LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Praktikum Mobile und Verteilte Systeme

Location-Based Services & Route Planning & Alternative Routes

Prof. Dr. Claudia Linnhoff-Popien
André Ebert, Sebastian Feld, Thomy Phan
http://www.mobile.ifi.lmu.de

SoSe 2018

LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Praktikum Mobile und Verteilte Systeme

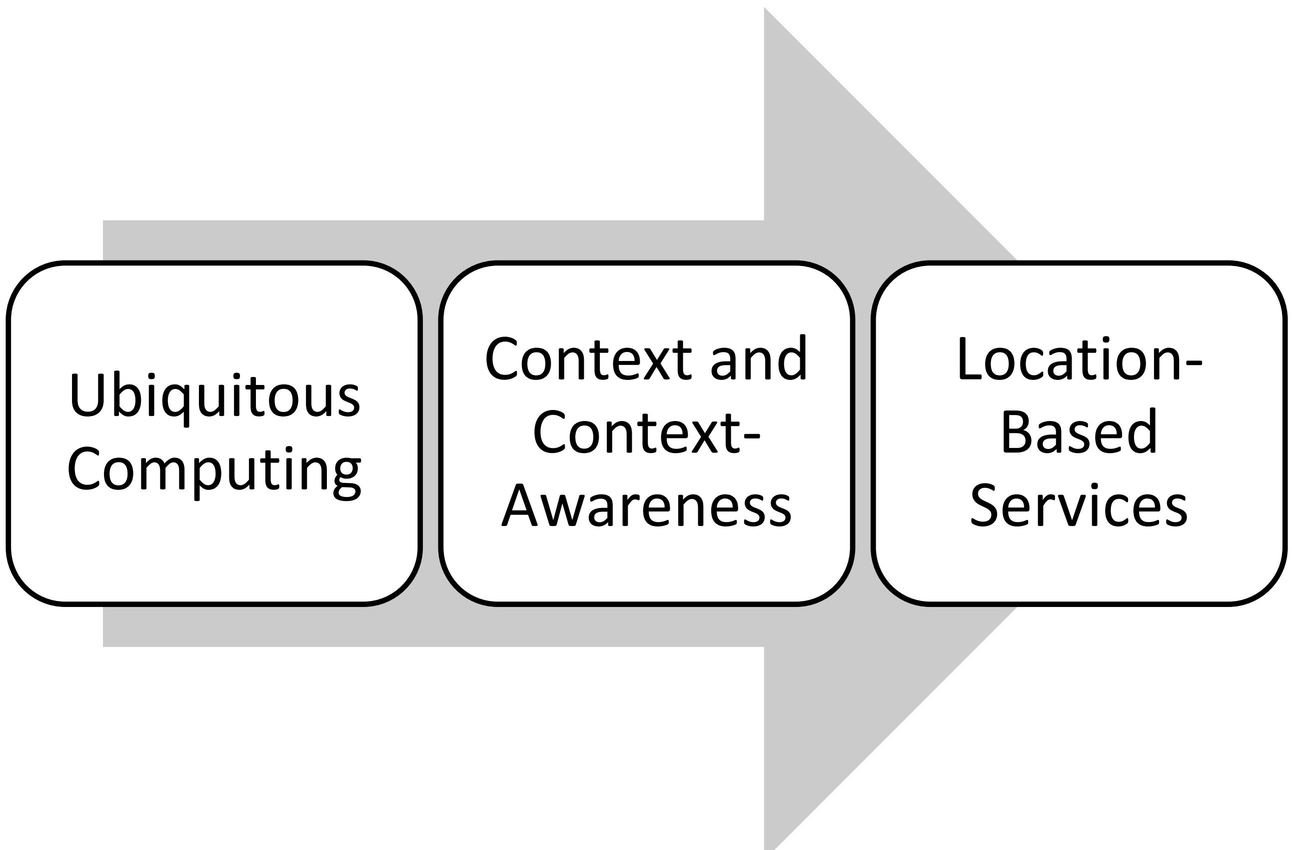
→ Location-Based Services

Prof. Dr. Claudia Linnhoff-Popien
André Ebert, Sebastian Feld, Thomy Phan
http://www.mobile.ifi.lmu.de

SoSe 2018

LOCATION-BASED SERVICES

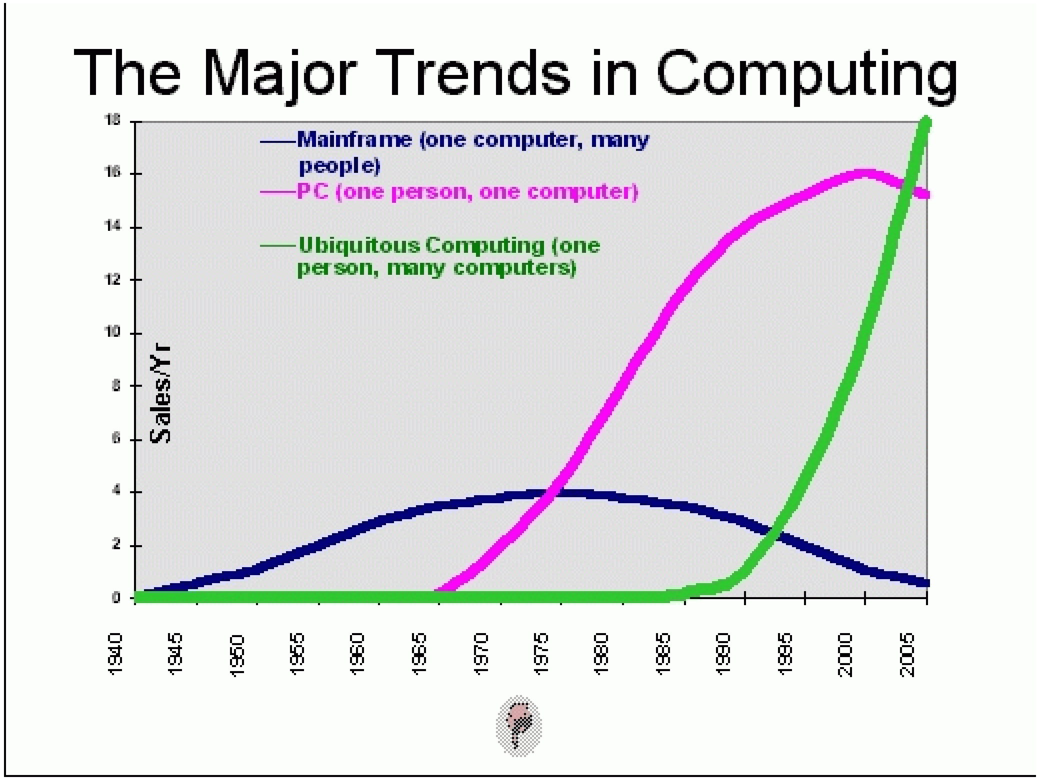
HISTORICAL OUTLINE



LOCATION-BASED SERVICES

UBIQUITOUS COMPUTING

Mark Weiser, Xerox PARC
“Nomadic Issues in Ubiquitous Computing”
Talk at Nomadic ‘96



http://www.ubiq.com/hypertext/weiser/NomadicInteractive/Sld003.htm

LOCATION-BASED SERVICES

CONTEXT & CONTEXT AWARENESS

Context is **any information** that can be used to characterize the **situation** of an **entity**. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves.

(Dey, Abowd, 1999)

Towards a Better Understanding of Context and Context-Awareness

Anind K. Dey and Gregory D. Abowd

Graphics, Visualization and Usability Center and College of Computing,
Georgia Institute of Technology, Atlanta, GA, USA 30332-0280
{anind, abowd}@cc.gatech.edu

ftp://ftp.cc.gatech.edu/pub/gvu/tr/1999/99-22.pdf

LOCATION-BASED SERVICES

CONTEXT & CONTEXT AWARENESS

Context-aware computing is a mobile computing paradigm in which applications can **discover and take advantage** of **contextual information** (such as **user location**, time of day, nearby people and devices, and user activity).

(Chen, Kotz, 2000)

A Survey of Context-Aware Mobile Computing Research

Guanling Chen and David Kotz
Department of Computer Science
Dartmouth College

Dartmouth Computer Science Technical Report TR2000-381

https://pdfs.semanticscholar.org/0c50/772e92971458402205097a67a2fd015575fd.pdf

LOCATION-BASED SERVICES

SENSING CONTEXT

Sensing location
- E.g. GPS (outdoor / indoor positioning)

Media capturing
- E.g. camera, microphone

Connectivity
- Mobile network, Bluetooth, WLAN, NFC

Time
- Day of week, calendar

Motion and environmental sensors
- Accelerometer, ambient temperature, gravity, gyroscope, light, linear acceleration, magnetic field, orientation, pressure, proximity, relative humidity, rotation vector, temperature

Further
- Active/running apps on device, remaining energy level, …

LOCATION-BASED SERVICES

DEFINITION OF LBS

Location-based Services – Fundamentals and Operation
Axel Küpper

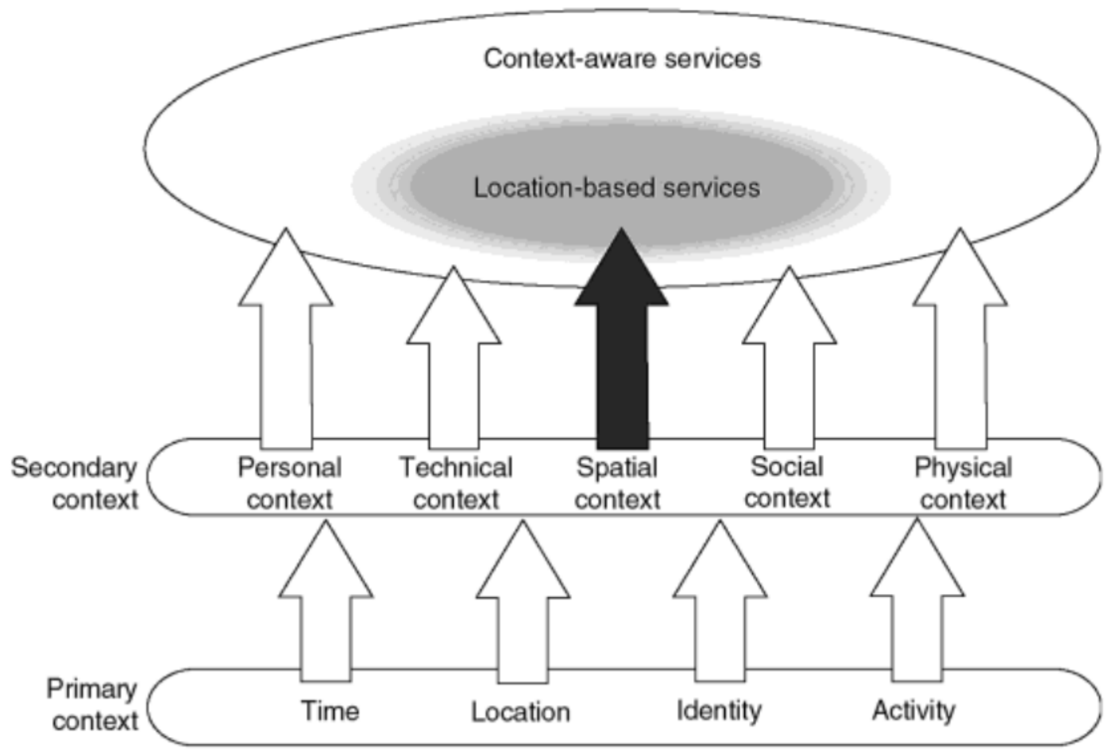


Figure 1.1 Context-aware and location-based services.

LOCATION-BASED SERVICES

CONVERGENCE OF TECHNOLOGIES

LBS as the **intersection of several technologies** (Brimicombe, 2002)

New information and communication technologies (→ "smartphones")

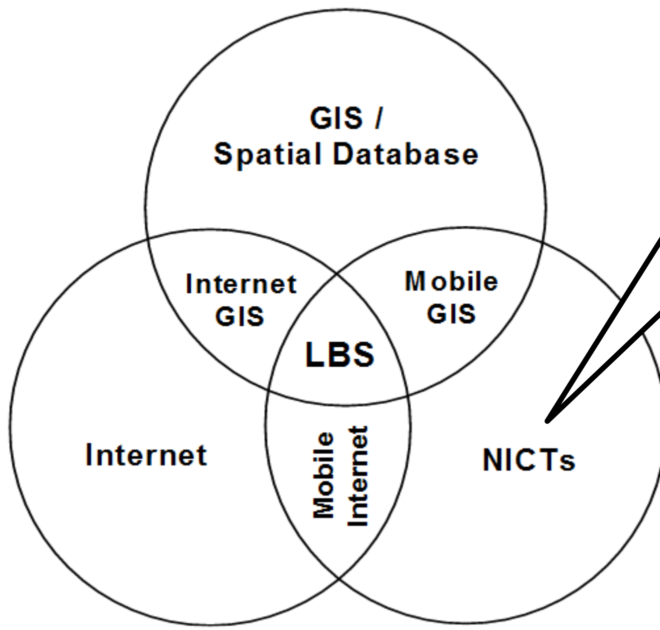


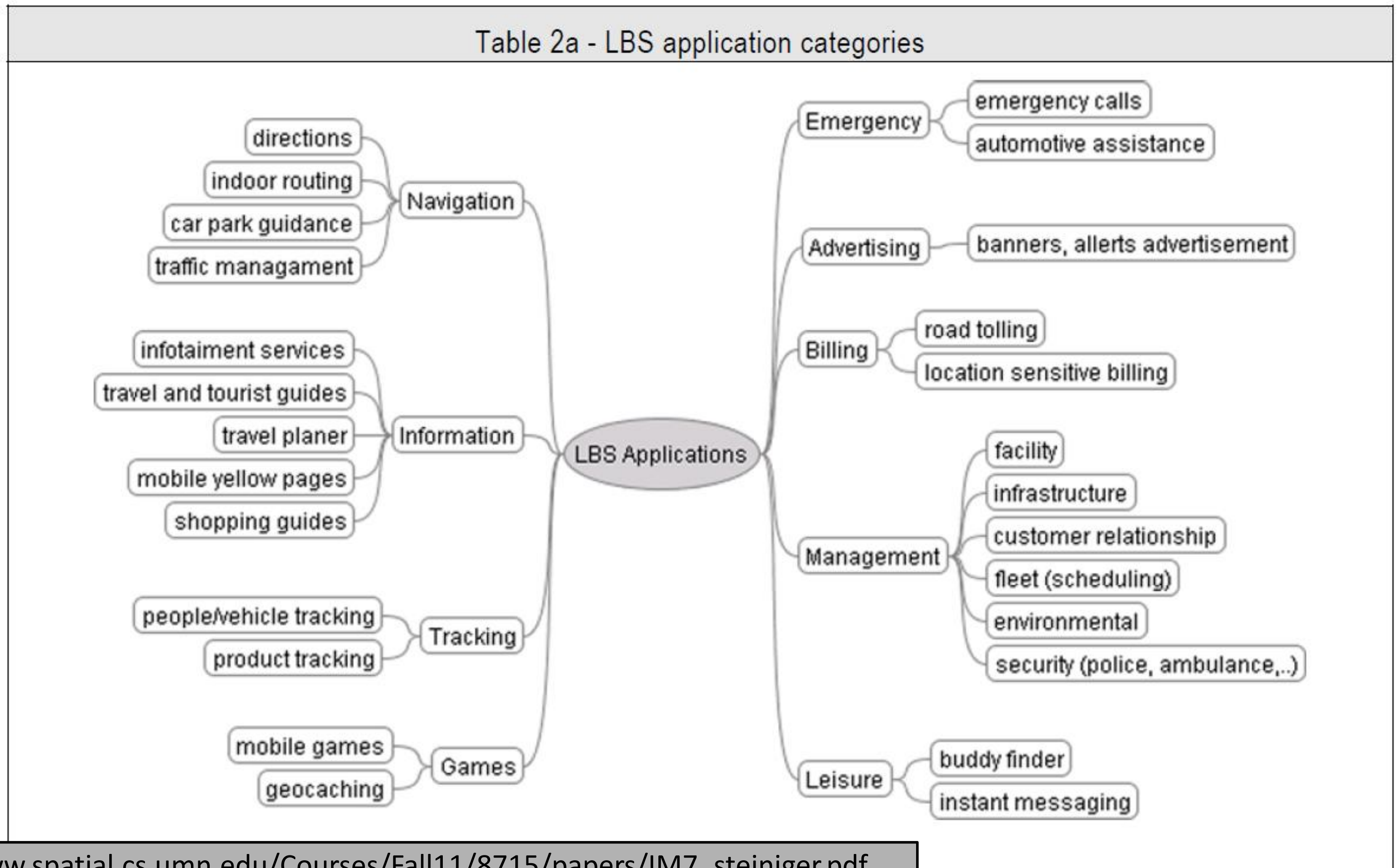
Figure 6: Convergence of technologies to create location-based services (LBS)

https://www.researchgate.net/profile/Allan_Brimicombe/publication/200621932_GIS_-_Where_are_the_frontiers_now/links/56006f3108aec948c4fa8ea3.pdf

LOCATION-BASED SERVICES

APPLICATION CATEGORIES

Foundations of Location Based Services, Lession 1, CartouCHe, Lecture Notes on LBS (Steiniger et al., 2011)



http://www.spatial.cs.umn.edu/Courses/Fall11/8715/papers/IM7_steiniger.pdf

LOCATION-BASED SERVICES

DEMARCATION

Mobile Cartography – Adaptive Visualisation of Geographic Information on Mobile Devices (Reichenbacher, 2004)

Table 7: Elementary mobile **user actions** with **spatial relation**

action		questions	objective
locating	**orientation & localisation** locating	where am I? where is {person| object}?	localise people and objects
navigating	**navigation** navigatingthrough space, planning a route	how do I get to {place name| address| xy}?	find the way to a destination
searching	**search** searching for people and objects	where is the {nearest| most relevant| &} {person| object}?	searching for people and objects meeting the search criteria
identifying	**identification** identifying and recognising persons or objects	{what| who| how much} is {here| there}?	identify people and objects; quantify objects
checking	**event check** checking for events; determining the state of objects	what happens {here| there}?	knowing what happens; knowing the state of objects

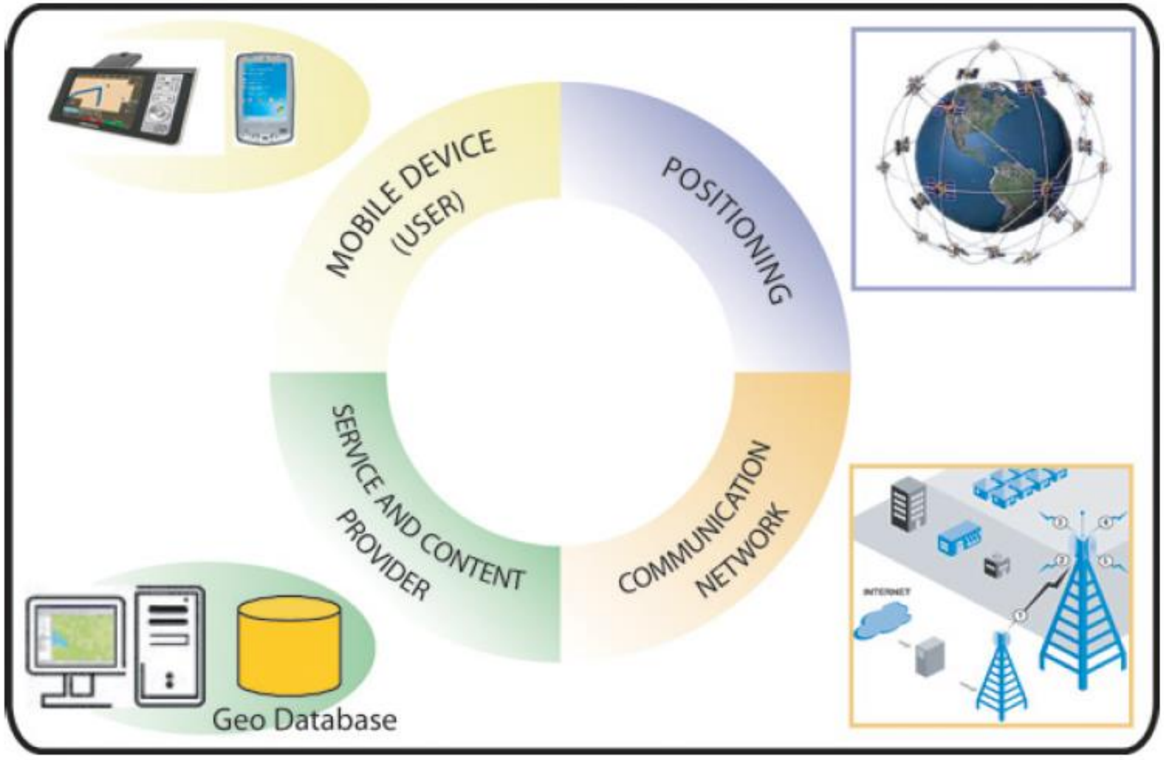
https://mediatum.ub.tum.de/doc/601066/601066.pdf

LOCATION-BASED SERVICES

DEMARCATION

Foundations of Location Based Services, Lession 1, CartouCHe, Lecture Notes on LBS (Steiniger et al., 2011)

Figure 3: The **basic components** of an LBS



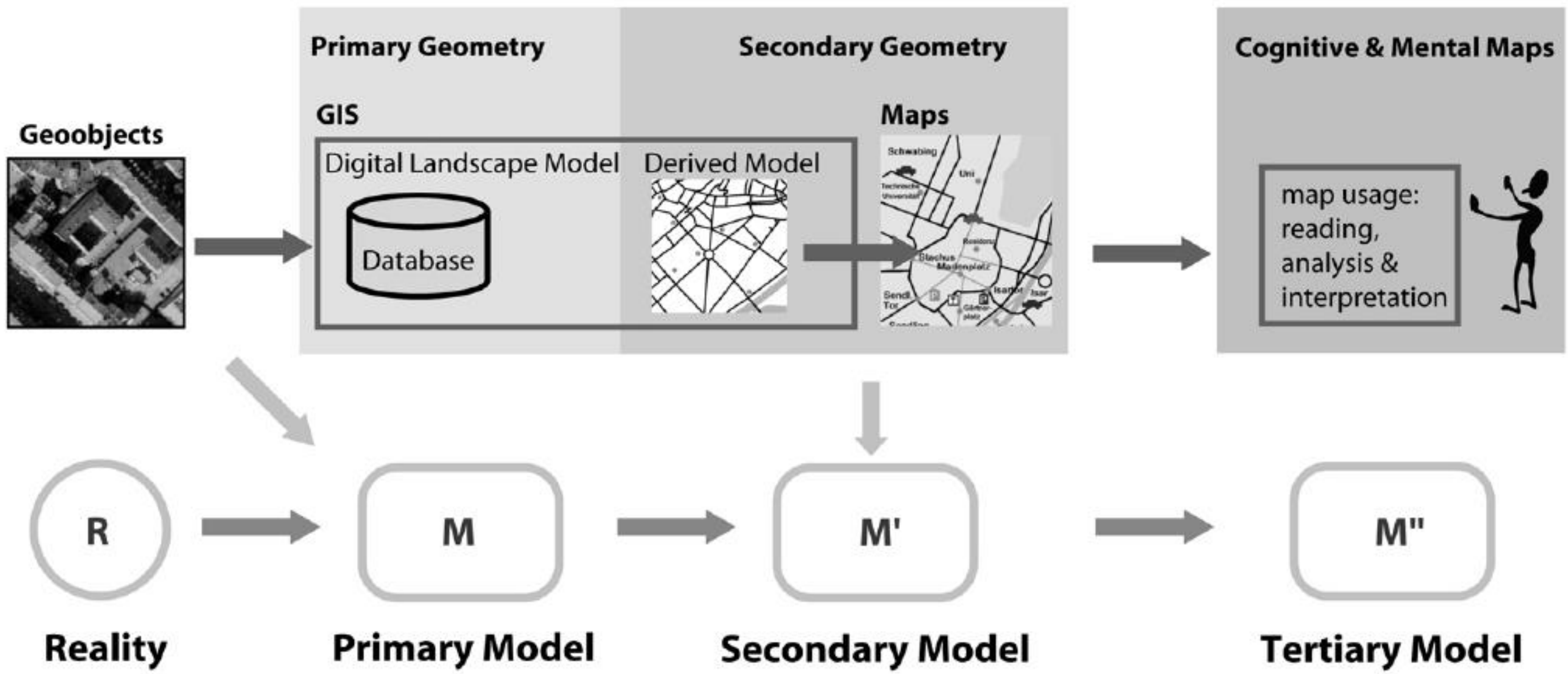
http://www.spatial.cs.umn.edu/Courses/Fall11/8715/papers/IM7_steiniger.pdf

LOCATION-BASED SERVICES

DEMARCATION

Mobile Cartography – Adaptive Visualisation of Geographic Information on Mobile Devices
(Reichenbacher, 2004)

Figure 28: Geographic information modelling



https://mediatum.ub.tum.de/doc/601066/601066.pdf

LOCATION-BASED SERVICES

CONCLUSION

Navigation and route planning as an important part of LBS

Spatial information as part of context-aware computing

Approaches and ideas to be discussed are more of **tools** rather than **applications**

Topics

- Trajectory Computing
- (Big) Data Analysis for Geospatial Trajectories
- Somewhat Information Retrieval

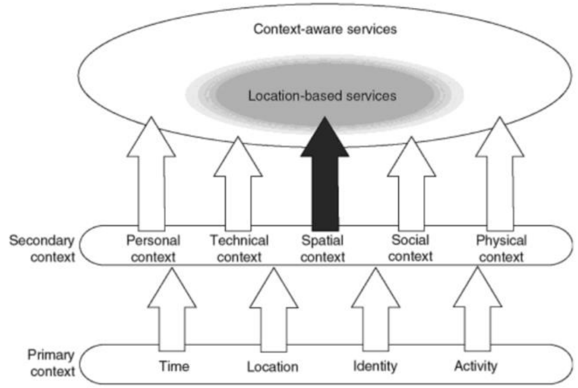
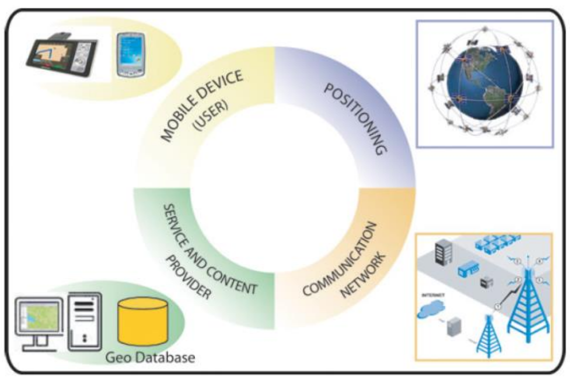


Figure 1.1 Context-aware and location-based services.



Praktikum Mobile und Verteilte Systeme

→ Route Planning

Prof. Dr. Claudia Linnhoff-Popien
André Ebert, Sebastian Feld, Thomy Phan
http://www.mobile.ifi.lmu.de

SoSe 2018

ROUTE PLANNING

SOME KIND OF REFERENCE BOOK

Route Planning in Transportation Networks

- Technical Report, Microsoft Research
- 37 pages of content, 234 references
- Ongoing updated, here: 08.01.2014

Chair for algorithms and data structures

Previously KIT, theoretic computer science

Practical computer science, data structures and efficient algorithms

Route Planning in Transportation Networks

HANNAH BAST
University of Freiburg

DANIEL DELLING
Microsoft Research

ANDREW V. GOLDBERG
Microsoft Research

MATTHIAS MÜLLER-HANNEMANN
Martin-Luther-Universität Halle-Wittenberg

THOMAS PAJOR
Microsoft Research

PETER SANDERS
Karlsruhe Institute of Technology

DOROTHEA WAGNER
Karlsruhe Institute of Technology

RENATO F. WERNECK
Microsoft Research

January 8, 2014
Technical Report
MSR-TR-2014-4

Algorithm theory / engineering

Chair for theoretic computer science

https://www.microsoft.com/en-us/research/wp-content/uploads/2014/01/MSR-TR-2014-4.pdf

ROUTE PLANNING

SOME KIND OF REFERENCE BOOK

Topics: Practical algorithms for routing in

- Road networks
- Schedule-based public transportation networks
- Multimodal scenarios (combining schedule-based and unrestricted modes)

Structure

- Shortest path algorithms for static networks
- Algorithm’s relative performance
- Journey planning on schedule-based public transportation
- Multimodal scenarios

> → Not to be taken for granted: Navigation can be seen as a shortest path problem in a graph!

SHORTEST PATHS ALGORITHMS

PRELIMINARIES

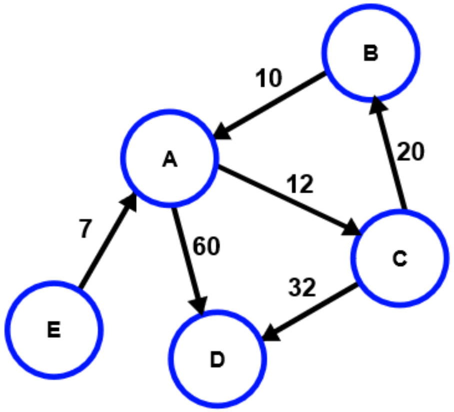
Let $G = (V, A)$ be a (directed) **graph** with a set V of **vertices** and a set A of **arcs**

Each arch $(u, v) \in A$ has an associated nonnegative **length** $l(u, v)$

The **length of a path** is the sum of its arc lengths

In the **point-to-point shortest path** problem, one is given as input the graph G, a source $s \in V$, and a target $t \in V$, and must compute the length of the shortest path from s to t in G

This is also denoted as $dist(s, t)$, the **distance** between s and t



Further problems

- One-to-all problem
- All-to-one problem
- Many-to-many problem
- All pair shortest path problem

https://de.wikipedia.org/wiki/Graph_(Graphentheorie)#/media/File:CPT-Graphs-directed-weighted-ex1.svg

SHORTEST PATHS ALGORITHMS

BASIC TECHNIQUES

https://www.microsoft.com/en-us/research/wp-content/uploads/2014/01/MSR-TR-2014-4.pdf

Dijkstra's algorithm
- Has got a "label-setting" property: Once a vertex u is scanned, its distance value $dist(s, u)$ is correct
- For point-to-point queries, the algorithm may stop as soon as it scans the target t

Bellman-Ford algorithm
- Label-correcting algorithm: vertices may be scanned multiple times
- Works in rounds and on graphs with **negative edge weights**

Floyd-Warshall algorithm
- Computes distances between all pair of vertices (**APSP**)

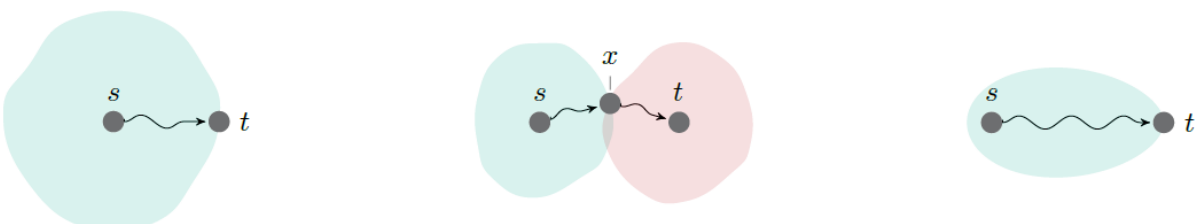


Figure 1: Schematic search spaces of Dijkstra's algorithm (left), bidirectional search (middle), and the A* algorithm (right).

Search space: The set of vertices scanned by the algorithm

SHORTEST PATHS ALGORITHMS

GOAL-DIRECTED TECHNIQUES

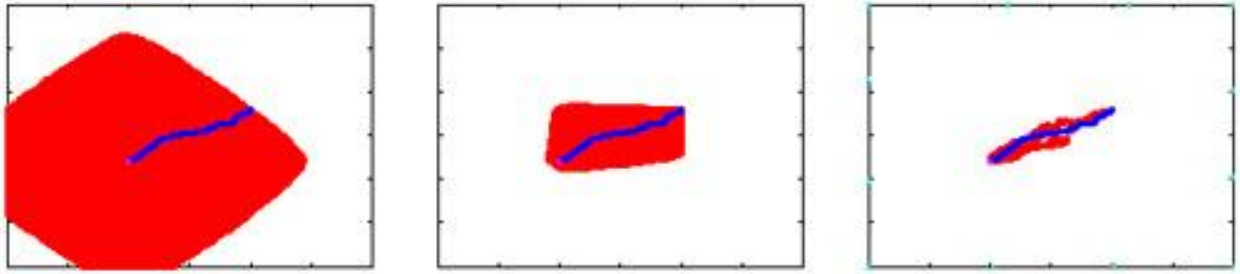
https://weekendtechnotes.files.wordpress.com/2012/11/searchboundsoptimization.jpg

A* Search

- **Potential function** on the vertices, which is a lower bound on the distance $dist(u, t)$
- Vertices that are closer to the target are scanned earlier during the algorithm
- In road networks with travel time metric, one can use the geographical distance

ALT (A*, landmarks, and triangle inequality) algorithm

- Preprocessing phase picks **small set of landmarks** and stores the distances between them and all vertices in the graph
- Triangle inequalities involving the landmarks are used to compute a valid lower bound on $dist(u, t)$



Dijkstra A* ALT

SHORTEST PATHS ALGORITHMS

FURTHER APPROACHES/TECHNIQUES

Further **Goal-Directed** Techniques

- E.g. Geometric Containers: precompute for each arc a set of vertices to which a shortest path begins with that arc

Separator-Based Techniques

- E.g. Vertex/Arc Separators: decompose graph into several components and create and overlay graph

Hierarchical Techniques

- Exploit the inherent hierarchy of road networks

Bounded-Hop Techniques

- Precompute distances between pairs of vertices, implicitly adding “virtual shortcuts” to the graph

Combinations

- Hybrid algorithms for additional speedups

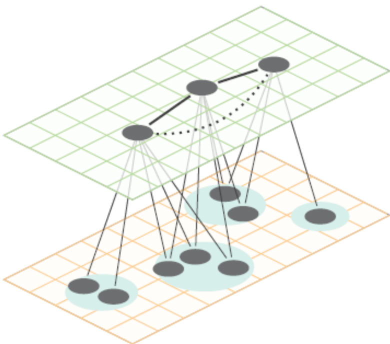


Figure 3: Multilevel overlay graph with two levels. The dots depict separator vertices in the lower (orange) and upper (green) level.

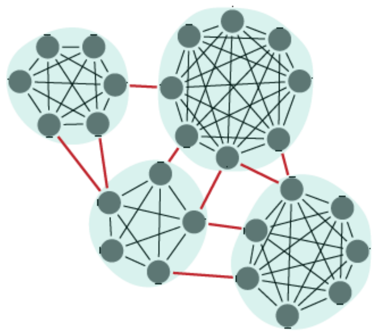


Figure 4: Overlay graph constructed from arc separators. Each cell contains a full clique between its boundary vertices, and cut arcs are thick red.

https://www.microsoft.com/en-us/research/wp-content/uploads/2014/01/MSR-TR-2014-4.pdf

SHORTEST PATHS ALGORITHMS

EXTENSIONS

Path Retrieval

- Retrieve the **shortest path itself**, not just the length
- No shortcuts (Dijkstra, A*, Arc Flags): Parent pointer
- With shortcuts (CH, SHARC, CRP): Additionally unpacking shortcuts

Batched Shortest Paths

- Source set, target set
- Point-of-Interest queries

Dynamic Networks

- Transportation networks have **unpredictable delays, traffic, or closures**
- If the modified network is stable for the foreseeable future, just rerun preprocessing algorithm
- Three other approaches

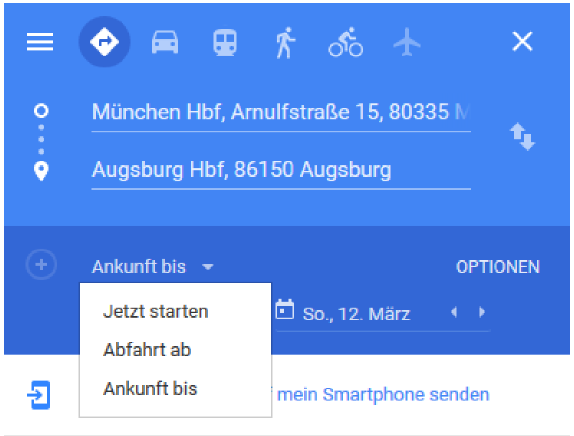
1. **“Repair” preprocessed data** instead of rebuilding it
2. **Adapt query algorithm** to work around “wrong” parts of the preprocessing phase
3. **Split preprocessing phase** into metric-independent and metric-dependent stages

SHORTEST PATHS ALGORITHMS

EXTENSIONS

Time-Dependence

- In real transportation networks, the best route often **depends on the departure time in a predictable way**
- Time-dependent shortest path problem
  → earliest possible arrival
  → last departure
- Profile searches
  → finding best departure time for minimizing total time in transit



Multiple Objective Functions

- Consider multiple cost functions
- Edge restrictions
  → e.g. certain vehicle types cannot use all segments
- Pareto Set
  → "take a more scenic route even if the trip is slightly longer"

APPLICATIONS

ALTERNATIVE ROUTES & CORRIDOR OF PATHS

Show the user **several "reasonable"** paths
(in addition to the shortest one)

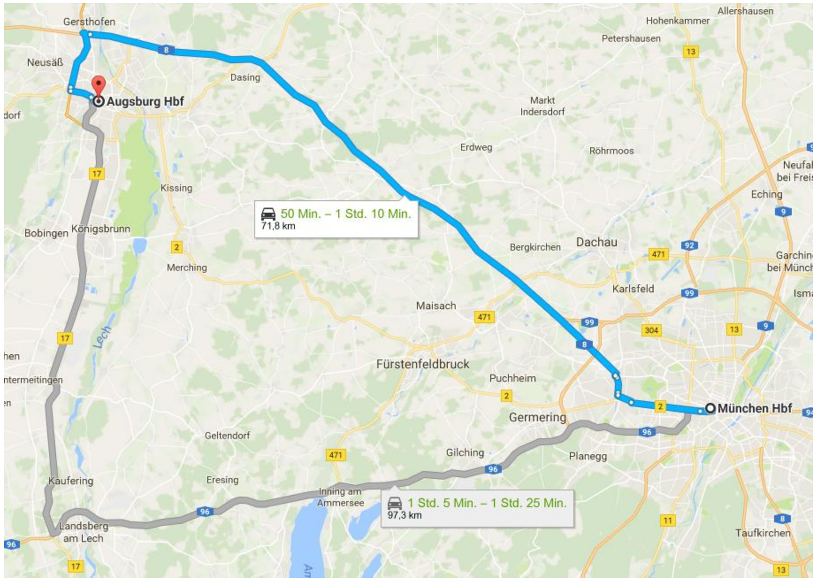
Alternative paths should be

- Short
- Smooth
- Significantly different from the shortest path and other alternatives

Alternative paths can be compactly represented as a **small graph**

Related Problem: **Corridor of paths**

- Allow deviations from the best route (while driving) to be handled without recomputing the entire path
- These robust routes can be useful in mobile scenarios with limited connectivity



APPLICATIONS

MISCELLANEOUS

Nontrivial cost functions

- Flexible arc restrictions such as **height or weight limitations**
- Multiple criteria (such as optimizing costs and travel time)

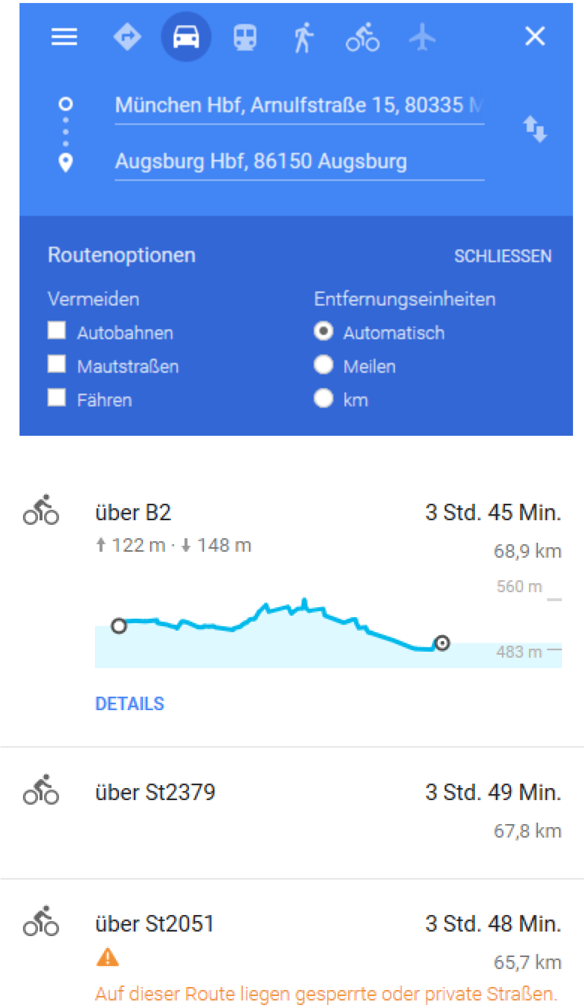
Minimizing the energy consumption of **electric vehicles**

- Recharging batteries when the car is going downhill

Optimal **cycling routes** (amount of uphill cycling)

Fast computation of **many (batched)** shortest paths

- Match GPS traces to road segments
- Traffic simulations
- Route prediction
- Ride sharing
- Point-of-interest queries



ROUTE PLANNING

FINAL REMARKS

Successful approaches **exploit different properties** of the road networks that make them easier to deal with

Geometry-based algorithms are consistently dominated by established techniques

Careful engineering is essential to unleash the full computational power of modern computer architectures (exploit locality of reference and parallelism)

The ultimate goal

- A worldwide multimodal journey planner, that takes into account real-time traffic and transit information, historic patterns, schedule constraints, and monetary costs
- Moreover, all the elements should be combined in a personalized manner

LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Praktikum Mobile und Verteilte Systeme

→ Alternative Routes

Prof. Dr. Claudia Linnhoff-Popien
André Ebert, Sebastian Feld, Thomy Phan
http://www.mobile.ifi.lmu.de

SoSe 2018

AR/AG IN STREET NETWORKS

EXAMPLES

Personalized routing based on preference

- CO_2-consumption
- toll pricing
- fuel consumption

… or based on experience

- scenic value
- risk of traffic jams

https://dl.acm.org/citation.cfm?id=2444019

AR/AG IN STREET NETWORKS

MOTIVATION

State-of-the-art

- Gather and sort existing work regarding **quality metrics** of **alternative routes** and **alternative graphs** in road networks

Constrained free space

- Clarify what **challenges** need to be tackled in order to create such metrics for constrained free space scenarios
- **Discussion** of possible courses of action, opportunities, and limitations

Examples

- Pedestrian navigation
- Even maritime or aviation scenarios

ALTERNATIVE ROUTES IN STREET NETWORKS

QUALITY METRICS – OVERVIEW

Central reference
- (Abraham et al., 2013) with predecessor (Abraham et al., 2010)
- Finding good alternatives by defining an "**admissible path**" using three measures

Approach
- Three measures as hard constraints for a target function
- Sort candidates and return first admissible path

Further improvements
- (Luxen, Schieferdecker, 2012)
- (Kobitzsch, 2013)

Alternative Routes in Road Networks

ITTAI ABRAHAM, DANIEL DELLING, ANDREW V. GOLDBERG
and RENATO F. WERNECK
Microsoft Research Silicon Valley

We study the problem of finding good alternative routes in road networks. We look for routes that are substantially different from the shortest path, have small stretch, and are locally optimal. We formally define the problem of finding alternative routes with a single via vertex, develop efficient algorithms for it, and evaluate them experimentally. Our algorithms are efficient enough for practical use and compare favorably with previous methods in both speed and solution quality.

Categories and Subject Descriptors: G.2.2 [**Graph Theory**]: Graph algorithms

General Terms: Algorithms, Experimentation, Measurement, Performance

Additional Key Words and Phrases: shortest paths, route planning, alternative paths, speedup techniques

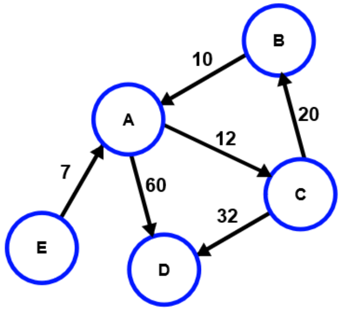
https://dl.acm.org/citation.cfm?id=2444019

ALTERNATIVE ROUTES IN STREET NETWORKS

PRELIMINARIES

Based on prosaic definitions, Abraham et al. formally define the class of paths to be found as "**admissible alternative paths**"

$G = (V, E)$	**directed graph** with nonnegative edge weights
$\|V\| = n$	number of **nodes**
$\|E\| = m$	number of **edges**
P	**path** in G
$\|P\|$	number of the path's edges
$l(P)$	sum of the edge weights
$l(P \cap Q)$	sum of edge weights **shared** by P and Q
$l(P \setminus Q)$	sum of edge weights **not shared** by P and Q
$Opt(s, t)$	point-to-point **shortest path** problem between s and t



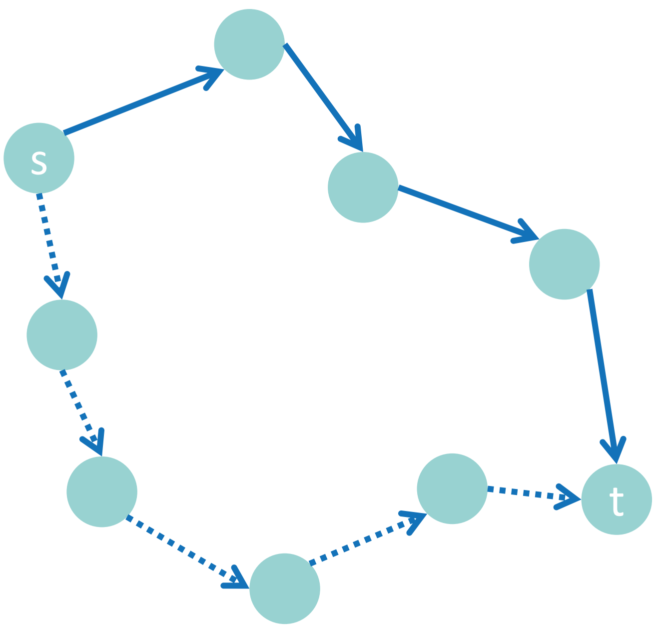
https://de.wikipedia.org/wiki/Graph_(Graphentheorie)#/media/File:CPT-Graphs-directed-weighted-ex1.svg

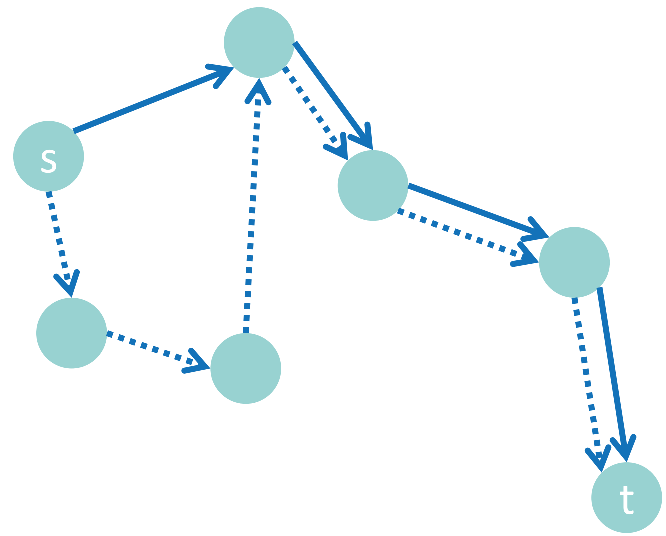
ALTERNATIVE ROUTES IN STREET NETWORKS

LIMITED SHARING

Limited Sharing

- The alternative path has to be **significantly different** to the reference path
- I.e., the total length of edges shared must be a small fraction of the reference route’s length





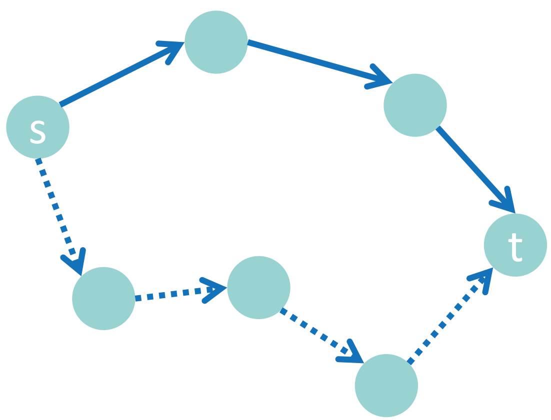
$$l(Opt \cap P) \leq \gamma \cdot l(Opt)$$

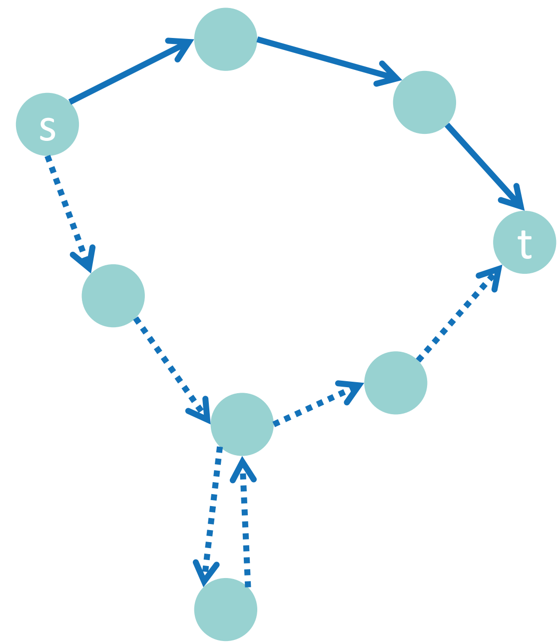
ALTERNATIVE ROUTES IN STREET NETWORKS

LOCAL OPTIMALITY

Local Optimality

- The alternative path must be **reasonable**
- I.e., no unnecessary detours are allowed
- Every local decision must make sense, so every subpath up to a certain length is a shortest path





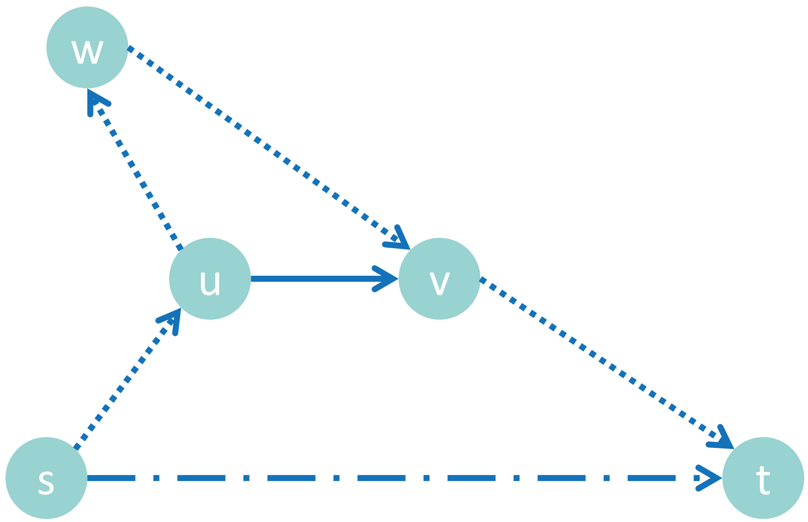
P is T-locally optimal for $T = \alpha \cdot l(Opt)$.
A path P is T-locally optimal if every subpath P' of P with $l(P') \leq T$ is a shortest path

ALTERNATIVE ROUTES IN STREET NETWORKS

UNIFORMLY BOUNDED STRETCH (UBS)

Uniformly Bounded Stretch (UBS)

- The alternative path must **not be much longer** than the reference path
- I.e., every subpath needs to have a good stretch
- This enhances local optimality: a path with high optimality may be shortened with a shortcut



P is $(1+\varepsilon)$-UBS.
A path P has $(1+\varepsilon)$-UBS if for every subpath P' of P with end points s', t', the inequality $l(P') \leq (1+\varepsilon) \cdot l\big(Opt(s',t')\big)$ holds

Rationale: the alternative through w is a concatenation of two shortest paths, s-w and w-t.

Although it has high local optimality, it looks unnatural because there is a much shorter path between u and v.

ALTERNATIVE GRAPHS IN STREET NETWORKS

QUALITY METRICS – OVERVIEW

Central reference

- (Bader et al., 2011), based on Dees' master's thesis (Dees, 2010)
- Preliminary aspects published before in (Dees et al., 2010)
- Definition of an **alternative graph** (AG) as the union of several paths having the same start and goal as a **compact representation** of multiple alternative routes

Approach

- Calculate shortest path
- Insert into AG
- Gradually calculate further alternative paths
- Insert greedily into AG optimizing a target function

Further work

- Efficient implementations: (Radermacher, 2012), (Kobitzsch et al., 2013)
- Higher quality: (Paraskevopolous, Zaroliagis, 2013)

Alternative Route Graphs in Road Networks*

Roland Bader[1], Jonathan Dees[1,2], Robert Geisberger[2], and Peter Sanders[2]

[1] BMW Group Research and Technology, 80992 Munich, Germany.
[2] Karlsruhe Institute of Technology, 76128 Karlsruhe, Germany.

Abstract. Every human likes choices. But today's fast route planning algorithms usually compute just a single route between source and target. There are beginnings to compute *alternative routes*, but there is a gap between the intuition of humans what makes a good alternative and mathematical definitions needed for grasping these concepts algorithmically. In this paper we make several steps towards closing this gap: Based on the concept of an *alternative graph* that can compactly encode many alternatives, we define and motivate several attributes quantifying the quality of the alternative graph. We show that it is already NP-hard to optimize a simple objective function combining two of these attributes and therefore turn to heuristics. The combination of the refined penalty based iterative shortest path routine and the previously proposed Plateau heuristics yields best results. A user study confirms these results.

https://link.springer.com/chapter/10.1007/978-3-642-19754-3_5

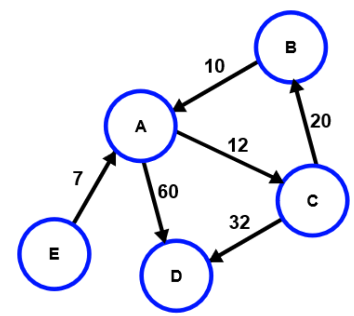
ALTERNATIVE GRAPHS IN STREET NETWORKS

PRELIMINARIES

After depicting the measures prosaically, Bader et al. turn to the **formal definitions**

$G = (V, E)$ — **graph** with edge weight function $w: E \to \mathbb{R}_+$
s, t — **source** node and **target** node

$H = (V', E')$ — **alternative graph** with $V' \subseteq V$ such that for every edge $e \in E'$ there exists a simple s-t-path in H containing e



For every edge (u, v) in E' there must be a path from u to v in G and the edge weights $w(u, v)$ must be equal to the path's weight

$d_G(u, v)$ — shortest path distance from u to v in G
$d_H(u, v)$ — shortest path distance from u to v in H

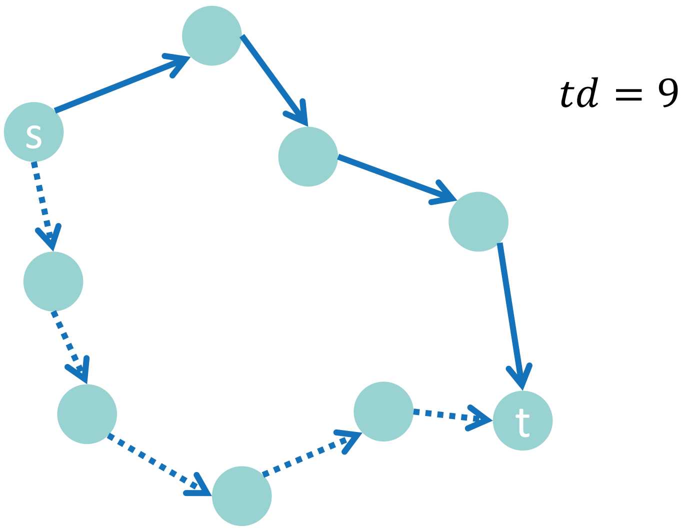
https://de.wikipedia.org/wiki/Graph_(Graphentheorie)#/media/File:CPT-Graphs-directed-weighted-ex1.svg

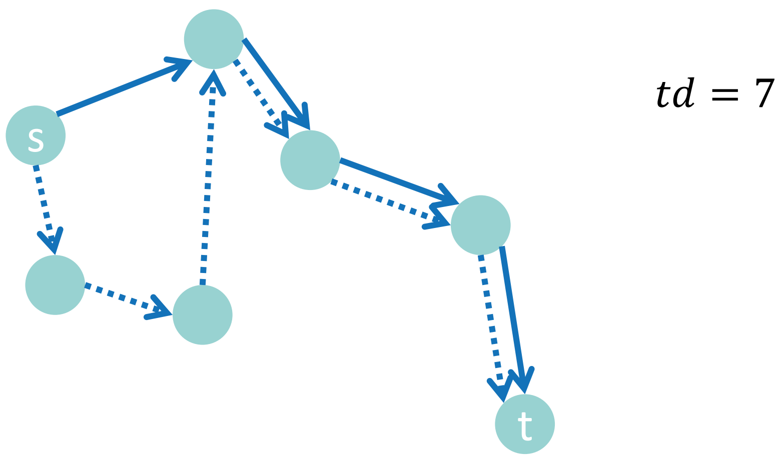
ALTERNATIVE GRAPHS IN STREET NETWORKS

TOTAL DISTANCE

Total Distance

- Describing the extent to which the routes defined by the AG are **non-overlapping**
- Maximum value when the AG consists of disjoint paths only





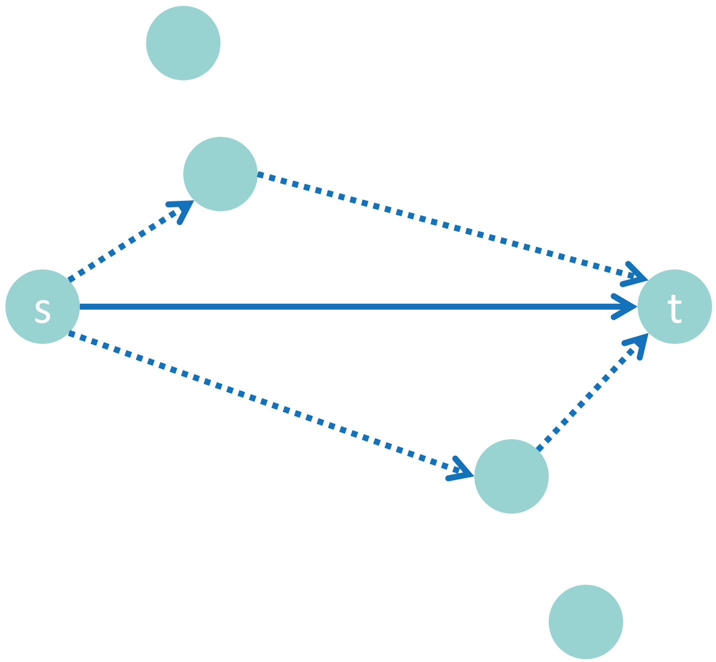
$$\sum_{e=(u,v)\in E'} \frac{w(e)}{d_H(s,u) + w(e) + d_H(v,t)}$$

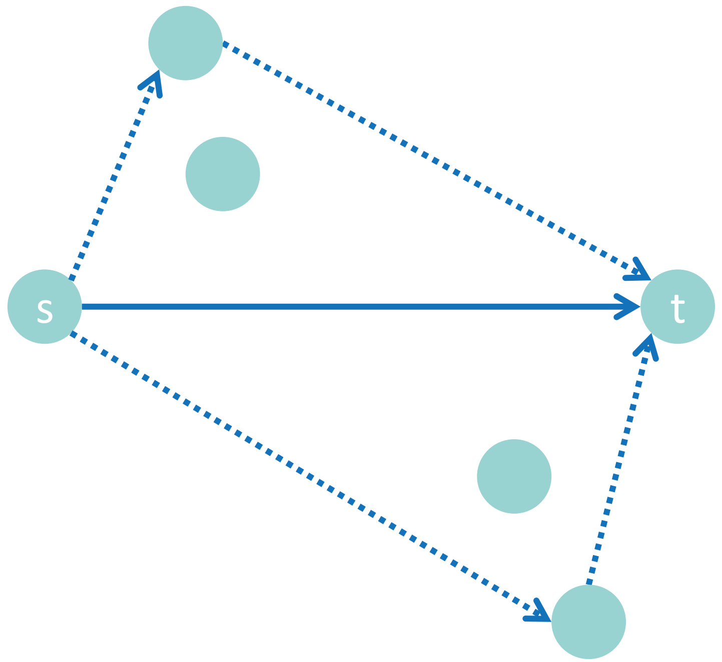
ALTERNATIVE GRAPHS IN STREET NETWORKS

AVERAGE DISTANCE

Average Distance

- Describing the quality as the **average stretch** of an alternative path





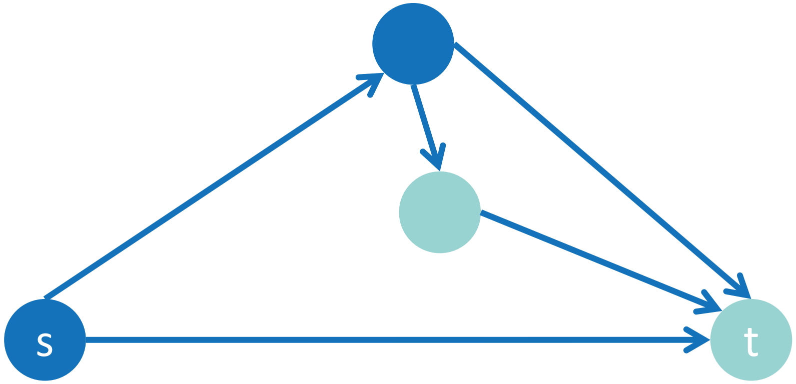
$$\frac{\sum_{e \in E'} w(e)}{d_G(s,t) \cdot totalDistance}$$

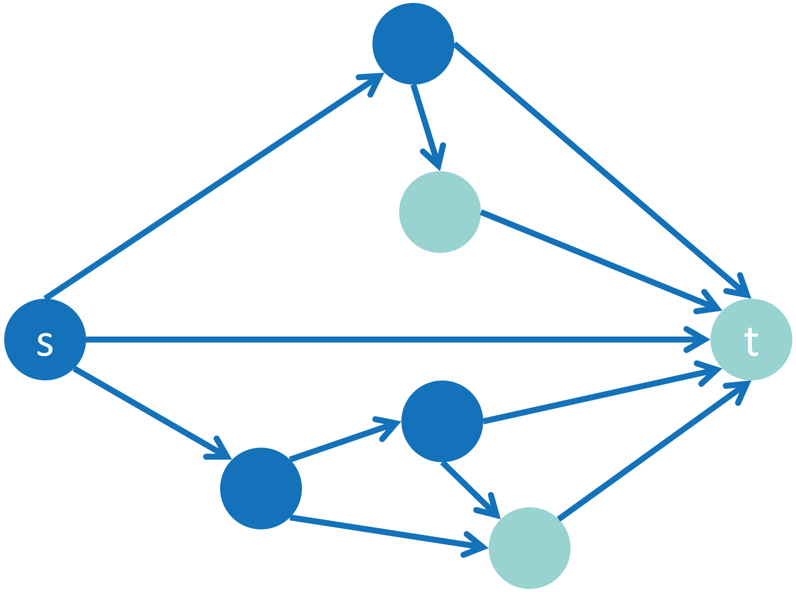
ALTERNATIVE GRAPHS IN STREET NETWORKS

DECISION EDGES

Decision Edges

- Describing the **complexity** of the AG
- Used to retain the representation easily understandable for human users





$$\sum_{v \in V' \setminus \{t\}} outdegree(v) - 1$$

AR/AG IN STREET NETWORKS

SUMMARY

Abraham et al. state that a proper **alternative route** should

- be substantially different from a reference path (“limited sharing”)
- not have unnecessary detours (“local optimality”)
- not be much longer than the shortest path (“uniformly bounded stretch”)

Bader et al. proposed that a good **alternative graph** should have

- low overlap of the included routes (high “total distance”)
- low stretch of included alternatives (low “average distance”)
- low complexity (few “decision edges”)

Praktikum Mobile und Verteilte Systeme

→ Lessons Learned

Prof. Dr. Claudia Linnhoff-Popien
André Ebert, Sebastian Feld, Thomy Phan
http://www.mobile.ifi.lmu.de

SoSe 2018

LOCATION-BASED SERVICES

CONCLUSION

Navigation and route planning as an important part of LBS

Spatial information as part of context-aware computing

Approaches and ideas to be discussed are more of **tools** rather than **applications**

Topics
- Trajectory Computing
- (Big) Data Analysis for Geospatial Trajectories
- Somewhat Information Retrieval

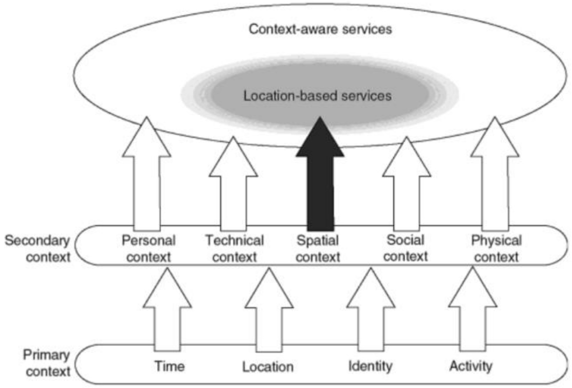
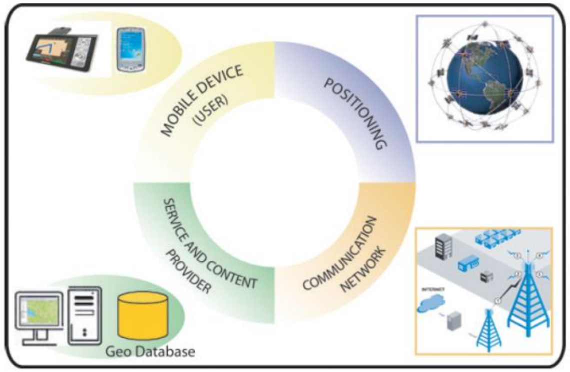


Figure 1.1 Context-aware and location-based services.



ROUTE PLANNING

FINAL REMARKS

Successful approaches **exploit different properties** of the road networks that make them easier to deal with

Geometry-based algorithms are consistently dominated by established techniques

Careful engineering is essential to unleash the full computational power of modern computer architectures (exploit locality of reference and parallelism)

The ultimate goal

- A worldwide multimodal journey planner, that takes into account real-time traffic and transit information, historic patterns, schedule constraints, and monetary costs
- Moreover, all the elements should be combined in a personalized manner

AR/AG IN STREET NETWORKS

SUMMARY

Abraham et al. state that a proper **alternative route** should

- be substantially different from a reference path ("limited sharing")
- not have unnecessary detours ("local optimality")
- not be much longer than the shortest path ("uniformly bounded stretch")

Bader et al. proposed that a good **alternative graph** should have

- low overlap of the included routes (high "total distance")
- low stretch of included alternatives (low "average distance")
- low complexity (few "decision edges")