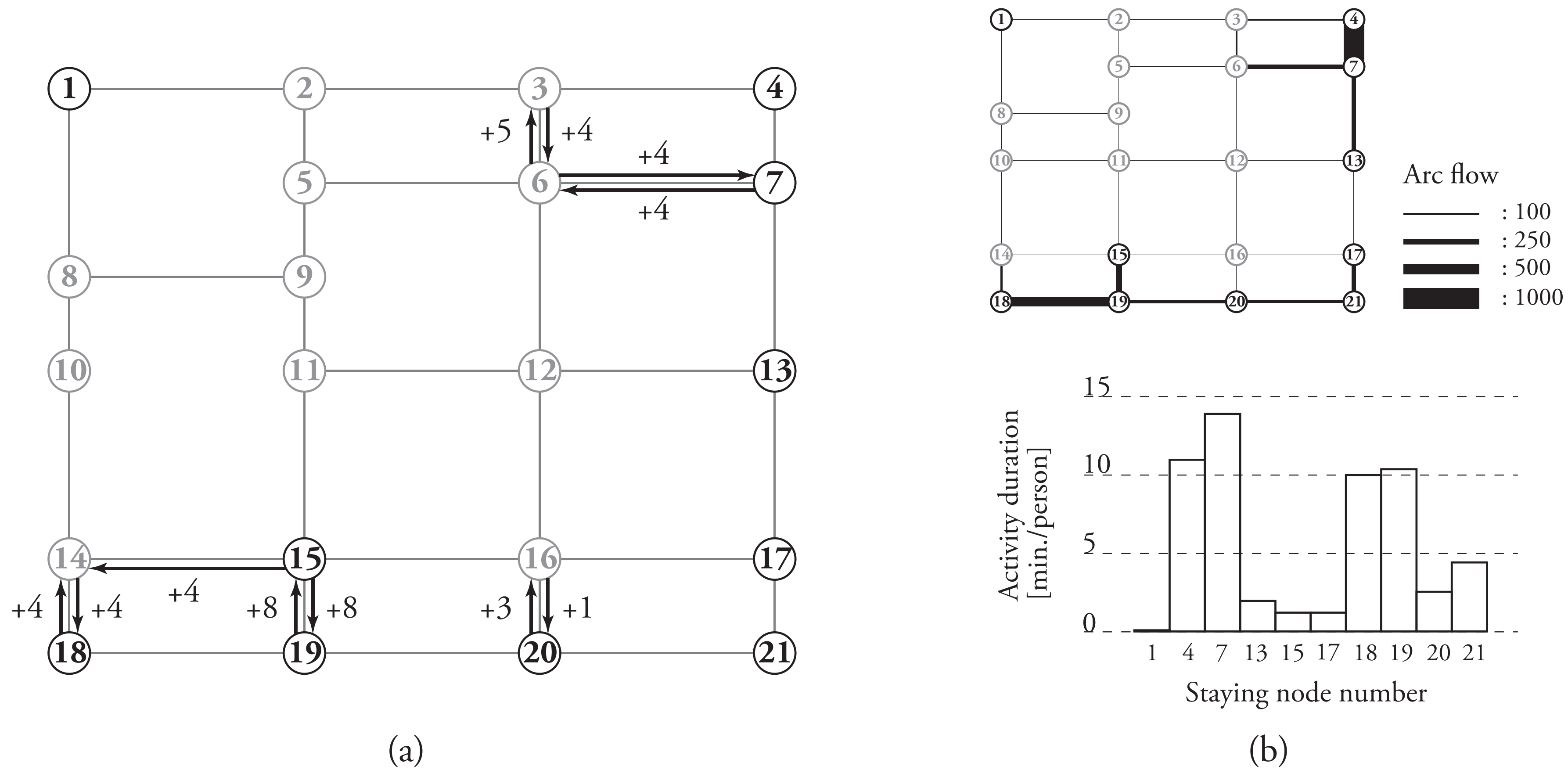


Figure: Variation of (a) a network configuration of an example solution A (b) activity flow in case of the network

Outline

1. Introduction
2. Pedestrian activity assignment
   A) Methodology
   B) Illustrative examples
3. Application to a network design
4. **Conclusion**

Conclusion

- **Modeling pedestrian behavior**
  - ***A probabilistic and dynamic activity path choice model*** is proposed based on the Markov decision process.
  - ***Time-constraint and time discount factor*** are significant parameters for pedestrian activities in city centers.
- **Computable algorithm**
  - ***Markovian assignment*** is equivalent to the MNL model but does not require path enumeration.
  - ***Time-space prism-based network restriction*** removes unreachable states in advance and reduces the size of path set.
- **Network design**
  - ***A pedestrian activity-based network design*** is presented.

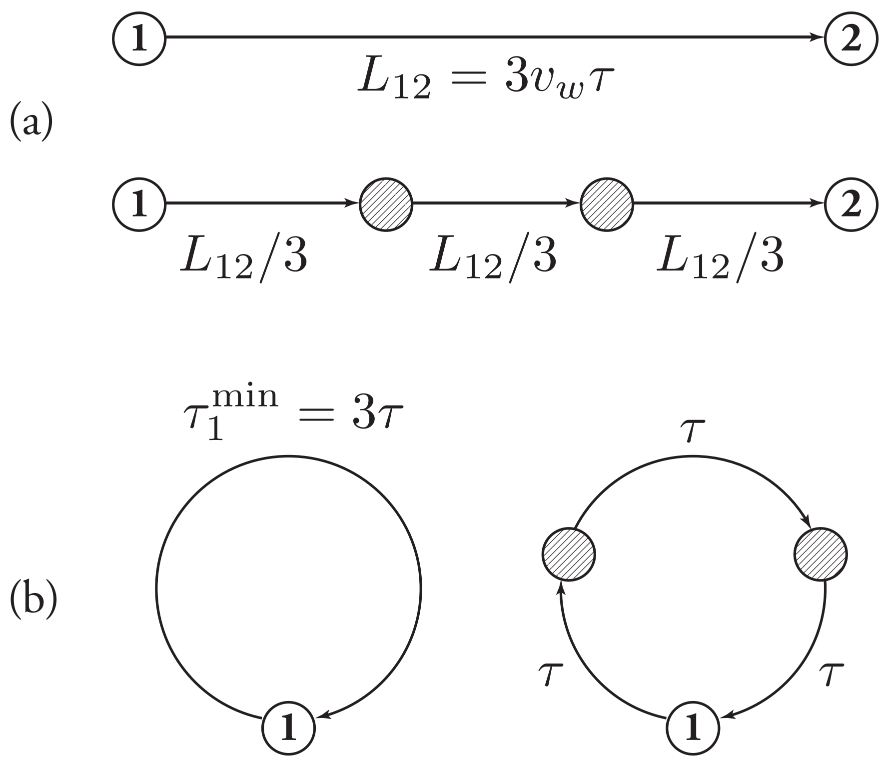
Thank you for attention.

Network restriction

Table: Restricted path set (**24 paths**)

	node number at each time												
path	$t = 0$	1	2	3	4	5	path	$t = 0$	1	2	3	4	5
1	13	8	3	3	8	13	13	13	14	9	9	14	13
2	13	8	7	7	8	13	14	13	14	14	13	13	13
3	13	8	8	8	8	13	15	13	14	14	14	13	13
4	13	8	8	8	13	13	16	13	14	14	14	14	13
5	13	8	8	13	13	13	17	13	14	15	15	14	13
6	13	8	9	9	8	13	18	13	14	19	19	14	13
7	13	12	7	7	12	13	19	13	18	17	17	18	13
8	13	12	11	11	12	13	20	13	18	18	13	13	13
9	13	12	12	12	12	13	21	13	18	18	18	13	13
10	13	12	12	12	13	13	22	13	18	18	18	18	13
11	13	12	12	13	13	13	23	13	18	19	19	18	13
12	13	12	17	17	12	13	24	13	18	23	23	18	13

Network standardization



L_a	: arc length [m]	τ	: interval of time discretization [s]
v_w	: walking speed [m/s]	$\tau_i^{\min}$	: minimum duration time of staying node [s]

Network design

Network Update

Remove-Random-Width

Remove a unit width $\tilde{n}$ from an arc randomly selected:

$$n_{ij}^{(h+1)} = n_{ij}^{(h)} - \tilde{n} \quad \text{s.t.,} \quad n_{ij}^{(h)} \geq \tilde{n}$$

Add-Random-Width

Add a unit width $\tilde{n}$ from an arc randomly selected:

$$n_{ij}^{(h+1)} = n_{ij}^{(h)} + \tilde{n} \quad \text{s.t.,} \quad n_{ij}^{(h)} + \tilde{n} \leq \kappa_{ij}$$

Likewise,

Remove-Worst-Width

Add-Best-Width

where the ***worst*** and ***best*** are defined with ***arc performance value*** (in the next slide).

Network design

Arc performance

Utility loss (gain) for identifying the worst (best) moving arc:

$$\phi_{ij}^{\text{loss}} = \{\hat{v}_{ij}(n_{ij}^{(h)} - \tilde{n}) - \hat{v}_{ij}(n_{ij}^{(h)})\} \cdot f_{ij} \qquad \forall (i,j) \in \mathcal{A}^m$$

$$\phi_{ij}^{\text{gain}} = \{\hat{v}_{ij}(n_{ij}^{(h)} + \tilde{n}) - \hat{v}_{ij}(n_{ij}^{(h)})\} \cdot f_{ij} \qquad \forall (i,j) \in \mathcal{A}^m$$

Network design

Parameters

$\tilde{n} = 1$: unit removal/additional width [m]

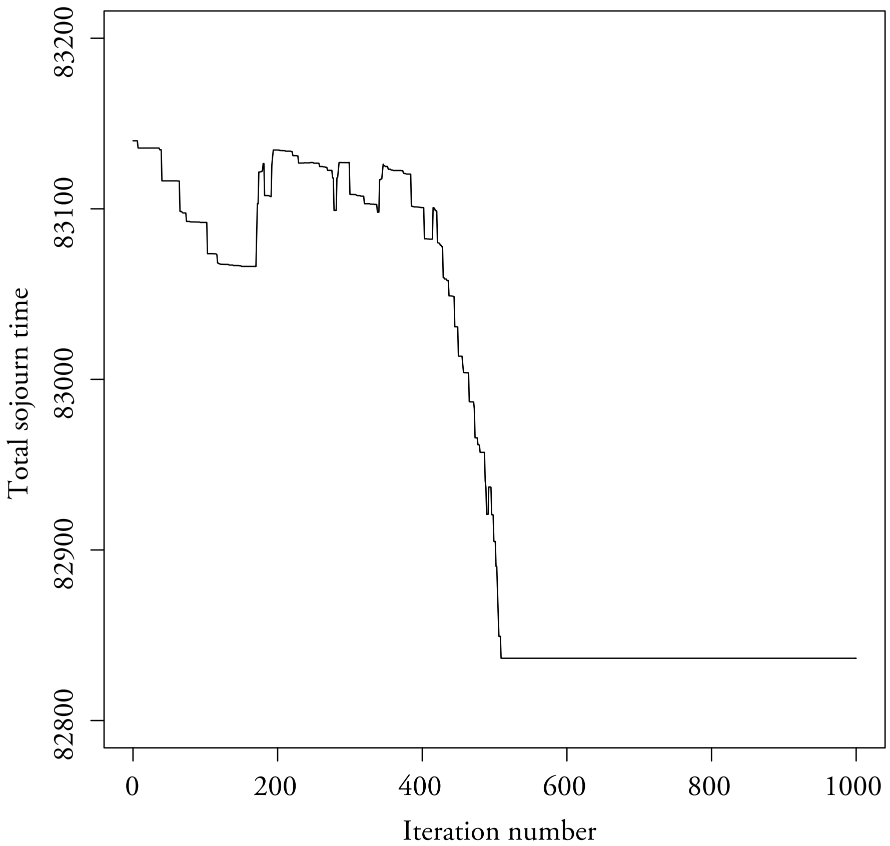
$\hat{c} = 1$: unit capital cost [Yen/m2]

$N = 2000$: Iteration number

Solution

- Neighborhood structure of Network Update is **selected randomly**.
- Initial solution is **full-equipped** network.

Pareto front search



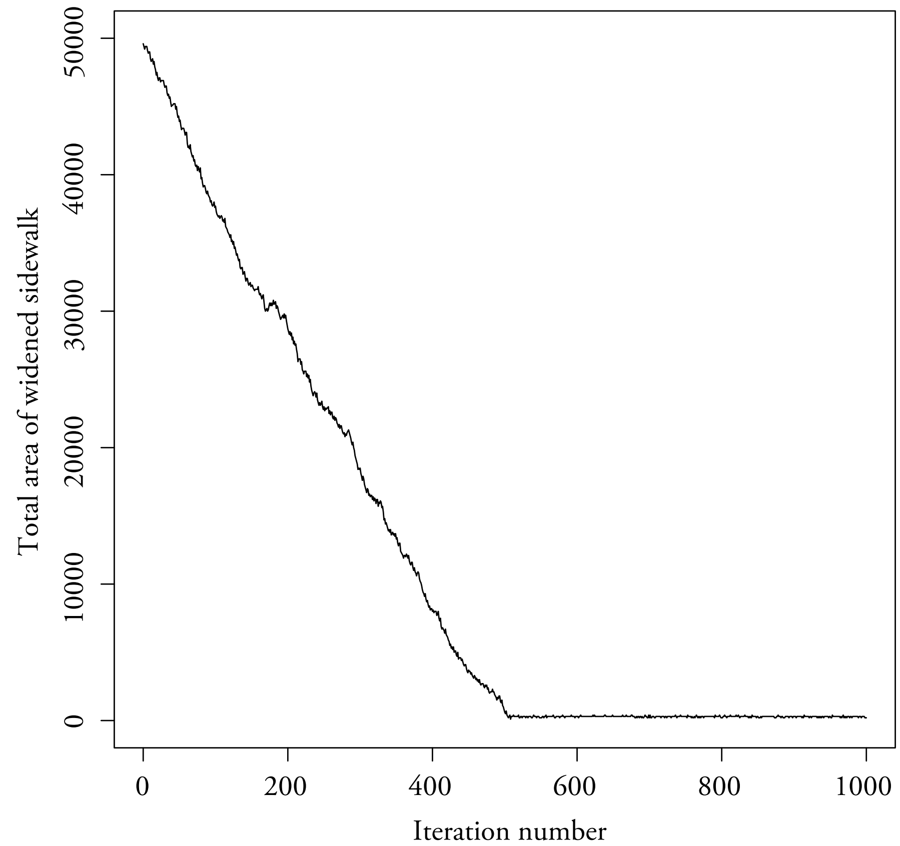


Figure: Variation of (a) total sojourn time and (b) total area of widened sidewalk in iteration process.