Knowledge-based route planning

Data modeling and multi-criteria time-constrained route optimization for people with disabilities

Dipl.-Ing. Bettina Pressl

Doctoral Thesis
for obtaining the academic degree of
Doctor of Technical Sciences

Institute of Navigation (INAS),
Graz University of Technology
A-8010 Graz, Austria

Graz University of Technology

August 2012

Supervisor 1: Dr. Manfred Wieser, Graz University of Technology
Supervisor 2: Dr. Josef Strobl, University of Salzburg

ACKNOWLEDGEMENTS

First of all, I would like to thank my supervisor Manfred Wieser for his good suggestions and inspiring meetings during my work and the chance to write this thesis at the Institute of Navigation. Another big thankyou goes to my second supervisor Josef Strobl for agreeing to referee this thesis and his fruitful advice. Many thanks go to the project partners for the joint research. In particular, I would like to thank: Alice Geiger for answering lots of questions about user behavior. As a blind person, she has lots of experience in this field. She was also the one who was motivating me during work with her enthusiastic and cheerful character; Christoph Mader and Christoph Kubesch from SOLVION information management for supporting the programming. Without them, the practical part would not be as it is.

Furthermore, I would like to thank my parents Christa and Karl Pressl and my boyfriend Martin Krammer for their patience and support. Finally, I apologize to my cat and my four bunnies for not having much time for them.

Bettina Pressl
Weiz, Austria, August 2012

Statutory Declaration

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitely marked all material which has been quotes either literally or by content from the used sources.

.... date

.... (signature)

Contents

Nomenclature **ix**

List of Figures **xiii**

1 Introduction **1**
1.1 Overview 1
1.2 Motivation and goals 2
1.2.1 Goals 2
1.2.2 Research questions 4
1.3 Awards 5
1.4 Existing systems and related literature 5
1.4.1 Portable navigation systems 5
1.4.2 Route planners and timetable information systems . . 5
1.5 Structure of the thesis 6

I Conceptual foundations **9**

2 Graph theory **11**
2.1 Graphs 11
2.2 Graph elements 12
2.2.1 Main graph elements 12
2.2.2 Complex graph elements 13
2.3 Properties of graphs 13
2.3.1 Geometry 17
2.4 Storage of graphs 17
2.4.1 Adjacency lists 17
2.4.2 Indexed adjacency lists 18

3 Shortest path problems **21**
3.1 Classical shortest path problems 21
3.1.1 Single-criterion optimization 21
3.1.2 Types of solution methodologies 22
3.2 Dijkstra's shortest path algorithm 22

3.2.1 Procedure of Dijkstra's algorithm 23
3.2.2 Dijkstra's algorithm with priority queue 25
3.2.3 Label-setting algorithms 26
3.2.4 Label-correcting algorithms 27

4 Multi-criteria optimization 29
4.1 Introduction 29
4.1.1 Terminology 30
4.2 Efficiency and dominance 31
4.2.1 Pareto optimality 31
4.2.2 Dominance 31
4.2.3 Strict dominance 32
4.3 Definition of optimality 33
4.3.1 Scalarization methods 33
4.3.2 Non-scalarization methods 33
4.4 Multi-criteria shortest path problems 34
4.4.1 Solutions of multi-criteria shortest path problems . . . 35
4.5 Dijkstra's algorithm in multi-criteria optimization 36
4.5.1 Extended Dijkstra's algorithm 37

5 Concepts for route planning 43
5.1 Overview 43
5.1.1 Strict Pareto optimal paths 43
5.1.2 Complexities of multi-criteria optimization 43
5.2 Single Pareto optimal solutions 44
5.2.1 Bi-criteria optimization with lexicographic optimality 44
5.2.2 Bi-criteria optimization using the weighted sum method 45
5.3 Subsets of Pareto optimal solutions 45
5.3.1 Subsets by combining single solutions 45
5.3.2 Subsets by dominance with restrictions 46

6 Modeling of schedules 51
6.1 Time-expanded graph model 51
6.2 Time-dependent graph model 52
6.3 Time model 52
6.3.1 Scheduling constraints 53
6.4 Optimization criteria 53
6.5 Optimization problems in public transport 53
6.5.1 Earliest arrival problem 53
6.5.2 Minimum number of transfers problem 54
6.5.3 Earliest arrival problem with minimum number of transfers problem 54

II Application **55**

7 User groups and demands **57**
7.1 User groups 57
7.2 Mobility 57
7.3 Accessibility of the environment 58
7.3.1 Guidelines for accessible construction 58
7.4 Constructional orientation aids 60
7.4.1 Tactile paving 61
7.4.2 Acoustic traffic lights 61
7.5 Navigating in unknown environments 62
7.5.1 Information needed 62
7.5.2 Obstacles 63
7.5.3 Preferred Routes 65
7.5.4 Content of guidance information 65
7.5.5 Public transport and accessibility 66
7.5.6 Accessibility of the user interface 67

8 Software requirements **69**
8.1 Functional requirements 69
8.1.1 Login 70
8.1.2 Profile selection 70
8.1.3 Search 70
8.1.4 Map display 71
8.1.5 Creating points of interest 71
8.1.6 Points of interest administration 71
8.1.7 User profile 72
8.1.8 Route planning 72
8.1.9 Pre-trip training 73
8.1.10 Soundfile download 73
8.2 Non-functional requirements 73
8.2.1 Usability 74
8.2.2 Accessibility 74
8.2.3 Web accessibility 75
8.2.4 Performance 76
8.3 Usage scenarios 77
8.3.1 Wheelchair user 77
8.3.2 Blind person 77

9 Geographical data **79**
9.1 Modeling the real world 79
9.1.1 Digital maps 80
9.1.2 Digital maps for route planning 80
9.2 Geographical data sources 81

9.3 OpenStreetMap 82
9.3.1 Quality aspects 82
9.3.2 OpenStreetMap data content 83
9.3.3 OpenStreetMap for route planning 83
9.3.4 OpenStreetMap data content for people with disabilities 84
9.4 Data for Pedestrians 85
9.4.1 Detailed pedestrian network for people with disabilities 85
9.4.2 Categories of information 86
9.4.3 Test area 88
9.4.4 Comparison of OpenStreetMap street network data and detailed data 89
9.5 Public transport network and timetables 90
9.6 Integration of different data sources 90
9.6.1 Multi-scale and multi-modal data 91
9.6.2 Data integration for route planning 92
9.7 User generated data 93
9.7.1 Points of Interest 94

10 Multi-criteria route planning in combined networks 97
10.1 Route planning in multi-modal networks 97
10.1.1 Homogeneous routing concept 97
10.1.2 Hierarchical routing concept 98
10.2 Modeling of public transport networks 100
10.2.1 Raw timetable data 100
10.2.2 Time-expanded model 101
10.2.3 Extensions for time-constrained optimization in real public transport networks 102
10.2.4 Transfer edges and transfer nodes 103
10.3 Multi-criteria time-constrained route planning in combined networks 104
10.3.1 Optimization criteria 105
10.3.2 Basic definitions on multi-criteria optimization 105
10.4 Optimization methods for multi-criteria time-constrained route planning 106
10.4.1 Multi-criteria optimization with lexicographic optimality 106
10.4.2 Multi-criteria optimization with restrictions on the criteria including additive values 106

11 Application 111
11.1 Software architecture 111
11.2 Technologies 112
11.2.1 Internet Information Services 112
11.2.2 Windows Presentation Foundation 113

11.2.3 Microsoft Silverlight 113
11.2.4 Windows Communication Foundation 113
11.2.5 Microsoft Sharepoint 2010 113
11.2.6 Microsoft SQL Server 2008 114
11.3 Graphical user interface 114
11.3.1 Menu structure 114
11.4 User profiles 115
11.4.1 User settings 115
11.5 Functionalities 117
11.5.1 Route planning 117
11.5.2 Pre-trip training 119
11.6 Route description 121
11.6.1 Route information based on different networks 121
11.6.2 Turn information 122
11.6.3 Route description in different networks 123
11.7 Validation 125

12 Conclusion and Outlook **129**
12.1 Conclusion 129
12.2 Research results and future work 129
12.2.1 Geographical data 129
12.2.2 Route planning 130
12.2.3 Requirements on the application 131
12.3 Outlook 132

Bibliography **133**

Nomenclature

API	Application Programming Interface
ASP	Active Server Pages
BAIM	BArrierefreie ÖV-Information für Mobilitätseingeschränkte Personen (Barrier-free public transport information for disabled people)
BMN	BundesMeldeNetz
BMVIT	Bundesministerium für Verkehr, Innovation und Technologie (Federal Ministry of Transport, Innovation and Technology)
CON	Connection Graph
CPU	Central Processing Unit
CSS	Cascading Style Sheets
DET	DETailed pedestrian network for people with disabilities
EAP	Earliest Arrival Problem
EFA	Elektronische FahrplanAuskunft (electronic timetable information)
FTP	File Transfer Protocol
FTPS	File Transfer Protocol Secure
GIS	Geographic Information System
GK	Gauß-Krüger
GPS	Global Positioning System
GUI	Graphical User Interface
HAFAS	HaCon Fahrplan-Auskunfts-System (HaCon timetable information system)

HTTP	HyperText Transfer Protocol
HTTPS	HyperText Transfer Protocol Secure
ID	IDentifier
IIS	Internet Information Services
JAWS	Job Access With Speech
MNTP	Minimum Number of Transfers Problem
MoBIC	Mobility of blind and elderly People Interacting with Computers
MOSP	Multi-Objective Shortest Path
MOTIS	Multi Objective Traffic Information System
MSPP	Multi-criteria Shortest Path Problem
NNTP	Network News Transfer Protocol
NP	Non-deterministic Polynomial-time
OGC	Open Geospatial Consortium
OSM	OpenStreetMap
POI	Point Of Interest, Points Of Interest
POPTIS	Pre - On - Post - Trip - Informations - System
PT	Public Transport
PTN	Public Transport Networks
RAM	Random Access Memory
SMTP	Simple Mail Transfer Protocol
SPP	Shortest Path Problem
SQL	Structured Query Language
TCRP	Time-Constrained Route Planning
UI	User Interface
VGI	Volunteered Geographic Information
WAI	Web Accessibility Initiative
WCAG	Web Content Accessibility Guidelines

WCF	Windows Communication Foundation
WGS84	World Geodetic System 1984
WPF	Windows Presentation Foundation
WPF/E	Windows Presentation Foundation Everywhere
XAML	Extensible Application Markup Language
ÖNORM	Österreichische NORM (Austrian standard)

List of Figures

2.1 Bridges of Königsberg 12
2.2 Corresponding graph of *The Seven Bridges of Königsberg* . . 12
2.3 Main graph elements 13
2.4 Complete graph 14
2.5 Planar graph (left) vs. nonplanar graph (right) 15
2.6 Directed graph 15
2.7 A cyclic graph (left) and an acyclic graph (right) 16
2.8 Valuated graph 16
2.9 Triangular inequality 16

3.1 Example graph with node numbers and edge costs 24
3.2 Procedure of Dijkstra's algorithm 28

4.1 Criterion values in the bi-criteria car example, modified from [17] 30
4.2 Pareto optimal solutions, [71] 32
4.3 Bi-criteria cost vectors for edges 35
4.4 All Pareto optimal solutions of the example graph in Figure 4.3 36
4.5 Illustration of Pareto optimal solutions of node 4 in Figure 4.4 36
4.6 Steps 1-4 of modified Dijkstra's algorithm for calculating all Pareto optimal paths, having two different criteria 41
4.7 Steps 5-10 of modified Dijkstra's algorithm for calculating all Pareto optimal paths, having two different criteria 42

5.1 Approximation methods of multi-criteria optimization for route planning 44
5.2 Lexicographic minimal solutions 46
5.3 Procedure of Dijkstra's algorithm calculating a single Pareto optimal solution (Lexicographic minimal solution with f_1 = 'time' and f_2 = 'number of transfers') 48
5.4 Lexicographic minimal solution with f_1 = 'number of transfers' and f_2 = 'time' 49

6.1 Difference between time-expanded and time-dependent graph models of a timetable with three stations $S1$, $S2$, $S3$. There are three public transport vehicles that connect $S1$ with $S2$ (u, v, w), one connection from $S3$ via $S2$ to $S1$ (x, y) and another connection from $S3$ to $S2$ (z) (example modified from [45]) . . . 52

7.1 Widths needed [cm] . . . 59
7.2 Inclination of sidewalk max. 6% . . . 59
7.3 Stairs . . . 60
7.4 Disabled parking space [cm] . . . 60
7.5 Patterns . . . 61

8.1 Functionality overview . . . 70
8.2 Braille characters . . . 75

9.1 Flowchart of a data modeling process (modified from [7]) . . 80
9.2 Geographical vector data and their attributes (modified from [34]) . . . 81
9.3 Geographical data for route planning including attribute information . . . 82
9.4 OSM data content (April 2010) . . . 84
9.5 OSM data content (February 2012) . . . 85
9.6 Street network in comparison with a more detailed pedestrian network . . . 86
9.7 Pedestrian network for route planning . . . 87
9.8 Test area for the detailed path network . . . 89
9.9 Comparison of OpenStreetMap street network data and detailed data from the land surveying office . . . 90
9.10 Comparison of OpenStreetMap street network data and detailed data from the land surveying office . . . 91
9.11 Connection of OSM network and pedestrian network using a connection graph with transfer nodes . . . 93
9.12 POI administration . . . 95

10.1 Circular search space . . . 98
10.2 Bidirectional search in homogeneous networks . . . 99
10.3 Hierarchical scheme of combined networks . . . 99
10.4 Hierarchical search in combined networks . . . 100
10.5 Raw timetable data from *Holding Graz Linien* . . . 101
10.6 Raw timetable data with data-set entries . . . 101
10.7 Timetable of *Holding Graz Linien* in time-expanded structure for route planning . . . 102
10.8 Public transport scheme, modified from [62] . . . 103

10.9 Left: Transfer edges connect stations directly with $n(n-1)$ arcs, Right: A transfer node reduces the number of arcs ($2n$ arcs) . . . 104

11.1 System overview . . . 111
11.2 Technologies . . . 112
11.3 Start page *(published in agreement with the project consortium of Route4you, see Chapter 1.2)* . . . 115
11.4 Main page *(published in agreement with the project consortium of Route4you, see Chapter 1.2)* . . . 116
11.5 Profile settings *(published in agreement with the project consortium of Route4you, see Chapter 1.2)* . . . 118
11.6 Stages from the user input of the address to the start of the route planning algorithm . . . 119
11.7 Route for wheelchair users *(published in agreement with the project consortium of Route4you, see Chapter 1.2)* . . . 120
11.8 Route for blind people *(published in agreement with the project consortium of Route4you, see Chapter 1.2)* . . . 120
11.9 Pre-trip Training *(published in agreement with the project consortium of Route4you, see Chapter 1.2)* . . . 121
11.10 Left: The course calculated from $\arctan(|\Delta y| \, / \, |\Delta x|)$ without any correction is an angle in the first quadrant clockwise from the x-axis. Right: The numbering of quadrants is shown. . . 123
11.11 A positive value of $course_{relative}$ defines a turn on the right . 124
11.12 Turn information, modified from [76] . . . 124

Chapter 1

Introduction

1.1 Overview

While route planners for car drivers are widespread over the Internet, navigation systems and route planners for pedestrians are rare, but are becoming more and more interesting. Available systems for pedestrians are mostly based on a street network of vehicle navigation systems. Such a data structure is insufficient to model the behavior of pedestrians and does not provide adequate route information. Particularly, people with disabilities will benefit from the use of a route planner to increase their mobility by getting specific information. However, people with disabilities have special demands on a route planner. This includes data content as well as route optimization, usability and accessibility. Concerning the data content, making their way without obstacles or avoiding routes containing impassable objects is one challenging task. Moreover, blind people need much more details on the route output to orientate in unknown environments. Another point is, that people with different types of disabilities have contrasting demands on route information, e.g., blind people have other requirements on a barrier-free route than wheelchair users. While for the latter, routes must not contain stairs, for blind people obstacles on the sidewalk (e.g., traffic signs) are dangerous. Furthermore, blind people need much more information about the surroundings like buildings or infrastructure when visiting unknown cities, [9]. These facts lead to the idea of developing a route planner for various user groups. To realize the planning of a barrier-free route for different demands, user profiles shall allow the specification of individual criteria for an optimal route from an arbitrary starting point to the desired destination. In the existing case of pedestrian users, footpaths as well as public transport are essential and have to be considered. Moreover, blind and visually impaired people have special requirements on the usability and accessibility of the user interface.

In context of this work, 'knowledge-based route planning' means the inclusion of user generated information in the route planner. In more detail, users have the possibility to influence route optimization and the content of the geographical database. In this work 'knowledge-based' is not associated with strategies of knowledge engineering.

1.2 Motivation and goals

This doctoral thesis came up with the Austrian research project *Route4you* funded by the Austrian grant program *ways2go* of the *BMVIT* (Federal Ministry of Transport, Innovation and Technology). The project lead was taken by the Institute of Navigation of Graz University of Technology. The content of the project was the conceptual design and development of a route planner for people with disabilities, covering different user groups like blind and visually impaired people as well as wheelchair users. The project was accomplished with further project partners. The software company SOLVION information management in Graz, Austria supported the project with their programming skills. The public transport company *Holding Graz Linien* provided their timetables for the route planner. Last but not least, Alice Geiger from an organization for people with disabilities in Graz, Austria ('Behindertenbeauftragte der Stadt Graz') represented the user group of blind people as she is blind herself and brought us into contact with various potential users (blind people, visually impaired people and wheelchair users). The project partners agreed with the publication of project results (screenshots of the route planner) in this thesis.

1.2.1 Goals

An innovation of the route planner is the integration of geographical data from different data sources tailored to the needs of people with disabilities. Moreover, providing user profiles where each user has the possibility to define their own route properties is a new aspect. Last but not least, user-specific multi-criteria time-constrained route optimization differs from the route planning algorithm used in most of existing route planners. These innovative topics are elaborated in the present thesis. It includes three main goals with respect to people with disabilities:

- Establishment of a geographical database structure for route planning
- Implementation of an algorithm for multi-criteria optimization
- Investigation of requirements on the application (user profiles, usability, accessibility)

Establishment of a geographical database for route planning

Route planning requires a geographical database including a path network to calculate an optimal route and to provide additional route information. In case of people with disabilities a pedestrian path network is needed which differs from vehicle navigation networks in degree of details. Such a pedestrian network is not available from existing data providers like it is for car navigation. Furthermore, people with reduced mobility need much more information than people without disabilities. Another main difference in their mobility is the appearance of obstacles and even more the occurrence of impassable objects. All in all, this leads to a combination of different data sources and the generation of a combined path network to fulfill the special requirements on the geographical database. Another idea is the implementation of user generated data, where users are able to add individual information like points of interest, obstacles, or accessibility information to the system.

Implementation of an algorithm for multi-criteria optimization

The inclusion of public transport and individual optimization criteria leads to a multi-modal and multi-criteria time-constrained shortest path problem, [12]. Such optimization cannot be solved with a conventional shortest path algorithm. An extended search algorithm has to be implemented, [56]. An important part of this work is the explanation of the methodology of such a multi-criteria optimization algorithm. Different solution methods are explained in detail and compared to each other. In the practical part of this work, a multi-criteria optimization algorithm is realized and discussed within a route planner for people with disabilities.

Investigation of requirements on the application

Blind and visually impaired people have other requirements on the usability of a route planner than sighted people have. This fact mainly concerns the user interface. A simple menu structure will help them to get an overview and make it easy to work with the system. Moreover, the design of the user interface includes a way to add information to the system and to select preferences, as well as a method for information output. As blind people think sequentially and have hardly any spatial sense, the route information has been adapted to their needs. A user profile will allow the specification of individual preferences and the selection of obstacles and points of interest. Furthermore, the user interface has to fulfill web accessibility, which is well defined in existing guidelines, [18].

1.2.2 Research questions

For the three goals described in the previous section, some research questions can be formulated. These research questions will be solved in the present thesis. Keeping the same sequence as just above, the first topic regarded is the geographical database.

Establishment of a geographical database for route planning

The development of a detailed database causes various questions:

- Which geographical databases can be combined to provide route planning for pedestrians with special needs?
- Are there any data sources providing detailed information needed by people with disabilities?
- How can different data sources be combined?
- How to model pedestrian networks and public transport networks?
- Which obstacles are dangerous or even impassable for the user groups?
- How to get these obstacles and further specific information in the geographical database?

Implementation of an algorithm for multi-criteria optimization

With regard to route planning, for the implementation of the multi-criteria time-constrained optimization algorithm some questions arise:

- Which algorithm can be used to solve a problem with multiple criteria?
- How best to combine different criteria?
- What is the performance of the algorithm?
- What is the quality of the results?

Investigation of requirements on the application

Concerning the realization of the application itself, questions like the following arise:

- Which are the special demands and requirements of people with disabilities on a route planner?
- Which individual information should be definable in the user profile?
- How can blind people use the functionalities of the route planner best?

1.3 Awards

The research project 'Route4you', where this thesis is based on, was accomplished by the Institute of Navigation of Graz University of Technology in close cooperation with three further project partners. The project 'Route4you' was awarded with various prizes:

- 1st place at the Austrian ebiz egovernment award 2010
- 1st place at the Styrian ebiz egovernment award 2010
- 2nd place at the Microsoft Innovation Award 2010
- 2nd place at the Constantinus Award 2010

1.4 Existing systems and related literature

There is already existing work about travel aids and assistive technologies for navigating blind people in unknown environments. Existing systems can be divided into portable navigation systems including a positioning module and timetable information systems in the Internet.

1.4.1 Portable navigation systems

One of the first pedestrian navigation system for blind people was introduced by the MoBIC Consortium (Mobility of Blind and Elderly People Interacting with Computers) in the nineties, [47, 52, 53]. Initial work was also done by Golledge, Klatzky and Loomis, [40, 23]. Another prototype of a navigation system for blind and visually impaired people is Drishti, [30, 61]. The first commercially available navigation systems for blind people are Trekker[1] and Sendero GPS[2]. The disadvantage of these systems is the use of data optimized for car navigation. Systems that try to include data for pedestrians are Tania, [33], Pontes and Odilia, [42], and Nav4blind, [72]. A new project was started at the end of 2009 at Ceit in Vienna which deals with an interactive digital city map for blind and visually impaired people, [74].

1.4.2 Route planners and timetable information systems

Two research projects concerning timetable information systems are BAIM (Barrierefreie ÖV-Information für mobilitätseingeschränkte Personen)[3] and BAIMplus from the 'Rhein-Main Verkehrsverbund' in Germany, [10]. These

[1] http://www.humanware.com
[2] http://www.senderogroup.com
[3] http://www.rmv.de/baim

projects deal with the provision of public transport information for people with reduced mobility in a test area in Germany.
In Austria, POPTIS (Pre - On - Post - Trip - Informations - System)[4] is the barrier-free timetable information from the Vienna public transport company ('Wiener Linien').
Regarding the group of wheelchair users, the routing-project 'Rollstuhlrouting'[5] was done in the city of Bonn where some necessary data for wheelchair users is mapped in the free digital map OpenStreetMap[6].
In [27] and [64] the MOTIS (Multi Objective Traffic Information System) is described. It is a realistic timetable information system for trains concerning multiple criteria to find suitable connections, especially night train connections. However, requirements of people with disabilities are not considered in this route planner.

1.5 Structure of the thesis

This thesis describes the conceptual design and development of a route planner for people with disabilities. Especially blind and visually impaired people as well as wheelchair users are considered. The main focus lies on the geographical data base and the personalized route planning algorithm. This thesis is divided into two parts: conceptual foundations (Chapter 2, Chapter 3, Chapter 4, Chapter 5, Chapter 6) and the conceptional design and realization of the route planner (Chapter 7, Chapter 8, Chapter 9, Chapter 10, Chapter 11). The content of the thesis is structured as follows:

Chapter 2 gives an overview of graph theory as a basis for route planning. Graph elements as well as specific properties of graphs and their relevance in route planning are discussed.

Chapter 3 discusses route planning algorithms, especially single-criterion route planning. Furthermore, Dijkstra's algorithm for single-criterion shortest path searches is presented as a basis for multi-criteria shortest path problems.

Chapter 4 concentrates on the theory of multi-criteria shortest path problems and introduces Pareto optimal paths. Different solution methods are discussed. Finally, a multi-criteria version of Dijkstra's shortest path algorithm is explained in detail by showing a step-by-step instruction.

Chapter 5 introduces concepts for the implementation of the multi-

[4] http://www.wl-barrierefrei.at
[5] http://rollstuhlrouting.de
[6] http://wiki.openstreetmap.org/wiki/DE:Rollstuhlfahrer-Routing

criteria algorithms in route planning. Based on the conceptual foundations in Chapter 4, different solution methods are compared to each other. By giving examples, the usage of the presented algorithms in route planning is shown.

Chapter 6 describes the modeling of public transport networks in a theoretical way and gives an overview of the time-expanded graph model. The earliest arrival problem and the minimum number of transfers problem as well as their combination are introduced.

Chapter 7 elaborates on the demands of blind and visually impaired people as well as wheelchair users on a pedestrian route planner. Results of a questionnaire are given for getting an overview on how the user groups orientate in unknown environments. Moreover, relevant content of the ÖNORM B1600 about needs in barrier-free built environments is shown and additional information like obstacles are described. An ÖNORM is published by the Austrian Standards Institute and defines national standards.

Chapter 8 deals with software requirements and formally describes functional requirements as well as non-functional requirements on the route planner. Furthermore, usage scenarios are given.

Chapter 9 describes the geographical database focused on the user groups and the tools used for implementation. The main part is the combination of data sources and the modeling of geographical data for pedestrian route planning. Most of this chapter is practical work concerning the establishment of a database for the route planner. This includes the description of the acquired footpath network as well as combining timetable information and the public transport network with the implemented footpaths and OpenStreetMap data.

Chapter 10 discusses the structure of timetable data and the practical implementation of the real public transport network in the route planning tool. It defines the extended graph model for time-constrained route planning. Moreover, the optimization criteria and the influence of user specific restrictions on the criteria are discussed. Finally, the implementation of multi-criteria route planning is presented.

In **Chapter 11** the application itself is shown. First, the software architecture and technologies are explained. This is followed by the description of the graphical user interface as well as functionalities, user profiles and the pre-trip training. Another aspect is web accessibility which is an important feature for blind people.

Chapter 12 is summing up the work and discusses difficulties. Moreover, an outlook on further ideas is given.

Part I

Conceptual foundations

Chapter 2

Graph theory

Graph theory deals with the definition and properties of graphs, topological questions, and the corresponding algorithms for solving these questions. One example is the shortest path problem, but there are many more. In this chapter some basic definitions and properties of graphs as well as storage techniques of graphs are given. Graph theory is one of the conceptual foundations for the route planning algorithm discussed in the present work.

2.1 Graphs

Path networks for route optimization are represented as a graph which forms the basis of every route planning algorithm, [20]. Initially, graph theory was introduced by the Swiss mathematician and physicist *Leonhard Euler* to solve the well-known problem of *The Seven Bridges of Königsberg* as it is shown in Figure 2.1[1]. The problem was to find a closed tour through the city of Königsberg that would lead over each of the seven bridges across the river Pregel only once. Euler modeled the land mass and bridges as a graph (Figure 2.2) and proved that the problem has no solution, because of the odd number of bridges touching the land mass. For a solution, this number (also called the degree of nodes, see Chapter 2.3) must be even for each land mass. A detailed description of the problem is given in [26].

From the mathematical point of view, a graph is a structure consisting of nodes and edges with topological relationships, [31]. In the following, let $G(V, E)$ be a graph of a set of nodes V (also called vertices) and a set of edges E, [65].

[1] Image source: http://www.wikipedia.org/wiki/Seven_Bridges_of_Koenigsberg

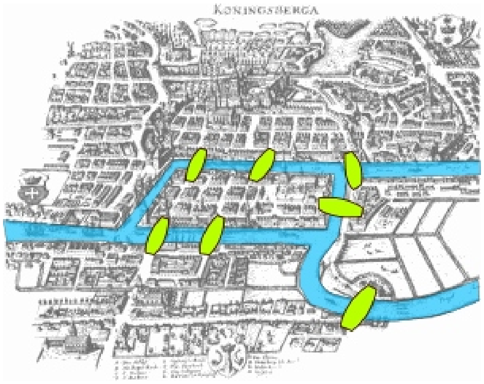


Figure 2.1: Bridges of Königsberg

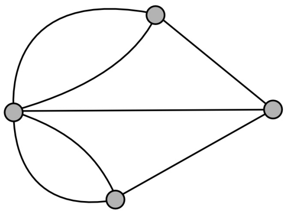


Figure 2.2: Corresponding graph of *The Seven Bridges of Königsberg*

2.2 Graph elements

2.2.1 Main graph elements

Nodes and edges A graph $G(V, E)$ consists of nodes $V = \{v_1, v_2, ..., v_n\}$ and edges $E = \{e_1, e_2, ..., e_m\}$, where edges are connecting the nodes. For an edge $e = (v_i, v_j)$ v_i and v_j are denoted as the corresponding nodes to e. Moreover, v_i and v_j are *incident* with the edge e and v_i and v_j are *adjacent* to each other connected by e. Nodes and edges are shown in Figure 2.3.

Polygon points and isolated points Further graph elements can also be polygon points to model a curved link between nodes geometrically and isolated points to describe objects off the graph (e.g., points of interest within road networks). The mentioned graph elements are shown in Figure 2.3.

Arcs An edge e can be replaced by directed edges for expressing directions. Directed edges are also called arcs. Two inversely oriented arcs are consistent with one edge, [31].

Traverses Traverses are used for representing complex edge-to-edge relations. A traverse t_{ijk} describes a link from a node v_i over a node v_j to a node v_k, [31].

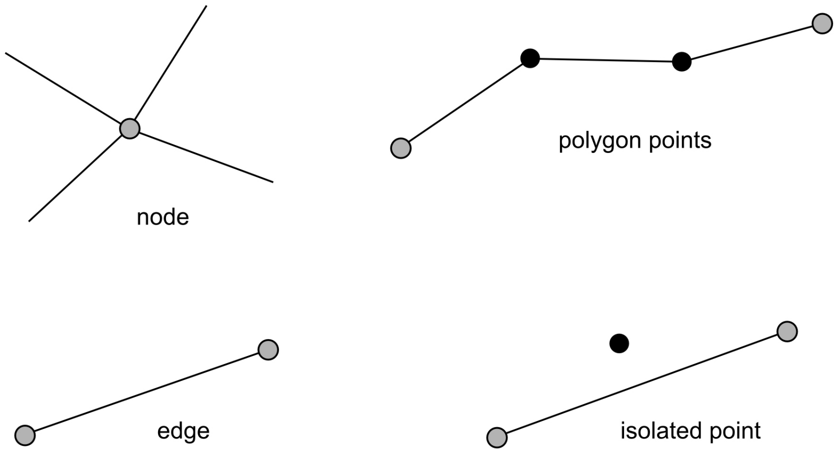


Figure 2.3: Main graph elements

2.2.2 Complex graph elements

Chain A chain is an alternating sequence $(v_s, e_1, v_1, e_2, v_2, ..., e_t, v_t)$ of nodes and edges beginning with the start node v_s and ending with the target node v_t.

Path A path is an alternating sequence $(v_s, a_1, v_1, a_2, v_2, ..., a_t, v_t)$ of nodes and directed edges (arcs) beginning with the start node v_s and ending with the target node v_t. In this work a path consists of different nodes. In shortest path optimization, it is not allowed to visit a node a second time.

Circuit A circuit is a closed chain. A chain is called a simple chain, if all nodes are distinct from each other except $v_s = v_t$.

Cycle A closed path is called a cycle with all nodes are distinct from each other except $v_s = v_t$.

2.3 Properties of graphs

A graph has to fulfill certain criteria which are relevant for the use in route planning, [31]. In the following, specific graph properties are discussed.

Degrees of nodes

The degree of nodes (deg) is defined by the number of edges which are incident with that node. Isolated points have a degree of zero. E.g., a node with one incoming and two outgoing edges has a degree of $deg = 3$.

Definition 1. *The sum of the degrees of nodes of a graph $G(V, E)$ is equal to twice the number of its edges (Euler), [2].*

$$\sum_{i=1}^{n} deg_i = 2m$$

Definition 2. *In any graph $G(V, E)$, the number of nodes of odd degree is even, [2].*

Regular graph If all nodes of a graph $G(V, E)$ have the same degree, $G(V, E)$ is called a *regular* graph.

Finite graphs Such a graph consists of a finite number of nodes V and edges E, which is always true for navigational networks. Throughout this work, the number of nodes is denoted by n and the number of edges by m. The number n is also called the order of the graph and m defines the size of a graph $G(V, E)$, [2].

Simple graphs Simple graphs are graphs without any loops (edges that have only one node) and parallel edges (edges that have the same nodes).

Connected graphs A graph is said to be connected, if a chain/path exists between every pair of nodes.

Definition 3. *A connected graph with n nodes contains at least $n - 1$ edges/arcs, [35].*

Complete graphs A graph is called complete, if the nodes of every pair of nodes V of $G(V, E)$ are adjacent in G (Figure 2.4).

Definition 4. *A complete graph on n nodes has $K_n = n(n-1)/2$ edges.*

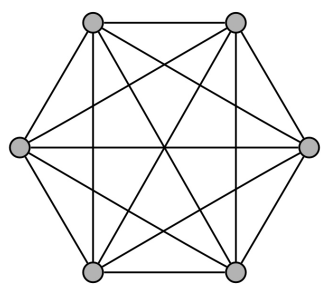


Figure 2.4: Complete graph

Planar graphs In graph theory, a planar graph is a graph that can be mapped to the plane without any edge crossings except in the nodes themselves. Two different graphs are shown in Figure 2.5.

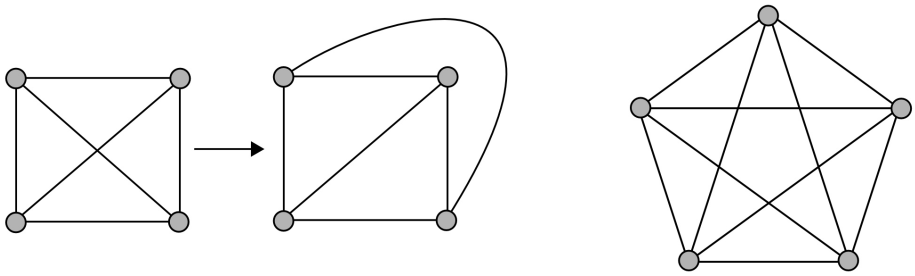


Figure 2.5: Planar graph (left) vs. nonplanar graph (right)

Sparse and dense graphs A strict distinction between sparse and dense graphs does not exist. A graph is said to be dense if the number of edges is close to the maximal number of edges. Otherwise, a graph with relatively few edges is a sparse graph.

Directed graphs A directed graph consists of directed edges (arcs). Edges passable in both directions are split into two arcs. A directed graph is also called a digraph. Figure 2.6 shows a directed graph.

Definition 5. *A directed graph can have at most* $n(n-1)$ *arcs. An undirected graph can have at most* $n(n-1)/2$ *edges.*

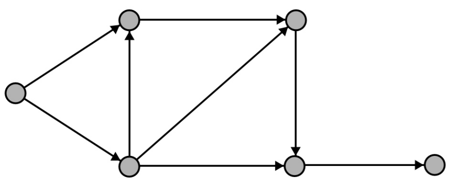


Figure 2.6: Directed graph

Cyclic graphs A digraph is called cyclic, if it contains at least one cycle. Otherwise it is called acyclic. The difference is represented in Figure 2.7.

Valuated graph A valuated graph, [31], associates a label with every arc in the graph. Labels can represent costs or weights. Costs and weights are values for representing the difficulty to pass an arc. Costs valuate arcs in a

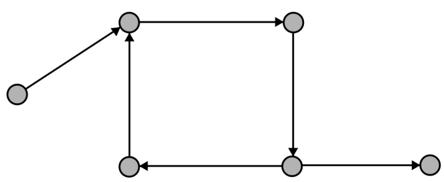


Figure 2.7: A cyclic graph (left) and an acyclic graph (right)

negative sense. High costs mean that it is difficult to pass the arc whereas high weights label arcs easy to pass. In case of costs, a cost value c_{ij} is assigned to an arc a_{ij} from node v_i to node v_j by a cost function $c(a_{ij})$. In Figure 2.8 an example for an valuated graph is given. Valuated graphs are also known as weighted graphs in the literature.

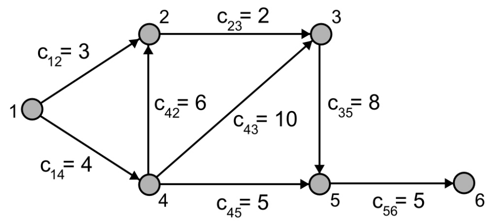


Figure 2.8: Valuated graph

A common cost value for arcs in route optimization is the geometric length. Cost values can also be time, speed, safety aspects or other evaluation criteria, depending on the thematic background of the graph. Such values are no longer a geometric quantity. Therefore, the geometric triangular inequality $c_{ij} + c_{jk} > c_{ik}$ does not always hold. The sum of the cost values c_{ij} from node v_i to node v_j and c_{jk} from node v_j to node v_k does not necessarily exceed the direct cost value c_{ik} from node v_i to node v_k (Figure 2.9). A valuation can also be applied on undirected graphs and the geometric triangular inequality is similar.

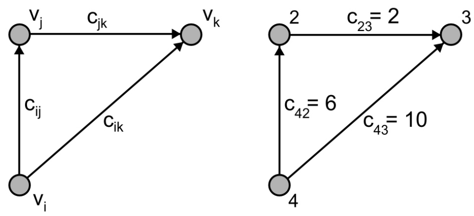


Figure 2.9: Triangular inequality

2.3.1 Geometry

Geometry is split up in metrics and topology.

Metrics

Metrics defines the position of graph elements in a coordinate system. The position is given by coordinates for every graph element. Positions do not play any role in optimization but are necessary for visualization and spatial analysis.

Topology

Network topology describes the relationship of objects in a network. It defines the arrangement of the elements and their interconnections in the graph. In detail, topology describes the relationship between nodes and edges/arcs of a graph. Topology is the basis of combinatorial optimization methods.

2.4 Storage of graphs

2.4.1 Adjacency lists

A graph can be stored using two lists: a node list with node attributes and an arc list (adjacency list) representing the interconnections within the graph between nodes and arcs. For the optimization itself, only the arc list is relevant. In detail, an arc list includes the start node (Startnode-ID) and the end node (Endnode-ID) of the arc. Startnode-ID and Endnode-ID are the identifier of specific nodes. Additionally, the costs c_{ij} are included. An example of an arc list based on the directed valuated graph in Figure 2.8 is given in Table 2.1.

Table 2.1: adjacency list: arc list

Arc-ID	Startnode-ID	Endnode-ID	c_{ij}
1	1	2	3
2	1	4	4
3	2	3	2
4	3	5	8
5	4	2	6
6	4	3	10
7	4	5	5
8	5	6	5

2.4.2 Indexed adjacency lists

A more sophisticated method than using adjacency lists are indexed adjacency lists. A graph is represented by using indexed lists. The basis is formed by a node list. In principle, the node list contains the smallest leaving arc (Table 2.2). In an indexed adjacency arc list for each node its successor is stored (Table 2.3). Indexed adjacency lists require a consecutive numbering of nodes and arcs.

Table 2.2: Indexed adjacency node list

Node-ID	Arc-ID
1	1
2	3
3	4
4	5
5	8
6	9
7	9

Table 2.3: Indexed adjacency arc list

Arc-ID	Endnode-ID	c_{ij}
1	2	3
2	4	4
3	3	2
4	5	8
5	2	6
6	3	10
7	5	5
8	6	5

Table 2.4: Incidence list

Node-ID	Arc-IDs
1	1,2
2	3
3	4
4	5,6,7
5	8

Arcs incident with an identical start node have to be stored sequentially, [31]. Note that node 7 and arc 9 are virtual graph elements to calculate the number of leaving arcs at node 5 and node 6.

Incidence list

Instead of the indexed adjacency node list an incidence list can be applied. This list stores all arcs that leave the referencing node (Table 2.4). With the drawback of more storage consumption, the numbering of nodes and arcs can be arbitrary. The needed arc list has the same structure as the indexed adjacency arc list.

Chapter 3

Shortest path problems

The shortest path problem is a basic problem in route optimization. The methodology of solving shortest path problems forms the basis of any route planning algorithm. The multi-criteria route optimization which will be used for the route planner developed in the present thesis is based on the shortest path problem.

Shortest path problems are combinatorial problems and mean finding the shortest distance between a pair of nodes in a graph. Let v_s denote the start node and v_t the target node or also referred to as destination node, the goal is to find the minimum v_s-v_t path in some respect: In many cases the shortest route is required, but often some other criteria define the 'best' path like time, cost, or safety, [35]. All algorithms described in this work are based on digraphs having directed edges. For an easier and widely-used notation the word 'directed' is omitted and the basic digraph elements are called nodes and edges.

3.1 Classical shortest path problems

The classical shortest path problem (SPP) appears in a broad range of applications and deals with the optimization of one single criterion. The problem can be divided into different cases: There are single-source/single-destination problems, but also single-source/multiple-destination problems or finding an optimal path from each of the multiple sources to each of several destinations, [4].

3.1.1 Single-criterion optimization

Single-criterion optimization is a widely used standard optimization approach and the easiest problem of shortest path computation. In single-criterion optimization one cost value c_{ij} for each edge e_{ij} from node v_i to node v_j exists. This could be length, time, or any other non-negative value depending on the topic of optimization. For calculation, a cost label l_i for each node $i = 1, ..., n$ is set and updated during the algorithm, where the path cost from start node v_s to the i^{th} node is stored. Moreover, the predecessor p_i of each node has to be remembered to get the calculated path by backtracking the predecessors.

In this thesis, it will be shown how additional criteria can be modeled and processed in an adapted algorithm. In the following, single-source questions with non-negative edge lengths are considered.

3.1.2 Types of solution methodologies

Exact algorithms

Such algorithms calculate an exact solution and have the huge advantage of finding and ensuring the optimal solution. However, for large problems they can be computationally intensive. For real-time applications, this could be a crucial fact and leads to the introduction of approximation algorithms or heuristics.

Heuristics

Heuristic search is used to reduce the search space, resulting in a smaller graph and a faster calculation speed. The main disadvantage is that the optimal solution may not be found. A well-known algorithm in using heuristic information is the A*-Algorithm, [31]. In this algorithm, a heuristic value is added to the cost value of each edge. A widely used heuristic in route planning is the straight-line distance, which reflects a goal-directed search. In this heuristic, the edge cost is influenced by the distance from the edge to the destination. In detail, the straight-line distance to the destination is added to the cost value of the edge. Consequently, this additive heuristic value is smaller for edges nearer to the destination than for edges farther away from the destination.

3.2 Dijkstra's shortest path algorithm

A well-known method for solving the shortest path problem is the algorithm of Dijkstra with the condition of a network with non-negative edge lengths. The algorithm determines the shortest path from one node to all other nodes in the network with the condition that at least one path exists between any two nodes, [14, 15, 37]. Dijkstra's algorithm is a label-setting algorithm. Temporary and permanent labels are set to nodes during the search which is explained in detail in Section 3.2.1. The set of nodes with a temporary label is said to be T and the set of nodes with a permanent label is called P. Each node of V in a directed graph $G(V, E)$ is assigned a *temporary distance* *label* l_i, [31]. For single-source/single-destination problems the algorithm terminates if the destination node is reached and removed from the set of temporary labeled nodes T (see Chapter 3.2.1).

3.2.1 Procedure of Dijkstra's algorithm

Initial condition

The algorithm starts at node v_s. The start node v_s is stored in the set of temporary labeled nodes T, the set of permanently labeled nodes P is initialized to be empty. A label l_s of zero is assigned to v_s as the distance from v_s to itself is zero. The labels l_i of all other nodes are set infinity. The initial condition is written as follows:

$$T = \{v_s\}, P = \{\}$$

$$l_s = 0, \quad l_i = \infty, \quad \forall i \neq s.$$

Algorithm

First loop: Starting at node v_s as the only entry in T, v_s is the node with the minimum label. This node with its minimum label is chosen, removed from T, and set permanent by adding this node to the set of permanently labeled nodes P. In the next step, the distances to the neighbors of node v_s are regarded and added to the label l_s. If the computed label is smaller than the existing one of a node, the old label is replaced. This is always true for the first step because of the initial condition where all labels are set to infinity except l_s. The corresponding nodes are marked as visited by adding the nodes to T and the predecessor p of the nodes is set to v_s. The neighbors of v_s now belong to T.

Second loop: Again, regarding all nodes in T, the node with its minimum label l_i is chosen, removed from T, and set permanent by adding this node to P. Then, the distances to the neighbors of node v_i are regarded and added to its label. This value is compared to the current label of the considered nodes (neighbors). If the computed label is smaller than the existing one of a node, the old label is replaced. The predecessor p of the updated nodes is set to v_i. The corresponding nodes are marked as visited by adding the nodes to T, if they do not already belong to T.

Remaining loops: The algorithm starts again by repeating the procedure of the second loop until termination. All nodes entered in P do not change their label any more and stay in P until termination. The algorithm proceeds in iterations and terminates when the candidate list T is empty, [67, 4]. The shortest paths from one node to all other nodes are found. In single-source/single-destination problems the algorithm terminates when the destination node v_t is processed and added to P.

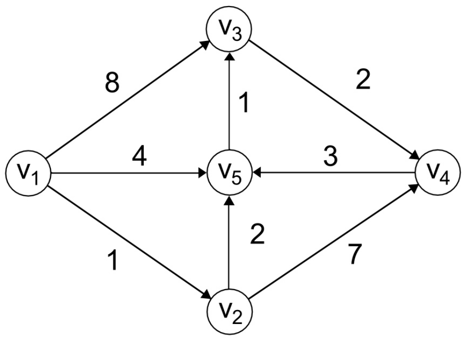


Figure 3.1: Example graph with node numbers and edge costs

Let us consider a practical example based on a given graph in Figure 3.1 which will demonstrate the algorithm more clearly. Figure 3.1 contains the graph with its node numbers and edge cost values. Figure 3.2 shows the iteration steps of Dijksta's algorithm for the graph of Figure 3.1 from the start node v_1 to the destination node v_4. The node numbers in Figure 3.1 are now replaced by the node labels set in the algorithm.

Algorithm steps for the given example

Initial step: First of all, the labels are initialized. For the start node v_1, this initial label is zero, whereas all other nodes are set to infinite.

First loop: Starting at node v_1 as the only entry in T, node v_1 is the node with the minimum label. This node with its minimum label is chosen, removed from T, and set permanent by adding this node to the set of permanently labeled nodes $P = \{v_1\}$. In the next step, the distances to the neighbors of node v_1 are regarded and added to its label. This affects the nodes v_2, v_3 and v_5. The corresponding nodes are marked as visited by adding the nodes to T. The set of temporary labeled nodes now becomes $T = \{v_2, v_3, v_5\}$ with their labels l_2=1, l_3=8, l_5=4, and the predecessors p_2=v_1, p_3=v_1, p_5=v_1.

Second loop: Again, regarding all nodes in T, the node with its minimum label is chosen and set permanent by adding this node to P. In the given example, this is node v_2 with its label l_2=1. P becomes $P = \{v_1, v_2\}$. Then, the distances to the neighbors of node v_2 are regarded and added to its label. This value is compared to the current label of the considered nodes (neighbors). If the computed label is smaller than the existing one of a node, the old label is replaced. At node v_5 the label value of l_5=4 from the first loop is updated to l_5=3. At node v_4 the label is set to l_4=8. The corresponding nodes are marked as visited by adding the nodes to T, if they do not already belong to T. T consists of $T = \{v_3, v_5, v_4\}$ with the labels l_3=8, l_5=3, l_4=8, and the predecessors p_5=v_2, p_3=v_1, p_4=v_2.

Third loop: The next node in T with minimum label is node v_5 with its label l_5=3. The set of permanently labeled nodes now becomes $P = \{v_1, v_2, v_5\}$. Then, the distances to the neighbors of node v_5 are regarded and added to its label. The new label value is l_3=4, T becomes $T = \{v_3, v_4\}$, and the predecessors are set to p_3=v_5, p_4=v_2.

Fourth loop: The next node in T with minimum label is node v_3 with its label l_3=4. The set of permanently labeled nodes now becomes $P = \{v_1, v_2, v_5, v_3\}$. Then, the distances to the neighbors of node v_3 are regarded and added to its label. The new label value is l_4=6, the predecessor becomes p_4=v_3 and only node v_4 is remaining in T ($T = \{v_4\}$).

Fifth loop: The only node left in T is the target node v_4. v_4 is set permanent and the set of permanently labeled nodes becomes $P = \{v_1, v_2, v_5, v_3, v_4\}$ and T is now empty ($T = \{\}$). Since the set of temporary labeled nodes T is empty, the execution of the algorithm terminates.

Finally, backtracking the predecessors gives the shortest path with the node sequence v_1-v_2-v_5-v_3-v_4 and the shortest path problem is solved.

For the treated algorithms, the corresponding pseudocodes are given in the following. Algorithm 1 shows the procedure for calculating the shortest path to all other nodes. Algorithm 2 determines a single path to one target node. A termination criterion is added in the while-loop to define the abort of the algorithm, when the target node is reached. The algorithm terminates, if the target nodes is set permanent and added to P. Another very illustrative description of Dijkstra's algorithm is given in the German book of Gritzmann and Brandenberg, [26].

3.2.2 Dijkstra's algorithm with priority queue

The main influence on runtime is given by the operation of node selection from T. Instead of scanning all nodes, the runtime can be reduced by storing the nodes according to their distance labels which is done by the use of a priority queue as introduced in [31] and [64]. Examples for implementations

Algorithm 1: Dijkstra's shortest path algorithm (modified from [31, 35, 37, 64])

Input: A digraph $G = (V, E)$, edge costs $c \in \mathbb{R}^+$, start node $v_s \in V$.
Output: Shortest paths from v_s to all other nodes $v \in V$, their costs, and predecessors $p(v)$.

```
Initialization:
l(v_s) := 0.
l(v) := ∞ for all v ∈ V \ v_s.
T ← {v_s}. P ← {}.
begin
    while T ≠ {} do
    Find a vertex v_i := v_j ∈ T with min. d(v_j);
    P ← P + {v_i}; T ← T \ {v_i};
    foreach neighbor v_j ∈ N(v_i) do
    if v_j ∈ P then
        continue;
    if v_j ∉ T then
        l(v_j) := l(v_i)+ c_ij; p(v_j) := v_i;
        T ← T + {v_j};
    if (v_j ∈ T) and (l(v_i) + c_ij < d(v_i)) then
        l(v_j) := l(v_i) + c_ij; p(v_j) := v_i;
```

of priority queues are Binary Heap, Fibonacci Heap and Dial's Implementation. For detailed information about these data organization structures the interested reader is referred to [64].

3.2.3 Label-setting algorithms

Label-setting means that node v_i removed from the set of temporary labeled nodes T is the node with minimum label. For label-setting algorithms the condition of non-negative edge lengths must hold. The characteristic of label-setting algorithms is that each node will enter T at most once and has its permanent value at the first time it is removed from T and is added to the set of permanently labeled nodes P. The important part which affects running time is the calculation of the minimum label node in T at each iteration. Several procedures are used in different algorithms, [4]. There are many efficient algorithms for solving single-criterion shortest path problems. One widely used label-setting algorithm is the aforementioned Dijkstra's shortest path algorithm.

Algorithm 2: Dijkstra's shortest path algorithm with one target node (modified from [31, 35, 37, 64])

Input: A digraph $G = (V, E)$, edge costs $c \in \mathbb{R}^+$, start node $v_s \in V$, target node $v_t \in V$
Output: Shortest path from v_s to one target node $v_t \in V$, the cost, and the predecessors $p(v)$.

```
Initialization:
l(v_s) := 0.
l(v) := ∞ for all v ∈ V \ v_s.
T ← {v_s}. P ← {}. p(v) := ∞.
begin
  while T ≠ {} and v_t ∉ P do
    Find a vertex v_i := v_j ∈ T with min. d(v_j);
    P ← P + {v_i}; T ← T \ {v_i};
    foreach neighbor v_j ∈ N(v_i) do
    if v_j ∈ P then
      continue;
    if v_j ∉ T then
      d(v_j) := l(v_i)+ c_ij; p(v_j):=v_i;
      T ← T + {v_j};
    if (v_j ∈ T) and (l(v_i) + c_ij < l(v_i)) then
      l(v_j) := l(v_i) + c_ij; p(v_j) := v_i;
```

3.2.4 Label-correcting algorithms

Dijkstra's algorithm can be easily extended to a label correcting algorithm. In comparison to label-setting algorithms the choice of node v_i removed from T may be less sophisticated. Therefore, a node may enter T multiple times to get its permanent value. One example is the Bellman-Ford method, [37].

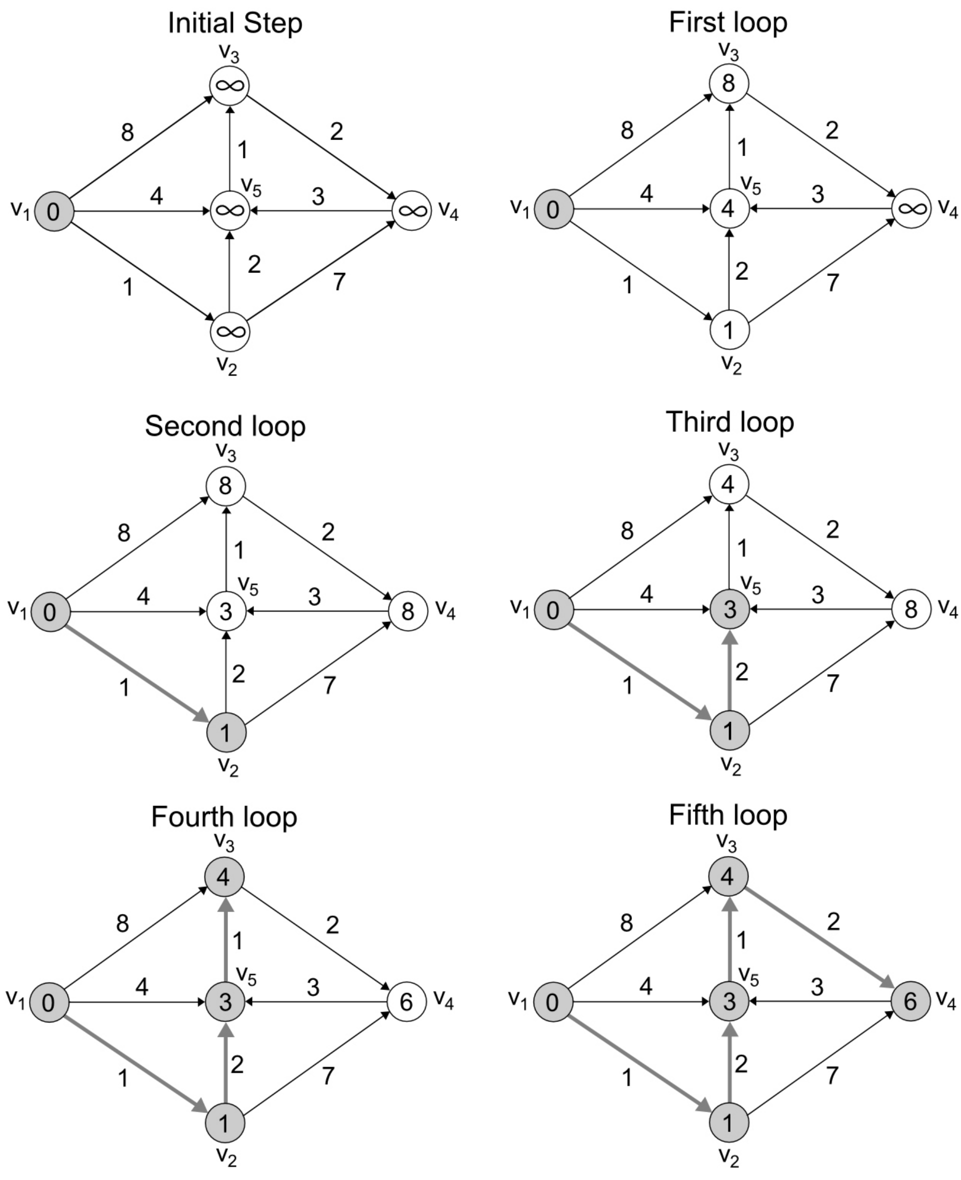


Figure 3.2: Procedure of Dijkstra's algorithm

Chapter 4

Multi-criteria optimization

Based on the conceptual foundations of single-criterion shortest path problems, this chapter deals with multi-criteria optimization methods in general, followed by the multi-criteria shortest path problems in detail. First of all, an overview on optimization with multiple criteria is given, where the reader is introduced in terminologies and methods for solving such optimization problems. Furthermore, the terms of efficiency and Pareto dominance are explained as a basis for determining the 'best' solution among various 'alternatives'. In a continuative part of this chapter, multi-criteria shortest path problems are considered. The goal of this work is providing route planning for people with disabilities and concerning different kinds of needs on a planned route. In this context, the following questions arise: 'How to include different requirements for an optimal route in a route planning algorithm during one optimization process?' and 'How can we provide more than one efficient solution to the user for decision?' An answer is given in this chapter by discussing the methodology of multi-criteria shortest path problems. In such a specific application, single-criterion optimization is not sufficient and a more sophisticated algorithm has to be implemented. Instead of calculating a shortest path with one single criterion (e.g., length), we have to deal with a number of different optimization criteria (e.g., travel time and number of transfers in public transport networks) as we will have later on. Such problems are referred to as *Multi-criteria Shortest Path Problem (MSPP)* with non-negative edge lengths. In literature one can also find the words *Multi-objective Shortest Path Problems (MOSP-Problems)* like in [70].

4.1 Introduction

Let us consider an example for a multi-criteria optimization problem in workaday life. We are interested in buying a new car and we take four car models into account, which we like best: An Audi A3, an Opel Astra, a Seat Altea, and a VW Golf. The cars differ in prize, petrol consumption and

Table 4.1: Specification of car models (data is invented), Example modified from [17]

		Alternatives			
		Audi	Opel	Seat	VW
Criteria	price (Euro)	28.000,-	24.000,-	23.000,-	26.000,-
	petrol consumption (l/100km)	5,5	6,0	5,0	4,5
	power (PS)	140	160	115	105

power. The specifications are given in Table 4.1. Our preference is a cheap and powerful car with low petrol consumption. This is an optimization problem with four alternatives and three criteria. Now, which car is the 'best' choice, when there is not any car that is cheapest, most powerful and has the lowest petrol consumption, [17]? So we have to decide one of the alternatives, considering the conflicting criteria.

4.1.1 Terminology

$\mathcal{X} = \{Audi, Opel, Seat, VW\}$ is called the *feasible set* or the set of *alternatives*. Considering two criteria, the criteria are denoted as f_1 (e.g., price) and f_2 (e.g., petrol consumption) and $f_1(x)$, $f_2(x)$ are *criteria functions* or *objective functions*. The optimization problem can be stated as

$$\min_{x \in \mathcal{X}} (f_1(x), f_2(x)), [17].$$

Let us now consider the car example with the first two criteria price and petrol consumption. In this example the values for $f_1(x)$ and $f_2(x)$ are: $f_1(x) = \{28.000; 24.000; 23.000; 26.000\}$ and $f_2(x) = \{5,5; 6,0; 5,0; 4,5\}$. In Figure 4.1 the criterion values of each solution are mapped.

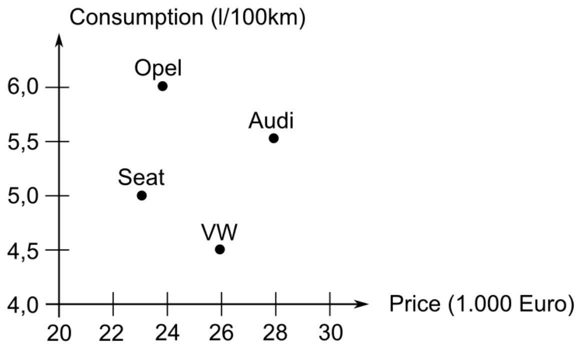


Figure 4.1: Criterion values in the bi-criteria car example, modified from [17]

4.2 Efficiency and dominance

4.2.1 Pareto optimality

As mentioned above, by dealing with multi-criteria optimization, more than one efficient solution does exist. In this context, the Italian economist and sociologist *Vilfredo Pareto* (1848-1923) has to be mentioned. He introduced efficient solutions in the sense of *Pareto optima* as follows, [70]:

Definition 6. *A solution is Pareto optimal if it cannot be improved upon from one criterion's perspective without making another criterion worse off.*

The challenge now is to find these Pareto optimal solutions and select the most attractive ones, [46]. The problem of finding all Pareto optimal solutions results in a (weakly) $\mathcal{NP}$-hard (non-deterministic polynomial-time) problem in general. Such problems can not be solved in polynomial time.

Before going into detail in solving multi-criteria optimization problems, the concept of dominance and non-dominance has to be defined. The technique of dominance reduces the size of the search-space by eliminating inefficient solutions and is much used in the field of multi-criteria optimization, [17, 69, 38].

4.2.2 Dominance

Let f_1, f_2,...,f_k be a number of k criteria, $f_1(x)$, $f_2(x)$,...,$f_k(x)$ the corresponding criteria functions and $f(x)$ be the *criteria function mapping* with $f(x) = \{f_1(x), f_2(x), ..., f_k(x)\}$. Then the definition of dominance can be written as:

Definition 7 (Ehrgott, [17]). *A feasible solution $\hat{x} \in \mathcal{X}$ is called efficient or Pareto optimal, if and only if it does not exist another $x \in \mathcal{X}$ such that $f(x) < f(\hat{x})$. If $\hat{x}$ is efficient, $f(\hat{x})$ is called non-dominated. If $x^1, x^2 \in \mathcal{X}$ and $f(x^1) < f(x^2)$ we say x^1 dominates x^2 and $f(x^1)$ dominates $f(x^2)$. The set of all efficient solutions $\hat{x} \in \mathcal{X}$ is called efficient set or the set of all Pareto optimal solutions.*

In case of two criteria ($k = 2$) in a discrete optimization problem, $f(x)$ is a set of $f(x) = \{f_1(x), f_2(x)\}$. Then Definition 6 comes up with:

Definition 8. *A solution is Pareto optimal if it cannot be improved upon from one criterion's perspective without making the other criterion worse off.*

Concerning the car example, in Figure 4.1, we can see that $Seat = (23.000, 5)$ and $VW = (26.000, 4, 5)$ are the efficient choices. The following formulation shall illustrate the Definition 8 more clearly: Regarding the VW, the solution cannot be made cheaper without increasing the value for petrol consumption. Thus, VW is a Pareto optimal solution. Inversely, the solution for the Seat cannot be made more fuel efficient without increasing the price. Therefore, Seat is another Pareto optimal solution.

Concerning two criteria in Definition 7, the Definition can be written as:

Definition 9. *A feasible solution $\hat{x} \in \mathcal{X}$ is called efficient or Pareto optimal, if and only if it does not exist another $x \in \mathcal{X}$ such that $f_1(x) < f_1(\hat{x})$ and $f_2(x) < f_2(\hat{x})$.*

Concerning the car example, for Seat and VW, there does not exist another solution that is both cheaper and more fuel efficient, [17].

4.2.3 Strict dominance

In Figure 4.2 Pareto optimal solutions are shown. Solutions x^3, x^4, x^5, x^6, x^7 are denoted as strict Pareto optimal solutions, [71]. Thus, the definition of strict Pareto optima is written as follows:

Definition 10. *A solution is strict Pareto optimal, if and only if does not exist another $x \in \mathcal{X}$ such that $f(x) \leq f(\hat{x})$ with at least one strict inequality. If $\hat{x}$ is efficient, $f(\hat{x})$ is called non-dominated. If $x^1, x^2 \in \mathcal{X}$ and $f(x^1) \leq f(x^2)$ we say x^1 dominates x^2 and $f(x^1)$ dominates $f(x^2)$. The set $\hat{x} \in \mathcal{X}$ is called the set of all strict Pareto optimal solutions.*

In other literature the solutions in Definition 7 are often called weak Pareto optimal solutions as in [71].

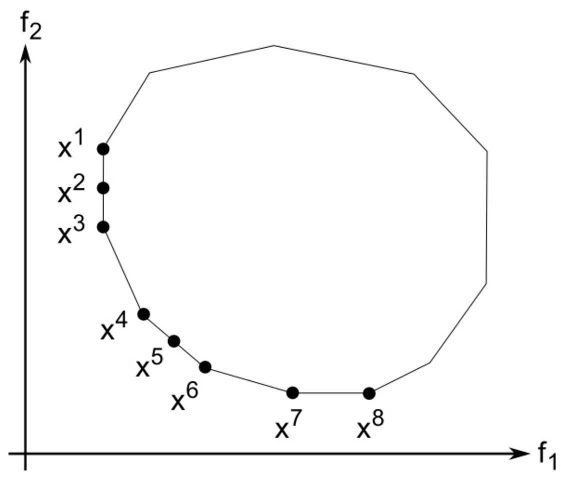


Figure 4.2: Pareto optimal solutions, [71]

4.3 Definition of optimality

To solve multi-criteria problems in polynomial runtime, approximation methods have to be applied. Approximation means the determination of a subset of all Pareto optimal solutions, since the determination of all Pareto optimal solutions cannot be solved in polynomial time. The following different methods for reducing the number of solutions and selecting efficient solutions can be distinguished.

4.3.1 Scalarization methods

Scalarization techniques are methods to transform the multi-criteria problem into a single-criterion problem.

Weighted sum method The *weighted sum method*, also named *weighted sum scalarization*, reduces the multi-criteria problem to a single-criteria problem by applying positive weights λ_i on k criteria f_i. A summation of the weighted criteria gives a scalar value, which will be minimized. The minimum sum can be written as, [17]:

$$\underset{x \in \mathcal{X}}{min} \sum_{i=1}^{k} \lambda_i f_i(x)$$

The parameters $\lambda_1, ..., \lambda_k$ express the difference in importance of the criteria. The optimization output is one single solution and is a weakly efficient solution, [43, 17]. This means that some interesting (and maybe better) alternatives may get lost. To overcome this problem, other scalarization methods (like the *ϵ-Constraint Method* as one example) are more flexible. Moreover, non-scalarization methods introduced in Chapter 4.3.2 treat all criteria simultaneously and separately.

The interested reader is referred to Ehrgott, [17]. In his book 'Multicriteria Optimization' he discusses different methods in detail and introduces some more methods like the *ϵ-Constraint Method*, the *Hybrid Method*, the *Elastic Constraint Method* and a few others, which are out of scope of this thesis.

4.3.2 Non-scalarization methods

Lexicographic order

In some situations a hierarchical order among the criteria is given. As one example the criterion vectors $f_1(x), f_2(x)$ can be compared lexicographically, [67]. Let us consider again our example concerning cars. The ranking of the criteria might be price before petrol consumption. In that case, the lexicographic order can be used. The minimization can be mathematically written as, [17]:

$$\min_{x \in \mathcal{X}} {}_{lex}(f_1(x), f_2(x))$$

Definition 11. *The lexicographical minimum for a bi-criteria case is defined by*

$$(f_1, f_2) <_{lex} (f_1', f_2') \ if \ (f_1 < f_1') \ or \ (f_1 = f_1' \ and \ f_2 < f_2')$$

This definition is easily extendable for more than two criteria. Considering our car example shown in Figure 4.1, we should choose the $Seat = (23.000, 5)$ among the Pareto optimal solutions (Seat and VW). Due to the ranking of the criteria this is the optimal solution (the cheapest). Note that by choosing the lexicographical minimum, in this example, even a very good value for petrol consumption cannot compensate a high price, [17].

In this thesis, a non-scalarization method is chosen to suggest more than one efficient solution to the user. Different efficient solutions are provided to the user in pre-trip training (see Chapter 8.1.9).

4.4 Multi-criteria shortest path problems

Multi-criteria shortest path problems appear in many applications like computer networks, route planning, scheduling, artificial intelligence, etc. MOSP-Problems can be divided into four categories, [70]:

- Number of optimization criteria
- Type of problem
  (lexicographic solution, etc.)
- Solution method
  (label-setting, label-correcting, and others)
- Data model
  (types of network models like time-expanded or time-dependent (see Chapter 6), etc.)

Applications which deal with the MOSP-Problem can be divided into two frequently used domains: computer networks and transportation, [70]. In comparison to single-criterion optimization, where the cost function is a scalar, in multi-criteria optimization a k-dimensional vector for each edge represents heterogeneous costs (e.g., travel time and numbers of transfers), [43]. To make it clearly, a bi-criteria example with two-dimensional cost vectors is given in Figure 4.3, where a digraph consisting of nodes and directed edges is shown. The goal is now to consider these different conflicting costs simultaneously in the algorithm. Especially, if these costs are equally important, it is not clear how to define the shortest path anymore. To solve a multi-criteria problem, Pareto-optimal paths are introduced in the next section.

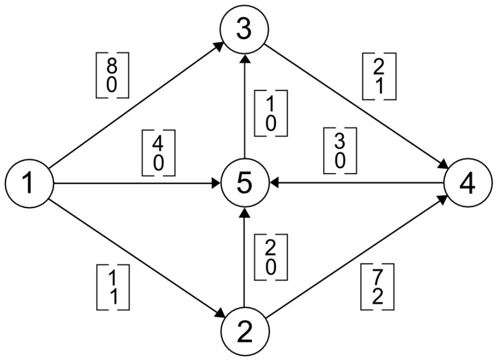


Figure 4.3: Bi-criteria cost vectors for edges

4.4.1 Solutions of multi-criteria shortest path problems

The calculation of efficient paths is based on a digraph $G = (V, E)$ consisting of a set V of nodes and a set E of edges. The graph has k-dimensional cost vectors $\vec{c}(e) = (c_1(e), ..., c_k(e))$ with k different edge costs for each edge $e \in E$ and k-dimensional label vectors $\vec{l}(v) = (l_1(v), ..., l_k(v))$ for each node $v \in V$. The whole cost for a path P is defined as following:

Definition 12. *The summation of costs c for a path P with m_p edges e is a vector sum, which is defined to be a component-by-component adding, [64]:*

$$\vec{c}(P) = \sum_{i=1}^{m_p} \vec{c}(e_i) = \left(\sum_{i=1}^{m_p} c_1(e_i), \sum_{i=1}^{m_p} c_2(e_i), ..., \sum_{i=1}^{m_p} c_k(e_i) \right)$$

Concerning Pareto optimal paths, some theoretical considerations have to be made:

Definition 13. *If a path P in $G(V, E)$ is a Pareto optimal path, then every sub-path of P is also a Pareto optimal path. Proof is given in [41] and [17].*

Definition 14. *Let P_1, P_2 and P_3 be three different Pareto optimal paths. If P_1 dominates P_2 and P_2 dominates P_3, it follows that P_1 dominates P_3, [41].*

Figure 4.4 shows all Pareto optimal solutions (displayed in red) for the example graph in Figure 4.3 from v_1 to all other nodes based on Definition 7. Labels in black show all other possible paths from node v_1 to each node. In

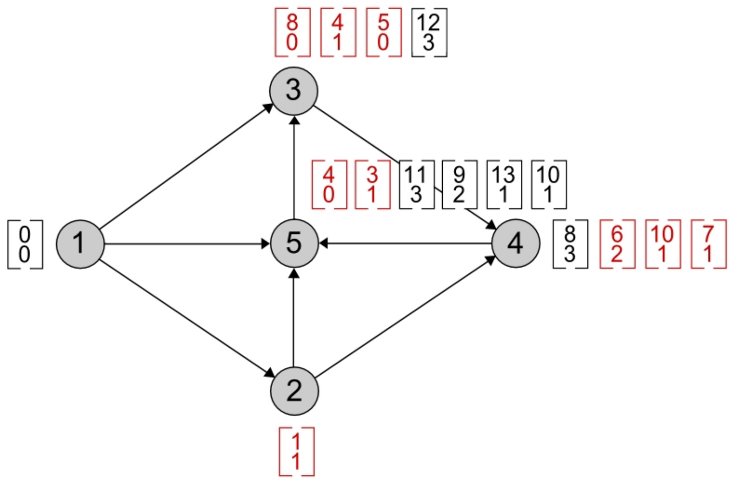


Figure 4.4: All Pareto optimal solutions of the example graph in Figure 4.3

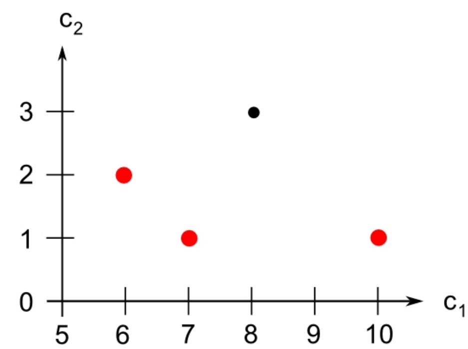


Figure 4.5: Illustration of Pareto optimal solutions of node 4 in Figure 4.4

Figure 4.5 the Pareto optimal solutions of node 4 are illustrated (large red points). Solutions $(6, 2)$ and $(7, 1)$ are strict Pareto optimal solutions.

The number of Pareto optimal paths increases with the number of nodes in the graph. In worst case the number of Pareto optimal paths is exponential in the number of nodes, [17, 64]. This is a crucial fact for the runtime of the algorithm. Therefore, approximation methods given in Chapter 5.2 and Chapter 5.3 are applied.

4.5 Dijkstra's algorithm in multi-criteria optimization

Hansen, [29], and Martins, [41], were the first, who introduced the label-setting algorithm of Dijkstra in multi-criteria optimization. For finding Pareto optimal paths an extended Dijkstra's shortest path algorithm can be used, with the restriction, that no negative costs exist in the graph, [41]. Otherwise, a label-correcting algorithm must be applied. Proof is given in [17]. In the following the use of Dijkstra's algorithm for multi-criteria optimization is described in detail.

4.5.1 Extended Dijkstra's algorithm

Dijkstra's algorithm can be used for calculating all Pareto optimal paths as well as for determining a subset of all Pareto optimal paths or only a single Pareto optimal solution. Compared to single-criterion optimization the following differences appear in the algorithm:

- Edge cost vectors
  Cost values in single-criterion optimization are replaced by cost-vectors.

- Node label vectors
  Edge cost vectors lead to node label vectors in the algorithm. The label of each node in single-criterion optimization is replaced by one or more node label vectors in multi-criteria optimization.

- Approximation methods
  Methods for selecting a set of Pareto optimal paths are used to reduce the number of Pareto optimal solutions for a proper runtime of the algorithm. Different methods are explained in Chapter 5.3.

- Restrictions
  Predefined limits and upper bounds for the vector-components reduce the amount of Pareto optimal solutions (see Chapter 5.3.2).

- Termination
  The termination criterion differs from single-criterion optimization. The algorithm does not terminate, if the first label at the destination node is made permanent.

The algorithm is extendable to further criteria only by the expansion of the edge cost vectors and node label vectors.

Calculation of ALL Pareto optimal paths

The algorithm explained afterwards was introduced by Martins in 1984, [41], and is a multi-criteria label-setting algorithm which determines all Pareto optimal paths. This algorithm is shown in detail in Algorithm 3. For multi-criteria optimization the node label vectors have to be expanded for solving this problem. In the algorithm, the lexicographical order is used for selecting the labels and the definition of dominance from Definition 7 is used for comparing, [41, 19].

Let v_i be a node of $G = (V, E)$. The node label vector of $\vec{l}(v_i)$ is a $k + 3$ tuple, where k is the number of costs. It writes $\vec{l}(v_i) = (c_1, ..., c_k, p_j, z, u)$ with $p_j \neq v_i$ is a node of $G = (V, E)$, z is the label number of node p_j, and u is the label number at node v_i. In other words, a label vector consists of a k-dimensional cost component, a node predecessor, identifying the node from which the label was created, a further label indicating from which of the several labels of the predecessor it was computed, and a label number at the current node. Further, let T denote a list of temporary labels, which is kept in a predefined (e.g. lexicographic) order, and a list P of permanent labels, which will store efficient paths. Note that labels are set permanent instead of nodes, [17].

Algorithm 3: Dijkstra's shortest path algorithm with multi-criteria extension - Calculation of all Pareto optimal paths

---

Input: A digraph $G = (V, E)$, edge cost vectors $\vec{c} = (c_1, ..., c_k, 0, 0, 0) \in \mathbb{R}^+$, start node $v_s \in V$, target node $v_t \in V$.

Output: All Pareto optimal paths from v_s to the target node v_t and all other nodes in the graph $G = (V, E)$ and the label vectors $\vec{l} = (c_1, ..., c_k, p_j, z, u)$ stored in the list of permanent labels P.

Initialization:
$\vec{l}(v_s) := (0, ..., 0, 0, 0, 1)$.
$\vec{l}(v) := \infty$ for all $v \in V \setminus v_s$.
$T \leftarrow \{\vec{l}(v_s)\}$. $P \leftarrow \{\}$.
begin
  while $T \neq \{\}$ **do**
  Find a label vector $\vec{l} \in T$ of node v_i with min_{lex} $\vec{l}(v_i)$;
  P $\leftarrow$ P + $\{\vec{l}(v_i)\}$; T $\leftarrow$ T $\setminus$ $\{\vec{l}(v_i)\}$;
  foreach *neighbor* $v_j \in N(v_i)$ **do**
  $\vec{l}(v_j) := \vec{l}(v_i) + \vec{c_{ij}}$; $l_{p_j}(v_j) := v_i$; $l_z(v_j) := l_u(v_i)$; set $l_u(v_j)$;
  T $\leftarrow$ T + $\{\vec{l}(v_j)\}$;
  if $\vec{l}(v_j)$ *is dominated by an existing label* $\in TorP$ *at node* v_j **then**
    T $\leftarrow$ T $\setminus$ $\{\vec{l}(v_j)\}$;
  if $\vec{l}(v_j)$ *dominates an existing label* $\vec{l}(v_x)$ *at node* v_j **then**
    T $\leftarrow$ T $\setminus$ $\{\vec{l}(v_x)\}$;

Based on the graph of Figure 4.3 an example of a modified Dijkstra's algorithm calculating all Pareto optimal paths is given. The iteration steps of the algorithm and the label vectors are described below and are shown in Figure 4.6 and 4.7. Dominated labels are deleted and crossed out.

Initial step: The start node $v_s = v_1$ is initialized with $\vec{l}(v_s) = (0,0,0,0,1)$ and enters the temporary list of labels T, other nodes are initialized with infinity.

First step: The label vector of the start node as the only entry in T is set permanent. Then the label vectors of the neighbor nodes become updated. This yields the new label vectors $(1,1,1,1,1)$ for node 2, $(8,0,1,1,1)$ for node 3, and $(4,0,1,1,1)$ for node 5. These label vectors are added to T.

Second step: As the label vector of node 2 is lexicographically the smallest vector in T, it is chosen next and set permanent. Based on this label the neighbors are updated with $(8,3,2,1,1)$ at node 4 and the additional label vector $(3,1,2,1,2)$ at node 5 which are added to T. Since both labels at node 5 do not dominate (see Definition 7) each other, they both stay in T.

Third step: The next label which is lexicographically smallest in T is the label $(3,1,2,1,2)$ at node 5. This label is set permanent and its neighbors are updated which forms the new label vector $(4,1,5,2,2)$ at node 3. The new label vector is added to T because it is non-dominated.

Fourth step: Now, the label vector $(4,0,1,1,1)$ at node 5 is lexicographically the smallest in T and is made permanent. Updating its neighbor node 3 yields an additional label vector $(5,0,5,1,3)$ at node 3. Also non-dominated, this label vector joins T.

Fifth step: The next label set permanent is the label $(4,1,5,2,2)$ at node 3. Updating the neighbors of node 3 yields the label vector $(6,2,3,2,2)$ at node 4. This label dominates the existing label $(8,3,2,1,1)$ at node 4. Therefore, label vector $(8,3,2,1,1)$ becomes deleted from T and the the label vector $(6,2,3,2,2)$ enters T.

Sixth step: Since the label vector $(5,0,5,1,3)$ at node 3 is lexicographically the smallest in T, it is set permanent and its neighbors become updated. This leads to the new label vector $(7,1,3,3,3)$ at node 4 and this non-dominated label vector enters T.

Seventh step: The next label which is lexicographically the smallest in T, is label $(6,2,3,2,2)$ at node 4 and is made permanent. The update process yields label $(9,2,4,2,3)$ at node 5. This label is dominated by the labels $(4,0,1,1,1)$ and $(3,1,2,1,2)$ and therefore it is instantly deleted.

Eighth step: The next label set permanent is the label $(7, 1, 3, 3, 3)$ at node 4. Updating the neighbors of node 3 yields the label vector $(10, 1, 4, 3, 4)$ at node 5, which is also dominated and deleted.

Ninth step: From the remaining labels in T, the label vector $(8, 0, 1, 1, 1)$ at node 3 is chosen and set permanent. From updating the neighbors, label $(10, 1, 3, 1, 4)$ at node 4 is created and enters T.

Tenth and final step: The label which is set permanent, is the recently created label $(10, 1, 3, 1, 4)$ at node 4, as it is the only entry in T. Updating its neighbor yields the label $(13, 1, 4, 4, 5)$ at node 5. This label is again dominated by existing labels and is therefore deleted. Since the list of temporary label vectors T is empty, the algorithm terminates. All Pareto optimal paths from v_s to all nodes are found and represented by their label vectors.

Finally, after applying this algorithm, the Pareto optimal paths of each node are found by backtracking the predecessor labels using p_j and z. For the given example of node number 4 all Pareto optimal paths with their node sequence $(1, 2, 5, 3, 4)$, $(1, 5, 3, 4)$, and $(1, 3, 4)$ and their cost pairs $(6, 2)$, $(7, 1)$, and $(10, 1)$ are obtained.

Now, the main difference between single-criterion optimization or finding a single approximate solution and generating multiple solutions must be mentioned. When calculating a single solution, the algorithm can be stopped once a label at node v_t is set permanent. This is not possible in multi-objective label-setting algorithms due to the existence of more than one efficient path between a pair of nodes. For a full solution, the algorithm will only stop if all efficient paths to all nodes are found. Otherwise an approximation can be done by defining a termination criterion. This could be either a spatial criterion in the search graph or a runtime restriction on the algorithm. As already mentioned, the calculation of all Pareto optimal paths can not be solved in polynomial time. Therefore, the calculation of a SUBSET of all Pareto optimal is reasonable in practice, which will be treated in the next chapter.

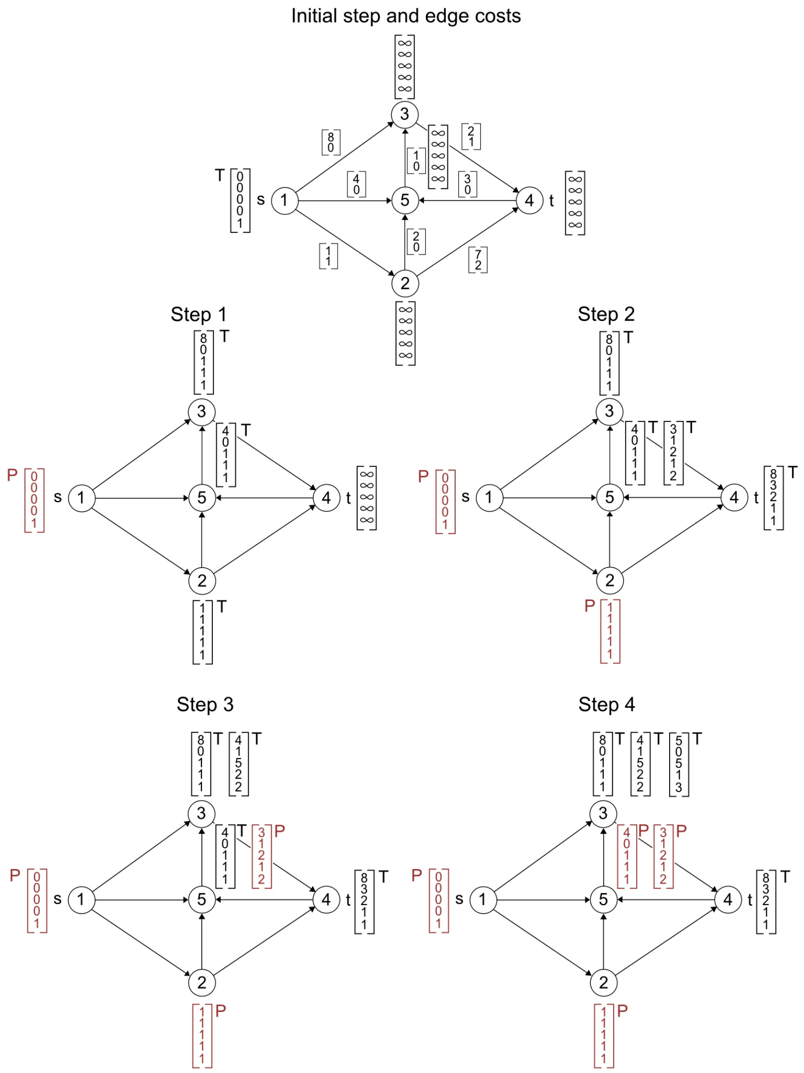


Figure 4.6: Steps 1-4 of modified Dijkstra's algorithm for calculating all Pareto optimal paths, having two different criteria

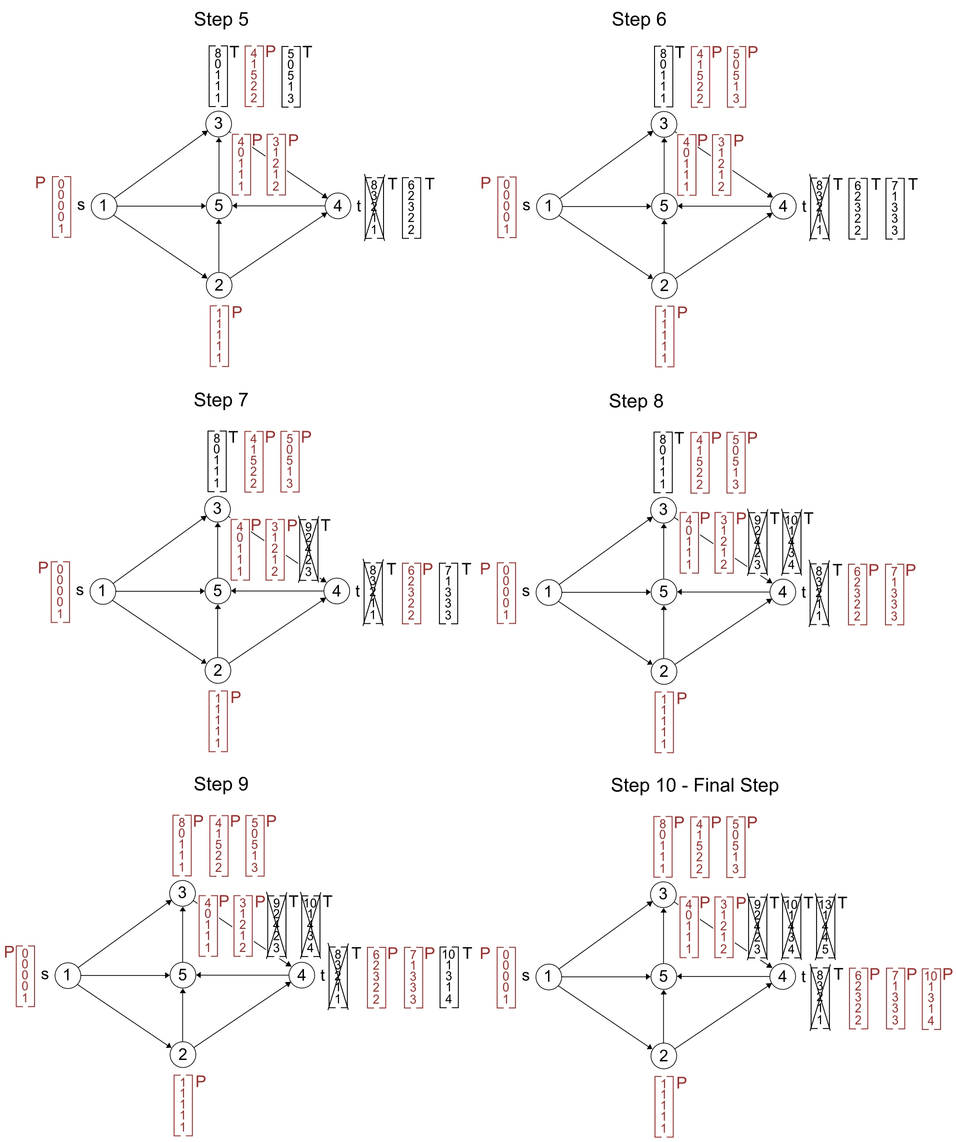


Figure 4.7: Steps 5-10 of modified Dijkstra's algorithm for calculating all Pareto optimal paths, having two different criteria

Chapter 5

Concepts of multi-criteria optimization for route planning

This chapter introduces different concepts of multi-criteria optimization for route planning, based on the theoretical definitions in Chapter 4. In the practical part of this work (Part II), a selection of these multi-criteria optimization concepts are implemented in a pre-trip route planner, especially developed for people with disabilities.

5.1 Overview

5.1.1 Strict Pareto optimal paths

In route planning, strict Pareto optimal solutions are focused, because weak Pareto optimal solutions are of less interest of a user. Let us assume the two criteria time and number of transfers, as often used in route planning. Let $(7, 1)$ be a strict Pareto optimal solution, where the first value represents a journey duration of 7 minutes, and the second value represents one transfer during the journey. Let $(10, 1)$ be a weak Pareto optimal solution of the same problem, then nobody would prefer the weak solution $(10, 1)$, since it took more time (10 minutes instead of 7 minutes) with the same number of transfers.

5.1.2 Complexities of multi-criteria optimization

In multi-criteria optimization, three different outputs and complexities of solutions can be distinguished. These three variants are, [64]:

1. ALL Pareto optimal solutions (exponential runtime, full solution)

2. A SUBSET of all Pareto optimal solutions (polynomial runtime, approximate solution)
3. A SINGLE Pareto optimal solution (polynomial runtime, approximate solution)

While the first was treated in Chapter 4, variants two and three are the main topic of this chapter. As already mentioned, variants two and three are necessary, since the determination of all Pareto optimal paths can not be solved in polynomial time.

In Figure 5.1, approximation methods for solving these two variants (single solutions and subset of solutions) are listed and explained in further sections. The description of the variants is based on bi-criteria optimization, but all methods can be easily extended to an optimization with multiple criteria.

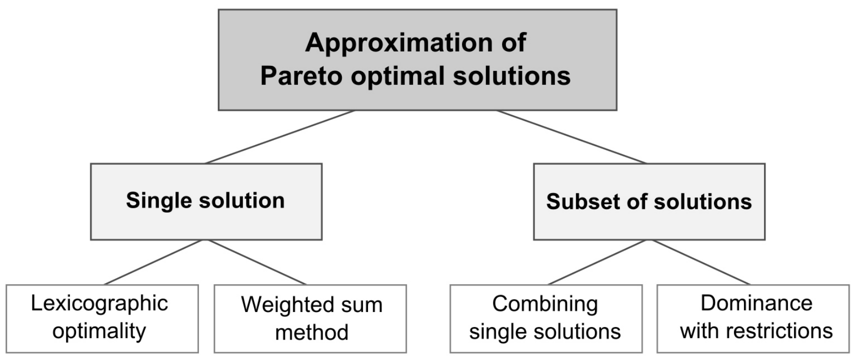


Figure 5.1: Approximation methods of multi-criteria optimization for route planning

5.2 Single Pareto optimal solutions

5.2.1 Bi-criteria optimization with lexicographic optimality

In a bi-criteria optimization problem, the two criteria f_1 = 'time' and f_2 = 'number of transfers' are considered. Applying the lexicographic optimality (see Chapter 4.3.2), the priority is set on the time and thus, the fastest path is computed. An example algorithm based on the graph in Figure 4.3 is given in Figure 5.3. The lexicographical minimum concerning f_1 = 'time' and f_2 = 'number of transfers' for target node 4 is solution $(6, 2)$ with (6 minutes, 2 transfers) as it is shown in Figure 5.3. Comparing this solution with all Pareto optimal solutions from the final step of Figure 4.7, note that the solutions with its cost vector $(7, 1)$ and $(10, 1)$ got lost due to the approximation with the lexicographic minimum.

5.2.2 Bi-criteria optimization using the weighted sum method

As explained in Chapter 4.3.1 the weighted sum method is a linear combination of the optimization criteria using positive weights. Applying this method on the bi-criteria optimization problem with the optimization criteria f_1 and f_2 the minimization of the linear combination is stated as:

$$\underset{x\in\mathcal{X}}{min}\sum_{i=1}^{2}\lambda_i f_i(x) = \lambda_1 f_1(x) + \lambda_2 f_2(x)$$

λ_1 and λ_2 are arbitrary values describing the importance of the criteria. E.g., if $f_1(x)$ is twice important than $f_2(x)$, then $\lambda_1 = 2$ and $\lambda_2 = 1$ and the linear combination can be defined as:

$$\underset{x\in\mathcal{X}}{min}(\lambda_1 f_1(x) + \lambda_2 f_2(x)) = \underset{x\in\mathcal{X}}{min}(2f_1(x) + 1f_2(x))$$

Note that this supposes that the criterion values of both $f_1(x)$ and $f_2(x)$ have the same range. Otherwise, λ_1 and λ_2 must be chosen in relation to the different ranges.

5.3 Subsets of Pareto optimal solutions

5.3.1 Subsets by combining single solutions

Bi-criteria optimization with lexicographic optimality using different orders of the criteria

Additionally to solution $(6, 2)$ from the example presented in section 5.2.1, we can easily gain another strict Pareto optimum by changing the order of the criteria to f_1 = 'number of transfers' and f_2 = 'time' and applying the algorithm with the lexicographic optimality a second time. This is possible in practice, since the runtime of the algorithm for one lexicographic solution is small. Now, in case of switched order, solution $(1, 7)$ which means (1 transfer, 7 minutes) remains at target node 4 as the only lexicographic minimal solution. The corresponding algorithm steps are shown in Figure 5.4. Note that the order of criteria in all edge cost vectors and label vectors is changed to (number of transfers, time). Compared to the algorithm steps in Figure 5.3, the sequence of how the labels are created differs. At step 2 the label of node 5 with $(0, 4)$ is set permanent instead of label of node 2 in the example from Figure 5.3. Therefore, label $(0, 5)$ at node 3 is already created in step 2 and is not dominated. This label is the predecessor of label $(1, 7)$ at the destination which therefore remains.

In that case, the solution $(1, 7)$ is the solution with the smallest number of transfers among all Pareto optimal solution, and among all solutions with the same number of transfers, the one with the smallest time component. It must be mentioned, that in worst case the time component can be extremely high, since a solution with a slightly higher number of transfers does not dominate a solution with a very high time value. E.g., there may also remain a solution $(1, 55)$ as the lexicographic minimum, if this is the only solution with minimum number of transfers. This circumstance leads to the introduction of restrictions on criteria in the next section.

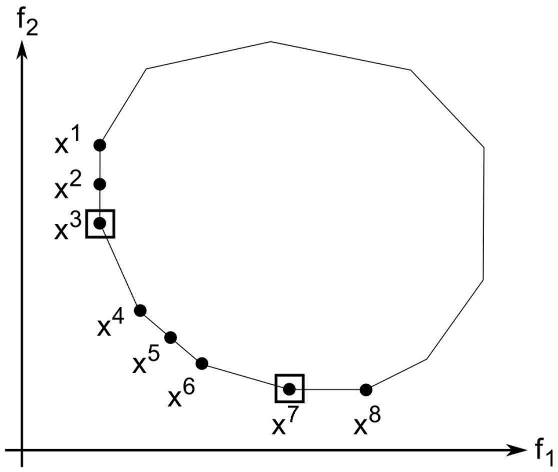


Figure 5.2: Lexicographic minimal solutions

Generally speaking, applying the lexicographic minimum with different orders of the criteria in bi-criteria optimization, the outcome are the two strict Pareto optimal solutions with the minimum value of f_1 in one case and the minimum value of f_2 in the other case. These solutions are pointed out in Figure 5.2. All other strict Pareto optima in between these two solutions cannot be found by applying this method.

The weighted sum method with different weight combinations

Subsets of Pareto optimal paths can be generated by applying the algorithm with various weight combinations. E.g., one solution for $\lambda_1 = 2$ and $\lambda_2 = 1$, another solution for $\lambda_1 = 1$ and $\lambda_2 = 2$, and so on.

5.3.2 Subsets by dominance with restrictions

Bi-criteria optimization with restriction on the criteria

Based on the method of determining all Pareto optimal paths, one method for calculating a subset of Pareto optimal paths is the definition of restrictions on the criteria. This reduces the number of Pareto optimal paths in the nodes and speeds up the algorithm. In this case only Pareto optimal paths P that satisfy prescribed restrictions on some criteria are located.

The restrictions are defined by the inequation $c_i(P) \leq \tau_i$, [28], for the i^{th} criterion of the path, where τ_i denotes the upper bound of the i^{th} criterion. This extension of the MOSP-Problem is referred to as Multi-objective Shortest Path Problem with restriction or shortly named restricted shortest path problem in general. In the algorithm, the components of the label vectors are checked against the predefined inequations. Non-dominated labels are deleted, if they exceed the bound values of at least one criterion.

Bi-criteria optimization with restriction on the criteria by additive values

Again, the two criteria time and number of transfers are considered, but their order is changed to f_1 = 'number of transfers' and f_2 = 'time', as the number of transfers has a higher priority. The label dominates if it has less transfers, but its time component is max. 10 minutes bigger. This value represents an approximate transfer time necessary for a wheelchair user to change vehicles and can be changed for other user groups like blind people, as their transfer time is smaller. Dominance can be written as follows:
A *feasible solution* $\hat{x} \in \mathcal{X}$ is called *Pareto optimal*, if and only if it *does not exist another* $x \in \mathcal{X}$ such that $f_1(x) < f_1(\hat{x})$ and $f_2(x) < (f_2(\hat{x}) + \delta)$. In our above example $\delta = 10$.

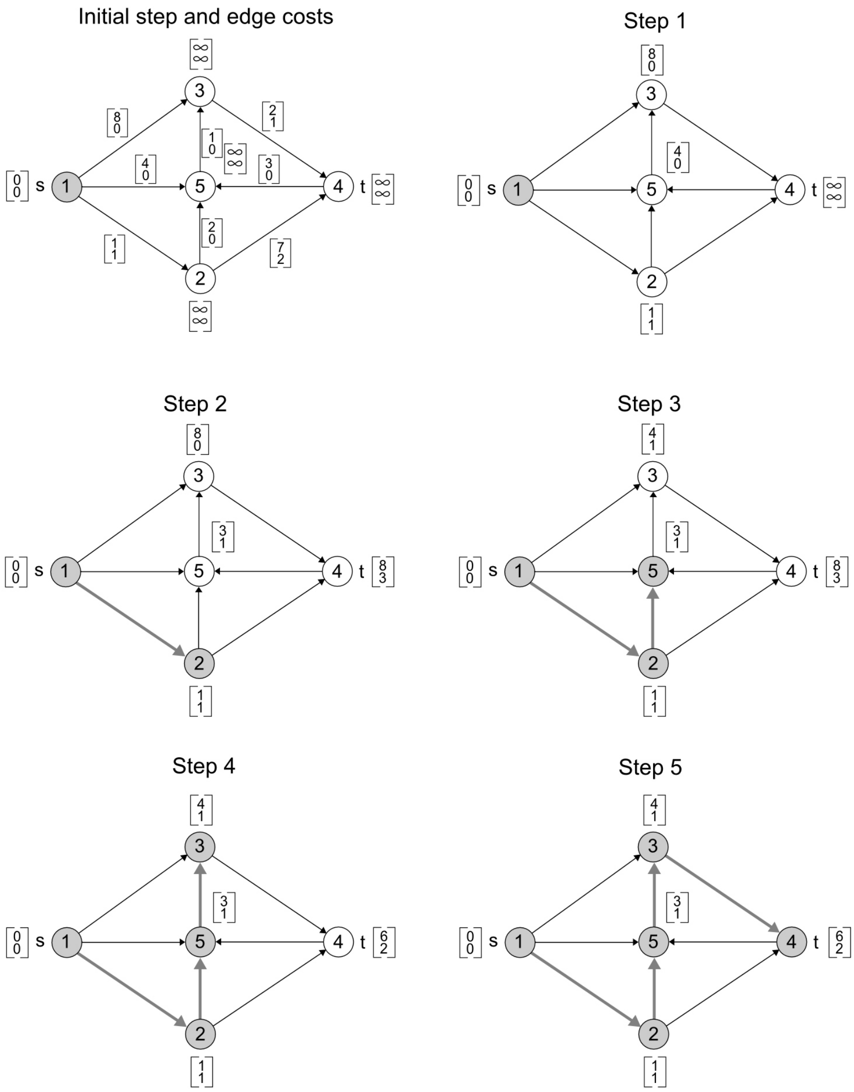


Figure 5.3: Procedure of Dijkstra's algorithm calculating a single Pareto optimal solution (Lexicographic minimal solution with f_1 = 'time' and f_2 = 'number of transfers')

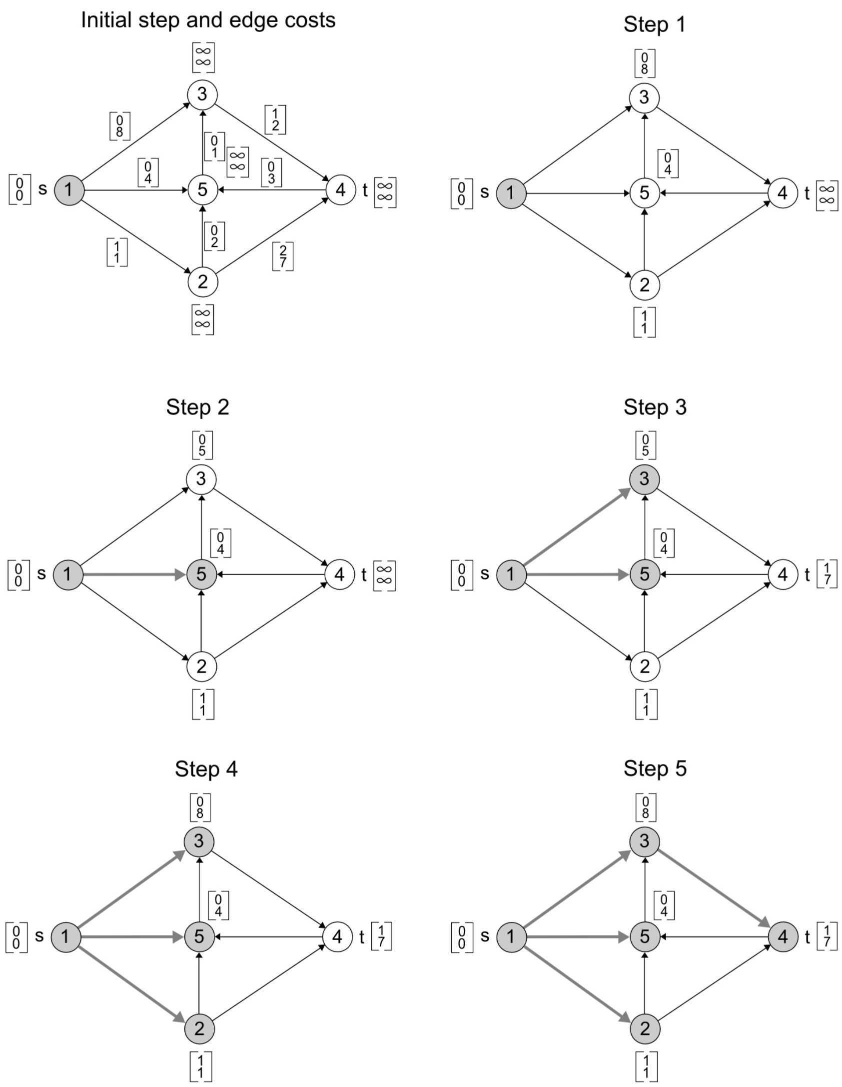


Figure 5.4: Lexicographic minimal solution with f_1 = ‘number of transfers’ and f_2 = ‘time’

Chapter 6

Modeling of schedules of public transport networks for route planning

Public transport plays an important role for pedestrians and even more for people with disabilities, as they are not able to use a bicycle or a car. In addition to the topological network of bus routes and streetcar routes, public transport is based on timetables, which have to be taken into account in the route planning algorithm. Timetables in public transport networks (PTN or PT networks) lead to a time-constrained network. In detail, every node and edge in the graph of a public transport network is restricted to specific time stamps. The main goal is to find an optimal path depending on the departure and arrival times of public transport vehicles.

Therefore, in public transport (PT) time-constrained route planning, hereinafter referred to as TCRP, has to be considered. A first step is the transformation of timetables to navigable networks. There are two basic modeling techniques for using time-constrained data in route planning: The *time-expanded graph model* and the *time-dependent graph model*.

6.1 Time-expanded graph model

In this graph model a time-expanded digraph $G = (V, E)$ is constructed where every node corresponds to a specific time event. Edges represent time-dependent elementary connections (Z, S_1, S_2, t_d, t_a), where Z is a vehicle traveling from station S_1 at departure time t_d and arriving station S_2 at time t_a. A connection between two stations S_1 and S_2 is elementary, if a public transport vehicle is moving between these two stations without stopping in between. By generating the digraph, all nodes are copied for every time event in the timetable. This results in a very large but sparse graph, [27, 45, 64, 67].

6.2 Time-dependent graph model

A time-dependent graph consists of a digraph with only one node per station. Edges represent the elementary connections or waiting within a station. The edge-length is a function of time and is calculated ‘on the fly’, depending on the time in which the particular edge is used, [45]. In Figure 6.1, the time-expanded model and the time-dependent model are compared to each other by giving a simple example.

In this thesis, the time-expanded graph model is considered and implemented. Some papers also introduce the time-dependent model like it is done in [51], [67] and [70]. For studying the time-dependent model in more detail, the interested reader is referred to [8] and [48]. An experimental comparison of the two models is given in [59].

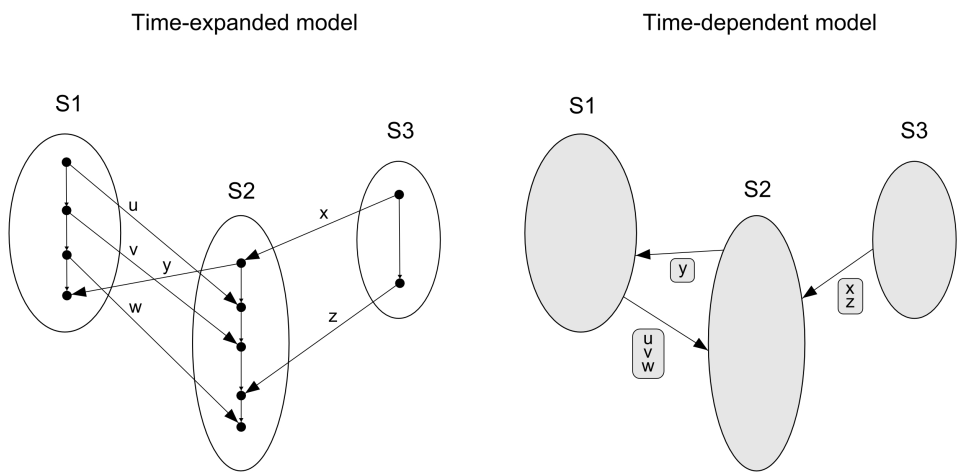


Figure 6.1: Difference between time-expanded and time-dependent graph models of a timetable with three stations $S1$, $S2$, $S3$. There are three public transport vehicles that connect $S1$ with $S2$ (u, v, w), one connection from $S3$ via $S2$ to $S1$ (x, y) and another connection from $S3$ to $S2$ (z) (example modified from [45])

6.3 Time model

Time is transformed in minutes for easier computation. The time of the current traffic day is given by $t = a \cdot 1440 + b$. At this formula, $a \in [0, 364]$ are traffic days starting with zero at the first of January, while $b \in [0, 1439]$ are minutes past midnight within a day, [58]. Hence, the actual time within a day is $t \ mod \ 1440$ and the current day is $t \ div \ 1440$. The travel time of an vehicle for each elementary connection is calculated from the departure and arrival times: $t_a - t_d$. The total time for a connection consists of the travel time, waiting time, and transfer time.

6.3.1 Scheduling constraints

The route planning problem in public transport networks involves the following time constraints, [1]:

- Departing from the start node is allowed only within a specified time window.
- Transfers are permitted within specified time windows.
- Arriving at the destination node is only possible at specific time windows.

6.4 Optimization criteria

For calculating shortest paths in public transport networks, two main optimization criteria exist:

- min. total time
- min. number of transfers

Instead of minimizing these criteria independently by the choice of the customer, simultaneous optimization of all criteria is the challenging task. This leads to a multi-criteria optimization problem.

6.5 Optimization problems in public transport

The main problems in public transport networks are the *earliest arrival problem* and the *minimum number of transfers* problem.

6.5.1 Earliest arrival problem

The earliest arrival problem (EAP) means finding the fastest connection from a given start to the destination and reach the destination node as early as possible by a given departure time, [45]. The fastest connection is computed by minimization of total time and leads to a single-criterion problem. The total time consists of total traveling time, total waiting time and total transfer time, [12].

6.5.2 Minimum number of transfers problem

The goal of the minimum number of transfers problem (MNTP) is to find a connection that minimizes the number of vehicle transfers in an itinerary from start to the destination. It is also a single-criterion problem. To solve the minimum number of transfers problem, transfer nodes and transfer edges have to be introduced in the graph model (see Chapter 10.2.4)

6.5.3 Earliest arrival problem with minimum number of transfers problem

For practical use we consider the combination of the earliest arrival problem and the minimum number of transfers problem. This leads to a bi-criteria shortest path problem (with the two criteria travel time and number of transfers) as introduced theoretically in Chapter 4 and Chapter 5.

Part II

Application

Chapter 7

User groups and demands

This chapter gives an introduction in user groups and their mobility aspects. To provide adequate route information for people with disabilities, the way those people orientate in unfamiliar environments has to be studied. This concerns mainly blind and visually impaired people but also wheelchair users. Another important question is which kind of obstacles can danger the itinerary or make it even impassable. Moreover, guidance information has to be tailored to the specific needs of the different user groups.

7.1 User groups

The route planner is mainly focused on three user groups with different disabilities. These three groups are:

- blind people
- visually impaired people
- wheelchair users

For these user groups, their requirements on route optimization are elaborated in detail. In the future, the route planner can be extended to further user groups.

7.2 Mobility

People with disabilities have special requirements on mobility. Besides specific information needed to orientate and navigate, people with different types of disabilities have contrasting demands on planned routes and additional information. Whereas wheelchair users face impassable barriers, blind and visually impaired people have hardly any spatial sense and are dependent on a detailed description of their close environment, [23]. R.G. Golledge did a lot of research concerning this topic. In his books *Spatial* *behavior*, [24], and *Wayfinding behavior*, [22], R.G. Golledge describes the behavior and spatial cognition of different types of people including blind and visually impaired people as well as wheelchair users. In addition to spatial cognition, people with disabilities have high demands on the built environment. This means that people with disabilities need more space to move, since they are dependent on specific travel aids. These travel aids can be guide dogs or white canes concerning blind and visually impaired people, and different wheelchairs or crutches in case of physically handicapped people.

7.3 Accessibility of the environment

People with disabilities have special needs on accessibility indoor and outdoor. The route planner focuses on outdoor environments. Accessibility has to be regarded already in the construction phase of infrastructure and is defined at the guidelines for accessible construction. This affects streets, sidewalks, footpaths, squares, parks and so on. Nevertheless, not all older buildings are easily accessible. For that reason, some infrastructure cannot be used by people with disabilities or dangers may occur. Especially, such situations are important to include in a route planner for people with disabilities.

7.3.1 Guidelines for accessible construction

For providing accessibility, guidelines for accessible construction are available and should be considered. In Austria, accessibility for the construction of infrastructure is defined at the ÖNORM B1600 (Austrian standard). In the following sections an extract of important guidelines is given.

Width needed

People with disabilities are dependent on mobility aids like white canes, wheelchairs or crutches. For this circumstance these people need more space to move. They rely on larger sidewalks and paths for a safe travel. In [36] standards of minimum widths are defined for each kind of disability (Figure 7.1, measures in cm).

Inclination of sidewalk

The inclination of the sidewalk influences the mobility of wheelchair users. For most wheelchair users, the maximum value that can be passed are 6%, see Figure 7.2, [36]. This value depends on the kind of wheelchair (electrical or manual). Compared to manual wheelchairs, electrical wheelchairs allow passing higher inclination values. Furthermore, the cross-fall of sidewalks

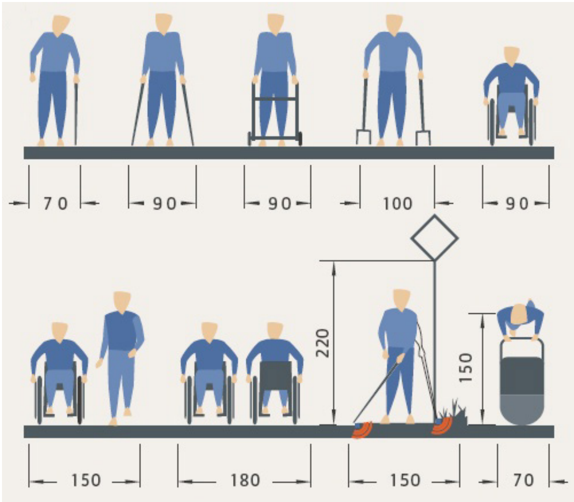


Figure 7.1: Widths needed [cm]

plays an important role for wheelchair users and must not exceed a maximum value of 2%, [36].

Figure 7.2: Inclination of sidewalk max. 6%

Stairs

Stairs are impassable for wheelchair users and can be dangerous for blind people. Steps on footpaths or at entrances should be replaced by ramps (Figure 7.3, [36]).

Toilets

Lots of cities provide public toilets especially for wheelchair users. These accessible toilets can be opened with a so called 'Eurokey'. This service is a Europe-wide locking system, which can be opened with a special universal key owned by wheelchair users. Accessible toilets provide more space and

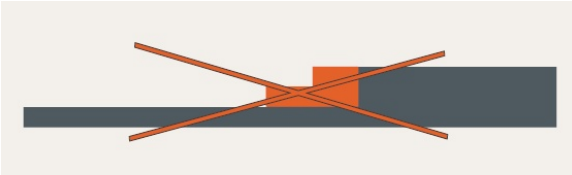


Figure 7.3: Stairs

guarantee toilet paper holder, wash basin, soap, and so on within reach. Moreover, they are equipped with handles (grab bars).

Disabled parking space

Marked parking slots are provided for people with disabilities. These parking areas are particularly near entrances and provide more space for getting on and off the vehicle (Figure 7.4, measures in cm, [36]).

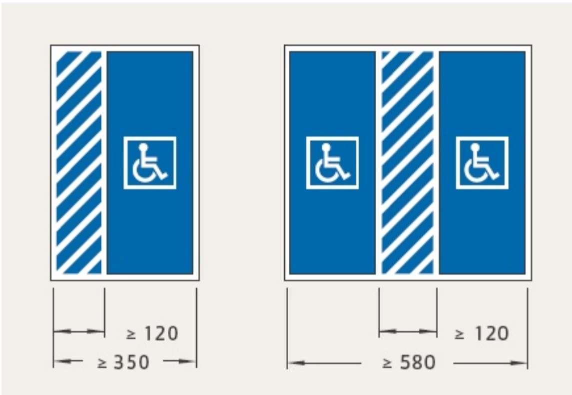


Figure 7.4: Disabled parking space [cm]

Material of path surface

The surface structure of a path is a crucial fact for wheelchair users or people with impaired mobility. In some situations the material of the surface can be dangerous for blind or visually impaired people, too. Decorative patterns like brick pattern are hard to handle for wheelchair users. People with vision and cognitive impairments have difficulties detecting the beginning and the end of the street, which can cause dangerous situations (Figure 7.5, [5]).

7.4 Constructional orientation aids

Orientation aids help people with disabilities to orientate in unknown environments. Besides guidelines for accessible construction, in Austria the

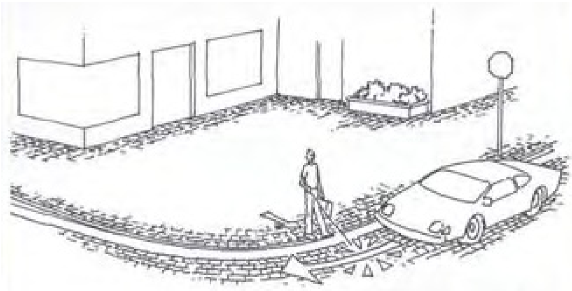


Figure 7.5: Patterns

ÖNORM defines such technical aids for blind and visually impaired people as well. In detail, the ÖNORM V 2102 contains these orientation aids. Such aids can be tactile or acoustic. In the following, constructional orientation aids for blind and visually impaired pedestrians are described.

7.4.1 Tactile paving

Tactile paving (also called truncated domes, detectable warnings, tactile ground surface indicators or detectable warning surfaces) are textured ground surface indicators on footpaths, train station platforms, etc. to assist blind and visually impaired pedestrians. These tactile warnings have a special surface pattern. Common patterns are truncated domes or truncated bars. These patterns can be recognized with the blind stick well. In the city of Graz (Austria) tactile paving is called 'Grazer T' as the truncated bar tiles are arranged as a 'T' in front of street crossings. Truncated dome tiles are placed at streetcar stations to mark the entrance point. Nevertheless, during winter, snow and gravel cover the tiles and make them hard to detect.

7.4.2 Acoustic traffic lights

Traffic lights with acoustic signals help blind and visually impaired people differ between walk and wait phase. A weak attention signal is audible during the wait phase within a distance of around three meters to find the traffic light. When the wait phase turns into the walk phase, the acoustic signal becomes faster and indicates that crossing the street is possible. At the traffic light itself, a tactile overview of the street is provided. It includes the number of lanes, the moving direction of the vehicles, bicycle paths, and other traffic infrastructure to give blind people an idea of the surrounding, [55].

7.5 Demands on navigating in unknown environments

People with different types of disabilities have contrasting demands on a planned route, e.g., blind people have other requirements on a barrier-free route than wheelchair users. While for the latter, routes must not contain stairs or narrow points, for blind people obstacles on the sidewalk (e.g., traffic signs) are dangerous. Moreover, blind people need much more information about the surrounding like infrastructure (e.g., traffic signals) or buildings. Infrastructure on the sidewalk (e.g., fire hydrants and telephone poles) are also used by blind people to estimate distances, [23].

To realize the planning of barrier-free routes for different user groups, additionally to the use of existing literature, two studies for evaluating these special requirements have been done. These questionnaires have been sent out to organizations of people with disabilities as well as lots of private individuals. The first part was an extensive questionnaire about preferred information needed before traveling to an unknown city and has contained the following topics:

- information needed
- obstacles on the sidewalk
- preferred routes
- content of guidance information
- questions about the use of public transport
- accessibility of the user interface

The second task was sorting the list of recommended information and obstacles by priority divided into user groups. The questionnaire was replied by various interested people. The groups of blind people and visually impaired people were combined in the analysis, since they had nearly the same results. The output of the whole questionnaire contains lists of preferred content of geographical data and obstacles, favored user functions, necessary landmarks which are essential for orientation, and examples of route instructions like turn information, landmarks, and so on, [57].

7.5.1 Information needed

This topic contains infrastructural requirements and dangerous obstacles which the system should include. Table 7.1 shows the first ten most important information listed at the correspondent questionnaire. Besides these ten objects, as many objects as possible from the whole priority list were included in the route planner dependent on the availability of geographical data. In [47] results from interviews with blind and visually impaired people

Table 7.1: Top ten additional informations defined by user groups

group	blind and visually impaired	wheelchair users
1	acoustic traffic lights	wheelchair accessible toilets
2	medical care	disabled parking spaces
3	tactile paving	wheelchair accessible hotels
4	shopping facilities	bars and restaurants
5	barrier-free toilets	sights
6	bars and restaurants	cash points
7	taxi	museum
8	cash points	tourist information
9	pharmacy	taxi
10	infrastructure facility (eg., police)	shopping facilities

about necessary information that shall be contained in an navigation system for blind people are presented. Compared to Table 7.1 similar results of [47] are infrastructure facilities like medical care, shopping facilities, cash points, restaurants, and so on. Such preferred objects are also called points of interest (POI). Additional results of [47] named by blind people are street information (e.g., number of buildings) or information about the road itself (e.g., number of lanes).

Besides various kinds of objects, information about accessibility is important in a route planner. For people with disabilities it is necessary to know whether or not accessibility guidelines (see Chapter 7.3.1) are followed. Questions like: 'Are there any stairs at the entrance of the restaurant?' or 'Is there a wheelchair accessible toilet?' have to be answered.

7.5.2 Obstacles

When establishing a route planner for people with disabilities, one main question arises: What defines the best route for people with different disabilities? One influencing aspect are obstacles. They can be divided into two different groups: *impassable objects* and *obstacles*. The way these two groups are considered in the route optimization differs. Paths containing impassable objects must not be part of the calculated route. Whereas obstacles have to be minimized and warnings are provided to the user.

Within these two groups, three different types of obstacles can be divided:

- permanent obstacles
- temporary obstacles, which can be hardly included in the route planner (e.g., bicycles on the sidewalk)

Table 7.2: Top ten obstacles defined by user groups

group	blind and visually impaired	wheelchair users
1	bus and train stops with exit at street level	stairs and steps
2	construction sites	material of path surface
3	crossings without traffic lights	missing sidewalk ramps
4	material of path surface	inclination and cross-fall of sidewalk
5	crossing of squares and parks	rails at street
6	cordons and gates	narrow points
7	narrow points	cordons and gates
8	rails	construction sites
9	mobile obstacles (eg., bicycles)	bus and train stops with exit at street level
10	bicycle paths crossing foot paths	crossing of squares and parks

- real-time information (e.g., construction sites)

Table 7.2 shows the first ten most important obstacles listed from potential users at the correspondent questionnaire.

Obstacles influencing wheelchair users

For wheelchair users, a route is usable, if it does not contain any stairs. Instead, ramps are needed. But not only this single aspect makes a route attractive. The surface structure of footpaths and the width of sidewalks play also an important role. Last but not least, obstacles can make the route hardly passable.

Impassable objects:

- stairs
- narrow points
- missing ramps, if step is too high
- cordons and gates

Main static obstacles:

- rough surface structure
- rails at street
- missing ramps

Obstacles influencing blind and visually impaired people

Considering blind and visually impaired people, there are not any impassable objects as they are for wheelchair users. However, there are obstacles which are dangerous in some way (individually different). Moreover, crossings without acoustic traffic lights have to be avoided. Besides this, the width of the sidewalk is influencing the optimality of the route. Information about the surface structure is used by blind people for orientation. The surface structure can be felt very well with the white cane.

Main obstacles:

- street crossings without acoustic traffic lights
- traffic signs
- post boxes
- cordons and gates
- stairs

7.5.3 Preferred Routes

In route optimization, various possibilities to define an optimal route can be distinguished. This can be the shortest route, the fastest route, the safest route, the most attractive route, etc. However, the route preferred by one individual is not necessarily the most favorite route of another person. Therefore, a route planner shall provide the possibility to choose the optimization method. R.G. Golledge did a study on how people select routes, [21]. The three most preferred optimization methods named in his study are: the shortest route, the fastest route and the route with least turns. R.G. Golledge calculates the route with fewest turns in his study on navigation aids for blind people, [23]. In this thesis, additionally, accessible routes with fewest obstacles and routes with least transfers in public transport are mainly considered, as these optimization methods were resulting from the questionnaire and often named by potential users. The minimization of transfers is especially important for wheelchair users and blind people, since changing vehicles is a very uncomfortable and time consuming procedure for them. A first approach of calculating the route with least hazards was done in [30].

7.5.4 Content of guidance information

The content of guidance instructions depend on the way people with disabilities orientate in unknown environments. Especially blind people have high demands on guidance instructions, as they are not able to see a route visualized on a map. Moreover, the representation of spatial knowledge for blind people is a challenging task, because people which are blind from birth have few if any spatial sense, [23]. In the questionnaire, most of people answered that guidance information shall include the following information:

- Route overview: A route overview shall provide general information about the complexity of the route. Route overview information shall contain the total length/time and the number of transfers. The number of streets, turns and crossings was also listed by the Mobic Consortium as part of the route overview, [47].
- Distances.
- Turn information (straight, left, right, etc.).
- Landmarks in combination with turn information: Guidance information shall include landmarks which can be easily recognized by blind people. This can be acoustic traffic lights, phone boxes and so on. Guidance information shall be stated as follows: ‘In 20 meters, *at the traffic light*, turn left.’
- Blind people often orientate on the geographic direction (north, east, south, west). This was mentioned in the questionnaire and can be found in literature, like in [47].
- Street names: name of streets to travel along, name of streets that have to be crossed, name of streets to turn into.
- Obstacles and dangerous parts of the route, if they cannot be avoided.
- Name of public transport stations (entrance station, exit station).
- Travel time of public transport vehicle and number of stations to travel.

7.5.5 Public transport and accessibility

Most of pioneer work done by R.G. Golledge, [23], the MOBIC-Consortium, [47], or S. Helal, [30, 61] does not consider public transport in their pedestrian guidance system. These systems are focused on traveling by feet. However, in the accomplished questionnaire, it turned out that public transport is very important for people with disabilities. Most of the asked people use public transport daily, but difficulties come up in using public transport vehicles. For wheelchair users, missing ramps reduce the usability of public transport vehicles and therefore, not every arriving vehicle within a station can be used. Blind people are not restricted in use, but the fact that they have problems to find the station of a specific vehicle number - especially at huge places - makes the use of public transport uncomfortable. These facts shall be considered in route planning to enhance the mobility of people with disabilities when using public transport vehicles. To overcome the problem of missing ramps, the information about accessibility of public transport vehicles shall be stored in the route planner. As a consequence, only accessible vehicles for wheelchair users shall be considered. To help blind people finding the station, detailed route descriptions shall provide the necessary information.

7.5.6 Accessibility of the user interface

Especially, when establishing a software for blind and visually impaired people, the user interface is a challenging task, as this user group is not able to see any graphical information visualized on the screen. Moreover, blind people cannot work within a software by using a computer mouse. To overcome this problem, specific Web Content Accessibility Guidelines of the W3C-Consortium[1] for the creation of accessible websites have to be observed. This guarantees an unrestricted use of websites for blind people. More details about these guidelines are given in Chapter 8.2.3.

[1] http://www.w3.org

Chapter 8

Software requirements

The goal of this work is the development of a web-based route planner, that supports people with disabilities traveling around unknown parts of cities. It was decided to create a web-application, because such an application can be used easily by a huge amount of people. Moreover, users are not dependent on expensive hard- or software. Every standard computer with an Internet connection is sufficient for the use of the route planner. Considering the functionality, the route planner is focused on pre-trip information in contrast to an on-trip navigation system. The route is planned before starting the journey and can be studied, compared to other routes, and learned by heart. This is especially important, when people with disabilities visit other cities as a tourist on their own. Those people can plan their journey at home and select accessible environments and locations for their trip (e.g., hotels, restaurants, etc.).

For the realization of the route planner, this chapter deals with the definition of software requirements. Functional and non-functional requirements form the basis of the software design and architecture, [75]. These requirements are derived from the demands of potential users described in the previous chapter.

8.1 Functional requirements

Within functional requirements, the behavior of the system is described. The main functionalities of the route planner are the following: entering via login, user profiles, route planning, map display with map functions like creating points of interest (POI), administration of POI, pre-trip training, and a soundfile of the route. These functionalities interact with each other. An overview of main functionalities with their interconnections is given in Figure 8.1. In the following sections these parts are described in detail.

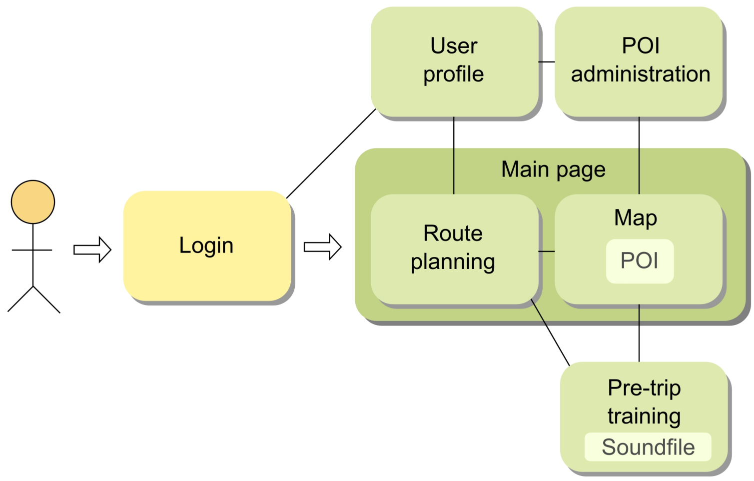


Figure 8.1: Functionality overview

8.1.1 Login

The system must provide a login to enable individual settings to the user.

Steps

The user should enter an username and a password and be able to confirm or cancel the input. The system verifies the login data and displays the profile selection, if the login data is correct.

Additional steps

If the system fails to verify the user's access rights because of an unknown user or a wrong password, the system shall display a message and shall remain at the login screen.

8.1.2 Profile selection

An unregistered user shall be able to select a standard profile.

Steps

The system shall display four standard profiles (blind, visually impaired, wheelchair, without preadjustments) at the start page. The user selects a standard profile and the system loads the main page.

8.1.3 Search

The user shall be able to search for cities or addresses.

Steps

The user enters a city name or an address and confirms the search. The system shall display search result on the map.

Additional steps

If the input cannot be found, the system shall display a message.

8.1.4 Map display

The map is mainly provided for wheelchair users and shall include standard functionalities like zoom and pan. Points of interest shall be displayed on the map with different symbols for each category. The map shall be as detailed as possible and shall contain different street categories with street names, addresses, points of interest, sidewalks, etc.

8.1.5 Creating points of interest

The user shall be able to add a new point of interest with additional information to the map.

Steps

The user shall be able to set a POI on the map. The system shall display a new dialog for entering the specification of the POI. The user shall enter the POI-name, the POI-category and a comment (e.g., information about accessibility). Furthermore, the user shall be able to select the status of the POI (public or private). After entering the information, the user confirms or cancels the input. With the confirmation, the system shall display the POI on the map, showing the corresponding category symbol. By the selection of a POI on the map, its additional information shall be shown.

8.1.6 Points of interest administration

Within the POI-administration users shall be able to manage their private POI and modify existing POI.

Steps

The system lists all private POI entered to the system (also those the user has set public). The user shall be able to edit or delete private POI. In editing mode the user shall be able to change the name of the POI, the category and the status (e.g., set a private POI to public or vice versa).

8.1.7 User profile

The user profile shall allow the user to specify individual settings for route calculation and additional information.

Steps

The user selects a standard profile with predefined settings. The user shall be able to change the predefined settings and define individual optimization criteria and restrictions. These are: maximum walking distance, maximum inclination, preferences for public transport(yes/no, bus only, tram only) and the maximum number of transfers. As a second part, the system shall provide setting possibilities for obstacles and points of interest, where the user can choose relevant categories and set the obstacles/POI on or off. Another part of the profile shall provide general settings and the possibility to change the personal login data. Finally, the user confirms or cancels the changes made to the settings. Moreover, the user shall be able to reset the individual settings to the predefined standard profile.

8.1.8 Route planning

Route planning provides an optimized route to the user.

Steps

The user enters start and destination addresses (street name and house number), the city of Graz is preselected. The user enters date and time for the consideration of available public transport vehicles. After that, the user confirms the route calculation. The system calculates the route with the chosen profile and displays the route on the map. The user can change the profile to wheelchair, blind, visually impaired, and personal. Personal profiles are available for registered users only. Within the personal profile, route settings of the individual user profile are considered in route planning. The system calculates the route for the selected profile and displays the route on the map.

Output: Route description

The route description appears at the main page. The route description shall be divided into an overview about the route with general information and a detailed route description. The general route information shall include the length/time of the journey, the number of transfers and the arrival time at the destination. The detailed route description shall contain all necessary information for orientation described in Chapter 7.5.4.

8.1.9 Pre-trip training

The pre-trip training provides three different routes with a detailed route description.

Precondition

A route must already be planned.

Steps

The system suggests three different routes with general and detailed route description (see Chapter 8.1.8). The user shall be able to compare these routes by selecting them in sequence. The system displays the selected route on the map and shows a detailed route description.

Additional steps

If a route has not yet been planned, the system displays a message that a route must be planned before.

8.1.10 Soundfile download

The route description shall be provided as a soundfile.

Precondition

A route must already be planned.

Steps

The user shall be able to download a soundfile of the current route. The system displays a download window where the user can decide between open, download, and cancel.

Additional steps

If a route does not already exist, the system displays a message that a route must be planned in advance.

8.2 Non-functional requirements

Non-functional requirements describe system qualities and strongly influence the software architecture, [75].

8.2.1 Usability

Usability defines, whether a product (software, etc.) is user friendly or usable, respectively. In other words, usability means how easily people can use the system in the sense of user interaction. In conjunction with web pages this is known as web usability. The menu structure should be self-explanatory and not be overloaded. Moreover, the use of the system should be easy to learn. This makes a system usable for inexperienced computer users. Usability can only be reached by knowing the user groups and their behavior. In the underlying case, users are blind and visually impaired people as well as wheelchair users. For blind people a well-structured menu without various links is very important.

8.2.2 Accessibility

Due to disabilities, lots of people cannot use general software products or web pages. They need special enabling technology for getting the content of the web page in an adequate way. E.g., blind people are not able to see the output on the screen. Moreover, they cannot navigate in a software program by using a computer mouse. Blind people need assistive technologies like screen readers or braille displays.

Screen readers

Screen readers are software products that convert text-based information to speech. They are widely used by blind people, as it is one of a few methods for getting information from the screen. The leading products are JAWS (Job Access With Speech) and Window-Eyes followed by others systems. Screen readers read the text on the screen in the predefined tab sequence, see Chapter 8.2.3.

Braille

Braille is a language for blind or visually impaired people to read and write. Braille consists of character cells. Each Braille character is made up of six dot positions. The characters of the Braille alphabet can be divided in different sets, see Figure 8.2[1]. Set 1 contains the first ten letters of the alphabet (a-j) which are formed using only the top four dots. Adding the number identifier (Figure 8.2, Set 5) these symbols also represent the digits 1 through 9 and 0. Adding dot three in the first column forms the next ten letters (Set 2), and adding dot 6 (last dot in the second column) forms the last six letters of the alphabet (Set 3). Additionally, there are combinations for other symbols and combined letters.

[1] Image source: http://commons.wikimedia.org

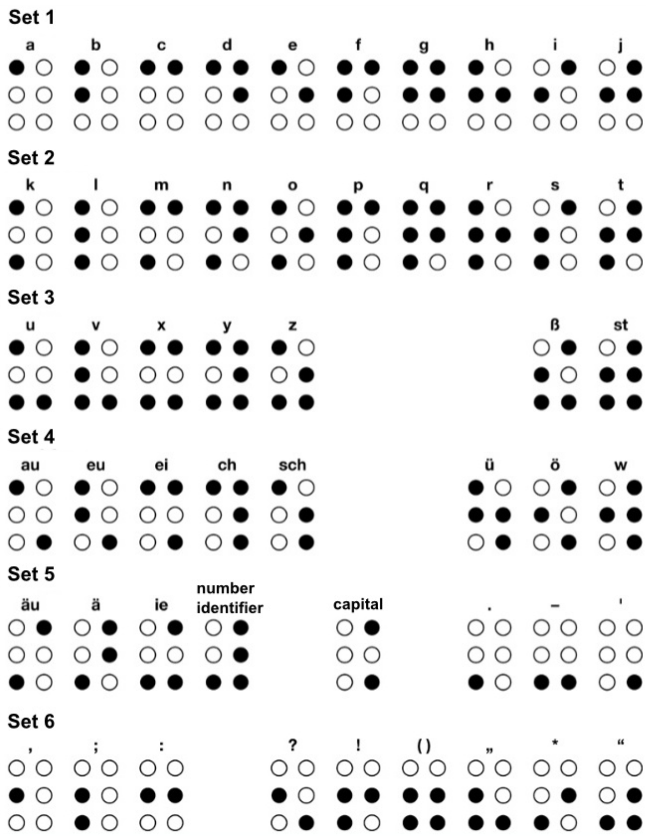


Figure 8.2: Braille characters

Braille displays

Blind or visually impaired computer users, who cannot use a normal computer monitor, use Braille displays or Braille terminals to read text output. These technical aids are tactile devices that display textual information as Braille characters. This is achieved by raising pins made of metal or nylon.

8.2.3 Web accessibility

Web accessibility is an important fact concerning websites for people with disabilities. Due to the development of barrier free website the Web Accessibility Initiative (WAI) from the W3C-Consortium must be mentioned. The W3C-Consortium has defined the Web Content Accessibility Guidelines (WCAG 2.0)[2] which cover a wide range of recommendations for making web content more accessible. WCAG 2.0 are not technology-specific. To get an overview, an extraction of important facts is given below. The WCAG are

[2]http://www.w3.org/TR/WCAG20/

also described in [18] by giving some examples.

Tab sequence

The tab sequence enables users to navigate through an application by using the keyboard. Moreover, the tab sequence defines the sequence of the screen readers output.

Text equivalents

Pictures can not be 'seen' by blind people. In addition, the content of pictures can not be read by screen readers. Information can only be provided to blind people using plain text. For non-text elements like pictures text equivalents must be defined. Moreover, information can not be provided by color differences.

Color scheme

Especially for visually impaired people, colors and contrast play an important role. Different color and contrast schemes for visually impaired people can be obtained by using Cascading Style Sheets (CSS), so that the users can choose their preferring color profile.

In context with a screen reader

For the use of screen readers, practical tests show the importance of avoiding special characters. These characters are read additionally as words. This can lead to expressions which are very difficult to understand. An example provides more insight: The word 'pre-trip' was spoken as *'predashtrip'* by the screen reader (JAWS) and was hardly understandable. Problems can also occur by the use of names that contain special sequences of characters (e.g., 'tth', at Matthias) or abbreviations with capital letters (e.g., ABC). Such phrases can especially be part of POI-names. Another point are foreign words. E.g., there appear lots of germanized English words in German language which are spoken by the screen reader with German pronunciation and therefore sound very strange. This circumstance has to be considered in the development phase of the route planner, especially in the geographical data base and the graphical user interface.

8.2.4 Performance

Performance is characterized by the amount of work accomplished by the system compared to the time and resources used. For the development of a route planner for people with disabilities the crucial values are the route calculation and the mapping of the POI because of the huge amount of data

Table 8.1: Performance of system activities

activity	max. duration
page loading	1sec
verifying the login	1sec
loading the map	5sec
loading POI	3sec
save a POI	1sec
city search	2sec
route calculation	4sec
mapping the route after calculation	1sec

that has to be processed. In Table 8.1 upper performance bounds of acceptable durations of the main system activities are given. These performance bounds are defined in cooperation with a test user. Since the expectancy is individually different, the performance bounds should be tested on various different users. However, such a study is out of scope of the present thesis.

8.3 Usage scenarios

The following two examples of possible scenarios illustrate the usage of the system.

8.3.1 Wheelchair user

A wheelchair user enters the system with the personal login data and selects the wheelchair profile in a second step. At the main page start and destination addresses, date, and time for finding a route to a restaurant at the main square are entered. After printing the map including the calculated route and route information, the wheelchair user starts the journey to the restaurant. The suggested route is free from obstacles. Arriving at the restaurant, steps at the entrance make it hard getting in, but luckily a friendly person was helping. Unfortunately, these steps are not included in the geographical data base. Coming back home, the user starts the route planner again and adds a public point of interest (obstacle) on the map with the additional information of stairs at the entrance of the restaurant. From now on, other users will be warned.

8.3.2 Blind person

A blind person enters the system and chooses '1' as the maximum number of transfers in the individual user profile. Additionally, the user selects some obstacles like traffic signs and post boxes which should be avoided at the route. Afterwards, the blind person plans a route to the shopping mall on the outskirts of the city. The calculated route contains foot paths as well as public transport vehicles taking the predefined restrictions under consideration. At the pre-trip training the blind person compares the personal route with the fastest route and finds out, that this route is five minutes faster but contains three transfers. So the user remains at the personal route. Finally, the user stores the detailed route description as mp3-file and transfers it on the mobile phone for taking it along.

Chapter 9

Geographical data

Geographical data is often named as spatial data, geospatial data or geographical information. Geographical data is used for modeling the real world as a digital map. In recent days, digital maps are very popular and even indispensable in economic and scientific analysis. Digital maps provide relevant information for a huge amount of different applications. In such applications, geographical data is not only used for visualization. Besides this, analysis based on geographical data can be done. Main topics in this context are planning questions (e.g., city and regional planning), predictions (e.g., flood prediction), routing applications or monitoring. Today, many companies provide digital maps for various topics. The widespread use of geographical data in the last years even motivated many volunteers to map their home town. Such volunteered geographic information (VGI) has now a comprehensive availability and can be freely used by everyone. The specific example of OpenStreetMap[1] will be shown later on in this chapter. Moreover, this chapter deals with digital maps for route planning, the creation of a pedestrian network, and the combination of different data sources like street networks, pedestrian networks, public transport networks, and specific information for people with disabilities.

9.1 Modeling the real world

The surface of the world can be described by mapping geographical data with the output of a digital map. Such a modeling process is always a simplification of the reality and means a reduction of complexity, [7]. The output is an abstraction of the real world. The reality can be mapped using points, lines and polygons, which form the digital map. The digital map is stored with its coordinates, topology and semantics in a database using specific data structures and file structures on a computer. The structure of a data modeling process is shown in Figure 9.1.

[1] http://www.openstreetmap.org

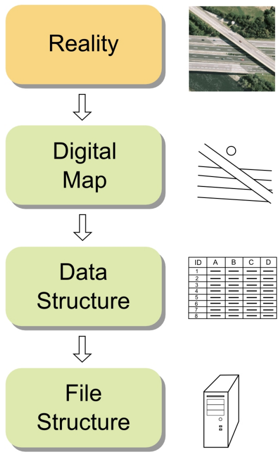


Figure 9.1: Flowchart of a data modeling process (modified from [7])

9.1.1 Digital maps

The real world can be mapped as digital maps for using them in different kinds of applications. In contrast to paper maps, digital maps are much more flexible in the sense of adding, editing, and updating data and can contain much more information. These information can be requested by spatial analysis functions. Digital maps can be represented by a vector data model, where entities are defined using points, lines, and polygons, [3, 11]. Thematic information is stored as attributes on each object and defines specific properties (see Figure 9.2). Moreover, interrelations between entities describe how entities are linked together. These interrelations are named topology as explained in Chapter 2.3.1.

9.1.2 Digital maps for route planning

Digital maps for route planning mainly consist of points and lines, which form a graph and are then called nodes and edges (see Chapter 2). In a digital city map for route planning, edges represent streets and footpaths, nodes are the generalization of crossings. Polygon features can be used for the modeling of buildings, but they can also be reduced by polygon points or isolated points in the network. A topological correct graph for route planning has to fulfill certain properties. Such a graph has to be connected, directed, and valuated (see the description of graph properties in Chapter 2.3). Figure 9.3 demonstrates a very simple abstraction of a street network for route planning consisting of nodes and edges and including examples of attribute information.

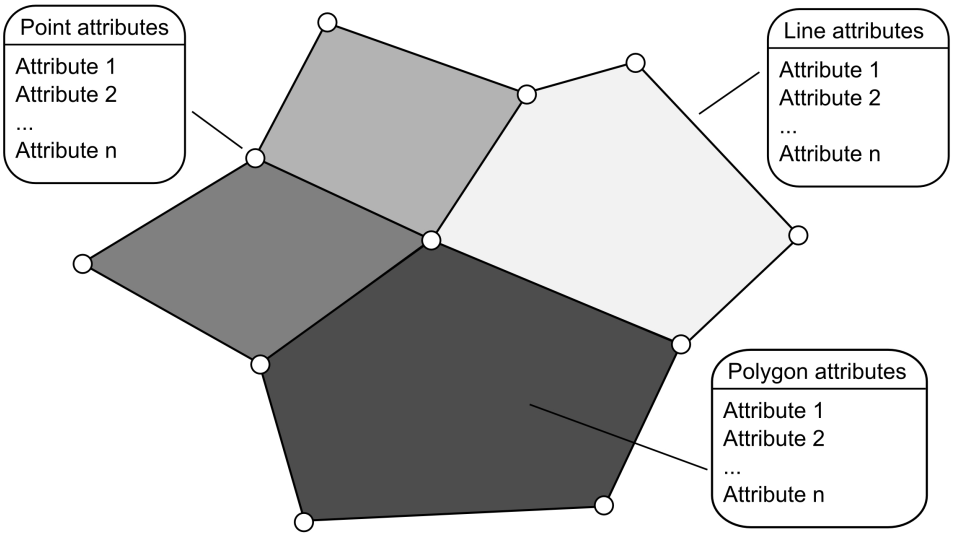


Figure 9.2: Geographical vector data and their attributes (modified from [34])

9.2 Geographical data sources

Today, various suppliers of digital maps provide local and global geographic data for many topics. When developing a new application based on geographical data, the main question is: 'Which digital map does support my requirements best?' The same question appears in context with the development of a route planner for people with disabilities. On the one hand, geographical data included in the route planner should cover large areas or even the whole world to provide the service to as many people as possible. On the other hand, a global solution contains less detailed information than local data focused on a specific topic. In case of people with disabilities detailed information is very important. However, a local solution with lots of specific information reduces the availability of the system. To overcome this problem, the idea of combining different data sources in one application was born. For data combination in the developed route planner for people with disabilities, three different data sources are chosen:

Global data

- OpenStreetMap data (OSM)

Local data

- Detailed pedestrian network for people with disabilities (DET) containing specific information for a part of the city of Graz, Austria

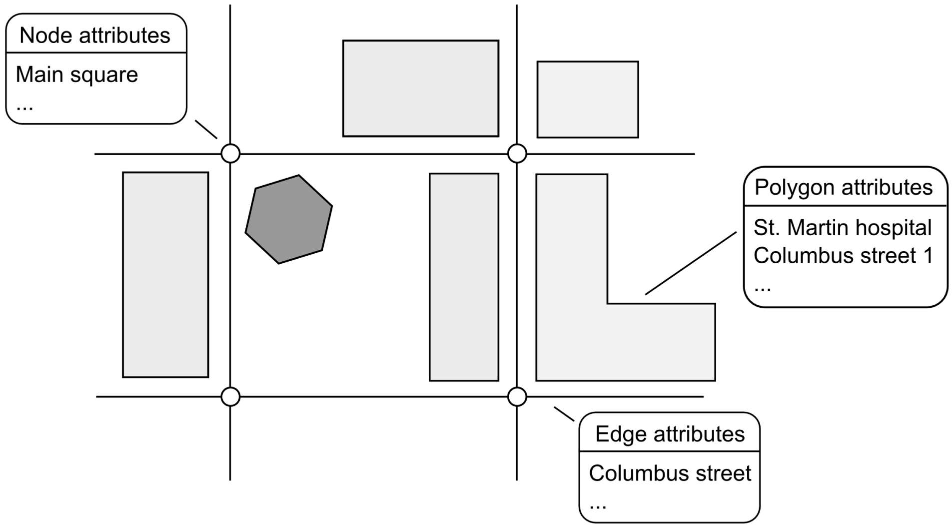


Figure 9.3: Geographical data for route planning including attribute information

- Public transport network (PTN) of the public transport company *Holding Graz Linien* from the city of Graz, Austria

9.3 OpenStreetMap

OpenStreetMap[2] is a free digital map created by volunteers. It was introduced in 2004 by *Steve Coast* and is now a popular map used in a wide range of applications, [60]. With the help of many volunteers this map is growing very fast. After registration at the OSM-community, everybody is enabled to add missing information or change existing information using the integrated Potlatch GIS-Tool or the JOSM Java-Editor[3].

9.3.1 Quality aspects

The quality and quantity of OSM data has strongly increased in the last years, since several hundred thousand volunteers have joined the OSM-community until the end of 2011. Those people are mapping their environment to share volunteered geographic information with everyone in the world, [50]. However, the availability of OSM data in a specific area depends on their population density. Highly populated areas are more complete than rural regions, since there are many more volunteer mappers available in cities than in countrified regions, [50]. Another quality aspect when dealing with

[2]http://www.openstreetmap.org
[3]http://josm.openstreetmap.de

volunteered geographic information is correctness. In this context we should ask: 'Have volunteers sufficient knowledge to map geographical data in a correct way?' Some volunteers have professional qualifications but many others do not have specific knowledge about mapping geographical data, [25]. This aspect plays an important role in the generation of path networks for route planning. Particularly in this field, knowledge about network topology and graph properties (given in Chapter 2) are indispensable to provide useful data.

In [50], a comparison between OSM and commercial data from 2007-2011 is given. In particular, OSM data has been checked against TomTom (TeleAtlas) and Navteq data in three categories: relative completeness, attribute accuracy and temporal accuracy. Regarding relative completeness, it was demonstrated that OSM surpassed TomTom in the total number of streets recorded in 2010. However, with respect to attribute accuracy, TomTom is the leading provider concerning street names and turn restrictions for car navigation, [50]. On the other hand, OSM has its strength in the enormous variety of information for different applications. Another advantage of OSM is the quick response time for data implementation and data correction, which assures us currentness of data.

9.3.2 OpenStreetMap data content

OSM contains street network data with different street categories, cycle ways and some footpaths. Many POI provide additional information about city infrastructure and leisure activities (e.g., public buildings, cash machines, restaurants, etc.). The included address data is useful and necessary to search for the start and destination address in route planning. Figure 9.4 shows OSM data of the campus area of Graz University of Technology in the city of Graz, Austria in the year 2010. In comparison to that, Figure 9.5 represents the same area in year 2012 and shall illustrate the enhancement of OSM in the last years.

9.3.3 OpenStreetMap for route planning

Besides a digital city map, OSM can also be used for route planning. However, this leads to the necessity of advanced knowledge about network modeling. Route planning requires a topological correct graph consisting of nodes and edges (see Chapter 2). Considering street networks, street crossings are modeled as nodes and streets are represented by edges (Figure 9.3). Due to the lack of professional knowledge about graph theory of some volunteered mappers, in OSM, the street network data does not exactly form an accurate graph. Giving one example, intersecting streets partially do not have a common node. Therefore, OSM data has to be preprocessed for route planning, [63]. The preprocessed OSM network for the city of Graz consists

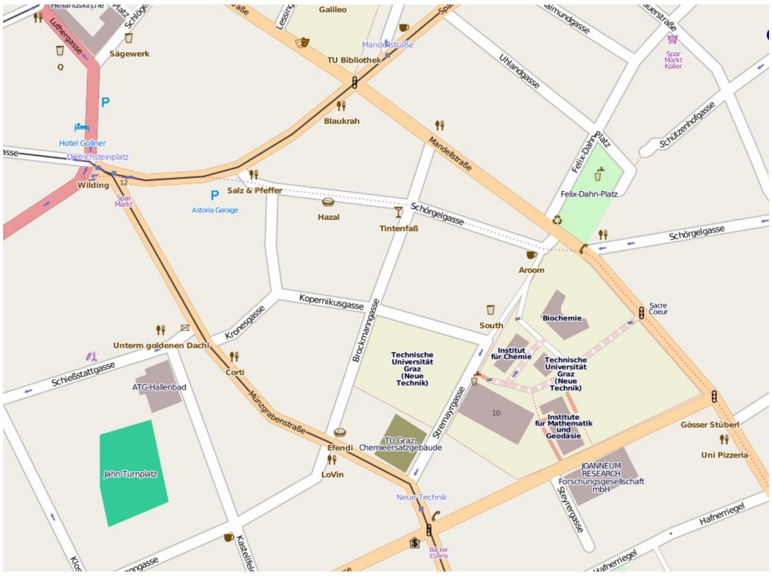


Figure 9.4: OSM data content (April 2010)

of about 10.000 nodes and 7.500 edges. Suggestions for increasing the quality of volunteered geographical data by providing quality metadata for the volunteers is given in [6].

Existing route planning projects based on OSM data are: 'OpenRouteService.org'[4] described in [63] and the 'OpenStreetMap Routing Service'[5].

9.3.4 OpenStreetMap data content for people with disabilities

OSM partially provides attribute data (OSM tags) to store information for people with disabilities, [44]. Unfortunately, these attributes are rarely collected. Moreover, some detailed geographic data need a specific object structure, which is not available in OSM. This affects sidewalks, which should be mapped as additional edges, to guide blind people along. A suggestion for modeling paths for pedestrians is given in Chapter 9.4. Information for people with disabilities which is already included in OSM data are stairs, toilets for wheelchair users, and information about the path surface structure. In a wheelchair routing project[6] at the University of Heidelberg, such OSM tags for wheelchair users have been described. While there are efforts to provide information for wheelchair users in OSM, there is a lack of information in OSM for blind and visually impaired people. A reason may be the fact, that blind people are not able to add data to OSM on their own, since

[4]http://www.openrouteservice.org/
[5]http://yournavigation.org/
[6]http://wiki.openstreetmap.org/wiki/DE:Wheelchair_routing

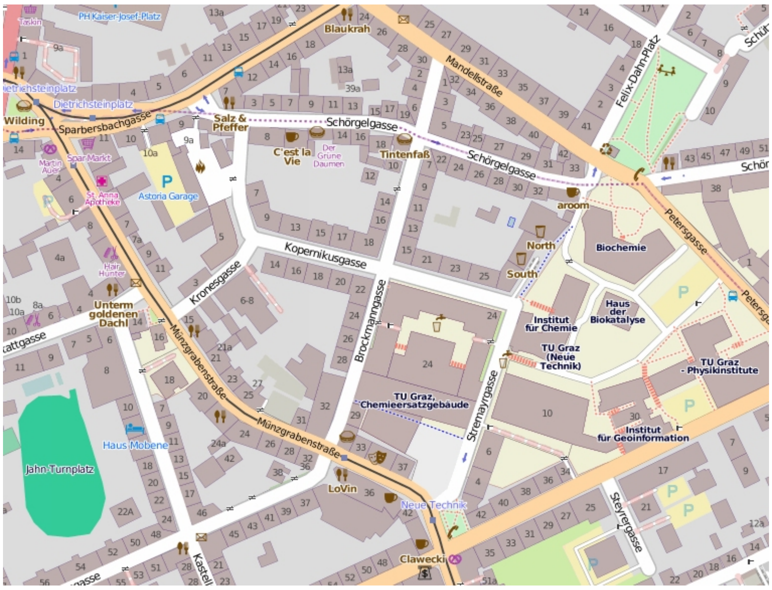


Figure 9.5: OSM data content (February 2012)

the mapping tools for OSM are based on graphic information. Therefore, the implementation has to be made by sighted people. On the other hand, sighted people do not exactly know the needs of blind and visually impaired people. In Chapter 7.5 and Chapter 9.4.1 the relevant information is summarized and was elaborated in close cooperation with blind and visually impaired people.

9.4 Data for Pedestrians

Pedestrians have other needs concerning route information than car drivers have. Especially for people with disabilities, the differentiation of the street side is very important. Moreover, they have to know about the occurrence of sidewalks in the streets. An important information for people with disabilities is the location of safe pedestrian crossings. Otherwise, especially for blind people, dangerous situations can occur. The idea is now, to model sidewalks and pedestrian crossings instead of streets. Figure 9.6 shows the difference between a street network and such a pedestrian network.

9.4.1 Detailed pedestrian network for people with disabilities

Besides information about the street side, people with disabilities are dependent on closer information about their environment. A solution for a pedestrian network for the navigation of blind people was generated in [54].

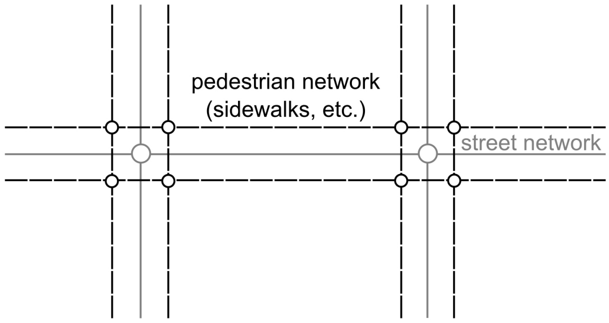


Figure 9.6: Street network in comparison with a more detailed pedestrian network

An extract of this pedestrian network is given in Figure 9.7. The network contains the following base objects:

- footpaths and sidewalks
- obstacles
- building entrances

This network was topologically extended for the underlying work and was enhanced with various attributes and information (like inclination of footpaths and sidewalks, structure of the path surface, obstacles, etc.). The new pedestrian network demonstrates an ideal network for guiding people with disabilities. The modified network contains three object types: *nodes, edges and points of interest*. In contrast to the network described in [54], obstacles are modeled as short edge-elements instead of polygon points. The new structure allows the inclusion of obstacles in multi-criteria optimization and with that, the minimization of obstacles within a route. The pedestrian network was derived from detailed data (*'Naturbestandsdaten'*) from the surveying office (*'Stadtvermessungsamt'*) Graz. The network was established in strong consideration of graph theory outlined in Chapter 2. The detailed path network in the test area consists of about 500 nodes and 600 edges.

9.4.2 Categories of information

Based on the questionnaire described in Chapter 7, categories for obstacles, points of interest, and the structure of the surface have been defined.

Obstacle Categories

Obstacles are split up into 16 categories. The following obstacle categories are important for people with disabilities, but are mainly focused on blind

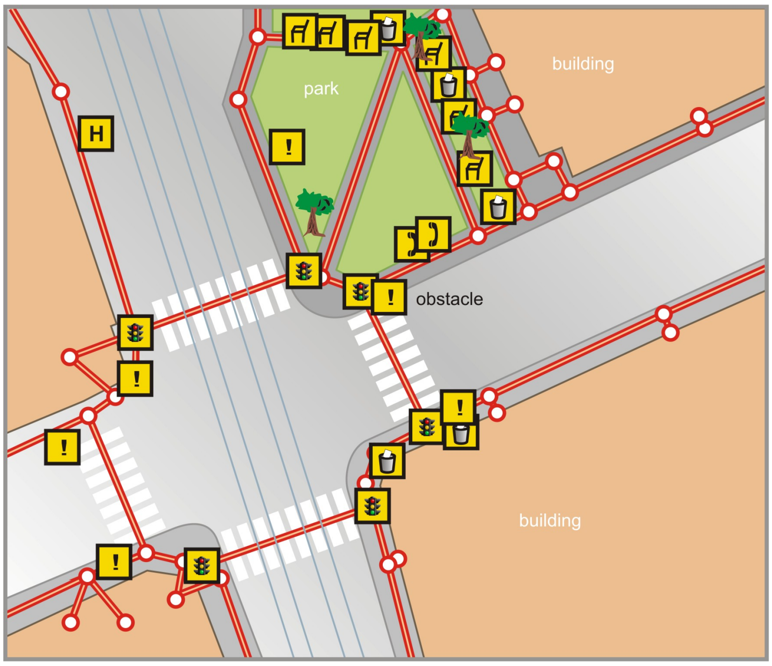


Figure 9.7: Pedestrian network for route planning

people. These objects can also be interesting information even for people who are not affected by obstacles.

- stations, which are not barrier-free
- stairs and steps
- narrow points
- crossings without traffic lights
- rails
- construction sites
- missing ramps
- stations with exit at the lane
- post boxes
- traffic signs
- cordons and gates
- pillars
- dustbins
- park benches
- parking ticket machine
- bicycle parking area

Points of Interest

Based on the questionnaire described in Chapter 7, the following POI-categories are considered in the route planner:

- bar or restaurant
- shopping facilitiy
- medical care
- pharmacy
- phone box
- toilet
- hotel
- disabled parking
- sightseeing
- museum
- acoustic traffic light
- cash point
- taxi
- tactile paving
- infrastructure facility
- tourist information
- elevator
- square or park
- fitness, sport and wellness
- service to repair wheelchairs
- station
- leisure
- pedestrian crossing

Categories of surface structure

The structure of the surface is important to know for wheelchair users and blind people. The following structures are distinguished:

- gravel
- cobblestone
- ground
- grass
- asphalt
- sand
- paving stones
- concrete plates

9.4.3 Test area

The test area for the detailed pedestrian network for people with disabilities is an area around the campus of Graz University of Technology, Austria (Figure 9.8). In this area all required information due to wheelchair users, blind people, and visually impaired people is mapped. With this network, an optimal solution for a pedestrian network is shown. The combination of the pedestrian network with the less detailed OSM network and the public transport network enlarges the test area to the whole city of Graz.

Figure 9.8: Test area for the detailed path network

9.4.4 Comparison of OpenStreetMap street network data and detailed data

Detailed data ('Naturbestandsdaten') from the surveying office ('Stadtvermessungsamt') Graz are based on terrestrial geodetic measurements with an accuracy of a decimeter-level. These data shall serve as reference data for an accuracy verification of the OSM data in the test area. In Figure 9.9 and Figure 9.10 the detailed data and OSM street network data are put on top of each other. The dense gray lines represent the OSM street network (center lines of streets), whereas the thin lines show the mapped 'Naturbestandsdaten'. Sometimes they fit well, but there are also street crossings with a mapping difference of over twenty meters. Such a mapping difference is shown in Figure 9.9 and marked with a circle. The dashed line represents the required position of the edge pointing to the north-east. The difference can occur due to source inaccuracy, sensor characteristics, actuality or errors, [16].

In Figure 9.10 a larger area is visualized to get an impression of the different accuracy within the OSM network. It is shown that the absolute accuracy varies across the map and does not contain a systematic error. Due to the local differences, global correction parameters cannot be derived. Moreover, the 'Naturbestandsdaten' is an expensive dataset only available in local areas and cannot be used as reference for global areas.

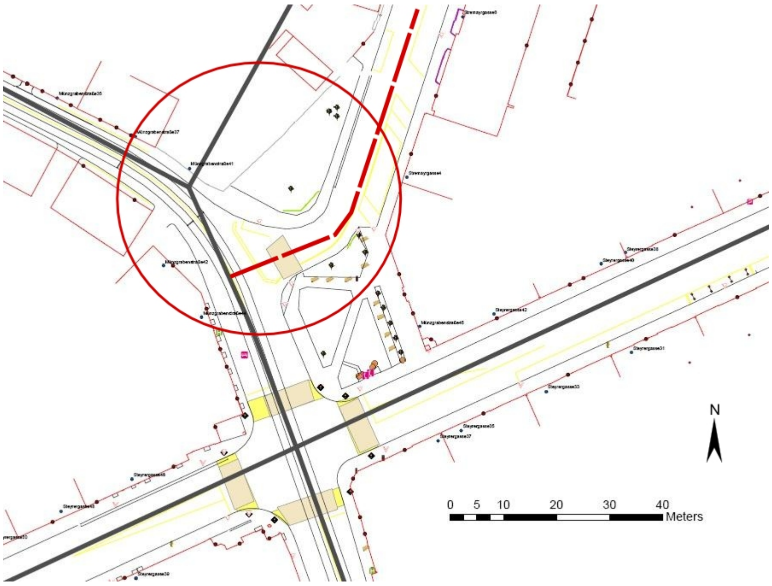


Figure 9.9: Comparison of OpenStreetMap street network data and detailed data from the land surveying office

9.5 Public transport network and timetables

The third network used is the public transport network from the city of Graz, Austria. Whereas in route planning for vehicle navigation the public transportation system is normally not included, it plays an important role for pedestrian route planning. Typical for this data is the necessity of conversion before it can be used in algorithms. In public transport, data is available as time schedules containing stops and departure times. A first step is the transformation of schedules to a graph model by generating a time-expanded node-edge structure from timetables (see Chapter 6). A second step is the connection of different lines by transfer edges and transfer nodes which reflect transfer times and additional distances (see Chapter 10.2.4). The public transport network and the generation of the network from original timetable data of the public transport company *Holding Graz Linien* is explained in Chapter 10.

9.6 Integration of different data sources

Nowadays, the existence of various different data sets containing heterogeneous information makes a wide range of useful information available for different kinds of applications. However, the information needed for the development of a specific application is often distributed in different data sources. This fact leads to the idea of integrating spatial data from different

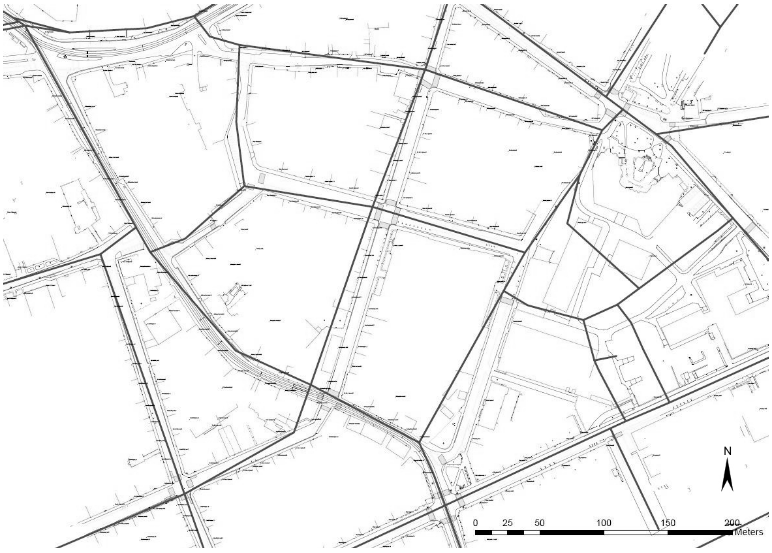


Figure 9.10: Comparison of OpenStreetMap street network data and detailed data from the land surveying office

sources to take advantage of data content and quality of more than one data set. As a consequence, high information content can be achieved.

9.6.1 Multi-scale and multi-modal data model

Data models of different scales and levels of detail are called multi-scale data models. Multi-modal data include different transportation networks like streets, footpaths, public transport and so on. The connection of different networks is made by defining connection nodes which generate a multi-modal network, [32]. The data combination in this work is based on a combined multi-scale and multi-modal data model.

The integration of different data sources in a multi-scale and multi-model data model comes up with some issues that have been considered:

- File conversion: Data sources are often stored in different file formats. For the integration of heterogeneous data, the conversion in an uniform file format is necessary.

- Coordinate transformations: Data with different coordinate systems have to be transformed in a homogeneous coordinate system.

- Topological connection: Corresponding objects of different data sets have to be merged for the establishment of an overall connected data source.

9.6.2 Data integration for route planning

Practically, the following data sources explained in Chapters 9.3, 9.4 and 9.5 have been combined and form the geographical database for the route planner:

- OSM
- Detailed pedestrian network for people with disabilities
- Public transport network

Coordinate systems

Within these data sources, three different coordinate systems are in use: ‘Bundesmeldenetz’ (BMN), ‘Gauß-Krüger’ (GK) and the World Geodetic System 1984 (WGS84). BMN and GK are local Cartesian coordinate systems, whereas WGS84 is a worldwide geodetic reference system. In Table 9.1 the coordinate systems of the used data sources are shown.

Table 9.1: Coordinate systems of used data sources

data base			
‘Naturbestand’		GK	
Pedestrian network (DET)		GK	
OpenStreetMap (OSM)			WGS84
Stations (PTN)	BMN	GK	

For coordinate transformations and file conversions the software FME[7] was used. FME is a GIS processing application with a flowchart-based interface. It is a very powerful tool for automating repetitive tasks, file conversion, coordinate transformation, and the processing of large data sets. With the support of over 200 data formats it is easy to handle different data sources. It even provides the conversion of OSM data in the shape-file format. For coordinate transformation, lots of different coordinate systems are available.

Topological connection

For the topological connection of different networks a *connection graph* (CON) was established. This graph represents the interconnections between the subgraphs of different networks to achieve a connected network. The following connections exist:

[7] http://www.safe.com/

- OSM-DET: OSM data has been visually connected using a Desktop GIS. The integration of OSM-DET is an integration between data of two different scale ranges. At the area, where DET is available, OSM data is skipped and corresponding streets and street crossings are connected. The detailed pedestrian network has a higher resolution and thus the accuracy decreases at the connection points to the OSM network. Due to the high resolution of the detailed pedestrian network, two edges of DET (sidewalks of the street) empty into one edge in the OSM network (center line of the street). Figure 9.11 shows one of several connection points.

- DET-PTN: The networks have been connected directly in the database by connecting the corresponding stations in both networks. The nodes with station attributes in DET are connected with the corresponding stations in PTN.

- PTN-OSM: The combination represents the topological connection between the coordinates of the stations in OSM and the corresponding PTN stations

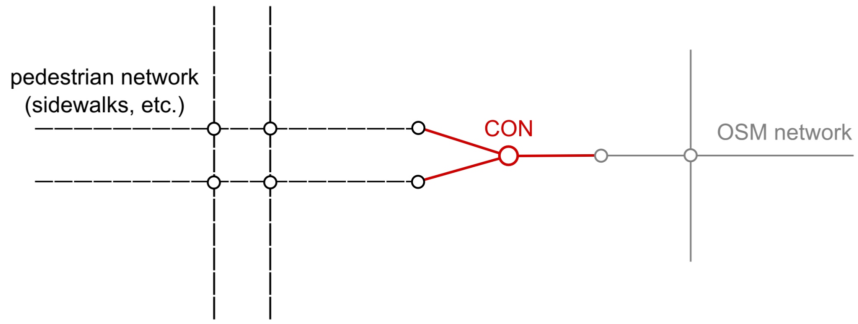


Figure 9.11: Connection of OSM network and pedestrian network using a connection graph with transfer nodes

9.7 User generated data

Due to the lack of information for people with disabilities in common data sources, the possibility of the addition of data from the user itself is essential. This acquires new knowledge to the data source and facilitates the communication between users. Moreover, individual needs and interests can be added to the database. However, user generated data content needs special administration effort. Tools for data integration have to be provided and the user generated content has to be proofed and managed. From the developer's point of view, the integration of user generated information is much easier for point information, than it is for topological information (e.g., streets or footpaths for navigation). Point information is represented by a pair of coordinates and can stand for its own. For the expansion of a path network for route planning, the skill required for mapping is much more advanced. By mapping new streets or paths, the topological connection to existing objects has to be established to provide a connected graph for route planning.

9.7.1 Points of Interest

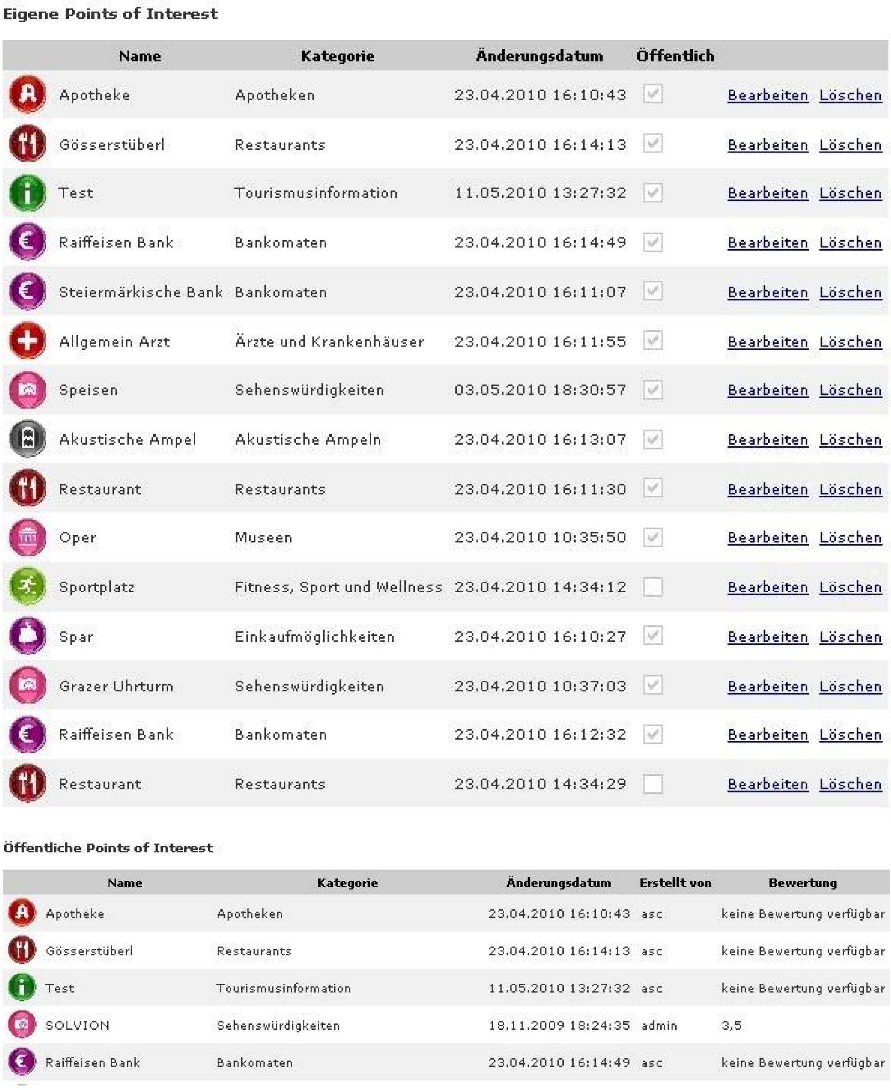
Points of interest are additional information and are visualized on the map as point information. In the developed route planner, the integration of user generated data is realized for such POI. Creating POI and sharing them with other users is an important tool for people with disabilities to add specific needs and information. User can create new POI directly in the web application. Furthermore, they can make comments on existing POI. Moreover, the addition of obstacle warnings by using a special POI category is possible. Dependent on the category, different symbols are displayed on the map. By moving over a POI, a window with more detailed information is shown. These information can also be complemented by users.

Creation of Points of Interest

When a new POI is placed on the map, a window for entering name, category and comments appears. Moreover, the sort of the POI (private or public) can be chosen. Whereas private POI are personal, public POI are available for all users. After saving, the new POI it is displayed on the map with the corresponding category symbol.

POI administration

As previously mentioned, user generated data has to be administrated. In the present application, users can administrate their POI and add comments to each POI (Figure 9.12) themselves. POI are divided into public POI seen by all users and private POI. Private POI (first table in Figure 9.12) are personal and can only be seen, set public, modified, or deleted by the user they belong to. Public POI can be seen and modified by all users. Public POI can be set private or deleted by the user, who created the public POI. Furthermore, all POI can be managed or deleted by the system administrator.



Eigene Points of Interest

Name	Kategorie	Änderungsdatum	Öffentlich	
Apotheke	Apotheken	23.04.2010 16:10:43	✓	Bearbeiten Löschen
Gösserstüberl	Restaurants	23.04.2010 16:14:13	✓	Bearbeiten Löschen
Test	Tourismusinformation	11.05.2010 13:27:32	✓	Bearbeiten Löschen
Raiffeisen Bank	Bankomaten	23.04.2010 16:14:49	✓	Bearbeiten Löschen
Steiermärkische Bank	Bankomaten	23.04.2010 16:11:07	✓	Bearbeiten Löschen
Allgemein Arzt	Ärzte und Krankenhäuser	23.04.2010 16:11:55	✓	Bearbeiten Löschen
Speisen	Sehenswürdigkeiten	03.05.2010 18:30:57	✓	Bearbeiten Löschen
Akustische Ampel	Akustische Ampeln	23.04.2010 16:13:07	✓	Bearbeiten Löschen
Restaurant	Restaurants	23.04.2010 16:11:30	✓	Bearbeiten Löschen
Oper	Museen	23.04.2010 10:35:50	✓	Bearbeiten Löschen
Sportplatz	Fitness, Sport und Wellness	23.04.2010 14:34:12		Bearbeiten Löschen
Spar	Einkaufmöglichkeiten	23.04.2010 16:10:27	✓	Bearbeiten Löschen
Grazer Uhrturm	Sehenswürdigkeiten	23.04.2010 10:37:03	✓	Bearbeiten Löschen
Raiffeisen Bank	Bankomaten	23.04.2010 16:12:32	✓	Bearbeiten Löschen
Restaurant	Restaurants	23.04.2010 14:34:29		Bearbeiten Löschen

Öffentliche Points of Interest

Name	Kategorie	Änderungsdatum	Erstellt von	Bewertung
Apotheke	Apotheken	23.04.2010 16:10:43	asc	keine Bewertung verfügbar
Gösserstüberl	Restaurants	23.04.2010 16:14:13	asc	keine Bewertung verfügbar
Test	Tourismusinformation	11.05.2010 13:27:32	asc	keine Bewertung verfügbar
SOLVION	Sehenswürdigkeiten	18.11.2009 18:24:35	admin	3,5
Raiffeisen Bank	Bankomaten	23.04.2010 16:14:49	asc	keine Bewertung verfügbar

Figure 9.12: POI administration

Chapter 10

Multi-criteria route planning in combined networks

Route planning for people with disabilities makes high demands on the path network. Including only one sort of network (e.g., public transport network) is not sufficient for these user groups. Instead, a combined footpath and public transport network is established using three different networks (see Chapter 9.6). The combination of different networks affects the route planning algorithm in its complexity and leads to multiple optimization criteria. Furthermore, user specific restrictions can be introduced and influence the route optimization. At first, route planning in combined networks is introduced in this chapter. Afterwards, the used modeling of public transport for route planning is described in detail. Finally, the implementation of multi-criteria optimization in combined networks based on the theoretical concept in Chapter 4 and Chapter 5 is discussed.

10.1 Route planning in multi-modal networks

The combination of different data sources increases the complexity of the network for route planning. Such a multi-scale and multi-model network leads to a multi-level routing where two model concepts can be distinguished: the homogeneous routing concept and the hierarchical routing concept, [31].

10.1.1 Homogeneous routing concept

This concept allows route planning within one homogeneous path network. Different networks are joined to one large graph. This allows an easy implementation of the route planning algorithm, but has its disadvantages in performance. By applying Dijkstra's shortest path algorithm in single-criterion optimization on the homogeneous routing concept, one large circular search space results having the start node in its center (see Figure 10.1). Recall that the single-criterion algorithm terminates once one path (the optimal) to the target node is found.

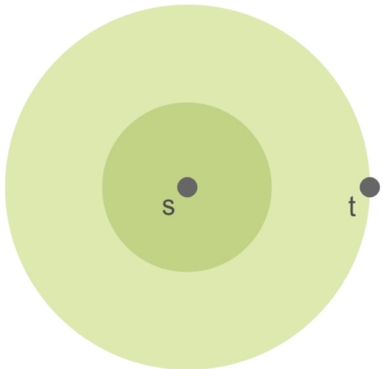


Figure 10.1: Circular search space

The search space in single-criterion optimization can be reduced by applying a bidirectional search on the network. In that case, a simultaneous search from the start node forward to the end node and from the end node backward to the start node is done (Figure 10.2). This results in two circular search spaces with each having half the radius of the search space in Figure 10.1. Note that one circle with radius $r(s,t)$ has twice the area of two circles with half the radius $r(s,t)/2$. Therefore, in best case, the bidirectional search can speed up the algorithm twice, [66]. However, to guarantee an optimal search result, an overlapping area of the two search spaces has to be considered, [31]. This is even more important in multi-criteria optimization, since more than one efficient path between a pair of nodes exists (see Chapter 3). As a consequence, in multi-criteria optimization no speed up can be aimed with a bidirectional search, without giving up Pareto optimal solutions. Therefore, the bidirectional search is not implemented in the route planner.

In the combined network consisting of the pedestrian network, OSM network, and public transport network (see Chapter 9.6), all networks are considered simultaneously in the route planning algorithm in a circular space. This means, if the algorithm is searching in the public transport network, all other networks are regarded as well. This is a very time consuming operation.

10.1.2 Hierarchical routing concept

In this concept, network parts containing a specific path type are ordered hierarchically. The route planning is done in different network parts separately. First, the search is performed in some local area around the start node in a lower level network. Then the algorithm switches to a sparse

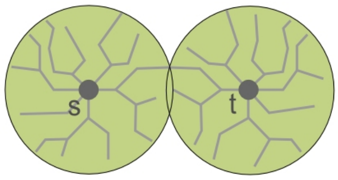


Figure 10.2: Bidirectional search in homogeneous networks

network of higher level for covering large distances in a sparse graph. By reaching the destination area, the algorithm goes back into the path network. The network parts are joined via connection nodes and edges. The selection of the optimal connection nodes is the challenging task in this concept. The advantage lies in the fast runtime due to the smaller number of nodes the algorithm has to proceed compared to the combined network. Let us consider again the three network types: pedestrian, OSM, and public transport network. Then, the public transport network is suited for a higher level network with stations forming connection nodes to the path network. In Figure 10.3, the hierarchical structure of the combination of path network and public transport network is given.

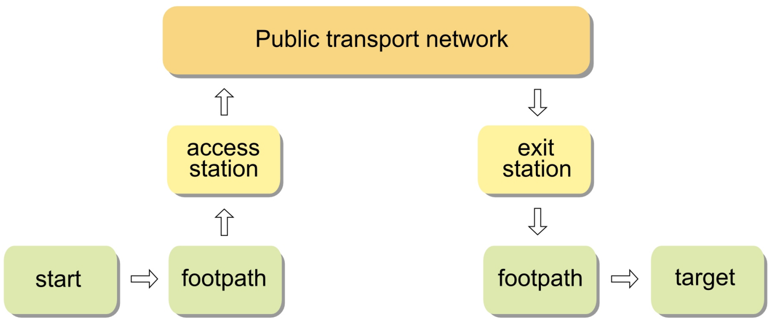


Figure 10.3: Hierarchical scheme of combined networks

For route planning in hierarchical networks, bidirectional search is essential. First, the optimal path in a local area from the start node to connection nodes of the higher level network is computed. At the same time, a backward search from the destination node to the connection nodes of the higher level network is done. Finally, the higher level network is proceeded regarding the calculated paths in the lower level networks. Figure 10.4 illustrates a hierarchical search in combined networks. Multi-criteria optimization in multi-modal networks is not well discussed in the literature and is a challenging task, [13]. In multi-criteria optimization a hierarchical concept is hard to realize, since the multi-criteria algorithm does not terminate, if one optimal path to the target node is found (see Chapter 3), [64]. Therefore, the homogeneous concept was chosen for multi-criteria optimization.

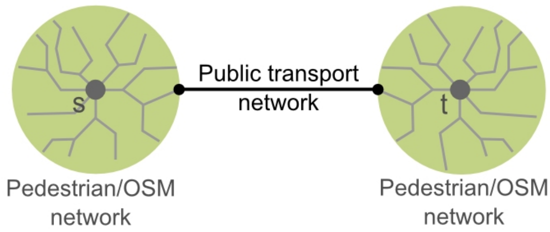


Figure 10.4: Hierarchical search in combined networks

10.2 Modeling of public transport networks

Public transport networks consist of a topological network and timetables. The use of a public transport network in a route information system requires special modeling of timetables for optimization in route planning. Existing public transport information systems are *HAFAS* (HaCon Fahrplan-Auskunfts-System), which is a trademark of the HaCon (Hannover Consulting) Ingenieurgesellschaft mbH in Germany and *EFA* (Elektronische Fahrplanauskunft) by Mentz Datenverarbeitung GmbH, Munich, Germany. However, these systems are rarely containing information concerning people with disabilities, [67, 68].

10.2.1 Raw timetable data

Timetable information used in the present work is provided by the public transport company of Graz (*Holding Graz Linien*) and contains all streetcar routes and some bus routes from the city of Graz.

Timetables are structured in text-files and are split in up to six timetable files for each route. These six files consist of:

- three day-types: There is one file for each existing day-type. These types are divided into three periods: Monday to Friday, Saturday, and Sunday or public holidays.

- two directions: Each day-type is stored for both directions.

Besides different files for day-types, the whole timetable with all its files is existing multiple times for different periods over the year. There are Christmas timetables with extended Saturday, school-holiday timetables, and night plans as a few examples. An extract from raw timetable data is shown in Figure 10.5. It represents the timetable information from streetcar line number six in the raw timetable structure of the public transport company *Holding Graz Linien*. In Figure 10.6 the data set entries are explained. The timetable contains the line-number, the direction, the sequence of the stations, the day-type, the course number, and the departure and arrival times at the stations as well as the length between two stations. The timetable could not be used directly for route planning. It has to be transformed in a graph model, as theoretically explained in Chapter 6.

```
FF6         1
FH0170371  - 0231   - 0221   - 0211   - 0201   - 0191   - 0183   - 0361   - 0681
FK2 00602000029
FA 04.59 04.59 00000 05.01 05.01 00306 05.02 05.02 00309 05.03 05.03 00302
FK1 00609000029
FA 04.59 04.59 00000 05.01 05.01 00306 05.02 05.02 00309 05.03 05.03 00302
FK1 00606000029
FA 05.19 05.19 00000 05.21 05.21 00306 05.22 05.22 00309 05.23 05.23 00302
FK2 00604000029
FA 05.19 05.19 00000 05.21 05.21 00306 05.22 05.22 00309 05.23 05.23 00302
FK1 00601000029
FA 05.39 05.39 00000 05.41 05.41 00306 05.42 05.42 00309 05.43 05.43 00302
FK2 00606000029
FA 05.39 05.39 00000 05.41 05.41 00306 05.42 05.42 00309 05.43 05.43 00302
FK1 00609000029
FA 05.59 05.59 00000 06.01 06.01 00306 06.02 06.02 00309 06.03 06.03 00302
FK2 00602000029
FA 05.59 05.59 00000 06.01 06.01 00306 06.02 06.02 00309 06.03 06.03 00302
```

Figure 10.5: Raw timetable data from *Holding Graz Linien*

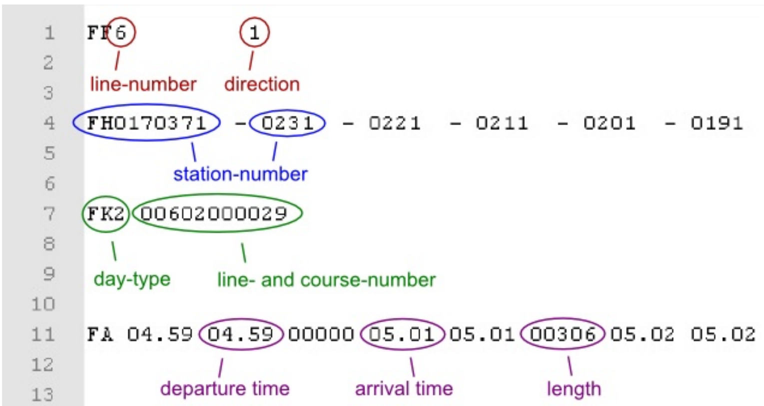


Figure 10.6: Raw timetable data with data-set entries

10.2.2 Time-expanded model

For route planning in public transport networks, timetables have to be transformed in a topological network. Due to multi-criteria optimization the time expanded model was chosen (see Chapter 6.1). This model was established with real timetable data from the public transport company *Holding Graz Linien* described in the previous section. For each vehilce a connection between two stations is represented as an edge element e of the following structure:

$$e = (Z, S_1, S_2, t_d, t_a, N, C, d, Dt, dist, D, T).$$

Z...edge-ID
S_1...station 1
S_2...station 2
t_d...departure time at S_1
t_a...arrival time at S_2
N...line-number
C...course number
d...direction
Dt...day-type
dist...distance
D...duration
T...transfers

Figure 10.7 shows a table with real timetable data from the *Holding Graz Linien* in the described structure. The time-expanded network consists of about 70.000 nodes and 75.000 edges.

	ID	station1	station2	td	ta	number	course	direction	day	distance	duration	transfer
1	148	301	3681	309	310	1	00152000069	1	3	401	1	0
2	149	3681	291	310	311	1	00152000069	1	3	424	1	0
3	150	291	281	311	312	1	00152000069	1	3	339	1	0
4	151	281	271	312	313	1	00152000069	1	3	301	1	0
5	152	271	261	313	314	1	00152000069	1	3	257	1	0
6	153	261	251	314	315	1	00152000069	1	3	398	1	0
7	154	251	241	315	316	1	00152000069	1	3	352	1	0
8	155	241	371	316	318	1	00152000069	1	3	526	2	0
9	156	371	231	319	321	1	00152000069	1	3	306	2	0
10	157	231	221	321	322	1	00152000069	1	3	309	1	0
11	158	221	211	322	323	1	00152000069	1	3	302	1	0
12	159	211	201	323	324	1	00152000069	1	3	463	1	0
13	160	201	191	324	326	1	00152000069	1	3	348	2	0
14	161	191	181	326	328	1	00152000069	1	3	538	2	0

Figure 10.7: Timetable of *Holding Graz Linien* in time-expanded structure for route planning

10.2.3 Extensions for time-constrained optimization in real public transport networks

For multi-criteria time-constrained optimization extensions concerning model and algorithm have to be made.

- Model: Besides timetables, *transfer nodes* and *transfer edges* have to be introduced in the graph model.

- Algorithm: Due to the huge graph caused by copying all nodes for every time event, a predefined *time interval* can reduce the search space of the route planning algorithm. For an inner-city ride this interval is set to a few hours.

Figure 10.8 represents a flowchart of a public transport itinerary.

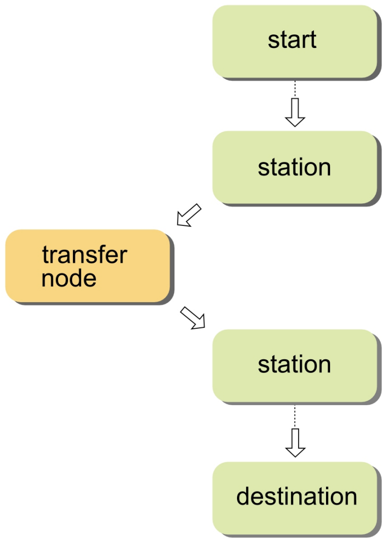


Figure 10.8: Public transport scheme, modified from [62]

10.2.4 Transfer edges and transfer nodes

Transfer edges connect stations where a transfer is possible. Transfer edges are modeled as time-independent edges. The 'transfer'-attribute marks these edges as a transfer and is set to value 1 for transfer edges and value 0 for elementary connections in the timetable.

Let n be the number of nodes per station. The total number of *directed transfer edges* (resp. arcs, see Chapter 2) results from $n(n-1)$. Transfer edges can be reduced by inserting a transfer node. Transfer nodes are time-independent as well. Note, that attributes for transfer and duration must not be set in both directions. They are only set to arcs pointing to the transfer node. Otherwise, if set in both directions, the shortest path algorithm would count the values twice. The transfer node model contains only $2n$ transfer arcs compared to the transfer edges model with $n(n-1)$ arcs. Therefore, the transfer node model is realized in the present work. Figure 10.9 shows the comparison between directed transfer edges and an inserted transfer node.

Transfer time

In real public transport networks, the transfer time is not negligible. In realistic itineraries the transfer time varies for each station and is not known before. Moreover, it also differs for various kinds of user groups. In a first step, the transfer time is set to a constant value for each different user group, since wheelchair users need more time for transfers than blind or visually impaired people. These different transfer times are set in the duration attribute of the transfer edges. The duration of travel from one station to another station is generated from t_a-t_d (see Chapter 6.3).

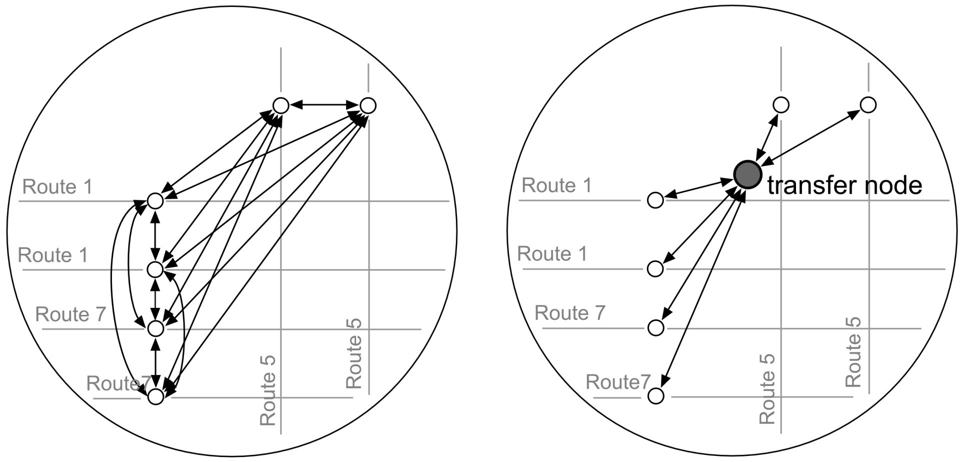


Figure 10.9: Left: Transfer edges connect stations directly with $n(n-1)$ arcs, Right: A transfer node reduces the number of arcs ($2n$ arcs)

10.3 Multi-criteria time-constrained route planning in combined networks

One of the most innovative aspects of the developed route planner is the possibility of comparing various routes in pre-trip training. The pre-trip training provides the comparison of different routes based on varying optimization criteria directly with each other. In contrast to common route planners, it is not necessary to calculate the route various times with different optimization preferences. Instead, different route suggestions are gained within one calculation. To realize this comfortable feature, multi-criteria optimization was chosen for implementation with the advantage to optimize miscellaneous criteria in one optimization process and consider them against each other.

Multi-criteria optimization was theoretically introduced in Chapter 4 and Chapter 5. This concerns the definition of Pareto optimality and dominance, general solution methods of multi-criteria optimization using an extended Dijkstra algorithm, as well as concepts for multi-criteria route planning. This chapter gives insights in the implementation of multi-criteria optimization in the route planner for people with disabilities and describes which of the introduced concepts in Chapter 5.3 have been used.

10.3.1 Optimization criteria

In contrast to single-criterion optimization, more than one criterion is considered in multi-criteria optimization. In case of combined networks, each network has to be affected by at least one optimization criterion. In route planning, a common criterion is time or distance. In addition to one optimization criterion for each network, supplemental criteria for each network can be defined to suggest adequate and user-specific routes. For the realization of the route planner, three optimization criteria have been chosen:

- time
- number of transfers
- number of obstacles

The common criteria for all three networks (public transport network, OSM network and pedestrian network) is time. With this criterion the total time of the route is minimized. In addition, for the public transport network the minimization of the number of transfers is essential to provide a reasonable route (see the optimization problems in Chapter 6.5). A further criterion due to the pedestrian network is the number of obstacles, which is a criterion especially selected for people with disabilities.

10.3.2 Basic definitions on multi-criteria optimization

In route planning, the feasible set $\mathcal{X}$ consists of all possible paths in the combined network. The three optimization criteria are defined as $f_1(x) =$ time, $f_2(x) =$ number of transfers and $f_3(x) =$ number of obstacles. The overall optimization problem is stated as:

$$\min_{x \in \mathcal{X}} (f_1(x), f_2(x), f_3(x))$$

The challenge is now to find the most interesting Pareto optimal paths for this optimization problem. Therefore, optimization methods introduced in Chapter 5 are applied. Due to route planning, only strict Pareto optimal solutions are considered, since other solutions would not be the favorite choice of a user (see Chapter 5.1.1).

The multi-criteria optimization problem can be solved using an extended Dijkstra's shortest path algorithm with expanded node label vectors. In contrast to single-criterion optimization the edge costs are not scalar values, but k-dimensional cost vectors for k criteria (see Chapter 4.5). In case of our three predefined optimization criteria, this leads to 3-dimensional cost vectors in the graphs. In networks, where one criterion is not affected, the value of this criterion in the cost vector is zero. E.g., in the public transport network the value for the number of obstacles in the cost vector is null.

Contrary, in the OSM and the pedestrian network the cost value for the number of transfers is null.

10.4 Optimization methods for multi-criteria time-constrained route planning

The goal is now to find subsets of Pareto optimal paths to provide a choice to the user. The following solution methods for the present multi-criteria optimization problem have been chosen from possible methods described in Chapter 5.

10.4.1 Multi-criteria optimization with lexicographic optimality

The multi-criteria optimization with lexicographic optimality is a method for the calculation of a single Pareto optimal path. This method was just implemented for reference to compare it with other solutions, provided by a method which in contrast generates subsets of Pareto optimal paths. The multi-criteria optimization with lexicographic optimality provides a better runtime with the restriction that some interesting alternatives get lost (see Chapter 5.2.1 for details). For a reasonable runtime of the route optimization algorithm the non-scalar approximation method of the lexicographical minimum (see Chapter 4.3.2) has been applied. Another method for calculating a single Pareto optimal solution is the weighted sum method, which is used in a routing project in [73]. A disadvantage of the weighted sum method is, that criteria cannot be considered dependent on each other.

10.4.2 Multi-criteria optimization with restrictions on the criteria including additive values

In the present work, a more sophisticated method is realized additionally to the lexicographic optimality and the weighted sum method. The multi-criteria optimization with restrictions on the criteria including additive values generates subsets of Pareto optimal paths. In this method, restrictions on the criteria from individual user profiles are considered (see Chapter 5.3.2) and criteria are regarded dependent on each other. Generally speaking, restrictions speed up the multi-criteria algorithm, since node label vectors are deleted, if a criterion value exceeds the predefined upper or lower bound. This decreases the number of node label vectors and therefore reduces the search space of the algorithm. However, it can occur, that no possible path can be found and the algorithm terminates without a solution. This situation has to be reported to the user and they have to relax their restriction values. Additionally to the introduction of restrictions, the order of the criteria in the cost and label vectors is changed to $f_1(x)$ = number of transfers, $f_2(x)$ = time and $f_3(x)$ = number of obstacles. Remember that the order of the criteria influences the optimization output as demonstrated in Chapter 5.3.1. The reordering has been made since a minimum number of transfers is more important for people with disabilities than time. However, the relation of transfers and time can be influenced by using restrictions with additive values as one will see later on.

Restrictions

Besides additional optimization criteria user specific restrictions influence the optimization. Such restrictions can affect optimization criteria (see Chapter 5.3.2), the definition of dominance, or single edges. Restrictions can be set up at the user profile by the definition of a maximum value.

Among others, three main restrictions are included:

- max. walking distance/time to a station (restriction concerning the pedestrian network or OSM network)

- max. number of transfers (restriction concerning the public transport network)

- max. slope/incline of single edge elements (restriction concerning the pedestrian network or OSM network)

Restrictions differ in the implementation method. Three differences can be distinguished: Some restrictions (the max. walking distance/time and the max. number of transfers) influence the optimization criteria and therefore the node label vectors. Labels are deleted in the modified Dijkstra's algorithm, if the threshold value is exceeded. Additive values (e.g., for the relation between the number of transfers and time) influence the definition of dominance as explained in Chapter 5.3.2. Restrictions like the max. slope influence single edge elements. This restriction does not influence the route optimization algorithm. Edge elements are checked against the max. value before the optimization process. Edge elements which exceed the threshold value are eliminated before the route calculation.

Restrictions with additive values

Restrictions with additive values influence not only one criterion, but also consider criteria against each other. This method is implemented for the criteria time and number of transfers. A label with less transfers dominates, if and only if its time component is max. δ minutes bigger (see Chapter 5.3.2) for details. The additive value δ may be set individually in the user profile.

Restrictions on edges

Some restrictions do not influence the optimization criteria, but make edges unavailable or even not passable (e.g., times when a vehicle is not available, restrictions on vehicles, or specific obstacles). Corresponding edges have to be eliminated or disabled for route planning. In this context the following restrictions are considered:

- Vehicles not available at given time: The availability of vehicles depends on days and times. A public transport vehicle is not available the whole day round. Moreover, some vehicles are not available on each day. This circumstance has to be considered in the modeling of the time-expanded network.

- Accessibility of vehicles: Three categories of accessibilities can be distinguished:

  – full accessible: low-floor vehicle
  – partial accessible: partial low-floor vehicle
  – not accessible

  The category of accessibility can be identified using the vehicle-number (course number). For realization, a problem occurs: the course numbers of daily timetables differ from the timetable of a period stored in the database. Daily timetables are set by *Holding Graz Linien* one day before usage. Only at this daily timetables the course numbers are in accord with the vehicles driving on that course. The implementation of daily timetables in real time is not yet possible because of technical issues at *Holding Graz Linien* but the developed software is designed in a way that timetables would be exchangeable at any time.

- Edges with maximum incline: The inclination of the path is especially relevant for wheelchair users. The common maximum value for inclination is 6% concerning wheelchair users. However, this value varies for various types of wheelchairs. Using an electrical wheelchair allows a higher value than moving with a manual wheelchair. The maximum value for inclination is also depending on the type of disability by using a manual wheelchair. The crucial fact is, how many force the user can bring up with his hands. Therefore, this bound can be set in the user profile individually. Edges exceeding this bound value have to be disabled in route planning.

- Bus or Tram: Some user prefer a single type of public transport vehicle. In the city of Graz two types of vehicles are available. These are buses or trams. The users are able to define the vehicle type in their individual profile. So they can decide if they would like to use either buses or trams or both of them.

- Edges with impassable obstacles: Obstacles can be divided into passable obstacles and impassable obstacles. For passable obstacles (e.g., traffic signs,...) a warning is given in the route description (see Chapter 11.6). However, for impassable obstacles the corresponding edges have to be eliminated or disabled in route planning.

  Impassable objects are:

  – stairs (wheelchair users): Stairs without ramps can not be passed by wheelchair users.
  – narrow points (wheelchair users): Footpaths with narrow points smaller than a wheelchair can not be used by wheelchair users. Narrow points are not impassable for blind people, but very dangerous.

  Types of obstacles depend on personal behavior and mobility. E.g., stairs may also be an obstacle for some blind people, while others can detect them with their white cane. Due to these individual differences, the obstacle categories relevant for route planning can be set in the user profile.

Chapter 11

Application

This chapter describes the architecture and functionalities of the developed route planner. The route planner is realized as a web application and provides adequate routes for people with disabilities. Another part of this chapter is the explanation of route description. In more detail, this involves the calculation of turn information and additional information dependent on the network type (e.g., obstacle warnings, information about public transport, information about the closer surrounding, etc.). Finally, a validation of the provided route information and the performance of the application are given.

11.1 Software architecture

Figure 11.1 shows an overview of the software architecture. The main part is a web server containing the database, the Route4you services and the web application. All calculations are accomplished at the server. Only the graphical user interface is rendered at the client. This approach has shown to be computationally efficient. The only requirement for the user client is a web browser with a freely available Silverlight plugin (see Chapter 11.2.3). The web server and the client need to be connected to the Internet.

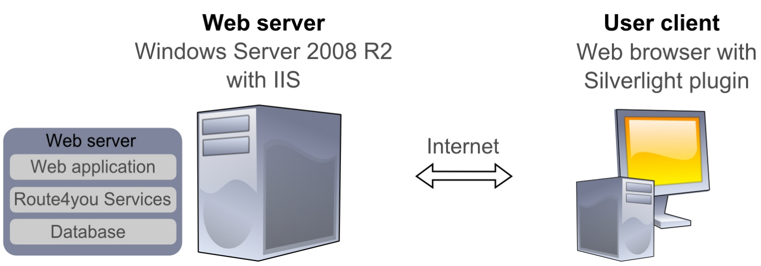


Figure 11.1: System overview

11.2 Technologies

The platform of the web application is a Windows Server 2008 R2 system with a 2,4 GHz Intel Xeon CPU and 6 GB RAM. This server acts as the web server as well as the database server with the advantage of fast data exchange. The web server application is Microsoft Internet Information Services (IIS 7.5) which is necessary for sharing applications, services, documents, and files in the network. The developed web application interacts with Microsoft Sharepoint. The graphical user interface (GUI) of the application is a combination of ASP.NET (Active Server Pages .NET) WebForms, Windows Presentation Foundation (WPF) and Silverlight. ASP.NET is a web application framework for building dynamic web sites, web applications, and web services. The Silverlight control allows a user interaction within the web page and provides objects, methods, and events for displaying maps. The communication between the application and the Route4you services is done using Windows Communication Foundation (WCF). The Route4you services trigger calculations like route planning and access the database. The database is managed with Microsoft SQL Server 2008 and contains geographical data, application data and user profiles. Figure 11.2 illustrates the application parts with their corresponding technologies. In the following, these technologies are described in more detail.

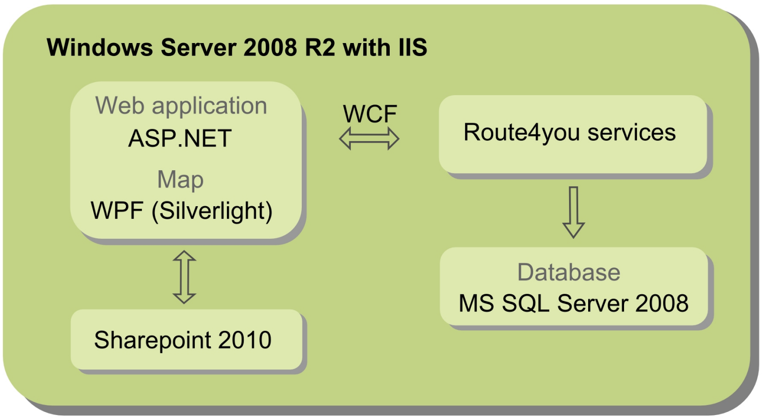


Figure 11.2: Technologies

11.2.1 Internet Information Services

IIS[1] is a web server application to provide and host ASP.NET web applications and web services in the Internet. IIS 7.5 is included in Windows Server 2008 R2 and supports HTTP, HTTPS, FTP, FTPS, SMTP and NNTP

[1] www.iis.net

which are different transfer protocols for hypertext, files, mail, and network news.

11.2.2 Windows Presentation Foundation

WPF is part of Microsoft .NET. WPF is a graphical subsystem for rendering user interfaces in Windows-based applications developed by Microsoft which clearly separates GUI from business logic. WPF implements XAML (Extensible Application Markup Language) to define the GUI by linking various GUI elements. WPF applications can be established as standalone desktop programs or can be used as an embedded object in a website, [49].

11.2.3 Microsoft Silverlight

Silverlight[2] is a powerful development platform for creating interactive applications. Silverlight, also known as WPF/E (Windows Presentation Foundation Everywhere), is a subset of WPF and focuses on web applications. This web application framework integrates multimedia, computer graphics, animation, and interactivity into a single runtime environment. Silverlight is comparable to Adobe Flash, but has its focus on a user interface (UI) object model and less on animation. Silverlight also supports screen reader, which is a crucial fact dealing with an application for blind people. Silverlight supports the accessibility definitions of WCAG 2.0 described in Chapter 8.2.3. The UI Automation API (application programming interface) allows text equivalents for pictures, text boxes, etc., and the definition of the tab sequence which is content of the WCAG 2.0. Silverlight has to be installed as a web browser plug-in by every user of the route planner and is freely available.

11.2.4 Windows Communication Foundation

WCF is also part of Microsoft .NET and allows the communication between applications (service endpoints) via different protocols. In this application, WCF is used to communicate with Route4you services using web services. The communication is implemented asynchronously which means, one service is not blocked while waiting for the response of another service.

11.2.5 Microsoft Sharepoint 2010

Microsoft Sharepoint is based on IIS and is responsible for data exchange and communication between users. It is used in the route planner for managing the user profiles and points of interest. A reason, why Microsoft Sharepoint was decided, is the idea of a communication platform for users in future

[2] www.microsoft.com/silverlight/

work. Such a platform within the route planner enables user to discuss, share experiences, and collect information.

11.2.6 Microsoft SQL Server 2008

For managing the geographical data as well as the user specific data and the user profiles, Microsoft SQL Server was used. The abbreviation SQL stands for structured query language. Microsoft SQL Server is a relational database management system provided by Microsoft. Microsoft SQL Server Database Engine is the database module whereas SQL Server Management Studio is used to administrate database objects. Microsoft SQL Server uses the SQL variant T-SQL (Transact-SQL) as query language, [39]. T-SQL mainly provides syntax for the use of stored procedures and transactions. SQL Server 2008 also supports structured and semi-structured data, including digital media formats for pictures, audio, video, and other multimedia data. Moreover, it provides a geography data type. The geography data type (resp. spatial data type) supports the storage of geographical objects with their coordinates. Methods defined in OGC-Standard (Open Geospatial Consortium) enable calculating with geographic objects which can be represented as points, lines or polygons. Some examples for such predefined methods are *Distance*, *Buffer* or *Length*.

11.3 Graphical user interface

11.3.1 Menu structure

Due to usability (see Chapter 8.2.1), the menu structure must be self speaking and not overloaded. Therefore, the number of menu points was minimized and sub menus have been avoided. The menu was designed in cooperation with an organization for people with disabilities in Graz, Austria ('Behindertenbeauftragte der Stadt Graz'). Potential users asked for an easy start page of the route planner, visualized in Figure 11.3. At this page a predefined user profile can be chosen or the login page can be accessed. After that, the user reaches the main page of the route planner (Figure 11.4).

Main menu

For blind people it is hard not to lose the orientation at the web page. Various links and sub menus are confusing for them. Therefore, the main menu shall be easy and consists of the following six pages:

- Home (main page) with route planning (see Figure 11.4)
- Pre-trip training
- Points of interest

Figure 11.3: Start page *(published in agreement with the project consortium of Route4you, see Chapter 1.2)*

  - visualization of POI (see Chapter 9.7)
  - creation of POI (see Chapter 9.7)
  - administration of POI (see Chapter 9.7)

- User profile
- Links
- Help

11.4 User profiles

At the menu point 'user profile' of the main menu, users are able to create their own profile to choose specific settings due to their individual preferences which subsequently are considered in route planning.

11.4.1 User settings

The user settings of every profile comprise optimization criteria and restrictions, obstacle categories, and POI categories which influence route planning and route description. The user profile with its individual setting possibilities is shown in Figure 11.5.

Basic profile

At first, a basic profile with predefined settings can be chosen at the user profile. Three profiles can be distinguished: wheelchair ('Rollstuhl'), blind ('Blind'), and visually impaired ('Sehbehindert'). The predefined settings are suggestions to the user groups derived from answers of the questionnaire.

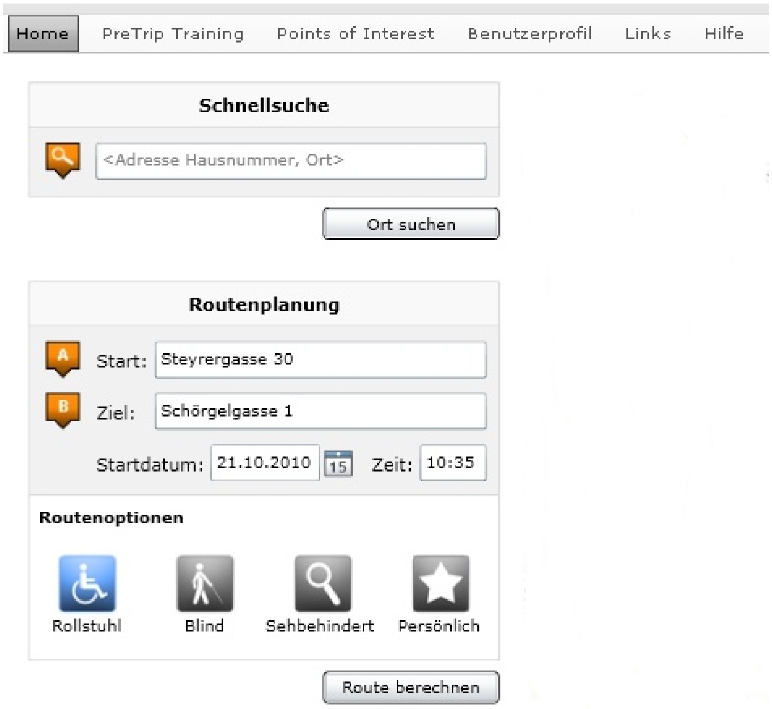


Figure 11.4: Main page *(published in agreement with the project consortium of Route4you, see Chapter 1.2)*

This basic profiles can be modified by the users and the resulting personal settings can be saved ('Einstellungen speichern'). Moreover, the personal profile settings can be reset to the basic profile settings ('Standardwerte laden').

Restrictions on optimization criteria

At the next part of the user profile, the definition of restrictions on optimization criteria ('Optimierungseinstellungen für die Routenplanung') can be done by the users themselves. Users can choose the following restrictions:

- max. walking distance to a station
- max. slope/incline
- public transport: yes/no
  - max. number of transfers
  - bus only
  - tram only

The implementation of these restrictions in multi-criteria route optimization is described in Chapter 10.4.2. In case that a route cannot be calculated with the chosen settings, users get a warning that no route meets their requirements. As a consequence, users have to relax their restrictions to get a result.

Obstacles

Below the settings of optimization criteria, obstacles ('Hindernisse') can be enabled or disabled. If obstacles are selected, obstacle categories appear to choose individual obstacle categories which will be considered in route optimization and route description. The selectable obstacle categories are the same categories as listed in Chapter 9.4.2.

Points of interest

At the next part of the user profile, the selection of categories for points of interest is provided. The selection of categories can be done the same way as the selection of obstacles. The POI categories are listed in Chapter 9.4.2.

11.5 Functionalities

The developed route planner provides different functionalities to the user. These are route optimization, pre-trip training which allows the comparison of different routes, as well as adding and administrating POI. In the following, the functionalities of the application are described in detail.

11.5.1 Route planning

Route planning is available directly at the left side of the main page. For route planning, a start address and a destination address have to be entered. As public transport is included, the input of date and departure time is necessary to compute a time-dependent route. The date can be chosen from an implemented calendar or written as *DD.MM.YYYY (day, month, year)* and time has to be written in the format *hh:mm* in the predefined text box, where *hh* refers to an hour between 00 and 23 and *mm* refers to minutes between 00 and 59. After confirming the entered data, the route planning algorithm starts its calculation with the predefined settings in the user profile. The method of route planning using multi-criteria optimization is realized as described in Chapter 10.3.

Geocoding

Geocoding is the process of determining a position from an address, [34]. The user input for planning a route is a start address, a destination address

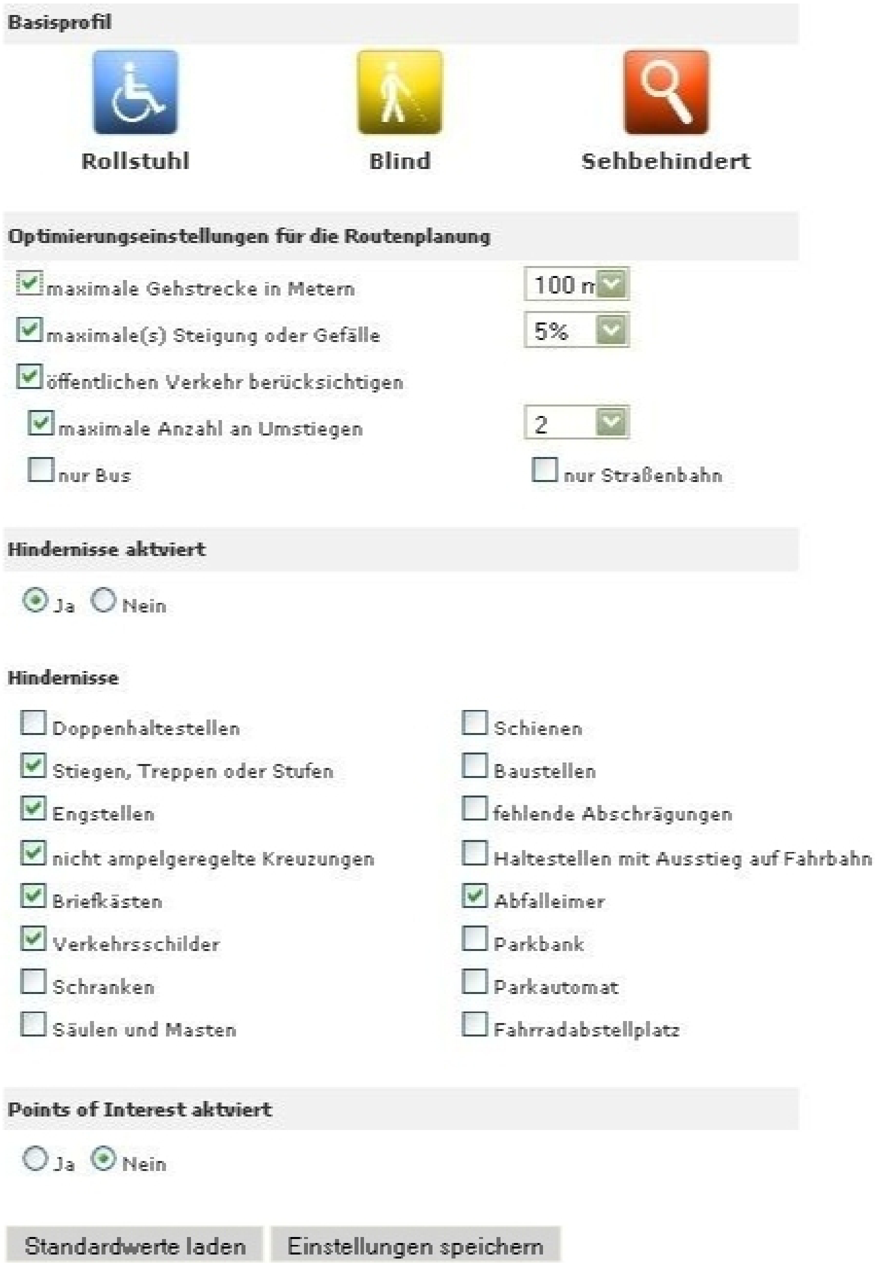


Figure 11.5: Profile settings *(published in agreement with the project consortium of Route4you, see Chapter 1.2)*

as well as date and time. This address information has to be transformed to coordinates and finally to node IDs in the digraph. For determining the corresponding coordinates and the start- and end-node IDs which are necessary for the route planning algorithm, geocoding has to be accomplished, followed by a time-dependent nearest-node search. This search computes the nearest node in the digraph closest to the position determined from geocoding. The stages of determining the node IDs for the input of the route planning algorithm are shown in Figure 11.6.

Figure 11.6: Stages from the user input of the address to the start of the route planning algorithm

Route results

Depending on the chosen profile and settings, differences between routes for wheelchair users, blind people, and individual profiles can be achieved. Figure 11.7 and Figure 11.8 show two different routes. In Figure 11.7 a route for wheelchair users is represented, which does not contain any stairs or missing ramps and public transport vehicles without accessibility, for instance. In contrast, Figure 11.8 shows a route for blind people with the same start and destination. This route contains a public transport ride. The public transport vehicle does not necessarily have to be a low-floor vehicle in case of blind people, but the route for blind people does exclude crossings without traffic lights.

11.5.2 Pre-trip training

The pre-trip training can be seen as a route decision support system. It provides different routes dependent on specific user settings by applying a multi-criteria optimization in the route planning algorithm. This allows the user the comparison of various routes (e.g., fastest route versus route with minimum number of transfers). The multi-criteria optimization methods and details of the implementation are previously explained in Chapter 10.3. The pre-trip training is shown in Figure 11.9. In addition, the download of a sound file containing the route description is provided. This enables the user to take the route description along on the journey. Moreover, blind users can listen to the sound file to memorize the route and get an idea of the surrounding.

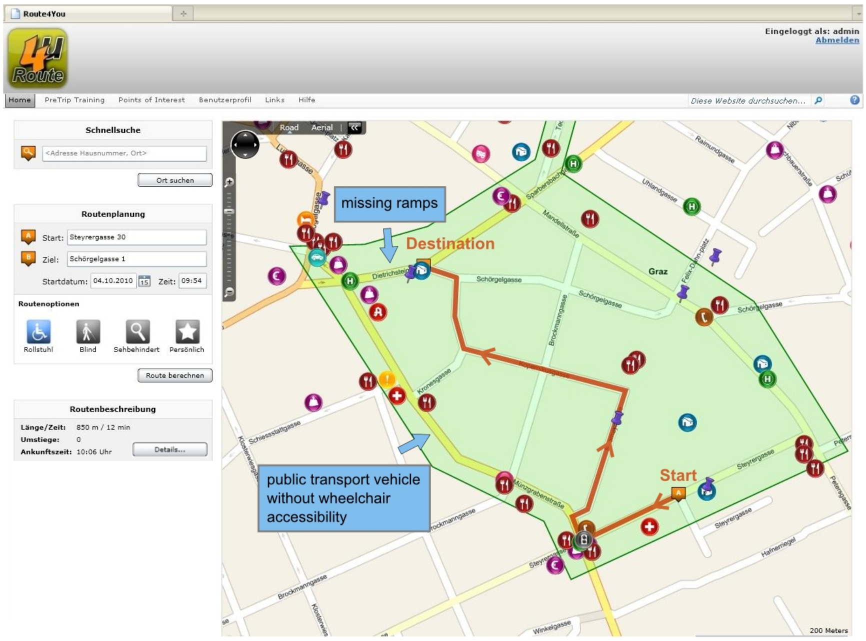


Figure 11.7: Route for wheelchair users *(published in agreement with the project consortium of Route4you, see Chapter 1.2)*

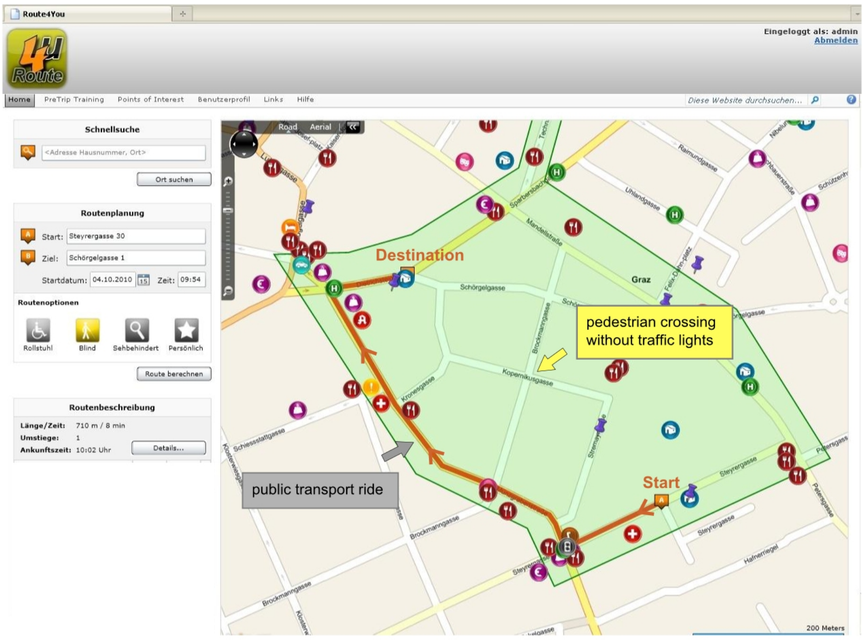


Figure 11.8: Route for blind people *(published in agreement with the project consortium of Route4you, see Chapter 1.2)*

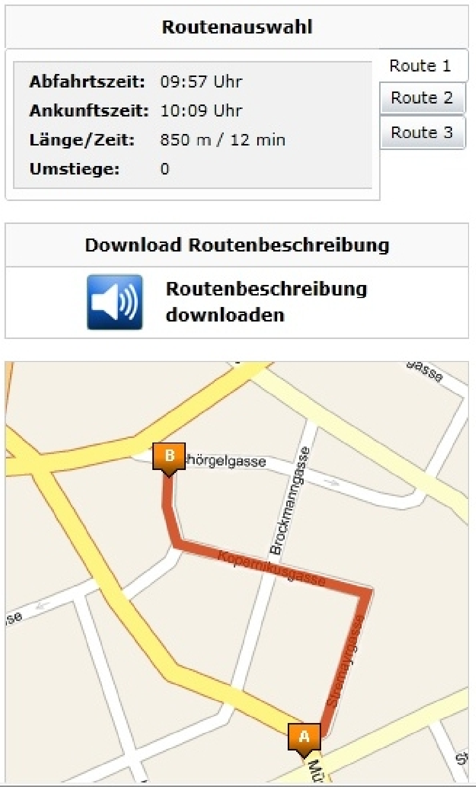


Figure 11.9: Pre-trip Training *(published in agreement with the project consortium of Route4you, see Chapter 1.2)*

11.6 Route description

The route description given to the user contains various information for orientation like turn information, street names, obstacle warning and additional information. The content of the route description depends on the current network. E.g., in the public transport network other information is provided than in the path network. This leads to different maneuver instructions for each network. The route description is composed of turn information and net information. First, an overview about the whole route is given and the total length, time, and number of transfers are provided to the user.

11.6.1 Route information based on different networks

Four types of networks and their corresponding information are distinguished in the route description:

- OSM: Street names, additional information about objects in the closer surrounding, turn information

- DET: Street names, obstacles, inclination, additional information about objects on the path or the closer surrounding, turn information
- PTN: Station names, public transport lines, accessibility of the public transport vehicles, time information
- CON: Transfer information

11.6.2 Turn information

Turn information affect the OSM network and the detailed path network. For the calculation of direction information (turn left, turn right, etc.) plane coordinates are necessary. Turn information is determined by calculating a relative course between two edges.

Course determination

Based on a geodetic coordinate system, the course is defined as the angle between an edge and the x-axis clockwise (Figure 11.10 left). The calculation of the course in degrees states as follows, provided that *arctan* delivers the angle in radians:

$$course = \arctan(\Delta y/\Delta x)/\Pi \cdot 180$$

Due to the division of $\Delta y/\Delta x$ the sign of both components gets lost and therefore the arctan does not distinguish between diametrically opposed directions. Let us consider an example: The result of arctan(10/10) is 45 degrees clockwise from the x-axis. However, the result of $\arctan(-10/-10)$ from the diametrically opposed direction is again 45 degrees even tough the correct angle should be 225 degrees in the third quadrant. Therefore, the correct quadrant has to be considered manually. For a manual correction, the absolute values $|\Delta y|$ and $|\Delta x|$ are used, to get an angle in the first quadrant even if the algebraic signs of Δy and Δx are different. The numbering of quadrants is shown in Figure 11.10 on the right. Note that without any correction $\arctan(|\Delta y| \,/\, |\Delta x|)$ always provides an angle in the first quadrant. The corrected angle, where the quadrant is considered, can be achieved by applying the definitions in Table 11.1. To avoid a division by zero, $|\Delta x| = 0$ has to be regarded separately. In case of $|\Delta x| = 0$ and $\Delta y > 0$ the corrected angle is 90 degrees and in case of $|\Delta x| = 0$ and $\Delta y < 0$ the corrected angle is 270 degrees. For $|\Delta y| = 0$ the course is either 0 degrees or 180 degrees dependent on the sign of Δx.

The relative course between two edges can be determined as follows:

$$course_{edge1} = \arctan(|\Delta y_1| \,/\, |\Delta x_1|)/\Pi \cdot 180$$

$$course_{edge2} = \arctan(|\Delta y_2| \,/\, |\Delta x_2|)/\Pi \cdot 180$$

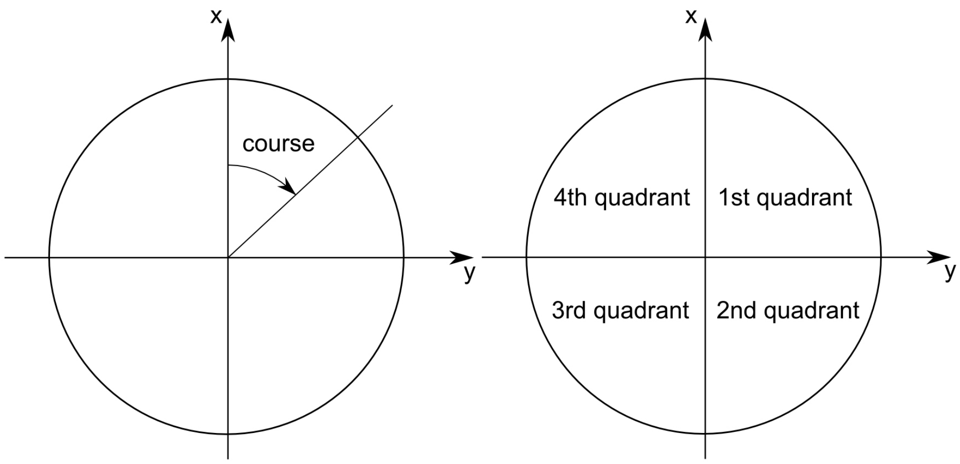


Figure 11.10: Left: The course calculated from arctan($|\Delta y| \,/\, |\Delta x|$) without any correction is an angle in the first quadrant clockwise from the x-axis. Right: The numbering of quadrants is shown.

Table 11.1: Quadrant corrections of arctan($|\Delta y| \,/\, |\Delta x|$)

Δx	Δy	quadrant	corrected angle ($course_{corr}$)
+	+	1st quadrant	course
-	+	2nd quadrant	180 - course
-	-	3rd quadrant	180 + course
+	-	4th quadrant	360 - course

$$course_{relative} = course_{edge2\_corr} - course_{edge1\_corr}$$

The turn information follows from the sign and value of the $course_{relative}$. A negative value of $course_{relative}$ means a turn on the left side, a positive value defines turns on the right as illustrated in Figure 11.11. Figure 11.12 shows the detailed distribution of turn informations in the route description depending on $course_{relative}$.

11.6.3 Route description in different networks

In OSM network and detailed path network turn information is the main information for orientation. Additionally, turn information is supplemented by object information which is very important for blind people (e.g., turn left *at the traffic light*). In the public transport network turn information is not necessary. However, in this network information about stations and the public transport vehicle (e.g., station names, vehicle numbers, transfer information, etc.) is relevant and must be provided to the user. In addition to maneuver instructions, the length and duration of each maneuver are

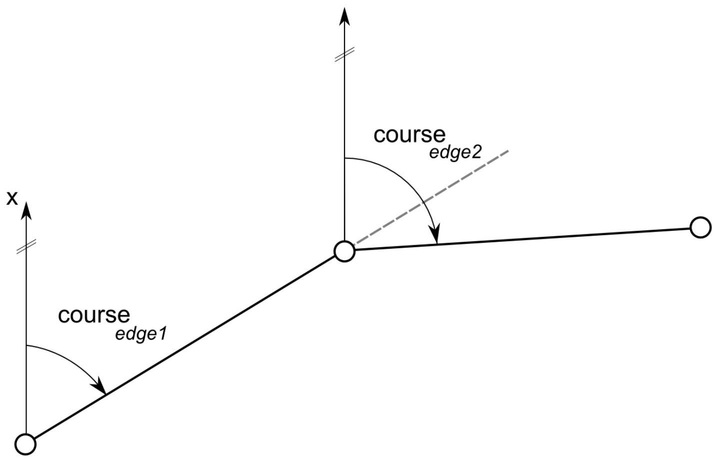


Figure 11.11: A positive value of $course_{relative}$ defines a turn on the right

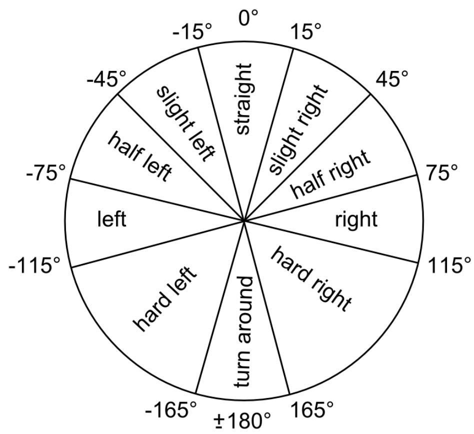


Figure 11.12: Turn information, modified from [76]

provided. Moreover, in OSM network and DET network, obstacle warnings are given to the user (e.g., 'attention, in 10m there is a *traffic sign on the sidewalk*'). For blind people, points of interest of the closer surrounding are stated (e.g., 'there is a *restaurant* on the left side'). The categories of points of interest that should be named can be set in the user profile. An example of a detailed route description is given in Table 11.2.

Based on this general example, the detailed route descriptions for the route examples in Figure 11.7 and Figure 11.8 are given in Table 11.3 and Table 11.4. Note that traffic signs are only important for blind people. The output of obstacles and points of interest depends on the settings in the user profile.

Table 11.2: Example of a detailed route description

route description	length	duration	direction
Go straight along Columbus street. Inclination is 4%. Asphalt.	20m	1min	north
Cross the park road.	5m	0min	-
Go straight along Columbus street. Asphalt.	30m	1min	north
Post box on the left.	3m	0min	-
At the traffic light turn left in Kepler street. Go straight along Kepler street to station 'University'. Inclination is 2%. Asphalt.	53m	2min	west
Chinese restaurant 'Molo' on the left.	5m	0min	-
Attention, there is a traffic sign on the sidewalk.	20m	1min	-
At station 'University' take the streetcar 6 in direction of central station. Departure time: 10:15. Travel 4 stations.	1,2km	8min	southwest
Exit streetcar 6 at station 'cinema'.	-	-	-
Change to streetcar 1 in direction of main square. Departure time: 10:30.	0m	6min	-
Travel 2 stations with streetcar 1.	800m	4min	west
Exit streetcar 1 at station 'opera'.	-	-	-
Turn right and go along Mozart street. Asphalt.	21m	1min	east
Attention, there is a fire hydrant on the sidewalk.	5m	0min	-
Cash point on the left.	5m	0min	-
Turn left to café 'Strauss'. 5cm step at the entrance. Destination reached.	-	-	-

11.7 Validation

The output of the route planner was tested with users. It turned out that the multi-criteria optimization with individual settings provides correct routes with very good route suggestions in pre-trip training. Maneuver instructions are easy to understand and provide enough details for a good orientation, which is especially relevant for blind people. Moreover, blind people have been excited about obstacle warnings. Although multi-criteria optimization is a powerful tool to provide exactly those route suggestions a user prefers, it has its disadvantages. Multi-criteria optimization is a very time consuming process. In some situations, the runtime of the algorithm has exceeded the expectancy value for route calculation given in Chapter 8.2.4. However, the response time of the route calculation at the user client does not only

Table 11.3: Detailed route description of the route for wheelchair users given in Figure 11.7

route description	length	duration	direction
Go straight along Steyrergasse. Slope is 5%. Asphalt.	150m	2min	south-west
Phone box on the right.	145m	2 min	-
At the traffic light turn right in Münzgrabenstraße. Asphalt.	50m	1min	north-west
Turn right in Stremayrgasse. Asphalt.	5m	0min	east
Turn slight left. Go straight along Stremayrgasse. Inclination is 1%. Asphalt.	135m	3min	north-east
Attention, there is a traffic sign on the sidewalk.	124m	-	-
Turn left in Kopernikusgasse. Asphalt.	208m	4min	north-east
Turn half right. Asphalt.	42m	1min	north-east
Turn slight right. Asphalt.	38m	1min	north
Turn left in Schörgelgasse. Asphalt.	27m	0min	north-east
Destination reached.	-	-	-

depend on the algorithm itself. It also depends on server hardware and the bandwidth of the Internet connection. Nevertheless, speed up techniques for multi-criteria optimization would be a topic for future work (see Chapter 12). The assumptions of all other acceptable duration values in Chapter 8.2.4, Table 8.1 have been achieved.

Table 11.4: Route description of the route for blind people given in Figure 11.8

route description	length	duration	direction
Go straight along Steyrergasse. Slope is 5%. Asphalt.	150m	2min	south-west
Attention, there is a traffic sign on the sidewalk.	16m	0min	-
Phone box on the right.	145m	2 min	-
Attention, there is a traffic sign on the sidewalk.	149m	0min	-
At the traffic light turn right in Münzgrabenstraße. Asphalt.	50m	2min	north-west
Cross the Stremayrgasse.	30m	0min	north-west
Turn half left. Go straight along Münzgrabenstraße to station 'Neue Technik'. Asphalt.	100m	2min	north-west
At station 'Neue Technik' take the streetcar 6 in direction of central station. Departure time: 10:00. Travel 1 station.	400m	2min	north-west
Exit streetcar 6 at station 'Dietrichsteinplatz'.	-	-	-
Turn left and go along Münzgrabenstraße.	5m	0min	north-west
Super market on the right.	2m	0min	-
Turn right in Schörgelgasse.	33m	0min	east
Destination reached.	-	-	-

Chapter 12

Conclusion and Outlook

12.1 Conclusion

This work describes the development of a route planning tool for people with disabilities, providing user specific route information. As one difference to standard route planners, users can customize route settings and additional information in their individual user profiles. Moreover, the needs of blind people, visually impaired people, and wheelchair users are considered in one combined route planner. The combination of different data sources and the implementation of restrictions, which depend on different optimization criteria, lead to a multi-criteria time-dependent search problem. The theoretical part of the work concentrates on the structural explanation of the multi-criteria route planning algorithm and the implementation of the public transport network in route optimization. A detailed description of multi-criteria optimization in route planning has been elaborated in this thesis. Different solution methods are presented and compared to each other. The practical part focuses on the conception of the route planner especially for people with disabilities. The special needs of the user groups and the system requirements are elaborated in cooperation with potential users. A geographical database consisting of different data sources was established as a basis for route planning. User profiles allow individual settings due to route optimization, obstacles, and points of interests. Moreover, the implementation of multi-criteria route optimization with restrictions is discussed. Finally, the route planner with its functionality is presented.

12.2 Research results and future work

12.2.1 Geographical data

For people with disabilities a geographical database with detailed information is important. Requirements on data content (e.g., obstacles, points of interest, etc.) for people with disabilities (blind people, visually impaired people, and wheelchair users) have been investigated in close cooperation with the user groups. It has been demonstrated that people with disabilities need much more information than it is considered in common route planners. A path network containing such detailed information like footpaths, sidewalks, obstacles, and lots of additional information is necessary to provide an adequate route description and has been established in this work. A detailed path network especially for blind people, visually impaired people, and wheelchair users has been modified from previous work in [54]. The existing network has been expanded for wheelchair users. Additionally, the topology of the detailed path network was adapted for multi-criteria optimization. With this detailed path network a perfect pedestrian network for people with disabilities has been demonstrated. The detailed path network was combined with OSM data and a public transport network to provide a larger coverage of the route planner. It has been shown that, on the one hand, OSM data is suited for the route planner. On the other hand, OSM data has disadvantages in degree of detail and there is a lack of information for people with disabilities. Due to public transport, the modeling of timetables and the implementation of public transport in route planning has been shown.

Ideas on future work

The huge step from a local prototype to a globally used route planner is influenced by some aspects. One difficulty is the acquisition of very detailed path networks and additional information especially for people with disabilities. Moreover, due to the huge amount of data, it is hard to warrant quality aspects like actuality, completeness, and reliability and expand the area to a comprehensive footpath network. In some way, this is solved by the user-driven data acquisition. However, this affects the maintenance of the system. The user inputs, due to the user generated data, have to be managed and controlled by the system owner. Another possibility for an efficient data acquisition would be a cooperation with the OSM community. In that case, an OSM data model for people with disabilities could be defined. However, such a model should be considered for worldwide use. Moreover, sighted OSM mapper should be sensitized to the needs of blind people so that more information for this user group can be included and provided. Another expansion would be the inclusion of real-time information (e.g., construction sites).

12.2.2 Route planning

Different optimization criteria can be considered in one optimization process by applying a multi-criteria optimization. It was demonstrated that for solving a multi-criteria optimization problem Dijkstra's algorithm can be used. In the theoretical part, different solution methods for multi-criteria optimization are described in detail. It has been demonstrated that in route planning only strict Pareto optimal solutions are reasonable. The combination of different criteria especially for people with disabilities is realized by the implementation of a multi-criteria optimization including restrictions with additive values. This allows the minimization of the number of transfers according to time. Different routes provided by the multi-criteria optimization algorithm can be compared to each other in pre-trip training. Individual restrictions can be defined in the user profile. It has been found out that multi-criteria route optimization with individual restrictions does perfectly fit for people with disabilities. Different user groups with contrasting demands can use the same route planner. However, multi-criteria optimization is computationally intensive. It has been mentioned that a common hierarchical concept with bidirectional search for a faster runtime of the algorithm cannot be applied on multi-criteria optimization without losing Pareto optimal solutions.

Ideas on future work

As already mentioned, multi-criteria optimization has more complexity than single-criterion problems, since the solution is ambiguous. Therefore, future work can be done in increasing the performance of the algorithm. Multi-criteria hierarchical route optimization would be a challenging approach.

Another idea for future work in route planning is the expansion of optimization criteria. A further criterion with respect to people with disabilities could be the transfer place. For people with disabilities it is important, if they have to take a walk or to cross a street to reach the next station in case of a transfer, or if they are able to remain at the same station and wait for the next vehicle. Waiting at the station is much more appreciated than walking to another station or crossing a street. E.g., in the city of Graz, Austria the streetcars 1, 6, and 7 pass various same stations before reaching the big transfer place 'Jakominiplatz'. Common route planners do not suggest a transfer at a convenient one of the shared stations. Instead, the route planners suggest the transfer at the very crowded and - for blind people - dangerous 'Jakominiplatz' with lots of street crossings.

12.2.3 Requirements on the application

In this thesis, special demands and requirements of people with disabilities on a route planner are elaborated. Results of a questionnaire demonstrate the behavior of people with disabilities in unknown urban environments and their needs for orientation. It was demonstrated that route planning for blind and visually impaired people as well as wheelchair users can be provided in a shared route planner by the introduction of a user profile. Due to different requirements, a user profile for adjusting individual settings was elaborated and implemented in the route planning system. Moreover, web accessibility and the compliance of the Web Content Accessibility Guidelines (WCAG 2.0) is a crucial fact which was considered in the route planner. Therefore, the route planner is easily operated by people with disabilities. A new investigation is the provision of a soundfile containing the route description.

Ideas on future work

Concerning the user interface, continuative studies and usage tests with people, having different kinds of disabilities, could be elaborated to provide additional personal settings on the visualization of the user interface.

12.3 Outlook

During the work, ideas for extending and improving the system arose. This includes the implementation of real-time data like the position of construction sites or daily time tables with delays. To realize this, a close cooperation with the responsible institutions is necessary, what was already established for the public transport data. Moreover, the system could be extended for other user groups (deaf people, parents with buggies). This affects the generation of new profiles as well as the acquisition of additional user-specific data. Another enhancement could be a web page specialized for smart phones, so that people can also search for their way outdoors. All in all, the route planner has a lot of potential for further extensions.

Bibliography

[1] K. N. Androutsopoulos and K. G. Zografos. Solving the multi-criteria time-dependent routing and scheduling problem in a multi-modal fixed scheduled network. *European Journal of Operational Research*, 192(1):18–28, 2009. [53]

[2] R. Balakrishnan and K. Ranganathan. *A textbook of graph theory*. Springer, New York, 1999. [14]

[3] N. Bartelme. *Geoinformatik - Modelle, Strukturen, Funktionen*. Springer, Berlin Heidelberg, 4th edition, 2005. [80]

[4] D. P. Bertsekas. *Network Optimization: Continuous and Discrete Models*. Athena Scientific, Belmont, Massachusetts, 1998. [21, 24, 26]

[5] L. Boodlal. *Accessible Sidewalks and Street Crossings – an informational guide*. U.S. Department of Transportation. [60]

[6] C. Brando and B. Bucher. Quality in user-generated spatial content: A matter of specifications. *13th AGILE International Conference on Geographic Information Science*, 2010. [84]

[7] A. Brimicombe. *GIS, Environmental Modeling and Engineering*. CRC Press, USA, 2nd edition, 2010. [xiv, 79, 80]

[8] G. S. Brodal and R. Jacob. Time-dependent Networks as Models to Achieve Fast Exact Time-table Queries. *Electronic Notes in Theoretical Computer Science*, 92:3 – 15, 2004. Proceedings of ATMOS Workshop 2003. [52]

[9] C. Bühler, H. Heck, and J. Becker. How to Inform People with Reduced Mobility about Public Transport. In *ICCHP '08: Proceedings of the 11th International Conference on Computers Helping People with Special Needs*, pages 973–980, Berlin, Heidelberg, 2008. Springer-Verlag. [1]

[10] C. Bühler, H. Heck, R. Wallbruch, J. Becker, and C. Bohner-Degrell. User Feed-Back in the Development of an Information System for Public Transport. In *ICCHP '10: Proceedings of the 12th International*

Conference on Computers Helping People with Special Needs, Part I, pages 273–279, Berlin, Heidelberg, 2010. Springer-Verlag. [5]

[11] P. A. Burrough and R. A. McDonnel. *Principles of Geographical Information Systems*. Oxford, Oxford New York, 1998. [80]

[12] Y. L. Chen and K. Tang. Minimum time paths in a network with mixed time constraints. *Computers and Operations Research*, 25(10):793–805, 1998. [3, 53]

[13] D. Costelloe, P. Mooney, and A. Winstanley. Multi-objective optimisation on transportation networks. *Proceedings of the 4th AGILE Conference on GIS*, 2001. [99]

[14] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959. [22]

[15] W. Domschke. *Logistik: Transport*. Oldenbourg Verlag, München, 1981. [22]

[16] D. Edwards and J. Simpson. Integration and access of multi-source vector data. *Proceedings of the Joint International Symposium on Geospatial Theory, Processing and Applications, Ottawa, Canada*, 2002. [89]

[17] M. Ehrgott. *Multicriteria Optimization*. Springer, 2005. [xiii, 30, 31, 32, 33, 34, 35, 36, 37, 38]

[18] P. L. Emiliani and L. Burzagli. W3C-WAI Content Accessibility Auditing. *Universal Access in Health Telematics*, pages 175–196, 2005. [3, 76]

[19] X. Gandibleux, F. Beugnies, and S. Randriamasy. Martins' algorithm revisited for multi-objective shortest path problems with a maxmin cost function. *4OR: Quarterly Journal of the Belgian, French and Italian Operations Research Societies*, 4(1):47–59, 2006. [37]

[20] M. Gietz. *Computergestützte Tourenplanung mit zeitkritischen Restriktionen*. Produktion und Logistik. Physica-Verlag, Heidelberg, 1994. [11]

[21] R. Golledge. *Defining the Criteria Used in Path Selection*. Working Paper, UCTC No. 278, The University of California Transportation Center, 1995. [65]

[22] R. Golledge. *Wayfinding behavior: cognitive mapping and other spatial processes*. The Johns Hopkins University Press, Baltimore, Maryland, 1999. [58]

[23] R. Golledge, R. Klatzky, J. Loomis, J. Spiegle, and J. Tietz. A geographical information system for a GPS based personal guidance system. *International Journal of Geographical Information Science*, 12(7):727–749, 1998. [5, 57, 62, 65, 66]

[24] R. Golledge and R. Stimson. *Spatial behavior: a geographic perspective.* Guilford Press, New York, 1997. [58]

[25] M. Goodchild. Citizens as sensors: The world of volunteered geography. *GeoJournal 2007*, 69:211–221, 2007. [83]

[26] P. Gritzmann and R. Brandenberg. *Das Geheimnis des kürzesten Weges - Ein mathematisches Abendteuer.* Springer, Berlin, Heidelberg, 3rd edition, 2005. [11, 25]

[27] T. Gunkel, M. Müller-Hannemann, and M. Schnee. Improved Search for Night Train Connections. In *ATMOS*, 2007. [6, 51]

[28] C. Hallam, K. J. Harrison, and J. A. Ward. A multiobjective optimal path algorithm. *Digital Signal Processing*, 11(2):133 – 143, 2001. [47]

[29] P. Hansen. *Multiple Criteria Decision Making Theory and Application*, volume 177 of *Lecture Notes in Economics and Mathematical Systems*, section Bicriteria path problems, pages 109–127. Springer, Berlin, 1979. [36]

[30] S. Helal, S. Moore, and B. Ramachandran. Drishti: An Integrated Navigation System for Visually Impaired and Disabled. *Proceedings of the 5th International Symposium on Wearable Computer*, pages 149–156, 2001. [5, 65, 66]

[31] B. Hofmann-Wellenhof, K. Legat, and M. Wieser. *Navigation – principles of positioning and guidance.* Springer, Wien, 2003. [11, 12, 13, 15, 18, 22, 23, 25, 26, 27, 97, 98]

[32] R. Huang. A Schedule-based Pathfinding Algorithm for Transit Networks Using Pattern First Search. *Geoinformatica*, 11(2):269–285, 2007. [91]

[33] A. Hub, S. Kombrink, K. Bosse, and T. Ertl. TANIA - A Tactile-Acoustical Navigation and Information Assistant for the 2007 CSUN Conference. *Proceedings of the California State University, Northridge Center on Disabilities, 22nd Annual International Technology and Persons with Disabilities Conference (CSUN 2007)*, 2007. [5]

[34] A. Jagoe. *Mobile Location Services - The Definitive Guide.* Prentice Hall, New Jersey, 2003. [xiv, 81, 117]

[35] D. Jungnickel. *Graphs, Networks and Algorithms.* Springer, Berlin, Heidelberg, 1999. [14, 21, 26, 27]

[36] C. Koch-Schmuckerschlag and O. Kalamidas. *Barrierefreies Bauen für ALLE Menschen.* Stadtbaudirektion Graz, 2006. [58, 59, 60]

[37] B. Korte and J. Vygen. *Combinatorial Optimization: Theory and algorithms.* Springer, Berlin, Heidelberg, 3rd edition, 2006. [22, 26, 27]

[38] F. A. Kuipers and P. V. Mieghem. Non-Dominance in QoS Routing: An Implementational Perspectivem. *Digital Signal Processing*, 9(3):267–269, 2005. [31]

[39] M. Lisin, J. Joseph, and A. Goyal. *Microsoft SQL Server 2008 Reporting Services Unleashed.* SAMS, USA, 1st edition, 2009. [114]

[40] J. M. Loomis, R. G. Golledge, and R. L. Klatzky. Navigation System for the Blind: Auditory Display Modes and Guidance. *Presence: Teleoperators and Virtual Environments*, 7(2):193–203, 1998. [5]

[41] E. Q. V. Martins. On a multicriteria shortest path problem. *European Journal of Operational Research*, 16(2):236–245, 1984. [35, 36, 37]

[42] B. Mayerhofer, B. Pressl, and M. Wieser. *Computers Helping People with Special Needs*, volume 5105 of *Lecture Notes in Computer Science*, section ODILIA - A Mobility Concept for the Visually Impaired, pages 1109–1116. Springer - Verlag, Berlin, Heidelberg, 2008. [5]

[43] R. H. Möhring. *Verteilte Verbindungssuche im öffentlichen Personenverkehr: Graphentheoretische Modelle und Algorithmen, No.624/99.* Technische Universität Berlin, 1999. [33, 34]

[44] A. Müller, P. Neis, M. Auer, and A. Zipf. Ein Routenplaner für Rollstuhlfahrer auf der Basis von OpenStreetMap-Daten. Konzeption, Realisierung und Perspektiven. *Angewandte Geoinformatik 2010 - Beiträge zum 22. AGIT-Symposium Salzburg*, pages 258–261, 2010. [84]

[45] M. Müller-Hannemann, F. Schulz, D. Wagner, and C. Zaroliagis. *Railway Optimization 2004*, volume 4359 of *Lecture Notes in Computer Science*, section Timetable Information: Models and Algorithms, pages 67–90. Springer - Verlag, Berlin, Heidelberg, 2007. [xiv, 51, 52, 53]

[46] M. Müller-Hannemann and K. Weihe. Pareto Shortest Paths is Often Feasible in Practice. In *WAE '01: Proceedings of the 5th International Workshop on Algorithm Engineering*, pages 185–198, London, UK, 2001. Springer-Verlag. [31]

[47] MoBIC-Consortium. Mobility of Blind and Elderly People Interacting with Computers. *Final Report*, 1997. [5, 62, 63, 66]

[48] K. Nachtigal. Time depending shortest-path problems with applications to railway networks. *European Journal of Operations Research*, pages 154–166, 1995. [52]

[49] A. Nathan. *WPF 4 Unleashed*. SAMS, USA, 1st edition, 2010. [113]

[50] P. Neis, D. Zielstra, and A. Zipf. The Street Network Evolution of Crowdsourced Maps: OpenStreetMap in Germany 2007-2011. *Future Internet 2012*, 4(1):1–21, 2012. [82, 83]

[51] A. Orda and R. Rom. Shortest-path and minimum-delay algorithms in networks with time-dependent edge-length. *Journal of ACM*, 37(3), 1990. [52]

[52] H. Petrie and V. Johnson. MoBIC: An Aid to Increase the Mobility of Blind and Elderly Travellers. *Proceedings of the Third European Conference for the Advancement of Rehabilitation Technology (ECART3)*, pages 247–249, 1995. [5]

[53] H. Petrie, V. Johnson, T. Strothotte, A. Raab, S. Fritz, and R.Michael. MoBIC: Designing a Travel Aid for Blind and Elderly People. *Journal of Navigation*, 49(1):45–52, 1996. [5]

[54] B. Pressl. Digitale Karte zur Zielführung in einem Navigationssystem für blinde Personen. Master's thesis, Graz University of Technology, 2005. [85, 86, 130]

[55] B. Pressl. *Navigationssystem für blinde Personen*. VDM Verlag Dr. Müller, Saarbrücken, 2008. [61]

[56] B. Pressl, C. Mader, and M. Wieser. User-Specific Web-Based Route Planning. In *ICCHP '10: Proceedings of the 12th International Conference on Computers Helping People with Special Needs, Part I*, pages 280–287, Berlin, Heidelberg, 2010. Springer-Verlag. [3]

[57] B. Pressl and M. Wieser. Datenmodellierung und Routenplanung für behinderte Personen auf der Basis von Fahrplänen. *Angewandte Geoinformatik 2010 - Beiträge zum 22. AGIT-Symposium Salzburg*, pages 262–270, 2010. [62]

[58] E. Pyrga, F. Schulz, D. Wagner, and C. D. Zaroliagis. Experimental Comparison of Shortest Path Approaches for Timetable Information. In *ALENEX/ANALC*, pages 88–99. SIAM, 2004. [52]

[59] E. Pyrga, F. Schulz, D. Wagner, and C. D. Zaroliagis. Efficient Models for Timetable Information in Public Transportation Systems. *ACM Journal of Experimental Algorithmics*, 12, 2008. [52]

[60] F. Ramm and J. Topf. *OpenStreetMap*. Lehmanns Media, Berlin, 1st edition, 2008. [82]

[61] L. Ran, S. Helal, and S. Moore. Drishti: An Integrated Indoor/Outdoor Blind Navigation System and Service. *Pervasive Computing and Communications, Proceedings of the Second IEEE Annual Conference*, pages 23–30, 2004. [5, 66]

[62] U.-J. Rüetschi. *Wayfinding in Scene Space Modelling Transfers in Public Transport*. PhD thesis, University of Zürich, 2007. [xiv, 103]

[63] S. Schmitz, A. Zipf, and P. Neis. New applications based on collaborative geodata - the case of routing. *In XXVIII INCA International Congress on Collaborative Mapping and SpaceTechnology*, 2008. [83, 84]

[64] M. Schnee. *Fully Realistic Multi-Criteria Timetable Information Systems*. PhD thesis, Technische Universität Darmstadt, 2009. [6, 25, 26, 27, 35, 36, 43, 51, 99]

[65] A. Schrijver. *Combinatorial Optimization - Polyhedra and Efficiency*. Volume A, Springer, Berlin, Heidelberg, 2003. [11]

[66] D. Schultes. *Route planning in road networks*. PhD thesis, University of Karlsruhe, 2008. [98]

[67] F. Schulz. *Timetable Information and Shortest Paths*. PhD thesis, University of Karlsruhe, 2005. [24, 33, 51, 52, 100]

[68] F. Schulz, D. Wagner, and K. Weihe. Dijkstra's Algorithm On-Line: An Empirical Case Study from Public Railroad Transport. *Journal of Experimental Algorithmics*, 5:110–123, 2000. [100]

[69] R. E. Steuer. *Multiple-criteria optimization*. John Wiley and Sons, Inc., Canada, 1986. [31]

[70] Z. Tarapata. Selected Multicriteria Shortest Path Problems: An Analysis of Complexity, Models and Adaptation of Standard Algorithms. *Int. J. Appl. Math. Comput. Sci.*, 17(2):269–287, 2007. [29, 31, 34, 52]

[71] V. T'kindt and J.-C. Billaut. *Multicriteria Scheduling*. Springer, Berlin, Heidelberg, New York, 2002. [xiii, 32]

[72] T. Völkel and G. Weber. A New Approach for Pedestrian Navigation for Mobility Impaired Users Based on Multimodal Annotation of Geographical Data. *Universal Access in Human-Computer Interaction. Ambient Interaction*, 4555:575–584, 2007. [5]

[73] T. Völkel and G. Weber. RouteCheckr: personalized multicriteria routing for mobility impaired pedestrians. *Assets 08: Proceedings of the 10th international ACM SIGACCESS conference on Computers and accessibility*, pages 185–192, 2008. [106]

[74] W. Wasserburger and J. Neuschmid. AmauroMap - interaktiver Online-Stadtplan für blinde und sehschwache Menschen. *Angewandte Geoinformatik 2010 - Beiträge zum 22. AGIT-Symposium Salzburg*, pages 1021–1026, 2010. [5]

[75] R. R. Young. *The Requirements Engineering Handbook*. Artech House, Inc., Norwood, MA, 2004. [69, 73]

[76] Y. Zhao. *Vehicle Location and Navigation Systems*. Artech House, Inc., Norwood, MA, 1997. [xv, 124]