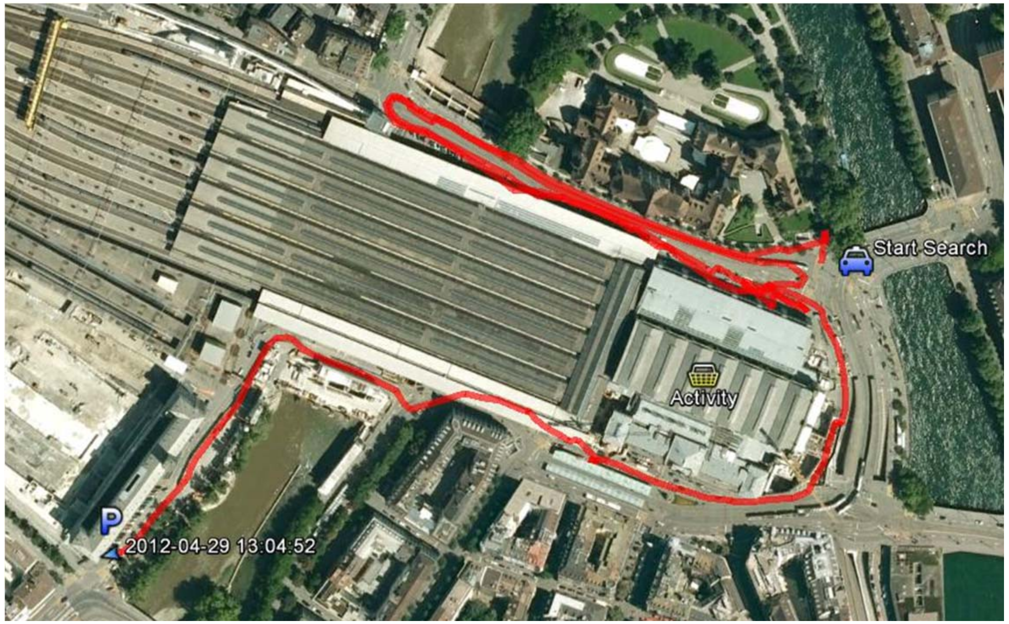
Parking

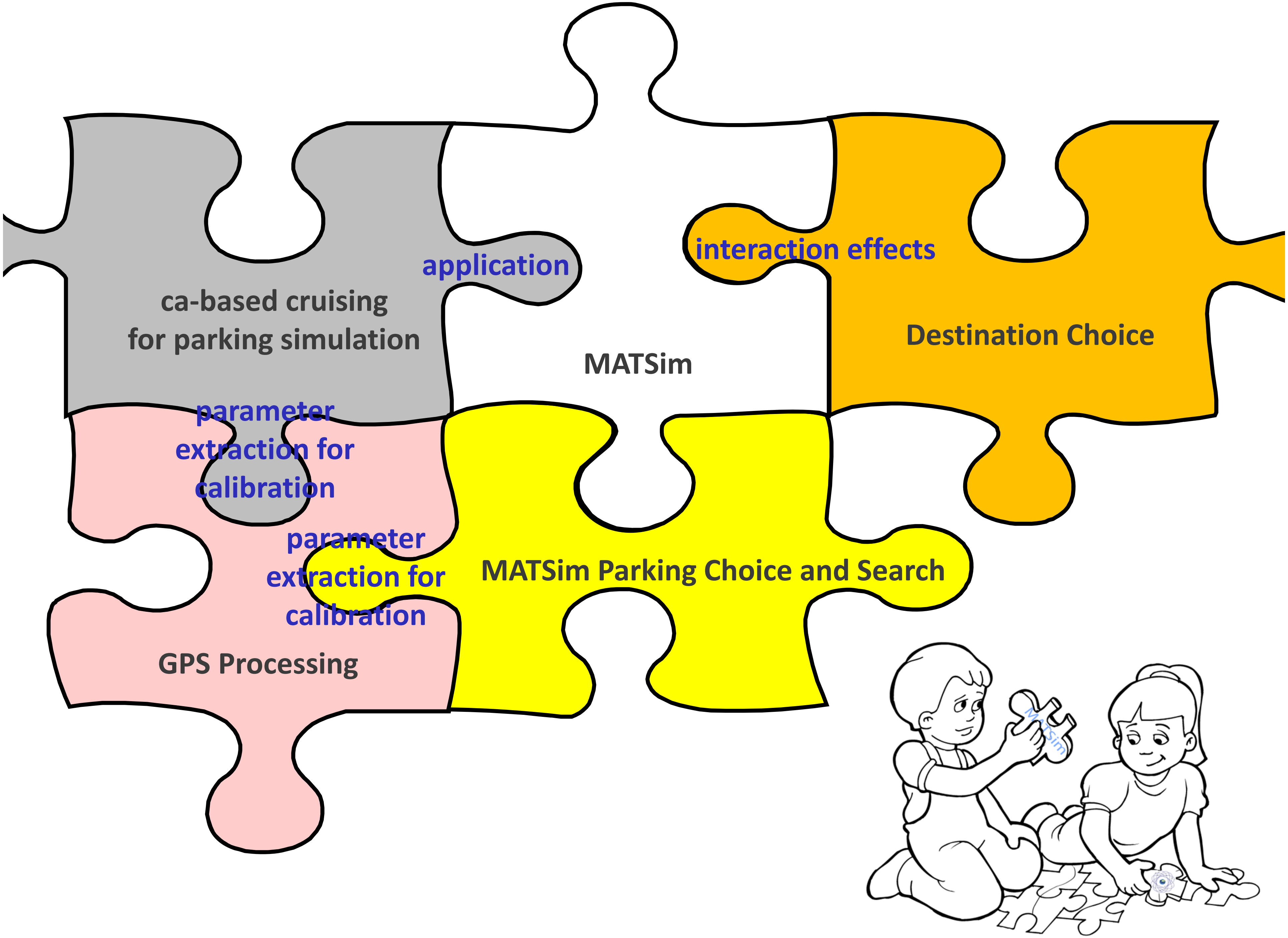


Horni A., L. Montini and R. A. Waraich

IVT
ETH
Zürich

June 2012

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich


application
interaction effects
ca-based cruising
for parking simulation
MATSim
Destination Choice
parameter
extraction for
calibration
parameter
extraction for
calibration
MATSim Parking Choice and Search
GPS Processing

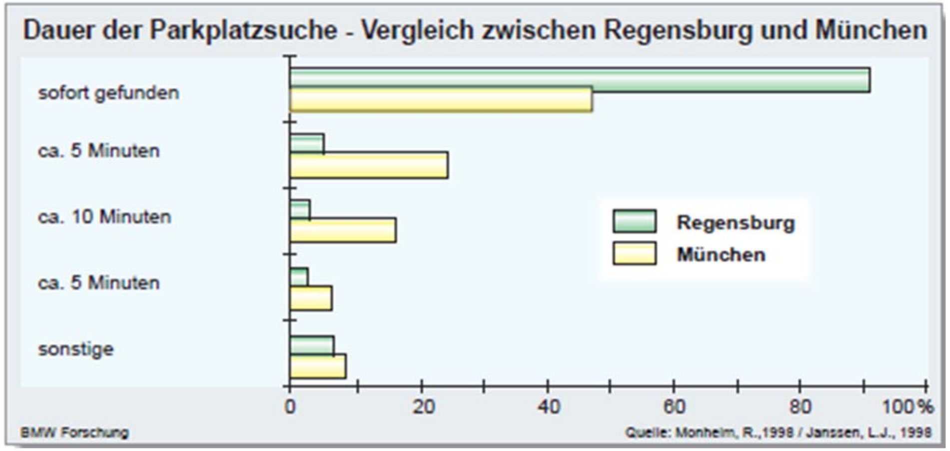
Parking Search – A Big Problem or a Phantom?

Studies

D. C. Shoup (2006) Cruising for parking

"[…], studies of cruising in congested downtowns have found that it took between 3.5 and 14 min to find a curb space, and that between 8 and 74 percent of the traffic was cruising for parking."

München / Regensburg

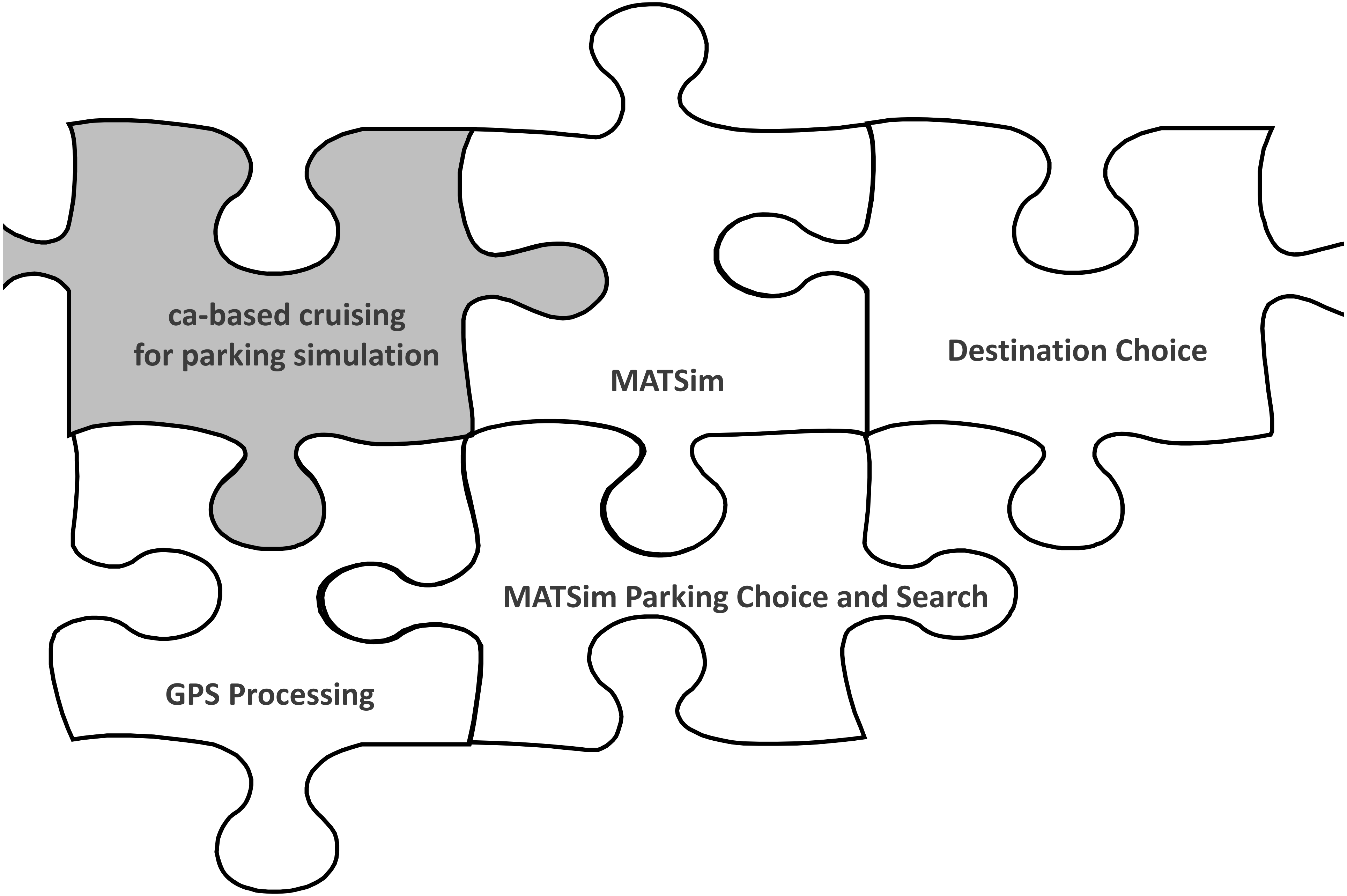


Frankfurt (1992- 1993)

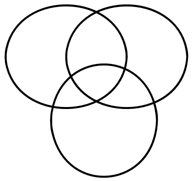
average search times on Saturdays 5.5 – 10 minutes (decreasing after introduction of PGI)

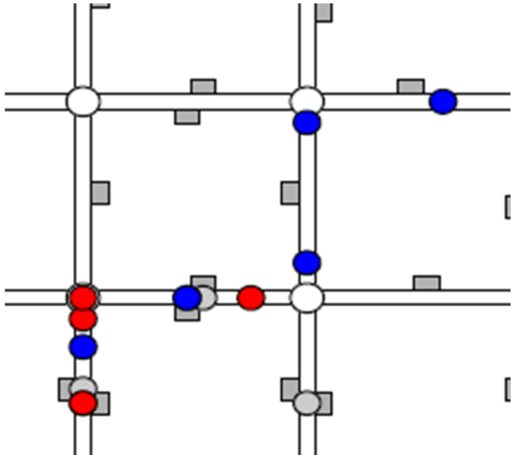
Zürich (2010)

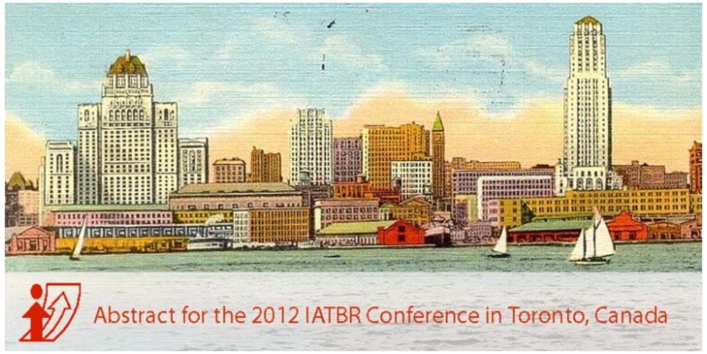
parking occupancy inner city (Saturday) ~ 97 %


ca-based cruising
for parking simulation
MATSim
Destination Choice
MATSim Parking Choice and Search
GPS Processing

LaHowara & Commander Spock: Lecture MSSSM







Exploring en-route parking type and parking-search route choice: decision making framework and survey design

Sigal Kaplan[1], Shlomo Bekhor[2]

Kaplan and Bekhor (2011): very similar approach, GPS study

→ personal (controversial) communication started

→ meeting Carlo Prato @ IATBR

Goal of CA

Implement agent-based cellular automaton cruising-for-parking simulation to generate aggregate models for parking search key measures:

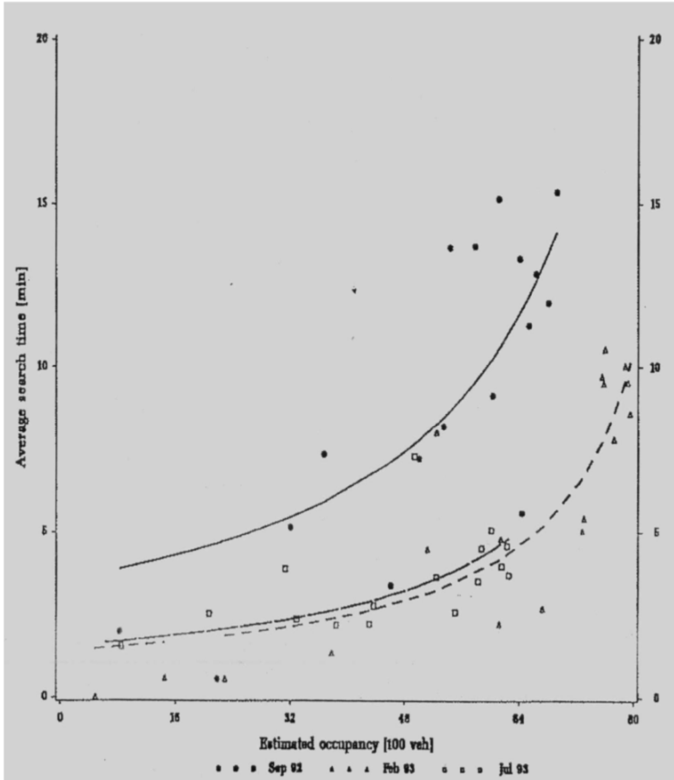
here: $\tilde{\boldsymbol{t}} = \boldsymbol{f}(\boldsymbol{supply})$

supply: # parking spaces in area

(future: density around destination)

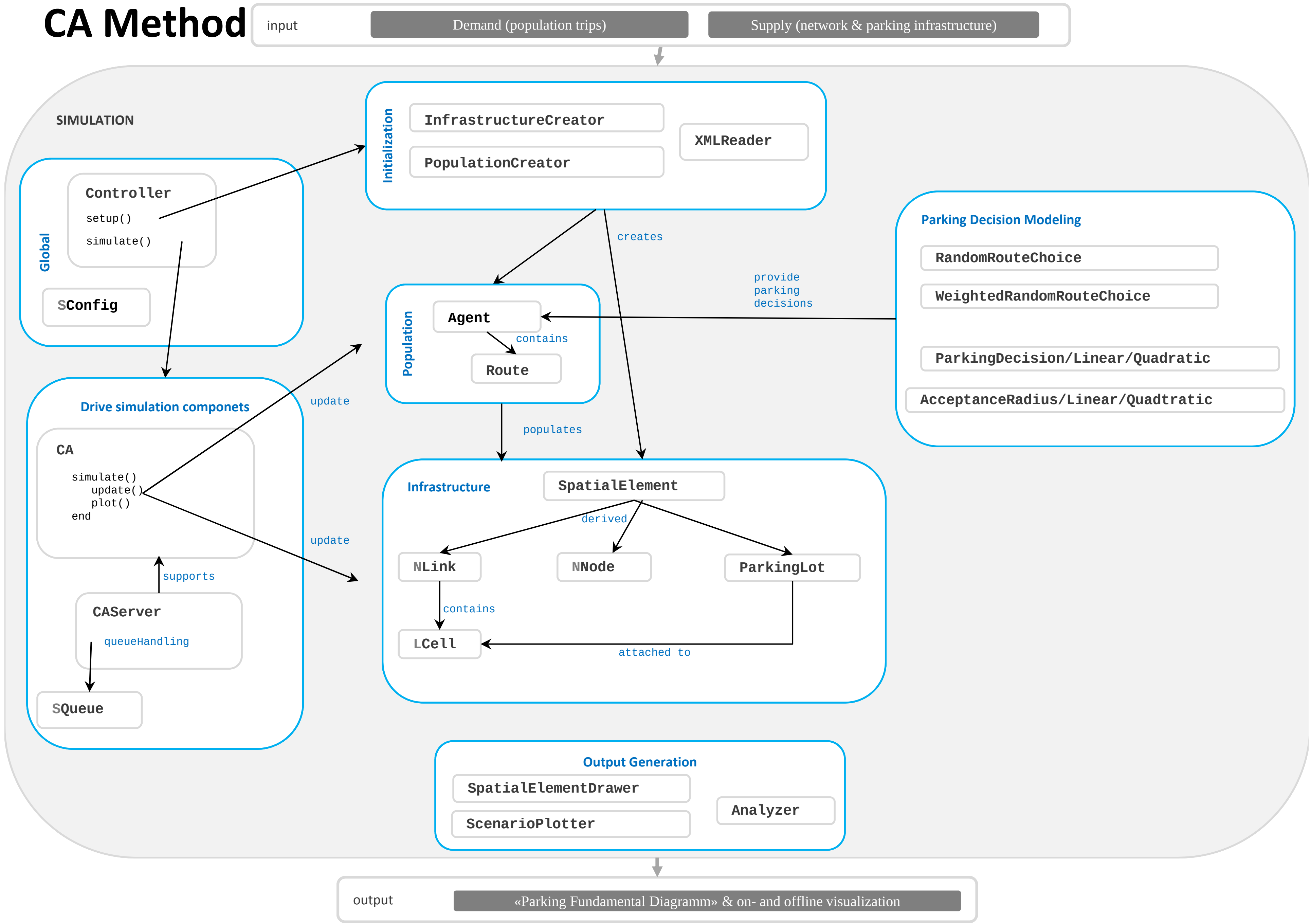
similar to estimated functions

such as …



Axhausen et al. (1994)

PGI Frankfurt a. M.


CA Method
input
Demand (population trips)
Supply (network & parking infrastructure)
SIMULATION
Initialization
InfrastructureCreator
PopulationCreator
XMLReader
Global
Controller
setup()
simulate()
SConfig
creates
provide
parking
decisions
Parking Decision Modeling
RandomRouteChoice
WeightedRandomRouteChoice
ParkingDecision/Linear/Quadratic
AcceptanceRadius/Linear/Quadtratic
Population
Agent
contains
Route
populates
Drive simulation componets
update
CA
simulate()
update()
plot()
end
update
Infrastructure
SpatialElement
derived
NLink
NNode
ParkingLot
contains
LCell
attached to
supports
CAServer
queueHandling
SQueue
Output Generation
SpatialElementDrawer
ScenarioPlotter
Analyzer
output
«Parking Fundamental Diagramm» & on- and offline visualization

CA Method: Technical but Important Details

- update process on randomly chosen links, nodes and parking lots as in famous Nagel and Schreckenberg (1992) CA
  - increase, decrease speed dependent on load of link (with v_{max} and v_{min})
  - randomize speeds
  - future: parking search speeds

- *CAServer* class for update process:
  - not naively iterating over all agents and infrastructure elements (e.g., cells) but only over occupied ones -> **queues** of agents, links nodes and parking lots

- resolution
  - Queue models – CA – car following models

- jam density used for cell size as in Wu and Brilon (1997)
  - Future: maybe pool cells in free flow conditions

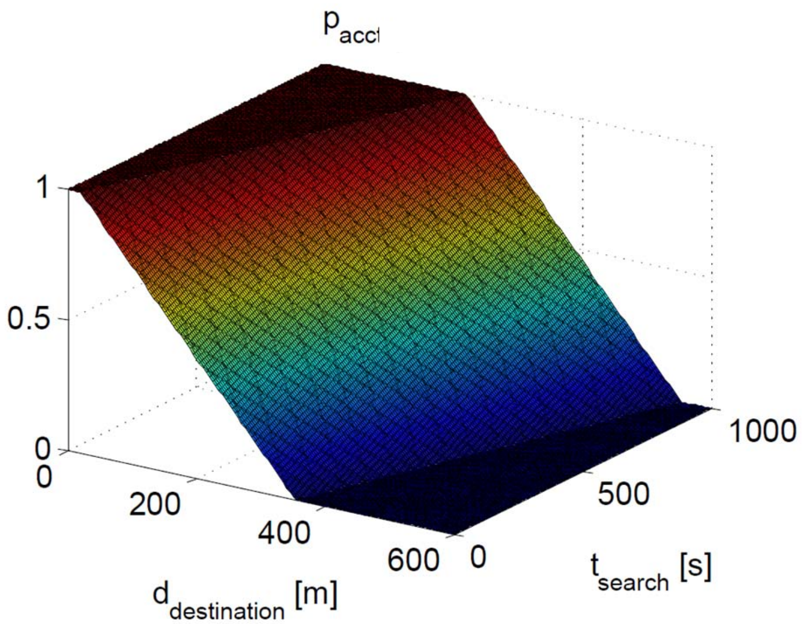
CA Method: Parking Search Modeling

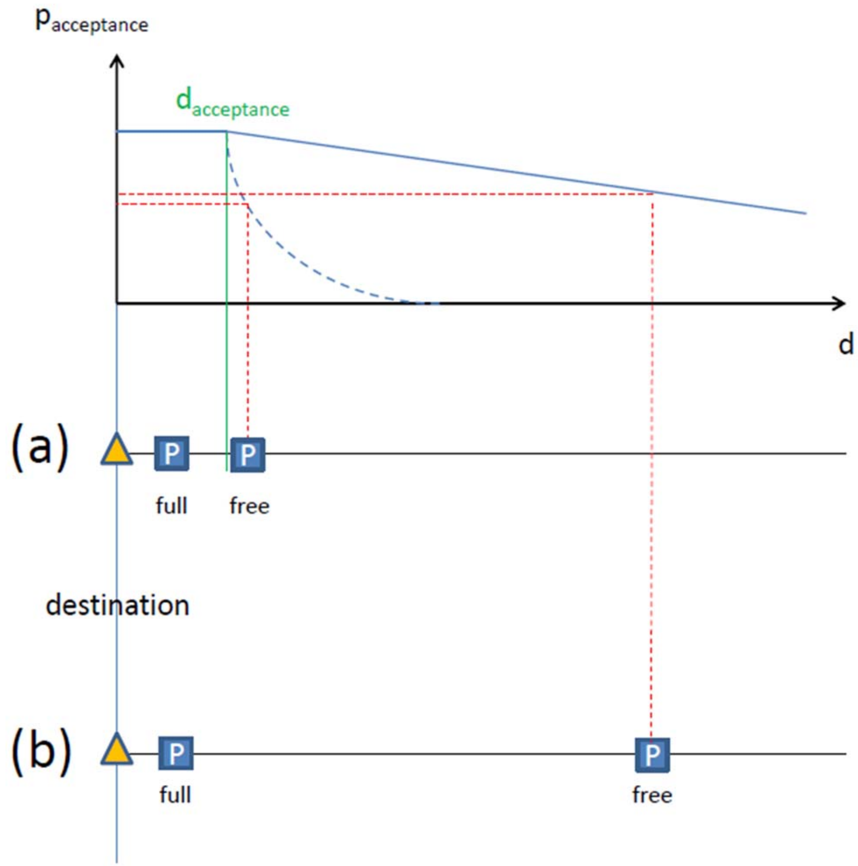
- parking type choice
  - exogenously, derived from supply (for ZH scenario only)

- search tactic
  - search starting point
  - weighted random walk
    - destination approaching efficiency
    - agent's memory of parking lots with free spaces

CA Method: Parking Search Modeling → Calibration

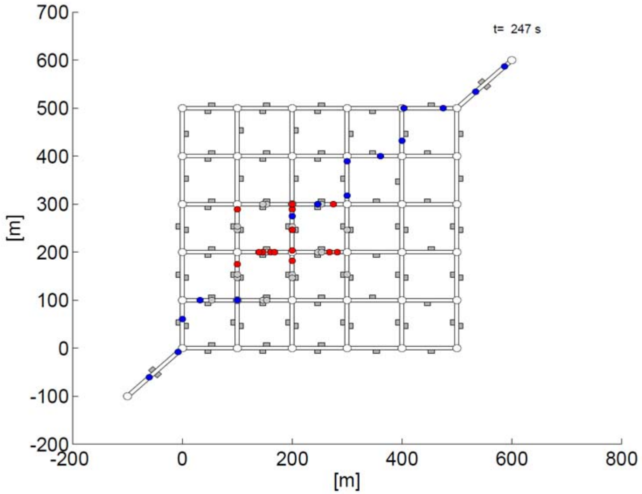
- parking lot choice
  - Acceptance radius

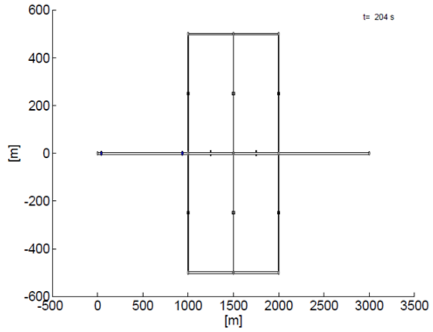


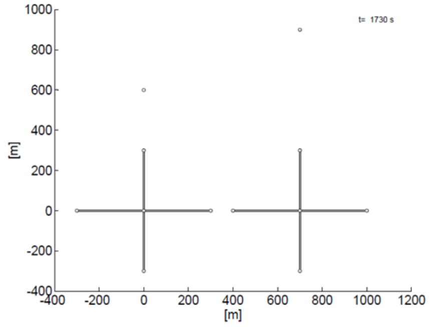


Results and Scenarios

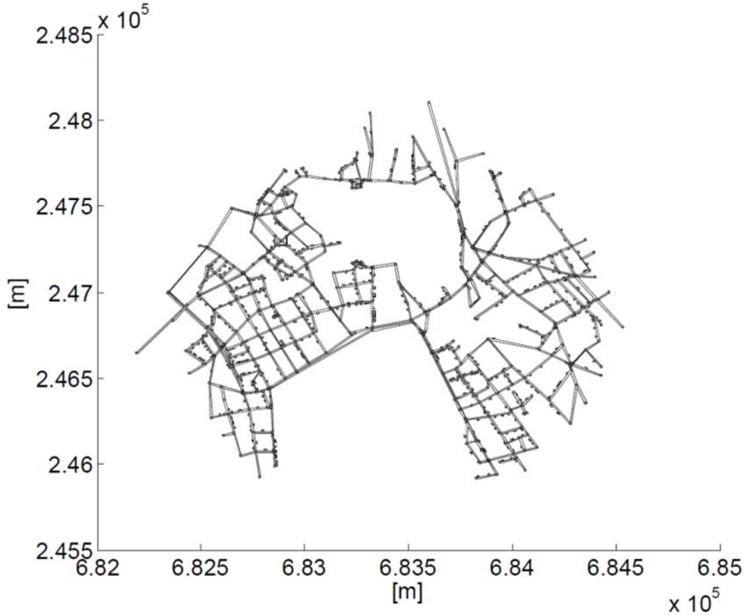
- 3 small-scale scenarios for development and calibration





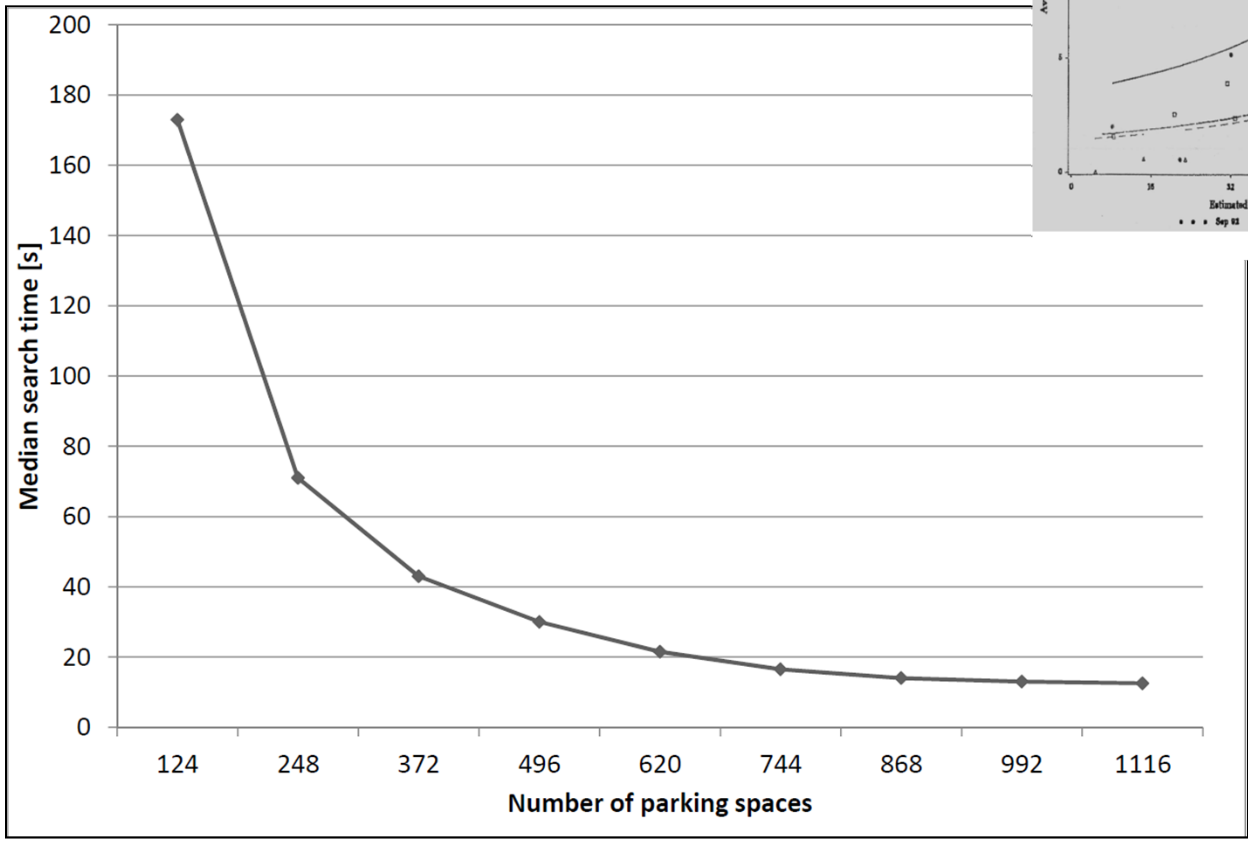


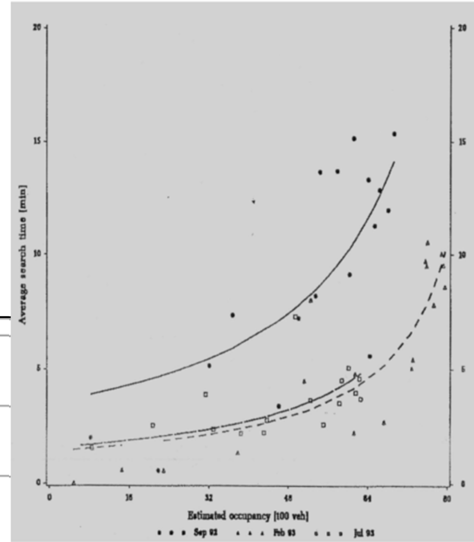
- Zurich Inner-city scenario
  - derived from real-world data (MATSim demand), navigation network
  - ready, but not yet calibrated & speed issues!



Results: Chessboard Scenario

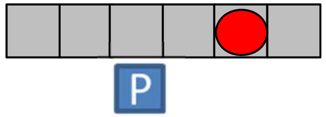
- 100 agents
- 2 origins, 1 destination
- 30 min simulated

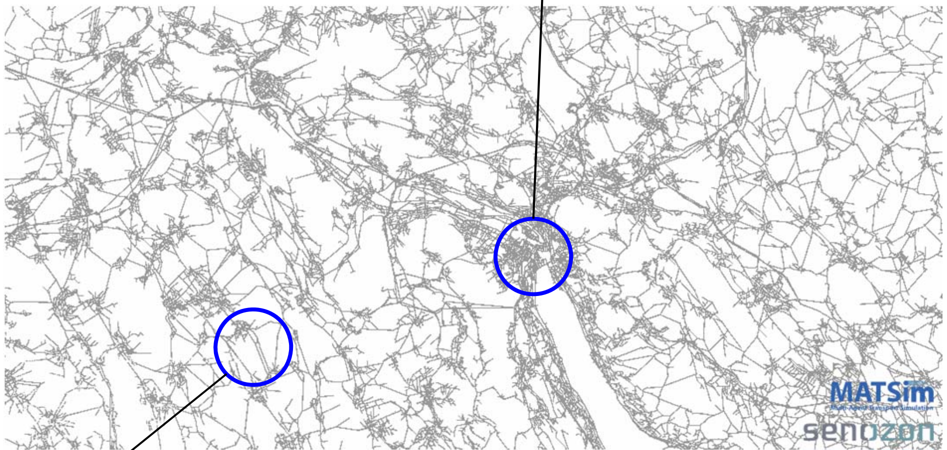




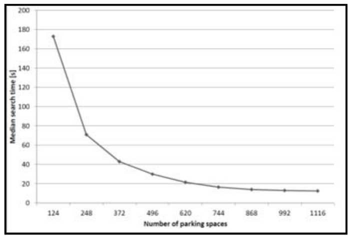
Future: Application in MATSim: Hybrid Approach

t_{search} = simulate with CA



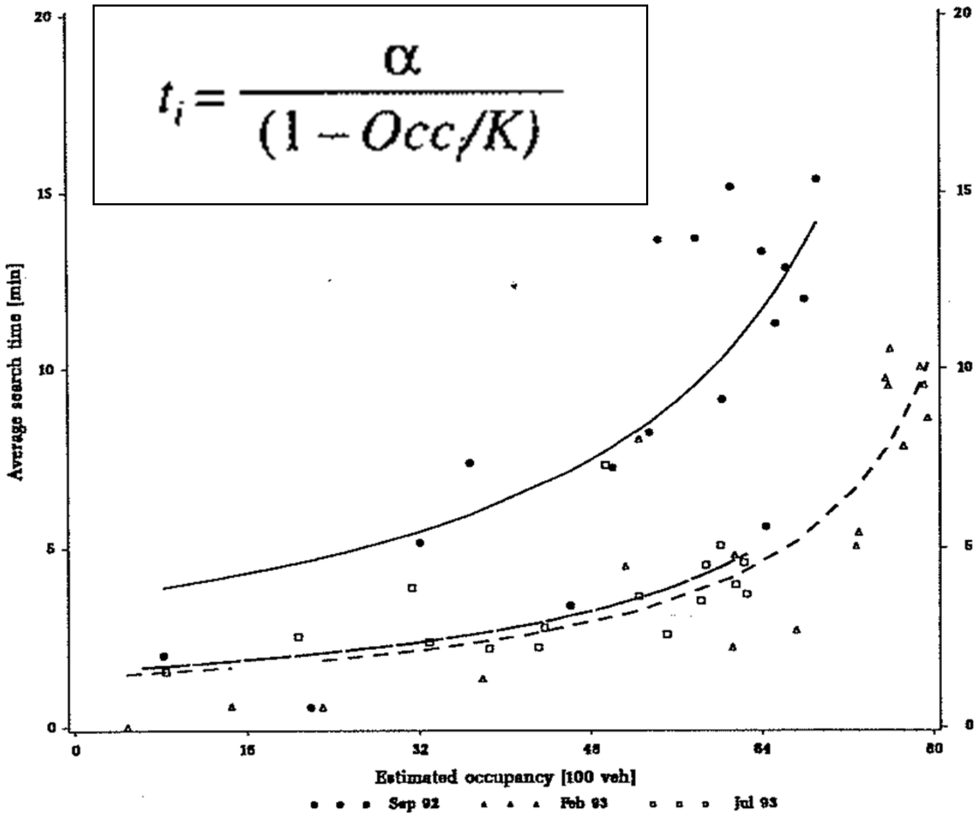


t_{search} = sample from aggregate functions



really necessary apart from parking studies?
costs?

Future: Aggregate Curves Corrections



Δ (GPS – t_i)

→ only rel. diff ($\Delta\Delta$) relevant (st vs. dt)

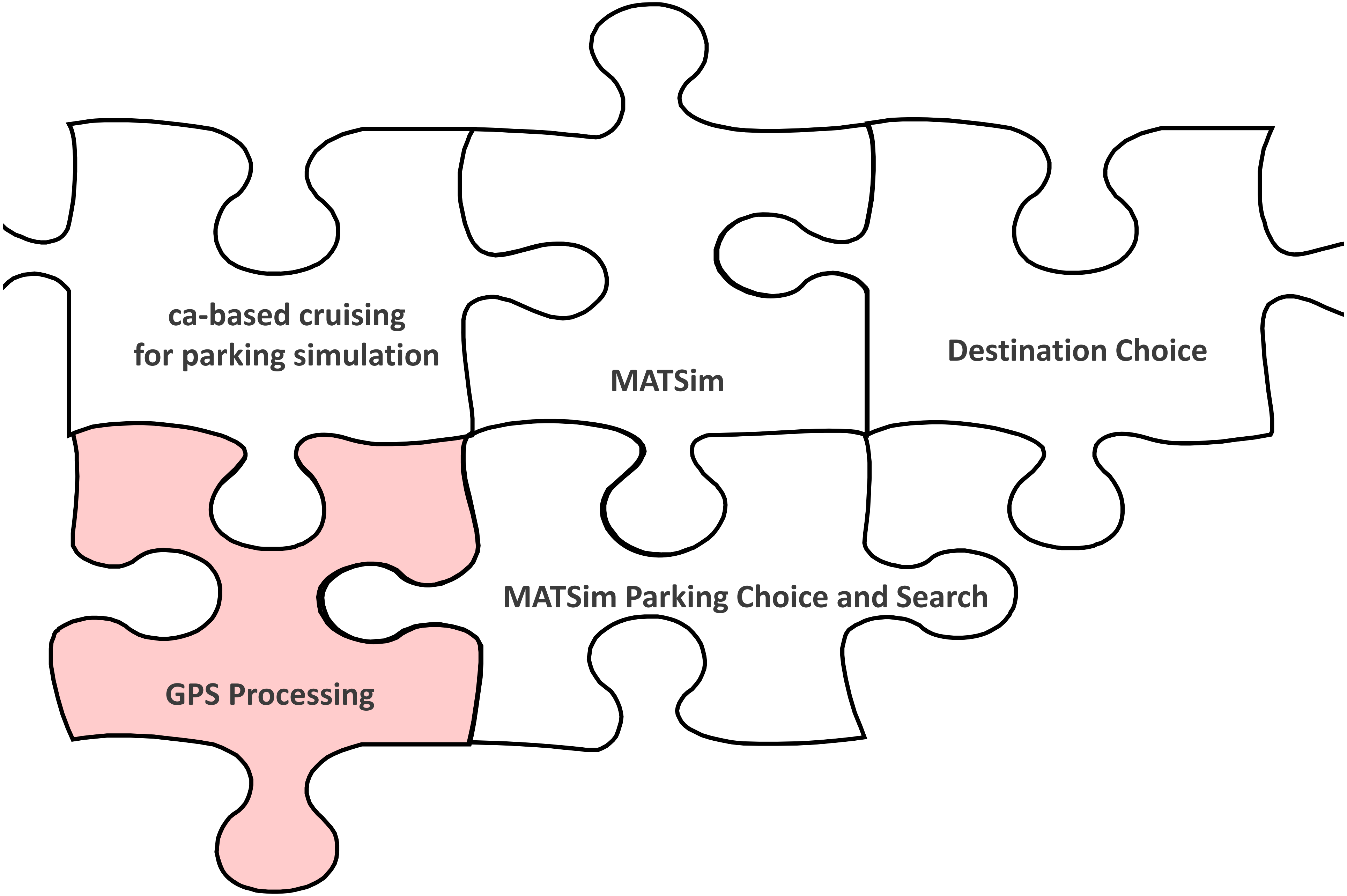
→ plain-colored?

fitting for high Occ

correction factor as a follow-up work?

Discussion: LaHowara & Commander Spock

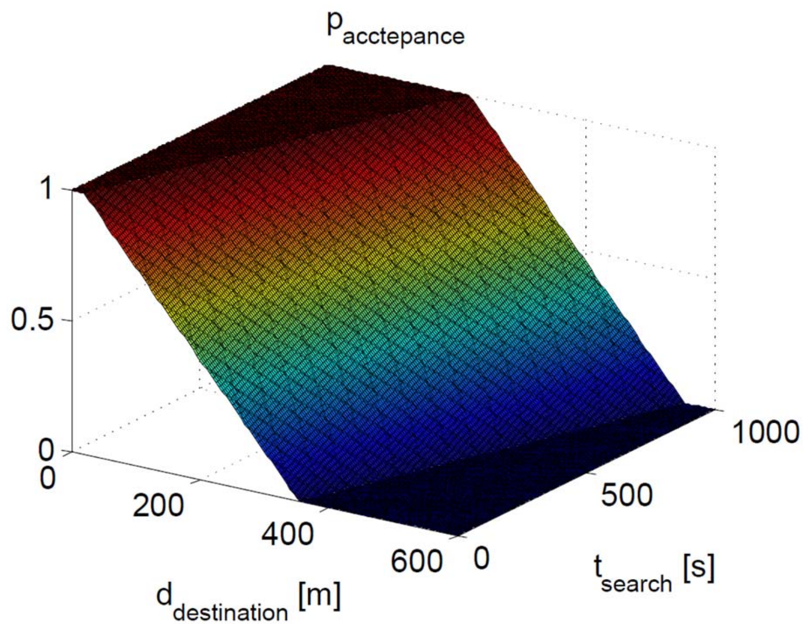
- estimated aggregate functions can be reproduced
- software structure very similar to MATSim -> easy migration
- high simulation costs


ca-based cruising
for parking simulation
MATSim
Destination Choice
MATSim Parking Choice and Search
GPS Processing

GPS Processing – Extract Parameters

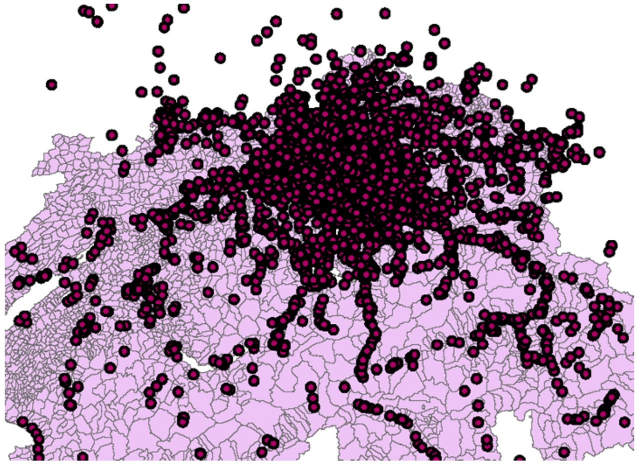
Calibration

- $p_{acceptance}$



GPS Data Available

approx. 32’000 person days from Zurich and Geneva

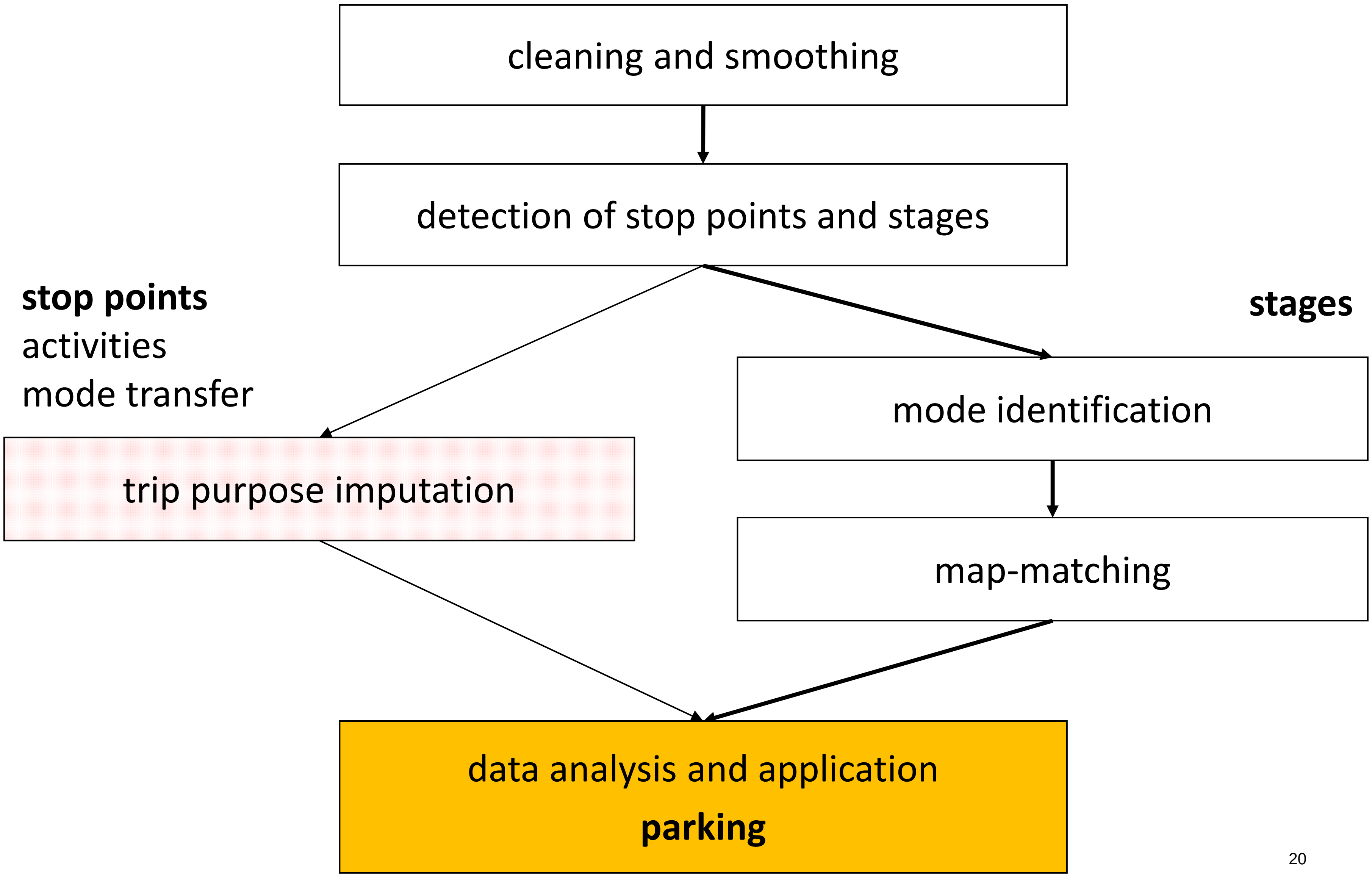


person-based and therefore multi-modal

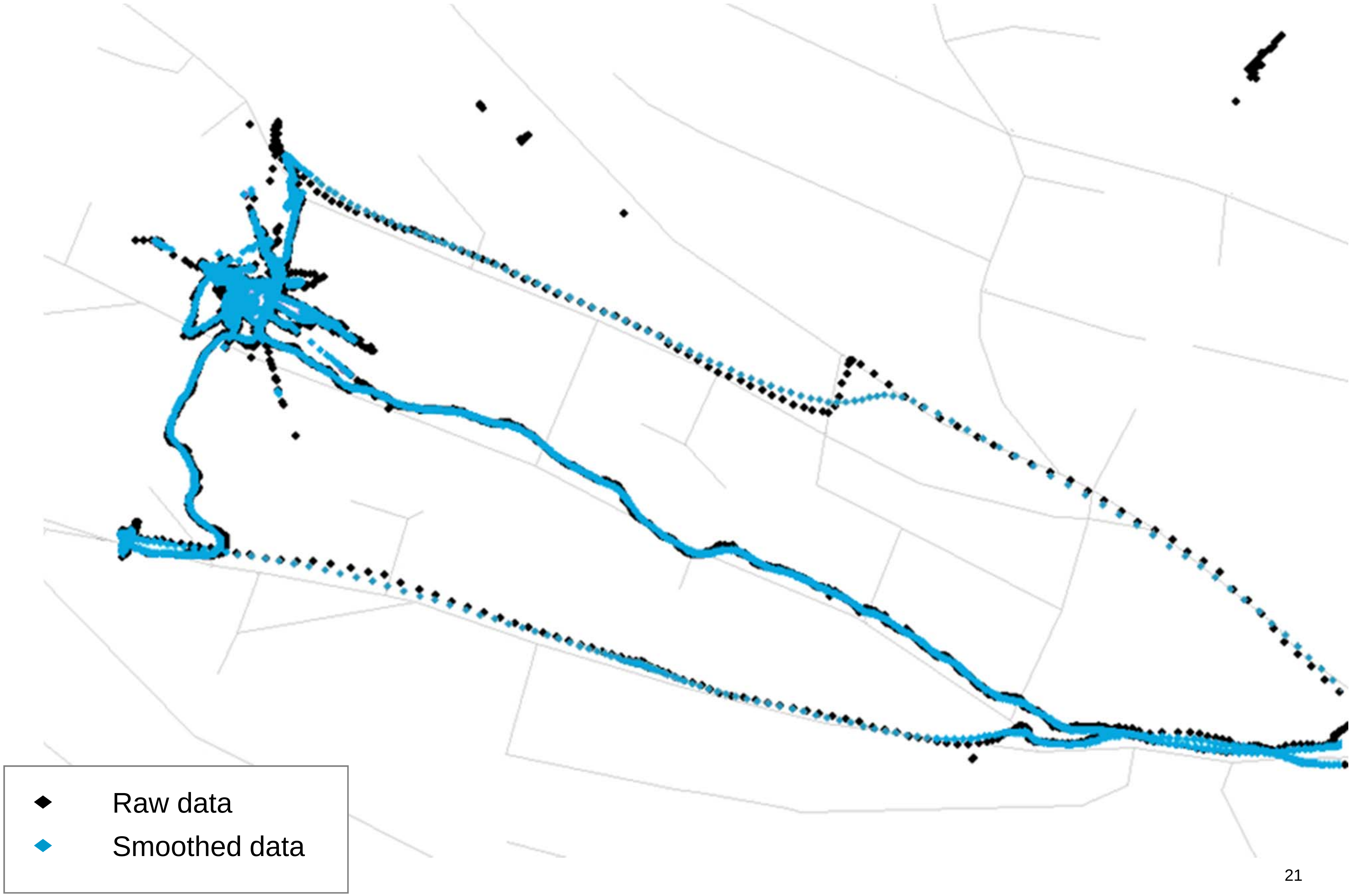
only raw data (x, y, z, timestamp)

no sociodemographics

Processing of GPS data

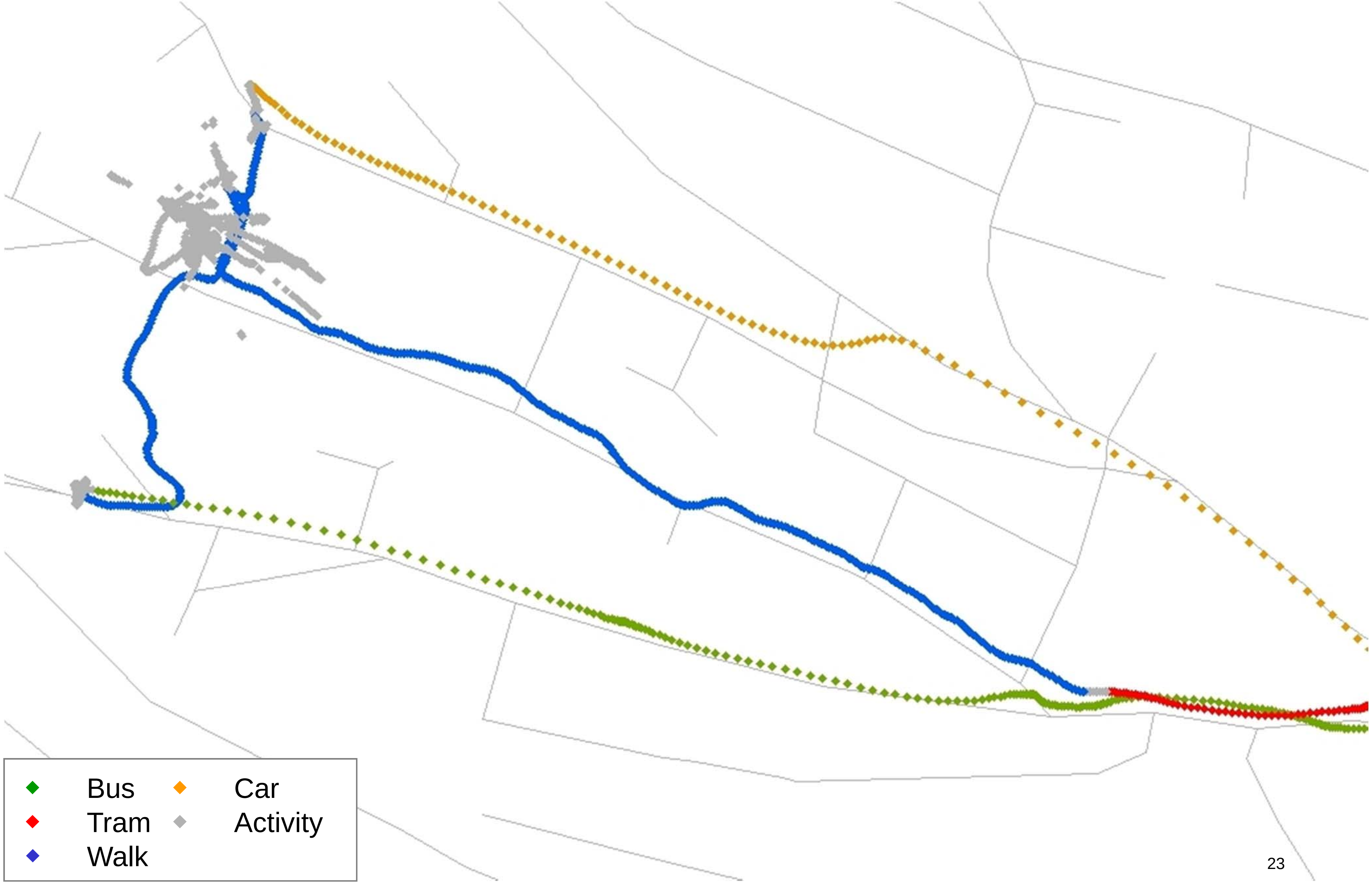


Data cleaning and smoothing

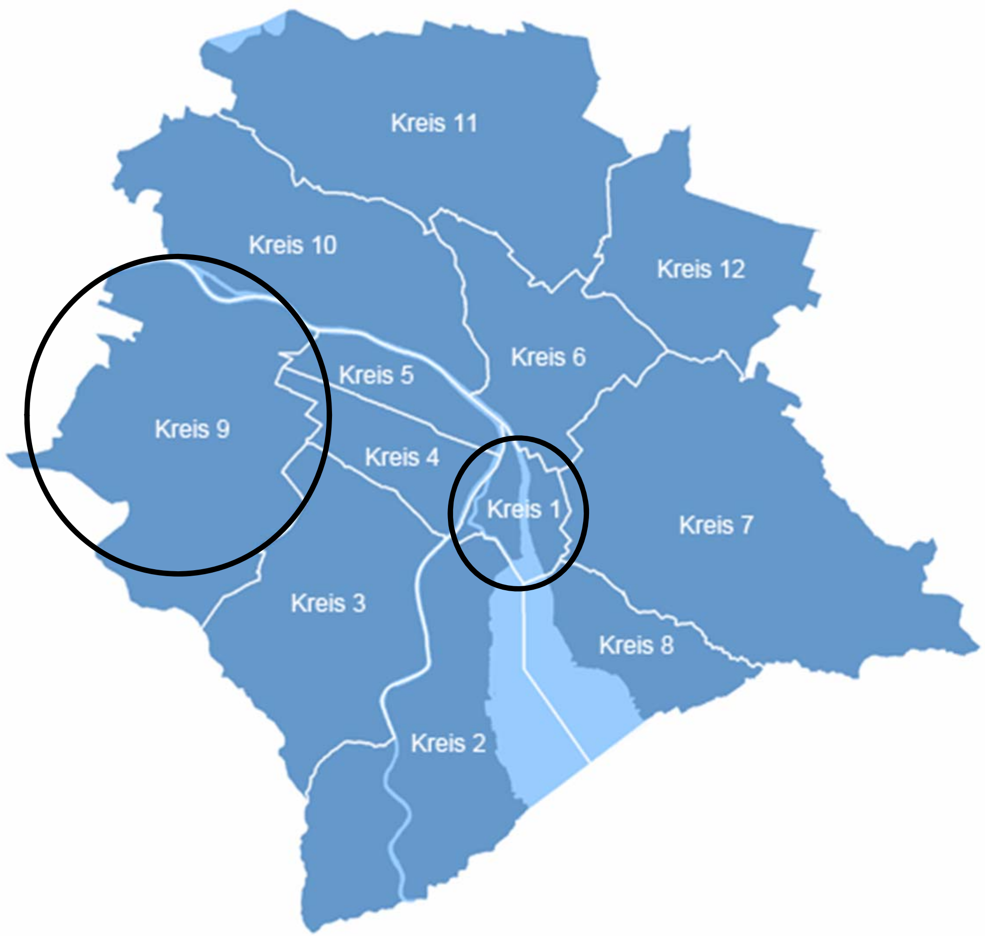


Determining stages and activities

Mode detection

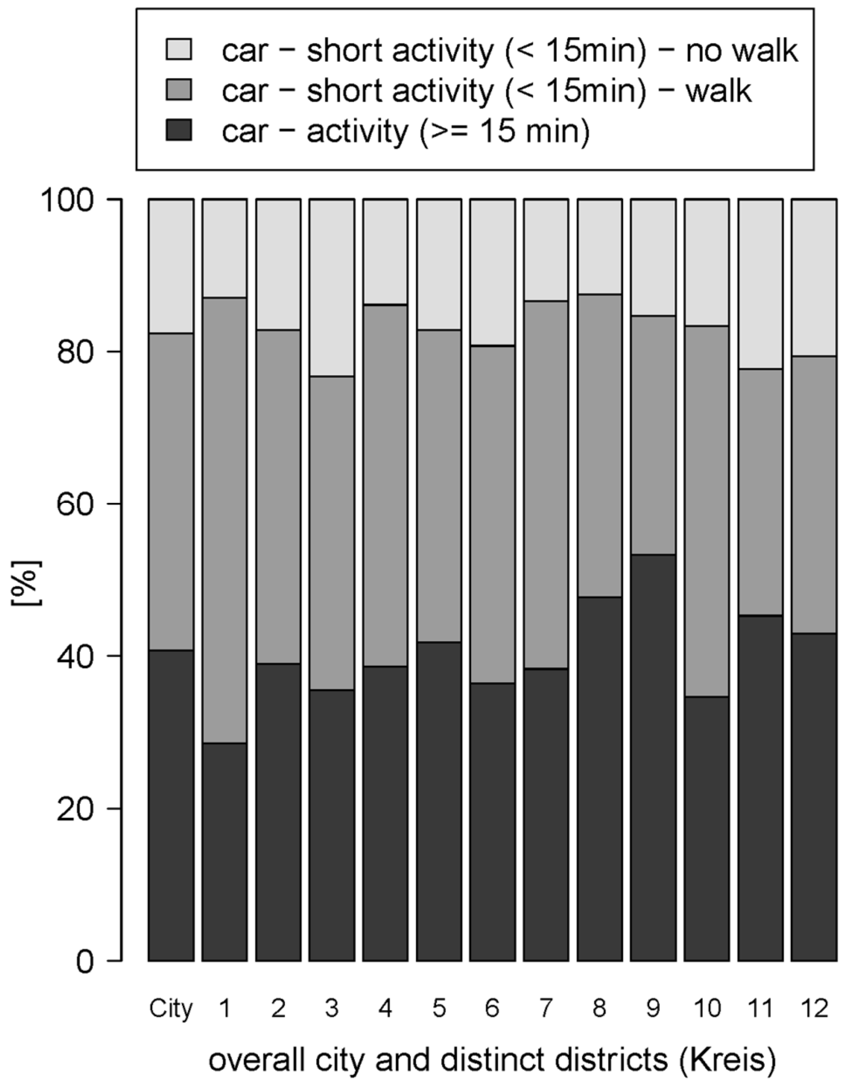


City of Zurich



District	PP/(Res.+Empl.)
1	0.13
2	0.39
3	0.32
4	0.31
5	0.34
6	0.35
7	0.42
8	0.39
9	0.45
10	0.41
11	0.40
12	0.39
City	0.36

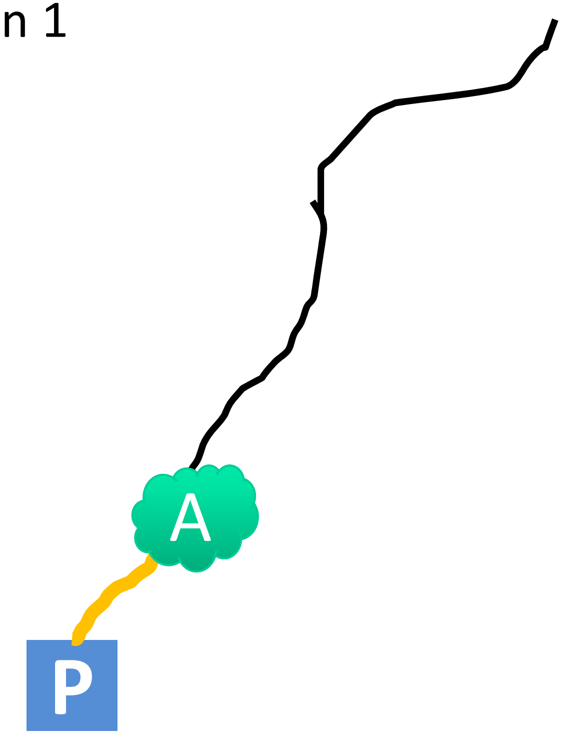
City of Zurich – Analysis per District



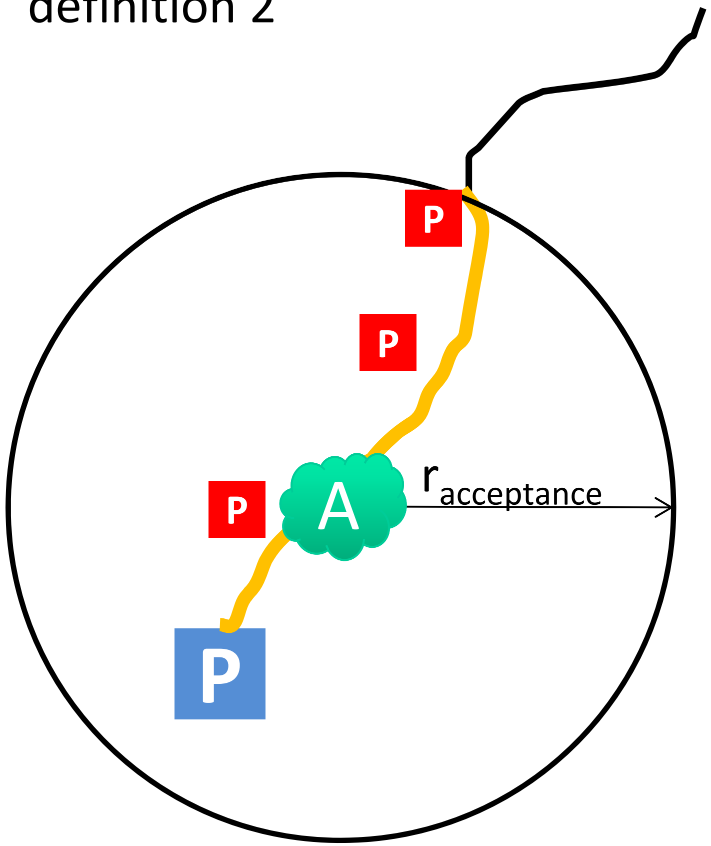
District	Cases
1	294
2	372
3	442
4	368
5	321
6	239
7	269
8	176
9	458
10	312
11	665
12	170
City	4086

Parking Search

definition 1



definition 2

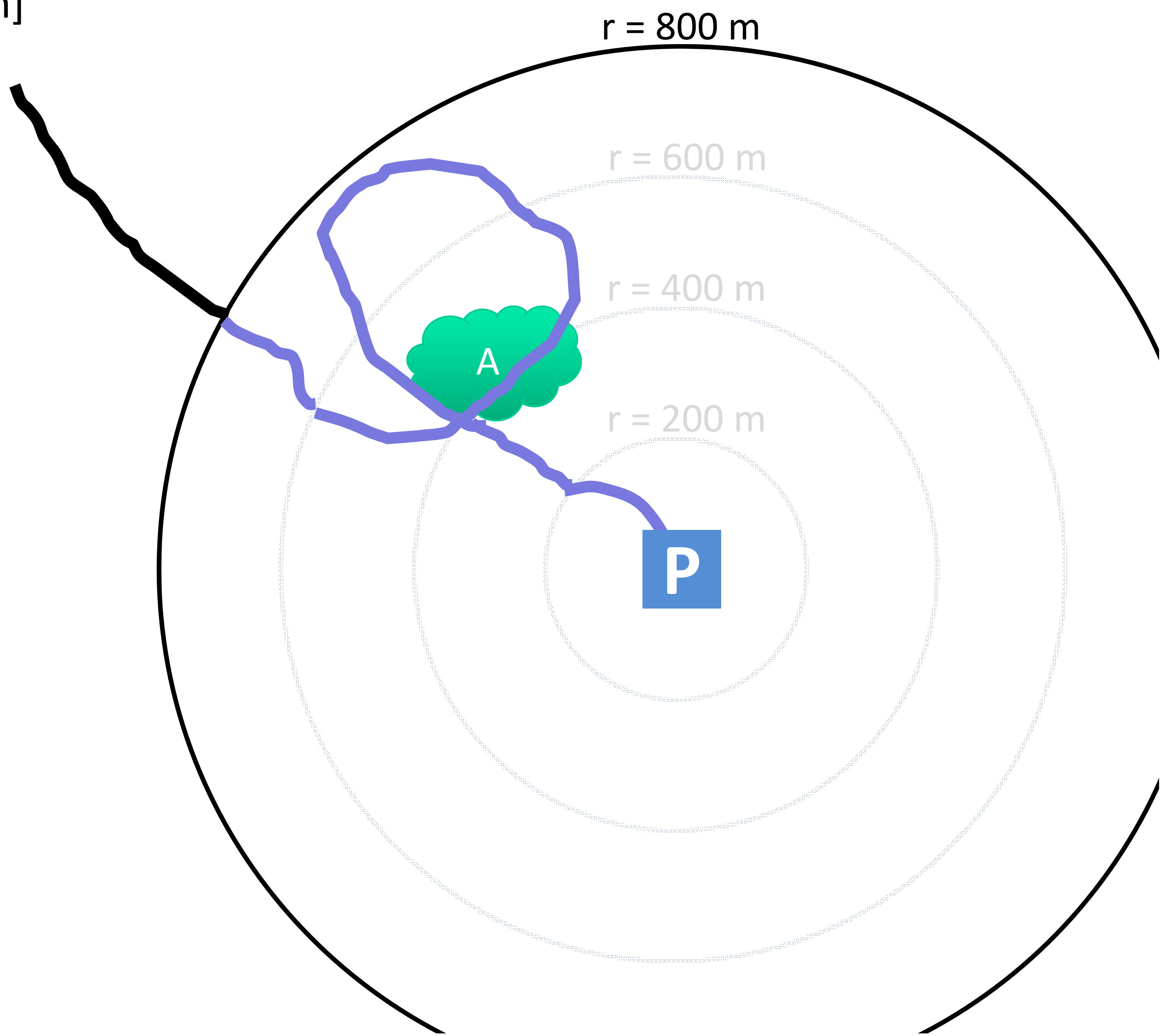


problem: search path does not have to cross the activity.

problem: $r_{acceptance}$ is individual and situation dependent.

Car Stage Characteristics

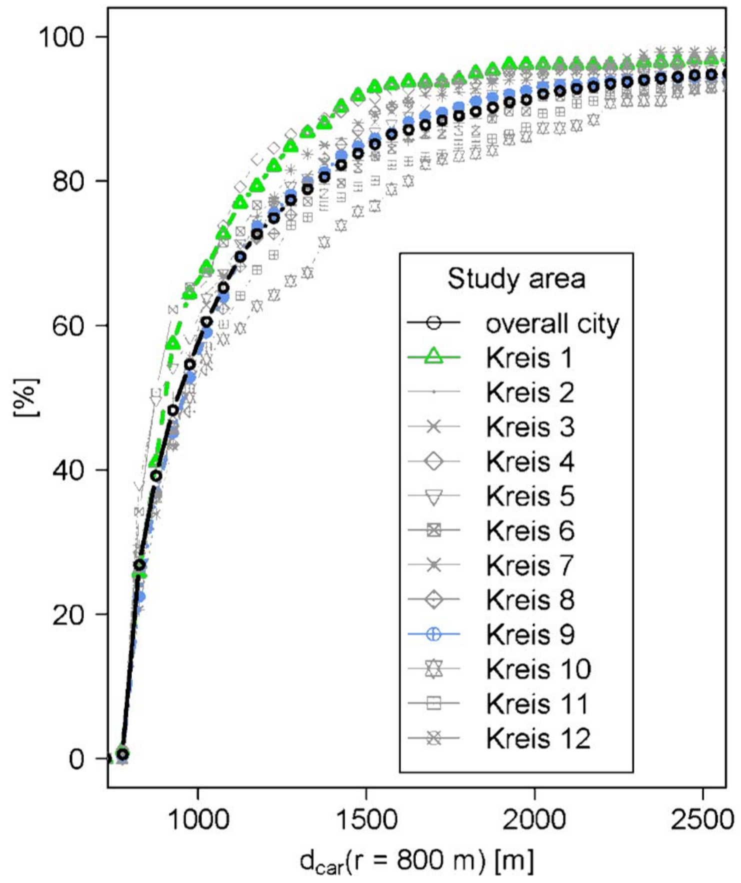
car stages > 10 [min]

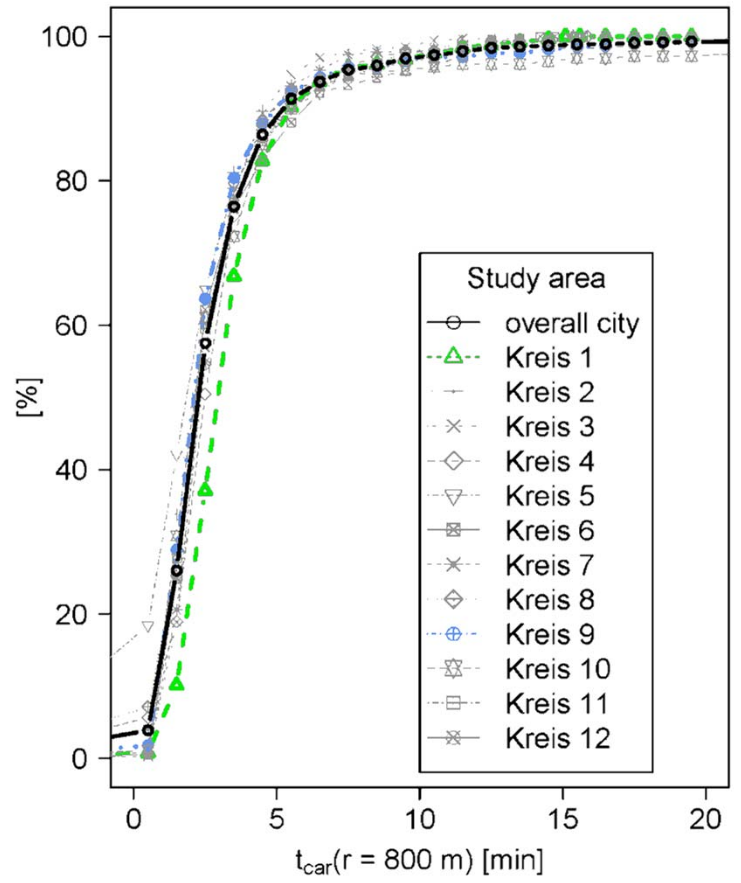


d_{car} (r=800m)

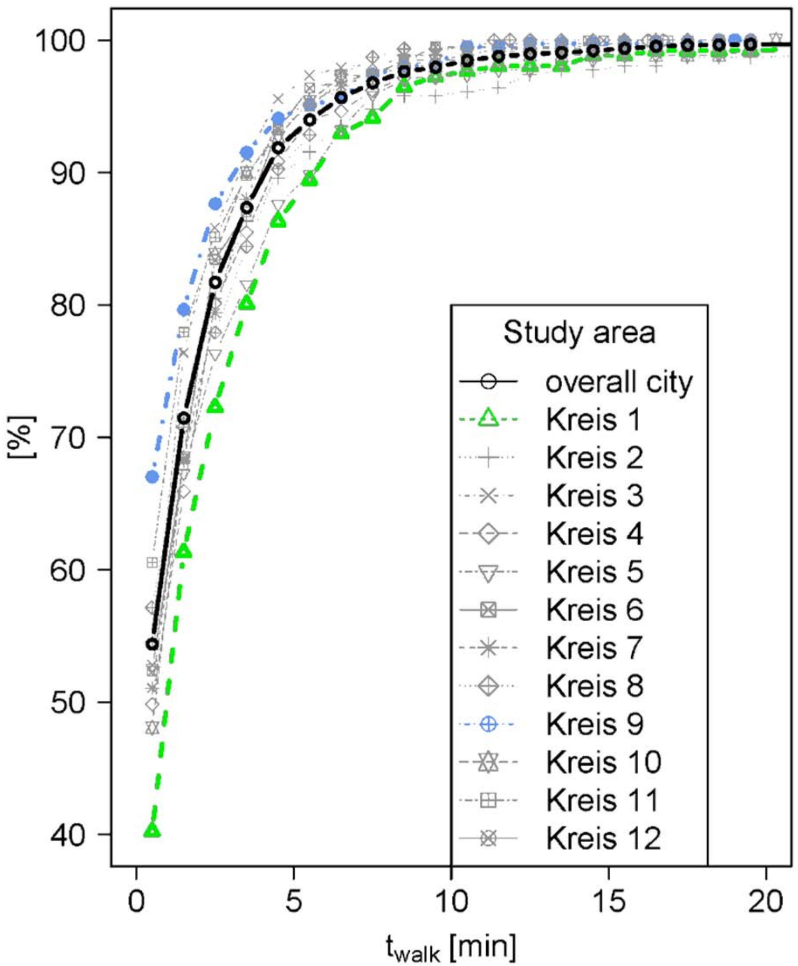
t_{car} (r=800m)

Distance and Time Within 800 m Radius Around Parking





Walk Times

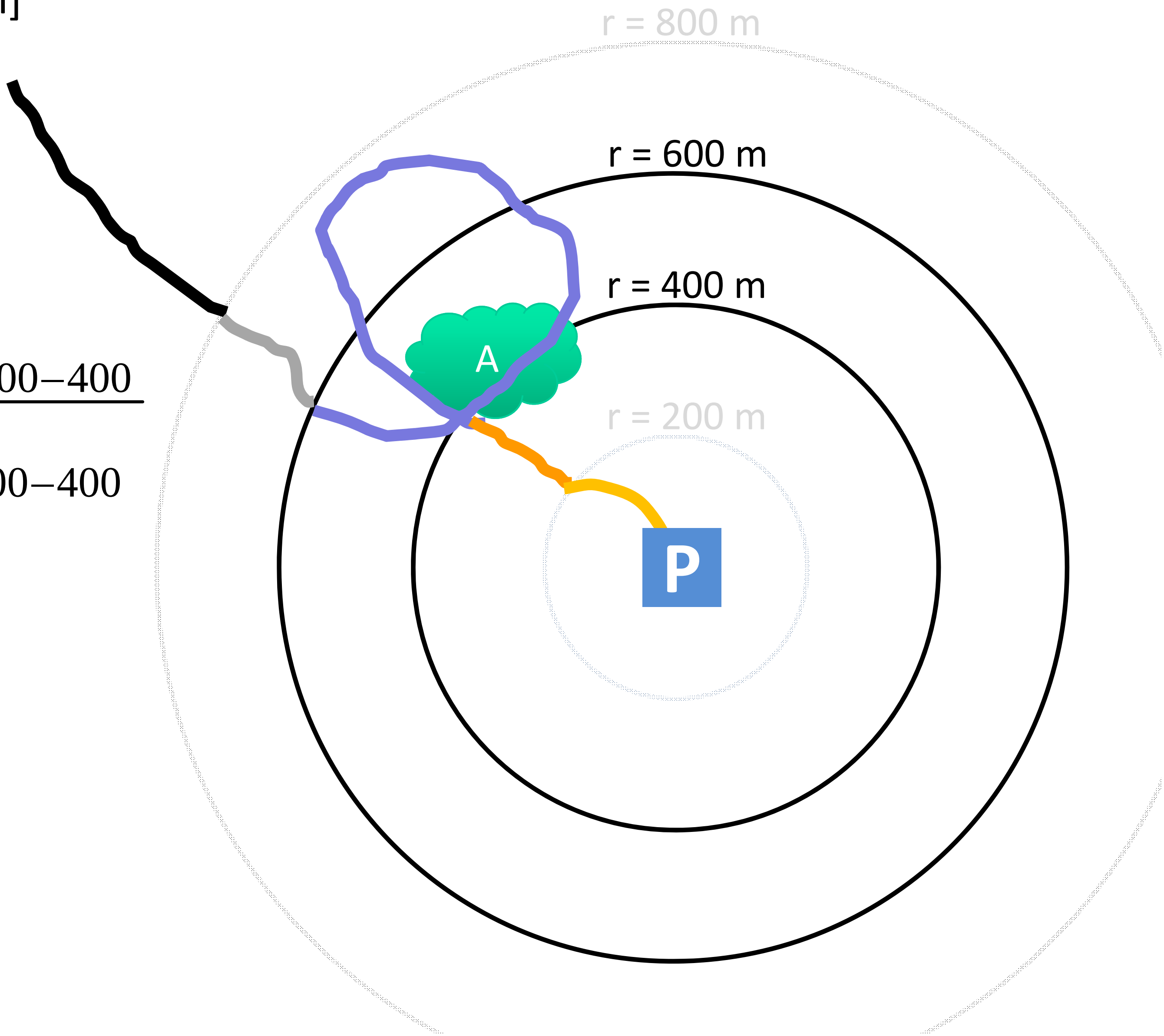


100
90
80
70
60
50
40
[%]
0
5
10
15
20
t_walk [min]
Study area
overall city
Kreis 1
Kreis 2
Kreis 3
Kreis 4
Kreis 5
Kreis 6
Kreis 7
Kreis 8
Kreis 9
Kreis 10
Kreis 11
Kreis 12

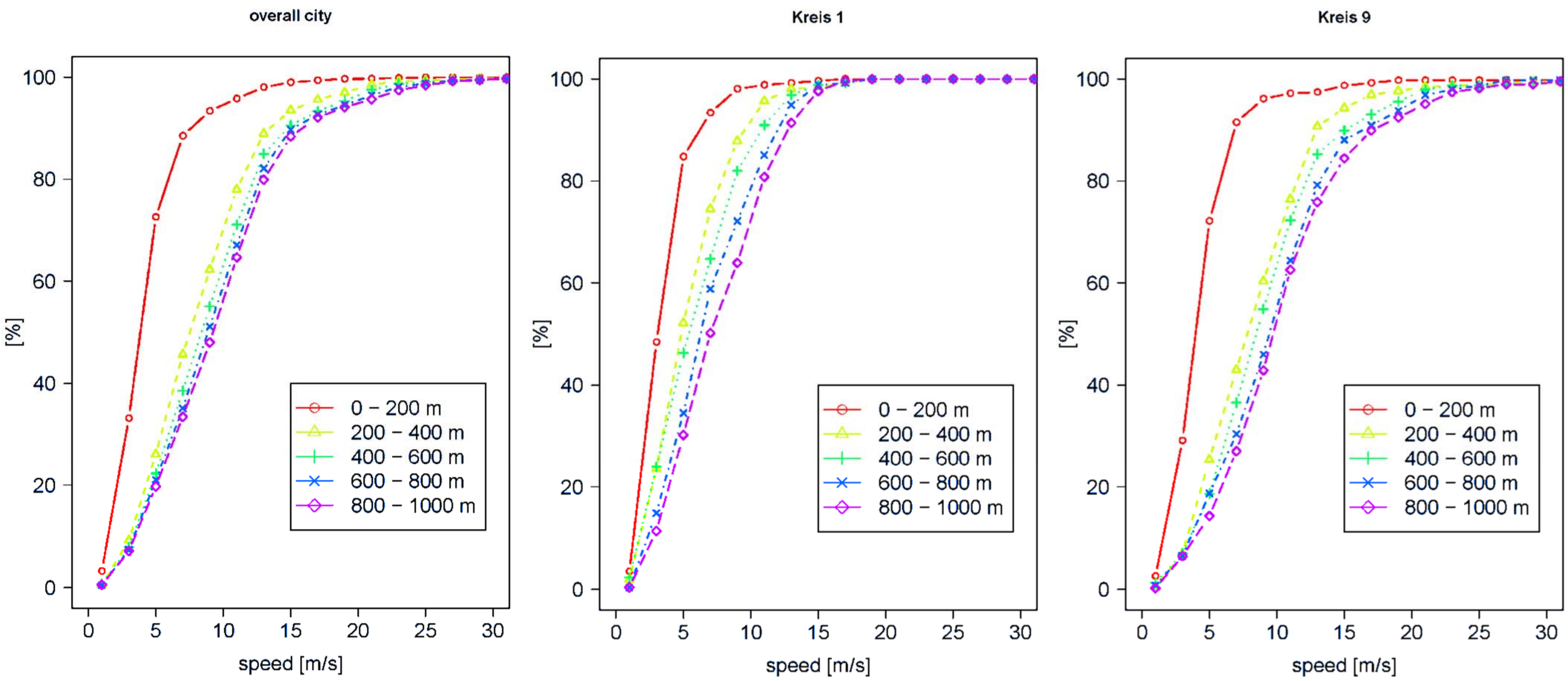
Car Stage Characteristics

car stages > 10 [min]

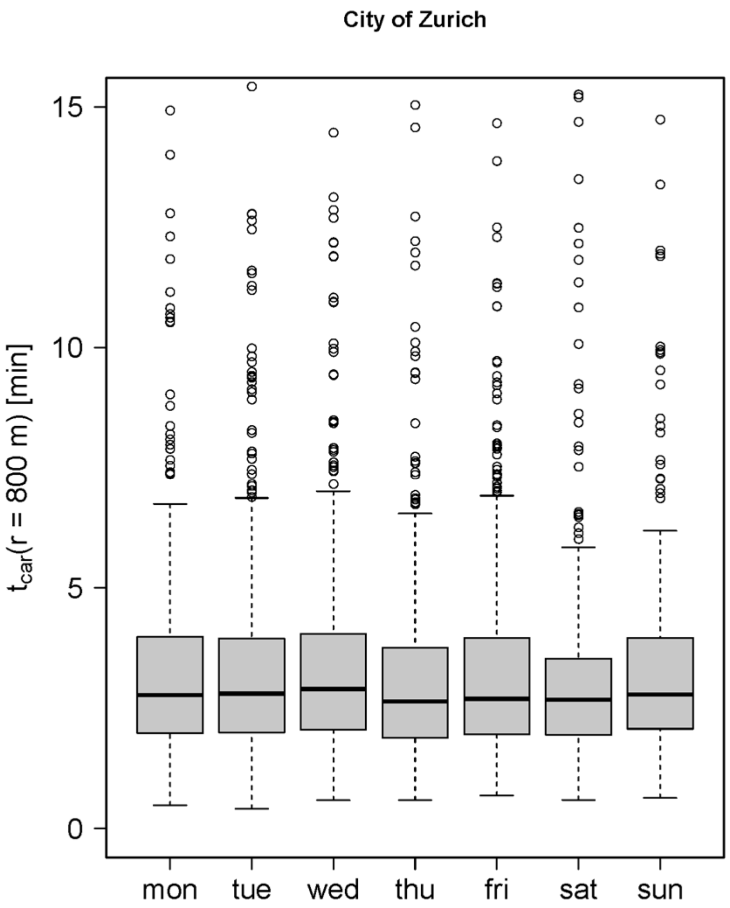
$$v_{600-400} = \frac{d_{600-400}}{t_{600-400}}$$

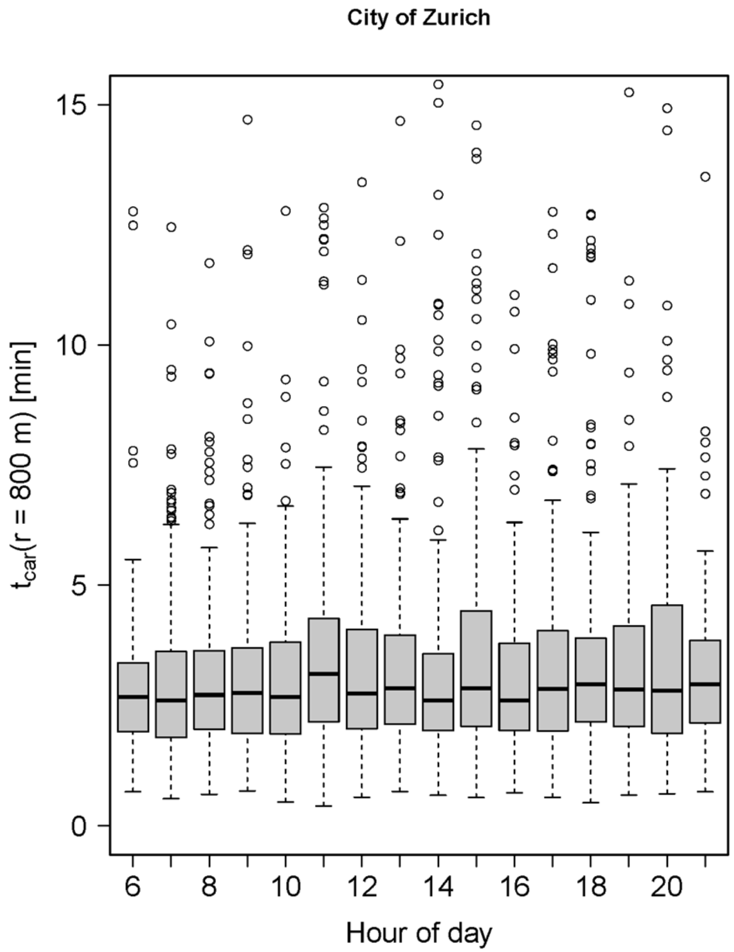


Speed Distributions



City of Zurich – Driving Times (r = 800 m)



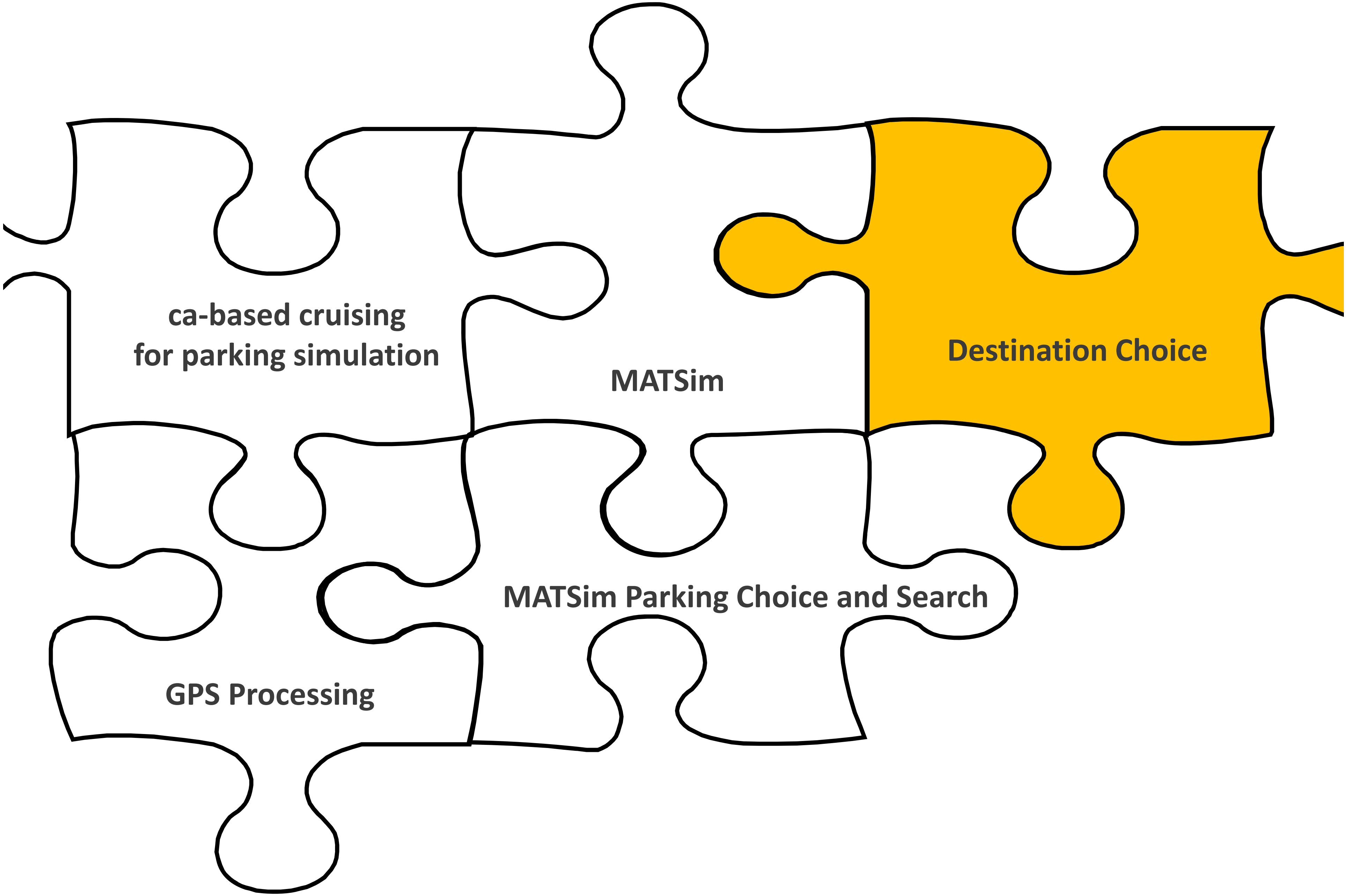


GPS Processing and Outlook

- garage vs. on-street / private vs. public
- comparison with other cities
- usage of new IVT data set

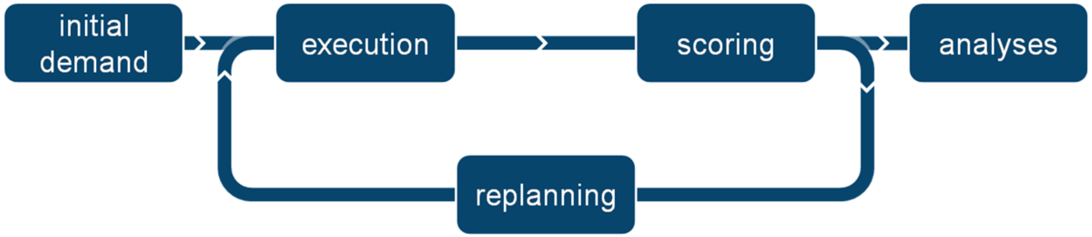
- next major step: trip purpose detection

Discussion: GPS Processing


ca-based cruising
for parking simulation
MATSim
Destination Choice
MATSim Parking Choice and Search
GPS Processing

MATSim Shopping and Leisure Destination Choice

MATSim
utility function
(absolute utilities)

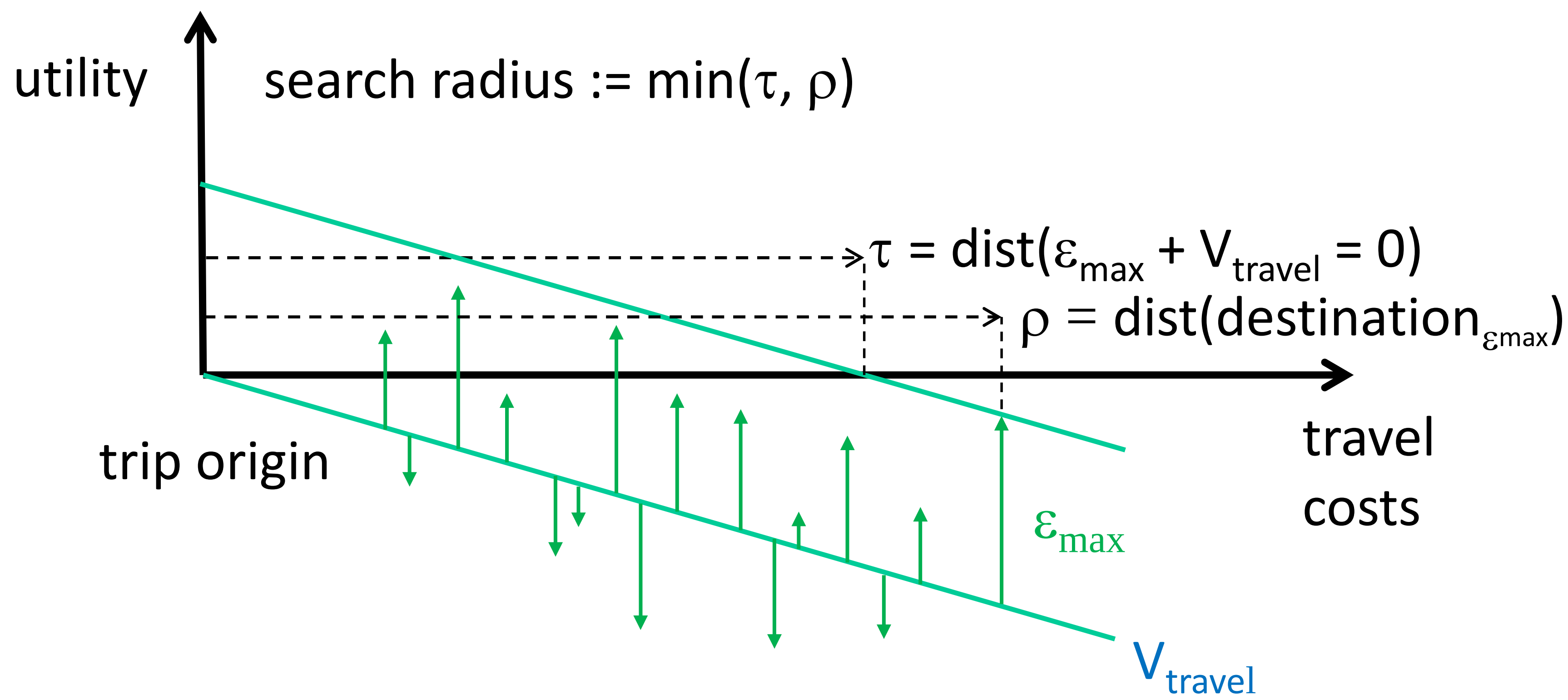


destination
choice

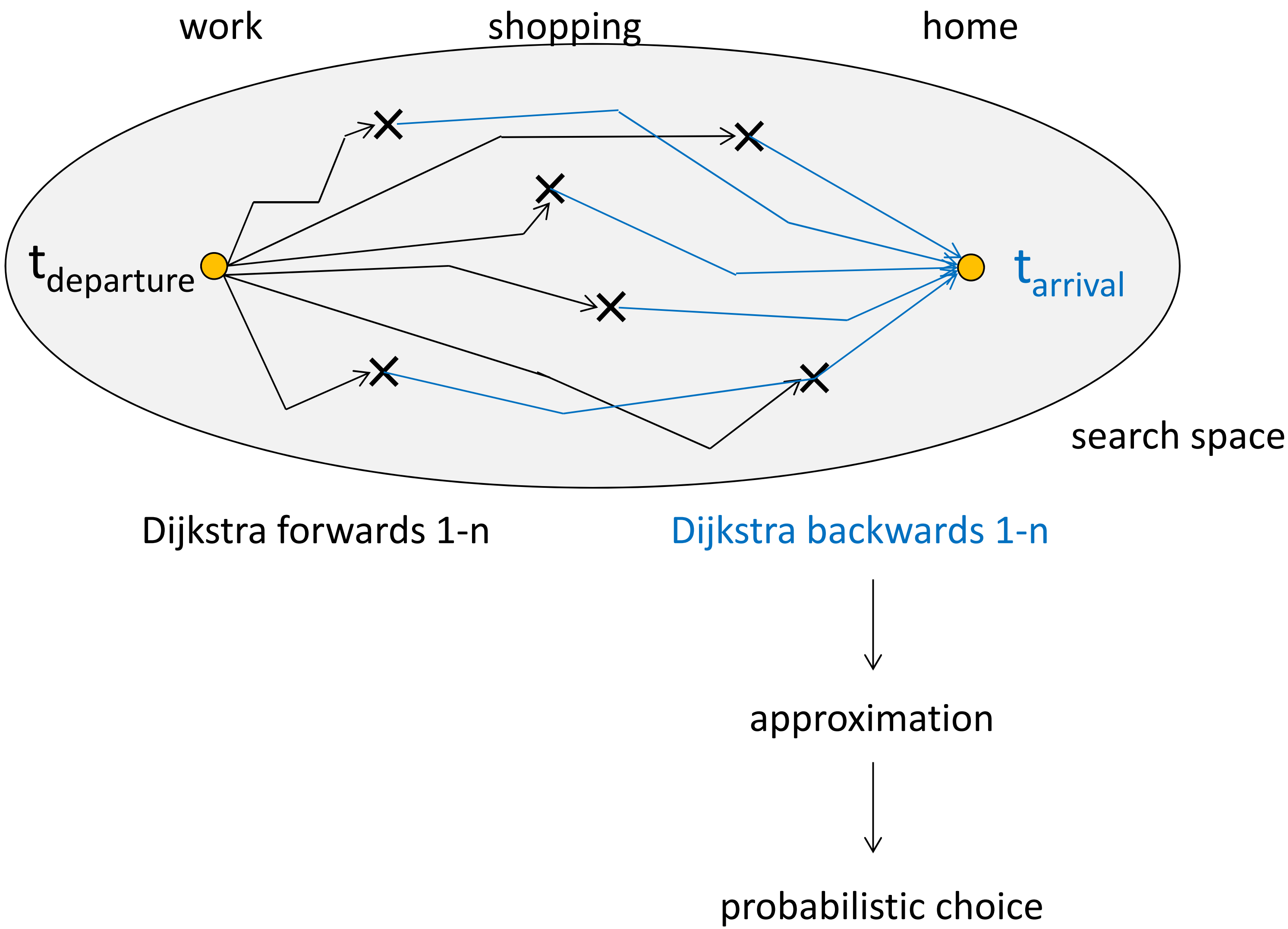
Unobserved Heterogeneity and Search Space

MATSim utility function

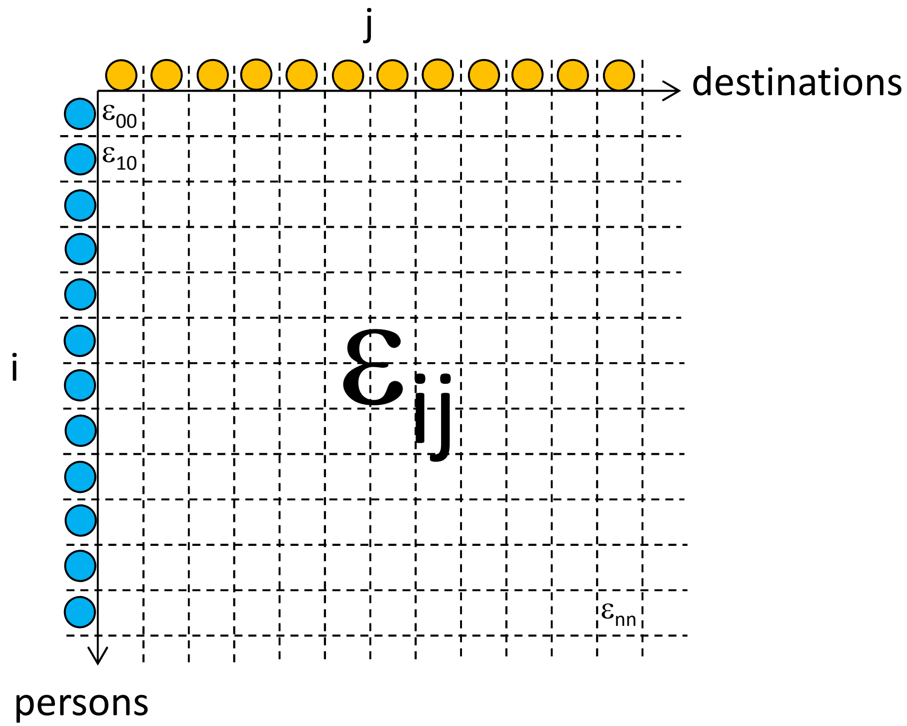
$$F = \underbrace{\sum_{i=1}^{n} U_{act,i}(type_i, start_i, dur_i) + \sum U_{trav,i}(loc_{i-1}, loc_i)}_{V} \quad + \varepsilon$$



Search Space Optimum



Repeated Draws: Quenched vs. Annealed Randomness



- ~~fixed initial random seed~~
- ~~freezing the generating order of ε_{ij}~~

one additional random number can destroy «quench»

- ~~storing all ε_{ij}~~

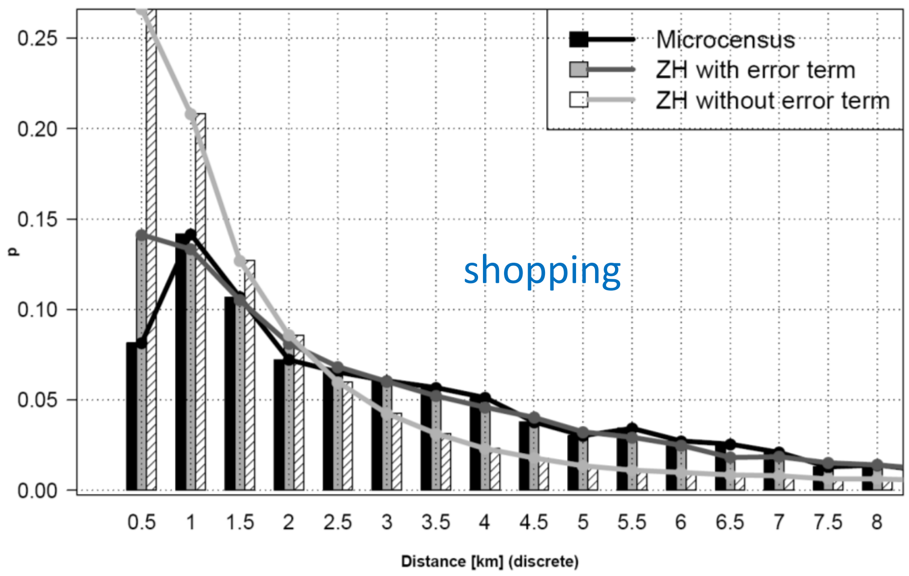
i,j ~ $O(10^6)$ -> 4×10^{12} Byte (4TByte)

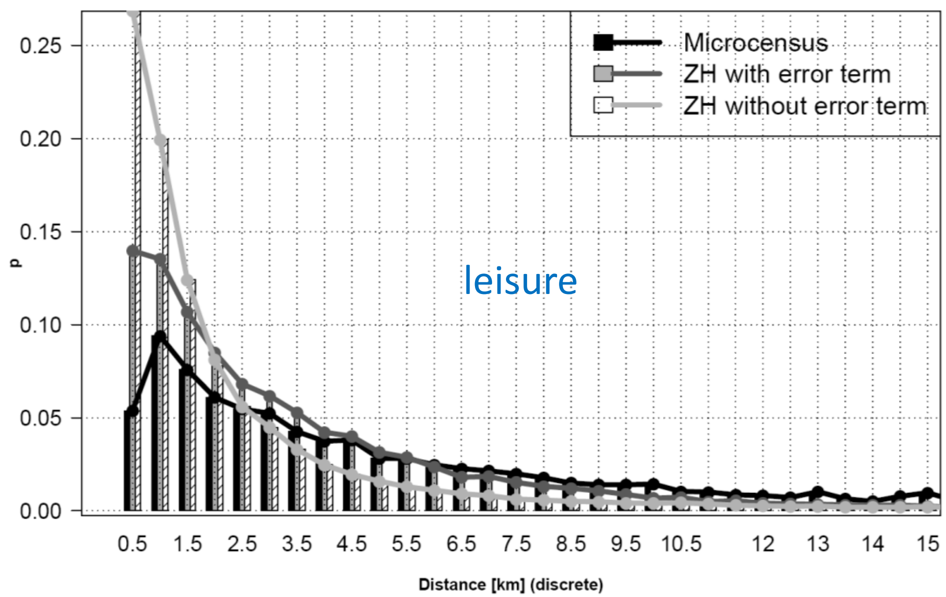
person$_i$ — store seed k_i

alternative$_j$ — store seed k_j

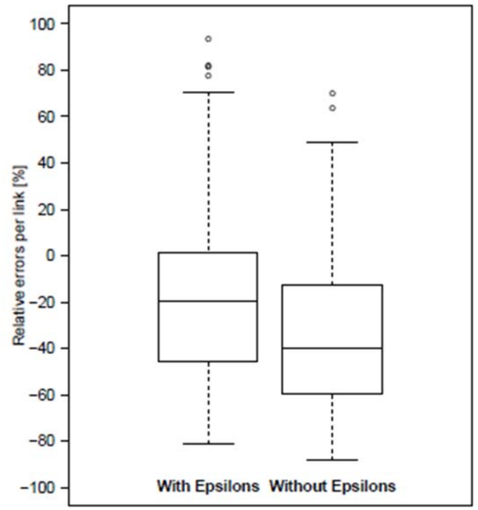
regenerate ε_{ij} on the fly with random seed $f(k_i, k_j)$

Results: Zurich Scenario



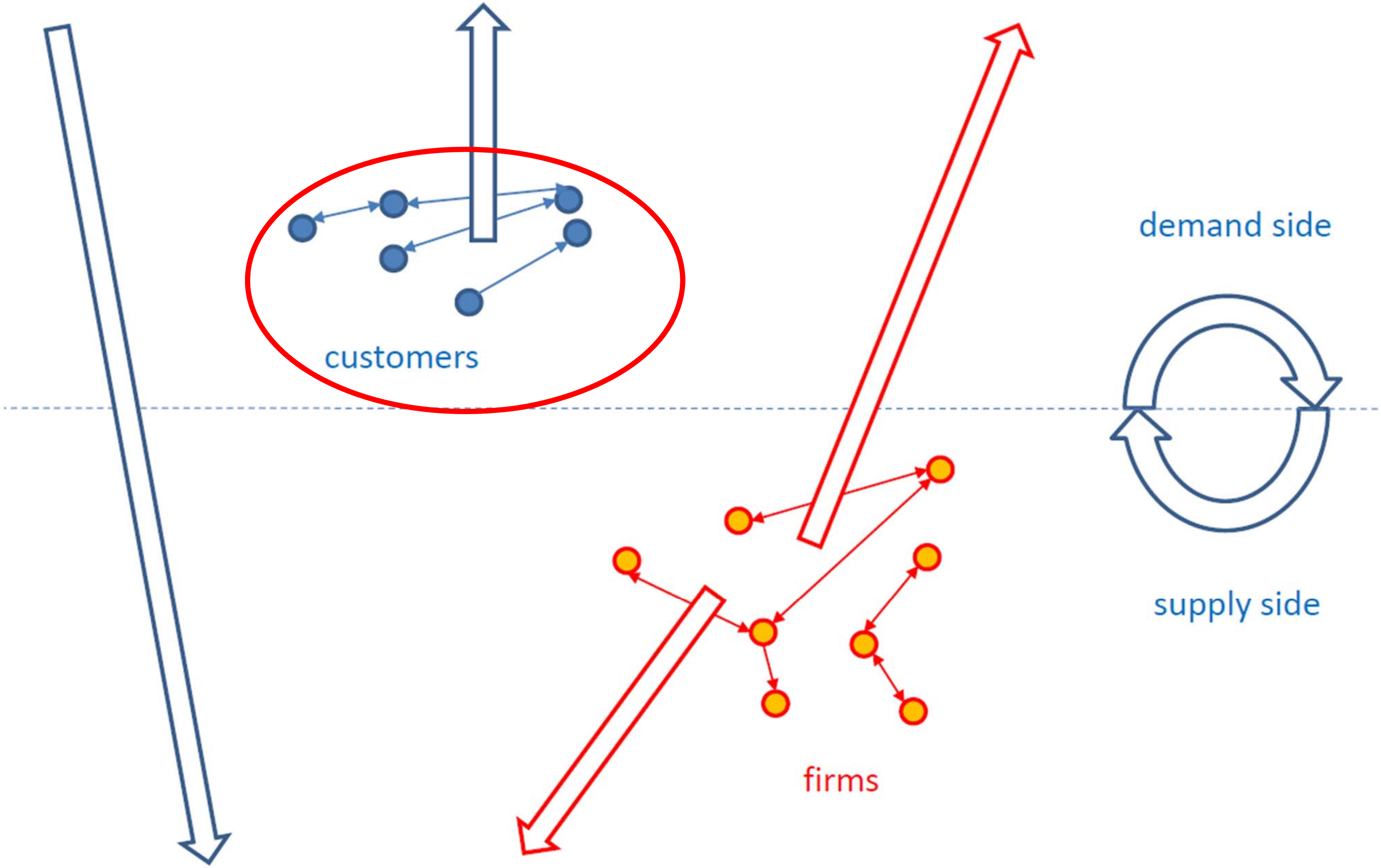


Daily traffic volumes for 123 links compared to traffic counts, iteration 100. Per link k the relative error is used, i.e., $(vol_{simulated,k} - vol_{counted,k})/vol_{counted,k}$.



Destination Choice Interaction Effects

$U_{customer}$ = f (customer interaction effects (+/-), spatial distribution of stores, …)

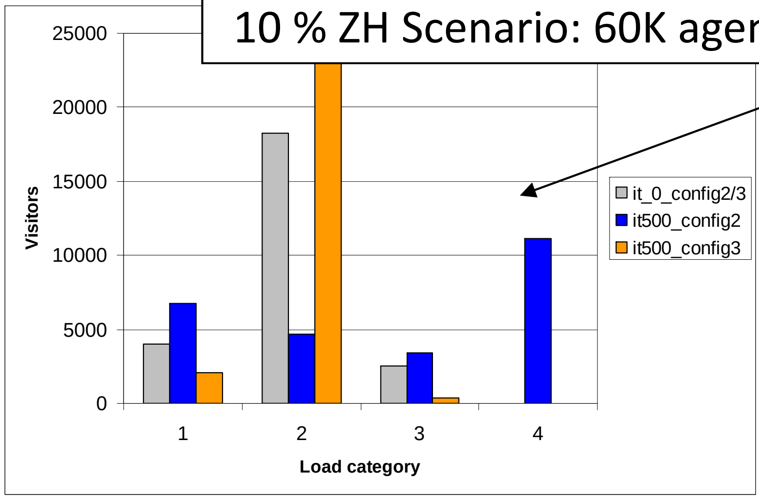


U_{firm} = f ($U_{customer}$, direct firm interaction effects (+/-))

Destination Choice Interaction Effects

- anlogon to road competition -> assignment

10 % ZH Scenario: 60K agents



reduces number of implausibly overloaded facilities

Load category
1: 0 – 33 %
2: 33 - 66 %
3: 66 - 100 %
4: > 100%

- positive effects (e.g., night life, sports fans, …)
- ongoing Master thesis -> mainly studies in marketing science

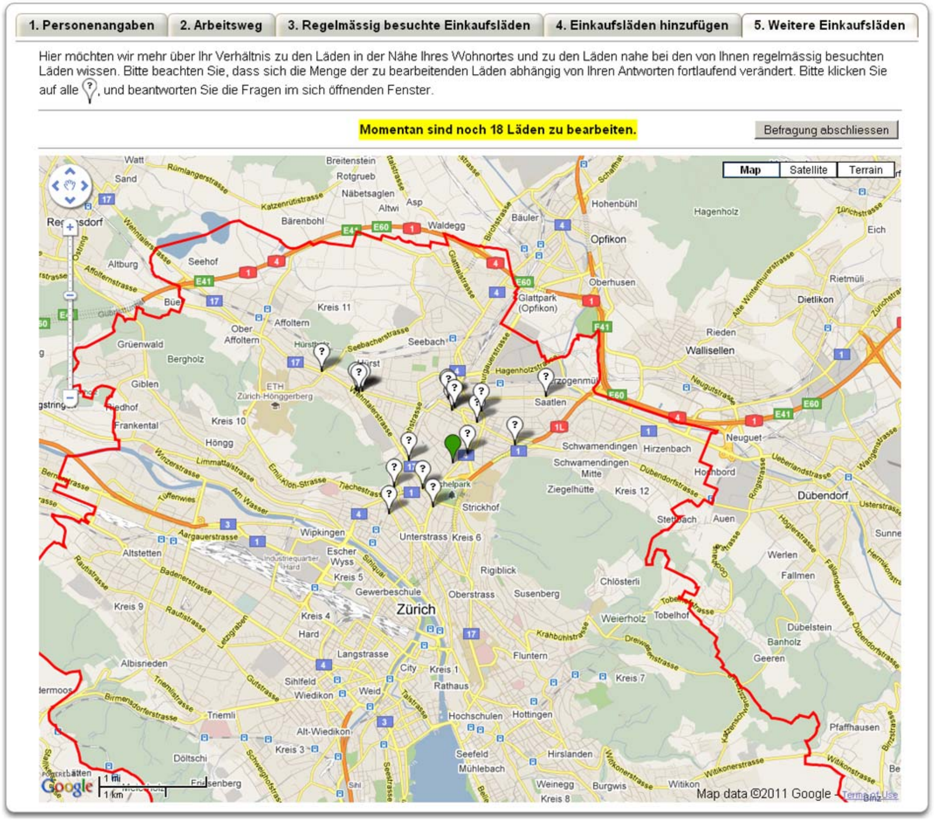
- MATSim utility function

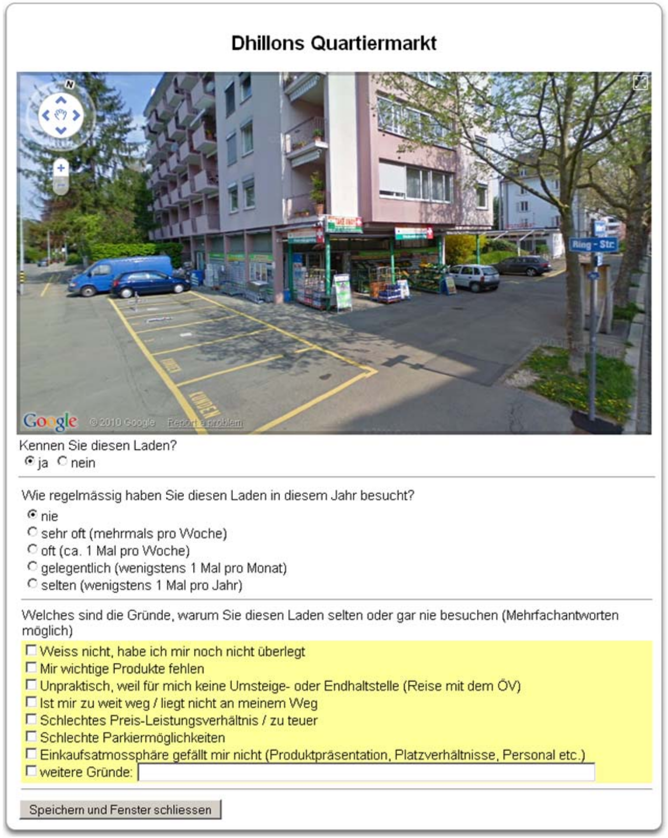
$$F = \underbrace{\sum_{i=1}^{n} U_{act,i}(type_i, start_i, dur_i) + \sum U_{trav,i}(loc_{i-1}, loc_i)}_{V} \quad + \varepsilon$$

no agglomeration terms and ε iid

Destination Choice: Outlook

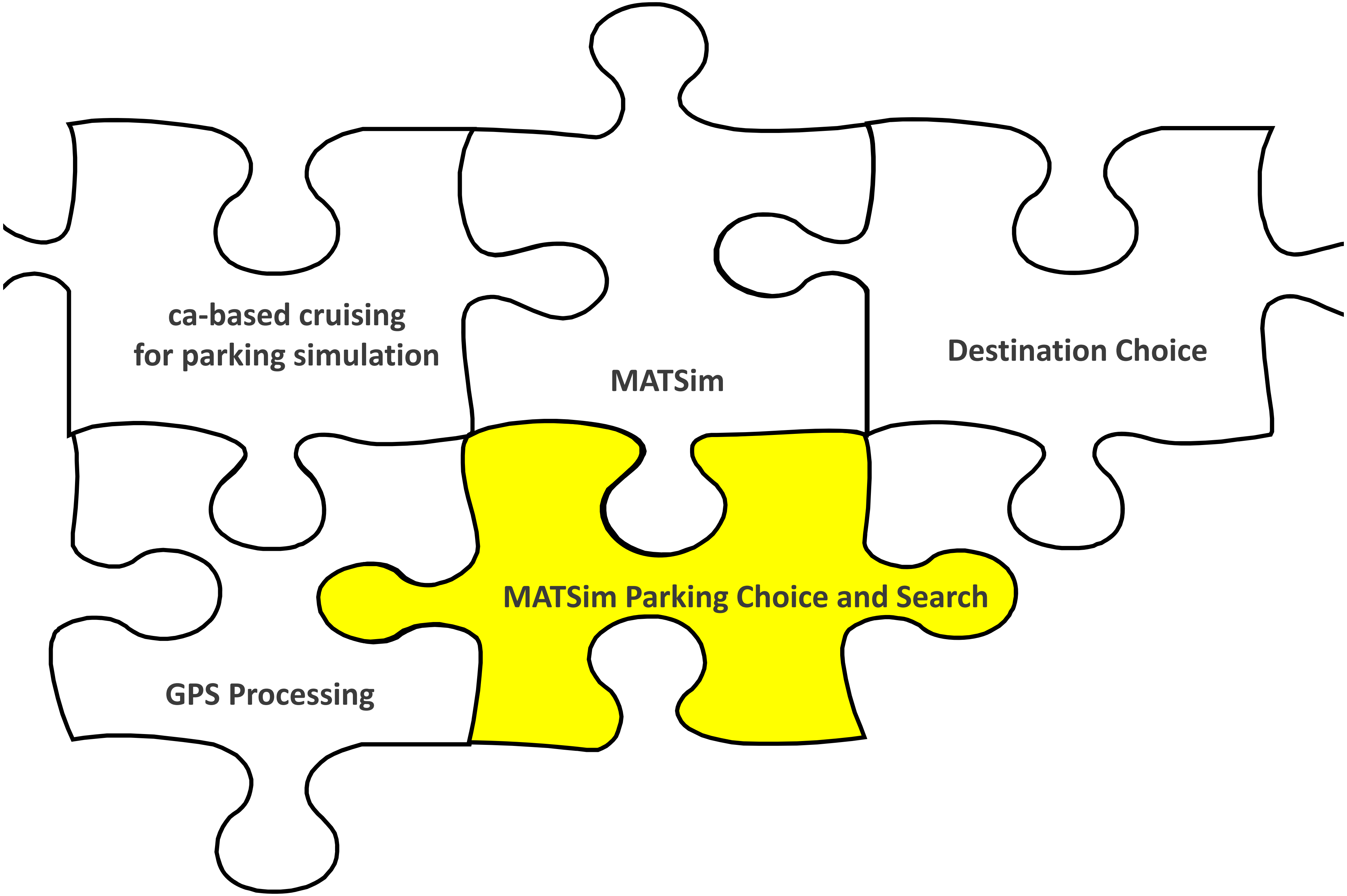
- UTF and search space (i.e., choice set) estimation based on survey





- further interaction effects, apart from parking (master thesis)
- supply side interactions (retailers, F. Ciari)

Discussion: Destination Choice Interaction Effects


ca-based cruising
for parking simulation
MATSim
Destination Choice
MATSim Parking Choice and Search
GPS Processing

Rashid's Parking Model Adventures in MATSim

Parking Choice + Pseudo Parking Search: STRC 2010

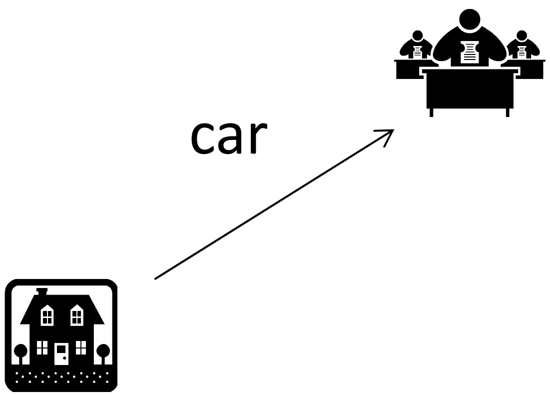
Parking Choice: Jul 2011 (for TRB)

Parking Search: IATBR 2012/ Jul 2012 (for TRB)

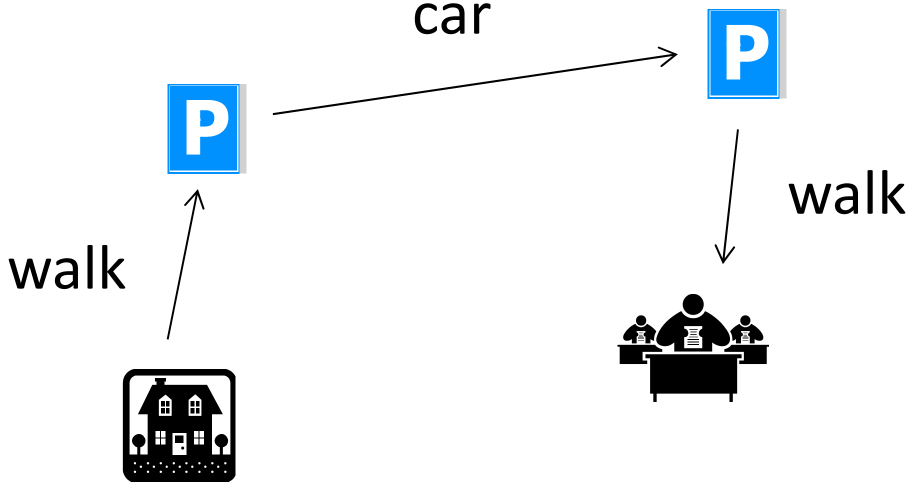
Parking Choice vs. Parking Search

Parking Choice + Pseudo Parking Search: STRC 2010

original MATSim plan



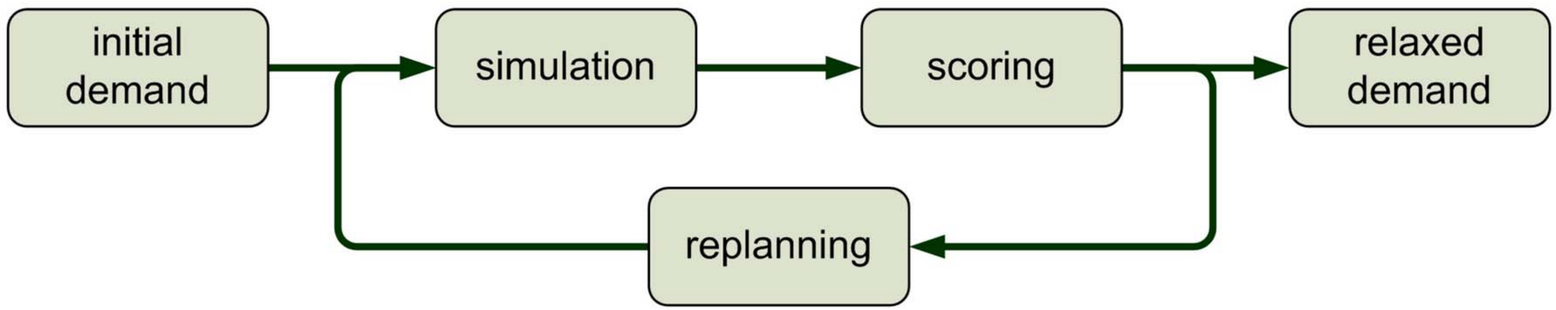
adding parking acts + walk legs



Parking Choice + Pseudo Parking Search: STRC 2010 (con't)

execute adapted plans, gather parking statistics

add parking score to overall score



allow agents to select new parking (improve parking/avoid overload)

Parking Choice + Pseudo Parking Search: STRC 2010 (con't)

Feature Summary:

- agent's plan changed
- Basically just try find parking which gives good utility (walk dist, price, etc.)
- if parking capacity violated, try assigning potentially empty parking (e.g. further away).
- incorporation of pseudo parking search paths possible

Major drawbacks:

- other strategy needs to know about parking or you have to think about all possible applications of your module and provide replanning strategies for them (not extendable).
- slow adaptation to overloads.

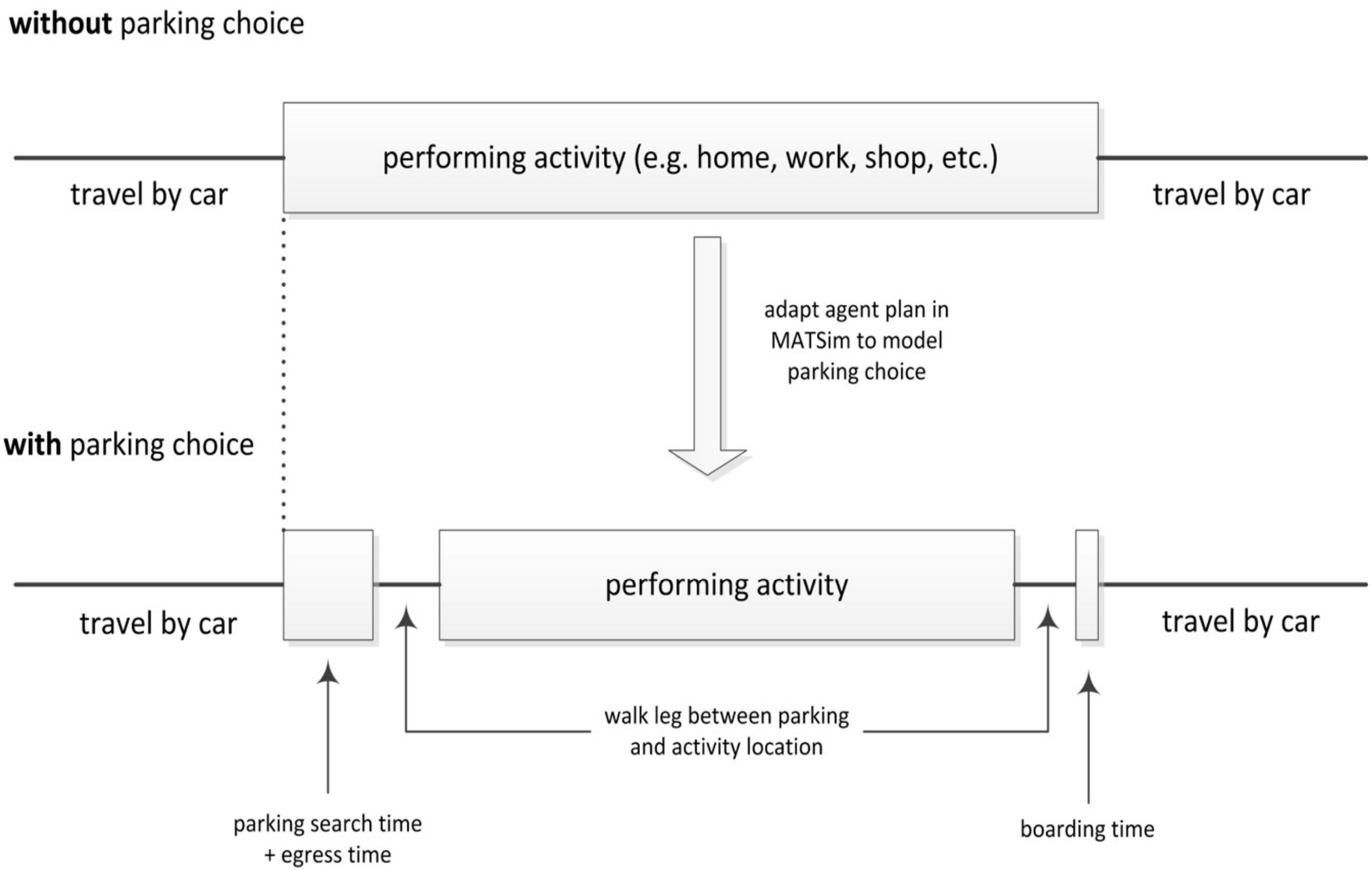
Parking Choice: Aug 2011 (for TRB)

Both major short comings solved:

- Simulate the occpancy of parking, as people arrive (no parking overload possible!)
- selection between public, private/reserved and prefered parking types
- parking selection according to best utility
- feedback to MATSim (e.g. mode change possible).
- applied on Zurich sceanrio -> helped to reduce car traffic in areas, where in reality little parking is available (solved problem within EV project).
- use post-processing to add the foot/correct car route
- good performance, independent of micro-simulation

[TODO: Experiment with Random Error Terms, try reusing work of A. Horni (2011)]

Parking Choice: Aug 2011 (for TRB) – con't



Parking Search – Starting Point

- Lot's of data collected on parking search (e.g. at IVT):
  - Axhausen (1989): Parking search strategies
  - Axhausen and Polak (1991): Decomposing parking search into components (e.g. parking search time, walking time, parking cost, etc.) and different valuation of these components
  - C. Weiss (2011): willingness to pay for search time reduction high for very short activities, higher income people willing to pay more for this, etc.
  - L. Montini (2012): «Searching for parking in GPS data»

Agent-based Parking Search

- Only a few
  - Benenson I., K.Martens, and S.Birfir (2008)
  - Spitaels, K., Maerivoet, S., De Ceuster, G., Nijs, G., Clette, V., Lannoy, P., Dieussaert, K., Aerts, K., Steenberghen, T. (2009)

- Challenges/Questions
  - How to model personal preferences, individual valuation of parking components, etc.?
  - Is just one search strategy enough?
  - How to built a system with more than one scenario in mind?

Parking Search Model Requirements

- Modelling individual preferences and individual valuation of parking components
- Allow modelling more than one parking search strategy; different parking search behaviour at different activities possible (e.g. home, shopping)
- Allow making trade-offs between parking strategies
- Also allow incorporating observed behaviour into model without trade-offs (fixed assignment of strategies to share of agents or groups)
- Parking search should have effect on longer term decisions (e.g. mode choice, location choice, etc.)
- Reserved parking, preferred parking (e.g. trade-off for EVs for parking with plug)
- Agent should respond to policy change: E.g. price, capacity change, law enforcement increase, no parking provided at work, etc.

=> Need to model a framework, which has all the necessary components to allow such simulations

Parking Search in MATSim

- Based on Within-Day replanning framework, C. Dobler (2011)
  - Interfaces for agents to act as they drive
  - Parallelization

- Don't change the plan, but only the execution (through strategy)
- Allowing implementing multiple parking strategies
- Individual preferences of agents can be incorporated in utility function, which includes terms for walking distance, search time and cost

Scenario Setup

User defines, which agent can use which strategies at which activity.
Example:

home: search for street parking (non-metered) - as agent has residence parking card for area close to home.

work: drive to parking provided by company - no parking search needed

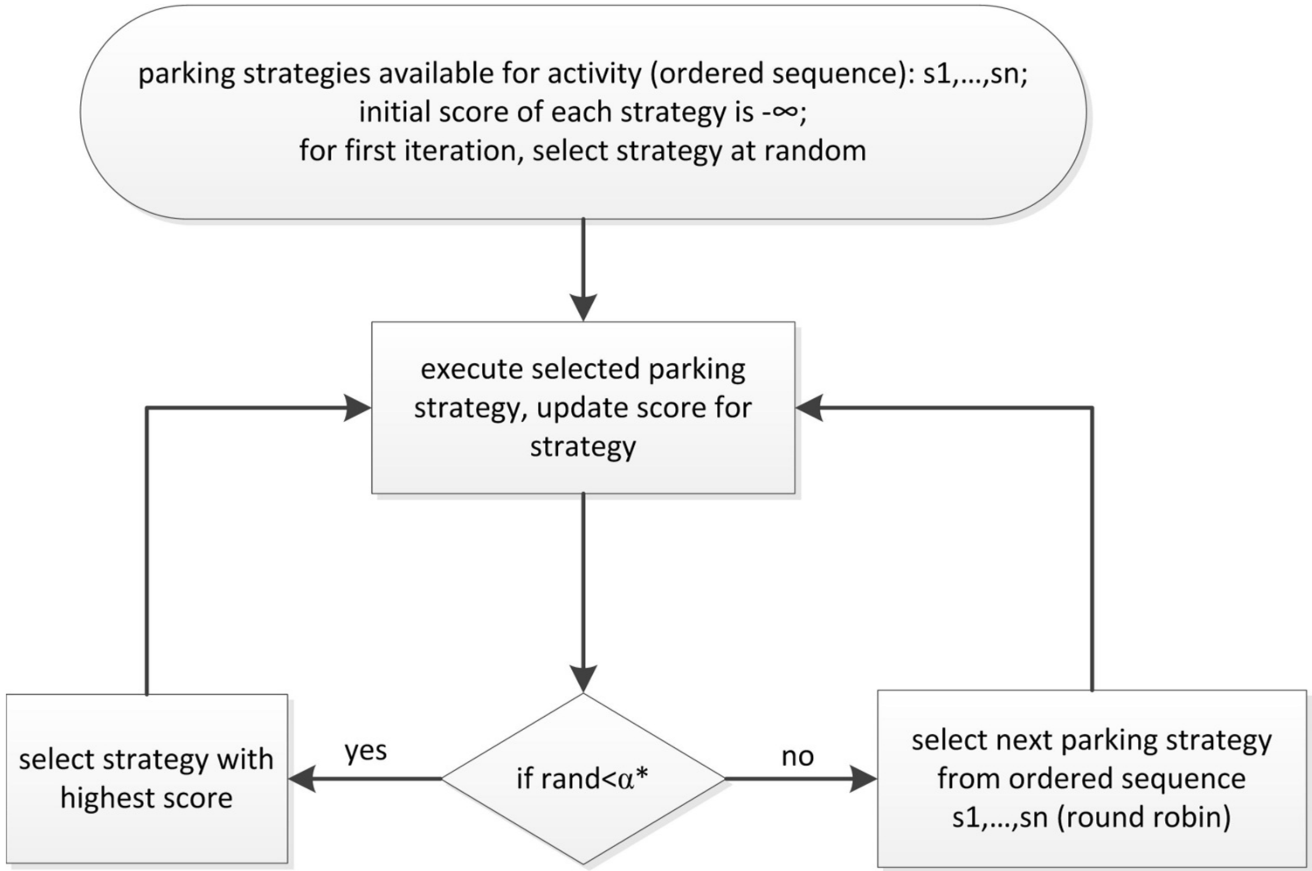
shop: 4 parking strategies available:

- search for street parking only (free + metered)
- garage parking
- combined street parking search with last resort garage parking
- illegal parking

Evaluating parking strategies

- make sure, that you don’t have too many strategy changes after each iteration, so that system can «relax» (similar condition to MATSim replanning).
- make sure, that if we stop the simulation at an iteration, with high probability a parking strategy is executed which is optimal (based on experience from strategies evaluated till that point).
- make sure, that each strategy is re-evaluated after some time, so that the strategy can be executed and scored for the «fresh/updated» environment of the agent

Evaluating parking strategies (con’t)



*) this value can be adjusted within limits: «too high» and «too low» values can both lead to undesireable results (unfrequent score updates resp. no focus on best strategy). Value of 0.9 is recommended based on experience from MATSim replanning.

Policy Changes (Example)

work: drive to parking provided by company - no parking search needed

New company policy: No free parking for employees

work: 3 parking strategies

- rent parking from company
- search on-street parking (high probability route set)
- search on-street parking and garage parking as last resort

(mode change, arrival time change, etc.)

Utility Function

$$U_{parking,i} = U_{P_{cost},i} + U_{P_{searchTime},i} + U_{P_{walk},i} + \epsilon_i$$

$U_{P_{cost},i}$: fixed rent/one-time cost (can depend on parking duration)/monetary fine/can also involve activity duration (willingness to pay changes)

$U_{P_{searchTime},i}$ *and* $U_{P_{walk},i}$: modelled explicitly (although there is already implicit disutility in default MATSim scoring function) => individual valuation of different search components.

parking search time variance: could be part of $U_{P_{searchTime},i}$ (stable medium search time vs. high variance in search time)

Current Work

Implementing framework/strategies:

-> Search free parking (e.g. high probability route)

-> Search both free + paid parking

-> Search street parking with last resort garage parking

-> drive to optimal garage parking (trade-off walking distance, price)

-> directly drive to optimal parking (perfect knowledge)

-> random search

-> etc.

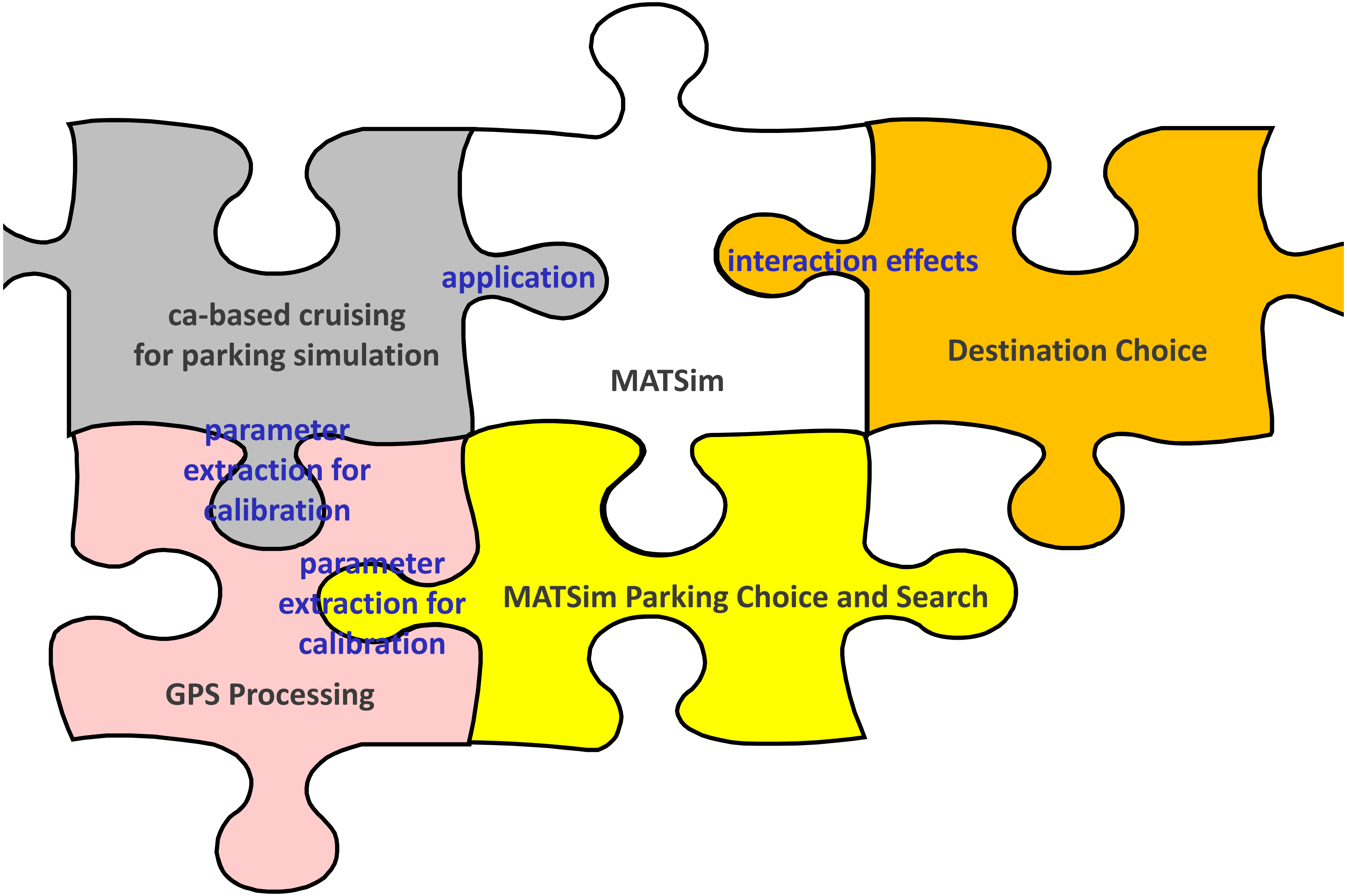
IATBR: Test cases, which demonstrate, that the search framework works

- model is sensitive to policy changes

=> TRB: Setting up Zurich Scenario

Discussion: MATSim Parking Choice and Search

- applying which concept?
- simulation costs?
- only walk trips PP-> Act
- parking studies


application
ca-based cruising
for parking simulation
MATSim
interaction effects
Destination Choice
parameter
extraction for
calibration
parameter
extraction for
calibration
MATSim Parking Choice and Search
GPS Processing