Context-Specific Route Directions

Generation of Cognitively Motivated Wayfinding Instructions

Kai-Florian Richter

DISKI 314. IOS Press / AKA.
ISBN 978-1-58603-852-6 (IOS Press); ISBN 978-3-89838-314-1 (AKA)
Volume 3 of the Monograph Series of the Transregional Collaborative Research Center SFB/TR 8. Thomas Barkowsky, Christian Freksa, Christoph Hölscher, Bernd Krieg-Brückner, Bernhard Nebel (series editors)

Dissertation zur Erlangung des Grades eines Doktors der Ingenieurwissenschaften
– Dr.-Ing. –

Vorgelegt im Fachbereich 3 (Mathematik & Informatik)
der Universität Bremen im April 2007.

Gutachter: Prof. Christian Freksa, PhD (Universität Bremen)
Prof. Dr.-Ing. Stephan Winter (University of Melbourne)
Datum des Promotionskolloquiums: 18. Mai 2007.

Zusammenfassung

In unbekannten Umgebungen benötigen Menschen Unterstützung, um erfolgreich ihr Ziel zu erreichen. Sie brauchen Informationen darüber, was sie tun müssen, um von ihrem derzeitigen Aufenthaltsort zum Ziel zu gelangen. Diese Aufgabe wird als Folgen einer Route bezeichnet; Unterstützung für diese Aufgabe wird meist in Form von Routeninstruktionen gegeben. Damit Routeninstruktionen hilfreich sind, sollten sie berücksichtigen, wie Menschen Wegfindungssituationen konzeptualisieren. Routeninstruktionen sollten so beschaffen sein, dass sie einfach zu merken sind und die Entscheidungssituationen, die beim Folgen einer Route auftreten, gut wiedergeben. In meiner Arbeit stelle ich einen Prozess für die Generierung kontext-spezifischer Routeninstruktionen vor, der solche Routeninstruktionen erzeugt.

Eine Analyse struktureller und funktionaler Aspekte von Routen, ebenso wie eine Analyse von Routeninstruktionen unter kognitiven und repräsentationstheoretischen Gesichtspunkten, identifiziert Klassen von Elementen, die in Routeninstruktionen verwendet werden können. Aus kognitiver Perspektive werden insbesondere die Rolle von Landmarken, die Zusammenfassung einer Sequenz von Instruktionen zu einer einzigen Anweisung (sogenanntes Chunking) und die Auswirkungen verschiedener Referenzsysteme diskutiert. Repräsentationstheoretische Aspekte umfassen Fragen der Granularität, verschiedene Arten von Wissen, die bei der Generierung und Interpretation von Referenzen eine Rolle spielen, und die Unterschiede zwischen explizit und implizit repräsentierter Information. Die identifizierten Elemente werden in einer Systematik kategorisiert. Diese basiert auf der Granularität der Elemente. Die Elemente können abhängig von ihrer Funktion verschiedene Rollen beim Wegfinden einnehmen, was ebenfalls in der hier vorgestellten Systematik festgehalten wird.

Mit Hilfe der Systematik wird Guard, ein Prozess für die Generierung von kontext-spezifischen Routeninstruktionen entwickelt. Diese basieren auf einer abstrakten, relationalen Repräsentation der Aktionen, die für das erfolgreiche Folgen einer Route ausgeführt werden müssen. Sie spiegeln konzeptuelle Elemente der Route wider und dienen als Basis für die Externalisierung in verschiedenen Modalitäten, zum Beispiel in Sprache oder Grafik. Die abstrakten Routeninstruktionen beziehen explizit räumliche Charakteristika und Eigenschaften der Route mit ein und passen sich damit der gegebenen Situation an. Daher werden diese Routeninstruktionen kontextspezifisch genannt.

In meinem Ansatz werden Routeninstruktionen mit Hilfe eines Optimierungsschrittes erzeugt. Für jeden Entscheidungspunkt einer Route werden zunächst alle möglichen Instruktionen generiert. Basierend auf dieser Menge von abstrakten Routeninstruktionen werden in der Optimierung die Instruktionen gewählt, die bezüglich eines gegebenen Optimierungskriteriums am besten sind. Das Verfahren deckt Prinzipien für gute Routen-

instruktionen ab, zum Beispiel Chunking und das Einbeziehen von Landmarken. Daraus ergibt sich eine Sequenz von Chunks; jeder Chunk repräsentiert eine Instruktion.

Optimierung kann lokal oder global erfolgen. Lokale Optimierung bezieht nur die gerade getroffene Entscheidung in die Verbesserung der derzeitigen Lösung ein. Globale Optimierung erzeugt jede mögliche Lösung und wählt dann die beste Gesamtlösung aus. In meiner Arbeit vergleiche ich beide Möglichkeiten. Der Optimierungsschritt folgt generellen Anforderungen an gute Routeninstruktionen; es ist möglich, verschiedene Prinzipen für die Erzeugung von Routeninstruktionen einzubeziehen, zum Beispiel für das Chunking oder die Optimierung selbst. Einige sinnvolle Prinzipien und ihre Konsequenzen für das Erzeugen von Routeninstruktionen werden in der Arbeit diskutiert.

Eine informelle empirische Auswertung meines Ansatzes zeigt, dass kontext-spezifische Routeninstruktionen von Versuchspersonen als sehr geeignet eingeschätzt werden. Sie halten sie für leicht merkbar und geeignet, sich ein Bild von der Route zu machen. Außerdem sind die Versuchspersonen überzeugt, damit ihr Ziel erreichen zu können, was sich in einem zweiten Teil der Studie so auch zeigt.

Zwei Ergebnisse der Arbeit sind besonders hervorzuheben. Zum einen bietet sie eine Analyse von Routeninstruktionen aus kognitiver und repräsentationstheoretischer Perspektive, die deutlich Zusammenhänge und Konsequenzen aufzeigt und in einer Systematik von in Routeninstruktionen verwendbaren Elementen mündet. Zum anderen schlägt sie einen computationalen Prozess zur Erzeugung von Routeninstruktionen vor. Dieser ist so angelegt, dass er sich leicht erweitern läßt, um zum Beispiel zukünftige Ergebnisse der Kognitionsforschung zu berücksichtigen. Er kann so als Grundlage für empirische Studien dienen.

Abstract

In unfamiliar environments, people need assistance in order to reach their destination successfully. They need information on which actions to perform to get from their current location (the origin) to their destination. This task is termed *route following*; assistance for this task can be provided through *route directions*. In order to be useful, these route directions should reflect people's conceptualization of wayfinding situations, they should be well memorable and they should cover the spatial situations to be encountered while following a route. In this thesis, I present a process for generating context-specific route directions that allows for the generation of such route directions.

An analysis of structural and functional aspects of routes, as well as an analysis of route directions from a cognitive and a representation-theoretic perspective, reveals classes of elements that can be addressed in route directions. From a cognitive perspective, especially the role of landmarks, the integration of a sequence of instructions into a single instruction (so-called *chunking*), and the effects of using different reference systems are considered. Representation-theoretic aspects addressed include granularity, different types of knowledge involved in determining and conceptualizing references, and the difference between explicitly and implicitly represented information. The identified route direction elements are categorized in a systematics based on their level of granularity. The elements may take different roles in wayfinding depending on their function, which is also captured in the proposed categorization.

Based on the systematics, I develop GUARD, a process for generating context-specific route directions. It employs *abstract turn instructions*, which are a relational representation of the actions to be taken for route following that reflect conceptual elements of the route. The abstract instructions may serve as the basis for externalizing route directions in different modalities, for example, verbal or graphical. They explicitly take into account environmental characteristics and a route properties, i.e. they adapt to the spatial situation at hand. Therefore, the resulting route directions are termed *context-specific*.

The generation of route directions is realized as an optimization process. For each decision point of the route, all possible instructions are determined. Based on the resulting sets of abstract turn instructions, the optimization process selects the optimal instruction with respect to a given optimization criterion. The optimization criterion identifies the instruction that eases conceptualization the most, and accounts for different principles of good route directions, for example, chunking and a preference for references to landmarks. This results in a sequence of chunks, each representing an instruction for following the given route.

Optimization can either be performed locally, i.e. just updating the cur-

rent solution according to new selections made, or globally, i.e. creating each possible solution and then selecting the optimal one. In this thesis, I compare both options regarding their performance. The optimization process, while adhering to general basic requirements of good route directions, is flexible and able to integrate different principles of generating route directions (e.g., different chunking principles and optimization criteria). I discuss some sensible principles and their consequences for route generation.

An informal empirical evaluation of my approach indicates that context-specific route directions are well accepted by human subjects. They are deemed suitable for easy memorization and for conceptualizing the route at hand, and subjects believe that they can reach the destination with their help. In the route-following part of the study, subjects are indeed able to find their way using context-specific route directions.

Results of this thesis are twofold. On one hand, it provides a thorough analysis of route directions from a representation-theoretic perspective resulting in a systematics of elements that can be employed in generating route directions. On the other hand, it provides a process for generating route directions which is designed to ease including further and new findings in cognitive science and may serve as a test-bed for empirical studies.

Acknowledgments

Writing this thesis took considerable time and effort. During this time, there have been many people throughout this time that helped me one way or another. Several of them deserve my special thanks.

I would like to sincerely thank Christian Freksa, my advisor, for his ongoing encouragement and support. I deeply enjoyed the openminded, interdisciplinary research environment that he created. He has given me great freedom in pursuing new ideas and directions, and has always been there if I needed advice. Our discussions and his comments helped me very much in sharpening and focusing my ideas.

I thank Stephan Winter for being my second supervisor and very much appreciate his insightful comments on my draft. Also, I am grateful that I can spend my first months as a post-doc in his group at the University of Melbourne.

I extend special gratitude to Alexander Klippel for a fruitful, inspriring collaboration. Our joint work contributed a lot to shaping my ideas. He also took it as his personal quest to keep me on track and make me go on.

I would like to thank Prof. John Bateman for all the interest he took in my work and especially, for our cooperation on generating natural language route directions from my abstract specifications. This has been invaluable in evaluating my work.

Holger Schultheis and Gregory Kuhnmünch need to be thanked for discussing my ideas for the evaluation study. Their feedback helped me a lot in desiging the study. And Holger has been invaluable in showing me what to do with my data in SPSS. Christopher Galbraith took care that my English is readable and taught me when to use a colon.

This work has been done in the project I2-[MapSpace] of the Transregional Collaborative Research Center SFB/TR 8 Spatial Cognition. I like to thank the Deutsche Forschungsgemeinschaft (DFG) for funding this highly inspiring research environment. I thank all the researchers of the SFB who took interest in my work and discussed my ideas. A special big thank you goes to all my colleagues of the Cognitive Systems group who beared with me talking about the same things over and over again for the last couple of years and who gave valuable feedback in our graduate seminar sessions and beyond.

Parts of this work has been presented at several conferences and workshops. Several colleagues world-wide provided feedback on these occasions either as reviewers or in discussions at these events. Especially, I would like to thank Sabine Timpf and Urs-Jakob Rüetschi for inspiring discussions whenever we met somewhere.

I want to thank my parents for always letting me do what I wanted to do and for all their patience if things took much longer than I expected them to take.

Finally, I owe much to my wife Anja. It is hard to express how much I appreciate her support during the last years. I am deeply indebted to all her emotional support, her belief in me, her support in getting my work done, and her love. She also offered a lot of practical advice, has always been willing to discuss my work, and spent much time and effort in taking and postprocessing the photographs for my evaluation study. I dedicate my thesis to her.

Contents

Chapter 1

Introduction

Finding your way in an environment where you have never been before is a demanding task. You need to identify your location, the location of your destination, and then determine a way that leads you to that location. This involves repeated informational updates about your location and of the destination's location in relation to yours, as well as repeated decisions regarding the way to take, while avoiding bumping into objects and stumbling over your feet.

There are several means of assistance to simplify this task. A guide that is familiar with the environment can lead your way—this guide may be another person or some electronic device, for example a car navigation system or software running on a personal digital assistant (PDA) or mobile phone. In the future, you may even be guided by a mobile robot, or so-called smart environments may have signage or devices that adaptively provide information on the way to take for each individual visitor. You can ask for directions either before starting your trip or while navigating in the environment. In the environment, you may get information from your mobile device or from people you encounter, or outside the environment from someone who knows the way or from some route information system. This way you may get oral or written instructions on how to reach your destination. You could also obtain a graphical depiction of your route, typically either an overview map showing the environment or a sketch or route map just showing the way to take, or a mixture of these kinds of instructions.

All these kinds of support differ in several respects. The first notable difference is that some rely on support by humans, while with others, support is provided by a computational system. This has consequences for interaction: with a human you often can engage in a dialog clarifying instructions you did not understand or requesting more information. With a computational system, you are usually stuck with the information it provides without the option to request additional information. In some cases assistance is given while you are already on your way to the destination, or the required information is requested before even reaching the unknown environment. This has consequences for communicating the information: if you receive assistance while already being in the environment the provided information is usually easier to understand as it is (at least partially) directly related to what you currently perceive. Depending on how the assistance is provided (e.g., when relying on a guide or mobile navigation system), it is also possible to present information stepwise, i.e. to always communicate only what you need to know when you need to know it. Acquiring instructions outside the environment before your trip starts does not offer this option. Here, all information required for reaching your destination needs to be communicated at once, and this information cannot be related to your current perception. Finally, the information may be communicated either graphically as some kind of map or verbally (oral or written), or as a combination of both. This has consequences for interpreting the information: maps necessarily display exactly one spatial situation (which sometimes may not cover the real situation well), while language may leave open more than one possible interpretation of the described spatial situation. Maps provide an overview on the way to take that is accessible all at once, while language is usually more focused on the actions to be performed and is necessarily sequential in providing information. To sum up, these differences result in different requirements, constraints, and problems in the way assistance needs to be given and in the way this assistance may be used by a wayfinder. Consequently, they result in different requirements and challenges for generating this assistance automatically that can hardly be captured in a single application.

In my thesis, I concentrate on wayfinding assistance communicated before starting to follow a route. The assistance provides information on the complete way to take. It is not restricted to a specific modality, but is represented as an abstract relational specification of the actions to be executed. Assistance is generated by a computational system that shall account for human principles of good wayfinding instructions.

An Example

To clarify this aim, let us look at an example of what different instructions for following a route may look like. Figure 1.1 shows a route that starts on one side at the bottom of a hill. It then runs upwards to the top, and from there it goes downhill to the other side of the hill. There are several intersections along the way that offer choices for paths in different directions. The example route is a hiking trail. In the real environment, it is marked with a sign showing a dot.

There are several ways to describe this route. The following examples list just some possible verbal instructions that illustrate important differences.[1]

[1] Figure 1.1 itself provides another way for describing the route: a map.

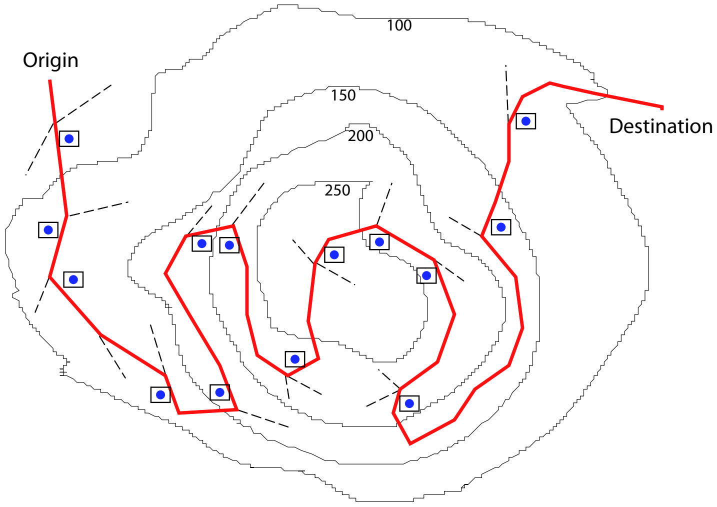


Figure 1.1: A fictitious route over a hill. The dotted lines indicate alternative ways that could be taken at the intersections; they are not completely depicted. The boxed dots along the route represent the signs marking the hiking trail.

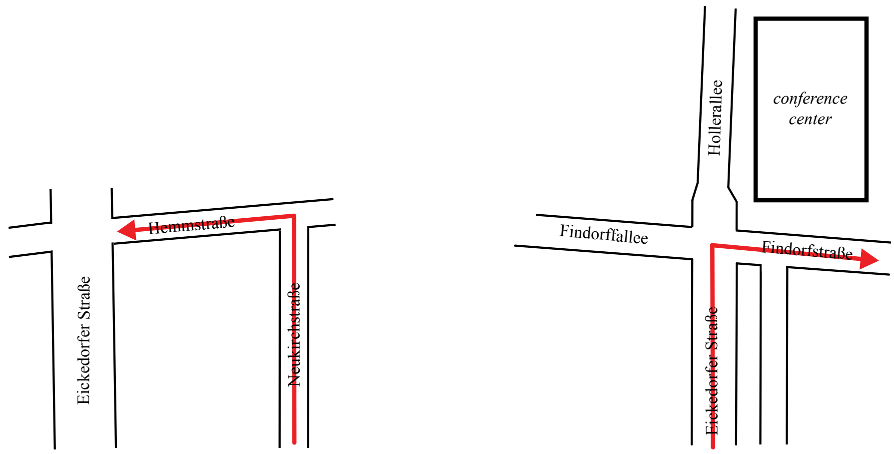
Each listed example consists of several instructions; each supposed to keep a wayfinder on the route.

1. "Go straight, turn right, turn left, turn left, turn right, turn left, turn right, turn right, turn left, go straight, turn right, turn right, turn left, turn right, turn right."
2. "Turn at the second right. Turn twice left. Turn right and left again. Turn twice right and then left. Then straight on, and again twice right. Then left, and twice right."
3. "Go up the hill and down again."
4. "Just follow the markers."

You can make several observations here. The most obvious is that the lists of instructions differ significantly in their length. This is not just the length of the printed text, but also length with respect to the mentioned number of actions to be performed. The single instructions (single steps) refer to different features of the environment. While 'turn left', for example, refers to an intersection that offers the option to choose a left branch, 'go up' refers to the hill itself, and 'follow the markers' to signage. The description of the actions to be performed also differs.

Less obvious, you can also observe that the complexity of understanding and of using the different lists of instructions differs. The first list consists of a lot of single steps, each referring to a single decision—a single intersection—while the third and fourth have only very few single instructions—two or one, respectively. The second list states the same instructions as the first, but combines several consecutive instructions into a single one. The first list of instructions is difficult to remember entirely, while the third and fourth are rather easy. On the other hand, the first and fourth lists are specific, i.e. when following these instructions correctly you inevitably follow one specific route. The third list of instructions, though, is much more ambiguous; you may well end up using a different route up or down the hill than the one shown in Figure 1.1.

This small example illustrates that the way instructions are given significantly influences the ease of route following. In providing instructions, exploiting features a wayfinder will encounter in the environment results in instructions that are well suited for easing this task. Thus, adapting the instructions to a route's properties and environmental characteristics is of great benefit for wayfinding assistance.



a) b)

Figure 1.2: Two different spatial situations treated similarly by internet route planners: a) turning left at a dead end; b) turning right at a landmark.

Such an adaptation to the characteristics of the environment is largely missing in today's automatic systems, especially internet route planners. Their instructions are generated for a specific route, but they only refer to street names and road category. For example, Figure 1.2 depicts two intersections in Bremen. The first is a T-intersection at which the wayfinder is supposed to turn left; at the second depicted intersection a right turn is required. Here, a big, well visible conference center is located that can be used as landmark to identify the intersection. However, both spatial situations are treated in the same way by internet route planners:[2] the generated instruction for the first intersection is "leave Neukirchstraße and turn left into Hemmstraße. Stay on for 76 m;" for the second intersection "leave Eickedorfer Straße and turn right into Findorffstraße. Stay on for 59 m." Taking into account the available environmental features, possible instructions would be "turn left where the road dead ends" for the first intersection and "turn right at the conference center" for the second. These instructions adapt to the spatial situation at hand. As will become apparent in the following, such instructions are much closer to the way humans structure their environment and to the way they provide wayfinding instructions.

1.1 Wayfinding, Route Directions, and Their Conceptualization

Route following is a specific kind of *wayfinding*, which—when performed with assistance—involves the *conceptualization* of *route directions*. These and other key concepts and terms are introduced in this section.

Navigation and Wayfinding

The terms *navigation* and *wayfinding* are used in different contexts with different meanings. In my thesis, I adopt them in the sense introduced by Golledge (1999) and Montello (2005).

According to Montello, *navigation* is "coordinated and goal-directed movement of one's self (one's body) through the environment" (2005, p.$\tilde{2}$58) that aims at getting from some location to a destination. Navigation consists of two components: *locomotion* and *wayfinding*. Locomotion refers to movement around the immediate surrounding environment, i.e. that part of an environment that is directly accessible to sensory and motor systems. In other words, locomotion concerns processes like obstacle avoidance and coordinated movement of one's legs for walking. Wayfinding, on the other hand, reflects the cognitive processes going on while getting from origin to destination. Golledge (1999) considers wayfinding to be a purposive, directed, motivated activity to follow a route from origin to destination. Wayfinding involves most of the issues addressed in the first paragraph of this chapter: localization of yourself and the destination, deciding on the way to take; but it also involves planning a way either at the start of or before a trip—with or without assistance.

[2] The example instructions are generated by www.map24.com (on 20 March 2007); other route planners generate similar instructions.

Structure and Function, Path and Route

In this thesis, wayfinding is about following a route from origin to destination, where origin and destination are places in an environment. While route following is a behavior for which only some features of the physical world are relevant, the world exists independent of that behavior. This difference is reflected in two distinctions explained in the following: the distinction between *structure* and *function*, and—based on that—the distinction between *path* and *route*.

Structure and Function Klippel (2003b) introduces the concepts of *structure* and *function* in wayfinding. With structure he refers to the object level, while function refers to actions. Structure denotes the physically present features of an environment, like streets or buildings, and their static configuration. Function, on the other hand, captures the relation of the structural elements to actions performed in the environment. The functional level demarcates those features relevant for wayfinding actions, i.e. it describes a dynamic situation and those parts of the structure that are relevant for it. This distinction reflects that conceptualization can be applied to both objects and actions. According to Klippel (2003b), on a structural level the conceptualization of an intersection stays the same no matter what action is performed; on a functional level there is a difference between going straight and turning left at an intersection, for example. This is captured in the distinction between path and route explained in the following.

Path and Route Based on the distinction between structure and function, Klippel (2003b) further distinguishes between path and route which takes up a distinction made by Montello (2005).[3] A path consists of a sequence of *path-segments*. Path-segments are (parts of) linear physical entities that you can walk along (like streets). They are undirected and connect two *branching points*. These are the points where three or more path-segments meet. They correspond to intersections in a street-network. Taken together, path-segments and branching points form a *path-network* (see Fig. 1.3a,b).

A *route* represents a behavioral pattern. It describes the way someone takes from origin to destination; thus, routes are directed. They are also bounded as they start at the origin and end at the destination (or another place if the intended destination is not reached for some reason). A route determines which path-segments of a path-network get traversed in which order. From a functional perspective, these path-segments are called *route-segments* and denote those parts of a route a wayfinder travels between

[3] Montello's book chapter has been available as 'in press' for several years before publication. This, and the fact that A. Klippel and D. Montello discussed this issue as part of their ongoing cooperation, explains the seemingly contradicting publication dates.

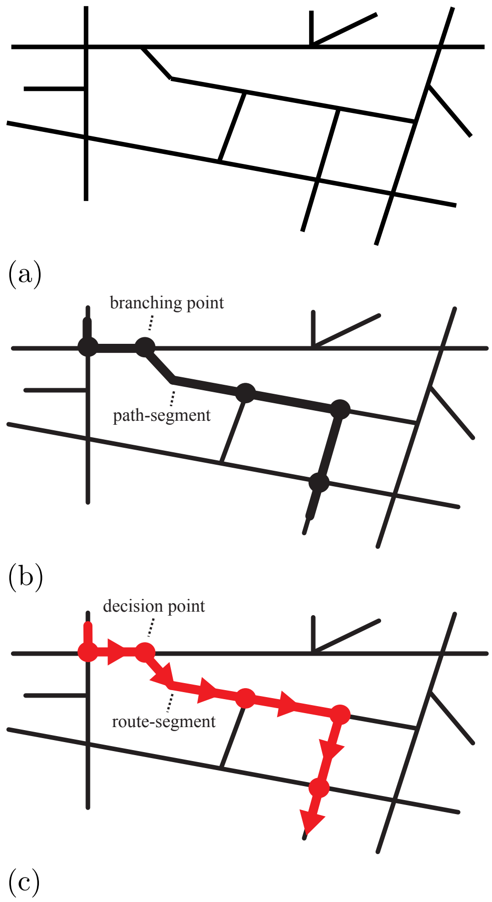


Figure 1.3: (a) A *path-network* with (b) emphasized *path* (the bold line); a *path-segment* connects two branching points. In (c), the corresponding *route* with *decision points* and *route-segments* is depicted.

two *decision points*. At a decision point, a wayfinder needs to decide which further way to take, i.e. to identify the next route-segment. A decision point corresponds to a branching point (an intersection) on the structural level (see Fig. 1.3c).

Typically, a path-network is represented as a graph. Nodes represent branching points and edges represent path-segments. Accordingly, a path in the graph represents a physical path in the environment. Whenever it is not evident from context, I will disambiguate the two usages of path by explicitly refering to “physical path” or “path in the graph”, respectively. The discussion returns to representing paths and routes in Section 5.1.2.

Table 1.1 summarizes the terms introduced in this section and lists their correspondences. There is no counterpart to a path-network on the functional level. Routes are defined as a behavioral pattern capturing the movement that a single person performs while getting to a destination. Even though a person might use (a combination of) known physical paths that she traveled on before to reach that destination, the route is still linear. Branching in routes is not possible. There are networks that represent multiple, possibly connected lines of travel, for instance public transport networks. These networks do not represent individual behavioral patterns, but a specification of the paths public transport vehicles take on a regular schedule. The movement of each individual vehicle then corresponds to a route. A wayfinder might use one or several of these individual vehicles that cover part of the way to the destination. This results in an individual behavioral pattern of the wayfinder, i.e. a route, which partially coincides with the route of one or several public transport vehicles. However, I do not address wayfinding in public transport networks in this thesis.

Table 1.1: Structure and function in wayfinding: differences and correspondences between path and route.

Structural Perspective	Functional Perspective
path	route
path-segment	route-segment
path-network	—
branching point	decision point

Route Directions

This thesis is about generating assistance for following a route, i.e. generating instructions on getting from origin to destination. I use the term *route directions* (cf. Daniel & Denis, 1998; Schweizer et al., 2000) to refer to these instructions. Route directions are not merely a description of what the wayfinder will encounter along the route, but also include references to actions at decision points, to landmarks, and confirmative information that the right track is still followed. A route is directed; consequently, route directions are also directed. Instructions for following a route from some point A to a point B differ from those for following a route from point B to point A. It is usually not possible to just reverse route directions to get instructions for the way back (cf. Schweizer et al., 1998).

A wayfinder may receive information on which actions to perform while following a route or before route following starts. That is, route directions may be provided online en route or offline beforehand. Route directions given online usually just state the next action to be performed, i.e. an instruction for the next decision point. That is, for example, what car navigation systems do. I term this kind of route directions *incremental route* *directions*. Providing route directions offline before route following starts requires that they are presented as a whole, i.e. that a wayfinder receives information on the complete route and, accordingly, is able to take a sensible decision at each decision point. I term this kind of route directions *in-advance route directions* (cf. also Habel, 2003).

Conceptualization

The notion of *conceptualization* plays an important role in this thesis. With conceptualization, I refer to a mental representation of a route that results from the interpretation of route directions. It also denotes the mental process leading to this representation.

Conceptualization of different kinds of route directions differ. In incremental route directions, for example, an instruction is given for the situation the wayfinder is currently in, i.e. the instruction and the spatial situation are co-located. Accordingly, matching the instruction and the encountered situation is easy. And the cognitive load is low, since the wayfinder does not need to form a conceptualization of (considerable parts of) a route, but just needs to correctly interpret the instruction for the next decision point to come.

With in-advance route directions, the situation is different. Here instructions are provided on the complete route while the wayfinder is outside the environment and has no immediate perceptual feedback. That is, from the received route directions a wayfinder needs to form a mental representation of what to expect: the actions to be performed while following the route and the spatial situations the actions refer to. Since wayfinding instructions mostly concern changes of movement direction at decision points, I concentrate on these direction changes in my thesis.

Thus, with conceptualization I denote the (process of forming a) mental representation of an (expected) decision point sequence with their accompanying actions.

1.2 Thesis, Aim, and Approach

In this thesis, I deal with in-advance route directions that are presented to a wayfinder before route following starts. As I already stated, in-advance route directions differ significantly from incremental route directions; thus, I restrict my work to them to keep my thesis focused. In generating route directions, the wayfinder is assumed to not know the environment. Accordingly, for every decision point an instruction is generated that indicates how to proceed at this decision point. The instructions are generated for a specific route, guiding a wayfinder along this route in such a way that at each decision point the branch to take is unambiguously identified. The wayfinder is assumed to move as a pedestrian in an outdoor path-network unknown to her—though many of the claims made in this thesis hold for other kinds of movements as well. I concentrate on the wayfinder's point of view, i.e. the focus is on the wayfinder's ability to use the route directions and on the consequences different possibilities for giving instructions have on a wayfinder's conceptualization.

The aim is to identify and apply cognitive principles for providing good route directions in a process of automatically generating route directions and, at the same time, to account for representation-theoretic considerations, for example, different kinds of spatial knowledge involved. The route directions shall take into account environmental characteristics and a route's properties. This way, the route directions adapt to the situation at hand, i.e. the current action to be taken in the current surrounding environment. This directly reflects Dey's definition of context: "[...] any information that can be used to characterize the situation of an entity" (2001, p. 5). Therefore, I term this kind of route directions *context-specific route directions*. The thesis of my work is:

> The generation process for context-specific route directions results in route directions that are easy to conceptualize as they adhere to principles of human direction giving and are easy to use since they adapt to the situations a wayfinder encounters while following a route.

The process results in instructions that are well suited for a given situation in that they are unambiguous and easy to use. In order to generate route directions that fulfill these demands, a good understanding of the principles and mechanisms of how humans give and interpret route directions is necessary. As a key contribution, I identify common principles that shall be employed in the automatic generation of route directions. To this end I analyze exisiting literature on wayfinding and route directions and relate the findings. There is a plethora of existing research on principles of good route directions, how humans give route directions, which features of route directions are helpful, and what the pitfalls in providing route directions are. What is mostly missing though, is a coupling of cognitive principles with representation-theoretic considerations that allows bridging the gap between human conceptualization of spatial situations in route following and a computational generation of route directions.

For an automatic generation of route directions it is necessary to automatically determine different kinds of references which includes identifying (configurations of) features of the environment that may serve as reference objects. I provide an analysis of which features of an environment may be addressable in route directions and which spatial knowledge is connected to them. Based on these analyses, I identify and categorize classes of elements that may be used for the automatic generation of route directions. These elements differ in their relation to the route and the surrounding environment, but also in the kind of spatial knowledge involved. The elements afford different actions in route following, which is reflected in the instructions by the different relations associated with the elements.

Automatically generating in-advance route directions requires several steps in the generation process. I identify the ones that are crucial for fulfilling the requested properties and I develop a process for generating context-specific route directions. This process is then implemented in a software prototype that automatically generates abstract route directions. This system is evaluated in different respects: a computational evaluation demonstrates the feasibility of the chosen approach; an exploratory human subject study provides hints that the route directions are well usable by wayfinders.

1.3 Contributions

The thesis provides several contributions to the field of spatial cognition in general and to research on route directions in particular. The key contributions are:

- A thorough analysis of human wayfinding instruction principles. This comprises an analysis of cognitive and linguistic principles of human wayfinding instructions as well as representation-theoretic considerations. It links both aspects by pointing out correspondences between them and by explaining consequences of cognitive principles on the kinds of spatial knowledge involved in giving and interpreting route directions. The analysis results in a systematics of route direction elements.

- Guard, a process for generating route directions that covers both the determination of all possible instructions that unambiguously represent the action to be performed at a single decision point and the selection of specific instructions to generate route directions for a complete route. This process provides methods for generating different kinds of instructions and, especially, for handling different types of landmarks in the automatic generation of route directions.

- An implementation of the process that may be used to test consequences of different route direction principles and their sensible combination. The system is flexible with respect to utilizing different principles and may, therefore, serve as test-bed for further (empirical) research.

1.4 Organization of the Thesis

In this thesis, I discuss how cognitive principles of giving and interpreting route directions and representation-theoretic considerations regarding route knowledge can be linked for exploiting a route's properties and environmental characteristics to improve route directions. The next chapter introduces relevant findings on which my approach is based. These findings stem from cognitive and linguistic research on wayfinding, route following, and giving and interpreting route directions. This is accompanied by a representation-theoretic view on route directions. I explain the basics of representation theory and present work on the (qualitative) representation of space, especially the representation of actions and landmarks. I also present other approaches to an automatic generation of route directions and their relation to my work.

Context-specific route directions are based on a systematics of elements that may be used in such route directions. Chapter 3 introduces this systematics—here I discuss the properties of route directions identified in Chapter 2 both from a cognitive and a representation-theoretic point of view. Chapter 4 presents GUARD, the process for context-specific route directions. It employs two main principles: grouping instructions for several single decision points into a single instruction for a sequence of decision points, and determining route directions for a complete route by means of optimization. In Chapter 5, I present a prototypical implementation of the process that generates abstract relational route directions; different optimization approaches are discussed in more detail. Chapter 6 demonstrates how the system works, presents results of its application to generating route directions, and discusses differences resulting from applying the different optimization approaches. Also, the results of an informal, exploratory human subject test are presented that provide hints for the performance of context-specific route directions. Chapter 7, finally, summarizes the achievements of the thesis and provides an outlook on further possible enhancements and applications as future work.

Chapter 2

Human and Automated Route Directions: A Survey

The process presented in this thesis reflects principles of human direction giving. It addresses different kinds of spatial knowledge. In this chapter, I detail the foundations of my work as drawn from the literature. I concentrate on cognitive and linguistic findings on giving and understanding route directions, as well as on questions of knowledge representation relevant for my work. Additionally, I present computational approaches that address the automatic generation of route directions or parts thereof that explicitly take cognitive considerations into account; I explain the relation of this research to my work.

2.1 Cognitive and Linguistic Findings on Route Directions

In spatial cognition, considerable effort has been spent in investigating how people produce and comprehend route directions. This has several reasons. Finding one's way in an environment is a prime problem of human beings and one of their important skills. Route directions focus on the actions to be performed in an environment; they allow investigating the dynamic aspects of spatial cognition, rather than being limited to static scenes or environments. Production and comprehension of route directions strongly hint at people's gathering and organization of spatial knowledge, providing a window into human cognition. And since route directions are inherently sequential, they are conceptually comparatively simple.

2.1.1 How Do People Give and Understand Route Directions?

A prerequisite for producing route directions is the availability of spatial knowledge about the environment—the instructor needs to have sufficient knowledge on the route in order to have the ability to provide the instructed agent (the *addressee*) with information that helps her to reach the destination. The addressee in turn does not need to know the environment; she just needs a general understanding of spatial concepts to be able to interpret the route directions.

The instructor may provide the required information in different modalities. Typically, this is done either verbally (written or oral) or graphically (sketch- or route-maps). Tversky & Lee (1999) assume a common conceptual structure underlying the knowledge represented in verbal and graphical instructions. They term the former *descriptions* and the latter *depictions*. Both descriptions and depictions are adequate to convey information sufficient for arriving at a destination. Tversky & Lee (1998, 1999) claim that descriptions and depictions consist of the same information schematized in the same way. Accordingly, descriptions and depictions are used interchangeably in route directions (Tversky & Lee, 1999). Klippel (2003a) agrees with Tversky and Lee on a common conceptual structure underlying route knowledge, however, he challenges their claim that graphical and verbal elements of route directions map onto one another. This issue will be further discussed in Section 2.3.

Most empirical research on route directions has focused on verbal route directions. According to Daniel & Denis (1998), route directions are a special kind of discourse in that they are a unique combination of four features: (1) their purpose (or function), (2) the content (a composite of several types of discourse), (3) their structure that is matched to the structure of objects they describe, and (4) the perspective they impose on their users. The purpose of route directions is to provide the addressee with a set of procedures and descriptions that allow her to form a conceptualization of the route in advance (Michon & Denis, 2001). Route directions focus on the actions necessary to reach the destination—they are procedural discourse. Often, however, instructors also describe spatial situations to foster conceptualization of the route. Thus, route directions also contain descriptional discourse. In route directions a *route perspective* is usually imposed on the addressee that reflects the successive positions of a wayfinder along the route and describes local views in an egocentric perspective. A route determines the sequence of spatial information based on the spatial sequence of locations encountered during route following. Therefore, the *linearization problem* that occurs in spatial discourse does not occur here (cf. Denis et al., 1999). Sentences are linear; speakers—or writers—are forced to make choices among many possible sequences when organizing spatial information that extends over two or three dimensions. In routes, this information is only one-dimensional.

Kuipers (1978) terms the combination of physical elements, which can be perceived in the current egocentric view, and motor activities *view-action pairs*. The core of route directions is a sequential representation of these view-action pairs, for instance 'walk' or 'turn [left]' (Lovelace et al., 1999). Allen (1997, 2000) provides a more distinct differentiation of route direction elements: route directions consist of a description of *environmental features*, i.e. entities that can be observed along a route, *delimiters*—these are distance and direction designators stating spatial or temporal distances between decision points ('300 meters', 'one block', '3 minutes'...) or spatial relations in different frames of reference ('left', 'north'...), delimiters stating the relation of a wayfinder to environmental features or between environmental features as prepositions (like 'toward', 'between', 'in front of'...), and *verbs of movement* and *state-of-being verbs*. According to Michon & Denis (2001), there are three different entities referred to in route directions: those on which moves are executed, reorientation points, and landmarks. Streets are a typical example of entities moves are executed on. These entities can be represented as vectors which also capture the movement direction. Reorientation points are identified by statements like "at the end of the street." They are represented as points on the vectors. Reorientation points may also be identified by landmarks. These are three-dimensional objects in the real world; in the representation, their location along the route can be abstracted to points on the vectors.

In using these elements, three principles of "best practice" are important to ease the comprehending and following of route directions (Allen, 2000). The *principle of natural order* states that references to environmental features should be in the order of the features' appearance in the environment, i.e. that instructions should be given in order of walking direction. According to the *principle of referential determinacy* use of delimiters should concentrate on descriptives or directives for decision points, i.e. communication should focus on what to do at decision points as the most important information. The *principle of mutual knowledge* restricts types of delimiters used to those known and comprehensible for both communication partners; for example, refraining from using cardinal directions if the addressee is not trained in determining her bearing.

Denis (1997) identifies the following structure of route directions: the addressee is located and orientated at the starting point (this usually involves references to a landmark), progress along the route then starts, which is the first action that is most often given only implicitly. With an announcement of a landmark, progress stops. Now, the wayfinder needs to re-orient. To this end, the wayfinder is provided with an instruction announcing the change of direction, often with reference to the landmark. After re-orientation, progress continues until the next landmark is identified, and so on. Thus, following a route basically consists of two processes: getting to a decision point and, there, deciding on the further direction to take (cf. Daniel & Denis, 1998). Accordingly, decision points are deemed crucial for wayfinding (e.g., Schweizer et al., 2000).

Producing and interpreting route directions involves cognitive costs (e.g., Freksa, 1999). Often, verbal route directions are accompanied by gestures as these facilitate a speaker's ability to produce spatial content (Allen, 2003). These gestures illustrate the spatial situations that will be encountered by the addressee and may, therefore, ease conceptualization of the route. But in general, everything that eases production of route directions for the instructor involves cognitive costs for the addressee (Schweizer et al., 2000). The following nicely summarizes the relevant characteristics of route directions that are to be taken into account when developing a computational process for the generation of route directions:

> "Because route descriptions are intended to help a mover navigate, the speaker must be able to create favorable conditions for communication, monitoring the amount of information transmitted to maintain adequate specificity but refrain from overspecification. In sum, the describer should adjust the output to suit the processing capacities of the addressee. Not only should the content be exact, but the amount and organization of that content should not exceed the addressee's processing capacities. Heavy demands are usually placed on the user's short-term memory capacities, especially when the description must be retained during the displacement." (Daniel & Denis, 1998, pp. 49–50)

Determinacy of instructions is crucial. In a set of experiments, Schneider & Taylor (1999) showed that both redudancy (termed overdeterminacy by Schneider and Taylor) and indeterminacy have a negative impact on the usage of route directions, with redundancy being even worse than indeterminacy. Redundancy usually results in long route directions with a lot of (excess) information. It, therefore, negatively influences learning and memory. Indeterminacy has not been immediately apparent from the directions used in the studies, but emerged when using them. Therefore, it results in errors during wayfinding. Indeterminacy does not have an influence on learning in this case since the subjects have not been aware that there are several possible interpretations. Another crucial aspect 'to create favorable conditions' is the inclusion of references to landmarks in route directions—especially such references that link landmarks to the actions to be performed (see Daniel & Denis, 1998; Daniel et al., 2003). This will be further discussed in Section 2.1.3.

Several processes are involved when navigating in an environment using route directions (Daniel et al., 2003): a wayfinder needs to elaborate an internal representation of the described environment, and needs to keep that internal representation in mind during route following. The internal representation is compared to perceptual information available along the route. Especially, it is used to identify the relevant landmarks, i.e. to match the characteristics of the spatial mental model derived from the description with the characteristics of the environment perceived during successive stages of navigation.

Many of these findings stem from a series of experiments done by Michel Denis and co-workers (see Denis, 1997; Daniel & Denis, 1998; Denis et al., 1999). The general setup of these experiments is the following: Denis and coworkers collect route directions for a given route from a number of subjects. These route directions are classified in a rating test as either 'good' or 'poor' by a group of judges. All statements of the subjects are then combined into a so called *megadescription*. This megadescription, again, is evaluated by a group of subjects; these subjects remove every statement they deem superfluous from the megadescription. This way, Denis and his coworkers generate a *skeletal description*; only statements that are selected by at least 70% of the judges contribute to this description. Finally, the performance of good, poor, and skeletal route directions are tested in a wayfinding study. The general findings of these studies are that there is no significant difference in people familiar or unfamiliar with an environment that rate different route directions with respect to their assumed quality. Along the same line, there is no significant difference in performance between people using a description rated as 'good' and people using a skeletal description. In conclusion, the generation process of skeletal descriptions can be considered a valid method of generating good route directions.[1]

All these findings cumulate in different, yet similar, cognitive models for producing route directions which are presented in the following.

2.1.2 Cognitive Models of Producing Route Directions

Common to all cognitive models of producing route directions is that they assume the instructor to activate her spatial knowledge on the environment, to select the relevant information for the chosen route, and to transfer this information into verbal statements.

Couclelis (1996) presents a model that represents a complex schema of route direction production governing the instructor's responses and integrating her goals, attitudes, and behaviors with the spatial and linguistic cognitive skills necessary for carrying out the task. In the *initiation step*, the addressee requests directions. This is confirmed by the instructor who states her intention to help and activates a *direction-giving schema*. Next, the necessary knowledge is activated in the *representation step*. Here, the instructor adopts a *linguistic stance*, i.e. she decides that the directions will

[1] But this method is, quite obviously, hard to automatize and hard to turn into a computational assistance system.

be communicated using natural language. To this end, *spatio-linguistic constructs* of the relevant basic-level actions and basic-level categories of objects are called up and a *relative frame of reference* is established.

The actual route is then planned, first on a coarse level connecting major points along the route, and then on the fine level, the route is traced mentally from beginning to end, recalling each necessary step and the most striking or memorable entities along the route—the landmarks. To verbalize this information, it needs to be *transformed* in the next step where the route is *linearized* and *segmented*, usually at landmarks and decision points. The segments will be turned into statements later on. *Segmentation* also involves *selection* of the relevant information to communicate. Now the information is organized such that it can be verbalized, which is done in the *symbolization step*. Here *expressions* are generated and *reinforcement* statements (like "OK?") or repetitions, are added. Finally, the interaction ends with *termination* phrases (like "Ok, thank you!")

Allen (1997) points out that production and comprehension of route directions occur in a context with cognitive, linguistic, social, and geographic dimensions. Implications of these dimensions are reflected in the structural organization of a route communication episode, which is divided into four phases. In the *initiation phase* communication begins with an *activating query* ("How do I get to…?"). This query may be answered by a *destination confirmation*, which may be seen as a social signal that the request is accepted, or a *state-of-knowledge query* of the instructor, which ideally determines the amount and type of information needed by the addressee. Second, and critically important, is the *route description phase*. Here, the instructor provides a set of *communicative statements* that contain sufficient information to reach the destination. These statements consist of directives and descriptives as explained in the previous section. The addressee may answer these statements with *comprehension queries* confirming that she understood the instructions. In the *securing phase*, the addressee reacts to the route description with *clarification queries* and *confirmation statements* which are answered by the instructor. Finally in the *closure phase*, the transaction is ended in a way that follows social conventions.

Denis (1997) proposes a model with stronger focus on the spatial knowledge and mental processes involved (cf. also Daniel & Denis, 1998; Denis et al., 1999). The instructor is believed to implement three sets of cognitive operations. The first operation is the *activation of an internal representation* of the environment. This spatial knowledge may be heterogeneous containing topographical information, visual aspects of the environment seen from an egocentric perspective, and procedural components derived from the instructor's prior exploration of the environment (see also the discussion on cognitive collages; Tversky, 1993). Second, the instructor *defines the route* that best fits the request within the currently activated mental representation. For determining the sequence of route-segments and decision points, the instructor may take into account a variety of criteria, like travel mode of the addressee or assumed previous knowledge. While this is mainly a pre-verbal operation, the instructor already needs to take into account that the route will be described verbally—that is, (small) detours may be admitted if they are easier to describe than the direct, shortest way. Finally, the third operation is *formulation of a procedure* to follow the route. Verbalization of the route is based on the subdivision of routes into route-segments connecting decision points; this subdivision determines the linear order of the statement sequence. All three operations may work concurrently. For example, formulation of the procedure may begin before the complete route has been defined.

Lovelace et al. (1999) present a simplified model of route direction production that summarizes the models presented above. It consists of three steps shown in Figure 2.1 and, like Denis' model, focuses on processing of spatial knowledge.

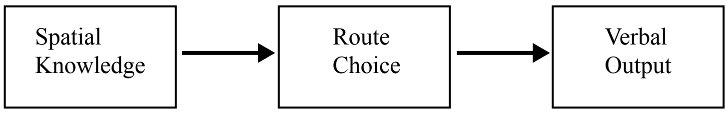


Figure 2.1: Route direction production as a three-step process (from Lovelace et al., 1999).

First, an instructor needs to *activate spatial knowledge* of the environment. The second step requires a *choice of a specific route* through the environment from origin to destination. This choice may be based on different criteria, like travel mode, desired route characteristics, or expected level of knowledge of the addressee. Third, the chosen route needs to be translated into a *set of verbal instructions*. Lovelace et al. consider these three steps the central part of a route direction discourse: initiation and termination activities are not relevant when dealing with written instructions. This holds, for example, for computational systems like internet route-planners.

2.1.3 Landmarks

Landmarks are crucial elements of route directions; humans heavily rely on them when giving wayfinding instructions. Consequently, the question arises what are landmarks and why are they so important for producing and comprehending route directions?

What is a Landmark?

Many definitions of the term *landmark* can be found in the spatial cognition literature. Common to all, landmarks are defined as entities that are easily recognizable and memorizable. Presson & Montello (1988) provide a general definition stating that everything that stands out of the background may serve as a landmark. In his seminal book 'The Image of the City', Lynch (1960) discusses why a feature may serve as a landmark:

> "Since the use of landmarks involves singling out of one element from a host of possibilities, the key physical characteristic of this class is singularity, some aspect that is unique or memorable in the context. Landmarks become more easily identifiable, more likely to be chosen as significant, if they have a clear form, if they contrast with their background; and if there is some prominence of spatial location. Figure-background contrast seems to be the principal factor. The background against which an element stands out need not be limited to immediate surroundings: [...]." (pp. 78–79)

Lynch continues to elaborate further on spatial prominence: "[it] can establish elements as landmarks in either of two ways: by making the element visible from many locations ([...]), or by setting up a local contrast with nearby elements, i.e. a variation in setback and height." (p. 80)

Sorrows & Hirtle (1999) pick up these considerations and list features that may let a landmark stand out: *singularity*, *prominence*, *meaning*, and *prototypicality*. Singularity applies to objects that are in sharp visual contrast with their surroundings. Prominence of objects refers to their spatial location in an environment such that they are visible from many other locations or are located at a significant point, for instance a major intersection. Some objects may be used as landmarks because they have a meaning common to many people stemming, for example, from their cultural or historical significance. Prototypicality characterizes objects that are referred to because they are typical representatives of a specific category. Such prototypes are easy to learn, recognize, and categorize and are, thus, good candidates for landmarks as they are commonly identifiable (cf. Rosch, 1978). Sometimes, such landmarks may represent a local environment on a larger scale, like the Eiffel Tower representing Paris (Sorrows & Hirtle, 1999).

According to these features, Sorrows & Hirtle (1999) identify three types of landmarks. A *visual landmark* is an entity used as a landmark primarily because of contrast with its surrounding, because it has a prominent spatial location, or because its visual characteristics are easily memorizable. A *structural landmark* is defined as having a significant spatial role or location in the structure of space, while a *cognitive landmark* stands out because of its typical or atypical characteristics in the environment, i.e. it stands out because of its unusual or important meaning or prototypicality.

It is important to note that these characteristics do not only apply to single physical (built) entities (like buildings, bridges, or trees), but may also result in larger physical entities (like parks), or configurations of physical entities (like street intersections, cf. Klippel et al., 2005c), functioning as landmarks.

Cognitive Function of Landmarks

There is general agreement in the cognitive science literature that landmarks are eminently important for acquiring and organizing knowledge about our surrounding space. This space is termed *environmental space* by Montello (1993). For example, there is evidence that spatial knowledge is organized hierarchically. In an experiment reported by Hirtle & Jonides (1985), subjects learned a spatial layout of a familiar environment (Ann Arbor, Michigan) for which they then had to recall learned places and give distance judgments between their locations. Hirtle & Jonides identified clusters of spatial knowledge in the subjects' recall data. Subjects consistently overestimated distances between entities across clusters, while they underestimated distances within clusters. Thus, in line with work by Sadalla et al. (1980), who found asymmetries in distance judgments from a landmark—which they call *reference point*—to other places and vice versa, these results show that landmarks structure spatial knowledge in that they function as anchor points for this knowledge (cf. also Couclelis et al., 1987; McNamara, 1992).

In acquisition of knowledge about space, landmarks are learned early on. The model of spatial knowledge acquisition of Siegel & White (1975) sets knowledge about landmarks to be the first kind of spatial knowledge acquired in a new environment. People recognize landmarks as a place they have been before; connecting these places is the next step, termed *route learning* which occurs with repeated exposure to the environment over time. Only in a final step are the different routes integrated into survey knowledge that allows, for example, calculating shortcuts. Montello (1998) questions this framework with regard to the (strict) order of these steps and claims that some kind of metric knowledge, i.e. knowledge about distances, is acquired from the very beginning. But he does not question the important role landmarks play in spatial knowledge organization: "Finally, I do not wish to suggest that discrete environmental features such as landmarks play no role in the organization and use of spatial knowledge; on the contrary, I believe they are important in this way" (p. 148).

But landmarks are not only an important organizing concept for spatial knowledge, they also serve as navigational tool (Golledge, 1999). Landmarks serve to identify decision points, origin and destination of a route, provide verification of route progress, provide orientation cues for homing vectors, and suggest regional differentiating features. Michon & Denis (2001) echo this in pointing out that people use landmarks in route directions to signal crucial (turning) actions, locate other landmarks in relation to the referenced landmark, and provide landmark information to confirm that a wayfinder is still on the right track.

Tom & Denis (2003) found that in city environments, people more often refer to landmarks than street names, and that instructions using landmarks are better understood, and lead to better wayfinding performance, than those referring to street names. Thus, even though street names provide concise information and theoretically resolve problems of reference as they unambiguously label the street to take, they are not often used. This is because they are hard to memorize, and often the signs stating the street names are hard to detect.

The importance of landmarks for human wayfinding and, consequently, the need to integrate them into computational assistance systems has been confirmed by Tracy Ross and coworkers. In a set of usability studies, they show that the inclusion of landmarks in assistance systems significantly improves users' confidence in correctly executing the given instructions and their navigation performance in both car navigation (May et al., 2001) as well as pedestrian wayfinding (Ross et al., 2004).

Finally, landmarks in route directions can be distinguished with respect to their relation to the route. This has consequences for the kind of spatial knowledge involved in producing and comprehending instructions which will be further discussed in Section 2.2.4.

2.1.4 Anchoring Actions in Space

Route directions provide the addressee with information on the actions she has to perform in order to reach her destination. The information provided by the instructor depends, firstly, on the instructor's spatial knowledge, i.e. her conception of the environment, and secondly, on the way the instructor externalizes this knowledge, for example, in language. That is, not only does "language structure space" (Talmy, 1983), but also "space structures language" (Tversky & Lee, 1998). The conceptualization of an environment is strongly influenced by how it is perceived; environmental characteristics determine this perception (Section 2.1.4.1). Communication of the spatial knowledge to be externalized then relies on the chosen reference system and perspective (Sections 2.1.4.2 to 2.1.4.4).

2.1.4.1 Environmental Characteristics

The physical environment plays an important role in wayfinding. The visual and structural characteristics of an environment can make wayfinding easier or more difficult (Montello, 2005). Gärling et al. (1986), building on work by Weisman (1981), provide an analysis of factors that influence wayfinding, especially in built environments. These factors are:

- *Differentiation* is the degree to which different parts of the environment look similar or different. In differentiated environments wayfinding is easier because the parts are more memorable. They create better landmarks (cf. also Lynch, 1960).

- *Visual access* is the degree to which different parts of the environment can be seen from various viewpoints. A better visual access makes orientation easier, as more of the environment can be taken into account when orienting.

- *Complexity of spatial layout* is difficult to characterize formally. However, it can be said that layouts which can be apprehended as a single simple shape ease wayfinding, while environments that are broken up into several different parts are more complex (cf. Montello, 2005). Werner & Long (2003), for example, point out that integration is eased if the perceived spatial structure of each location in an environment is aligned with the global structure.

In built environments, signage plays an important role (Arthur & Passini, 1992). Signage may relieve some of the problems resulting from environments whose structure results in increased complexity (on any level). According to Wener & Kaminoff (1983), signage reduces perceived crowding of an environment, as well as confusion, anger, and discomfort resulting from not being able to identify relevant information in the environment. Ideally, it indicates the correct navigational choice at relevant places (Werner & Long, 2003). Still, while signage is an add-on to environmental structures that cannot fully compensate for design flaws, architectural design of environments and design of signage should be done collaboratively as wayfinding design (cf. Arthur & Passini, 1992; Dogu & Erkip, 2000).

In summary, environmental characteristics determine the conceptualization of an environment and, along with familiarity and access to orientation aids (Gärling et al., 1983), strongly influence the acquisition of spatial knowledge on an environment and, consequently, the knowledge that may be exploited in producing route directions. How this knowledge is addressed in route directions, i.e. how actions are related to environmental space in the instructions depends on the choice of reference systems, which are discussed in the following.

2.1.4.2 What is a Reference System?

When describing a scene, i.e. a spatial configuration of (different) entities, the speaker tries to make sure that the addressee can identify the entities in the scene the speaker refers to. In order to achieve this, she needs to relate the entities to each other or to an entity that is not part of the described scene, usually the addressee or the speaker herself. Depending on the entity chosen as reference object—the *relatum* or *ground*—these relations, the *references*, rely on different coordinate systems, called *reference systems*.

Rock (1992) defines a reference system (or *frame of reference*)[2] as "[...]a unit or organization of units that collectively serve to identify a coordinate system with respect to which certain properties of objects, including the phenomenal self, are gauged" (p. 404). This is reflected in the following, more concrete, definition: "For a fairly standard working definition, an R[eference]F[rame] is taken to be the imposition of some measure of orientation such that an entity's location can be indicated with respect to some landmark object and/or observer" (Pederson, 2003, p. 287).

Talmy (1983) defines *figure*, which is the entity to be described, to be "a moving or conceptually moveable object whose site, path, or orientation is conceived as a variable the particular value of which is the salient issue." *Ground* then is defined as "a reference object (itself having a stationary setting within a reference frame) with respect to which the Figure's site, path, or orientation receives characterization" (p. 232).

Different reference systems can be distinguished with respect to which kind of object is used as relatum and what kind of orientation relation gets established: which properties of reference objects are exploited and what is the reference object's relation to the described scene?

2.1.4.3 Categorization of Reference Systems

Many different categorizations of reference systems have been proposed in the literature (for an overview see Levinson, 1996; Pederson, 2003). In the following, I concentrate on the categorization proposed by Levinson (1996) as it captures the distinctions necessary for the argumentations and modeling to come (see Chapters 3 and 4). This categorization builds on work by Talmy (1983) who provides an in-depth analysis of how a spatial configuration, and the geometry and possible motion of figure and ground determine which kind of reference is produced in language. Levinson argues that the relevant element in reference systems is the inducing coordinate system, not the reference object. According to Levinson, there are three different reference systems that get employed to describe spatial scenes: intrinsic, relative, and absolute.

Intrinsic Reference System An intrinsic reference system is object-centered. The underlying coordinate system is determined by the 'sideness' of an object, i.e. the object has an intrinsic orientation, for example, a distinguished front. A couch or a human being are good examples of objects with intrinsic orientations that have specific sides as front. The intrinsic

[2]The terms *reference system* and *frame of reference* seem to be used synonymously in the literature (cf., e.g., Tenbrink, 2005). Eschenbach (1999), however, distinguishes both terms, defining a reference system as a collection of concrete objects and a frame of reference as a collection of geometric entities modelling the geometric structure of a spatial reference system. I will use the term *reference system* in this work.

features of an object are usually determined by its function, for example, the front of a car is considered as the side that is in front when the car is in forward motion.[3]

More precisely, Levinson defines an intrinsic reference system to be a binary spatial relation R with arguments F (*figure*) and G (*ground*). $R(F, G)$ states that F is on (or near) a line projected from (the center of) G through an anchor point A, which usually corresponds to R—a named part of G. The origin of the coordinate system is in (the center of) G.

Relative Reference System In a relative reference system, a viewpoint is introduced. This viewpoint is different from figure and ground. Therefore, references in this reference system employ a triangulation of three points with coordinates fixed on the viewpoint. "The box is to the left of the column" is such an example, with the viewpoint being either fixed on the speaker or on the addressee—which can only be determined from context. It is also possible to refer to an object relative to a group of objects of the same class, like in the "the left ball" (meaning the leftmost ball of a group of balls), which also depends on the viewpoint. Moratz & Fischer (2000) term this *group-based reference* (cf. also Tenbrink, 2005).

Thus, a relative reference system is a ternary spatial relation $R(F, G, V)$, with V and G being distinct. The coordinate system's origin is in V and R defines the position of F relative to G as seen from V.

Absolute Reference System Absolute references relate figure and ground relative to some fixed bearings, called *cardinal directions*. Examples for such directions are 'north', 'east', etc. A person always needs to know in which direction these fixed bearings are in order to understand absolute references, for example "the church is in the north of the square."

An absolute reference system is a binary spatial relation $R(F, G)$; the coordinate system is anchored to the fixed bearings and centered on G. Then R states that F is in direction of the fixed bearing R from G.

2.1.4.4 Choice of Reference Systems and Perspective

Carlson (1999) claims that initially all possible reference systems are simultaneously active and, by inhibition of the systems not selected the relevant one gets selected. She concludes this from reaction time results of several experiments, in which subjects are asked to decide whether a spatial relational term (using 'above' as relation) correctly describes a depicted configuration. According to Pederson (2003), social conditions, like register, gender, expert vs. novice, or task at hand, determine which reference system gets chosen.

[3] According to Levinson (1996), in some languages intrinsic features are rather selected by an object's shape than its function. Such differences are not further considered in this work as cross-linguistic issues are not addressed.

He also proposes three interacting scales for the selection of a reference system: the relative geometric relation between figure and ground (termed *"topological space" scale* by Peterson), the relative scale between figure and discourse participants (*"functional" scale*), and the degree to which figure and ground are accessible within the discourse (*"perspective" scale*).

Perspective determines the viewpoint of the description. It must be distinguished from the reference system itself, which becomes especially apparent in relative references. In a statement like "the box is left of the column," which is clearly a relative reference as the ground (the column) does not have any intrinsic properties, 'left' can either denote left from the viewpoint of, i.e. relative to the speaker, from the viewpoint of the addressee, or—in principle—from the (imagined) viewpoint of another object, like in "as seen from the desk." Thus the perspective, also called the *origin*, can either be egocentric (viewpoint of the speaker) or allocentric (viewpoint of someone or something else) (Levinson, 1996).

Adopting an allocentric perspective is cognitively more demanding than using an egocentric one. This requires mentally shifting the viewpoint from your perspective to a point somewhere else in (the surrounding) space which involves cognitive effort. On the other hand, taking the addressee's perspective eases processing of the references, i.e. lowers the addressee's cognitive load. This is especially done if the speaker assumes lower cognitive or linguistic capabilities in the addressee, for example, when talking to a child (cf. Tenbrink, 2005).[4]

Tversky et al. (1999) elicited in their studies reasons for people to mix perspectives in a single description of a scene. Even though switching perspectives affords cognitive costs, i.e. requires increased processing effort, keeping the same perspective often is at least as costly. When keeping the perspective constant in route descriptions, for example, there is a change of viewpoint in (nearly) every utterance, each time demanding a re-orientation of the addressee. Furthermore, according to Tversky et al. (1999), speakers try to select reference objects that are salient, i.e. easy to identify, and references that are easy to produce and comprehend, which often involves switching perspective to achieve this goal.

2.1.5 Principles of Good Route Directions

From the findings reported so far, several authors derive principles on how to generate 'good' route directions, i.e. what should be taken into account when providing route directions. Habel (1988), for example, lists the following principles:

1. Segmentation occurs at decision points.

[4] The same seems to be true for human-robot interaction; here, human communication partners consistently choose the robot's perspective in relative references (cf. Moratz & Fischer, 2000; Fischer & Moratz, 2001).

2. If there is no instruction provided for a decision point, the convention is to continue straight.
3. Decision points are best described using landmarks.
4. Along long route segments, the wayfinder should be provided with reassurance that she is still on the right track.

The route direction principles provided by Denis (1997) are similar to Habel's, but have been developed independently. The main difference is that Denis places even stronger emphasis on the importance of landmarks for route directions and that there is no explicit principle of reassurance:

1. Route directions should consist of a limited number of statements.
2. Redundancy and over-specification should be avoided.
3. Route directions should contain references to visible, permanent, and relevant landmarks.
4. Determinate descriptions should be preferred over indeterminate.
5. Landmarks and actions should be explicitly associated.

With determinate and indeterminate descriptions, Denis refers to the way a landmark is described relative to another landmark. Statements like "there is a post office by the church" are indeterminate—they just announce the presence of a landmark, but not its location. "When standing in front of the main entrance, to the right of the church there is a post office," on the other hand, is a determinate description. It explicitly describes the location of the second landmark relative to the first.

Lovelace et al. (1999) summarize the principles identified by Denis, as well as those by others (e.g., Couclelis, 1996; Allen, 1997, presented in Section 2.1.2), and present their view on route directions with the following principles:

1. The wayfinder should be primed for upcoming decision points.
2. Landmarks at decision points should be mentioned.
3. Route directions should include "you're gone too far, if" statements that indicate that a decision point has been missed.
4. An instructor should use landmarks, not street names in the route directions (cf. also Tom & Denis, 2003).
5. Distances between decision points should be given.
6. Route directions should indicate which direction to take at a decision point.

7. They should also contain information for error recovery, i.e. information that may get a wayfinder back on track after taking a wrong decision.

8. The information should be presented in linear fashion, i.e. emphasizing the sequence of actions by, for example, using “then”.

9. Route directions should only provide a limited amount of redundant information.

All these principles, i.e. the quality of route directions, can be measured on different levels (Lovelace et al., 1999). They can be analyzed quantitatively, for example, by counting the number of landmarks referenced or the number of turns given, or they may be qualitatively evaluated in rating tests, or their performance may be tested in wayfinding studies.

2.1.6 Discussion

Route directions are an external representation of which actions to be performed, and what is encountered, while following a route. Producing route directions relies on an internal representation of the instructor (her spatial knowledge); a major step in producing route directions is the selection of the information relevant for the task. The mental representation of this information needs to be externalized to be communicated to the wayfinder. This external representation needs to account for several principles that ease comprehending route directions: (1) the given instructions must be sequential, i.e. in the order of the route; (2) the representation should focus on decision points as the crucial places along a route; (3) it should avoid redundancy; (4) it should use a reference system that eases interpretation of the instructions for the wayfinder; (5) and landmarks should be used to anchor actions in space. The adressee conceptualizes the spatial situations described in the route directions and constructs her own mental representation. Ideally, route directions support the wayfinder’s conceptualization process. Therefore, an automatic generation of route directions should adhere to the listed principles as well; the principles are a prerequisite for producing cognitively ergonomic route directions.

This holds especially for landmarks—next to decision points, landmarks are the crucial elements in route directions. They are fundamental in forming a mental representation of an environment, and they support sequencing the spatial knowledge of the route for communication. Landmarks are well suited to identify relevant places along a route, for example decision points; people rely on landmarks in their wayfinding and in providing assistance for wayfinding.

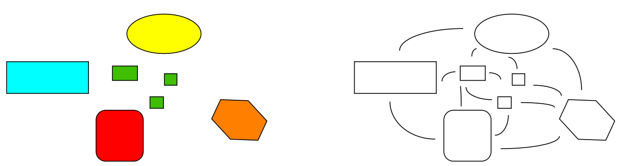
2.2 A Representation-Theoretic View on Route Directions

Selecting the relevant information for providing route directions results in a representation of the route and actions to be described and, depending on the selection process, may be of varying complexity. I start with a discussion of the fundamentals of representation and its relation to complexity (Section 2.2.1), and schematization as a means to reduce this complexity (Section 2.2.2). I then detail work on representing actions (Section 2.2.3) and landmarks (Section 2.2.4) for use in wayfinding assistance systems.

2.2.1 Representation Theory

A route direction is a representation of that part of an environment, and the actions to be performed there, that is essential for reaching a destination. The communicated instructions are based on the spatial knowledge of the instructor—her internal representation. But what is a representation? Palmer (1978) discusses this question in depth.

The core of any *representation* is the distinction between the *represented world* and the *representing world*. Each world consists of *objects* and *relations*. Relations can either hold between objects or between parts of a single object. The objects, and especially the relations, define the structure of a world. Hence, disregarding effects of organization, the number of relations determines the complexity of the representing world; the more relations, the more complex the world (see Fig. 2.2). The representing world can be used instead of the represented world, i.e. inferences about the represented world can be drawn in the representing world (Palmer, 1978). This can be advantageous. The representing world is less complex than the represented world because the representing world abstracts from many aspects of the represented world and concentrates just on certain aspects (see Section 2.2.2).



(a) (b) (c)

Figure 2.2: Example of a represented world (a) and two different representing worlds (b and c). World (c) abstracts more from world (a) than world (b); hence, (c) is less complex.

Other important elements of representation that need to be considered are *processes*. Processes determine the relations that hold in a representation. Or, to state it with Palmer (1978): "It is axiomatic within an information-processing framework that one cannot discuss representation without considering processes. The role of processing operations in the present analysis is that they functionally determine the relations that hold among the objects" (p. 265). Thus, relations in a representation are *operational relations* rather than *apparent relations*, i.e. relations that are defined by the processes that interpret the representation.

2.2.1.1 Representation, Complexity, and Difficulty

A distinction has to be made between *complexity* and *difficulty*. Both complexity and difficulty are used to describe the simplicity or ease of something. Complexity is used when referring to structure; difficulty is used when referring to performance.[5] For example, while the instruction "go up the hill and down again" for the route presented on page 3 is not complex, following this route, i.e. walking up the hill and down again, may be difficult depending on the steepness of the hill and the condition of the footpaths.[6]

Processes define the interpretation of a representation as well as determe the difficulty in understanding a representation. Thus, the *complexity of the processes* is an important factor when judging the *complexity of a structure*. A seemingly complex structure can be easy to interpret if the processes working on this structure are simple and, vice versa, there can be a seemingly simple structure that is hard to understand as the processes working on it are complex. Naturally, this also holds for spatial situations, for example, intersections in a street network and their representation in instructing which action to be performed at an intersection. That is, in wayfinding there is a represented world, i.e. the real world, and a representing world, for example, a map, a verbal route direction, or a mental conceptualization. We can distinguish different kinds of complexity and difficulty in wayfinding:

- *Navigational difficulty*
  This is the difficulty of actually following route directions. It comprises problems of identifying decision points, remembering the right decisions, and then matching these remembered decisions with the situations at hand, amongst others.

[5]Merrian-Webster defines the terms as follows: *complex*: hard to separate, analyze, or solve; having many parts or aspects that are usually interrelated. *difficult*: hard to do, make, or carry out; hard to deal with, manage, or overcome (Merriam-Webster Online; www.m-w.com; 17.4.2006).

[6]The distinction between complexity and difficulty has also been discussed at the International Spatial Cognition Summer School 2003 in Bad Zwischenahn (personal communication by Dan Montello, Christian Freksa, Markus Knauff, Urs-Jakob Rüetschi, Jan Malte Wiener, and myself).

- *Descriptional difficulty*
  This is the difficulty of producing good, correct, comprehensible route directions for a given route. The production aims at directions that are not complex. This difficulty must be distinguished from the diffculty of comprehending the route directions.

- *Visual difficulty*
  This is the difficulty of extracting information from a visual representation (like a map). It depends on the complexity of the visual representation, i.e. on the amount of visual clutter (cf. Phillips, 1979) and its apparent meaningfulness (cf. Werner & Long, 2003; Klippel, 2003b).

- *Conceptual difficulty*
  This comprises problems when forming a mental representation of a route, i.e. the difficulty of the conceptualization process and the complexity of the resulting mental representation (its structure and the number of its relations). The resulting conceptualization needs to be compact enough to be remembered, detailed enough to suffice for following the correct route, and flexible enough to cope with problems in the actual wayfinding (see navigational difficulty above).

2.2.2 Schematization

The real world is complex; agents acting in it have to deal with an excess of information in different modalities (like visual input or sound). It is essential to reduce this information overload to those bits relevant for the task at hand in order to be able to act sensibly; this is the main reason to bother with generating representations in the first place. All our representations of the world, be they mental or external, abstract from aspects of the real world. This abstraction may be done to different degrees.

In cognitive science, such abstraction is referred to as *schematization*. Talmy (1983) defines it as "schematization—a process that involves the systematic selection of certain aspects of a referent scene to represent the whole, while disregarding the remaining aspects" (p. 225). Herskovits (1998) identifies three distinguishable processes in schematization: abstraction, idealization, and selection. Generally, schematization refers to a reduction of information content. But unlike *generalization*, which is used in cartography to refer to simplification of information due to technical and perceptual constraints, schematization aims at cognitive adequacy and, therefore, reduces information intentionally beyond technical needs (cf. Klippel et al., 2005b). This is not only true for language, but also for graphical representations, for instance maps (e.g. Tversky & Lee, 1998, 1999, further discussed in Section 2.3). In the Aspect Maps project (Barkowsky & Freksa, 1997; Berendt et al., 1998), a computational approach to schematizing geographic information has been developed. The constructed maps represent the specific knowledge for a given task; they focus on the relevant *aspects*. This knowledge is extracted from existing knowledge prior to map construction. Accordingly, these maps are *task-specific maps* (cf. Freksa, 1999). Schematic maps as cognitively adequate representations are discussed in further detail in Klippel et al. (2005b).

Producing good route directions can be considered a schematization process as well. Selection of relevant spatial knowledge, concentration on relevant parts (the decision points), and specifying actions with just the necessary precision and level of detail involve selection, abstraction, and idealization processes (see Section 2.1.1).

2.2.3 Representing Actions

People represent spatial knowledge, such as distances and directions, qualitatively. They encode perceived metric knowledge into a discrete set of categories which represent crucial differences between possible values of the metric at hand (Freksa, 1991). These categories depend on context (Freksa, 1991), for example the task, and the spatial scale they are perceived in (Montello, 1993). This holds especially for spatial relations between entities. Even more, these relations often are biased. For example, in estimation of spatial location, Huttenlocher et al. (1991) found that people shift their estimate towards prototypes representing a direction category when they recall previously perceived positions of dots. Especially, people remember turns to be closer to 90 degrees than they actually are (Sadalla & Montello, 1989, reported in Montello & Frank, 1996). Also, people estimate distances of reference points to other points in a space differently than distances of those points to the reference points (Sadalla et al., 1980; McNamara, 1992).

Since distances and directions are represented as qualitative categorical knowledge, people apply these categories also in route directions. Most important for my work is directional information that allows locating entities in space. Directional relations are used in several respects in route directions: they state the location of entities encountered along the route (like landmarks) with respect to the wayfinder or other entities; they announce a change of heading at decision points, i.e. represent turning actions; and they may relate these actions to an entity's location to better anchor them in space. Executing an action often results in changes in these relations. Depending on the task and the spatial situation, the granularity of the categorical knowledge may vary. The level of granularity needs to be chosen such that it allows for the disambiguation of the directions relevant for the task. On the other hand, the chosen categories need to result in directions that are meaningful and distinguishable for a human.

In research on qualitative spatial reasoning, several qualitative direction models have been proposed. These models divide the two-dimensional space into (labeled) regions. These *sectors* map all possible angular bearings by a usually small, discrete set of categories. Just as with human references, directions may either be represented in an absolute reference frame or a relative one. Typically, in absolute references cardinal directions are used; these may be represented by a homogeneous division of space as, for example, proposed by Frank (1992). Here, pie-slice-shaped sectors represent directions in different levels of granularity (see Fig. 2.3).

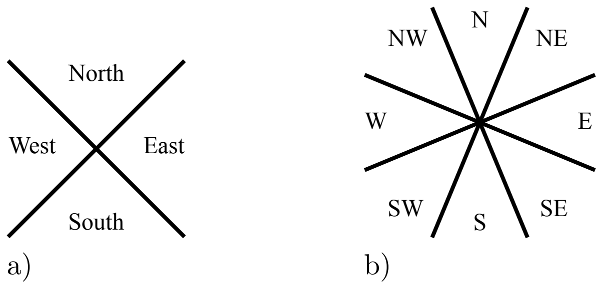


Figure 2.3: Cardinal direction models dividing the plane in a) 4 and b) 8 half-planes (*sectors*) (cf. Frank, 1992).

Cardinal directions are fixed by compass bearings—they are a man-made division of directions on earth. Representing them in a homogeneous direction model directly reflects this division. However, relative directions, i.e. those used in intrinsic or relative references stem from human conceptualization of space. The question arises whether such relative directions are also adequately represented by homogeneous direction models as they are often used in qualitative reasoning approaches?

Montello & Frank (1996) address this question. They tested direction estimation data collected in human subject tests (from Sadalla & Montello, 1989) against homogeneous four- and eight-sector models in a computational simulation. Neither of these models adequately reproduced the empirical results. Montello and Frank then tested heterogeneous direction models with small pie slices representing orthogonal turns and bigger ones representing oblique turns. These heterogeneous models cover the empirical data well, suggesting that relative directions are not homogeneously conceptualized by humans.

Klippel et al. (2004) empirically elicited such a heterogeneous direction model for turning actions in wayfinding. In a grouping task experiment, subjects were asked to sort icons into as many groups as they felt appropriate. Subjects were told to imagine these icons representing turning actions at intersections. A hierarchical cluster analysis of the collected data reveals the heterogeneous direction model shown in Figure 2.4. Orthogonal turns, i.e. *left*, *right*, and *straight* are represented by categories subsuming a small number of angular bearings—in fact, *straight* is represented by an axis. Oblique turns are represented by large sectors. There is a clear division between front- and back-plane at 90 degrees. This direction model is taken as basis for determining relative directions in context-specific route directions.

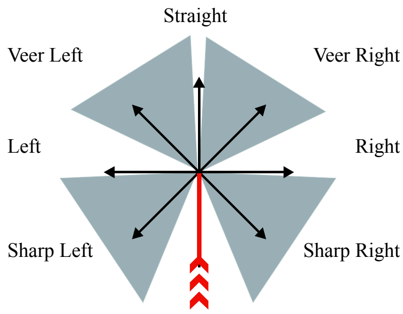


Figure 2.4: Heterogeneous egocentric direction model taken from Klippel et al. (2004) (slightly revised). Arrows indicate the turning prototypes; the *back sector* is not covered in this model.

These direction models represent spatial relations between two point objects, for example, a decision point and a landmark, or direction changes at decision points. However, route directions may also contain linear references to describe actions to be performed, like in "go past the building" or "follow the river." Kray & Blocher (1999) term such relations *path relations*. In a series of experiments Kray et al. (2001) identified factors that determine the use of path relations, especially "entlang" ("along") and "vorbei" ("past"). Subjects were asked to produce prototypical trajectories for these relations. This has been done in two conditions: with a regular and an irregular shaped building as reference object. In a second set of experiments, subjects were asked to use either of the two relations to describe a presented trajectory. These experiments showed that the application of path relations depends on orientation and distance relations between the involved objects. In the case of "along" and "past", the crucial relation is parallelism between a trajectory and the reference object for applicability of "along"; a secondary criterion is closeness between the objects. These criteria are assessed qualitatively; absolute parallelism, for example, is not required. Thus, such relations can also be modeled using qualitative information.

An entity may be located using more than one other entity as reference object, or more than one entity may be located in a reference. In these cases, the entities can be considered to form a spatial configuration which may be represented by ordering information. Ordering information derives from the linear, planar, or spatial ordering of features (Schlieder, 1995). Ordering is closely linked to orientation. A route, being a linear and directed (oriented) entity, induces an order on the entities along that route. That is, the orientation of the route determines in which spatial and temporal order these entities are encountered (Fig. 2.5a). Furthermore, the configuration of entities at an intersection, i.e. its branches or landmarks located there, can be described using circular ordering information (Fig. 2.5b).

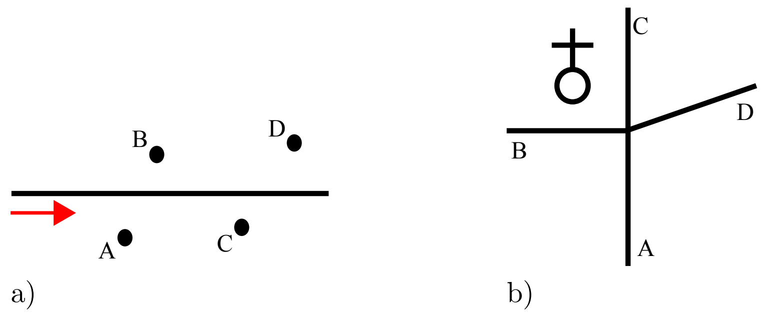


Figure 2.5: a) Linear order of entities induced by the orientation of a route. The route is directed from left to right; the resulting order is $A < B < C < D$. b) Circular order of entities at an intersection: starting with A, the order is $A < D < C < Church < B < A$.

Qualitative ordering and direction information is the basic spatial knowledge exploited in generating context-specific route directions.

2.2.4 Representing Landmarks

Landmarks are essential for human spatial knowledge organization and a crucial part of route directions as has been detailed in Section 2.1.3. Landmarks are entities in an environment that are easily recognized and memorized. This broad definition leaves many options for entities functioning as landmarks, even though most authors seem to refer to single built objects, especially buildings (or façades), when talking about landmarks. Also, landmarks may take different functions in route directions, be it the same (type of) entity in different configurations, or different (types of) entities. For properly representing and identifying landmarks in an automatic generation process of route directions, a more elaborate discussion of landmark types is called for (cf. also Hansen et al., 2006).

Different types of landmarks may be distinguished with respect to a landmark's geometry, its position with respect to route and environment, or a combination of these. A first, broad distinction is provided by Sorrows & Hirtle (1999) who identify two types of landmarks: *distant* (or global) landmarks outside the current environment and *local* landmarks. The former allow coarse orientation in an absolute reference system induced by that landmark, while the latter are part of the environment and may, thus, take any function of those detailed in Section 2.1.3.

Krieg-Brückner et al. (1998), taking the perspective of robot navigation, further differentiate local landmarks with respect to how they may be used for localization in an environment. With *landmark*, they refer to an entity that is used to determine an agent's position in enclosed or open space. These are used independent from a route; at least three of them are needed to properly localize the agent using triangulation. *Routemarks* are located along a route; as routes are one-dimensional, a single routemark suffices to determine an agent's position.

Lovelace et al. (1999) distinguish *choice point* landmarks and *non-choice point* landmarks (choice point is their term for decision point). Non-choice point landmarks are further broken down into *on-route* landmarks, that are along the route between decision points, and *off-route* landmarks. These may either be in some distance to the route but part of the surrounding environment, for example a church spire, or global landmarks in the sense of Sorrows & Hirtle (1999), for example a mountain.

A landmark's location relative to the route determines its possible function in route directions. Which characteristics result in an entity being easily recognizable and memorable is the next question that needs to be answered. Based on Lynch's (1960) discussion of landmarks, Sorrows & Hirtle (1999) identify three aspects that constitute a landmark: *visual*, *structural*, and *cognitive* (see Section 2.1.3). Much earlier, Appleyard (1969) identified similar characteristics from an analysis of collected descriptions and maps of a city environment.

Picking up this classification, Raubal & Winter (2002) developed a measure for determining a landmark's saliency. This measure combines the characteristics above and calculates a real-valued saliency value which allows comparing different landmarks with respect to their saliency. This measure is defined as follows:

$$s_{vis} * w_{vis} + s_{sem} * w_{sem} + s_{str} * w_{str}$$

Here, s stands for salience measure and w for a weighting factor; the indices vis, sem, and str denote *visual*, *semantic*, and *structural* saliency. That is, the salience measure is the sum of the weighted single characteristics defining a landmark. The measure has been extended to respect for *advance visibility*, i.e. to rate those entities at a decision point higher that are visible early on (Winter, 2003), and to account for the structure of the underlying street network, i.e. selecting landmarks based on their position along the route—preferring those at decision points (Klippel & Winter, 2005). Determining appropriate weights for the single salience measures requires empirical testing.[7] Winter et al. (2005) set weights for different criteria (like shape or color) for building façades for both day- and night-time. These stem from

[7]Consequently, in the initial model by Raubal & Winter, w_{vis}, w_{sem}, and w_{str} are all set to 1.

a statistical regression analysis of data collected in an online survey where subjects rated different façades, shown in a panorama photograph taken by night and by day. Subjects had to decide how well the façade in question stands out from all other façades shown on the same photo.

Gapp (1995) also uses weighted real-valued factors. His approach aims at identifying the optimal reference objects for object localization in a configuration of objects. Referentiality (object visibility and occlusions), mobility, distance to localized object, visual saliency, whether the object has been mentioned before, functional dependencies, and prior knowledge of speaker and hearer are the factors used; these are partly interdependent.

In a different approach, Elias aims at extracting objects suitable as landmarks from spatial databases. Initially, objects of specific types, for instance swimming pools or churches, have been extracted and used to enhance route maps by visualizing them as landmark objects (Elias & Sester, 2003). Since this does not guarantee that these objects really are salient at their location—for example, because there are several swimming pools nearby—in Elias (2003), methods from data mining are employed to identify those objects in a surrounding that are salient. She employs ID3, a supervised learning algorithm that constructs a decision tree, and Cobweb, an unsupervised learning algorithm that performs hierarchical clustering for landmark extraction. In this approach, saliency is defined as either objects with a short description tree (ID3), or objects that are singled out early on in clustering (Cobweb).

Since specialized spatial databases are expensive to build and need constant updating, a case study by Tomko (2004) explores augmenting automatic route directions with landmark information extracted from the World Wide Web. His small user study shows that it is principally possible to implement a web service that queries location-based search engines based on, for example, street names found in the route directions, and adds its weighted search-results as additional landmark information. But while availability and reliability of such information is usually good, searches also result in a lot of spam making it impossible to process all searched data. Furthermore, the search queries' identification of the correct location can be problematic, as features are usually encoded as point-like based on post-codes with no relation to the surroundings, and are not specifically provided for wayfinding purposes.

In summary, common to all approaches trying to automatically identify suitable landmarks from a set of candidates is that they require a rich and reliable data set offering information in different dimensions. This is problematic since such data is hard to get and maintain, the identification process may involve complex, lengthy calculations, and correctly setting the parameters requires much and difficult empirical testing (cf. Gapp, 1995; Raubal & Winter, 2002). Therefore, Klippel et al. (2005c) propose augmenting route directions with references to salient configurations of an environment, such as T-intersections or roundabouts. These can be easily determined based on the underlying graph representation of the street network's geometry, which is already available as it is needed for path-search algorithms.

2.2.5 Discussion

Route directions represent real world situations; they communicate how to get from an origin to a destination in the real world. To be of good use, route directions should reduce navigational and conceptual difficulty; the communicated information should be schematized such that the directions concentrate on the relevant information for the task—the turning actions at decision points. Information about directions and distances should be communicated qualitatively, reflecting categories employed by humans to represent this information. This holds especially for describing direction changes at decision points. Landmarks can be distinguished with respect to their relation to the route at hand, for example, differentiating landmarks located at the route and landmarks distant to the route. Landmarks are mostly assumed to be conceptualizable as a point, i.e. they are abstracted to a point in their representation. However, it is possible to refer to other features of an environment in route directions, for instance linear entities or salient intersections. This is a significant addition to using landmarks in the cognitively ergonomic generation of route directions.

2.3 Computational Approaches to Generating Route Directions

Route directions are prominent in cognitive research, and wayfinding assistance is an interesting market for applications (e.g., today's widespread use of internet route planners and car navigation systems). This is reflected in research on computational approaches to route directions. Several approaches exist that deal with the automatic generation of route directions, or at least parts of the generation process. In this section, I will present those approaches that are relevant to my work.

2.3.1 Cognitively Adequate Route Directions

Amongst the approaches presented in this section, two recent ones are especially relevant for my work: Klippel's theory of wayfinding choremes and the work by Dale and coworkers.

2.3.1.1 Theory of Wayfinding Choremes

In his work on wayfinding choremes, Klippel aims at identifying conceptual representations of turning actions at decision points that underlie pictorial and verbal route directions. Klippel (2003b) defines *wayfinding choremes* as “mental conceptualizations of primitive functional wayfinding and route direction elements” (p. 1). His work is based on the distinction between structure and function explained in Section 1.1—wayfinding choremes represent procedural knowledge, focusing on functional aspects of wayfinding.

To elicit these conceptual primitives, Klippel had subjects sketch spatial situations in street networks. Subjects received a verbal description, like “four-way intersection” or “turn left at the three-way intersection” and had to draw a sketch map depicting this situation (see Klippel, 2003a). Figure 2.6 shows prototypical turning concepts drawn by subjects. The small icons are a graphical representation of the wayfinding choremes identified from these drawings. Later on in further experiments, these have been refined to the direction model shown in Figure 2.4 on page 34.

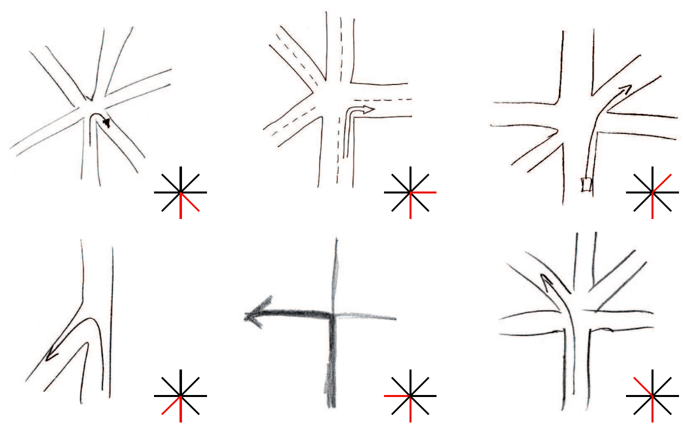


Figure 2.6: Drawings and icons of prototypical turning concepts; from Klippel (2003a).

Wayfinding choremes allow specifying information on the turning actions to be performed in a modality-independent, abstract route-string. Seven primitives represent the basic actions: wc_s, wc_l, wc_r, wc_{hl}, wc_{hr}, wc_{sl}, wc_{sr}. The first three represent *straight*, *left*, and *right*; termed *standard turning concepts* by Klippel (2003b). The last four are *modified turning concepts*, making left or right turns more precise. These primitives may be further annotated with landmark information or the type of intersection corresponding to the decision point (e.g., T-intersection). Each such primitive represents a turning action at a single decision point. A grammar, the *wayfinding choreme route grammar* (WCRG) along with term-rewriting rules, allows combining actions for single decision points into so called *higher order route direction elements* (HORDE) (Klippel et al., 2005e):

$$wc_s wc_s wc_r wc_s wc_s wc_s wc_s wc_l^T = dwc^{3r} dwc^{Tl}$$

The combined terms may then, for example, be externalized verbally, like in "turn right at the third intersection. Turn left at the T-intersection." The combination rules are based on work by Klippel et al. (2003) which is further detailed in Section 3.1.2.

Klippel's work is not strictly computational, but it is meant to be applicable in computational systems. The WCRG, for example, might easily be implemented as a process running before presenting route information to a user, either verbally or graphically, to increase cognitive adequacy of the directions. Graphic externalization of wayfinding choremes is presented in Klippel et al. (2005d): here choremes, as prototypical turns, are used in schematic wayfinding maps to emphasize which action to take at a decision point.

The approach taken by Klippel is one starting point for my work. I pick up the aim of representing route following actions such that they match human conceptualization, but extend the possibilities of these descriptions by, for example, taking into account a variety of landmark types and environmental features, which can be used as reference objects and result in different conceptualizations of the actions to be performed. Also, while Klippel's focus is on the empirical determination of the conceptualizations, I focus on a computational realization of generating the conceptual representations.

2.3.1.2 CORAL

Dale and coworkers developed a system, called *CORAL*, for generating natural language route descriptions based on commercially available GIS data. Their aim is to produce more natural route directions (Dale et al., 2002, 2003). To that end, they try to bridge three main differences between human and automated route directions. First, humans often omit steps that are considered unimportant or obvious—something automated systems cannot do. Second, these systems lack the inclusion of landmarks. And third, humans produce complex clause structures combining several pieces of related information, while the systems produce one-sentence-per-step directions (Dale et al., 2003).

CORAL incorporates methods and techniques from natural language generation. It uses discourse structure to facilitate understanding of the route structure, it combines information into multiclausal sentences by using aggregation techniques, and it generates referring expressions to produce user-oriented descriptions (Dale et al., 2005). The generation process consists of three steps: text planning, micro planning, and linguistic realization (see Fig. 2.7). It is based on a graph representation of the environment's path-network which is extracted from GIS data and in which street segments that belong to the same street are already joined to a single path (called *arc aggregation*; see Dale et al., 2005). Underlying the system's route representation are three types of elements called *messages*: *points* state a user's position, usually by reference to a landmark; *directions* correspond to turns at decision points, while *paths* describe continuous movement along parts of the road network. The text planning step generates an alternating sequence of point, direction, and path messages. Micro planning is then concerned with how to cluster these messages into clause-sized units and how to refer to each element.

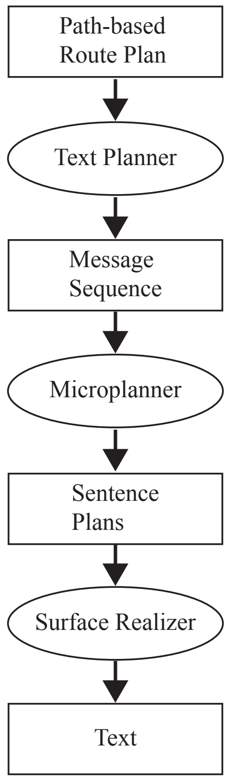


Figure 2.7: Architecture of the CORAL system; from Dale et al. (2002) (modified). The ellipses represent the single processing steps; the boxes the input and output representation of each step. The arrows indicate the processing order.

Route directions in CORAL are hierarchical; each segment of a route is summarized, but may be viewed in detail if needed. According to Dale et al. (2005), this increases recognizability and memorability of the route directions. Segmentation is further explained in Section 3.1.2; segmentation groups elements of the route that are related. *Aggregation* processes these segments and builds sentences which communicate several pieces of information at once. To this end, two strategies are employed: folding a point description into a description of a path ("continue on until you get to the next set of lights"), and combining a turn direction with a specification of the location where this instruction is to be executed ("...at the end of the street turn right.").[8]

For generating referring expressions, the principles of *sensitivity* (taking into account what the addressee can know), *adequacy* (unambiguous identification of referent), and *efficiency* (no more information than necessary) should be accounted for (Dale et al., 2005). Here, Dale and coworkers follow the view of Dale & Reiter (1995). They use a general purpose algorithm, i.e. one that is applicable in several domains, which accesses domain-specific ranked preferences to add enough content to a description to identify the referent. For route directions, these preferences range from using a landmark at an intersection (most preferred), to a reference to the name of the intersecting street (least preferred).

The route directions are represented in an XML-based specification called *Route Planning Markup Language* (RPML). This specification is device-independent and supports annotation for multi-modal content, for example links to graphical representations. It allows rendering route directions as web-pages or on mobile devices (Geldof & Dale, 2002).

While the organization of information by segmenting the route is similar to the organization of route information in my work, the main focus of Dale and coworkers is in applying general natural language generation principles and methods to verbal route directions in order to demonstrate their applicability. Restricting the approach to existing GIS data also limits the range of possible references and, consequently, the instructions' possible adaptations to the current environment.

2.3.2 Focus on Language Generation

Habel (2003) presents an approach to incremental generation of multi-modal route instructions. This work is part of a research aim for developing a component for utterance planning of a natural language generation system, i.e. a *what-to-say* component. The focus is on incremental generation of these utterances; accordingly, the component is called *Incremental Conceptualizer* (INC). This system allows for the generation of conceptual representations of motion events as preverbal messages that may then be verbalized (Guhe et al., 2003). INC consists of four main processes: the *construction* process builds up internal conceptual representations of the current state of affairs (called *current conceptual representation*); the *selection* process selects situations for verbalization from this current representation which are then ordered by the *linearization* process, and finally, a preverbal message is generated. INC has primarily been developed for producing an on-line description of events (cf. Guhe et al., 2003). Its focus is on language production; according to Habel (2003), it is located between a pre-processing

[8]Examples taken from Dale et al. (2005).

unit that provides basic conceptual representations and a formulator that performs linguistic encoding.

Another approach for dealing with the generation of incremental route directions in natural language is presented by Maaß (1994, 1995). Here, an artificial agent moves through a virtual environment. Its task is to generate instructions on the way to take in natural language, thereby using information extracted from a visual perception system. The agent's knowledge is either represented as mode-dependent, in the case of visual perception and natural language, or as mode-independent, in the case of conceptual and spatial knowledge. From perceived information the system extracts spatial relations between objects in the scene, stores them in a so-called *configuration description*, and then extracts the relevant spatial information to be included in the instruction. Depending on the time it takes to reach the next decision point, the system varies the amount of details in the given instructions: if time is short, it utters a simple 'turn-now' instruction, and with more time, landmarks are included. The focus of this work is on the (inter-)relationship between the representational levels, i.e. between visual, abstract spatial, and linguistic representations.

Marciniak and Strube focus on the process of mapping between the semantic content of route directions and their grammatical form. They develop an ontology that captures the semantic content of route directions which they claim to capture conceptualizations of the interactions that take place between route follower and features of the environment (Marciniak & Strube, 2005b). Based on this ontology, they annotate a corpus of route directions that is applied to train and test a natural language generation system. This system is realized as a modular system of classification tasks; each such task is solved using integer linear programming (Marciniak & Strube, 2004, 2005a). Thus, they are mostly concerned with *how* to say something in route directions rather than *what*, which is the focus of my work.

2.3.3 Between Language and Maps

As stated in Section 2.1.1, Tversky and Lee assume a common conceptual structure underlying the knowledge represented in verbal and graphical route directions. In a series of experiments, they found that subjects use the same elements in both verbal route descriptions and graphical route depictions (Tversky & Lee, 1998). From the route directions collected in these experiments, they generate a graphical and a verbal toolkit (shown in Figure 2.8). Each toolkit is sufficient to produce route directions. They tested this hypothesis in another set of experiments where subjects used the toolkits to produce graphical and verbal route directions (Tversky & Lee, 1999). It turned out that the toolkits are largely sufficient to produce route directions, but subjects use additional elements, for example arrows, to enhance the route directions.

Tversky and Lee conclude that an automatic translation between maps and verbal instructions may be possible. Klippel (2003a), however, points out that such translation is problematic since the graphical toolkit focuses on structural aspects of an environment (like configurations of intersections), while the verbal toolkit focuses on functional aspects, i.e. the specification of actions.

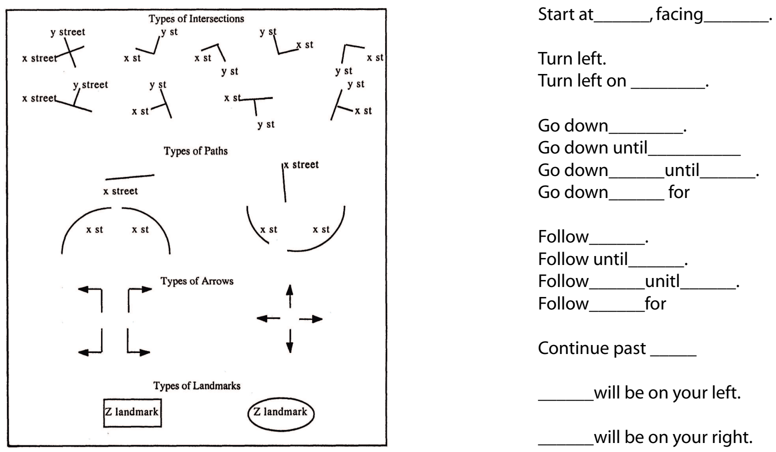


Figure 2.8: Graphical and verbal toolkit of Tversky & Lee (1999) (modified).

The work by Tversky and Lee is, strictly speaking, not computational. But Agrawala & Stolte (2000, 2001) base their work on what they call *effective route maps* on these toolkits. These maps schematize routes through street networks in a way similar to that proposed in the graphical toolkit. Additionally, they focus on those areas where route following actions take place by using different scale levels, i.e. they deemphasize long straight segments and emphasize intersections where turns occur.

An automatic translation from verbal to graphical route presentation is pursued by Fraczak (1995, 1998). The verbal route directions are split into global and local units on a conceptual level: global units reflect the general structure of a route consisting of *sequences* (route-segments) and *connections* (decision points); local units further specify actions, objects, and spatial relations between objects or a wayfinder and objects. This division into units is based on an analysis of a route direction corpus. A sketch map is then generated using a set of graphical primitives. It shows the route as a sequence of arrows reflecting the turns and markers representing landmarks, and is annotated with textual labels, for example to give descriptions of the landmarks.

A major problem of Fraczak's approach is that language may leave much spatial information underspecified (for example, the distance between two turns, on which side of a street a landmark is located, etc.). According to Fraczak (1995), some missing information may be derived from linguistic markers used in the verbal route directions, but remaining ambiguities have been solved by making arbitrary decisions (Fraczak, 1998).

2.3.4 Route Search

The approaches presented so far deal with generating instructions to be presented to a wayfinder. They consider processes and methods needed to produce route directions. However, they are not—or only to a limited extent—concerned with selecting the route to describe, i.e. with calculating a route through an environment that will take a wayfinder from origin to destination. Generating easily comprehended route directions may, however, benefit from selecting a route based on its (supposed) cognitive or descriptional complexity.

Duckham & Kulik (2003) present such an approach. They extend standard shortest path search by a heuristics that associates a weight with each pair of connected edges (rather than each edge as in classic shortest path approaches). This weight reflects the complexity of passing the node in-between these two edges—nodes represent intersections of a street network. Weighting is based on work by Mark (1986) who classifies different types of intersections according to the complexity of describing the action to be performed there and assigns so called *slot values* to each type depending on this complexity. Duckham & Kulik, accordingly, term their algorithm *simplest path* algorithm. They vary Mark's assignment slightly, for example, to account for intersections with an arbitrary number of branches—with a higher number of branches resulting in a higher slot value (see Fig. 2.9). In a simulation experiment, they show that their algorithm generally results in paths that are only slightly longer than the shortest path. While the weighting employed accounts for structural differences of intersections, it does not account for functional aspects, for example, (possible) ambiguity in the direction to take at an intersection, nor landmarks or other environmental characteristics that might be exploited to ease conceptualizing the action to be performed at an intersection—with the exception of T-intersections.

Caduff & Timpf (2005) present an approach to base route search on the presence of landmarks. The so called *landmark spider* weaves a net of landmarks along the path-network that are used to calculate the "clearest route." Only point landmarks at decision points are taken into account, and their saliency is assumed to be known. Based on this information, a modified Dijkstra algorithm calculates a route from origin to destination. The weighting of the edges is based on the weighted sum of distance and orientation of a landmark relative to a moving wayfinder and its saliency. Landmarks close to a decision point, which are in front of the wayfinder, are preferred. The approach suffers from a low density of landmarks as,

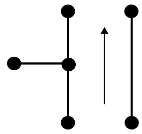
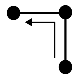
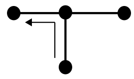
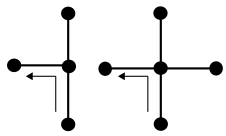
straight on		1 slot
turn (not at intersection)		4 slots
turn left or right at T-intersection		6 slots
turn left or right at intersection		$5 + deg(v)$ slots

Figure 2.9: Weighting (slot values) of different intersection types; from Duckham & Kulik (2003) (modified). $deg(v)$ denotes the degree of an intersection, i.e. the number of branches meeting at this intersection.

in principle, landmark-based routing requires a landmark at each decision point. Taking into account further features, for example the path-network's structure or further types of landmarks, might lessen this problem.

Timpf is further concerned with complexity of routes in multi-modal wayfinding. Multi-modal wayfinding here refers to traveling with a combination of different types of public transport. Specific to this kind of travel are changes from active wayfinding at the stops (termed *transfer nodes* by Timpf, 2002) and passive transportation in buses or trains. The passive transportation part is fairly easy since a wayfinder only needs to identify the station to get off; it is in the active wayfinding part that complexity is of concern (cf. Heye & Timpf, 2003).

Based on an ontological distinction between a wayfinder and the perspective of the public transport system on multi-modal wayfinding (Timpf, 2002), Timpf determines factors influencing complexity in this kind of wayfinding and defines means of computing this complexity for automatic wayfinding assistance. Heye & Timpf (2003) identify such factors with help of a web survey. Besides personal preferences, physical characteristics of the stops have great influence on the perceived complexity. These include the number of streets to cross, the number of lines, changes between different types of transportation—for example bus to tram, the number of possible directions, and signage. The transportation network is modeled hierarchically on different levels of granularity. The first level comprises the transportation lines, with nodes representing stops. On the second level, these nodes are refined with subgraphs representing different platforms at a stop and the footpaths between them (Timpf & Heye, 2002). Timpf and Heye define a measure calculating the complexity of a route. It is the sum of the complexity of each node on the second level of the hierarchy, i.e. it takes into account the actual complexity of changing transport at that place. That is, the complexity of a single node is the sum of the number of stops, number of links, and the complexity of nodes and links on the second level of the hierarchy. With this measure, Timpf & Heye (2002) weight nodes in the public transport network depending on whether, and how, changes of transportation means occur there, and use this weighting to calculate the least complex route from origin to destination.

2.3.5 Discussion

The presented approaches to an automatic generation of route directions concentrate either on in-advance or on incremental route directions. Different approaches focus on different aspects of the generation process. Some focus on the selection of the information to communicate, others focus on how to communicate this information. Also, they mostly focus on a specific modality (e.g., language) in generating instructions. Some of the approaches acknowledge the importance of including landmarks in route directions. However, these approaches mostly restrict themselves to using only point objects as landmarks. While some of the approaches take specific spatial situations into account, none explicitly focuses on the adaption to environmental characteristics.

Chapter 3

Route Directions: An Analysis of Properties and Elements

Automatically generating route directions requires representing the parts of a route and the accompanying actions such that they are addressable by a producing system. While the next chapter elaborates this process, in this chapter I present a systematics of the basic set of elements used in the generation process. Taking into account the research presented in Chapter 2, this systematics is based on properties of route directions that capture how people provide and conceptualize route directions (Section 3.1) and on which kind of spatial information is involved in references to different types of objects (Section 3.2). The properties and different kinds of spatial information have information-theoretic consequences discussed in Section 3.3. The systematics is presented in Section 3.4.

3.1 Properties of Route Directions

In this section I present general observations and principles underlying route directions that form the basic frame of the systematics. In Section 3.1.1, I detail the consequences of referring to landmarks in route directions for producer and wayfinder, Section 3.1.2 introduces chunking as an important mechanism in organizing route directions into sensible parts, and Section 3.1.3 explains which kinds of reference systems and reference objects are applicable in route directions.

3.1.1 Landmarks in Route Directions

As has become apparent in Chapter 2, landmarks are important in wayfinding and the conceptualization of routes. They are also prominent elements in route directions. While their integration in the generation process for context-specific route directions and the process of deciding which environment's elements function as landmarks in generating route directions is explained elsewhere (see Sections 3.4, 4.2, 5.2), in the following I discuss the consequences of referring to landmarks in route directions.

A primary reason for integrating landmarks in route directions is that they help wayfinders to identify relevant places along the route, typically decision points. Actions necessary for route following are linked with these landmarks, i.e. landmarks are used as triggers for actions (e.g., Denis, 1997; Michon & Denis, 2001). Furthermore, landmarks are important structuring features in spatial knowledge (e.g., Siegel & White, 1975; Schweizer et al., 1998). They help producers of route directions to structure their instructions and to correctly recall the order of decision points. Thus, the producer of route directions—the instructor—chooses landmarks according to her spatial knowledge, the (assumed) landmarks' saliency, and their applicability to the route.

This has consequences for the wayfinder. The wayfinder—who has no, or only minimal, knowledge of the current environment—expects to encounter those elements that are mentioned in the route directions in the environment. These elements become landmarks for the wayfinder as she starts looking for them. This is the case even though the elements might not have been remarkable for the wayfinder if they were not mentioned in the directions. By looking for the mentioned elements, a wayfinder's attention is drawn to these elements. Most likely, they then become part of the spatial representation the wayfinder acquires of the environment (cf. principles of (visual) attention and levels of processing in memorization and learning; e.g., Craik & Lockhart, 1972; Anderson, 1999). This does not exclude the fact that she will recognize and acquire additional elements as landmarks—in fact, this is likely to happen. But the landmarks mentioned in the route directions help both producer and wayfinder to organize a route into sensible parts.

3.1.2 Chunking in Route Directions

Route directions provide instructions on how to proceed at every decision point, yet not every decision point and the accompanying action needs to be mentioned explicitly.[1] Often, actions for several decision points may be combined into a single instruction. This combination is an important mechanism in route directions and the conceptualization of routes. There are several terms used in the literature to denote this mechanism. I adopt the term *spatial chunking* as used by Klippel et al. (2003); Dale et al. (2005), for example, refer to it as *segmentation*. Spatial chunking groups several decision point/action pairs into a single segment. Klippel (2003b) terms such

[1] Compare Habel's (1988) principle of going straight if no instruction is given.

segments *Higher Order Route Direction Elements* (HORDE), indicating that several decision point/action pairs, which belong together functionally, are conceptually grouped into a single entity.

Dale et al. (2005) identify two principles underlying this grouping mechanism which they term *landmark-based* and *path-based* segmentation. In landmark-based segmentation, landmarks at decision points delimit a part of the route to be followed. The route is decomposed into segments; each segment leads to a landmark. Path-based segmentation is based on three features of paths: road category (highways, main roads, etc.), path length, and turn saliency (e.g. T-intersections). Routes can be segmented by employing any of these features or a combination thereof.[2]

Klippel et al. (2003) differentiate three kinds of spatial chunking:

- **Numerical chunking**: A sequence of decision points that involve no direction change (DP-) and one decision point with a direction change (DP+) are combined into a single decision. This is done by counting the decision points until a direction change occurs, for example "turn left at the third intersection" (see Fig. 3.1a). Similarly, a sequence of decision points with equal direction changes can be grouped, for example "turn twice right." According to Klippel (2003b), the number of decision points chunkable in this way is limited—usually to no more than three. Otherwise it may result in instructions like "turn left at the 27th intersection" or "turn seven times right," which can hardly be considered typical instructions produced by humans (see also Klippel et al., 2005e).

- **Landmark chunking**: This kind of chunking is similar to numerical chunking. However, an unambiguous landmark identifying the DP+ is utilized to mark the point where a direction change occurs, instead of counting the decision points without direction change. "Turn left at the church" is an example for a chunk resulting from landmark chunking (see Fig. 3.1b). The number of intermediate decision points is not specified in this kind of chunking. Thus, it allows for the combination of an (basically) arbitrary number of decision points into one chunk. However, if this number becomes large, i.e. if there are many decision points passed before the next direction change, a wayfinder should be given some confirmational information reaffirming that she is still on the right track (cf. Habel, 1988).

- **Structure chunking**: Spatial structures that are unique in a given local environment are exploited in structure chunking. For example, the dead end of a T-intersection unequivocally marks the need for a direction change; one either needs to turn left or right, as keeping

[2] Dale et al. (2005) do not specify how exactly these features can be combined into a chunking threshold.

straight on is impossible. This way, it is possible to chunk several DP- and the relevant DP+ located at such a structure into chunks like “turn left at the T-intersection” (Fig. 3.1c). Such structures are comparable to landmarks in their function (cf. Klippel et al., 2005c). An instruction like “follow the river” also rests upon structure chunking as it combines actions for several decision points that are located along the river into a single instruction.

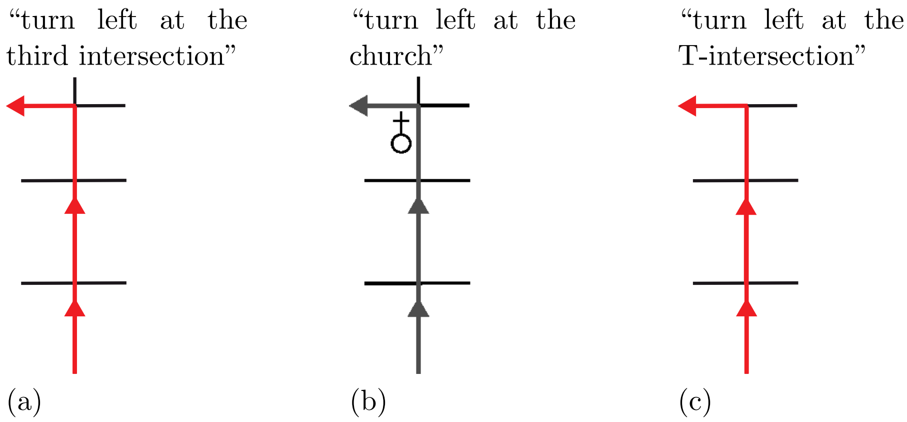


Figure 3.1: Examples of spatial chunking according to Klippel et al. (2003): (a) numerical chunking, (b) landmark chunking, (c) structure chunking.

More than one kind of spatial chunking might be applicable for a sequence of decision points at the same time. Consider the landmark shown in Figure 3.1b being present at the T-intersection in Figure 3.1c. In this case all three kinds of spatial chunking would be applicable to combine the three depicted decision points.[3] Preference rules are needed to decide on which kind of spatial chunking to use in such situations.

Spatial chunking as presented by Klippel et al. (2003) combines instructions denoting necessary turns based on the configuration of decision points’ branches—potentially annotated by landmarks or spatial structures. But it is possible to combine sequences of instructions for several decision point/action pairs that employ other kinds of references as well, references that do not rely on a decision point’s configuration. The instruction “go uphill,” for example, may combine actions for several decision points that happen to be in a line uphill into a single instruction, which can also be considered a HORDE in Klippel’s terms. Usually, either landmark or structure chunking are used to combine these kinds of instructions into single chunks. However, such chunks need additional information that marks their end,

[3]Figure 3.1 depicts a different situation for each kind of spatial chunking to better illustrate the differences.

i.e. that denotes the point until the chunked instruction holds (cf. Section 3.4.3). All of these kinds of chunking locally combine instructions for single decision points into a single instruction. The actions to be performed for each of these decision points is still inferable; there is still a specific route underlying the chunked instructions.

3.1.3 Reference Systems in Route Directions

In the canonical case a wayfinder performs a forward motion in following a route; this motion induces an inherent directionality in the route-segments, as well. Thus, as shown in Fig. 3.2, a route induces an intrinsic reference system (Pederson, 2003; Tenbrink, 2005). Instructions on the actions to take involve intrinsic references, either with the wayfinder or some element of the route as reference object. An exception are references to global features like cardinal directions (see Section 3.4.1); here an absolute reference system is used.


(a) (b)

Figure 3.2: (a) A static situation ("the church north of the street"). (b) A route induces an intrinsic reference system ("the church left of the street").

As a route is elongated and, therefore, is in most cases not perceivable from a single vantage point, route directions involve several viewpoint changes and often also several changes of perspective (see Section 2.1.4.4).

3.1.3.1 Perspective

For in-advance route directions, the route is not perceivable while giving (and receiving) instructions. They do not involve an egocentric perspective in the true sense. The speaker cannot provide spatial references with respect to her current position in the instructions. Therefore, strictly speaking from a linguistic point of view, the perspective is allocentric. Even more, according to Pederson (2003), in instructions like "turn left," 'left' can be interpreted either in relation to an imagined traveler or in relation to the inherent directionality of a route-segment, making the imagined traveler superfluous. The term *imagined traveler* denotes the imagined position and perspective at a specific part of a route that the speaker adopts to give instructions for that part.

For practical purposes, the linguistic differentiation between allocentric perspective, i.e. an imagined traveler, and egocentric perspective, i.e. spatial references from the speaker's true point of view, is not pursued in this thesis. Instead, I assume a sequence of egocentric perspectives. As spatial references stay the same despite whether an egocentric or imagined traveler-based perspective is chosen, the instructions using these references are the same. A sequence of egocentric perspectives does not change the characteristics of route directions, but allows for the reflection of the wayfinder's perspective when following the route, rather than the speaker's perspective in giving directions.

3.1.3.2 Reference Objects

Most environment's elements referred to in route directions can be assumed to be abstracted to points in the conceptualization of their geometry. They are located in a small, restricted part of the environment and are only perceivable for a short part of a route—usually one vista part of the environmental space a wayfinder moves in. These elements may function as landmarks and are used to annotate instructions employing an intrinsic reference system with the imagined traveler as reference object, for example, "turn left at the church." They are usually not used as the reference system's reference object.[4]

However, certain elements of an environment are considered linear, such as a river, or areal, such as railway tracks. These elements may be perceivable for a considerable part of a route. Thus, they may well be used as reference objects in instructions, like in "follow the tracks." In this case they inherit the directionality imposed on the route-segments and, accordingly, provide intrinsic features (see Fig. 3.3). The distinction between point, linear and areal elements is taken up again in Section 3.4.3, which discusses different types of landmarks. Global features may also be used as reference objects, like in "towards the mountains" (see Section 3.4.1).

3.1.3.3 Chunking and Reference Systems

Chunking combines several instructions into a single one. This requires that the instructions use the same reference system and the same reference object. It is, for example, impossible to sensibly chunk instructions provided in an intrinsic reference system with those given in an absolute, but it is also impossible to chunk instructions using the wayfinder as reference object with those using an environment's element, for instance a river. That is, instructions like "north" and "turn right after the post-office," or "turn left" and "follow the river," cannot be combined into a single chunk.

[4] An example for a possible exception is "take the street opposite to the church's front", which relies on the church's intrinsic front. Such instructions are not covered in the approach presented in this thesis.

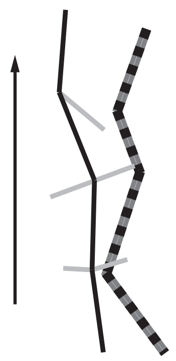


Figure 3.3: A linear landmark, some tracks, next to a route. In this case, the tracks inherit the directionality of the route and, accordingly, its intrinsic features.

3.1.3.4 Route-Maps and Reference Systems

The issues discussed about reference systems in this section apply to propositional route directions. They do not directly map onto route directions presented with maps. Maps are static representations using a survey perspective, which emphasize structural aspects. Spatial relations read off a map rely either on an absolute reference system using cardinal directions—usually assuming that up in the map corresponds to north (e.g., Shepard & Hurwitz, 1984)—or on a relative reference system, i.e. relating one object's position to that of another; for example, "the church is (depicted) right of the street."[5] The latter is especially appropriate when using sketch maps that often do not adhere to the up-is-north principle.

However, as route-maps are used for assistance, i.e. to provide route directions, functional aspects become important. In using a map a wayfinder mentally imposes directional (dynamic) characteristics on the map. In studies by Meilinger and Knauff on comparing route-maps and verbal route directions, for example, participants reported to transform and sequentialize the pictorial information read off the route-maps into a propositional mental representation (Meilinger & Knauff, 2004; Meilinger, 2005). If wayfinders rely on such a representation in wayfinding, the points raised in this section on reference systems apply to route-maps as well. The transformation induces directionality—a route character—on the depicted information. Therefore, it also induces an intrinsic reference system (see Fig. 3.2) which suppresses relative references and leads to the same consequences as explained in Sections 3.1.3.1 and 3.1.3.2 (cf. also Tversky & Lee, 1998, 1999).

[5]Disregarding functional aspects, these relative relations depend on the orientation of the map. Turning the map by 90 degrees counterclockwise, for example, would turn this relation into "the church is (depicted) above the street".

3.1.4 Discussion

Chunking is an important mechanism in giving route directions; it combines several single instructions for consecutive decision points into a single instruction. In combination with references to landmarks, this mechanism is crucial for organizing route knowledge. When referring to landmarks, salient features that are well identifiable should be chosen. However, it is not essential to necessarily pick those features a wayfinder would remember when navigating unaided—the features mentioned in route directions become salient for the wayfinder.

Routes are directed; this directedness is induced on linear and areal features of an environment that are used as reference objects; it is also used when reading off route information from a map. Especially, the directedness allows for asigning features a direction relative to the route and determining whether a feature is left or right of the route.

3.2 Properties of Route Direction Elements

In the previous section, I discussed several properties of route directions and their effects on conceptualization from a cognitive and linguistic point of view. The present section elaborates on different kinds of knowledge needed to determine references in route directions that arise from properties of the different features addressable in the directions. This is in line with research by, for example Freksa & Röhrig (1993), on different dimensions of qualitative knowledge.

First, let us consider the features' dimensionality of space. Routes, being a sequence of decision points connected by route segments, are one-dimensional. They are embedded in a two-dimensional graph structure—the path-network.[6] Furthermore, additional features—that may function as landmarks, for example—are embedded in the path-network. These features may be represented as points (zero-dimensional), lines (one-dimensional), or areas (two-dimensional).

In characterizing the knowledge necessary to address the features dealt with in generating context-specific route directions, a broad distinction can be made between direction information, distance information, and information on the ordering of features (cf. also Tschander et al., 2003). Generally, the knowledge addressed here is qualitative; it reflects the way humans communicate such knowledge in route directions. The distinction of different kinds of knowledge is picked up again in Section 3.4.5 where the elements of the systematics are related to the kind of knowledge references to them afford.

[6]Which, in turn, is abstracted from a three-dimensional world. I keep this abstraction from the third dimension consistent; at most, 2.5D information is used to consider a slant or a feature's visibility (see next Section and Chapter 4).

3.2.1 Direction Information

Even though a route is one-dimensional, due to its embedding in a two-dimensional structure, a wayfinder usually experiences changes of direction—her heading—in route following. In route directions, changes of direction at decision points result in instructions announcing these changes.[7]

Changes of direction can be described either in an intrinsic or an absolute reference system. Intrinsic references denote changes of direction from the perspective of the wayfinder (see Section 3.1.3); the direction changes are described by, for example, relations like 'left' or 'straight.' Such direction information can be formalized in a relative qualitative direction model. Changes of direction can also be stated by using cardinal directions, like 'north' or 'southeast,' which can also be formalized in qualitative direction models.

Finally, some instructions may involve moving towards or away from specific features, for example in references to global landmarks ("go towards the mountain range"). This can be considered to involve coarse (qualitative) one-dimensional direction information, i.e. keeping in direction of the features or the opposite direction, respectively.

3.2.2 Ordering Information

To describe a configuration of features qualitatively, ordering information may be used. Here it is used to describe static configurations of features, i.e. features' configurations as perceived by a wayfinder from a given (unchanging) point of view, and dynamic situations, i.e. the relative location of environmental features to a wayfinder and changes in this relation due to the movement of the wayfinder.

In references to landmarks at a decision point, the location of the landmark is typically given relative to the decision point. Ordering information relating wayfinder, decision point, and landmark is exploited to describe this location qualitatively, for example in "turn left after the post-office." Here, when arriving at the decision point, the wayfinder perceives the post-office—the landmark—to be between herself and the decision point (see also Section 3.4.3).

Specific features referable in route directions require topological knowledge to describe their relation to the wayfinder while she moves. Some such references result in instructions telling the wayfinder to keep a specific relation as, for example, in "follow the river." Such an instruction may be interpreted as trying to maintain the relation to the feature, keeping the river next to oneself, and at the same time to maintain the order between

[7] Changes of heading occurring due to bends and curves in the road that occur in the absence of intersections are usually neglected as they do not involve a decision by the wayfinder on the further way to take.

oneself and the feature, i.e. not changing to the other shore. Other instructions tell a wayfinder to pass across a feature, for example, in "cross the river." Here, the according action involves changing the ordering relation between feature and wayfinder.

3.2.3 Distance Information

The aim of route directions is to provide a wayfinder with information on how to proceed at decision points encountered along the route; they give information on the *direction* to take. Route directions as generated by the process presented in this thesis focus on decisions to take at decision points, according to research on route directions (see Section 2.1). They provide only little information on the way in-between. Hence, distance information is of minor importance.[8]

Still, some references in route directions allow inferring qualitative distance information. It is possible to instruct a wayfinder to move in direction of a feature or away from it (see Section 3.2.1). Such distance information provides only coarse and relative information. The corresponding instructions state only that the distance to the reference object should increase or decrease, respectively. The process does not generate instructions that provide information about distances between features; relative distance is the only distance information provided by context-specific route directions.

3.3 Information-Theoretic Analysis of Route Directions

Route directions are a specific form of communication. They provide information on the route to follow, i.e. they are a representation of a route encoded according to the chosen communication modality; for example a verbal description. Route directions can be analyzed in this light allowing for the consideration of route directions' structure and its consequences on conceptualization, which is presented in the following subsections.

3.3.1 Route Directions: Syntax and Pragmatics

Route directions are a means of assistance that gets communicated. Accordingly, when considering them as messages in an information-theoretic sense (cf. Shannon & Weaver, 1949), some information-theoretic consequences may be identified (Richter et al., 2004). A message—here, a route direction—contains information that gets extracted when the message is interpreted or

[8]This differs from commercial route assistance systems, for instance internet route planners. In these systems, distances, given as quantitative information, play a major role: these systems provide instructions like "stay on Parkallee for 257 meters."

used to make a decision; for example, to decide at an intersection on which further direction to take.

For communication the information needs to be encoded in a specific representational format that is primarily determined by the communication modality at hand. The symbols used in that representation form the syntactic level. This level particularly allows for the measurement of the amount of data needed to store route directions; for example, by counting how many words are used in a verbal route direction, which corresponds to its size. A message's size seems to provide important information: according to research on working memory capacity (e.g., Miller, 1956; Cowan, 2001), it offers clear hints at the difficulty of remembering the message's contents, i.e. keeping all information in memory. However, route directions are meaningful, linked information, i.e. their different parts relate to each other. Therefore, considering relational complexity as argued for by Halford et al. (1998), seems to be more appropriate here. As the following discussions will show, a short message is not necessarily easier to conceptualize.

Taking yet another perspective, the consequences, i.e. the effects a message has, determine the pragmatics of that message. Frank (2003) provides an analysis of route directions' pragmatic information content. According to Frank, the pragmatic information content can be measured against a practical situation in which it is used. With pragmatics, he refers to the consequences a message has on decision making given a specific decision context. He claims that two route directions that differ on the syntactic level may be considered equal from a pragmatic point of view if they both lead agents to take identical routes. In this case, both route directions lead to the same result (the wayfinder being at her destination), and to the same actions (the wayfinder took the same route using either of the two route directions). On the other hand, a single route direction may be different to different users, as users may differ in their knowledge on the environment or the task they try to perform with the given directions.

As a consequence, following Frank's (2003) approach of pragmatic information content, two messages that differ in size, for example in the number of words, can be equal with respect to their pragmatics if they describe the same route, and if used correctly, lead a wayfinder to take the same route when using them as means of wayfinding assistance.

3.3.2 Route Directions: Representation and Conceptualization

Underlying the communication of route directions is a representation of the route, i.e. what a wayfinder is about to encounter and the actions necessary to follow the route, encoded in the respective representation modality. Here, I will not discuss the influence of different modalities on route following, as this is not in the focus of this thesis (but cf. Section 7.2; see also, e.g., Habel, 2003; Meilinger, 2005). Instead, I discuss the consequences differences in the representation of route directions' elements have on conceptualizing the route.

Conceptualization is about mental representation. Which and how elements of the route are represented in the route directions needs to be analyzed with respect to their influence on how this route and the actions to be performed are represented mentally. Two different route directions that are equal from a pragmatic perspective, i.e. result in the wayfinder being at the same location having used the same route, may well differ on the conceptual level (cf. Richter & Klippel, 2005).

In order to use a route direction successfully, a wayfinder needs to conceptualize the route she is to encounter—or at least parts of it. As route directions may differ with respect to the ease they are understood and to the extent which they support cognitive processes, the conceptualization of two route directions for the same route may differ. On the one hand, this difference may reside in the conceptualization itself. For example, the instruction "go straight, straight, and then turn left" results in a different conceptualization than "follow the signs to the train station." On the other hand, the resulting conceptualizations may be similar, but the difference resides in the processing of the route directions that leads to the conceptualization. For example, "go straight, straight, and then turn left" leads to a similar concept as "turn left at the third intersection" (cf. Klippel, 2003b; cf. also Dale et al., 2002, 2003). But the former requires more processing than the latter. This is because in the latter example, directions for several decision points are already chunked into a single direction, while in the former this still has to be done by the wayfinder herself (see Section 3.4.3).

Furthermore, as can be seen from these two examples, information on a route may be represented explicitly or implicitly in route directions which has a twofold consequence. On one hand, representing information implicitly reduces a route direction's size which, in principle, eases remembering the directions. On the other hand, the implicit information needs to be (easily) inferable in order to successfully follow the route as has been discussed in Section 3.4.6.

3.3.3 Discussion

Different kinds of spatial knowledge are involved in route directions. This knowledge is qualitative and direction and ordering information is predominant as these kinds of knowledge allow locating the features addressed in the directions and enable the wayfinder to locate herself along the route.

Route directions may differ in several respects. Two route directions for the same route may differ in their length; they may use of a different number of instructions and a different amount of data to communicate the way to take. They may also differ in their pragmatics, i.e. with respect to their usefulness for a wayfinder. Finally, route directions may differ in how they are conceptualized: this involves both the process of conceptualization as well as the resulting mental representation. What is represented in route directions and how it is communicated determines the ease with which the route directions can be used.

3.4 A Systematics of Route Direction Elements

As has been argued in the previous sections, different elements of an environment and a route may be employed for giving instructions, i.e. can be used in spatial references. The elements can be cataloged in a systematics according to three levels. These three levels reflect differences in the referred elements' granularity (see Section 3.4.7). There are elements that are not part of the immediate surrounding environment (the level of global references), elements that impose a global structure on the environment (the level of environmental structure), and elements that belong to path and route itself (the level of path and route). All three levels provide valid means to generate route directions; elements of different levels can be combined in the same route direction (see Chapter 4). However, elements of the first two levels are especially suited in providing coarse overview information for a route (see Section 7.2.1; cf. also Höök, 1991).

In the following, I will present the three levels in more detail (Sections 3.4.1 to 3.4.4). Section 3.4.5 links the systematics' elements to the kind of spatial knowledge they afford. Section 3.4.6 discusses the consequences of explicitly or implicitly representing elements in route directions, while in Section 3.4.7 I argue for different levels of granularity in and between the different levels of the systematics.

3.4.1 Level of Global References

Cardinal directions and global landmarks are cataloged on this level. These are elements referred to in route directions that are not part of the immediate surrounding environment in which the actions take place, i.e. elements not immediately reachable by the wayfinder. An example for an instruction on this level is "towards the sea," with 'the sea' being an example of a global landmark.

Since these references rely on elements outside the environment itself,[9] which usually results in rather coarse relations to these elements, the information they provide is coarse. Often, the direction given by global references is applicable to more than one option at a decision point, i.e. a wayfinder is confronted with a situation where several branches lead in the global

[9] Or, in case of cardinal directions, even elements independent of the current environment.

reference's direction. Hence, for instructions on following a specific route, global references often lead to ambiguous information and may not be used. They are, however, well suited for providing overview information on a route (Section 7.2.1).

3.4.2 Level of Environmental Structure

Elements of an environment may have a global influence on that environment. They impose a structure on the environment which results in a partition of that environment into distinctive parts. These elements, or a reference to the emerging distinctive parts, can be exploited in route directions if this provides unambiguous direction information.

Typical features are those elements of an environment Lynch (1960) termed *edges* and *districts*. Edges are either impenetrable barriers or uniting seams where they connect two districts. Districts are recognizable, distinct (large) areas of an environment with fuzzy boundaries, i.e. their extension cannot (always) be clearly defined (cf. Lynch, 1960). Both these elements are mainly important to structure coarse overview information on a route (see Section 7.2.1). But especially edges may be employed in route directions as distant landmarks, linear landmarks, or end qualifiers (see below). Furthermore, a slant provides structuring information as well. It results in part of the environment being uphill and part being downhill, which can be exploited in instructions like "go uphill until you reach a gas station."

3.4.3 Level of Path and Route

The third level of the systematics comprises elements that relate to paths and routes; these elements are considered to be part of a route itself. Elements cataloged on this level are decision points, as well as path annotations and landmarks functionally relevant for the route.

Configuration of Branching Points Instructions on the level of path and route typically employ an intrinsic reference frame with the wayfinder as reference object (see Section 3.1.3), like in "turn right" or "go straight." Instructions refer to the positions of an intersection's branches relative to the wayfinder's point of view; I term such references *egocentric references*. When creating route directions on this level the configuration of a branching point needs to be considered. Since the aim is to create route directions that are unambiguous the branch to take when following a route needs to be unequivocally identifiable. From a functional perspective, several branches at a decision point may broadly lead in the same direction and, therefore, the functionally relevant branch needs to be further specified. For example, if two branches lead to the right, they may be distinguished into 'veer right' and 'sharp right' and the spatial reference used in the abstract route directions may be 'sharp right' (see Sections 2.2.3, 4.2.3).

The branching point's configuration may also serve to identify this branching point, i.e. it may serve as a landmark (see Section 3.1.2). For example, dead ends of T-intersections or forks in the road provide such salient configurations. References to these configurations may change the direction relation to choose (see Section 4.2.3).

Landmarks I consider landmarks that are part of the surrounding environment and that can be used in route directions, i.e. that are functionally relevant for a route, to be part of the route. Therefore, they are listed on this level. The functional role of landmarks is also discussed in Hansen et al. (2006).

From a structural perspective, landmarks can be considered as a point, as linear, or as areal. Point landmarks are located in small, restricted areas of an environment, such as salient buildings (like a church). The other two kinds, linear and areal landmarks, extend across an environment (like a river or a forest). Furthermore, in relation to a route a landmark can either be at a decision point, at a route-segment between two decision points (Herrmann et al., 1998), or in some distance to, but visible from the route. I call the latter kind of landmarks *distant landmarks* (cf. Lovelace et al., 1999). Distant landmarks differ structurally and functionally from global landmarks. Distant landmarks are part of the surrounding environment, i.e. they are in principle reachable by a wayfinder and may, for example, function as a landmark at a decision point for some other routes through that environment. Global landmarks are outside the current environmental space and function as global reference objects independent of a specific route through the environment.[10]

From a functional perspective, landmarks at a decision point can be used to identify a decision point ("turn left *at* the church"). Landmarks between decision points may be employed to further describe the route and to function as confirmation that one is still on the right track ("you *pass* a church"). Finally, distant landmarks are like beacons. Assuming they are visible while passing a decision point—or ideally even several ones—they can be used as pointers to a certain direction ("go *towards* the church").

Linear and areal landmarks function differently to point routemarks. Examples of their applicability are depicted in Figure 3.4. Such landmarks not only identify a decision point, but may also allow chunking several decisions into one decision (see Section 3.1.2), which results in a linear pattern. Examples for such instructions are "follow the river" or "walk along the forest." Here, both kinds of landmarks function linearly; they determine the

[10]That does not imply that they need to be (sensibly) applicable as reference objects in route directions for every route.

actions for several decision points. Therefore, such instructions require additional information, an *end qualifier*, that establishes the point until where the instructions hold, for example "until you reach the gas station."

Instructions may also involve references that afford an areal structure of the reference object, like in "go around the church until you are at the main gate." In such cases, a landmark functions areal. This is conceptually different to an instruction like "follow the river," where the function is linear. Computationally, both cases can be treated similarly, as is demonstrated in Section 5.2.3.

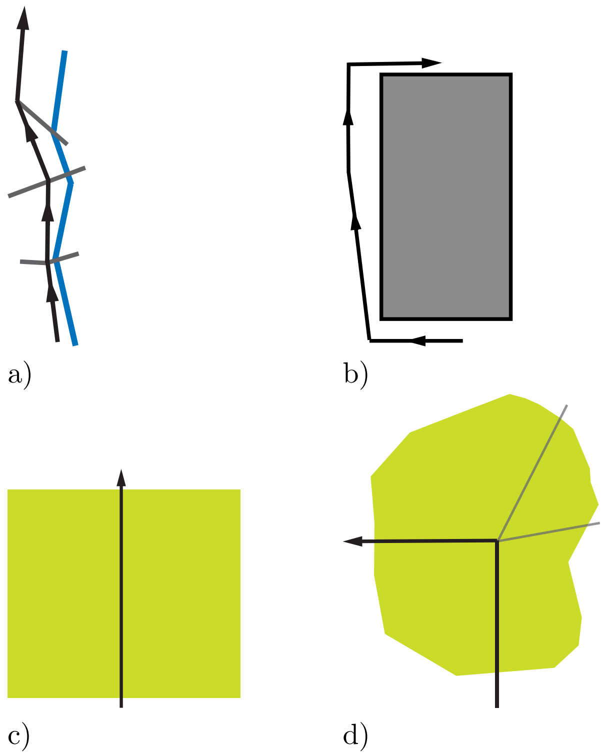


Figure 3.4: References to linear and areal landmarks: a) "follow the river" (the grey line next to the route), b) "go around the building", c) "cross the park", d) "turn left in the forest".

In principle, instructions that tell a wayfinder to turn in an areal landmark are also possible, for instance "turn left in the forest;" another example for referring to areal landmarks is something like "cross the park." I do not consider these possibilities. Generation of instructions is done on the level of single decision points; instructions shall be determined that are unambiguous for each decision point. An instruction "cross the park" may be used if there is no option to choose different routes within the park, i.e. if there are no decision points. In this case the instruction may only be given for a decision point at the park's entrance and functions rather like a beacon, i.e. "follow the park's interior until you get to the other side of the park." If there are further decision points in the park, i.e. a wayfinder needs to decide which way to take within the park, "cross the park" is an ambiguous instruction for the decision points in the park since other routes may also lead through the park. An implicit assumption associated with this kind of instruction may be to always go straight, but this assumption is not sufficient to make the instruction really unambiguous.

Similar arguments hold for instructions like "turn left in the forest." To be unambiguous, there can only be a single decision point in that forest.[11] If there are several decision points in the areal feature and the relevant one is further specified (like in "[while] in the forest turn left at the third intersection"), the areal feature ('forest') does not function as an areal landmark indicating the action to be performed any more. In this case it rather provides a kind of confirmation information priming the wayfinder for the upcoming decision.

For landmarks at a decision point, their position at that decision point is important for the conceptualization of a turning action (cf. Klippel, 2003b). An action may be taken after passing a landmark ("turn *after* the church"), before passing the landmark ("turn *before* the church"), or the landmark may not be located at a functionally relevant branch at all ("turn where the church is"). Especially landmarks immediately before a turn are easily conceptualized, as the turning action occurs immediately after them. They are, thus, a good identifier for a decision point.

Figure 3.5 provides an overview on the types of landmarks considered in this thesis. It does not cover the usage of structural elements, such as a slant or T-intersections, as landmarks.

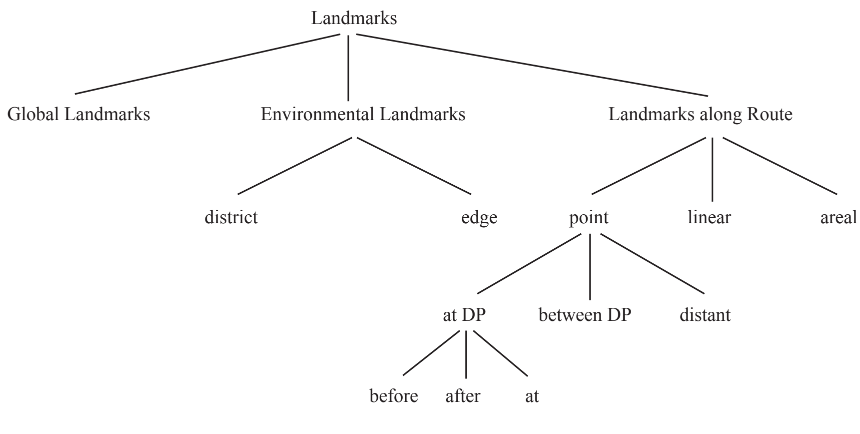


Figure 3.5: An overview of the types of landmarks considered.

[11] Or at least only one decision point where a left turn is possible.

Path Annotations With *annotations* of a path I denote markers that are set up to unequivocally identify that path. In a city environment, the most typical examples are street names. Furthermore, street signs, like those indicating the direction to a hospital or the main station, may function as such annotations. Another example are markers of a hiking trail which are also meant to unequivocally provide direction information to hikers. Such annotations can be considered artificial landmarks usually put up to guide people to specific, prominent goals. In principle, references to annotations are a good means to provide unambiguous, easily followable directions—taken that the annotations are well spotable and identifiable. There is evidence, however, that in generating route directions people refer more often to landmarks than to street names and that landmarks are more effective than street names in route guidance (cf. Tom & Denis, 2003),

Table 3.1 summarizes all identified elements that can be used in route directions, cataloged according to the three levels of granularity identified: global references, references to elements imposing a structure on the environment, and references to elements of the route itself.

Table 3.1: The elements cataloged in the systematics.

Global References
cardinal directions
global landmarks

Environmental Structure
edges
districts
slant

Path and Route
egocentric references
landmarks at decision point
landmarks between decision points
distant landmarks
linear and areal landmarks
path annotations

3.4.4 Confirmation Annotations

When following instructions that do not require a direction change for a considerable amount of a route, for example, when an instruction like "turn left at the church" involves no turn at intermediate decision points, a wayfinder benefits from confirmation information, i.e. references to elements of the environment passed along the route. This information indicates that she is still on the right track and keeps her from trying to recover from an assumed wayfinding error which consequently would lead to an error that actually has not occurred yet.

Suitable elements for confirmation information are landmarks between decision points and edges. Both can easily be referred to by the instructor and can be well identified by the wayfinder. I term references to these elements for confirming the right track *confirmation annotation.*

3.4.5 Spatial Knowledge Connected with the Systematics' Elements

Picking up the discussion from Section 3.2 on different kinds of spatial knowledge involved in route directions, Table 3.2 lists the qualitative knowledge connected to the systematics' elements that allows determining and interpreting references to the different elements.

Table 3.2: Qualitative knowledge connected to the different elements

Global References	
cardinal directions	direction
global landmarks	direction, distance
Environmental Structure	
edges	ordering (dynamic)
districts	ordering (dynamic)
slant	direction
Path and Route	
egocentric references	direction
landmarks at decision point	ordering (static)
landmarks between decision points	direction, ordering (dynamic)
distant landmarks	direction, distance
linear and areal landmarks	ordering (dynamic)
path annotations	ordering (dynamic)

Cardinal directions and egocentric references require directional knowledge. They involve changes of the current movement direction, like turning 'north' or 'left'. These turns may result in arbitrary angular deviations from the previous direction. To indicate these changes, the relations indicating the direction change may be refined or annotated with further information to resolve possible ambiguities (see Sections 4.2, 5.2), which results in finer granularity of the provided direction information. Global landmarks, distant landmarks, and landmarks between decision points also provide direction information, though this is restricted in that they are not used to indicate arbitrary directional changes. The first two allow a wayfinder to keep a coarse overall direction, for example, in instructions like "keep the sea behind you" or "go in direction of the TV tower." That is, a wayfinder either moves in the direction of the referred feature or in the opposite direction. Global and distant landmarks can also be referenced in clock directions, as in "keep the mountains in the direction of 2 o'clock," which for instance, is sometimes done when flying or sailing. Such directions also indicate the overall direction to keep. They do not indicate a direction change as cardinal or egocentric directions do. Clock directions are not further addressed in my thesis. Landmarks between decision points only allow for references telling a wayfinder to move in their direction, i.e. can be used like a beacon to mark the direction to take. Such a landmark is passed while getting from current decision point to the next. Hence, the relation between wayfinder and landmark changes from when the landmark is in front of the wayfinder to when the landmark is behind the wayfinder. Accordingly, this element also involves dynamic ordering information in the sense introduced in Section 3.2.2. A slant, finally, can be used for one-dimensional direction information. However, when considering a three-dimensional space, the relevant dimension differs from the ones involved so far, i.e. such references state changes in relative height by moving either up- or downhill.

Ordering information describes the relation of environmental features to a wayfinder and changes in these relations occurring during movement of the wayfinder. Such information is connected to several elements of the systematics: edges, districts, linear and areal landmarks, and path annotations all require a wayfinder to either change (edges, districts) or to keep (linear and areal landmarks, path annotations) the relation while moving. Ordering information connected to landmarks between decision points has already been discussed in the previous paragraph. Path annotations are special—they stand for the route-segment they annotate. Path annotations of the same kind may annotate a sequence of segments, for example, a red dot re-occurring along a path marking a hiking trail (like in Fig. 1.1). Thus references to the annotations, for instance "follow the markers," instruct a wayfinder to follow the route-segments annotated by the markers. The physical signs are relevant to indicate these route-segments, but the spatial relation between signs and the wayfinder is not. A wayfinder has to follow this sequence of route-segments; this can be considered similar to following a linear landmark. Ordering information, finally, is used to determine and interpret the position of a landmark at a decision point relative to a wayfinder and that decision point (see Section 3.2.2).

As argued in Section 3.2.3, distance information is of minor importance in context-specific route directions. Still, references to global and distant landmarks allow inferring distance information. As a wayfinder either moves in the direction of these features or away from them, the distance of the wayfinder to these features changes—while the ordering relation stays the same. This change of distance provides relative distance information on a coarse qualitative level. That is, a wayfinder may check whether she followed an instruction involving references to these features correctly by judging whether her distance to the referred feature changes according to the given instruction: increasing distance when moving away from the feature and decreasing when moving towards it.

The different kinds of knowledge involved are also reflected in the relations used to refer to the elements in route directions (cf. Section 4.2).

3.4.6 Explicit vs. Implicit Representation

A route can be represented as a sequence of decision points. In order to successfully follow a specific route a wayfinder has to take the correct decision at each individual decision point. However, the application of chunking combines several of these decision points into a single instruction which is, as argued in Section 3.1.2, an important mechanism in route directions. Hence, the resulting representation of a route does not necessarily contain every single decision point anymore. For those decision points not explicitly represented in route directions, the decision due at those decision points may also not be stated explicitly.

On the other hand, when following a route a wayfinder needs to take a decision at every decision point encountered along the route. She must be able to infer the instructions that are only implicitly represented in the route directions in order to know which further direction to take. Consequently, the route directions need to be complete, i.e. provide instructions on a route that leads from origin to destination. They also need to be correct, i.e. provide the instructions such that every decision necessary can be derived from them.

3.4.7 Granularity in the Systematics

Granularity is one of the fundamental aspects of knowledge representation (cf. Zadeh, 1979; Hobbs, 1985). The three levels of the systematics reflect different levels of granularity of the cataloged elements which offer access to knowledge on an environment. Accordingly, route directions generated using these elements provide information on how to follow a route on different levels of granularity. Changes in granularity reside between the levels of the systematics as well as within these levels. In the systematics, granularity refers to how closely route directions are linked to individual decision point/action pairs and the corresponding branching point's configuration, i.e. to what extent route directions abstract from a description of the exact sequence of decision point/action pairs.

Comparing the systematics' levels, the coarsest granularity level is that of global references. Referring to elements that are not part of the surrounding environment results in coarse direction information which is not explicitly based on the structure of the environment itself. Such instructions exploit the fact that they lead to unequivocal choices at those decision points they hold for; usually these elements are used rather for providing overview information on a route (Section 7.2.1). Route directions on the second level of the systematics—the level of environmental structure—still provide coarse information on the further direction to take, but which explicitly takes into account an environment's structure and is, accordingly, closer related to the path embedded in that environment. Consequently, this kind of instruction is on a finer granularity level than those route directions using elements of the systematics' first level. The level of path and route contains elements of the route itself. Route directions employing these elements usually refer explicitly to decision points; their structure is close to the decision point/action pairs themselves. The third level is the finest level I consider in my work.

A change of granularity also occurs within the levels of the systematics. Chunking instructions (see Section 3.1.2) increases the degree of abstraction from individual decision point/action pairs. This holds for all three levels, but especially for the third one. Different kinds of chunking result in route directions on different levels of granularity. Landmark and structure chunking, for example, combine a number of decision points that is not specified in the resulting instruction, i.e. they abstract from the exact number of decision points involved, and may provide a single instruction for a large part of a route. In numerical chunking as presented in Klippel et al. (2003), the number of decision points that are grouped into a single instruction is explicitly mentioned. Such instructions are only sensibly applicable for a small number of decision points (cf. Klippel, 2003b; Klippel et al., 2005e) and thus, also only for a small part of a route. Hence, landmark and structure chunking result in instructions on a higher granularity level than numerical chunking as they abstract to a greater degree from a specific sequence of decision points.

Finally, taking into account the configuration of a branching point, for example, by further qualifying a turning instruction, is on the finest level of granularity, as this is directly based on an individual decision point and its structure.

3.5 Summary

Which elements of an environment and which actions are considered important for instructions on how to follow a route, determines how the route is conceptually represented in route directions. This determines to a great extent which information on the route and the environment gets externalized in communicating route directions. Consequently, this likewise determines to a great extent the wayfinder's conceptualization of the route. Choices made on what to communicate in route directions have information- and representation-theoretic, as well as cognitive, consequences. Different choices on elements and actions to be—explicitly or implicitly—part of the route directions result in different representations of a route, and therefore, in differences in the ease of conceptualizing the route.

In this chapter, I identified elements that can be employed in route directions. These elements are cataloged in a systematics according to three levels: the level of global references, the level of environmental structure, and the level of path and route. These levels reflect different levels of granularity, i.e. how closely route directions are linked to individual decision point/action pairs. Spatial chunking allows combining several instructions into a single one, which results in some instructions being only implicitly represented. These instructions still need to be easily inferable.

The next chapter presents the process for genereating context-specific route directions that is based on the systematics and results in an abstract specification of the actions to be performed that takes into account the principles and requirements identified in this chapter.

Chapter 4

Context-Specific Route Directions

In this chapter, I elaborate on the generation process for context-specific route directions. It is based on a graph representation of the environment (Section 4.1) and makes use of the systematics of elements presented in the previous chapter. These elements are used in generating instructions, which are represented using *abstract turn instructions* (Section 4.2). Generation of route directions is realized as an optimization process as argued for in Section 4.3. The generation process is called GUARD (Generating Unambiguous, Adapted Route Directions).[1] It is detailed in Section 4.4; the process is flexible and able to integrate different route direction principles (Section 4.5).

4.1 Representing a Route

Decision points are pertinent elements of a route (cf. Daniel & Denis, 1998). Here, a wayfinder needs to decide on the further direction to take. Accordingly, I concentrate on decision points in generating context-specific route directions. A route is modeled as a sequence of decision points.

The aim is to adapt route directions to the current situation, i.e. to the current action to be performed along the route in the current surrounding environment. While it is sufficient to concentrate on a sequence of points in representing the route itself, a representation of the surrounding environment is needed for the generation of route directions. To this end, I use an embedded graph of the environment's path-network, annotated with additional information; for example on landmarks (see Section 5.1.2 for further details). This graph reflects the network's geometry; it allows calculating

[1] The acronym reflects that the generated route directions adapt to environmental characteristics and the route's properties and that they unambiguously identify the branch to take at each decision point.

distances and directions between the environment's features. Nodes correspond to intersections and edges represent the path-segments. Each decision point of a route is linked to a node in the graph. Such a graph may be considered a *Route Graph* (cf. Werner et al., 2000). In GUARD, determining instructions is based on this graph.

4.2 Abstract Turn Instructions

Route directions provide information on how to proceed from each decision point of a route.[2] They are a sequence of instructions, where each instruction states the action to be performed at a decision point and may contain references to environmental elements, for example landmarks, to ease identifying a decision point and its accompanying action.

In GUARD, generation of instructions takes into account environmental characteristics and a route's properties. This is in contrast to internet route-planners, for example. Such planners generate instructions always using the same pattern, disregarding environmental features as has been illustrated in Chapter 1. The systematics of elements presented in Section 3.4 provides the basis for generating context-specific route directions. Decisions that need to be taken along a route are described in reference to these elements. Such a description is represented in a formal notation and is termed an *abstract turn instruction* (ATI). ATIs represent a decision point and its accompanying turning action. They consist of a decision point's identifier and a so-called *direction relation*, which represents the action at a decision point. The direction relation may be annotated with a reference to a landmark or environment's structure. Additionally, there might be a further annotation that represents confirmation information; for instance, a landmark passed which indicates that a wayfinder is on the right track (see Section 3.4.3). For example, the abstract turn instruction $(\mathtt{DP}_1\mathtt{,left\ /\ church})$ may represent a situation, where at the first decision point of a route, a wayfinder has to turn left at a church. The general structure of an abstract turn instruction is as follows:

$$(\mathtt{ID}_{\mathtt{DP}}, \mathtt{direction\,relation} \;/\; \mathtt{annotation}; \mathtt{confirmation\,annotation})$$

For each element of the systematics there is a corresponding set of direction relations. These sets differ across the elements. Each relation represents information on which action to take at the corresponding decision point. In that, they resemble symbolic operators describing directional phrases as defined, for example by Jackendoff (1990), or Eschenbach et al. (2000). However, it is important to note that the direction relations are not meant to be actual (verbal) output of, for example, a wayfinding assistance system.

[2] Remember that due to chunking, this information may be represented only implicitly (cf. Section 3.4.6).

They are an abstract specification of actions representing the applicability of the systematics' elements for a given decision point. They represent possibilities for how a decision point/action pair can be described according to the systematics.[3] The externalization of abstract turn instructions, i.e. their transformation into concrete—verbal or graphical—route directions is outlined in Sections 6.2.2 and 7.2; however, keep in mind that this is not the focus of the presented work. In the following, I list the sets of direction relations used to represent instructions referring to the different systematics' elements. These sets are the basic representation used in context-specific route directions.

4.2.1 Representation of Global References

There are two elements on the level of global references: cardinal directions and global landmarks. Cardinal directions are represented using the relations `north`, `east`, `south`, and `west`; these may be combined with more fine grained relations, for example `southwest` or `east-northeast`. Cardinal directions are most often employed when providing coarse overview information for a route (see Section 7.2.1).

Global landmarks are not part of the current environment; they cannot be reached by the wayfinder. To denote references to global landmarks, `towards` or `away`, annotated with the landmark's label, is used when moving in direction of the global landmark or away from it, respectively. That is, moving in direction of the global landmark 'sea,' for example, results in `(towards / sea)`. This corresponds to an instruction like "go in direction of the sea." Table 4.1 summarizes the direction relations employed on the level of global references:

Table 4.1: Direction relations for the elements on the level of global references.

cardinal directions	`north` | `east` | `south` | `west`
global landmarks	`towards` | `away` + global-landmark-label

4.2.2 Representation of Environmental Structure

Similar to global references, elements on the level of environmental structure are mainly used for coarse overview information. But, as explained in Section 3.4.2, edges (for example highways cutting an environment), may function as distant or linear landmarks. Accordingly, the same relations as

[3]Still, I choose relation terms like `left` or `towards` because they are more readable than terms like `a`, `b`, or `c`.

for distant and linear landmarks, respectively, are used (see Section 4.2.3). If an edge is used as distant landmark, `towards` or `away` annotated with the edge's label are employed; `follow` in the other case. For a slant, the relations `uphill` and `downhill` are used. Table 4.2 summarizes the direction relations for this level:

Table 4.2: Direction relations for the elements on the level of environmental structure.

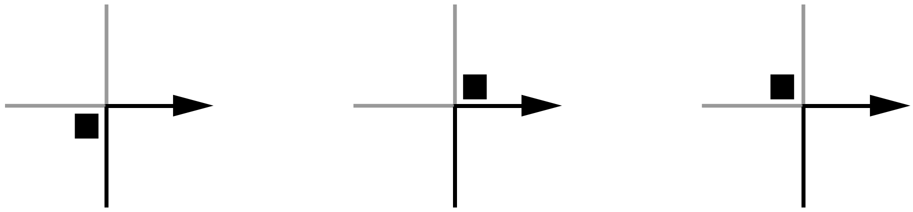
edge	`towards` | `away` | `follow` + edge-label
slant	`uphill` | `downhill`

Districts are not used for route directions on this level. As explained in Section 3.4.3, they may be used as areal landmarks, but their application usually results in ambiguous instructions. A district may also function as a distant landmark, i.e. as a beacon indicating to move in its direction. But this also results in rather coarse instructions that are often ambiguous due to a district's size. A district covers a considerable part of an environment and often has fuzzy boundaries. Districts are used as a structuring feature in coarse route directions, though (see Section 7.2.1).

4.2.3 Representation of Path and Route

Instructions on the level of path and route usually employ an intrinsic reference system with the wayfinder as reference object. To represent egocentric references, I use the sector model elicited in the wayfinding choreme theory (see Section 2.2.3). The model comprises three basic directions—`straight`, `left`, `right`—and two additional qualifiers for left and right—`veer` and `sharp`—leading to seven different directions. Both `veer` and `sharp` can be used to disambiguate a branch if several branches head off to the left or right.

These egocentric relations can be annotated with a landmark-label if a landmark is present at a decision point. As explained in Section 3.4.3, a landmark's location at a decision point can be specified relative to the turning action. It may be passed before the action takes place, after the action, or the landmark may not be located at a functionally relevant branch. To distinguish these three cases, an additional label is added to the abstract turn instruction: $\texttt{lm}^{<}$ if a landmark is passed before the turn occurs; $\texttt{lm}^{>}$ is used if a landmark is passed after the turn occurred; and $\texttt{lm}^{-}$ denotes that a landmark is present at a decision point, but is not located at a functionally relevant branch (see Fig. 4.1). That is, an instruction like "turn left before the church" is represented by the abstract turn instruction $\texttt{(left / church lm}^{<}\texttt{)}$.



a) b) c)

Figure 4.1: A landmark passed a) *before* an action ($\texttt{lm}^{<}$), b) *after* an action ($\texttt{lm}^{>}$), or c) not located at a functionally relevant branch ($\texttt{lm}^{-}$).

Additionally, specific configurations of an intersection's branches may serve as a landmark. In this case, the egocentric direction relation may need to be adapted. Dead ends of T-intersections, for example, only allow turning left or right. Such a configuration is comprised of one branch heading off to the left and one to the right. Therefore, the egocentric direction is coarsened to `left` or `right`, respectively, and annotated with a label describing this configuration (`t-intersection`). A resulting abstract turn instruction may be `(left / t-intersection)`.

A reference to distant landmarks or to landmarks between decision points is represented by `towards` or `away` annotated with the landmark's label—similar to global landmarks (Section 4.2.1). Linear and areal landmarks afford an action to move along them as they stretch in the direction of further movement. This is denoted by the relation `follow` annotated with the landmark's label. The same holds for path annotations which indicate the further direction of movement by either pointing in that direction or by being located at the intersection's branch to follow, i.e. marking the branch to take. Table 4.3 summarizes all relations used on this level:

Table 4.3: Direction relations for the elements on the level of path and route.

egocentric reference	`straight` | `[veer` | `sharp] left` | `[veer` | `sharp] right`
T-intersection	`left` | `right` + `t-intersection`
landmark at DP	landmark-label + $\texttt{lm}^{<}$ | $\texttt{lm}^{>}$ | $\texttt{lm}^{-}$
distant landmark	`towards` | `away` + landmark-label
landmark between DP	`towards` | `away` + landmark-label
linear landmark	`follow` + landmark-label
area-like landmark	`follow` + landmark-label
path annotation	`follow` + annotation-label

4.2.4 Confirmation Annotations

As has been stated in Section 3.4.3, landmarks between decision points, linear landmarks and edges are good candidates to be used as confirmation annotations in route directions. Landmarks between decision points are located on the route; they are passed along a route-segment. Accordingly, to represent confirmation information that refers to landmarks, the relation `pass` annotated with the landmark-label is employed. A linear landmark or edge that is functionally relevant for a route may function linearly. If this is not applicable it may still be used for confirmation information in case it is crossed while following the route, for example, by walking over a bridge across a river—the edge. Hence, the relation `cross` annotated with the linear landmark or edge's label is used to represent such confirmation information; the example of crossing the river is represented as `(cross river)`. Table 4.4 lists the structure of these confirmation annotations:

Table 4.4: Relations of confirmation annotations.

edge	`cross` + edge-label | linear-landmark-label
landmark between DP	`pass` + landmark-label

4.2.5 Summary of Abstract Turn Instructions

Abstract turn instructions are a formal specification of an action to take at a specific decision point. They make use of the elements of the systematics presented in Section 3.4. Table 4.5 summarizes the relational representations of these elements as they are used in Guard. For a given decision point often several of these representations are possible, i.e. more than one element may be applicable to unequivocally describe the direction to take at a decision point. In this case, it is necessary to decide which of these possibilities to choose for the context-specific route directions. Taking this choice can be seen as an optimization problem as will be elaborated next.

4.3 Generating Route Directions: An Optimization Problem

With Guard I aim at generating an abstract representation of route directions. This abstract representation can be externalized in a specific modality; the resulting route directions ease the conceptualization of a route. The abstract representation that best fits this aim needs to be chosen from all possibilities to generate such representations. More precisely, for each decision point along the route the abstract turn instruction that corresponds

Table 4.5: Summary: Direction relations of abstract turn instructions.

<table>
<tr><th colspan="2">Level of Global References</th></tr>
<tr><td>cardinal directions</td><td>north | east | south | west</td></tr>
<tr><td>global landmarks</td><td>towards | away + global-landmark-label</td></tr>
<tr><th colspan="2">Level of Environmental Structure</th></tr>
<tr><td>edge</td><td>towards | follow + edge-label</td></tr>
<tr><td>slant</td><td>uphill | downhill</td></tr>
<tr><th colspan="2">Level of Path and Route</th></tr>
<tr><td>egocentric references</td><td>straight | [veer | sharp] left | [veer | sharp] right</td></tr>
<tr><td>T-intersection</td><td>left | right + t-intersection</td></tr>
<tr><td>landmark at DP</td><td>landmark-label + $\mathtt{lm}^{>}$ | $\mathtt{lm}^{<}$ | $\mathtt{lm}^{-}$</td></tr>
<tr><td>distant landmark</td><td>towards + landmark-label</td></tr>
<tr><td>landmark between DP</td><td>towards + landmark-label</td></tr>
<tr><td>linear landmark</td><td>follow + landmark-label</td></tr>
<tr><td>areal landmark</td><td>follow + landmark-label</td></tr>
<tr><td>path annotation</td><td>follow + annotation-label</td></tr>
<tr><th colspan="2">Confirmation Annotations</th></tr>
<tr><td>edge</td><td>cross + edge-label | linear-landmark-label</td></tr>
<tr><td>landmark between DP</td><td>pass + landmark-label</td></tr>
</table>

to the easiest conceptualization of the action to be performed needs to be chosen.

The kind of instruction to choose may depend on the kind of instruction chosen for previous, or future, decision points. For example, for a given decision point, possible abstract turn instructions may be (`follow / river`), (`left`), and (`left / church` $\mathtt{lm}^{<}$). Since instructions for several consecutive decision points may be chunked, it is not sufficient to look at these instructions in isolation. For example, if there is an instruction that contains the direction relation (`straight`) for the predecessor of the given decision point, it is advantageous to choose one of the instructions containing `left` as these can be chunked with the predecessor's instruction. Thus, all abstract turn instructions that can be generated for a given decision point need to be judged according to their consequences regarding conceptualization. To that end, possible abstract turn instructions for other decision points of the route need to be taken into account. This dependence of a local choice—

an abstract turn instruction for a single decision point—on choices made elsewhere—abstract turn instructions for other decision points—clearly indicates that the generation process of context-specific route directions can be realized as an optimization problem. The aim is to generate optimized route directions for a route.

To decide which abstract turn instruction best fits the aim of easy conceptualization, there is need for a means to compare different abstract turn instructions with respect to their conceptualization. Rules for deciding which abstract turn instruction to choose in which situation are needed. These rules define an optimization criterion for the optimization process (i.e. a function to be maximized or minimized). Potentially, for each decision point, many abstract turn instructions may be applicable. And the choice made for one decision point may depend on abstract turn instructions chosen for other decision points. Accordingly, it is not sensible to define a specific rule for every situation that might occur in creating context-specific route directions. Instead, there is need for some heuristics: general rules are needed which provide guidelines and lead to sensible choices.

Guard, the process used for generating context-specific route directions, is presented in more detail in the next section. It is flexible with respect to the chosen optimization criterion. I argue for sensible criteria in Section 4.4.4.1 and detail how such criteria get applied in Section 4.4.4.2. Section 4.5 then discusses how different route direction principles (see Section 2.1.5) can be transformed and even combined into optimization criteria for the process.

4.4 Guard: A Computational Process for Generating Context-Specific Route Directions

In this section I outline the components of Guard, the computational process used for generating context-specific route directions and their interplay. Here, I concentrate on providing an overview on Guard and on explaining the underlying principle considerations, while the next chapter details the algorithmic realization.

Guard consists of four major steps depicted in Figure 4.2. First, for each decision point all applicable abstract turn instructions are generated (Section 4.4.1). The individual abstract turn instructions are then combined applying simple chunking rules (Section 4.4.2); these chunks are adapted to general chunking principles in a post-processing step (Section 4.4.3). Finally, in an optimization process (Section 4.4.4) those chunks from the generated set of chunks that result in the optimal context-specific route direction (Section 4.4.5) are chosen.

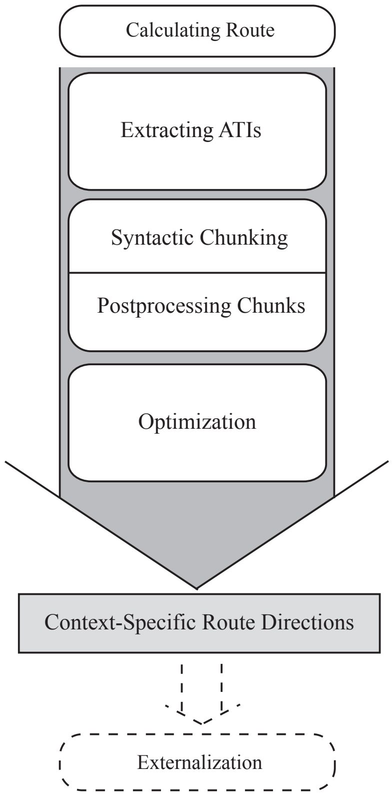


Figure 4.2: Overview on GUARD, the generation process for context-specific route directions. For a given route context-specific route directions are determined in a four-step process. These may then be externalized, which is not covered in the presented process.

4.4.1 Extracting Abstract Turn Instructions

GUARD begins by generating all applicable abstract turn instructions for each decision point, which is comparable to generating all possible conceptual elements for an instruction (cf. Marciniak & Strube, 2005b).

Checking for an element's applicability entails two considerations: first, for each element of the systematics it needs to be tested whether it is applicable in principle. For example, the presence of a landmark at a decision point is a prerequisite for annotating a direction relation with a landmark. Second, if an element is applicable it needs to be tested whether a corresponding abstract turn instruction is unambiguous, i.e. whether such an instruction refers to the route-segment to take without being suitable for other branches, as well. Chapter 5 details how this can be realized algorithmically for each of the systematics' elements. This extraction step (termed *extraction*) yields a set of applicable abstract turn instructions. The sets of

ATIs represents several ways for describing the action to be performed for each decision point. The sets need to be further processed to result in a coherent, optimized context-specific route direction. The next step in that direction is chunking the individual abstract turn instructions.

4.4.2 Chunking Abstract Turn Instructions

Chunking is an important mechanism in the conceptualization of routes (see Section 3.1.2). The second step of the generation process combines individual abstract turn instructions for single decision points into chunks, i.e. into instructions for several consecutive decision points.

Abstract turn instructions consist of a decision point's identifier and a (potentially annotated) direction relation that represents the action to be performed at the decision point. In GUARD, chunking is realized by applying rules defining valid combinations of these direction relations. The basic chunking rules are simple and straightforward. In a post-processing step (Section 4.4.3), more sophisticated chunking principles that reflect cognitive and structural considerations in combining instructions are applied. These adapt the initial chunks so that they meet these requirements, and turn them into sensible elements of route directions. This division of the chunking process allows for flexible handling of different considerations in chunking and generating route directions (see Sections 2.1, 4.5).

The initial chunking process is based on simple, syntactic properties of the abstract turn instructions. First, if two consecutive direction relations are equal, for example, if two abstract turn instructions contain the relation `left`, they can be combined. This simple rule may lead to an arbitrary number of equal direction relations being chunked—it chunks a complete sequence of equal direction relations into one chunk independent of the sequence's length. Second, the relation `straight` can be combined with any other egocentric direction relation, for example `right`. Such a combination may result in the instruction `straight` being only implicitly represented in externalized route directions (cf., e.g., Klippel, 2003b; Klippel et al., 2003) as, for example, in "turn left at the third intersection." The chunk is completed as soon as `straight` is combined with another egocentric direction relation.[4] Table 4.6 summarizes the basic chunking rules.

However, these rules are too simplistic as they do not account for annotations, like references to landmarks. These annotations need to be taken into account in order to determine whether two abstract turn instructions are chunkable. This adds pre-conditions to the rules which are further explained in the algorithmic realization (Section 5.3).

[4]Combining `straight` with `straight` does not end chunking, since this corresponds to the first rule, i.e. chunking equal direction relations.

Table 4.6: Chunking rules for abstract turn instructions.

equal direction relations	`relation` + `relation`
egocentric references	`straight` + [`veer` | `sharp`] `left` | `straight` + [`veer` | `sharp`] `right`

4.4.3 Postprocessing Chunks

The simple chunking rules presented in the previous section are sufficient to generate every chunk possible based on the previously determined abstract turn instructions. However, these chunks are not guaranteed to be sensible, i.e. some may violate cognitive or structural principles. For example, chunks referring to linear landmarks (using ATIs like `follow / river`) require an end qualifier. Therefore, the last chunk's abstract turn instruction needs to be annotated with such a qualifier. Or—according to Klippel et al. (2005e)—instructions like "turn left at the 27th intersection," which corresponds to chunking 26 times `straight` with `left`, are cognitively implausible and rarely produced by human instructors.

Therefore, the generated chunks need to be checked regarding their appropriateness. Separating the initial, simple combination process and this check (termed *postprocessing*) allows taking into account different chunking principles in Guard, for example, those presented in Klippel et al. (2003) or Dale et al. (2003), without need to change the chunking process itself. This is advantageous and desirable with respect to modularity and adaptability of modeling and system implementation.

Chapter 5 provides details on how the adaptation is realized for sample chunking principles. Generally, adaptation is performed by step-wise pruning of chunks. Each chunk is checked for whether it obeys the chunking principles. If this is not the case, the abstract turn instruction for the chunk's last decision point is removed, and the check is repeated until the chunk is deemed appropriate or it is empty, in which case it is removed. This results in a set of chunks that are valid according to the given chunking principles, and may be part of a context-specific route direction for the route at hand. Which chunks are actually chosen is determined in the optimization process detailed next.

4.4.4 Optimization

For optimization purposes we need a criterion that tells us if something is optimal. Such a criterion allows comparing and rating two solutions. And we need a process that optimizes a given solution with respect to the optimization criterion. In this section, I first present arguments that are to be taken into account in finding suitable optimization criteria for generating context-specific route directions. Then I elicit the optimization process.

4.4.4.1 Optimization Criteria

Context-specific route directions—the result of the optimization process—are supposed to be well conceptualizable. They need to be easily processable and usable by a wayfinder. The chosen optimization criterion needs to reflect this aim. It needs to account for principles of good route directions (see Sections 2.1.5, 4.5), as well as principles that support an easy conceptualization argued for in the following. These principles rely on chunking and high-level route directions, i.e. route directions on a coarse level of granularity.

From an information-theoretic perspective, a small number of chunks reduces the amount of information that needs to be communicated. The size of the message decreases, which leads to a decrease in memory load: the wayfinder needs to remember less information. This relates to Grice's (1975) principles of communication, especially the ones he terms *quality* and *quantity*: the information provided needs to be correct and should not contain any details that are unnecessary for the message's purpose. To put it another way, a reduction in the number of chunks results in a decreased amount of information explicitly represented, and an increase of the information that needs to be inferred. These inferences need to be easy in order to fulfill the requirements of good conceptualization (cf. Section 3.4.6).

The application of chunking also reduces the wayfinder's processing involved in conceptualizing a route. As argued before, a principle of cognitive ergonomics is to combine several instructions into a high-level instruction if possible. Performing this chunking requires additional processing of the route directions, i.e. increases the cognitive load of a wayfinder. But since chunking of instructions is already done in the generation of context-specific route directions, the wayfinder does not need to perform this herself anymore, which eases the cognitive processing of the route directions.

Furthermore, instructions on a coarse level of granularity reduce the problem of matching an expected decision point/action pair with the real environment. High-level instructions are less prone to errors if a conceptualized decision point/action pair does not (exactly) match the actual situation in the environment. An instruction "turn right," for example, might get a wayfinder into trouble if the configuration of branches that meet at an intersection does not seem to include a branch she considers to lead to the right. An instruction such as "follow the signs to the train station," on the other hand, does not depend at all on the configuration of the intersections; all that is required is that there is actually a sign pointing in the direction of the train station. Thus, with route directions on coarser levels of granularity, a wayfinder's dependence on the environment meeting her conceivded model of the spatial situation is lessened. To put it another way, conceptualizations stemming from high-level route directions are less dependent on an actual, exact configuration found in the environment. But, as they are tightly coupled to the current environmental characteristics, they abstract from this configuration in such a way that matching the conceptualization with the current situation is well supported.

Finally, it can be argued that providing route directions on a coarse level of granularity shifts the task of wayfinding in an environment towards the direction of the task of trip planning through an environment (see Section 7.2.1; cf. also Timpf et al., 1992). Such route directions include fewer 'real' decision points, i.e. fewer decision points where a wayfinder actively needs to remember a direction change. As high-level route directions combine several decision points into one big decision, a wayfinder just needs to remember the point until such an instruction holds. All decisions in-between can be easily inferred. For example, "turn left at the third intersection" indicates a direction change at the third intersection to come. The information implicitly represented is that the wayfinder has to maintain her current direction of movement at the first and second intersection. The advantages of high-level route directions are even more obvious when looking at instructions like "follow the markers" for a hiking trail. Here, a single instruction suffices to lead a wayfinder to her destination. But following the markers may involve many direction changes while hiking. That is, a route direction on a coarse level of granularity may render decision points that actually involve a direction change, into decision points that do not require a change of action. It practically turns them into decision points without a (conceptualized) change of direction.

To account for these principles, a suitable optimization criterion is to aim at a minimal number of distinct parts with, everything else being equal, abstract turn instructions at the coarsest granularity levels possible. Using this criterion, context-specific route directions consist of a small number of chunks whose elements are high-level instructions. Such route directions comply with the principles previously detailed. In Section 4.5, I suggest additional possible optimization criteria that result from integrating the route direction principles presented in Section 2.1.5.

4.4.4.2 Optimization Process

The optimization process generates context-specific route directions applying an optimization criterion. It is the fourth step in Guard. It selects the best choice for each decision point, i.e. the abstract turn instruction most appropriate according to the chosen optimization criterion. This involves the selection and sequencing of the chunks generated beforehand in such a way that they cover the complete route. The chunks need to provide (explicit or implicit) instructions for all decision points between origin and destination, and they must best fulfill the optimization criterion. In other words, after the optimization step, the resulting route directions need to be correct and complete. The algorithmic realization of this step is described in Section 5.4, as well as a discussion of two different optimization processes: global and local optimization.

4.4.5 Resulting Context-Specific Route Directions

The optimization process results in a sequence of chunks. These chunks provide abstract turn instructions that can be considered to be on a coarse level of granularity.

The chunks are represented as tuples consisting of a sequence of decision points and a sequence of their accompanying direction relations. An example for such a chunk is (($\mathtt{DP}_3$, $\mathtt{DP}_4$, $\mathtt{DP}_5$), (`straight`, `straight`, `left`)). Thus, even though the chunked abstract turn instructions are on a coarser level of granularity than the instructions for single decision points, on the abstract representational level they still contain all the information of the underlying single decision points' abstract turn instructions (this relates to 'zooming into' instructions in Dale et al., 2005). Keeping all information on the representational level is advantageous. It offers flexibility with respect to the externalization process and ensures that such a process can access all needed information. For externalization, it is possible to define a parser that transfers the chunks into presentable instructions, i.e. to turn the abstract turn instructions into either a verbal or graphical presentation of a route direction. This externalization then reduces the presented information according to the performed chunking exploiting the means offered by the medium (e.g., language). For the example chunk given above, such parsing may, for example, result in a verbal instruction "turn left at the third intersection."

4.4.6 Summary of Guard

The process for generating context-specific route directions is called Guard (short for *Generating Unambiguous, Adapted Route Directions*). It consists of four steps: *extraction*, *chunking*, *postprocessing*, and *optimization*.

The extraction step generates every possible abstract turn instruction for each decision point; this may result in several ATIs applicable for a single decision points. From these, the one that is best suited for easing the route's conceptualization needs to be chosen. This cannot be done for a single decision point in isolation; the chunking step combines several single instructions into one, which reflects an important mechanism in conceptualizing routes. The postprocessing step ensures that these chunks are sensible, i.e. adhere to cognitive and representation-theoretic considerations. Based on these chunked instructions, the selection of the best abstract turn instructions is performed in the optimization step resulting in optimal context-specific route directions.

4.5 Integrating Different Route Direction Principles in Guard

In Section 2.1.5, I listed different principles of good route directions that can be found in the spatial cognition literature. These principles can be integrated in Guard; either as post-processing principles in chunking or as (part of an) optimization criterion—or even both. This is further explained in this section by taking up the principles of Section 2.1.5 and relating them to the process presented in the last section.

4.5.1 Principles According to Habel

Habel's (1988) first route direction principle, segmentation at decision points, perfectly fits Guard—the chunks generated by the chunking algorithm segment a route at decision points.

Habel's third principle, describing decision points using landmarks, is also mapped in a straightforward manner in my approach. The abstract turn instructions may contain landmark information to further specify an action at a decision point. This principle also indicates a preference for landmark chunking in generating route directions (see Section 3.1.2). Hence, this may be integrated in an optimization criterion. Chunks employing landmark chunking should be preferred over other chunks. Landmark chunking also implicitly covers Habel's second principle ('go straight if not stated otherwise'). Referring to a landmark at the decision point where a direction change occurs allows for not mentioning the decision points without direction change, i.e. the ones where a wayfinder needs to go straight.

The last principle, i.e. reassurance on long segments, is also partly integrated in Guard (see Sections 3.4.3, 4.2.4). It is possible to generate confirmation annotations. This information then needs to be taken into account in an externalization process to provide reassurance to the wayfinder.

4.5.2 Principles According to Denis

Denis (1997) postulates five route direction principles. The first principle claims a limited number of statements should be used in route directions. This can be transformed into an optimization criterion such that the number of chunks in a route direction is restricted to some maximum value.[5]

One of Denis' main principles calls for an explicit association of actions and landmarks. This is easily made possible with the chosen structure of abstract turn instructions. It also hints at a preference for landmark chunking, with the same consequences as discussed for Habel's principles. As Guard adopts a functional perspective in route directions (cf. Klippel, 2003a), and may be extended to select landmarks according to their saliency (Section

[5] This value probably needs to be defined relative to the number of decision points.

7.2), the landmarks employed are consequently visible and relevant, fulfilling Denis' principle of reference to visible, relevant landmarks. The route directions generated with the presented process are a chunked sequence of decision point/action pairs. Just one piece of information for each decision point is communicated and no decision point is represented more than once. This avoids redundancy and drastically limits opportunities for overspecification; this covers the second principle of Denis.

Denis' fourth principle, a preference of determinate over indeterminate descriptions, does not fit well in GUARD as only landmarks used as end qualifiers are located relative to another landmark. Here, it would be possible to store additional information in order to turn this into a determinate description. But this is not included, and it is not clear whether this would improve the resulting route directions.

4.5.3 Principles According to Lovelace et al.

Some of the principles listed by Lovelace et al. (1999) directly map onto generating context-specific route directions: GUARD employs landmarks at decision points, indicates (unambiguously) which direction to take at a decision point, does not provide redundant information, and the generated route directions are presented in a linear fashion. This covers the second, sixth, eighth, and ninth principle of Lovelace et al. (1999).

The other principles are not supported. Currently, context-specific route directions do not provide information for error recovery or "you've gone too far, if" statements. These could be integrated in the same way as confirmation annotations. However, additional research is needed to elicit in which situations and how often such information should be provided. Without proper heuristics, the generated directions might be cluttered with such information, making them useless. Distances between decision points are also not represented since the underlying representation used in GIBO is a sequence of decision points abstracting from the route-segments, i.e. deliberately ignoring this distance information.

4.5.4 Principles according to Dale et al.

The approach to generating route directions of Dale and colleagues (cf. Dale et al., 2003, 2005) introduces two different chunking principles: these are termed landmark-based and path-based segmentation (see Section 3.1.2).

Landmark-based segmentation can be performed just as Klippel et al.'s (2003) landmark chunking, which is explained in the next subsection. Path-based segmentation relies on a threshold for deciding when to end the generation of the current chunk and to begin generating a new one. It is possible to calculate this threshold for each chunk and prune chunks accordingly in GUARD's postprocessing step. It seems more sensible, though, to account for this threshold already in the generation of the initial chunks, i.e. to provide a new chunking algorithm that chunks abstract turn instructions until the threshold is reached. The threshold could also be used as part of an optimization criterion, which could be another option for employing path-based segmentation in Guard.

4.5.5 Principles according to Klippel et al.

Klippel et al. (2003) present three different chunking principles: numerical, structure, and landmark chunking (see Section 3.1.2). These chunking principles can be employed in Guard in a straightforward manner.

In landmark and structure chunking, chunks end with a direction relation that is further annotated with a landmark or structural element. In numerical chunking, chunks have a maximum length, i.e. can cover at most a specific number of decision points. This can be easily turned into a post-processing principle: a chunk is valid if it either ends with a landmark or structural element, or has at most the defined maximum length. This way, it is possible to cover all three kinds of chunking with a single principle. This principle can further be refined to account for different maximal lengths depending on the kind of relations combined in a chunk as they have been defined in Klippel's theory of wayfinding choremes (cf. Klippel, 2003b).

For direction relations that represent references to linear, areal, or global landmarks, the principle needs to include a test for an end qualifier. These chunks need to end with an abstract turn instruction that contains such an end qualifier (see Section 3.4.3).

4.6 Summary

In this chapter, I presented Guard, a process for generating context-specific route directions. The process is based on abstract turn instructions which are relational representations of the actions to be performed at decision points. Abstract turn instructions are the conceptual elements used in the generated instructions. The relations instantiate the systematics' elements (see Section 3.4), i.e. determine how to refer to these elements in route directions.

Generation of context-specific route directions is realized as an optimization process. Guard consists of four steps. First, for each decision point every applicable abstract turn instruction gets generated. Next, based on these single abstract turn instructions, all possible chunks are determined. The initial chunks are then adapted to match the chosen chunking principles in a post-processing step. Finally, the optimization step chooses those (chunked) abstract turn instructions that lead to an optimal route direction. Optimality depends on the chosen optimization criterion: I suggest to aim for a minimal number of distinct parts with route directions on the coarsest granularity level possible. GUARDallows for the integration of different route direction principles as they have been identified in the spatial cognition literature in a straightforward manner.

Chapter 5

Generating Context-Specific Route Directions

This chapter details the algorithmic realization of the generation of context-specific route directions. It covers the underlying spatial representation (Section 5.1), and GUARD—the generation process—itself (Sections 5.2 to 5.4).

In a nutshell, GUARD works as follows. In a first step, for all decision points of the route, all possible abstract turn instructions are determined (Section 5.2). Based on these sets of abstract turn instructions, a chunking process tries to group sequences of decision points into single instructions (Section 5.3.1), which in turn are tested in a postprocessing step with respect to their validity according to the chosen chunking principles (Section 5.3.2). Finally, using the generated chunks an optimization process generates the context-specific route directions (Section 5.4).

5.1 Input: The Underlying Spatial Representation

The automatic generation of context-specific route directions requires a representation of the environment that contains all information needed and allows computing a route from some origin to a destination (Section 5.1.1). I employ an embedded graph representing the environment's path-network (Section 5.1.2). The graph reflects the layout of the environment's path-segments—it preserves information on angles between branches and on distances. Calculating a route in such a graph results in a sequence of nodes that need to be traversed to get from an origin to a destination (Section 5.1.3). Such a sequence of nodes annotated with additional information on, for example, landmarks, is the underlying representation used in GUARD.

5.1.1 Spatial Data

The spatial data used for generating context-specific route directions can be any geographic data set used for Geographic Information Systems (GIS) that fullfills the following prerequisites. First, the data needs to contain information on a path-network. This is crucial for generating the graph. Second, geometric information on the represented features, i.e. coordinates determining their shape and (relative) location in the represented environment, is needed.

The data that is represented in a specific data format, for instance ATKIS (*Amtliches Topographisches Kartographisches Informationssystem*; AdV, 2003), is converted to an internal data structure based on several feature classes. Extraction and conversion of such geographic data is not specifically implemented for GUARD, but has been realized as part of a schematization toolbox developed in project I2-[MapSpace] of the SFB/TR 8 Spatial Cognition (e.g., Neumann, 2003).

5.1.2 Annotated Graph

The spatial data used as input is already an abstraction from the real world. It is further abstracted to a graph representation. Nodes and edges are annotated with information about landmarks. An elevation model may be represented by associating elevation information with the nodes.

Nodes in the graph that have a degree greater than two represent intersections of the street-network. Nodes with a degree of two reflect the geometric structure of the street-network; they represent bends and curves in the streets (see Fig. 5.1). Following Klippel's terminology (see Section 1.1), nodes representing intersections are branching points on the structural level and decision points on the functional level. The generation of context-specific route directions focuses on these nodes. Route-segments are represented by a sequence of edges connecting all two degree nodes between two decision points. On a geometric level, these edges form a polygonal line, or *poly-line* for short. To enable geometric computation, a coordinate is attached to each node.

Point landmarks are represented as points located at specific coordinates. This allows for calculating their distance to a given node or to a route-segment, for example. Linear and areal landmarks are represented by poly-lines consisting of a sequence of coordinates. A global landmark is represented by a relation denoting the cardinal direction to the landmark from the environment. Path annotations represent signage and are denoted as labels. These labels are attached to the edges that represent the streets that are marked by the represented signage in the real world. In the following, I will use the terms of the represented world if it is clear to which world I am referring; nodes representing decision points are referred to as decision

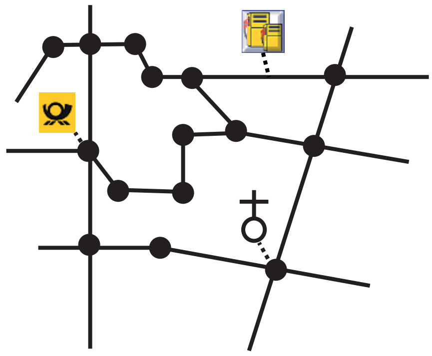


Figure 5.1: Example of an annotated graph as used in GUARD. Nodes with a degree greater than two represent intersections; nodes with a degree of two represent curves. The graph is annotated with pointers to landmarks, indicated by the dotted gray lines from the church and post office symbols to their neighboring decision points, and the gas station symbol to the closest edge.

points for short, for example.

5.1.2.1 Modeling a Decision Point

I model the structural aspects of wayfinding in an environment as a graph. Functionally, however, the configuration of branches meeting at a decision point needs to be considered. This corresponds to human mental conceptualization. The two functionally relevant path-segments, the route-segments, are part of the decision point. This also reflects an action-oriented aspect: people usually do not turn on the spot like some robots do, but rather in an extended process, by first deciding on the direction to turn and then changing direction gradually while keeping their forward movement (cf. Tversky & Lee, 1999; Klippel, 2003b).

I abstract from this gradual direction change in determining abstract turn instructions. I model decision points such that all path-segments that belong to the decision point meet in a single point (see Fig. 5.2). The two functionally relevant branches are termed *incoming route-segment* and *outgoing route-segment* (cf. also Richter & Klippel, 2007).

5.1.3 Determining a Route

For determining a route, any graph-based path-search algorithm can be used, for instance A* or Dijkstra's (1959) shortest-path algorithm. These algorithms result in a sequence of nodes that need to be passed to get from origin to destination. They calculate the path between two nodes based

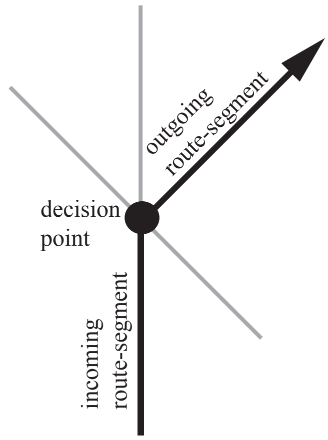


Figure 5.2: A decision point with two functionally relevant branches: the incoming and outgoing route-segment. The branches' meeting point is the actual decision point.

on a weight measure used to compare nodes with respect to which node to include in the path. As has been presented in Section 2.3.4, this measure does not need to be a spatial measure like euclidean distance. For example, in determining a route in a graph, the edge weight might be the (assumed) complexity of an intersection (cf. Duckham & Kulik, 2003). Other possible weights might be the number of nodes passed, thereby concentrating only on the decision points, euclidean direct-line distance between decision points, or movement distance taking bends and curves into account. In the following, movement distance is chosen as edge weight to account for the distance actually covered by a wayfinder. In my thesis, I concentrate on finding the best route directions for a given route; Section 7.2.5 discusses how my approach may be used to find 'simplest routes' in line with Duckham and Kulik's approach.

5.1.4 Visibility

Visibility plays an important role in applying many of the systematics' elements in generating instructions. Application of distant landmarks and global landmarks depend on their visibility from a decision point; the same holds for linear landmarks and even for point landmarks at a decision point. Calculating their visibility in a city-like environment requires a detailed three-dimensional model of the environment and complex algorithms (e.g., ray-based). In order to decide which features are applicable, position and view-direction of a wayfinder needs to be modeled and all features visible from this position need to be identified—based on a model of the wayfinder's field of vision.

My work is set in the context of spatial cognition and its focus is on cognitive principles for, and the generation process of, route directions. It is not a thesis on algorithms for computer graphics and geometry. Therefore, information on whether some feature, for instance a distant landmark, is visible from a given decision point is assumed to be known.

5.2 Extraction

Different elements of the systematics of route direction elements require different methods to determine whether they are applicable for the decision point at hand. How to determine abstract turn instructions for all elements of the systematics is explained in the following.

5.2.1 Level of Global References

On the level of global references, cardinal directions and global landmarks may be used in route directions.

Cardinal Directions It is assumed that the bearing of the environment with respect to cardinal directions is known. Therefore, it is always possible to determine the cardinal direction of the route-segment to take at a decision point. To that end, the angle of the corresponding graph's edge with respect to a reference direction (e.g. 0 as north), is calculated and matched to a qualitative value denoting the cardinal direction. These values are represented as angle intervals based on homogeneous sector-based direction models (cf. Section 2.2.3), which represent cardinal directions on different levels of granularity. An example for such a matching is shown in Figure 5.3.

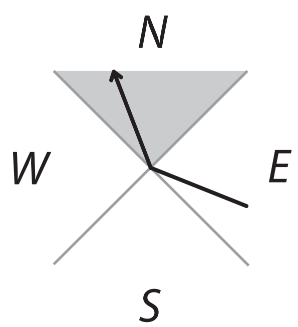


Figure 5.3: Determining cardinal directions: matching a branch's angle to a sector of a qualitative direction model.

As cardinal directions are a coarse classification, often several branches are in the same interval. To resolve ambiguities resulting from this coarse classification, a finer direction model can be used. However, in pratical terms, for the wayfinder this is only helpful to a small extent. Without using a compass, a wayfinder hardly can distinguish ‘north-northwest’ from ‘northwest’ or ‘west-northwest’. Therefore, resolving ambiguity on the representational level by refining the classification results in ambiguity of determining the intended direction in the real world. Accordingly, cardinal directions are mostly used in providing coarse overview information on a route (cf. Sections 7.2.1).

Global Landmarks According to the systematics, global landmarks can be referred to if they are either in the same, or in the opposite direction of, the movement direction of the wayfinder. The direction to a global landmark is determined using a four-sector cardinal direction model (see Section 2.2.3); the same model is used to determine the cardinal heading of the outgoing route-segment. If these two directions match, or one is the opposite of the other,[1] a reference to the global landmark may be possible.

In a second step, it is checked whether such a reference is unambiguous. To this end, the cardinal heading of all other branches of the decision point—except for the incoming route-segment—is determined. If none of these branches head in the same direction as the outgoing route-segment an abstract turn instruction referring to a global landmark is generated and stored in the set of possible abstract turn instructions for this decision point.

5.2.2 Level of Environmental Structure

On the level of environmental structure, edges as defined by Lynch (1960) (for instance highways dividing an environment into two parts) and slants, may be used in abstract turn instructions.

Edges Edges are mostly used in generating overview information on a route (see Section 7.2.1). But they may function as distant or linear landmarks. Accordingly, generating abstract turn instructions referring to edges rely on the same methods used to determine references to distant landmarks and to linear landmarks, respectively. These methods are detailed below.

Slant In order to determine a slant an elevation model is needed. Using such a model, generating abstract turn instructions based on a slant is simple; the elevation difference between the end-point of the outgoing branch and the current decision point is calculated. If this difference is bigger than a certain threshold, the slant can be used to indicate the direction to take.

[1] “Opposite” in cardinal directions: `north` ↔ `south`; `east` ↔ `west`

An abstract turn instruction is generated that employs, as direction relation, either `uphill`—in case of a positive difference—or `downhill`—in case of a negative difference. Additionally, it needs to be checked whether other branches head in the same vertical direction which would make such an instruction ambiguous. The threshold is used to reflect that humans only notice a slant if it exceeds a certain slope.

5.2.3 Level of Path and Route

While determining egocentric references is a simple process, calculating references to different kinds of landmarks requires some more considerations. The latter will cover most of this section.

Egocentric References As with cardinal directions, determining egocentric references is based on matching an angle to an interval representing a qualitative direction. With egocentric references, the angle between the incoming and outgoing route-segment is under consideration. These two segments are always present at a decision point; therefore, calculating egocentric references is always possible. The heterogeneous direction model presented in Klippel et al. (2004) is used for this matching (see Section 2.2.3). Figure 5.4 shows an example for the matching process. The direction model has been adapted to incorporate a small sector of 6 degrees representing direction 'straight' instead of an axis as in the original model. This is in line with the discussion of axes vs. sectors in Haque et al. (to appear) and also accounts for problems of imprecision (e.g., rounding errors) resulting from calculating with real numbers.

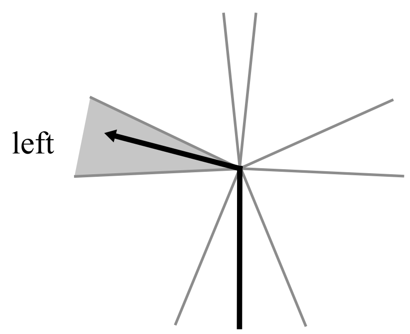


Figure 5.4: Determining an egocentric reference: matching the angle between route-segments at a decision point to a sector of a qualitative direction model.

To check for ambiguities, egocentric references to all other branches of the decision point get calculated. If a reference to any other branch is equal to the one of the outgoing route-segments, this is an ambiguous situation.

In this case further refinements need to be provided. This can be done by referring to a landmark (see next paragraph). Or the situation may be disambiguated by counting the branches, for example, in instructions like "take the second to the right."

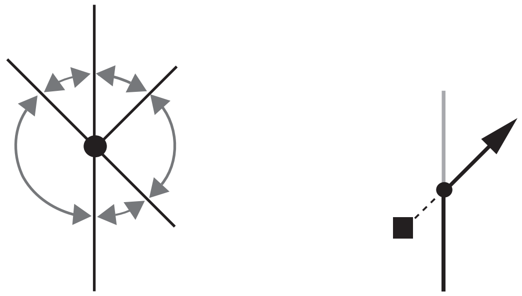
References to T-intersections are determined by evaluating the intersection's configuration. First, exactly three branches need to meet at the intersection. Second, the angle between the incoming route-segment and the other two branches needs to be around 90 degrees (a deviation of 15 degrees).

Landmark at Decision Point The relations $\texttt{lm}^{<}$, $\texttt{lm}^{>}$, and $\texttt{lm}^{-}$ are used to represent a landmark's location at a decision point (see Section 4.2.3). In order to determine which of these relations holds, we need to know a landmark's location relative to the functionally relevant branches (see Fig. 4.1 on page 77). If $\texttt{lm}^{<}$ holds, i.e. a landmark is passed *before* the decision point, it must be *next to* the incoming route-segment. If it is *next to* the outgoing route-segment, $\texttt{lm}^{>}$ holds and the landmark is passed *after* the turning action took place. If a landmark is neither next to the incoming nor the outgoing route-segment, it is not at a functionally relevant branch; denoted by $\texttt{lm}^{-}$. The landmark is *at* the decision point, but its position with respect to the route-segments cannot be further specified.[2]

To determine the location of a landmark, ordering information can be exploited. The branches of a decision point form a circular ordering as they are neighbors based on their spatial position—with the last branch in the ordering being neighbored to the first one (see Fig. 5.5a). The ordering can be calculated by determining the angle of a branch relative to some reference direction, for example north representing a bearing of 0, and then sorting all branches according to their direction angle. By creating a new virtual branch ranging from the decision point to the landmark in consideration, a landmark's location is introduced to the branches' ordering (see Fig. 5.5b). The ordering is recomputed and the landmark's location is determined with respect to the newly established ordering.

By modeling a decision point as the point where incoming and outgoing route-segments meet (see Fig. 5.2), the area around the decision point gets divided into a *region before action* and a *region after action* as a wayfinder's direction of motion induces a directedness of the functionally relevant branches. These regions are termed before- and after-region for short. The landmark needs to be located in the before-region for the relation $\texttt{lm}^{<}$ to hold and in the after-region for the relation $\texttt{lm}^{>}$ to hold. To further delimit the before- and after-region, I introduce two additonal virtual branches to the ordering. These virtual branches start at the deci-

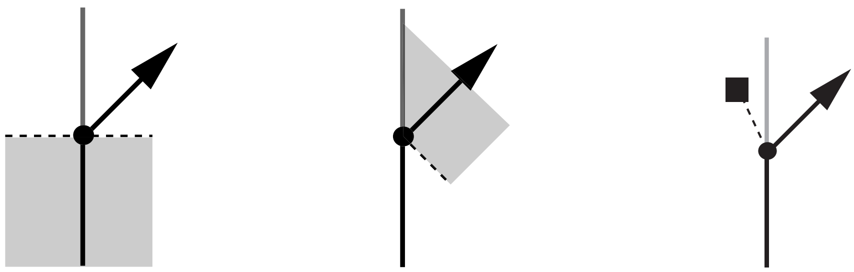
[2]The approach to determining these relations has been presented in Richter & Klippel (2007).



a) b)

Figure 5.5: a) An intersection with five branches. The arrows indicate the neighborhood relation between the branches. The branch pointed at is a neighbor of the branch the arrow starts from. This relation is symmetric. b) A virtual branch connects a landmark to the decision point, introducing this landmark into the branches' order.

sion point and head perpendicularly to the functionally relevant branches to the left and right relative to these branches (Fig. 5.6a,b). That is, to create the before-region, two virtual branches perpendicular to the incoming route-segment are introduced; their calculation is straightforward. For the after-region, this is done correspondingly (this is comparable to the axes demarcating front/back in the double cross calculus; cf. Freksa, 1992). These virtual branches are needed to ensure that a landmark that is located next to a functionally relevant branch is indeed in the corresponding region. Figure 5.6c illustrates this necessity: here, a landmark is next to the incoming route-segment in the ordering, but located after the decision point.



a) b) c)

Figure 5.6: Regions of a landmark's possible location relative to the functionally relevant branches: a) before the turning point, b) after the turning point. c) Example of a situation in which a landmark is next to the incoming route-segment, but the landmark is located after the decision point.

The branches' ordering establishes a neighborhood relation between the branches. A landmark is next to a functionally relevant branch if it is a neighbor of this branch in the ordering. This allows for the determination of the reference relation to be used for a landmark at a decision point. If the landmark's virtual branch is a neighbor of the incoming route-segment, $\texttt{lm}^{<}$ holds; if it is a neighbor of the outgoing route-segment, $\texttt{lm}^{>}$ holds. The virtual branches demarcating before- and after-region ensure that the functionally relevant branch and a landmark's virtual branch cannot be neighbors if the landmark's location is after (in case of the before-region) or before (for the after-region) the decision point. Put differently, if they are neighbors, route-segment and landmark are on the same side of the decision point as required by the definition of before- and after-region. Figure 5.7 shows examples for determining a landmark's position at a decision point using this approach. The virtual branches denoting the before-region are introduced to the ordering when checking whether $\texttt{lm}^{<}$ holds; the virtual branches of the after-region when checking for $\texttt{lm}^{>}$. It is not possible to introduce both sets simultaneously as this may result in overlaps and, hence, unwanted restriction of the two regions (see Fig. 5.7c).

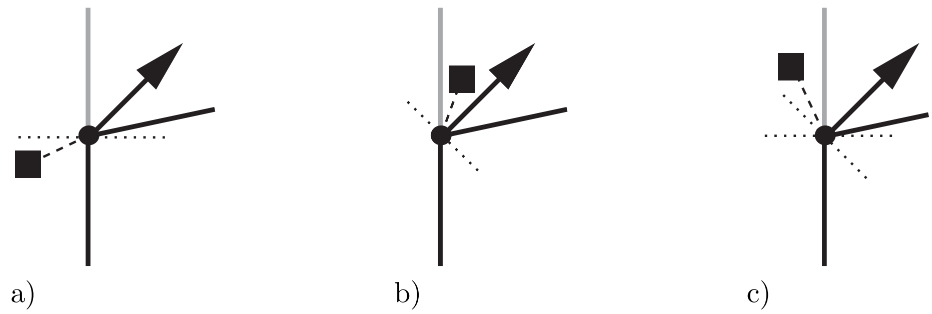


Figure 5.7: Examples of a decision point's configuration and the resulting landmark's reference relation: a) $\texttt{lm}^{<}$, b) $\texttt{lm}^{>}$, c) $\texttt{lm}^{-}$.

If incoming and outgoing route-segments are neighbors in the circular ordering, the before- and the after-region may overlap. In this case both $lm^{<}$ and $lm^{>}$ are valid results for determining the landmark's location at the decision point. However, in configurations as shown in Figure 5.8, the preposition 'before' is not applicable, as it does not conceptually cover the situation model. It is, therefore, never used. Hence, $lm^{<}$ represents this configuration and $lm^{>}$ is excluded.

For relation $lm^{>}$ we need to consider a further restriction. The corresponding preposition '[turn] before' is the most restricted in its semantic scope. To be applicable for identifying the branch to take the most constraints need to be fulfilled. To indicate the correct branch, the landmark needs to be left of the outgoing branch when turning into the right half-plane and right for turning left (see Fig. 5.9). Otherwise, references to a landmark

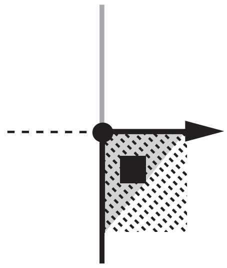
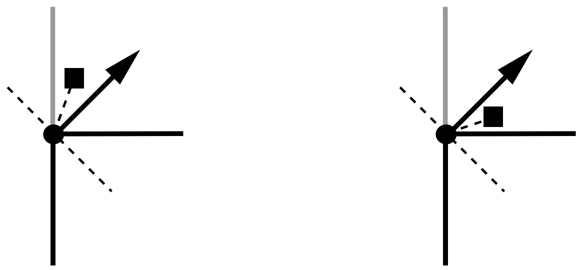


Figure 5.8: Neighboring incoming and outgoing route-segments. Here, the before-region (the gray area; shown only partially for better readability) and the after-region (the dashed area) may overlap.

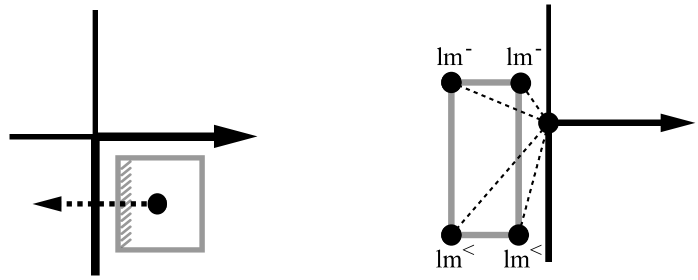
using the preposition 'before' may mislead the wayfinder. In Figure 5.9b the default assumption would be to take the branch heading off 90 degrees to the right as this branch is directly in front of the landmark in movement direction. However, the intended branch is the one heading off 45 degrees to the right. The order of branches and landmark is calculated relative to some reference bearing. Determining whether a landmark is left (or right) of the outgoing branch is straightforward. In this case, the landmark is predecessor (or successor) of the outgoing branch in the ordered sequence of entities under consideration.



a) b)

Figure 5.9: "Turn right *before* landmark": a) landmark correctly indicating outgoing branch; b) ambiguous situation where using the preposition 'before' may mislead wayfinder.

For some features of the environment their landmark character may be dominated by a specific part that may not be visible from everywhere around the feature. This holds, for example, for buildings housing shops that have a salient store-front (cf. Raubal & Winter, 2002). Such landmarks are represented as point landmarks; the salient side (the façade) is represented by a vector pointing in direction of the salient side (see Fig. 5.10a). To generate references to such landmarks, the location of the landmark at the decision point is determined as explained above. Then the vector representing the façade is intersected with the functionally relevant branches. If $\texttt{lm}^<$ holds, the vector needs to intersect the incoming route-segment to be applicable. In this case, the landmark is next to the incoming route-segment and the façade is visible from that route-segment. Correspondingly, if $\texttt{lm}^>$ holds, it needs to intersect the outgoing route-segment.



a) b)

Figure 5.10: Extending the approach for determining a landmark's relation to a decision point: a) for representing façades (the hatched area in the rectangle) the point representing the landmark is associated with a vector heading in the façade's direction; b) a landmark that cannot be abstracted to a point. Here, for each corner point the relation to the decision point gets determined; in this example the resulting relation is $\texttt{lm}^-$.

Based on the approach for determining the location of a point landmark at a decision point, landmarks that are represented by more than one point can be handled. For these landmarks, the relation of every coordinate point to the decision point at hand is determined. Patterns of these single relations define whether a certain relation for referencing the landmark is applicable. To check whether a certain relation holds for the landmark, the resulting sequence of relations that hold for the single points is compared to the pattern that corresponds to the relation.

For example, it may not be possible to abstract a landmark at a decision point to a single point (see Fig. 5.10b). In this case, the relation of all coordinate points forming the landmark's shape to the decision point are determined. For $\texttt{lm}^<$ to hold for the landmark, $\texttt{lm}^<$ needs to hold for every coordinate point; for $\texttt{lm}^>$ this applies correspondingly. If different relations hold for different points, especially if $\texttt{lm}^-$ holds for any of the points, the relation of the landmark to the decision point is $\texttt{lm}^-$.

Linear Landmark Linear landmarks are represented as a sequence of coordinate points. Several of these points may be near a decision point and, hence, need to be considered in determining a linear landmark's functional role at that decision point. For an object to function linearly at a decision point, one part of the coordinate points needs to be next to the incoming route-segment; the other part needs to be next to the outgoing route-segment. A linear landmark can then be linked to both route-segments and this way be used to indicate the direction to take. The pattern representing this situation is as follows: the first l coordinate points of the linear landmark need to be in the before-region and the remaining m points need to be in the after-region of the decision point ($l, m \geq 1$; see Fig. 5.11).

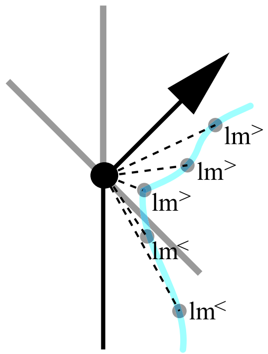


Figure 5.11: Determining whether a linear landmark functions linearly: checking for each coordinate point of the landmark its location relative to the decision point.

The process checking for a linear landmark's applicability sequentially determines the relation of each coordinate point to the decision point. For the first point, the relation must be $\texttt{lm}^<$. As long as the resulting relation is $\texttt{lm}^<$, the process continues until there is a switch to the relation $\texttt{lm}^>$. From now on, all the remaining relations need to be $\texttt{lm}^>$. As soon as any resulting relation is $\texttt{lm}^-$, or there is a switch back to $\texttt{lm}^<$, the process stops. In such cases, the object does not function linearly. It can still be referred to as landmark at a decision point with $\texttt{lm}^-$, denoting its location relative to the decision point. If the process finishes the calculation with the last coordinate and the resulting relation is $\texttt{lm}^>$, the object functions linearly. The direction relation `follow` denotes this situation.

Areal Landmark Determining references to areal landmarks is done similarly as for linear landmarks. However, the difference is in assigning landmark types in the abstract turn instructions. Each coordinate point defining the landmark's geometry is assigned to one of the route's decision points surrounding the areal landmark. For all points assigned to a decision point, it is checked whether there is a single switch from $\texttt{lm}^<$ to $\texttt{lm}^>$ occuring between any consecutive points. If this is the case, `follow` is applicable for the areal landmark at this decision point. If not, $\texttt{lm}^-$ holds, i.e. there is a turn *at* the object.

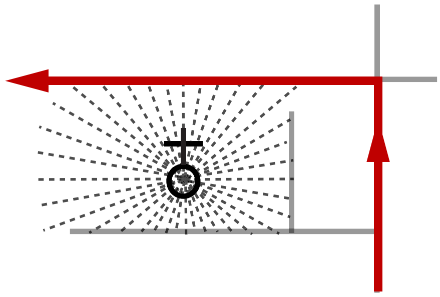


Figure 5.12: Determining visibility of a landmark between decision points. A radial sweep-line is used to check whether the poly-line representing the connecting route-segment is visible from the landmark.

Landmark Between Decision Points Landmarks between decision points need to be near to, and visible from, the route to be referred to in route directions. Nearness is checked by calculating the distance to the poly-line representing the route-segment. If the distance is below a threshold, the landmark is near the route-segment. It is also checked whether the landmark is in the area of one of the decision points. Such landmarks are used for references to landmarks at a decision point; they may not be employed in instructions referring to landmarks between decision points.

Visibility from the route is tested using a radial sweep-line algorithm with the landmark as base point (see Fig. 5.12). If any of the lines forming the poly-line are visible, then the landmark is visible from the route. In this case, an abstract turn instruction is generated employing the direction relation `towards` and the landmark's label.

Distant Landmark Determining abstract turn instructions referring to a distant landmark works the same way as determining references to a global landmark (see Section 5.2.1). However, instead of a four-sector model, an eight-sector model is used to better account for the landmark's size and smaller distance to the route.

Path Annotation Street signs and street names indicate the further direction to take by pointing to, or being located at the street to take, at an intersection. In the graph, such signage is represented by path annotations. To determine an abstract turn instruction based on a path annotation, it is checked whether such an annotation is present at a decision point, and whether the outgoing route-segment is annotated with the path annotation.

5.2.4 Confirmation Annotations

Two kinds of confirmation annotations are used in context-specific route directions: references to passing a landmark and references to crossing a linear landmark. Determining confirmation annotations referring to passing a landmark is based on the same method used in determining references to landmarks between decision points. This method results in candidate landmarks for the confirmation annotation. For each candidate, an annotation is generated: the direction relation `pass` is combined with the landmark's label, and the landmark's location relative to the route-segment. The relation `left` is used if the landmark is left of the route-segment, `right` otherwise.[3]

Confirmation annotations referring to crossing a linear landmark are calculated based on geometric configurations. If a linear landmark cuts the route-segment between two decision points, it is crossed while moving from one decision point to the other. This can be determined by calculating the point of intersection between the poly-line representing the landmark and the poly-line representing the route-segment. If such an intersection point exists a confirmation annotation is generated—`cross` combined with the edge's label.

These two methods potentially generate many possible confirmation annotations for a decision point's abstract turn instructions. Deciding which annotation to use is part of the optimization process detailed in Section 5.4.

5.3 Chunking

Chunking is a crucial part of GUARD (see Section 3.1.2). Chunking is realized as a two-step process. In an initial step, simple syntactic grouping is performed (Section 5.3.1). The resulting chunks are then adapted to the chosen chunking principles (Section 5.3.2).

5.3.1 The Initial Chunking Process

The inital chunking process is simple, but powerful. It is based on syntactic properties of the abstract turn instructions. Starting off with an abstract turn instruction of a given decision point, it tries to chunk this with as many of the following decision points as possible. This is done iteratively. The current ATI is compared to every possible ATI of the next decision point. If it is chunkable with one of these ATIs, the next decision point is added to the chunk. The matching ATI of the next decision point becomes the current one; chunking continues with this instruction. This is repeated until there is no match, i.e. an abstract turn instruction is not chunkable with any of those of the subsequent decision point.

[3]If a candidate landmark is already used in an instruction for the decision point at hand, it cannot be used again as a confirmation annotation.

Two cases are sufficient to determine whether two abstract turn instructions are chunkable. Two ATIs match if they either share a common direction relation, or the first contains `straight` as a direction relation and the second an egocentric direction relation. In the first case—chunking equal direction relations—the comparison needs to be extended to include annotations, i.e. references to landmarks or structural elements that may be part of an abstract turn instruction. These annotations are an integral part of abstract turn instructions and are carried over to the chunk to become part of the route directions; they cannot be ignored. If both ATIs have different annotations, i.e. refer to different features they are not chunkable. If there are no annotations, or just one ATI is annotated, they can be chunked. If both ATIs refer to the same feature, chunking is also possible. In case of a reference to a landmark at a decision point, the prepositions used to further specify this landmark's position are additionally checked to not be in the order $\mathtt{lm}^>, \mathtt{lm}^<$.[4] Algorithm 1 summarizes the chunking process.

Algorithm 1 The chunking process.

function chunking
  Store current ATI in chunk.
  Set chunking flag to true
  while chunking flag is true **and** finished mark not set **do**
    Pick next DP as current DP.
    {Rule equal direction relations:}
    if two consecutive direction relations are equal **and** (there are no annotations **or** only one ATI has annotation **or** (both have equal annotation **and** preposition combination is not $\mathtt{lm}^>, \mathtt{lm}^<$)) **then**
      add second ATI to chunk
      {Rule egocentric references:}
    else if first relation is `straight` and second relation is egocentric direction relation **then**
      add second ATI to chunk.
      Set finished mark.
    else
      Set chunking flag to false {no chunking possible}
    end if
    Store chunk in set of generated chunks.
  end while
  Return set of generated chunks.

[4]This check is made to make the process failsafe; it would not be necessary. Such a preposition combination would—from a functional perspective—correspond to an impossible spatial situation. A preposition combination $\mathtt{lm}^>, \mathtt{lm}^<$, referring to the same landmark, would represent a sequence of actions like "Turn left after the church, then turn left before the church."

As an example, consider the following sequence of decision points with their sets of abstract turn instructions:

1: `(straight, follow / river)`

2: `(left, follow / river)`

3: `(veer right, downhill, follow / river)`

The first abstract turn instruction of the first decision point (`straight`), is chunkable with the second decision point, according to the rule of chunking egocentric references. The first ATI's direction relation is `straight`. The second decision point's set of ATIs contains an abstract turn instruction using an egocentric reference (`left`). According to the employed rule, chunking ends with combining these two relations. The resulting chunk is $((\mathtt{DP}_1,\mathtt{DP}_2),(\mathtt{straight},\ \mathtt{left}))$. This corresponds to an instruction like "turn left at the second intersection." The second abstract turn instruction of the first decision point (`follow / river`), is chunkable with decision points 2 and 3, according to the rule of chunking equal direction relations. All three decision point's sets of ATIs contain an element employing a common direction relation (`follow`) which is annotated with the same landmark (`river`). Accordingly, the resulting chunk is $((\mathtt{DP}_1,\mathtt{DP}_2,\mathtt{DP}_3),(\mathtt{follow\ /\ river,\ follow\ /\ river,\ follow\ /\ river}))$; corresponding to something like "follow the river." This chunk could still be extended, but in the small example provided here, it already covers all decision points of the route.

5.3.2 Postprocessing

The simple chunking algorithm presented above may result in chunks that are nonsensical or even useless. The algorithm only performs syntactical chunking and does not account for any semantic information or context. Therefore, the chunks need to be postprocessed in order to ensure their validity. Algorithm 2 summarizes this process. Dividing the generation of chunks and the check for validity into two separate processes allows for the easy integration of different principles of either chunking or postprocessing. For example, for postprocessing the route direction and chunking principles presented in Section 4.5 may be employed.

The chunk at hand is tested for its validity according to the chosen postprocessing principles. If the chunk is invalid, the chunk's last decision point is removed and the truncated chunk is checked again. This is done until either the chunk is valid or all decision points have been removed. Removal of all decision points indicates that the combination of abstract turn instructions that has been produced by the chunking algorithm is not valid in the context of the current chunking principles. For example, references to linear landmarks require an end qualifier, i.e. an additional landmark that denotes

Algorithm 2 The postprocessing process.

```
function postprocess-chunk
  if chunk is invalid or not empty then
    if chunk is invalid then
      remove last decision point from chunk.
      postprocess-chunk chunk.
    else
      Return chunk.
    end if
  else
    Return chunk.
  end if
```

until which point along a route this linear landmark should be followed (see Section 3.4.3). If none of the abstract turn instructions that are combined in a chunk provide such a qualifier, the chunk will be removed in the postprocessing step. This is what happens to the chunk `(DP1,DP2,DP3),(follow / river, follow / river, follow / river)` generated in the small example above. Here, none of the abstract turn instructions contain an end qualifier; consequently, the chunk is removed in the postprocessing step.

5.4 Optimization

Generating route directions that cover all decision points between origin and destination is realized as an optimization process (see Section 4.4.4). Different optimization criteria may be used in the optimization process (Sections 4.4.4.1, 4.5). In the following, the optimization criterion argued for in Section 4.4.4.1 will be used in the given examples. The optimization process aims at a minimal number of distinct parts with, everything else being equal, abstract turn instructions on the highest granularity levels possible.

In Section 5.4.1, I present the general characteristics of the optimization process. The process may employ local or global optimization strategies (Section 5.4.2); their algorithmic realization is presented in Section 5.4.3. Section 5.4.4 compares the computational complexity of both strategies.

5.4.1 General Characteristics of the Optimization Process

In this section I point out general characteristics that the optimization process needs to account for. These characteristics arise from the general characteristics of routes.

- The order of decision points must not be changed. Routes are directed and route directions must provide instructions on the actions to be performed in the order of the route. The first instruction of the route directions must entail the first decision point, the last instruction the last decision point, and in-between, the order of decision points needs to be preserved. That is, optimization must not generate a sequence of chunks in an arbitrary order. For example, compare instructions like "go left twice, then follow the tracks until the post office. There, turn right and turn left again at the third intersection" with "follow the tracks until the post office after turning twice left. Turn left again at the third intersection after you turned right at the post office." Even though the latter instructions allow inferring the correct order of executing the instructions, the former are easier to parse as they present the actions to be performed in order of their execution.

- Each decision point must be part of the route. A wayfinder needs to be able to decide on the further direction to take at every decision point. Accordingly, there needs to be an instruction for every decision point—even if it is just implicitly represented. "Turn left at the third intersection" is a valid instruction for a situation in which a wayfinder has to go straight at two consecutive intersections, and to turn left at the third one; "turn left" is not. The former instruction implicitly contains information on the first two intersections; the latter instruction does not.

- No decision point may appear more than once in the route directions. There must be only one instruction per decision point in order to keep the ordered sequence of decision points and to avoid redundancy. Consequently, there may be no overlapping chunks in the context-specific route directions.

- Chunks may have arbitrary length. The different chunking principles may lead to chunks of different length; chunks do not have a fixed size. While the overall length of the solution is known, it covers the complete route from origin to destination, the lengths of its constituting parts cannot be predetermined. The optimization process must be able to deal with partial solutions of different length. For example, there might be two chunks starting with the first decision point: ($\mathtt{DP}_1$,$\mathtt{DP}_2$),(left,left) and ($\mathtt{DP}_1$,$\mathtt{DP}_2$,$\mathtt{DP}_3$),(follow / river, follow / river, follow / river, gas station). Both chunks have different length and either may be part of the optimal solution.

- Choices once made may be revoked later. It is sensible to start generating route directions from the origin, i.e. selecting chunks for the first decision points first. But it must be possible to revoke these decisions later on if it turns out that other chunks that cover decision points already dealt with are better suited. Using the example chunks from above, initally the second chunk (the one with `follow / river`) is chosen. Later on, it might turn out that it is best to choose a chunk starting with $\mathtt{DP}_3$, but this decision point is already covered by the initially chosen chunk. Therefore, that chunk is dropped and the other chunk (the one with `left`) is chosen instead.

5.4.2 Local vs. Global Optimization

Local and global optimization can be distinguished in looking for a solution. In local optimization, only a part of the set of possible solutions is taken into account; in global optimization, the complete set is considered. The generation of context-specific route directions aims at finding an optimal instruction for each route's decision point. Regarding local and global optimization, this means whether to only take into account possible instructions for neighboring decision points or to consider every possible sequence of instructions in choosing an instruction for a given decision point.

Global optimization considers every possible solution to a problem. This guarantees finding a solution that is optimal for a given optimization criterion. On the other hand, global optimization usually is computationally demanding: taking every possible solution into account and evaluating it against all other solutions results in high computational complexity due to the huge solution space.

Local optimization may help to reduce the computational complexity by restricting the search space for solutions to the neighboring decision points. This does not guarantee an optimal solution, but for route directions, it still leads to near optimal results (see Section 6.1). Routes are considered to be an ordered sequence of decision points. Choosing a specific instruction for a decision point mainly has consequences for chunking. That is, this choice determines whether the decision point at hand may be chunked with its neighboring decision points (its predecessor, its successor, or both). Choosing a certain instruction may be more advantageous with regard to chunking than another if the same choice is also possible for several neighboring decision points. Since it is rarely possible to cover a complete route with just one chunk, choices made at the end of the route usually have only little influence on choices made at the beginning. Therefore, results of local and global optimization are usually very similar.

5.4.3 Optimization Algorithms

Both local and global optimization share the same initialization step. In both processes, the first step is to determine for each decision point of a route, all abstract turn instructions that are possible according to the systematics of route direction elements. This results in a set of abstract turn instructions for each decision point (cf. Section 4.2). Based on these sets of abstract turn instructions, optimization is performed as explained in the following.

5.4.3.1 Local Optimization

Local optimization takes into account a subset of all possible solutions. Context-specific route directions are built sequentially in the route's order. For each decision point, the (chunked) abstract turn instruction that is best, according to the optimization criterion, is chosen. For this choice only previous choices made for this decision point are considered. That is, while the process allows updating previous decisions, it does not look ahead.

To illustrate the process of local optimization—and that of global optimization explained below—consider the following example of a route with five decision points. Determining abstract turn instructions for each decision point yields the sets:

- DP_1 : {`(left)`}
- DP_2 : {`(left), (left / church lm`$^<$`), (follow / river)`}
- DP_3 : {`(straight)`, `(follow / river)`}
- DP_4 : {`(straight)`, `(follow / river, gas station)`}
- DP_5 : {`(right)`}

For local optimization, the first element of the set of abstract turn instructions of the first decision point is chosen. Then, all possible chunks for this ATI are generated. The chunking algorithms defined in Sections 5.3.1 and 5.3.2 are applied. The chosen ATI is chunked with ATIs of the following decision points according to the chunking principles at hand. The generated chunks are stored and the process chooses the next ATI of the first decision point. The process tries to chunk it with as many of the following decision points as possible. This is repeated until all abstract turn instructions for the first decision point have been processed. The chunk that is best, according to the optimization criterion, is stored as the current context-specific route direction. The optimization process then continues with the second decision point, again building chunks for every abstract turn instruction of that decision point. It ends when the last decision point's ATIs have been processed.

Looking at the example, the first decision point's set of abstract turn instructions contains only one element (`(left)`). Building all chunks for this element yields {`((DP`$_1$`),(left))`, `((DP`$_1$`,DP`$_2$`),(left,left))`, `((DP`$_1$`,DP`$_2$`),(left, left / church lm`$^<$`))`}. According to the chosen optimization criterion, the last chunk is best. Compared to the other chunks, it covers the most decision points (together with the second chunk), and is on a higher granularity level than the other chunks. It is stored as the current context-specific route direction. The process continues with DP$_2$: left and straight as well as left and follow are not chunkable, so no chunks except for the trivial ones result from the first two ATIs of this decision point.[5] The ATI (follow / river) appears in the set of ATIs for the following two decision points (DP$_3$, DP$_4$), as well. According to the postprocessing principles, references to linear landmarks—represented by the direction relation follow—need to end with an end qualifier. Therefore, both chunks ((DP$_2$),(follow / river)) and ((DP$_2$,DP$_3$), (follow / river, follow / river)) are removed in the postprocessing step, resulting in {((DP$_2$), (left)), ((DP$_2$), (left/church lm$^<$)), ((DP$_2$,DP$_3$,DP$_4$), (follow / river, follow / river, follow / river, gas station))}.

Chunks are stored in a table that has as many entries as there are decision points in the route. At each entry, the best chunk ending with the decision point that corresponds to the entry's number is stored. For example, at entry four, the best chunk that ends with the fourth decision point of the route is stored. This table allows for the construction of partial route directions. This is needed for rebuilding context-specific route directions if an update is required. The set of chunks is iterated to generate the table. For each chunk, the algorithm checks whether the chunk currently stored in the table at the entry corresponding to the chunk's last decision point, is better than the chunk at hand. If the entry is empty or the new chunk is better, the new chunk is placed at this table's entry. After generating chunks for the first two decision points, the table of chunks for the small example is as following:

1: ((DP$_1$), (left))
2: ((DP$_1$, DP$_2$), (left, left / church lm$^<$))
3: *empty*
4: ((DP$_2$, DP$_3$, DP$_4$), (follow / river, follow / river,
follow / river, gas station))
5: *empty*

Since no chunks have been built so far ending with DP$_3$ or DP$_5$, the entries 3 and 5 in the table are empty.

After processing each decision point, the algorithm updates the current context-specific route direction. To this end, it picks the best newly generated chunk, and checks whether it needs to be integrated in the current route direction. If none of the decision points contained in the newly generated chunk are already stored in the current solution, the chunk does not overlap with the current route direction. In this case, the chunk is added to the context-specific route direction. If there is an overlap, and the newly generated chunk is better than the chunk currently used in the route direc-

[5] A trivial chunk covers only a single decision point. The direction relation of this chunk is not combinable with any other decision point. For the example route, a sample trivial chunk is (DP$_2$),(left).

tion that contains the overlapping decision points, the context-specific route direction are rebuilt using the new chunk.

There is no guarantee that simply removing the last chunk of the current solution, and adding the new chunk, still covers every decision point of the route. Therefore, context-specific route directions are rebuilt if an update is required. Generating a new route direction starts with the new chunk, i.e. the chunk that needs to be integrated in the route direction. As this is the (currently) last chunk of the route direction, the instructions to use for the route's part prior to this chunk need to be determined. To this end, the first decision point of the current chunk, i.e. the one closest to the route's origin is chosen. If it does not correspond to the origin, the chunk stored in the table of chunks at the entry previous to it is added to the context-specific route direction. This is the best chunk that covers the part of the route immediately before the last chunk added. This updating process is repeated until the origin is reached. Algorithm 3 summarizes the local optimization process; here, *CSRD* is short for context-specific route direction.

In the example, after $\mathtt{DP}_1$ has been processed the current context-specific route direction is `((DP`$_1$`,DP`$_2$`),(left, left / church lm`$^<$`))`. The best chunk generated for $\mathtt{DP}_2$ is `((DP`$_2$`, DP`$_3$`, DP`$_4$`), (follow / river, follow / river, follow / river, gas station))`. This chunk overlaps with the current context-specific route direction at $\mathtt{DP}_2$ and is better than the chunk currently covering this decision point. Therefore, an update is required. A new context-specific route direction is generated starting with the current chunk—the one covering decision points two to four. This does not cover the origin, yet. Accordingly, the table's entry previous to the current chunk, which is entry 1 in this case, is added: `((DP`$_1$`), (left))`. The process then continues with decision point three; the best chunk here is `((DP`$_3$`, DP`$_4$`, DP`$_5$`), (straight, straight, right))`. This chunk overlaps with the current context-specific route direction at decision points three and four, but it is not better than the chunk covering these decision points (due to its finer level of granularity). Hence, no change in the route direction occurs and the chunk is not added to the table. Processing $\mathtt{DP}_4$ also results in no changes. $\mathtt{DP}_5$, finally, is the last and cannot be chunked with any other decision points. Hence, `((DP`$_5$`), (right))` is its best chunk which does not overlap with the current context-specific route direction and is added. So the context-specific route direction resulting from local optimization is: `((DP`$_1$`), (left)), ((DP`$_2$`, DP`$_3$`, DP`$_4$`), (follow / river, follow / river, follow / river, gas station)), ((DP`$_5$`), (right))`. This may be verbalized as "Turn left, then follow the river until the gas station. There, turn right."

Algorithm 3 The local optimization process.

function generate-chunk-table
  for all chunks **do**
    current entry number ← last decision point of current chunk
    if current entry in table is empty or new chunk is better **then**
      replace current entry with new chunk
    end if
  end for
  Return new table of chunks.

function rebuild-csrd
  Delete current CSRD.
  Add current chunk to CSRD.
  while origin not covered **do**
    current position ← number of first DP of last added chunk
    new chunk ← chunk at current position - 1 in chunk table
    append new chunk to current CSRD.
  end while
  Return new CSRD.

function local-optimization
  for all DPs **do**
    determine every ATI possible according to the systematics resulting in a set of ATIs for each DP.
  end for
  while DP **do**
    while ATI **do**
      try to chunk current ATI with as many of the following DPs as possible.
      Store generated chunk.
    end while
    generate-chunk-table
    if the best newly generated chunk does not overlap with current CSRD **then**
      add it to CSRD.
    else if there is an overlap **then**
      if newly generated chunk is better than current chunk **then**
        rebuild-csrd
      end if
    end if
  end while
  Return current CSRD as resulting CSRD.

5.4.3.2 Global Optimization

Global optimization takes into account every possible solution; it looks at every route direction that can be generated. It guarantees finding the optimal solution, but has higher computational complexity and takes more time than local optimization.

Just as in local optimization, in the first step every possible abstract turn instruction for each decision point of the route is determined. Then chunking is performed, resulting in a set of chunks for every decision point. These chunks are the elements global optimization works on. For the small example of a route with five decision points (see the previous section), the sets of chunks are listed in Table 5.1:

Table 5.1: Sets of chunks of the example route.

DP$_1$:	{((DP$_1$), (left)), (DP$_1$, DP$_2$), (left, left)), ((DP$_1$, DP$_2$), (left, left / church lm$^{<}$))}
DP$_2$:	{((DP$_2$), (left)), ((DP$_2$), (left / church lm$^{<}$)), ((DP$_2$, DP$_3$, DP$_4$), (follow / river, follow / river, follow / river, gas station))}
DP$_3$:	{((DP$_3$), (straight)), ((DP$_3$, DP$_4$), (straight, straight)), ((DP$_3$, DP$_4$, DP$_5$), (straight, straight, right)), ((DP$_3$, DP$_4$), (follow / river, follow / river, gas station))}
DP$_4$:	{((DP$_4$), (straight)), ((DP$_4$), (follow / river, gas station)), ((DP$_4$, DP$_5$), (straight, right))}
DP$_5$:	{((DP$_5$), (right))}

Finding an optimal solution is realized as a recursive process. It starts by selecting the first chunk of the first decision point's set of chunks as the current solution. In the following, in each iteration it is checked whether the current solution covers the destination. If this is the case, the route direction is complete. If a newly generated complete route direction is better than the current optimum, it is stored as the new current optimum. The current optimum is returned to indicate that a new complete route direction is found. Partial solutions are also checked whether they are better than the current optimum. This check allows for disregarding partial solutions that will not lead to a new optimum. If a partial solution is already worse than the current optimum any complete solution generated from it will also be worse than the current optimum due to the way optimization criteria are formulated. In case continuing with the current solution is still worthwhile, the next chunk is added recursively. The optimization process determines the next decision point of the route that is not covered by the current solution, and recursively tries adding each chunk of this decision point to generate a new current solution.

This way, all possible solutions are considered, guaranteeing to find the optimal solution. Disregarding partial solutions that will not lead to an optimum helps to speed up the process considerably, diminishing the drawback of increased computational complexity. Algorithm 4 summarizes the global optimization process.

In our example, the first complete context-specific route direction the process returns is `((DP`$_1$`), (left)), ((DP`$_2$`), (left)), ((DP`$_3$`), (straight)), ((DP`$_4$`), (straight)), ((DP`$_5$`), (right))`. Since the current optimum is empty, this solution is set as the new current optimum. Then the next solution also starting with `((DP`$_1$`), (left))` gets generated, resulting after some iterations in the solution `((DP`$_1$`), (left)), ((DP`$_2$`, DP`$_3$`, DP`$_4$`), (follow/river, follow/river, follow/river, gas station)), ((DP`$_5$`), (right))`. This solution is the global optimum; therefore, all further partial solutions will be aborted prematurely. This is the same solution as found by local optimization.

5.4.4 Computational Complexity of Optimization

In the following, I analyze the computational complexity of local and global optimization. To this end, I assume that a route consists of n decision points. For each decision point there are m different abstract turn instructions. Generating the abstract turn instructions for the single decision points is not considered as this needs to be done prior to both local and global optimization. Thus, it does not contribute to differences in the complexity of the optimization approaches.

5.4.4.1 Local Optimization

Looking at the algorithm for local optimization (algorithm 3 on page 114), there are three functions that contribute to the complexity.

Function `local-optimization` consists of two while-loops. The inner loop iterates all m abstract turn instructions which are subject to chunking. Theoretically, all m ATIs may be chunkable with all m ATIs of the subsequent decision points. In other words, the m ATIs of the first decision point may be chunked with all m ATIs of the second decision point. These chunks may again be chunked with the m ATIs of the third decision point, and so on. This results in m^n chunks that are theoretically possible for the first decision point. For each subsequent decision point, the power determining the number of possible chunks decreases by one, resulting in $\sum_{l=0}^{k-1} m^{k-l}$ possible chunks; with k being the number of decision points from current decision point to destination. Consequently, the complexity for the inner while-loop is $O(m^k)$.

After processing the inner while-loop function `generate-chunk-table` is called. This function iterates through all previously generated

Algorithm 4 The global optimization process.

function get-all-chunks
  for all DPs **do**
    determine every ATI possible according to the systematics resulting in a set of ATIs for each DP.
  end for
  for all DPs **do**
    for all ATIs **do**
      chunk ATI with following DPs according to chunking principles.
    end for
  end for
  Return set of all chunk sets.

function global-optimization-rec
  if current-chunk contains elements **then**
    append current-csrd with first element of current-chunk.
    if current-csrd covers destination **then**
      if current-csrd is better than current-optimum **then**
        current-optimum ← current-csrd
      end if
      Return current-optimum.
    else
      if current-csrd is better than current-optimum **then**
        global-optimization-rec (set of chunks of next DP to be added) chunks current-csrd current-optimum
      end if
    end if
    global-optimization-rec (next element of current-chunk) chunks (remove chunk added last from current-csrd) current-optimum
  else
    Return current-optimum.
  end if

function global-optimization
  chunks ← get-all-chunks
  current-chunk-set ← first element of chunks
  current-csrd ← NIL
  current-optimum ← NIL
  global-optimization-rec current-chunk-set chunks current-csrd current-optimum

chunks in a for-loop. Thus, its complexity is also $O(m^k)$. Function `generate-chunk-table` reduces the number of chunks from $\sum_{l=0}^{k-1} m^{k-l}$ to k, as it stores only the best chunk for each of the k decision points in the table.

Next it is checked whether the best newly generated chunk should be added to the context-specific route direction. The best chunk can be determined in the inner while-loop simply by comparing the newly generated chunk to the currently best chunk. This has constant complexity and, thus, does not add to the overall complexity. If the chunk is to be added, the context-specific route directions may need to be rebuilt using function `rebuild-csrd`. This function uses the chunk table for rebuilding the route directions. It needs to access this table at most n times—once for each decision point—resulting in a complexity of $O(n)$.

Combining all the single complexities by adding them up results in a complexity of $O(m^k)$. Finally, the outer while-loop of function `local-optimization` contributes to the complexity. It iterates all n decision points of the route. In each iteration, k decreases by one; therefore, the overall number of chunks is $\sum_{k=1}^{n} \sum_{l=0}^{k-1} m^{k-l}$. Thus, the overall complexity for the chunking part is $O(m^n)$. Generating route directions from the chunk table has a complexity of $O(n^2)$ as, at most, `rebuild-csrd` is called n times for n decision points. The chunking part dominates the complexity of local optimization. Therefore, it has a complexity of $O(m^n)$. However, as will be demonstrated below, in practice the number of chunks is significantly lower and the generation part dominates the algorithm's execution time.

5.4.4.2 Global Optimization

Global optimization (algorithm 4 on page 117), starts by generating all possible chunks for all m abstract turn instructions of the n decision points. I assume that an abstract turn instruction of the first decision point is chunkable with all of the m abstract turn instructions of the subsequent decision points. This results in m^{n-1} chunks starting with this abstract turn instruction that cover the complete route. If this holds for all m abstract turn instructions of the first decision point, there are m^n chunks that cover the complete route.

As global optimization checks every possible context-specific route direction, looking only at those chunks that cover the complete route is not sufficient. Assuming that every part of a complete chunk is a valid chunk again, for each complete chunk there exist 2^{n-1} possible partitions of this chunk, such that the sequence of these parts covers the complete route. Since there are m^n complete chunks this results in $m^n 2^{n-1}$ chunks. This number also corresponds to the number of possible combinations to cover the complete route, i.e. to generate valid route directions from origin to destination. The m^n complete chunks already contain all possible combinations of the abstract turn instructions; these partitions correspond to possible route directions. Consequently, the complexity of global optimization corresponds to the number of possible directions; it is $O(m^n 2^{n-1})$.

5.4.4.3 Reconsidering the Number of Possible Chunks

Theoretically, the number of possible chunks that result from the chunking process dominate the complexity of the optimization algorithms. In practice, however, significantly less chunks are possible. The way chunking is implemented allows combining only specific abstract turn instructions. For most of the elements of the systematics, these are abstract turn instructions that are based on the same element (see Table 5.2). Accordingly, the number of chunks that really can be generated is much smaller than that assumed in the considerations above. It is bounded by the number of elements applicable in route directions, which is 12, and the number of their possible combinations, which is 14.

Table 5.2: Chunking options for the elements of the systematics.

element	chunkable with
cardinal directions	cardinal directions
global landmarks	global landmarks
edges	edges
slant	slant
egocentric reference	egocentric reference T-intersection landmark at decision point
T-intersection	egocentric reference
landmark at decision point	egocentric reference
distant landmark	distant landmark
landmark between decision points	—
linear landmark	linear landmark
area-like landmark	area-like landmark
path annotation	path annotation

Furthermore, in practice the number of elements that are applicable for any decision point is significantly lower than 12; hardly more than 4 to 5 different elements are applicable. Many of these elements depend on the presence of landmarks. In principle, the number of landmarks is unbounded. That is, theoretically an unlimited number of landmarks may be applicable for a single decision point. This would result in a large number of abstract turn instructions and, consequently, a large number of chunks. However, first, for most decision points the number of applicable landmarks is low and, second, for most elements these landmarks need to be applicable for consecutive decision points in order for chunking to be possible. This usually only holds for a few landmarks and a subsequence of the decision points.

In conclusion, the number of chunks that is possible theoretically, is much smaller in practice. In fact, the number of chunks does not increase exponentially with the number of decision points, but rather by a constant factor. The number of chunks can be assumed to be in the order of the number of decision points. Therefore, the complexity of the optimization algorithms is dominated by finding the optimum. This is $O(n^2)$ for local optimization and $O(2^{n-1})$ for global optimization.

5.5 Summary

In this chapter I presented the algorithmic realization of the generation process for context-specific route directions. It is based on an annotated graph that represents the environment. Using this representation, I detailed how the different abstract turn instructions corresponding to the elements of the systematics presented in Chapter 3, can be determined automatically. In particular, I explained how to determine the relation of the different landmark types considered to the route.

Chunking is realized as a two-step process. This way, simple, syntax-based grouping of abstract turn instructions can be separated from cognitive principles of chunking which allows flexibly applying different sets of these cognitive principles. Two optimization processes, local and global optimization, have been introduced and their algorithmic realization explained. The latter guarantees optimal results; the former is less computationally complex while usually still resulting in the global optimum as is demonstrated in the next chapter.

Chapter 6

"How Do I Get To...": Evaluation of Context-Specific Route Directions

In this chapter, I report results from the evaluation of the presented approach to context-specific route directions. This evaluation is twofold; first, the computational performance of the implemented system is analyzed, and local and global optimization are compared with respect to resulting differences in the generated route directions. The second part presents results of an informal, exploratory human subject test that is meant to check the plausibility of the chosen approach. It tests whether the generated directions are usable by humans and compares context-specific route directions to human-made and route-planner route directions.

6.1 Computational Evaluation

The computational evaluation of Guard serves two purposes. First, it shall demonstrate how the system works, especially how chunking can significantly reduce the information to be communicated, even for arbitrary routes. And second, I compare the results of local and global optimization to demonstrate that local optimization, which is significantly faster than global optimization, leads to optimal results in practice, even though it does not guarantee this theoretically.

6.1.1 Randomized Generation of Routes

As underlying data for evaluating the computational system, I collected data on real world intersections. Panorama photographs of these intersections have been taken. I analyzed them afterwards and encoded the depicted spatial situation in an XML-description. For encoding, the egocentric perspective captured in the photographs is transformed into a birds-eye view. The description includes the number of branches and their angle relative to the incoming route-segment, with 180 degrees denoting 'straight'. Also, landmarks and their location are encoded. In the XML-description, the functional role of a landmark is stored in attribute `c-type`, and the type of object in attribute `type`.[1]

If a landmark has a specific name this is given by the attribute `name`. A landmark's location is encoded relative to the intersection; attribute `position` lists the branches that the landmark is located between (see Fig. 6.1). Figure 6.2 shows an example photograph with the corresponding XML-description of the intersection. The incoming branch, i.e. the street from which the photo is taken, is not explicitly encoded. 47 intersections have been created this way. These intersections also form the basis for the human-subject test presented in Section 6.2. The intersections used in this test together with their XML-description are listed in Appendix A.

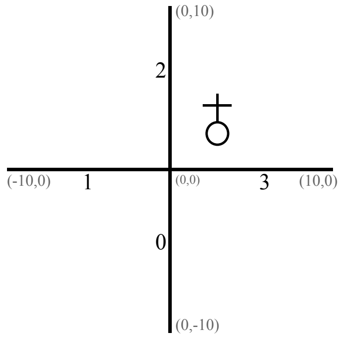


Figure 6.1: Relative location of a landmark with respect to an intersection's branches. In this example, the landmark is located between branch 2 and 3. Accordingly, the value of attribute `position` is '2-3'.

The XML-descriptions are read in by GUARD and then transformed into intersection- and landmark-objects. These objects are grounded in a local coordinate system; initially all intersections are centered around point $(0, 0)$ (see Fig. 6.3a). The incoming branch, which is only implicitly encoded, starts at point $(0, -10)$. When generating routes using the intersection objects, it needs to be ensured that this branch is always the one the inter-

[1]Values of the `type`-attribute adhere to the common-sense ontology of outdoor objects developed in project I1-[OntoSpace] of the SFB/TR 8 Spatial Cognition (personal communication with Joana Hois and John Bateman). This ensures that the generated route directions can be properly externalized verbally using their language generation systems. The common-sense ontology is based on the NEXUS augmented world model (Nicklas & Mitschang, 2001).

```
<intersection id="2">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <branch no="3" direction="90"/>
 <landmark position="0-1" type="TrafficLight"
  c-type="routemark"/>
 <landmark position="0-3" type="TrafficLight"
  c-type="routemark"/>
 <landmark position="2-3" type="PoliceStation"
  c-type="routemark"/>
</intersection>
```

Figure 6.2: Photo of an intersection and its XML description.

section is entered on. Depending on the outgoing branch of the previous intersection, the current intersection needs to be rotated according to the corresponding turning angle (Fig. 6.3b), and then translated to the outgoing branch's endpoint (Fig. 6.3c).

This way, it is possible to generate a connected sequence of intersections, i.e. a route based on previously isolated intersections. There are two options for doing this. First, specifying a fixed order of intersections, and the turns to take, by stating the corresponding branch's number. The routes used in the human-subject tests are generated this way (see Section 6.2). For the computational evaluation, another approach is taken. All 47 intersections are read in at once and then arbitrary routes are generated. First, the length of the route is determined randomly; it consists of 6 to 14 decision points. Then, intersections are randomly selected out of the set of 47, ensuring that no intersection occurs twice in the route. Finally, for each intersection, one of its branches is randomly determined as the outgoing branch.[2] Routes gen-

[2]Principally, this procedure may result in intersections sharing the same coordinate. By rotating and translating, an intersection may end up at a coordinate of a previous

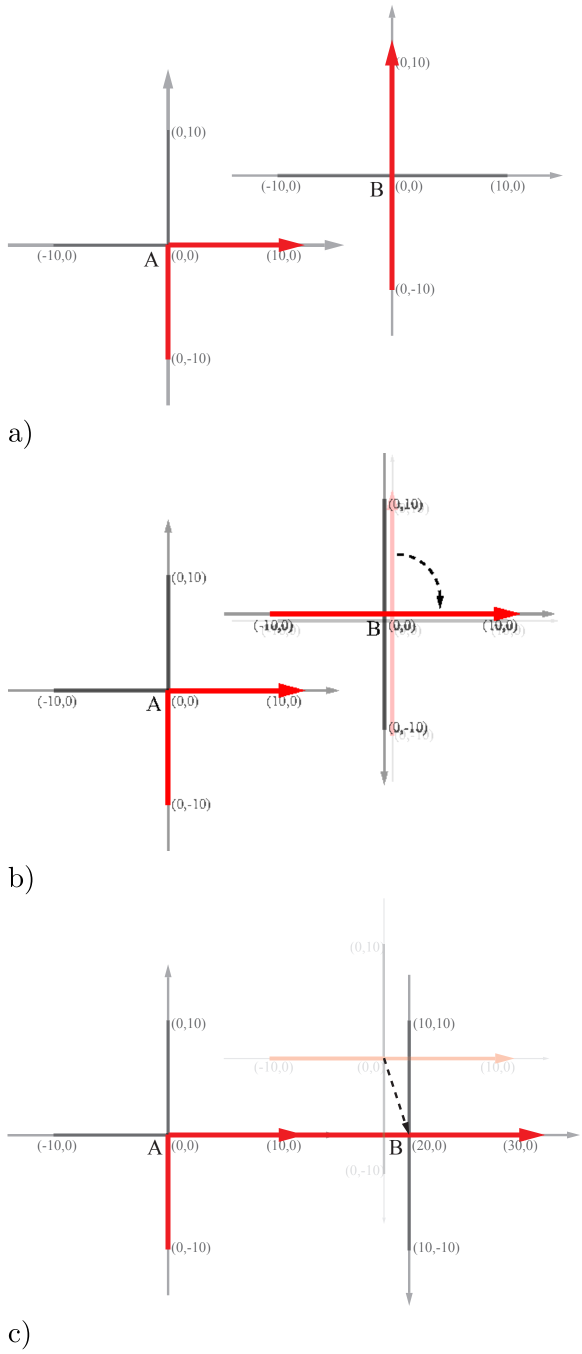


Figure 6.3: Randomized generation of routes: a) two intersections in their local coordinate system; the decision point is at $(0, 0)$, the incoming branch starts at $(0, -10)$. b) Rotating intersection B according to the outgoing branch's angle of intersection A. c) Translating intersection B to meet the outgoing branch of A.

erated this way can be used as input for GUARD. The system's performance is tested using such random routes.

6.1.2 Analyzing the Generated Route Directions

The aim of the computational evaluation is to elicit the general performance of GUARD with respect to chunking and the differences between local and global optimization. To that end, different mean values are calculated; to get meaningful values this requires recording many single runs of the system. Using the procedure of generating arbitrary routes described in the last section, several runs of the system have been recorded to evaluate its performance. A single test run consists of creating 1000 routes and then building context-specific route directions for each route, using both local and global optimization. As a first result, all runs were successful, i.e. for each route GUARD was able to generate route directions. Table 6.1 lists more detailed results of five test runs. These runs are good examples of the system's behavior. I performed several additional test runs that resulted in similar values; none of the runs of GUARD failed to return a result. The five reported runs adequately reflect the system's performance.

Table 6.1: Computational evaluation: results of five test runs showing the system's general performance with respect to chunking.

	run 1	run 2	run 3	run 4	run 5
average chunk ratio	0.57251	0.57349	0.57671	0.57787	0.57931
best ratio	0.16667	0.1	0.16667	0.16667	0.14286
worst ratio	1	1	1	1	1

The *average chunk ratio* states the average ratio between the number of chunks in the route directions and the number of a route's decision points based on global optimization. On average, chunking reduces the number of needed instructions by about 42%; that is, about two consecutive instructions are chunked into a single instruction. While in each of the runs there have been routes where no chunking could be performed (*worst ratio*), this only happened once (runs 2 to 4) or twice (1 and 5), respectively. On the other hand, *best ratio* shows that the chunking ratio for routes can be significantly lower than 0.5; for some routes even all instructions were chunked into a single instruction. These were simple routes that comprised of going straight for all but the last decision point; at the last decision point, a left or right turn at a landmark was

intersection in the route. But this would only be a problem if this happened with two consecutive intersections, which is impossible. Because the route is processed in forward direction, i.e. from first to last intersection, intersections before the current intersection and its immediate predecessor are not relevant anymore for determining abstract turn instructions.

required. For such routes, all decision points can be combined into a single chunk using landmark chunking. For example in run 1, one such route had six decision points; the first five are in a straight line, the last one requires a left turn at a river. The resulting context-specific route direction is accordingly `((DP`$_1$`, DP`$_2$`, DP`$_3$`, DP`$_4$`, DP`$_5$`, DP`$_6$`),(straight, straight, straight, straight, straight, left / river lm`$^<$`))`. Just as with routes for which no chunking is possible, routes that allow combining all decision points into a single chunk rarely occured. It happened once in runs 1, 3, 4, and 5 and twice in run 2.

Table 6.2: Computational evaluation: comparing local and global optimization.

	run 1	run 2	run 3	run 4	run 5
local CSRD longer	1	2	9	5	8
times of differences in chunks	895	915	901	897	908
times of diff. in chunks w/o landmarks	299	303	263	282	294
average number of differences	2.33073	2.34208	2.27969	2.30881	2.27974
av. number of diff. w/o landmarks	0.67821	0.67978	0.61265	0.64214	0.64978

When comparing local and global optimization, it turns out that local optimization indeed does not always produce the optimum. But, as *local CSRD longer* in table 6.2 shows this happened in less than 1% of the test runs. Thus, with respect to the number of chunks, there is hardly ever a difference between local and global optimization. Differences in the chunks themselves are quite frequent though, as can been seen in line *times of differences in chunks*. On average, there are between 2 and 3 differences in chunks (*average number of differences*). However, these differences mostly result from different landmarks being referred to. Since GUARD currently only provides limited means to decide which of all possible landmarks to refer to—especially a landmark's saliency is not taken into account—which landmark is chosen is, to some extent, arbitrary (see Section 5.4). Additionally, in the data used for the evaluation, traffic lights are encoded individually, i.e. each pole is represented as an individual landmark (see Fig. 6.2). Often, either of these poles is referred to in the route directions which explains many of the differences. When disregarding these differences, the number of times there are differences in chunks drops from about 90% to about 30%, and the average number of differing chunks to less than 1 (*av. number of diff.*

w/o landmarks). The remaining differences mostly occur when a sequence of actions can be chunked in two (or more) ways that result in the same number and granularity of chunks. A sequence of actions like `straight`, `left`, `left`, for example, may be chunked either as `(straight, left)`, `left` or as `straight`, `(left, left)`. Local optimization proceeds forward through the route, where it comes up with the former possibility; global optimization with the latter.

6.1.3 Discussion

GUARD succeeded in generating route directions for every route. This demonstrates that the employed mechanisms, and the cognitive principles of good route directions realized in the system, allow for an automatic generation of route directions. The inclusion of such principles in a computational system is possible.

Chunking is a valid, sensible mechanism for reducing the amount of information communicated in route directions. Chunking greatly enhances the ability for wayfinders to conceptualize and remember route directions. Even with a randomized generation of routes, it allows for a reduction of the number of instructions needed, to about one instruction for every second decision point. The routes for which no chunking is possible, are comprised of alternating left and right turns that cannot be described using any other element of the systematics. Such alternation is a regular pattern that, in principle, could be chunked, like in "go zigzag," but such patterns are currently not covered.

With respect to local and global optimization, no significant differences in the resulting route directions can be observed. Both methods result in route directions that consist of an equal number of chunks. The differences within these chunks are explainable from ambiguities in other parts of the system and are not caused by the optimization methods themselves. Thus, local optimization, which is computationally less complex and therefore faster than global optimization, can be used for generating context-specific route directions without degrading the resulting route directions.

6.2 Human-Subject Test

With the help of an exploratory human-subject test, I try to answer two questions: most importantly, are the route directions generated by GUARD understandable, i.e. do they allow someone to find her way? Second, and subordinate to the first question, are they better suited for assistance than those route directions generated by typical internet route planners?

To answer these questions, I performed a three-part study based on six different routes. Part two and three compare context-specific route directions with route-planner directions: first in a rating test (Section 6.2.3);

second, human subjects are asked to find their way along a route using these route directions (Section 6.2.4). To further evaluate context-specific route directions and, especially, to get a clearer picture of their acceptance, I also compared them to directions given by humans. To this end, I collected written route directions for four of the study routes in a first part of the study (Section 6.2.2).

6.2.1 Study's Aim and General Approach

While the computational analysis presented in the last section demonstrates how Guard performs in generating context-specific route directions, checking whether the implemented principles (see Chapter 4) result in useful route directions can only be answered in a human-subject test. In the following, I present the general approach chosen for this test.

Guard generates an abstract, relational specification of the actions to be performed in route following. Such a specification cannot be directly tested with human subjects as it is hardly understandable to them. Accordingly, it needs to be externalized, for example, turned into a verbal or graphical representation, in order to be communicated to human users. In the study, I chose verbal externalization which is easier to generate and better suited for a comparison with internet route-planners. This externalization step is not in the scope of my thesis; the verbal externalizations are provided by Prof. John Bateman as a result of collaborative work (see Section 6.2.2 for more details).

The focus of my thesis is on representation-theoretic and modeling ggquestions, i.e. on the development of a process for generating route directions and of methods to deal with the different elements of the systematics. The evaluation of context-specific route directions is important for demonstrating their appropriateness; but it should be kept as simple as possible, while still providing informative results. The presented study, therefore, is exploratory and informal; it aims at showing that the chosen approach is plausible. A more rigorous study may be conducted for further evaluation as future work (see Section 7.2.6).

A desktop photo-based study is performed to better control the routes' properties and to keep the effort reasonable (details are presented in Sections 6.2.3 and 6.2.4). The routes used in the study are encoded and generated in the same way as presented in Section 6.1. The routes are based on the same set of panorama-photographs as used in the computational evaluation. The photos have been taken at various places in Hamburg and Bremen. The depicted intersections have been encoded in XML as explained in Section 6.1.1.

Practical considerations lead to such a desktop study. In order to be a valid test, the routes the human subjects have to follow during the study need to be unknown to them. Doing this in a real world, outdoor setting would require finding routes that are most probably unknown to many potential subjects and then finding subjects that indeed do not know these routes. This may result in doing the study in a different city than the one the subjects are living in. No data sets of geographic information exist that contain all the information needed for Guard to fully exploit all its capabilities. This is especially true for information on landmarks. Accordingly, this information needs to be hand-coded. The tested routes shall provide opportunities to employ Guard's key principles, such as linear landmarks being followable or functioning as point landmarks, chunking, or references to landmarks and structural features, and be balanced.

Generating routes for the study follows two principles. First, intersections are combined in a route in a way that the sequence of photos is plausible, i.e. changes of the environment do not occur too rapidly. Second, the emerging routes are designed such that they allow for testing Guard's main features like references to linear landmarks or chunking. Six routes are generated for the study; the photos and their corresponding XML descriptions are listed in Appendix A. Putting the routes together this way, with intersections in an arbitrary, yet purposive order, guarantees that no subject knows the route beforehand, even though single intersections may be known to some of the subjects.[3] The six routes are the basis for the study material used.

6.2.2 Giving Route Directions

The study's aim is to test context-specific route directions with respect to their acceptance and performance in wayfinding. The directions are generated by Guard as described in Section 6.1; global optimization is employed to guarantee optimal directions. These route directions shall be compared against other kinds of route directions (route directions that are like those of internet route planners and route directions given by humans). The aim is to get hints about how context-specific route directions perform compared to these kinds of route directions to be able to better judge the plausibility of the chosen approach. Route directions cannot be generated automatically by an online route-planner as the selected routes do not correspond to existing street-networks. Therefore, I generated the tested route-planner directions manually, following the instruction pattern used by one typical online system (www.map24.com). The route directions given by humans have been collected in the first part of the study.

[3]Encountering a known intersection in an unknown context might cause some irritation. As the study's setting is rather artificial, this should have only very limited influence on a subject's performance. No subject reported any problems performing the task that resulted from seeing a known intersection.

Material and Methods

I collected route directions for routes 1, 2, 3, and 5. These four routes covered the different types of routes used in the study, and reflected the routes' properties to be tested. Six subjects, from friends and family, participated in this part of the study. Each received a folder with study material. This folder contained written instructions and material for two different routes. Each subject received a sketch map of the routes, for example the one shown in Figure 6.4. The map showed the route as red line with arrow-heads indicating the direction of movement. The streets were labeled with fictional names which reflected typical German street names. All objects that were identified as potential landmarks in the photographs were depicted in the map. Additionally, subjects received an ordered set of photos, one photo for each intersection of the route. They were numbered on the back in case a subject needed to restore their order. The photos for each route are shown in Appendix A.

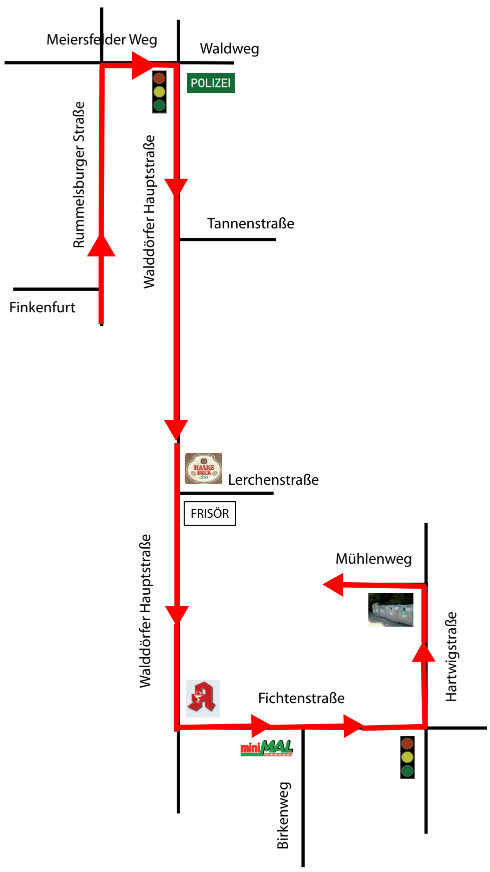


Figure 6.4: A sketch map showing route 1.

Subjects were asked to study the given material, i.e. the sketch and the photos, for the first route thoroughly for as long as they liked, and then to write down route directions that they thought would enable someone who does not know the route, to successfully find her way. Afterwards, they were asked to put the material aside and to continue with the second route. Each subject had to write down route directions for a route offering the possibility to employ landmark chunking based on a linear landmark for considerable parts of the route (routes 2, 3) and for a route that does not (routes 1, 5). The routes given to subjects were as follows, listed as tupels giving the route numbers in the order they were found in the folder: (1,2); (2,1); (1,3); (2,5); (3,5); (5,3). For each route I collected three route directions. The sketch maps are shown in Appendix B.

Results

Subjects use different mechanisms to link actions to be performed to the location these actions need to be performed at. The two most prominent are references to *landmarks* and references to *street names*. Subjects also refer to *structural elements*, like the type of intersection (e.g., T-intersections, as in "Du gehst geradeaus bis zur T-Kreuzung"; *"go straight till the T-intersection"*) or the width of the street one is about to enter ("...dies ist eine etwas größere Straße"; *"...this is a slightly bigger street"*). Also, they employ *counting*, like references to the next ("nächste") or first ("erste") intersection. Occasionally, subjects provide *confirmation information*, as well (e.g., "Auf der linken Seite siehst du das Eiscafé Venezia und ein Stück weiter kommt Schlecker"; *"On the left side you see the ice cream parlor Venezia and a little bit further there is Schlecker"*). That is, subjects use the same mechanisms also employed by GUARD.

While subjects have a tendency to rely more strongly on one specific mechanism, usually they use most of the listed mechanisms in their route directions. For the first route, there is a predominance of street names, but overall references to landmarks and street names occur about equally often. The frequent references to street names may be due to their prominence on the provided map, which is the main source for determining the direction to take.

Often, subjects even combine different mechanisms into a single instruction. Commonly, they integrate a reference to a landmark in the sentence structure and additionally state the relevant street name in brackets at the end of the sentence (like in "Hinter der Apotheke links einbiegen (Am Park)"; *"Turn left after the pharmacy (Am Park)"*), or vice versa. In the same line, they employ counting in combination with references to a landmark or street name (e.g., "Die 3. Straße biegst du dann links ab, an einer Apotheke, in die Fichtenstraße"; *"Turn left at the 3. street, at a pharmacy, into Fichtenstraße"*). Sometimes landmarks are mentioned twice in sequence: the first reference anchors the location of an action to take, the second states the actual action to be performed at that landmark (e.g., “Folge dieser bis zu einer Apotheke auf der linken Seite. Biege hinter der Apotheke links ab.”; *“Follow it till a pharmacy on the left side. Turn left after the pharmacy.”*).

Overall, a lot of redundancy in the instructions can be observed (see, e.g., the sentence above starting with “Die 3. Straße...”). This may be attributed to giving written route directions and may also be due to the fact that the subjects did not know the routes beforehand themselves. Spoken route directions can be expected to be shorter, as usually the instructor has less time to generate the instructions, has to assume that the addressee needs to memorize the spoken instructions without having further assistance to take along, and cannot revise the instructions given afterwards (cf. Lovelace et al., 1999). This is different for written route directions; indeed some subjects obviously inserted additions after having written down their instructions. In unfamiliar environments, you probably pay more attention to details than you would in well-known environments; consequently, you also integrate more details in the instructions. But this would need to be verified experimentally in a different study.

Table 6.3: Length of collected route directions for four different routes: listed is the number of decision points (*#DP*), and the number of instructions (*#instructions*) and words used (*#words used*).

Route	#DP	#instructions	#words used
1	9	3	42
		5	52
		6	89
2	10	4	71
		4	40
		5	50
3	10	6	85
		4	44
		6	51
5	8	5	59
		6	86
		7	188

The length of directions differs, as well. This is true both for the number of instructions and the number of words used. Table 6.3 lists the number of instructions and words used in the collected route directions, compared to the number of decision points, for each route. Except for route 1, the number of instructions is about the same in each direction, but the number of words used varies notably within the directions for each route. This again can be attributed to different degrees of redundancy in the given instructions.

All route directions for the different routes are listed in Appendix B. For the following parts of the study, I chose for each route one of the directions such that they cover the variety of collected route directions: the longest (route 5), the shortest (route 2), one with a predominance of references to landmarks (route 3), and one that mostly relies on street names (route 1).

Verbal Externalization of Context-Specific Route Directions

Verbal externalization of the route directions generated by Guard has been done by Prof. John Bateman, using their natural language generation systems. The following description of the externalization process is based on personal communication with John Bateman.

The generation is ontologically-based. Entities referred to in the route directions are grounded in a common-sense ontology of outdoor objects (cf. Section 6.1.1). The correspondence between temporal and spatial precedence of events and entities in route following is used to freely deduce possible linguistic forms of instructions. Minimal semantic commitments for each style of instruction are defined, which are then used to construct a semantic dependency structure that is based on directed graphs with the event (i.e. the action) as root.

The semantic structures, the graphs, are related to concrete semantics to express them—the content is mapped to the linguistic semantics of the Generalized Upper Model (cf. Bateman et al., 1995), which ensures that appropriate linguistic realizations can be found. Imperatives are chosen; the actor, i.e. the wayfinder, is designated to be the hearer/reader of the instructions. The possible range of linguistic forms produced are restricted by defining constraints, i.e. by defining a collection of ways how content is to be expressed. The chunks generated by Guard are transferred into the produced texts, i.e. the sentence structure reflects the chunks. The language generation process is free to perform linguistic aggregation on these chunks as long as the chunk structure is kept. For example, a chunk like $(\mathtt{DP}_1,\mathtt{DP}_2)$,`(straight, left)` may be verbalized as “go straight, and then turn left” or as “turn left at the second intersection” applying different forms of aggregation. The verbal route directions generated this way are listed in Appendix B.

6.2.3 Rating Route Directions

17 subjects participated in part two and three of the empirical evaluation.[4] The study was performed in German. All subjects were university students and employees that are unfamiliar with my research. They did not receive any payment; each session lasted about 20 minutes. As already explained, the study was desktop-based. Subjects were placed in front of a computer screen using keyboard and mouse as input devices. Instructions and study material were presented on this screen.

The third part of the study, detailed in the next section, evaluates the performance of the different route directions in a desktop-based wayfinding task. The second part of the study, explained here, evaluates the route directions' adequacy and appropriateness based on a rating test. Subjects have to answer questions that assess whether the route directions communicate information well and their confidence that the route directions are suited to lead someone to a destination location.

Material and Method

Each subject received two route directions in the rating task. These were presented sequentially, i.e. the second route direction was presented after rating of the first one was done. One was a context-specific route direction, the other route direction either one of the collected ones, or one of the route planner-style directions. 11 subjects were tested with the route-planner directions; 6 subjects performed the study with the collected directions. The order of presentation, i.e. whether the context-specific route direction was presented first or second was varied across subjects. The two directions stated instructions for two different routes.

For the rating, route directions were displayed on a computer screen, and the rating questions were listed underneath (see Fig. 6.5). Subjects could take as long as they wanted for the rating. They had to choose their rating for each question by clicking one of the radio buttons and then, after having selected a rating for all five questions, press the 'proceed'-button. Ratings ranged from 1 (the best), to 6 (the worst). The ratings chosen by a subject were stored in a database for later evaluation.

The five rating questions were: "Halten Sie diese Routenbeschreibung für verständlich?" ("Are these route directions understandable?"); "Können Sie sich die Beschreibung leicht merken?" ("Can you remember these instructions?"); "Würden Sie eine solche Routenbeschreibung selbst geben?"

[4] I am aware that the study would be more stringent if participants did not perform both parts of the study. But this would require a much greater number of subjects—which are hard to find—and a lot more time in conducting the study. This is an informal study aiming to test my approach for plausibility. Rating appropriateness of route directions most likely has no influence on using them for route following. Thus, I argue, there is no bias here that leads to questionable results.

Figure 6.5: A screenshot showing an example of the setting in the rating study. The route directions are listed on top, underneath are the rating questions.

("Would you give route directions like these?"); "Würden Sie das Ziel mit dieser Beschreibung erreichen?" ("Would you reach the destination using these route directions?"); "Gewinnen Sie durch die Beschreibung eine Vorstellung von der Route?" ("Do you get an impression of the route, after reading these route directions?"). For each question, there were labels at both ends of the row of radio buttons further specifying the range of possible ratings. For example, for the second question, the positive end (1) was labeled "sehr leicht" ("very easy"), and the negative end (6) "gar nicht" ("not at all") (see also Fig. 6.5).

Results

In the following, I concentrate on comparing context-specific route directions with route-planner-style directions. This comparison is my main interest, especially for the route following part of the study.

Table 6.4 lists the results of the rating test. Given are the mean ratings for each of the five questions and the standard deviation. The questions are abbreviated with a keyword (e.g., 'remember' stands for "Can you remember these instructions?") and are in the order as they are listed in the 'Material and Method' section. The rows denote the different kinds of route directions tested: 'context-specific' lists the ratings for context-specific route directions; 'route planner' those for route directions that resemble route-planner-style directions.

Table 6.4: Results of the rating test: mean ratings for each of the five questions comparing context-specific route directions and route-planner directions (s.d. = standard deviation).

	understand		remember		self	
	mean	s.d.	mean	s.d.	mean	s.d.
context-specific	2.09	1.221	2.18	1.168	2.55	1.036
route-planner	2.18	0.874	4.09	1.221	4.73	1.009

	reach		concept	
	mean	s.d.	mean	s.d.
context-specific	1.64	0.809	2.27	1.348
route-planner	1.91	1.136	3.36	1.362

Looking at the means, ratings are generally better for context-specific route directions than for route-planner directions. An analysis of variances confirms this observation; the effect is statistically relevant with a level of significance of 0.015. Only for the question "Are these route directions understandable?" is the mean of route-planner directions about equal to the mean of context-specific route directions. Some subjects reported in the informal post-study interview (some even during the rating test), that they did not really understand what this question aimed at. Thus, subjects might just have considered whether they understand the directions at all, i.e. know generally what the directions were about.

To further evaluate the acceptance of context-specific route directions, they are also compared to the directions given by human subjects. This comparison shows similar results. Table 6.5 lists the rating results in the same way as Table 6.4. Here, 'collected' denotes the directions collected from human subjects.

In this rating, there are no pronounced differences between the means for context-specific route directions and those for the collected directions. Not surprisingly, an analysis of variances does not show any statistical significance (significance level 0.794). It seems that ratings for context-specific route directions are slightly worse compared to those in the route-planner test. But this has not been analyzed further. A discussion of the rating results can be found in Section 6.2.5.

Table 6.5: Results of the rating test: mean ratings for each of the five questions comparing context-specific route directions and directions given by humans.

	understand		remember		self	
	mean	s.d.	mean	s.d.	mean	s.d.
context-specific	2.83	1.472	2.50	1.643	3.33	1.633
collected	3.17	1.835	4.00	1.549	3.33	1.862

	reach		concept	
	mean	s.d.	mean	s.d.
context-specific	2.33	1.211	3.33	1.862
collected	2.33	1.862	2.50	1.761

6.2.4 Route Following Using Route Directions

The third part of the empirical evaluation tests the performance of context-specific route directions in assisting wayfinding. To this end, a screen-based study was conducted which subjects completed immediately after the rating study. The aim wass to elicit whether these directions allowed a wayfinder to reach her destination and to compare these directions with the two other kinds of route directions already used in the rating study.

Material and Method

Each subject had to follow four of the six routes used as study material in sequence. Route following took place only virtually—subjects performed it sitting at a desk (see below). For two of the routes, subjects received context-specific route directions; the other two were route directions collected from human subjects or route-planner directions, respectively. Each subject received route directions she had not previously rated. The order of presentation was varied across subjects. Some subjects received context-specific route directions first, then the other kind, some vice versa, for some there was an alternating order.

After completing the rating part, subjects received new instructions for the route following part. They were asked to read them carefully and to ask for clarifications. When subjects were ready, the first trial began. The route directions for the first route were presented on screen in the same way they had been presented in the rating study.[5] Subjects had time to

[5]Without the rating questions, of course.

read and memorize these directions for 60 seconds, or until they pressed a ‘proceed’-button (whichever came first). There was a timer indicating the time left.

Then route following began. For each decision point along the route, subjects were presented with a screen that showed a photo of the corresponding intersection annotated with street-names. These were the same photos as used in the other parts of the study. Additionally, the intersection’s branches were numbered from left to right, starting with 1. On top of the photo an array of numbered buttons corresponding to the branches’ numbers was displayed. Subjects decided on which direction to take by clicking the button corresponding to the branch’s number they considered to be correct. Figure 6.6 shows an example page; the annotated photos for each route are listed in Appendix A.

Figure 6.6: A sample screen as they were presented to subjects during the route following study.

After selecting a branch, the next intersection appeared. After a route’s last decision point, subjects received a screen announcing that they were finished with this route and that they may continue with the next as soon as they were ready. Pressing the ‘proceed’-button began the next route or presented a ‘thank-you’ screen, in the case that they completed the fourth route. For later evaluation, subjects’ choices were stored in a database that recorded for each route, each decision as the clicked-on number. Additionally, the time it took a subject to complete a route was stored.

All alternatives (branches) at a decision point were always presented such that they were identifiable, and subjects were always presented with the correct next decision point no matter what they chose. They could not go wrong and received no feedback on whether their decision had been correct. As a consequence, they were explicitly told that they did not need to recover from wayfinding errors that they realized themselves, and they could not make follow-up errors. This allows rating a subject's performance on a route simply by counting the errors made.

Results

To judge the performance of context-specific route directions, and to compare them with route-planner (and collected) directions, I collected errors made during route following and the time it took a subject to complete a route. An error occurs if a subject chose a different branch than the intended one, i.e. if the stored number is different to the expected one. Since the tested routes have different lengths, the number of errors and time need to be normalized. Both errors and time for each route are divided by the number of decision points. This is necessary as, for example, the chance to choose a wrong turn on a longer route is greater than on a shorter one. All subjects followed four different routes, two assisted by context-specific route directions and two by other directions. In order to not double the number of subjects in the statistical analysis, for each subject error and time are averaged over the two identical conditions. Due to an error that occurred in encoding the route's intersections for use with Guard that I detected only after performing the study, route 4 had to be excluded from the evaluation.

Based on this data, a t-test has been performed to check for significant differences between the different kinds of route directions. The mean error for context-specific route directions is 0.159, i.e. subjects made 0.159 errors per decision points (or one error about every 6 decision points). The mean error for the route-planner directions is 0.122. This, however, is not a statistically relevant difference (significance level of 0.361). When looking at the time each decision took, for context-specific route directions, the mean time is 6.41 seconds; for route-planner directions 5.56 seconds, which results in a nearly statistically significant difference (0.093). Comparing context-specific route directions with the collected directions clearly results in no significant difference neither for mean errors (0.694) nor for time (0.721).

6.2.5 Discussion

The performed study is informal and exploratory aiming at testing the plausibility of the chosen approach for generating route directions. The collected data on the subjects' performance allows drawing conclusions in this respect, but some of the observations may be attributed to the study's design.

The rating test shows a preference for context-specific route directions compared to the route-planner directions. This preference is statistically significant and especially obvious with questions two and three. Question three is "Would you give route directions like these?" Here, the results are not surprising as people usually prefer landmark over street information (cf. Tom & Denis, 2003). Even more, people rarely employ exact distances, as they have been used in the route-planner directions, but rather rely on either qualitative (e,g., 'shortly', 'quite long'), or quantitative ('about 2 kilometers', 'just 50 meters') categories when referring to distances. Question two asks if subjects consider it easy to remember the directions. Here, route-planner directions are rated unsuitable, while subjects consider context-specific route directions to be memorable.

There are no clear differences compared to the collected directions. However, there seems to be a preference over the collected directions with respect to the 'remember'-question. This may be attributed to context-specific route directions presenting route information concisely, while the collected directions contain a lot more redundancy—possibly due to the direction givers' unfamiliarity with the routes. Overall, the results of the rating test demonstrate the subjects' acceptance of context-specific route directions and their confidence that the directions are of good use. These results are in line with current usability studies (e.g., Ross et al., 2004).

The route-following part of the study, however, does not show such clear results. Generally, it can be observed that context-specific route directions are suited for wayfinding assistance. On average, subjects make a route following error about every sixth decision point. With the study routes being between seven and ten decision points long, this means that there is about one error per route.[6]

Neither is there a significant difference to the collected directions nor to the route-planner directions. Therefore, the route-following part does not allow for ranking the different kinds of directions with respect to their performance. When looking at the mean time a decision takes, route-planner directions seem to be advantageous; this may be attributed to the stimulus material. Street names are clearly readable on the photos. This significantly reduces the time it takes to identify the relevant information, compared to scanning the complete photo for a landmark. Since subjects prefer context-specific route directions to route-planner directions, and these directions do not differ in their performance, assisting human wayfinders with context-specific route directions is not only feasible, but also favorable. Human wayfinders can be expected to be more satisfied with the presented directions and to be more confident that they will reach their destination (similar to results presented by Ross et al., 2004).

Repeating the experiment in a real-world setting, where subjects have to follow an unknown route in an outdoor setting with realistic street signs

[6] As always with such statistical analyses, this mean does not illustrate the details. There are subjects (and routes) with more errors. But there are also several subjects that do not make any errors at all. However, inter-subject or inter-route differences have not been further analyzed.

and visibility of landmarks, would allow for a more profound analysis and comparison of context-specific route directions with other kinds of direction. Here it can be expected that performance results shift in favor of context-specific route directions. However, such a study is left for future work (see Section 7.2.6).

6.3 Summary

The evaluation of context-specific route directions shows that the chosen approach is plausible and results in route directions that are accepted by human users.

Combining principles of good route directions in an automatic generation process is clearly possible, as computational evaluation shows. GUARD succeeded in generating route directions from origin to destination for every tested route. Especially, I was able to demonstrate that chunking is a powerful method for reducing the information explicitly communicated in route directions. Even for routes stemming from a randomized generation, only about every second decision has to be explicitly stated. I was also able to show that local optimization results in optimal results in practice even though it does not guarantee this theoretically.

In the empirical evaluation, the rating test shows a preference for context-specific route directions over route directions that are similar to those produced by internet route planners. Subjects consider context-specific route directions to be more memorable and they are more confident in their ability to reach their destination using these directions. The route-following part of the study indicates that subjects are able to follow a route using context-specific route directions. The mean error rate is low, and errors may partially be attributed to the artificial setting used in the study. A comparison to route-planner and collected directions does not result in significant differences with respect to wayfinding performance. However, to get a clearer picture of the performance of context-specific route directions, a more rigorous real-world study is called for, which is left to future work.

Chapter 7

Conclusions and Outlook

This chapter concludes my thesis. I provide a summary of the presented work, take a look back at the contributions that I announced in Chapter 1, give an outlook on possible extensions and future work, and finally end with some concluding remarks.

7.1 Summary

Finding one's way around in an environment is an everyday task for humans. This is especially true for getting from some origin to a specific destination, i.e. for following a route. In environments that have not, or only infrequently, been visited before humans need assistance for this task. This assistance can be provided through route directions. As this situation occurs frequently, giving and receiving such directions is a task often performed. Accordingly, wayfinding and wayfinding assistance is a well established research area with contributions from different fields; for instance cognitive science, linguistics, behavioral psychology, and artificial intelligence.

In my work, I am concerned with an automatic generation of route directions as an abstract specification that accounts for cognitive principles. In Chapter 2, I summarized the research relevant for my work and put it into context. In that, I concentrated on cognitive and linguistic findings on giving and understanding route directions, and on representation-theoretic considerations regarding route directions. Taking this discussion up, in Chapter 3 I analyzed properties of route directions, especially with respect to spatial chunking and the reference systems used. Identifying the kinds of knowledge involved in different kinds of instructions, I presented a systematics of elements that may be used in route directions. This systematics is the basis for generating context-specific route directions. The elements are grouped according to their level of granularity, i.e. according to how much they abstract from a detailed description of the action to be performed at a single decision point. There are three levels: the level of global references, the level of environmental structure, and the level of path and route.

I detailed Guard, the process for generating context-specific route directions in Chapter 4 (short for Generating Unambiguous, Adapted Route Directions). Each element of the systematics is linked to a set of direction relations that represent the required action at a decision point. These direction relations are the key elements of abstract turn instructions (an abstract modality-free representation of the turning actions). Generation of context-specific route directions is a four-step process. In the first step, for each decision point, all possible abstract turn instructions are generated. These instructions are then combined by applying simple syntactic chunking rules. The chunks generated this way are validated in the third step. In this postprocessing, it is ensured that chunks adhere to general chunking principles that may, for example, be derived from the route direction principles identified beforehand. Finally, the fourth step is an optimization process that selects the chunks that result in the optimal context-specific route directions. Optimality depends on the chosen optimization criterion.

Guard is flexible with respect to the chunking and optimization principles. I demonstrated how different principles for good route directions found in the spatial cognition literature, can be integrated in the generation process in Section 4.5. I suggested, and argued for, a sensible optimization criterion in Section 4.4.4. The aim is to generate route directions with a minimal number of chunks, such that the used elements are on the coarsest level of granularity possible. This criterion is used in Guard's evaluation.

Chapter 5 provided details on the algorithmic realization of Guard. I introduced the underlying data structure—an embedded graph of the path-network annotated with information on landmarks, for example. I then presented methods for implementing the four steps of the generation process, concentrating especially on the handling of landmarks and on the optimization step. I discussed two different optimization approaches—local and global optimization. While global optimization guarantees finding the optimal solution, local optimization is computationally less complex.

The computational evaluation presented in Chapter 6 showed that local and global optimization result in equally good context-specific route directions in almost all cases. Thus, even though local optimization does not guarantee finding the optimal solution, in practical terms, it almost always does. Most of the differences between local and global optimization can be attributed to the underspecified handling of landmark references in Guard (see also Section 7.2.3). Context-specific route directions have also been subject to an exploratory empirical evaluation in an informal screen-based study. The results indicate that people are able to find their way using context-specific route directions. While there is no significant difference in performance, subjects prefer context-specific instead of route-planner route directions.

7.1.1 Contributions of This Thesis

In this section, I look back at the contributions of my thesis listed in Section 1.3, and discuss whether the claims made are justified.

The analysis of direction giving principles made in Chapters 2 and 3, provides a concise compilation of properties and principles of both routes and route directions. This analysis focuses on the wayfinder's perspective, and clearly shows the connections between cognitive and representation-theoretic aspects, and the consequences the one has for the other. Linking cognitive with representation-theoretic properties allows for bridging the gap between human abilities and cognitive principles, and more abstract, formal aspects of route following and route directions, which is especially helpful for integrating these properties in computational approaches. The analysis results in a systematics of route direction elements that is the basis for the generation of context-specific route directions. The systematics nicely illustrates different kinds of knowledge connected to the elements, and different levels of granularity with respect to the route at hand. It also provides a compact representation of this information which allows for easy access, transfer to other approaches, and extension of the systematics based on new research results.

Based on this analysis, I developed GUARD, a process for the generation of context-specific route directions. GUARD comprises methods for automatically handling different kinds of references to a variety of features that may provide relevant information for route following. This includes especially a sophisticated handling of different types of landmarks and how to address them in route directions. Except for global landmarks, any kind of landmark can be dealt with based on ordering information. This provides a solid framework for handling landmarks that is easily refinable and extensible to incorporate further ways of addressing the landmarks used or, if need be, further landmark types.

The implementation of GUARD is able to produce route directions as has been demonstrated in the computational evaluation by successfully generating directions for arbitrary routes. This shows that seeing the production of route directions as an optimization problem is a valid approach. A main advantage of such an approach is that modularity comes quite naturally, which is reflected in the generation process. It consists of four steps, one of them being the optimization step. These steps correspond to individual modules in the system's architecture. This modularity results in great flexibility in all major parts of the generation process. It allows combining different principles of initial chunking with different rules that define valid chunks, with different optimization criteria that define what good route directions are. Due to this modularity of the proposed architecture, GUARD may well serve as a test-bed for further empirical studies that, for example, compare different principles of route directions (see Section 7.2.6).

7.2 Possible Future Research

In my thesis I focused on the principle approach and methods for context-specific route directions, i.e. on how to account for cognitive and representation-theoretic considerations in automatically generating route directions. There are a couple of possible enhancements, and further applications, that can be pursued in the future.

7.2.1 Overview Information on a Route

The presented principles and methods for generating context-specific route directions can be adapted to produce overview information on a route to take. The route directions discussed so far provide information on how to proceed for every decision point along a route. This information is needed to correctly execute route following, i.e. to get from origin to destination along the specified route. However, in some situations, a coarse overview on the route is sufficient or desirable; for example in planning a route (especially comparing different alternatives of route options), or in an initial examination of a route that allows for familiarization before detailed information is presented. Such coarse route directions are suited for initial, quick information on a route; they reflect the planning level in wayfinding (cf. Timpf et al., 1992). They are not only desirable for providing an overview on a route, but may also be more adequate when providing wayfinding assistance in partially known environments (Tomko & Winter, 2006a; Schmid & Richter, 2006). Here, a wayfinder does not need detailed, turn-by-turn instructions for the already known parts of a route. Instead instructions like “go to the main station, I’ll guide you from there” suffice in this case (Schmid & Richter, 2006).

Coarse route directions allow a wayfinder to get an initial idea on what to expect along the route, without bothering her with details on how to actually execute route following. Hence, conceptualization of such coarse route directions can also just be coarse and leaves many parts of the route underdetermined. Coarse route directions do not guarantee that a wayfinder follows exactly the intended route, i.e. the route a direction giver may have in mind, or the one determined by a computational system. Segmentation of coarse route directions is done at major reorientation points, not at decision points. Reorientation points mark those points along a route where a change of actions occurs; for instance switching from following a river to going in the direction of a tower visible in the distance. Reorientation points divide a route into regions. The regions comprise the area between two reorientation points; each instruction of a coarse route direction covers one such region. Consequently, coarse route directions guide a wayfinder from one region to another without fixing a specific route between those regions. If a wayfinder just relies on these coarse directions, it is up to the wayfinder to fill the gaps with her own decisions on the exact route to take (see Wiener & Mallot, 2003, for an overview and principles of human region-based wayfinding).

In order to generate coarse route directions, the major reorientation points along a route and their accompanying regions need to be identified. The reorientation points are determined using the elements of the systematics (see Section 3.4) that are best suited for coarse route directions. Looking at the systematics, those elements on coarser levels of granularity, i.e. those that abstract from single decision point/action pairs to a great extent are good candidates for providing overview information on a route. Accordingly, elements of the first two levels of the systematics—the level of global references and the level of environmental structure—are used. On the level of path and route, the elements distant landmark, linear landmark, areal landmark, and path annotation are used as they also strongly abstract from single decision point/action pairs and allow chunking an unspecified number of instructions into a single one (see Section 3.4.7). All these elements require an end qualifier, i.e. additional information on when the corresponding action, for instance following a linear landmark, ends and a change of action occurs.

Reaching such an environmental feature (like a distant landmark),[1] or deviating from the direction towards a feature, indicates a reorientation point. Accordingly, the generation of coarse route directions comprises of identifying *regions of equal directedness*. This directedness is relative to some feature, i.e. to one of the route directions' elements. Examples include "follow the river" or "go in direction of the TV tower." In the former example a linear landmark induces the directedness—'keep next to the river'—in the latter, a distant landmark sets the direction—'lessen your distance to the tower and keep it in front of you.' The generation also involves finding proper end qualifiers, i.e. deciding when the next reorientation point is reached, and strategies on how to ignore minor deviations from the current directedness that do not need to result in a new reorientation point. The latter requires some heuristics that allow leaving parts of a route unconsidered, and generating directions that are not necessarily unambiguous.

This approach to generating coarse overview information on a route has already been outlined in Richter (2005, 2007).

7.2.2 Externalization of Context-Specific Route Directions

Generating context-specific route directions with the process presented in this thesis results in an abstract specification of the actions to be performed while route following. This specification needs to be externalized in order to be understandable to a human wayfinder, i.e. to become usable wayfinding assistance.

[1] Which, when reaching it, is not distant anymore, of course.

I already took first steps towards externalization. As explained in Section 6.2.2, I collaborated with Prof. John Bateman on generating verbal route directions based on the abstract specification. To this end, the entities referred to in the context-specific route directions are grounded in the common-sense ontology of outdoor objects developed in Prof. Bateman's group. Using their linguistic expression generation component to verbalize context-specific route directions works well already, as can be seen in the study's results. However, there are still some open questions, for example, with respect to which information contained in the abstract turn instructions should be explicitly verbalized and which may be left implicit. Pursuing these questions is already planned; the collaboration with Prof. Bateman will continue as part of the work going on in the SFB/TR 8. Projects I2-[MapSpace] and I5-[DiaSpace] will jointly work on research efforts which are, at least partly, based on context-specific route directions.

Graphical externalization of context-specific route directions, on the other hand, is still an open issue. As the data underlying the generation process is a coordinate-based representation of the environment, this can be exploited in presenting route directions graphically. That is, this information can be used to, for example, link the route directions with the path-network's coordinates. This way, problems arising from spatial underspecification of the abstract route directions can be avoided. A major challenge, however, is visualizing chunks such that they keep their characteristics, i.e. maintain a compact, combinational representation of actions to be performed for several consecutive decision points. One approach may be to not present a coherent map, but to split the graphical presentation into a sequence of sketches, each showing one chunk. To appropriately cover the different elements of the systematics, a graphical 'vocabulary' may be called for, which allows for covering the different ways of representing a route-following action (this is comparable to the toolkit approach by Tversky & Lee, 1999, but focusing on the functional perspective).

Furthermore, of special interest is combining different modalities, especially verbal and graphical externalization. This has two main advantages. First, some information communicated in route directions is easier to communicate and understand using verbal directions, while other is more suited for graphical presentation. Second, combining both modalities may allow for reducing the explicitly communicated information even more; thus, also reducing the wayfinder's cognitive load. By combining both modalities, it may be possible to counter the ambiguity and underspecification often connected to verbal expressions with an illustrating graphical representation, which, at the same time, can be kept simple and schematic as it is enriched with verbal information.

7.2.3 Refining the Landmark Model

Landmarks play a crucial role in good route directions. Guard accounts for this by incorporating different landmark types and different ways of referring to landmarks. However, determining suitable candidates for landmarks from all environmental features and deciding which of these candidates to choose is realized only to a limited extend.

A sensible first extension is to take a landmark's saliency into account. Integrating a saliency model for landmarks, for example the one of Raubal & Winter (2002) (cf. also Winter, 2003; Klippel & Winter, 2005), is straightforward. In the implemented system, each landmark already has a saliency value attached; currently, this value is fixed at 1. Given that the data required for determining a landmark's saliency is available, a further step can be added to Guard. This step would determine the saliency of each landmark candidate by taking into account structural and functional characteristics of a route and, thereby, assigning discriminating saliency values to each landmark. These values could then be used to select landmarks when generating abstract turn instructions. Saliency may also become part of an optimization criterion, i.e. selecting abstract turn instructions based on the referred landmarks' saliency.

Such a saliency model does not only requires detailed data on each landmark candidate, it first of all requires a set of landmark candidates. That is, while a saliency model allows for deciding which landmark from a given set of candidates is most suitable, it does not generate this candidate set itself. Getting a set of landmark candidates requires further complex preprocessing of the spatial data at hand. This preprocessing—while necessary for a truly automatic generation of route directions—is independent of a specific route and, therefore, can and should be performed separately from and previous to generating route directions. A given data-set can be processed once to identify all potential landmarks and to store them in a candidate set (cf. Elias, 2003). When generating route directions, landmarks are chosen from this set according to their applicability to the given route.

Furthermore, the inclusion of structural landmarks, i.e. references to salient configurations of intersections can be extended. Prime candidates are forks in the road, i.e. intersections where the road branches off to the left and right at approximately 45 degrees, and roundabouts. The former can be handled like T-intersections—an intersection's configuration is checked against a model of a fork intersection and, if applicable, the direction relation is adapted accordingly. The latter, roundabouts, require a different approach. Here, instead of using egocentric direction relations, counting branches is more appropriate, as in "take the second exit" (cf. Klippel et al., 2005a). Accordingly, references to roundabouts require determining the outgoing branch's number relative to the incoming one, i.e. to implement a concept of counting which again can be based on ordering information.

7.2.4 Instructions on Origin and Destination

Guard focuses on generating optimized instructions for actions to be performed at decision points. Correctly executing these turning actions is crucial for successfully following a route. Likewise, it is important to launch a wayfinder in the right direction at the origin and to provide information for identifying the destination location (cf. Michon & Denis, 2001). At the moment, the process does not provide any special means for generating such instructions.

The generated route leads a wayfinder to the destination. Accordingly, there is no need for a specific instruction that explains how to reach it. However, the destination's location relative to the route should be determined and integrated in the route directions as a special confirmation annotation. Providing information on the direction at the origin, on the other hand, requires a specific instruction. This instruction is, ideally, independent of a wayfinder's orientation or previous movement direction and needs to unambiguously identify the direction to take. For such instructions, references to landmarks play an important role.

Finally, for better integration in ongoing research, representing context-specific route directions in the XML schema specified in Cognitive OpenLS (cf. Hansen et al., 2006) is a worthwhile extension that is planned in the near future. This way, context-specific route directions may become more easily processed by other systems.

7.2.5 From Best Route Directions to Best Route

Context-specific route directions optimize the descriptional and conceptual complexity for a given route, i.e. they provide the best instructions for a specific route. Other approaches, like the one of Timpf & Heye (2002) and, especially, the one of Duckham & Kulik (2003), try to find the optimal route by taking into account navigational complexity, but do not consider descriptional complexity.

Guard can be extended to account for both descriptional and navigational complexity. This, however, requires clever algorithms since generating context-specific route directions for every possible route between origin and destination is computationally complex. This can be somewhat reduced by applying the same approach used in global optimization (see Section 5.4.3.2)—partial solutions can be dropped if they are already worse than the current optimum due to the monotony of the complexity function. But to significantly reduce this complexity, heuristics are needed that identify promising (sub-)routes prior to generating directions for them. This may involve combining measures for the complexity of route directions with measures for the complexity of routes, i.e. measures of their structural and functional aspects.

7.2.6 Using GUARD as a Test-Bed for Further Studies

GUARD is devised such that it is flexible with respect to which cognitive or representation-theoretic principles to adhere to in the generation process. Throughout my thesis I proposed principles for chunking and optimization I deem sensible, which are also implemented in the presented system. To evaluate whether GUARD produces route directions that are comprehensible to human wayfinders, and to test the proposed principles, I conducted the study presented in Section 6.2. The results indicate that this is the case.

As already discussed, the empirical evaluation of GUARD can be continued in a more rigorous way, for example, by having people follow routes in a real-world setting. Such an experiment is much more complex and time-consuming to perform, but would also lead to clearer results.

Furthermore, the process' flexibility with respect to the considered route direction principles allows for the generation of a variety of directions for a single route. This way, it can serve as a test-bed for empirical studies concerned with route following. This includes studies utilizing a simulated cognitive agent performing wayfinding tasks, as well as human-subject studies. It is, for example, possible to implement a cognitive agent that navigates through a virtual path-network by following these instructions and to evaluate its performance, especially by looking at the places it goes wrong and, by systematically varying these places, to identify problems and new ways to adapt route directions to the environment at hand. Also, different chunking or optimization criteria can be evaluated and ranked by generating instructions, and then having human subjects follow a route using these different instructions.

Further, the kind and amount of information available in the underlying data of the environment can be systematically varied. This way, it can be evaluated which consequences this has for the generation of route directions and how wayfinders perform with these directions. The question is whether GUARD shows some kind of gentle degrade, or whether there is a rapid drop in performance in the generation as well as in the utility of the route directions if certain information is not available anymore.

7.2.7 Bridging the Gap Between Top-Down and Bottom-Up Route Directions

GUARD generates route directions for every decision point of a route. In that, it starts with generating instructions for single decision points and then combining them to complete route directions. Such an approach can be considered to be bottom-up as its initial focus is on the smallest element—the decision point. The resulting route directions are turn-by-turn directions providing information on every turning action needed to reach the destination. As has been outlined, these route directions can be coarsened to provide an overview on a route. Still, the generation of coarse route directions is based on generating instructions for individual decision points; it is still bottom-up.

The approach presented by Tomko & Winter (2006a,b), on the other hand, can be considered to be top-down. Here, the focus is on the destination of a route. The aim is to provide information on its location at the coarsest granularity level possible, and then to refine this information the closer the destination gets. This approach is based on a hierarchy of regions that partition the environment. At some point, a switch to turn-by-turn directions is required—the latest when reaching the region on the lowest level of the hierarchy in which the destination is located (Tomko & Winter, 2006b).

Both approaches share the same goal—providing a wayfinder with instructions on how to reach her destination—and, despite coming from different directions, are able to generate instructions at about the same level of granularity. Still, there is a gap between both, i.e. a transition from one approach to the other is currently not possible. Accordingly, the question arises how can these approaches be combined? That is, how can the transition from top-down to bottom-up (from destination-based to turn-by-turn) directions be made automatically and vice versa, and what are the consequences for communicating route information?

Ideally, this transition should happen in a seamless way, i.e. such that it does not result in significant cognitive costs for the wayfinder. Combining both approaches requires identifying when such a transition is appropriate, i.e. which information is more helpful in which context. Also a mapping between the decision point-based route representation and the region-based hierarchical representation of the environment is needed. Such a mapping allows, for example, determining the first (decision) point to start with turn-by-turn directions when switching from destination-based directions. Such a switch may also involve providing instructions without precisely knowing where the wayfinder will be when switching (cf. also Duckham et al., 2003).

With respect to communication, the question is whether this transition needs to be announced explicitly, or whether it can be done implicitly without the wayfinder getting confused.

7.3 Concluding Remarks

As my thesis has demonstrated, an automatic generation of route directions that rely on cognitive principles is possible. The link between cognitive research and representation theory has proven fruitful and is well worth being pursued further. It allows a structured handling of different kinds of features and information in a coherent, simple, and adaptable way.

Today, many different (computational) approaches to route directions, or more generally wayfinding assistance, exist. These approaches deal with different aspects of wayfinding assistance on different levels. Often, most notable is the difference between in-advance and incremental route directions, and the mode of transportation under consideration. Still, common to most of these approaches is that they need extensive data in order to achieve their (full) potential; it is especially true if they deal with landmarks. This also holds true for GUARD, even though it is also able to produce route directions based just on a simple path-network.

Accordingly, sophisticated data handling is called for in order to exploit all possibilities that these advanced, cognitively inspired approaches offer, while keeping computational complexity and runtime at an acceptable level. Also, a good understanding of the kinds of schematization of the (perceived) information that humans apply while giving and receiving route directions, as well as while navigating in an environment, improves such approaches tremendously. Knowing about these processes allows restricting the data to deal with only that, which is relevant for the task at hand. Also it allows for the presentation of information in a way that it is easy for humans to process.

Finally, since all the different approaches concentrate on specific issues regarding wayfinding assistance, they are quite naturally, especially good at these issues. For example, some approaches focus on identifying and extracting landmarks from a given data-set while others focus on how to communicate best the information needed for route following. Bridging all these approaches in a coherent framework would result in a powerful system covering the complete process of generating wayfinding assistance, from determining the relevant pieces of data, to communicating the required information in a cognitively adequate way. I hope that my work is one step in that direction.

Bibliography

AdV (2003). *Amtliches Topographisch-Kartographisches Informationssystem ATKIS. ATKIS Objektartenkatalog.* Arbeitsgemeinschaft der Vermessungsverwaltungen der Länder der Bundesrepublik Deutschland. http://www.adv-online.de/veroeffentlichungen/atkis_index.htm [07.11.2003].

Agrawala, M. & Stolte, C. (2000). A design and implementation for effective computer-generated route maps. In *Proceedings of the AAAI Spring Symposium on Smart Graphics*. Stanford, California, USA.

Agrawala, M. & Stolte, C. (2001). Rendering effective route maps: Improving usability through generalization. In *SIGGRAPH 2001*, (pp. 241–249). Los Angeles, California, USA: ACM Press.

Allen, G. L. (1997). From knowledge to words to wayfinding: Issues in the production and comprehension of route directions. In S. C. Hirtle & A. U. Frank (eds.), *Spatial Information Theory. A Theoretical Basis for GIS*, (pp. 363–372). International Conference COSIT, Berlin: Springer. LNCS 1329.

Allen, G. L. (2000). Principles and practices for communicating route knowledge. *Applied Cognitive Psychology*, 14(4):333–359.

Allen, G. L. (2003). Gestures accompanying verbal route directions: Do they point to a new avenue for examining spatial representation? *Spatial Cognition and Computation*, 3(4):259–268.

Anderson, J. R. (1999). *Cognitive Psychology and Its Implications.* New York: Worth Publishers.

Appleyard, D. (1969). Why buildings are known – a predictive tool for architects and planners. *Environment and Behavior*, 1:131–156.

Arthur, P. & Passini, R. (1992). *Wayfinding: People, Signs, and Architecture.* Toronto: McGraw-Hill Ryerson.

Barkowsky, T. & Freksa, C. (1997). Cognitive requirements on making and interpreting maps. In S. C. Hirtle & A. U. Frank (eds.), *Spatial Information Theory. A Theoretical Basis for GIS*, (pp. 347–361). International Conference COSIT, Berlin: Springer. LNCS 1329.

Bateman, J. A., Henschel, R., & Rinaldi, F. (1995). Generalized upper model 2.0: Documentation. *Tech. rep.*, GMD/Institut für Integrierte Publikations- und Informationssysteme, Darmstadt, Germany.
URL `http://purl.org/net/gum2`

Berendt, B., Barkowsky, T., Freksa, C., & Kelter, S. (1998). Spatial representation with aspect maps. In C. Freksa, C. Habel, & K. F. Wender (eds.), *Spatial Cognition: an Interdisciplinary Approach to Representing and Processing Spatial Knowledge*, (pp. 157–175). Berlin: Springer. LNAI 1404.

Caduff, D. & Timpf, S. (2005). The landmark spider: Representing landmark knowledge for wayfinding tasks. In T. Barkowsky, C. Freksa, M. Hegarty, & R. Lowe (eds.), *Reasoning with mental and external diagrams: computational modeling and spatial assistance - Papers from the 2005 AAAI Spring Symposium*, (pp. 30–35). Menlo Park, CA.

Carlson, L. A. (1999). Selecting a reference frame. *Spatial Cognition and Computation*, 1:365–379.

Couclelis, H. (1996). Verbal directions for way-finding: Space, cognition and language. In J. Portugali (ed.), *The Construction of Cognitive Maps*, (pp. 133–153). Kluwer Academic Publishers.

Couclelis, H., Golledge, R. G., Gale, N., & Tobler, W. (1987). Exploring the anchor-point hypothesis of spatial cognition. *Journal of Environmental Psychology*, 7:99–122.

Cowan, N. (2001). The magical number 4 in short-term memory: A reconsideration of mental storage capacity. *Behavioral and Brain Sciences*, 24(1):87–186.

Craik, F. I. M. & Lockhart, R. S. (1972). Levels of processing: A framework for memory research. *Journal of Verbal Learning and Verbal Behavior*, 11:671–684.

Dale, R. & Reiter, E. (1995). Computational interpretations of the Gricean maxims in the generation of referring expressions. *Cognitive Science*, 19.

Dale, R., Geldof, S., & Prost, J.-P. (2002). Generating more natural route descriptions. In *Proceedings of 2002 Australasian Natural Language Processing Workshop*, (pp. 41–48). Canbera, Australia.

Dale, R., Geldof, S., & Prost, J.-P. (2003). Coral: Using natural language generation for navigational assistance. In M. J. Oudshoorn (ed.), *Proceedings of 26th Australasian Computer Science Conference (ACSC2003)*, (pp. 35–44). Adelaide, Australia.

Dale, R., Geldof, S., & Prost, J.-P. (2005). Using natural language generation in automatic route description. *Journal of Research and Practice in Information Technology*, 37(1):89–105.

Daniel, M.-P. & Denis, M. (1998). Spatial descriptions as navigational aids: A cognitive analysis of route directions. *Kognitionswissenschaft*, 7:45–52.

Daniel, M.-P., Tom, A., Manghi, E., & Denis, M. (2003). Testing the value of route directions through navigational performance. *Spatial Cognition and Computation*, 3(4):269–289.

Denis, M. (1997). The description of routes: A cognitive approach to the production of spatial discourse. *Cahiers Psychologie Cognitive*, 16(4):409–458.

Denis, M., Pazzaglia, F., Cornoldi, C., & Bertolo, L. (1999). Spatial discourse and navigation: An analysis of route directions in the city of Venice. *Applied Cognitive Psychology*, 13:145–174.

Dey, A. K. (2001). Understanding and using context. *Personal and Ubiquitous Computing*, 5(1):4–7.

Dijkstra, E. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271.

Dogu, U. & Erkip, F. (2000). Spatial factors affecting wayfinding and orientation — a case study in a shopping mall. *Environment and Behavior*, 32(6):731–755.

Duckham, M. & Kulik, L. (2003). "Simplest" paths: Automated route selection for navigation. In W. Kuhn, M. Worboys, & S. Timpf (eds.), *Spatial Information Theory. Foundations of Geographic Information Science*, (pp. 169–185). International Conference COSIT, Berlin: Springer. LNCS 2825.

Duckham, M., Kulik, L., & Worboys, M. (2003). Imprecise navigation. *GeoInformatica*, 7(3).

Elias, B. (2003). Extracting landmarks with data mining methods. In W. Kuhn, M. Worboys, & S. Timpf (eds.), *Spatial Information Theory. Foundations of Geographic Information Science*, (pp. 375–389). International Conference COSIT, Berlin: Springer. LNCS 2825.

Elias, B. & Sester, M. (2003). Landmarken für Wegbeschreibungen — Identifikation, Extraktion und Visualisierung. *Kartographische Nachrichten*, 2:51–57.

Eschenbach, C. (1999). Geometric structures of frames of references and natural language semantics. *Spatial Cognition and Computation*, 1(4):329–348.

Eschenbach, C., Tschander, L., Habel, C., & Kulik, L. (2000). Lexical specifications of paths. In C. Freksa, W. Brauer, C. Habel, & K. F. Wender (eds.), *Spatial Cognition II – Integrating Abstract Theories, Empirical Studies, Formal Methods, and Practical Applications*, (pp. 127–144). Berlin: Springer. LNAI 1849.

Fischer, K. & Moratz, R. (2001). From communicative strategies to cognitive modelling. In *Proceedings of the First International Workshop on 'Epigenetic Robotics'*.

Fraczak, L. (1995). From route descriptions to sketches: A model for a text-to-image translator. In *33rd Annual Meeting of the Association for Computational Linguistics (ACL)*, (pp. 299–301). San Francisco: Morgan Kaufmann Publishers.

Fraczak, L. (1998). Generating "mental maps" from route descriptions. In P. Olivier & K.-P. Gapp (eds.), *Representation and Processing of Spatial Expressions*, (pp. 185–200). Mahwah NJ, USA: Lawrence Erlbaum Associates.

Frank, A. U. (1992). Qualitative spatial reasoning about distances and directions in geographic space. *Journal of Visual Languages and Computing*, 3:343–371.

Frank, A. U. (2003). Pragmatic information content – how to measure the information in a route description. In M. Goodchild, M. Duckham, & M. Worboys (eds.), *Perspectives on Geographic Information Science*, (pp. 47–68). London: Taylor and Francis.

Freksa, C. (1991). Qualitative spatial reasoning. In D. M. Mark & A. U. Frank (eds.), *Cognitive and Linguistic Aspects of Geographic Space*, (pp. 361–372). Dodrecht: Kluwer Academic Publishers.

Freksa, C. (1992). Using orientation information for qualitative spatial reasoning. In A. U. Frank, I. Campari, & U. Formentini (eds.), *Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*, (pp. 162 – 178). Berlin: Springer. LNCS 639.

Freksa, C. (1999). Spatial aspects of task-specific wayfinding maps - a representation-theoretic perspective. In J. S. Gero & B. Tversky (eds.), *Visual and Spatial Reasoning in Design*, (pp. 15–32). University of Sidney: Key Centre of Design Computing and Cognition.

Freksa, C. & Röhrig, R. (1993). Dimensions of qualitative spatial reasoning. In N. Piera Carreté & M. G. Singh (eds.), *Qualitative Reasoning in Decision Technologies, Proceedings of QUARDET'93*, (pp. 483–492). CIMNE Barcelona.

Gapp, K.-P. (1995). Object localization: Selection of optimal reference objects. In A. U. Frank & W. Kuhn (eds.), *Spatial Information Theory. A Theoretical Basis for GIS*, (pp. 193–206). International Conference COSIT, Berlin: Springer. LNCS 988.

Gärling, T., Lindberg, E., & Mäntylä, T. (1983). Orientation in buildings: Effects of familiarity, visual access, and orientation aids. *Journal of Applied Psychology*, 68(1):177–186.

Gärling, T., Böök, A., & Linberg, E. (1986). Spatial orientation and wayfinding in the designed environment – a conceptual analysis and some suggestions for postoccupancy evaluation. *Journal of Architectural and Planning Research*, 3:55–64.

Geldof, S. & Dale, R. (2002). Improving route directions on mobile devices. In *Proceedings of the ISCA Workshop on Multi-modal Dialogue in Mobile Environments*. Kloster Irsee, Germany.

Golledge, R. G. (1999). Human wayfinding and cognitive maps. In R. G. Golledge (ed.), *Wayfinding Behavior — Cognitive Mapping and Other Spatial Processes*, (pp. 5–46). Baltimore, USA: John Hopkins University Press.

Grice, H. P. (1975). Logic and conversation. In P. Cole & J. L. Morgan (eds.), *Speech Acts*, vol. 3 of *Syntax and Semantics*, (pp. 41–58). New York: Academic Press.

Guhe, M., Habel, C., & Tschander, L. (2003). Describing motion events: Incremental representations for incremental processing. In M. Guhe, C. Habel, & L. Tschander (eds.), *Proceedings of the 5th International Workshop on Computational Semantics (IWCS-5)*, (pp. 410–424). Tillburg.

Habel, C. (1988). Prozedurale Aspekte der Wegplanung und Wegbeschreibung. In H. Schnelle & G. Rickheit (eds.), *Sprache in Mensch und Computer: kognitive und neuronale Sprachverarbeitung*, (pp. 107–133). Opladen: Westdeutscher Verlag.

Habel, C. (2003). Incremental generation of multimodal route instructions. In *Natural Language Generation in Spoken and Written Dialogue*. Palo Alto, CA: AAAI Spring Symposium 2003.

Halford, G. S., Wilson, W. H., & Phillips, S. (1998). Processing capacity defined by relational complexity: Implications for comparative, developmental, and cognitive psychology. *Behavioral and Brain Sciences*, 21(6):803–865.

Hansen, S., Richter, K.-F., & Klippel, A. (2006). Landmarks in OpenLS - a data structure for cognitive ergonomic route directions. In M. Raubal, H. Miller, A. U. Frank, & M. F. Goodchild (eds.), *Geographic Information Science - Fourth International Conference, GIScience 2006*, (pp. 128–144). Springer; Berlin. LNCS 4197.

Haque, S., Kulik, L., & Klippel, A. (to appear). Algorithms for reliable navigation and wayfinding. In *Proceedings of Spatial Cognition 2006*. Berlin: Springer.

Herrmann, T., Schweizer, K., Janzen, G., & Katz, S. (1998). Routen- und Überblickswissen — konzeptuelle Überlegungen. *Kognitionswissenschaft*, 7(4):145–159.

Herskovits, A. (1998). Schematization. In P. Olivier & K.-P. Gapp (eds.), *Representation and Processing of Spatial Expressions*, (pp. 149–162). Mahwah NJ, USA: Lawrence Erlbaum Associates.

Heye, C. & Timpf, S. (2003). Factors influencing the physical complexity of routes in public transportation networks. In K. Axhausen (ed.), *10th International Conference on Travel Behaviour Research*. Lucerne, Switzerland.

Hirtle, S. C. & Jonides, J. (1985). Evidence of hierarchies in cognitive maps. *Memory & Cognition*, 13(3):208–217.

Hobbs, J. R. (1985). Granularity. In A. Joshi (ed.), *Proceedings of 9th International Joint Conference on Artificial Intelligence*, (pp. 432–435). San Francisco: Morgan Kaufmann.

Höök, K. (1991). *An Approach to a Route Guidance Interface*. Licentiate thesis, Dept. of Computer and System Sciences, Stockholm University.

Huttenlocher, J., Hedges, L. V., & Duncan, S. (1991). Categories and particulars: Prototype effects in estimating spatial location. *Psychological Review*, 98:352–377.

Jackendoff, R. (1990). *Semantic Structures*. Cambridge, MA: MIT Press.

Klippel, A. (2003a). Wayfinding choremes. In W. Kuhn, M. Worboys, & S. Timpf (eds.), *Spatial Information Theory. Foundations of Geographic Information Science*, (pp. 320–334). International Conference COSIT, Berlin: Springer. LNCS 2825.

Klippel, A. (2003b). *Wayfinding Choremes — Conceptualizing Wayfinding and Route Direction Elements*. Ph.D. thesis, Universität Bremen.

Klippel, A. & Winter, S. (2005). Structural salience of landmarks for route directions. In A. Cohn & D. Mark (eds.), *Spatial Information Theory*, (pp. 347–362). International Conference COSIT, Berlin: Springer. LNCS 3693.

Klippel, A., Tappe, H., & Habel, C. (2003). Pictorial representations of routes: Chunking route segments during comprehension. In C. Freksa, W. Brauer, C. Habel, & K. F. Wender (eds.), *Spatial Cognition III – Routes and Navigation, Human Memory and Learning, Spatial Representation and Spatial Learning*, (pp. 11–33). Berlin: Springer. LNAI 2685.

Klippel, A., Dewey, C., Knauff, M., Richter, K.-F., Montello, D. R., Freksa, C., & Loeliger, E.-A. (2004). Direction concepts in wayfinding assistance systems. In J. Baus, C. Kray, & R. Porzel (eds.), *Workshop on Artificial Intelligence in Mobile Systems (AIMS'04)*, (pp. 1–8). SFB 378 Memo 84; Saarbrücken.

Klippel, A., Hansen, S., Davies, J., & Winter, S. (2005a). A high-level cognitive framework for route directions. In *Proceedings of the SSC 2005 Spatial Intelligence, Innovation and Praxis: The National Bienneal Conference of the Spatial Science Institute*, (pp. 311–320).

Klippel, A., Richter, K.-F., Barkowsky, T., & Freksa, C. (2005b). The cognitive reality of schematic maps. In L. Meng, A. Zipf, & T. Reichenbacher (eds.), *Map-based Mobile Services - Theories, Methods and Implementations*, (pp. 57–74). Berlin: Springer.

Klippel, A., Richter, K.-F., & Hansen, S. (2005c). Structural salience as a landmark. In *Workshop Mobile Maps 2005*. Salzburg, Austria.

Klippel, A., Richter, K.-F., & Hansen, S. (2005d). Wayfinding choreme maps. In S. Bres & R. Laurini (eds.), *Visual Information and Information Systems. 8th International Conference, VISUAL 2005*, (pp. 94–108). Springer; Berlin. LNCS 3736.

Klippel, A., Tappe, H., Kulik, L., & Lee, P. U. (2005e). Wayfinding choremes — a language for modeling conceptual route knowledge. *Journal of Visual Languages and Computing*, 16(4):311–329.

Kray, C. & Blocher, A. (1999). Modeling the basic meanings of path relations. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI)*, (pp. 384–393). San Francisco: Morgan Kaufmann.

Kray, C., Baus, J., Zimmer, H., Speiser, H., & Krüger, A. (2001). Two path prepositions: Along and past. In D. R. Montello (ed.), *Spatial Information Theory. Cognitive and Computational Foundations of Geopraphic Information Science*, (pp. 263–277). International Conference COSIT, Berlin: Springer. LNCS 2205.

Krieg-Brückner, B., Röfer, T., Carmesin, H.-O., & Müller, R. (1998). A taxonomy of spatial knowledge for navigation and its application to the Bremen autonomous wheelchair. In C. Freksa, C. Habel, & K. F. Wender (eds.), *Spatial Cognition – An Interdisciplinary Approach to Representing and Processing Spatial Knowledge*, (pp. 373–397). Berlin: Springer. LNAI 1404.

Kuipers, B. (1978). Modelling spatial knowledge. *Cognitive Science*, 2:129–153.

Levinson, S. C. (1996). Frames of reference and Molyneux's question: Crosslinguistic evidence. In P. Bloom, M. A. Peterson, L. Nadel, & M. F. Garett (eds.), *Language and Space*, (pp. 109–169). Cambridge, MA: MIT Press.

Lovelace, K. L., Hegarty, M., & Montello, D. R. (1999). Elements of good route directions in familiar and unfamiliar environments. In C. Freksa & D. M. Mark (eds.), *Spatial Information Theory. Cognitive and Computational Foundations of Geopraphic Information Science*, (pp. 65–82). International Conference COSIT, Berlin: Springer. LNCS 1661.

Lynch, K. (1960). *The Image of the City*. Cambridge: The MIT Press.

Maaß, W. (1994). From visual perception to multimodal communication: Incremental route descriptions. *Artificial Intelligence Review Journal*, (pp. 68–82).

Maaß, W. (1995). How spatial information connects visual perception and natural language generation in dynamic environments: Towards a computational model. In A. U. Frank & W. Kuhn (eds.), *Spatial Information Theory. A Theoretical Basis for GIS*, (pp. 223–240). International Conference COSIT, Berlin: Springer. LNCS 988.

Marciniak, T. & Strube, M. (2004). Classification-based generation using TAG. In A. Belz, R. Evans, & P. Piwek (eds.), *Third International Conference on Natural Language Generation*, (pp. 100–109). Berlin: Springer. LNCS 3123.

Marciniak, T. & Strube, M. (2005a). Discrete optimization as an alternative to sequential processing in NLG. In *Proceedings of the 10th European Workshop on Natural Language Generation*, (pp. 101–108). Aberdeen, UK.

Marciniak, T. & Strube, M. (2005b). Modeling and annotating the semantics of route directions. In *Proceedings of the Sixth International Workshop on Computational Semantics (IWCS-6)*, (pp. 151–162). Tilburg, The Netherlands.

Mark, D. (1986). Automated route selection for navigation. *IEEE Aerospace and Electronic Systems Magazine*, 1:2–5.

May, A. J., Ross, T., Bayer, S. H., & Burnett, G. (2001). Using landmarks to enhance navigation systems: Driver requirements and industrial constraints. In *Proceedings of the 8th World Congress on Intelligent Transport Systems*. Sydney, Australia.

McNamara, T. P. (1992). Spatial representation. *Geoforum*, 23(2):139–150.

Meilinger, T. (2005). Wayfinding with maps and verbal directions. In *Proceedings of Twenty-Seventh Annual Conference of the Cognitive Science Society*, (pp. 1473–1478).

Meilinger, T. & Knauff, M. (2004). Nach dem Weg fragen oder Karte studieren, was ist besser? ein Feldexperiment. In D. Kerzel, V. Franz, & K. Gegenfurtner (eds.), *Experimentelle Psychologie. Beiträge zur 46. Tagung experimentell arbeitender Psychologen*, (p. 169). Lengerich: Pabst Science Publishers.

Michon, P.-E. & Denis, M. (2001). When and why are visual landmarks used in giving directions? In D. R. Montello (ed.), *Spatial Information Theory. Cognitive and Computational Foundations of Geopraphic Information Science*, (pp. 400–414). International Conference COSIT, Berlin: Springer. LNCS 2205.

Miller, G. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, 63:81–97.

Montello, D. R. (1993). Scale and multiple psychologies of space. In A. U. Frank & I. Campari (eds.), *Spatial Information Theory. A Theoretical Basis for GIS*, (pp. 312–321). European Conference COSIT, Berlin: Springer. LNCS 716.

Montello, D. R. (1998). A new framework for understanding the acquistion of spatial knowledge in large-scale environments. In M. J. Egenhofer &

R. G. Golledge (eds.), *Spatial and Temporal Reasoning in Geographic Information Systems*, chap. 11, (pp. 143–154). New York: Oxford University Press.

Montello, D. R. (2005). Navigation. In P. Shah & A. Miyake (eds.), *Handbook of Visuospatial Thinking*, (pp. 257–294). Cambridge: Cambridge University Press.

Montello, D. R. & Frank, A. U. (1996). Modeling directional knowledge and reasoning in environmental space: Testing qualitative metrics. In J. Portugali (ed.), *The Construction of Cognitive Maps*, (pp. 321–344). Kluwer Academic Publishers.

Moratz, R. & Fischer, K. (2000). Cognitively adequate modelling of spatial reference in human-robot interaction. In *Proceedings of the 12th IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*, (pp. 222–228). Vancouver, Canada: IEEE Computer Society.

Neumann, J. F. (2003). *Interpretation of ATKIS DLM Data and Usage of GPS Device for Map Alignment*. Diplomarbeit, Fachhochschule Wolfenbüttel.

Nicklas, D. & Mitschang, B. (2001). The NEXUS augmented world model: An extensible approach for mobile, spatially aware applications. In Y. Wang, S. Patel, & R. Johnston (eds.), *Proceedings of the 7th International Conference on Object-Oriented Information Systems: OOIS'01*, (pp. 392–401). Berlin: Springer.

Palmer, S. E. (1978). Fundamental aspects of cognitive representation. In E. Rosch & B. B. Lloyd (eds.), *Cognition and Categorization*, chap. 9, (pp. 259–303). Hillsdale, New Jersey: Lawrence Erlbaum Associates.

Pederson, E. (2003). How many reference frames? In C. Freksa, W. Brauer, C. Habel, & K. F. Wender (eds.), *Spatial Cognition III – Routes and Navigation, Human Memory and Learning, Spatial Representation and Spatial Learning*, (pp. 287–304). Berlin: Springer. LNAI 2685.

Phillips, R. J. (1979). Making maps easy to read: A summary of research. In P. A. Koler, M. E. Wrolstad, & H. Bouma (eds.), *Processing of Visual Language*, vol. 1, (pp. 165–174). New York: Plenum Press.

Presson, C. C. & Montello, D. R. (1988). Points of reference in spatial cognition: Stalking the elusive landmark. *British Journal of Developmental Psychology*, 6:378–381.

Raubal, M. & Winter, S. (2002). Enriching wayfinding instructions with local landmarks. In M. Egenhofer & D. Mark (eds.), *Geographic Infor-*

mation Science - Second International Conference GIScience 2002, (pp. 243–259). Berlin: Springer. LNCS 2478.

Richter, K.-F. (2005). "Cross the river and then follow the tracks": Overview information on the way to take. In G. Gartner (ed.), *Location Based Services & Telecartography - Proceedings of the Symposium 2005*, Geowissenschaftliche Mitteilungen, (pp. 191–195). TU Vienna.

Richter, K.-F. (2007). From turn-by-turn directions to overview information on the way to take. In G. Gartner, W. Cartwright, & M. P. Peterson (eds.), *Location Based Services and TeleCartography*, (pp. 205–214). Berlin: Springer.

Richter, K.-F. & Klippel, A. (2005). A model for context-specific route directions. In C. Freksa, M. Knauff, B. Krieg-Brückner, B. Nebel, & T. Barkowsky (eds.), *Spatial Cognition IV. Reasoning, Action, Interaction: International Conference Spatial Cognition 2004*, (pp. 58–78). Berlin: Springer. LNAI 3343.

Richter, K.-F. & Klippel, A. (2007). Before or after: Prepositions in spatially constrained systems. In T. Barkowsky, M. Knauff, G. Ligozat, & D. R. Montello (eds.), *Spatial Cognition V - Reasoning, Action, Interaction*, LNAI 4387, (pp. 453–469). Berlin: Springer.

Richter, K.-F., Klippel, A., & Freksa, C. (2004). Shortest, fastest, - but what next? a different approach to route directions. In M. Raubal, A. Sliwinski, & W. Kuhn (eds.), *Geoinformation und Mobilität - von der Forschung zur praktischen Anwendung. Beiträge zu den Münsteraner GI-Tagen 2004*, IfGIprints, (pp. 205–217). Institut für Geoinformatik, Münster.

Rock, I. (1992). Comment on Asch and Witkin's "Studies in space orientation.". *Journal of Experimental Psychology: General*, 121(4):404–406.

Rosch, E. (1978). Principles of categorization. In E. Rosch & B. B. Lloyd (eds.), *Cognition and Categorization*, chap. 2, (pp. 27–48). Hillsdale, New Jersey: Lawrence Erlbaum Associates.

Ross, T., May, A., & Thompson, S. (2004). The use of landmarks in pedestrian navigation instructions and the effects of context. In S. Brewster & M. Dunlop (eds.), *Mobile Computer Interaction – MobileHCI 2004*, (pp. 300–304). Berlin: Springer. LNCS 3160.

Sadalla, E. K. & Montello, D. R. (1989). Remembering changes in direction. *Environment and Behavior*, 21:346–363.

Sadalla, E. K., Burroughs, J., & Staplin, L. J. (1980). Reference points in spatial cognition. *Journal of Experimental Psychology: Human Learning and Memory*, 6(5):516–528.

Schlieder, C. (1995). Reasoning about ordering. In A. U. Frank & W. Kuhn (eds.), *Spatial Information Theory. A Theoretical Basis for GIS*, (pp. 341–349). International Conference COSIT, Berlin: Springer. LNCS 988.

Schmid, F. & Richter, K.-F. (2006). Extracting places from location data streams. In *UbiGIS 2006 - Second International Workshop on Ubiquitous Geographical Information Services*. Workshop at GIScience 2006.

Schneider, L. F. & Taylor, H. A. (1999). How to get there from here? Mental representations of route descriptions. *Applied Cognitive Psychology*, 13:415–441.

Schweizer, K., Herrmann, T., Janzen, G., & Katz, S. (1998). The route direction effect and its constraints. In C. Freksa, C. Habel, & K. F. Wender (eds.), *Spatial Cognition – An Interdisciplinary Approach to Representing and Processing Spatial Knowledge*, (pp. 19–38). Berlin: Springer. LNAI 1404.

Schweizer, K., Katz, S., & Janzen, G. (2000). Orientierung im Raum — kognitive Grundlagen und sprachliche Realisierung. *Tourismus Journal*, 4(1):79–104.

Shannon, C. E. & Weaver, W. (1949). *The Mathematical Theory of Communication*. Urbana, Illinois: The University of Illinois Press.

Shepard, R. N. & Hurwitz, S. (1984). Upward direction, mental rotation, and discrimination of left and right turns in maps. *Cognition*, 18:161–193.

Siegel, A. W. & White, S. H. (1975). The development of spatial representations of large-scale environments. In H. Reese (ed.), *Advances in Child Development and Behaviour*, (pp. 9–55). New York: Academic Press.

Sorrows, M. E. & Hirtle, S. C. (1999). The nature of landmarks for real and electronic spaces. In C. Freksa & D. M. Mark (eds.), *Spatial Information Theory - Cognitive and Computational Foundations of Geopraphic Information Science*, (pp. 37–50). International Conference COSIT, Berlin: Springer. LNCS 1661.

Talmy, L. (1983). How language structures space. In H. Pick (ed.), *Spatial Orientation Theory: Research and Application*, (pp. 225–282). New York: Plenum Press.

Tenbrink, T. (2005). Semantics and application of spatial dimensional terms in English and German. *Tech. Rep. 004-03/2005*, SFB/TR 8 Spatial Cognition, Universität Bremen / Universität Freiburg.

Timpf, S. (2002). Ontologies of wayfinding: A traveler's perspective. *Networks and Spatial Economics*, 2(1):9–33.

Timpf, S. & Heye, C. (2002). Complexity of routes in multi-modal wayfinding. In M. Egenhofer & D. Mark (eds.), *Geographic Information Science - Second International Conference GIScience 2002*. Extended Abstracts, Berlin: Springer. LNCS 2478.

Timpf, S., Volta, G. S., Pollock, D. W., Frank, A. U., & Egenhofer, M. J. (1992). A conceptual model of wayfinding using multiple levels of abstraction. In A. U. Frank, I. Campari, & U. Formentini (eds.), *Proceedings of Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*, (pp. 348–367). Berlin: Springer. LNCS 639.

Tom, A. & Denis, M. (2003). Referring to landmarks or street information in route directions: What difference does it make? In W. Kuhn, M. Worboys, & S. Timpf (eds.), *Spatial Information Theory. Foundations of Geographic Information Science*, (pp. 362–374). International Conference COSIT, Berlin: Springer. LNCS 2825.

Tomko, M. (2004). Case study – assessing spatial distribution of web resources for navigation services. In *4th International Workshop on Web and Wireless Geographical Information Systems W2GIS04*. Koyang, Korea.

Tomko, M. & Winter, S. (2006a). Identification of the initial entity in granular route directions. In A. Riedl, W. Kainz, & G. A. Elmes (eds.), *Progress in Spatial Data Handling. 12th International Symposium on Spatial Data Handling*, (pp. 43–60). Berlin: Springer.

Tomko, M. & Winter, S. (2006b). Recursive construction of granular route directions. *Journal of Spatial Science*, 51(1):101–115.

Tschander, L. B., Schmidtke, H. R., Habel, C., Eschenbach, C., & Kulik, L. (2003). A geometric agent following route instructions. In C. Freksa, W. Brauer, C. Habel, & K. F. Wender (eds.), *Spatial Cognition III – Routes and Navigation, Human Memory and Learning, Spatial Representation and Spatial Learning*, (pp. 89–111). Berlin: Springer. LNAI 2685.

Tversky, B. (1993). Cognitive maps, cognitive collages and spatial mental models. In A. U. Frank & I. Campari (eds.), *Spatial Information Theory. A Theoretical Basis for GIS*, (pp. 14–24). European Conference COSIT, Berlin: Springer. LNCS 716.

Tversky, B. & Lee, P. U. (1998). How space structures language. In C. Freksa, C. Habel, & K. F. Wender (eds.), *Spatial Cognition: An Interdisciplinary Approach to Representing and Processing Spatial Knowledge*, (pp. 157–175). Berlin: Springer. LNAI 1404.

Tversky, B. & Lee, P. U. (1999). Pictorial and verbal tools for conveying routes. In C. Freksa & D. M. Mark (eds.), *Spatial Information Theory*

- *Cognitive and Computational Foundations of Geopraphic Information Science*, (pp. 51–64). International Conference COSIT, Berlin: Springer.

Tversky, B., Lee, P., & Mainwaring, S. (1999). Why do speakers mix perspectives? *Spatial Cognition and Computation*, 1:399–412.

Weisman, J. (1981). Evaluating architectural legibility: Way-finding in the built environment. *Environment and Behaviour*, 13(2):189–204.

Wener, R. E. & Kaminoff, R. D. (1983). Improving environmental information — effects of signs on perceived crowding and behavior. *Environment and Behavior*, 15(1):3–20.

Werner, S. & Long, P. (2003). Cognition meets Le Corbusier – cognitive principles of architectural design. In C. Freksa, W. Brauer, C. Habel, & K. F. Wender (eds.), *Spatial Cognition III – Routes and Navigation, Human Memory and Learning, Spatial Representation and Spatial Learning*, (pp. 112–126). Berlin: Springer. LNAI 2685.

Werner, S., Krieg-Brückner, B., & Herrmann, T. (2000). Modelling navigational knowledge by route graphs. In C. Freksa, W. Brauer, C. Habel, & K. F. Wender (eds.), *Spatial Cognition II - Integrating Abstract Theories, Empirical Studies, Formal Methods, and Practical Applications*, (pp. 295–316). Berlin: Springer. LNAI 1849.

Wiener, J. M. & Mallot, H. A. (2003). 'Fine to coarse' route planning and navigation in regionalized environments. *Spatial Cognition and Computation*, 3(4):331–358.

Winter, S. (2003). Route adaptive selection of salient features. In W. Kuhn, M. Worboys, & S. Timpf (eds.), *Spatial Information Theory: Foundations of Geographic Information Science*. International Conference COSIT, Berlin: Springer. LNCS 2685.

Winter, S., Raubal, M., & Nothegger, C. (2005). Focalizing measures of salience for wayfinding. In L. Meng, A. Zipf, & T. Reichenbacher (eds.), *Map-based Mobile Services - Theories, Methods and Implementations*, (pp. 127–142). Berlin: Springer.

Zadeh, L. A. (1979). Fuzzy sets and information granularity. In M. M. Gupta, R. K. Ragade, & R. R. Yager (eds.), *Advances in Fuzzy Set Theory and Applications*, (pp. 3–18). New York: North-Holland Publishing Company.

List of Figures

List of Tables

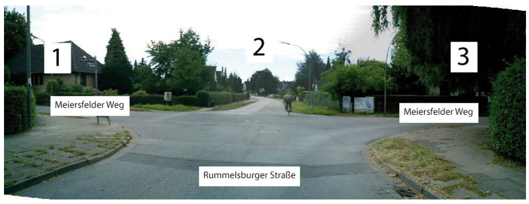
Appendix A

The Routes Used in the Empirical Evaluation

In this Appendix, I list the photographs and XML descriptions of the six routes used as the basic material in the empirical evalutation presented in Section 6.2. This list is to be read from left to right, top to bottom with the corresponding XML description underneath each photo. The photos are those presented to the subjects during the route following study.

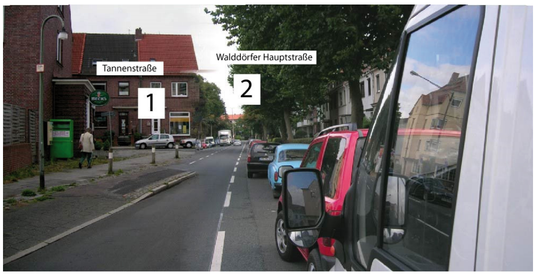
Route 1


```
<intersection id="0">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
</intersection>
```




```
<intersection id="1">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <branch no="3" direction="90"/>
</intersection>
```


```
<intersection id="2">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <branch no="3" direction="90"/>
 <landmark name="Ampel"
  position="0-1" type="routemark"/>
 <landmark name="Ampel"
  position="0-3" type="routemark"/>
 <landmark name="Polizei"
  position="2-3" type="routemark"/>
</intersection>
```

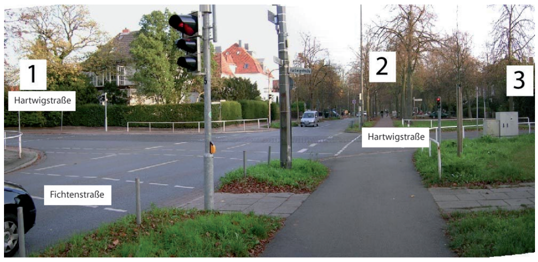



```
<intersection id="3">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <landmark name="Kneipe"
  position="0-1" type="routemark"/>
</intersection>
```

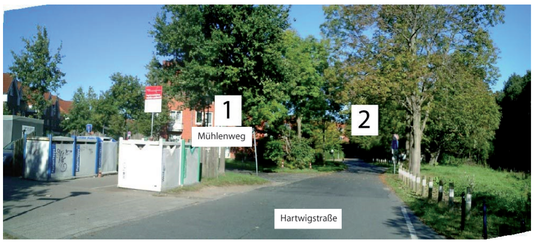

```
<intersection id="4">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <landmark name="Frisoer"
  position="2-1" type="routemark"/>
</intersection>
```


```
<intersection id="5">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <landmark name="Apotheke"
  position="0-1" type="routemark"/>
</intersection>
```


```
<intersection id="6">
 <branch no="1" direction="180"/>
 <branch no="2" direction="90"/>
 <landmark name="Minimal"
  position="0-2" type="routemark"/>
</intersection>
```

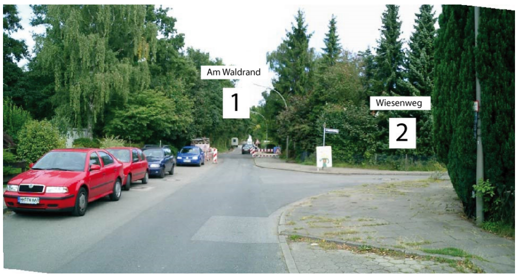



```
<intersection id="7">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <branch no="3" direction="90"/>
 <landmark name="Ampel"
  position="0-1" type="routemark"/>
 <landmark name="Ampel"
  position="0-3" type="routemark"/>
</intersection>
```




```
<intersection id="8">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <landmark name="Muellcontainer"
  position="0-1" type="routemark"/>
</intersection>
```

Route 2

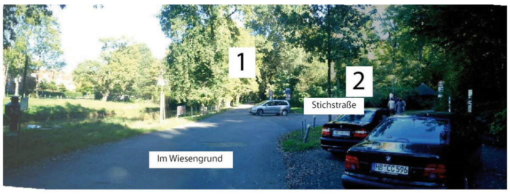



```
<intersection id="0">
 <branch no="1" direction="180"/>
 <branch no="2" direction="90"/>
 <landmark name="Baustelle"
  position="1-2" type="routemark"/>
</intersection>
```


```
<intersection id="1">
 <branch no="1" direction="190"/>
 <branch no="2" direction="90"/>
 <landmark name="Karte"
  position="1-2" type="routemark"/>
</intersection>
```

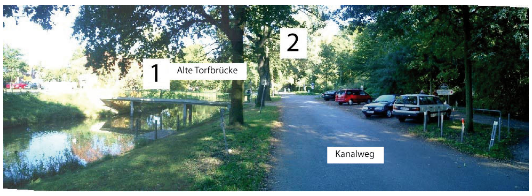

```
<intersection id="2">
 <branch no="1" direction="180"/>
 <branch no="2" direction="90"/>
 <landmark name="Fluss" position="0-1"
  type="linear" id="100" side="left"/>
</intersection>
```


```
<intersection id="3">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <landmark name="Fluss"
  position="0-1" type="edge"/>
 <landmark name="Fluss"
  position="0-2" type="linear"
  id="100" side="left"/>
</intersection>
```




```
<intersection id="4">
 <branch no="1" direction="180"/>
 <branch no="2" direction="90"/>
 <landmark name="Fluss"
  position="0-1" type="linear"
  id="100" side="left"/>
</intersection>
```

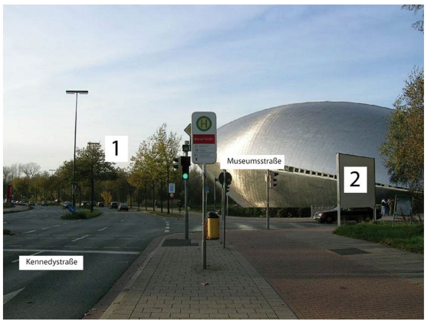

```
<intersection id="5">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <landmark name="Fluss"
  position="0-2" type="linear"
  id="100" side="left"/>
</intersection>
```




```
<intersection id="6">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <landmark name="Fluss"
  position="0-1" type="edge"/>
 <landmark name="Fluss"
  position="0-2" type="linear"
  id="100" side="left"/>
</intersection>
```


```
<intersection id="7">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <landmark name="Fluss"
  position="0-1" type="edge"/>
 <landmark name="Telekom Schild"
  position="0-1" type="routemark"/>
 <landmark name="Fluss"
  position="0-2" type="linear"
  id="100" side="left"/>
</intersection>
```

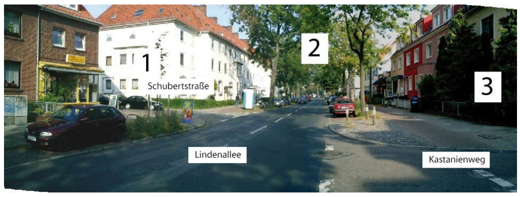

```
<intersection id="8">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <landmark name="Universum"
  position="0-2" distant="t"
  type="routemark"/>
 <landmark name="Fluss"
  position="1-2" type="routemark"/>
</intersection>
```



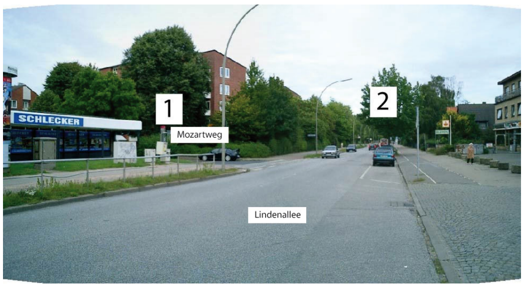

```
<intersection id="9">
 <branch no="1" direction="180"/>
 <branch no="2" direction="90"/>
 <landmark name="Universum"
  position="1-2" type="routemark"/>
 <landmark name="Bushaltestelle"
  position="0-2" type="routemark"/>
</intersection>
```

Route 3


```
<intersection id="0">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
</intersection>
```




```
<intersection id="1">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <branch no="3" direction="90"/>
 <landmark type="Restaurant"
  name="Venezia" position="0-1"
  c-type="routemark"/>
</intersection>
```



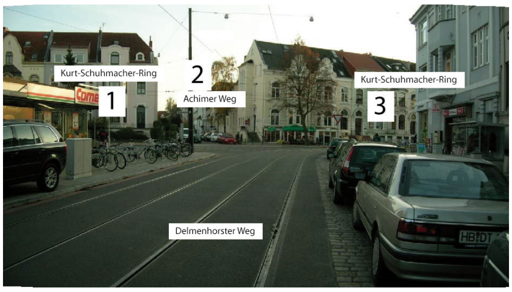

```
<intersection id="2">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <landmark type="Supermarket"
  name="Schlecker" position="0-1"
  c-type="routemark"/>
</intersection>
```


```
<intersection id="3">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <branch no="3" direction="90"/>
 <landmark type="Supermarket"
  name="Extra" position="2-3"
  c-type="routemark"/>
 <landmark type="TrafficLight"
  position="0-3" c-type="routemark"/>
 <landmark type="TrafficLight"
  position="0-1" c-type="routemark"/>
 <landmark type="SimpleTram"
  position="1-3" id="100" side="left"
  c-type="linear"/>
</intersection>
```

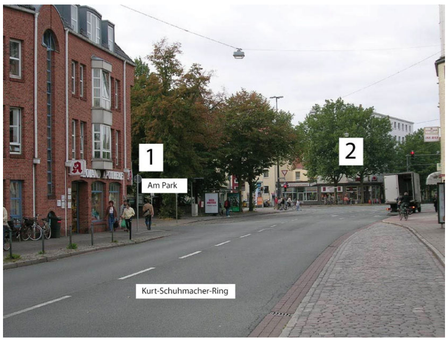

```
<intersection id="4">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <branch no="3" direction="90"/>
 <landmark type="Signage"
  name="gruenes Schild"
  position="2-3" c-type="routemark"/>
 <landmark type="ComplexJunction"
  name="Bahnuebergang" position="1-2"
  c-type="routemark"/>
 <landmark type="SimpleTram"
  position="0-2" id="100" side="left"
  c-type="linear"/>
</intersection>
```


```
<intersection id="5">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <branch no="3" direction="90"/>
 <landmark type="Supermarket"
  name="Extra" position="1-2"
  c-type="routemark"/>
 <landmark type="TrafficLight"
  position="0-1" c-type="routemark"/>
 <landmark type="TrafficLight"
  position="0-3" c-type="routemark"/>
 <landmark type="SimpleTram"
  position="0-2" id="100" side="left"
  c-type="linear"/>
</intersection>
```

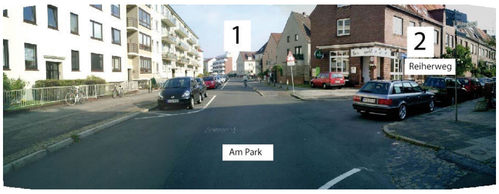

```
<intersection id="6">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <branch no="3" direction="90"/>
 <landmark type="SimpleTram"
  position="0-2" id="100" side="left"
  c-type="linear"/>
</intersection>
```




```
<intersection id="7">
 <branch no="1" direction="285"/>
 <branch no="2" direction="195"/>
 <branch no="3" direction="105"/>
 <landmark type="Supermarket"
  name="Comet" position="0-1"
  c-type="routemark"/>
 <landmark type="SimpleTram"
  position="0-3" id="100"
  side="left" c-type="linear"/>
 <landmark type="SimpleTram"
  position="1-3" id="101"
  side="right" c-type="linear"/>
</intersection>
```




```
<intersection id="8">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <landmark type="Pharmacy"
  position="0-1" c-type="routemark"/>
 <landmark type="BusStop"
  position="1-2" c-type="routemark"/>
</intersection>
```



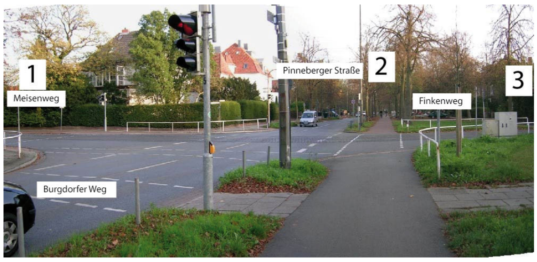

```
<intersection id="9">
 <branch no="1" direction="180"/>
 <branch no="2" direction="110"/>
</intersection>
```

Route 4


```
<intersection id="0">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <branch no="3" direction="90"/>
 <landmark type="Supermarket"
  name="Comet"
  position="1-2" c-type="routemark"/>
 <landmark type="SimpleTram"
  position="0-2" id="100" side="left"
  c-type="linear"/>
 <landmark type="SimpleTram"
  position="0-1" id="100" side="left"
  c-type="linear"/>
</intersection>
```

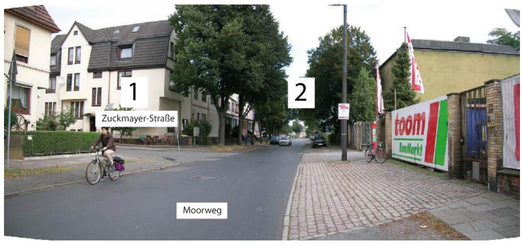

```
<intersection id="1">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <branch no="3" direction="90"/>
</intersection>
```


```
<intersection id="2">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <landmark type="River" position="0-2"
  id="101" side="left"
  c-type="linear"/>
</intersection>
```




```
<intersection id="3">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <branch no="3" direction="90"/>
 <landmark type="TrafficLight"
  position="0-3" c-type="routemark"/>
 <landmark type="TrafficLight"
  position="0-1" c-type="routemark"/>
</intersection>
```

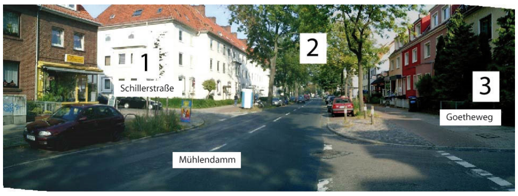

```
<intersection id="4">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <branch no="3" direction="90"/>
 <landmark type="Signage"
  name="gruenes Schild" position="2-3"
  c-type="routemark"/>
 <landmark type="ComplexJunction"
  name="Bahnuebergang" position="1-2"
  c-type="routemark"/>
 <landmark type="SimpleTram"
  position="0-2" id="100" side="left"
  c-type="linear"/>
</intersection>
```




```
<intersection id="5">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <landmark type="Supermarket"
  name="Toom" position="0-2"
  c-type="routemark"/>
</intersection>
```

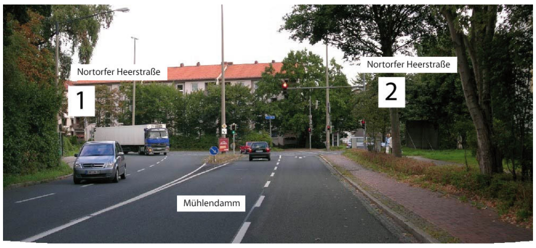

```
<intersection id="6">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <landmark type="DeadEndTrafficSign"
  position="1-2" c-type="routemark"/>
</intersection>
```

Route 5

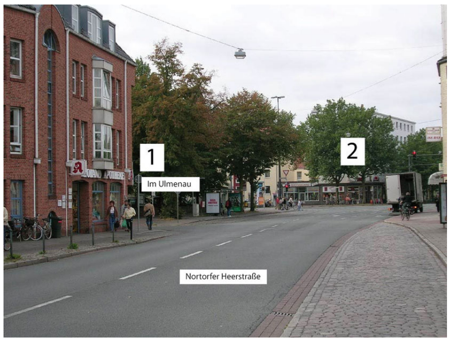



```
<intersection id="0">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <branch no="3" direction="90"/>
 <landmark type="Restaurant"
  name="Venezia" position="0-1"
  c-type="routemark"/>
</intersection>
```


```
<intersection id="1">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <landmark type="Pub"
  position="0-1"
  c-type="routemark"/>
 <landmark type="Barber"
  position="1-2"
  c-type="routemark"/>
</intersection>
```

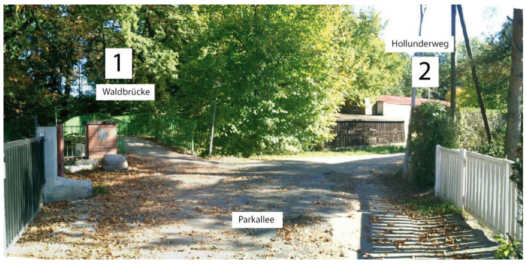



```
<intersection id="2">
 <branch no="1" direction="270"/>
 <branch no="2" direction="90"/>
 <landmark type="TrafficLight"
  position="0-2" c-type="routemark"/>
 <landmark type="TrafficLight"
  position="0-1" c-type="routemark"/>
</intersection>
```




```
<intersection id="3">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <landmark type="Pharmacy"
  position="0-1" c-type="routemark"/>
 <landmark type="BusStop"
  position="1-2" c-type="routemark"/>
</intersection>
```


```
<intersection id="4">
 <branch no="1" direction="270"/>
 <branch no="2" direction="90"/>
 <landmark type="ConstructionSite"
  position="1-2" c-type="routemark"/>
</intersection>
```




```
<intersection id="5">
 <branch no="1" direction="200"/>
 <branch no="2" direction="100"/>
</intersection>
```

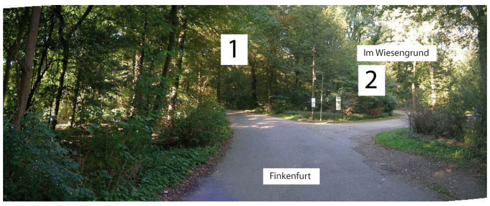

```
<intersection id="6">
 <branch no="1" direction="270"/>
 <branch no="2" direction="90"/>
 <landmark type="Church" position="1-2"
  c-type="routemark"/>
</intersection>
```


```
<intersection id="7">
 <branch no="1" direction="200"/>
 <branch no="2" direction="115"/>
 <landmark type="Shop"
  name="Fusspflegeladen"
  position="0-2"
  c-type="routemark"/>
</intersection>
```

Route 6

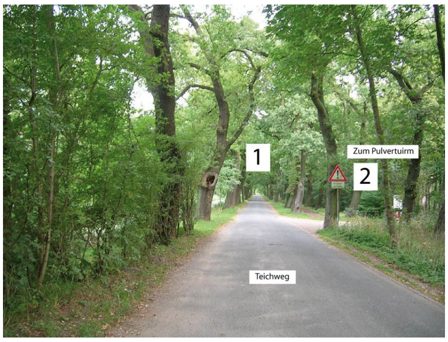

```
<intersection id="0">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <landmark type="Shop" name="Partyservice"
  position="0-1" c-type="routemark"/>
</intersection>
```


```
<intersection id="1">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <branch no="3" direction="90"/>
 <landmark type="TrafficLight"
  position="0-1" c-type="routemark"/>
 <landmark type="TrafficLight"
  position="0-3" c-type="routemark"/>
 <landmark type="PoliceStation"
  position="2-3" c-type="routemark"/>
</intersection>
```



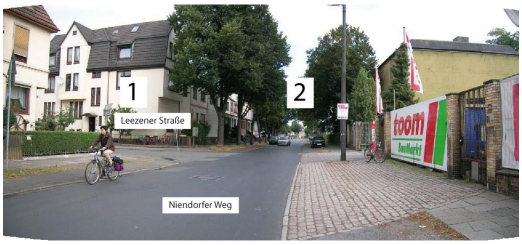

```
<intersection id="2">
 <branch no="1" direction="185"/>
 <branch no="2" direction="90"/>
 <landmark type="Map" position="1-2"
  c-type="routemark"/>
</intersection>
```


```
<intersection id="3">
 <branch no="1" direction="200"/>
 <branch no="2" direction="100"/>
</intersection>
```

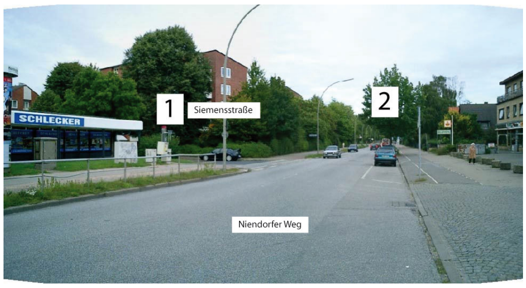



```
<intersection id="4">
 <branch no="1" direction="180"/>
 <branch no="2" direction="90"/>
</intersection>
```


```
<intersection id="5">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <branch no="3" direction="90"/>
</intersection>
```




```
<intersection id="6">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <landmark type="Supermarket"
  name="Toom" position="0-2"
  c-type="routemark"/>
</intersection>
```



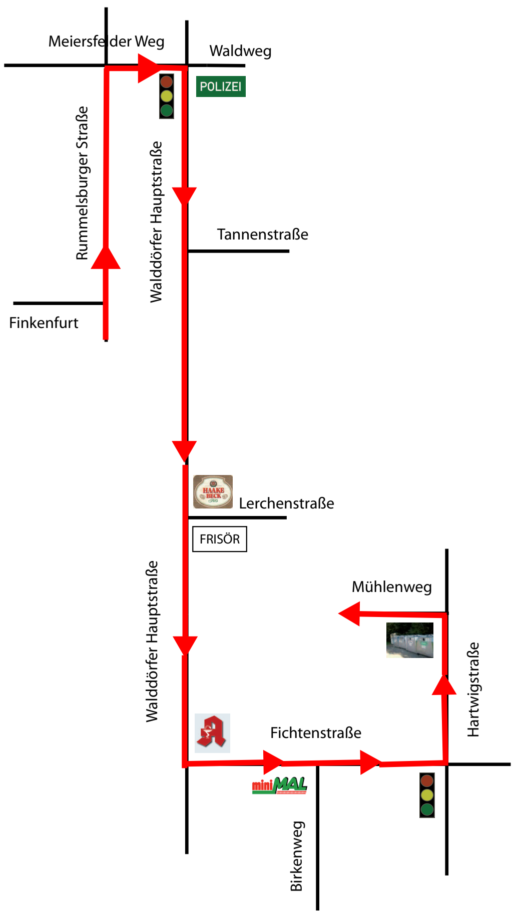

```
<intersection id="7">
 <branch no="1" direction="270"/>
 <branch no="2" direction="180"/>
 <landmark type="Supermarket"
  name="Schlecker" position="0-1"
  c-type="routemark"/>
</intersection>
```

Appendix B

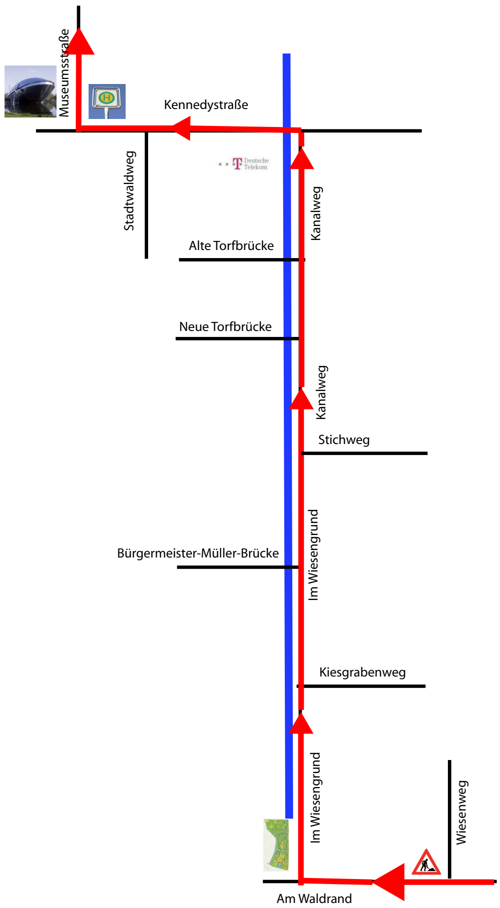
The Study Material

B.1 The Sketch Maps Used in Giving Route Directions

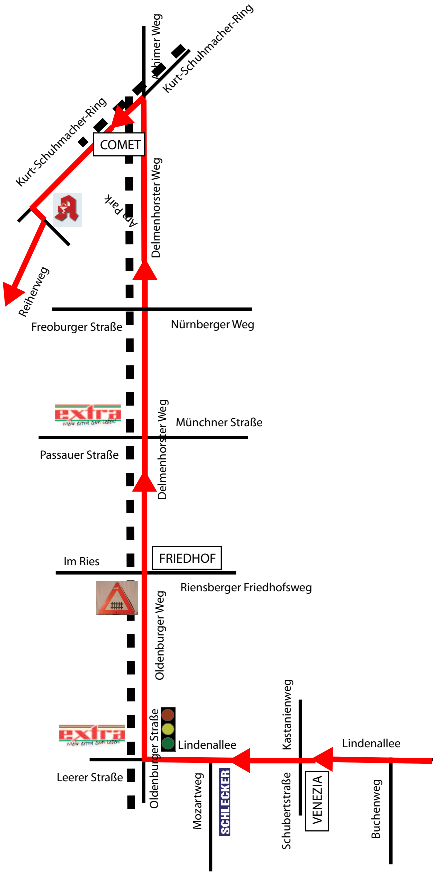
Route 1:



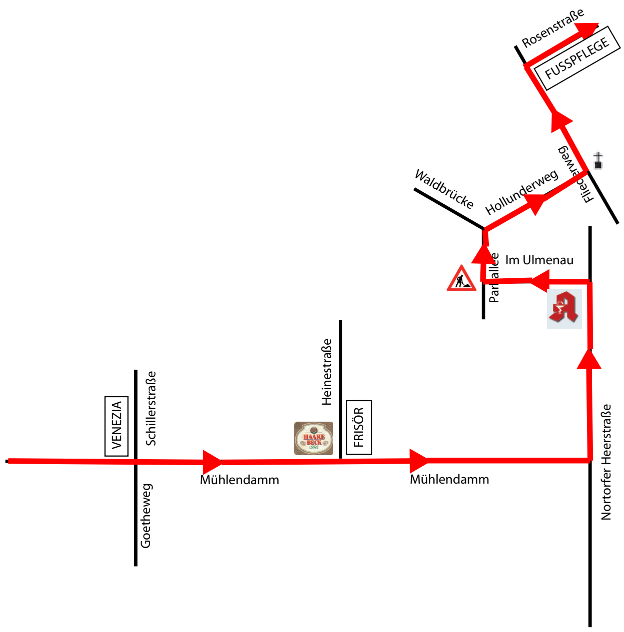
Route 2:


Museumsstraße
Kennedystraße
Stadtwaldweg
Kanalweg
Alte Torfbrücke
Neue Torfbrücke
Kanalweg
Stichweg
Bürgermeister-Müller-Brücke
Im Wiesengrund
Kiesgrabenweg
Im Wiesengrund
Wiesenweg
Am Waldrand

Route 3:


Kurt-Schuhmacher-Ring
Kurt-Schuhmacher-Ring
COMET
Delmenhorster Weg
Reiherweg
Freoburger Straße
Nürnberger Weg
extra
Münchner Straße
Passauer Straße
Delmenhorster Weg
FRIEDHOF
Im Ries
Riensberger Friedhofsweg
Oldenburger Weg
Kastanienweg
extra
Oldenburger Straße
Lindenallee
Lindenallee
Leerer Straße
Mozartweg
SCHLECKER
Schubertstraße
VENEZIA
Buchenweg

Route 5:


Rosenstraße
FUSSPFLEGE
Waldbrücke
Hollunderweg
Im Ulmenau
Heinestraße
VENEZIA
Schillerstraße
FRISÖR
Mühlendamm
Mühlendamm
Goetheweg
Nortorfer Heerstraße

B.2 The Route Directions Used in Rating and Route Following

B.2.1 The Context-Specific Route Directions

Listed are the context-specific route directions for each route as they are produced by GUARD and the verbal route directions generated by Prof. John Bateman. Every sentence of the verbal directions corresponds to one chunk of the abstract specifications.

Route 1

```
((9)     ((("left" ("LM" 9 "Dumbster") "lm<") 1.6)))
((7 8)   ((("straight") 1)
          (("left" ("LM" 7 "TrafficLight") "lm<") 1.6)))
((4 5 6) ((("straight") 1) (("straight") 1)
          (("left" ("LM" 5 "Pharmacy") "lm<") 1.6)))
((2 3)   ((("right") 1)
          (("right" ("LM" 0 "TrafficLight") "lm<") 1.6)))
((1)     ((("straight") 1)))
```

"Gehe geradeaus."
"Biege an der nächsten Kreuzung rechts ab und dann biege nach der Ampel wieder rechts ab."
"Gehe geradeaus und dann biege nach einer Apotheke links ab."
"Gehe an der nächsten Kreuzung geradeaus und dann biege nach der Ampel links ab."
"Biege nach einem Müllcontainer links ab."

Route 2

```
((9 10)  ((("towards" ("LM" 9 "SightSeeingObject" "Universum")
                       "" ("LM" 12 "BusStop")) 1.5)
          (("right" ("LM" 12 "BusStop") "lm<") 1.6)))
((3 4 5 6 7 8)
         ((("follow" ("LM" 100 "River") "") 1.7)
          (("follow" ("LM" 100 "River") "") 1.7)
          (("follow" ("LM" 100 "River") "") 1.7)
          (("follow" ("LM" 100 "River") "") 1.7)
          (("follow" ("LM" 100 "River") ""
           ("LM" 7 "Signage" "Telekom Schild")) 1.7)
          (("left" ("LM" 7 "Signage" "Telekom Schild")
                    "lm<") 1.6)))
((1 2)   ((("straight") 1)
          (("right" ("LM" 1 "Map") "lm>") 1.5)))
```

"Gehe an der nächsten Kreuzung geradeaus und dann biege vor einem Schild rechts ab."
"Gehe den Fluss entlang bis einem Telekom Schild und dann biege nach dem Telekom Schild links ab."
"Gehe an der nächsten Kreuzung zum Universum und dann biege nach der Bushaltestelle rechts ab."

Route 3

```
((10)    ((("right") 1)))
((9)     ((("left" ("LM" 17 "Pharmacy") "lm<") 1.6)))
((5 6 7 8)
         ((("follow" ("LM" 100 "SimpleTram") "") 1.7)
          (("follow" ("LM" 100 "SimpleTram") "") 1.7)
          (("follow" ("LM" 100 "SimpleTram") ""
           ("LM" 14 "Supermarket" "Comet")) 1.7)
          (("sharp left" ("LM" 14 "Supermarket" "Comet")
                          "lm<") 1.6)))
((1 2 3 4)
         ((("straight") 1) (("straight") 1) (("straight") 1)
          (("right" ("LM" 3 "TrafficLight") "lm<") 1.6)))
```

"Gehe geradeaus und dann biege nach der Ampel rechts ab."
"Gehe die Strassenbahnschienen entlang bis zu einem Comet Supermarkt und dann biege nach dem Comet Supermarkt links ab."
"Biege nach einer Apotheke links ab."
"Biege rechts ab."

Route 4

```
((7)     ((("right") 1)))
((5 6)   ((("left") 1)
          (("left" ("LM" 9 "Supermarket" "Toom") "lm<") 1.6))
          (("cross" 100 "SimpleTram")))
((1 2 3 4)
         ((("straight") 1) (("straight") 1) (("straight") 1)
          (("right" ("LM" 4 "TrafficLight") "lm<") 1.6)))
```

"Gehe geradeaus und dann biege nach der Ampel rechts ab."
"Biege an der nächsten Kreuzung links ab und dann biege nach einem Toom Supermarkt wieder links ab."
"Dabei überquerst du die Strassenbahnschienen."
"Biege rechts ab."

Route 5

```
((8)     ((("right" ("LM" 9 "Shop" "Fusspflegeladen")
                     "lm<") 1.6)))
((7)     ((("left" ("ST" "t-intersection")) 1.6)))
((5 6)   ((("right" ("ST" "t-intersection")) 1.6)
          (("right") 1)))
((4)     ((("left" ("LM" 5 "Pharmacy") "lm<") 1.6)))
((1 2 3) ((("straight") 1) (("straight") 1)
          (("left" ("ST" "t-intersection")) 1.6)))
```

"Gehe geradeaus und dann biege an der T-Kreuzung links ab."
"Biege nach einer Apotheke links ab."
"Biege an der nächsten Kreuzung rechts ab und dann biege wieder rechts ab."
"Biege an der T-Kreuzung links ab."
"Biege nach einem Fusspflegeladen rechts ab."

Route 6

```
((6 7 8) ((("straight") 1) (("straight") 1)
          (("left" ("LM" 6 "Supermarket" "Schlecker")
                    "lm<") 1.6)))
((5)     ((("right") 1)))
((3 4)   ((("half left" ("LM" 4 "Map") "lm>") 1.5)
          (("half left") 1)))
((1 2)   ((("straight" ("LM" 0 "Shop" "Partyservice")
                        "lm<") 1.5)
          (("straight") 1)))
```

"Gehe an der nächsten Kreuzung geradeaus und dann gehe wieder geradeaus."
"Biege vor einem Schild an der nächsten Kreuzung links ab und dann biege wieder links ab."
"Biege rechts ab."
"Gehe geradeaus und dann biege nach einem Schlecker Supermarkt links ab."

B.2.2 The Collected Route Directions

Route 1

Route directions used in the study:
Du gehst die Rummelsburger Straße immer geradeaus.
Irgendwann kommst du an eine Kreuzung, dort biegst du rechts in den Waldweg ein.
Dann gehst du an der nächsten Möglichkeit sofort wieder rechts in die Walddörfer Hauptstraße, dies ist eine etwas größere Straße.
Diese gehst du dann immer geradeaus, ganz lange.
Die 3. Straße biegst du dann links ab, an einer Apotheke, in die Fichtenstraße.
Hier gehst du dann auch wieder geradeaus, bis links die Hartwigstraße kommt, hier biegst du links ein.
Die erste Straße links ist dann der Mühlenweg.

The other collected directions:
Rummelsburger Straße geradeaus bis zur nächsten Kreuzung.
Meiersfelder Weg rechts bis zur Ampel und wieder rechts in Walddörfer Hauptstraße.
Geradeaus bis "Apotheke" auf linker Seite.
Hier links ab in die Fichtenstraße.
Vorbei an "Minimal" bis zur Ampel geradeaus.
Links ab in Hartwigstraße und nächste Straße wieder links hinter Containerstandort in Mühlenweg einbiegen.

Biege an der nächsten Kreuzung rechts ab und dann sofort die nächste wieder rechts auf die Waldörfer Hauptstraße.
Immer geradeaus, direkt hinter der Apotheke links in die Fichtenstraße.
Dann die 1. links in die Hartwigstraße und sofort wieder link in den Mühlenweg.

Route 2

Route directions used in the study:
Durquere eine kleine Baustelle,
biege an der nächsten Möglichkeit rechts ab.
Fahre solange am Bach entlang, bis es geradeaus nicht mehr weiter geht.
Überquere den Bach hier links in die Kennedystraße.
Direkt vor dem Metallufo biegst du rechts in die Museumstraße ab.

The other collected directions:
Folge der Straße “Am Waldrand”, lasse den Wiesenweg rechts liegen und biege dann danach rechts in die Straße “Im Wiesengrund” ein.
Folge jetzt dem Straßenverlauf immer am Kanal entlang. Der “Im Wiesengrund” wird später zum “Kanalweg”.
Am Ende des “Kanalweges” (an der Telekom) biegst du links in die “Kennedystraße” ein.
Folge dieser bis zum auffälligen silbernen Gebäude auf der rechten Seite.
Biege vor dem Gebäude rechts in die “Museumsstraße” ein!
Herzlich willkommen!

Du gehst erst geradeaus, die zweite Möglichkeit biegst du rechts in die Straße “Im Wiesengrund” ein.
Diese gehst du dann ganz lange geradeaus, immer am Fluss entlang.
Irgendwann kommst du an ein Telekomschild. Hier biegst du dann links in die Kennedystraße ein.
Die erste Möglichkeit rechts ist dann die Museumsstraße.

Route 3

Route directions used in the study:
Du gehst zuerst geradeaus. Auf der linken Seite siehst du das Eiscafé Venezia und ein Stück weiter kommt Schlecker.
Du gehst aber weiter geradeaus und biegst an der Ampel rechts ab (in die Oldenburger Straße).
Dann gehst du wieder ein längeres Stück geradeaus... am Straßenbahnübergang und an einem “Extra-Markt” (auf der linken Seite) vorbei.
Wenn du links “Comet” siehst, überquerst du die Straße und gehst hinter “Comet” links in den Kurt-Schumacher-Ring.
An der Apotheke dann links
und die nächste sofort wieder rechts.

The other collected directions:
Lindenallee geradeaus an “Schlecker” vorbei bis zur Ampelkreuzung und hier rechts in Oldenburger Weg.
Geradeaus Richtung Bahnübergang bis “Comet”.
Hier links in Kurt-Schuhmacher-Ring und nächste Straße hinter “Apotheke” wieder links in Am Park und erste Straße wieder rechts in Reiher Weg.

Auf der Straße geradeaus
immer geradeaus bis zu der großen Kreuzung (Oldenburger Str.)
Dort rechts ab und immer geradeaus bis zur Gabelung Kurt-Schumacher-Ring.
nicht rechts den Straßenbahnschienen folgen sondern hinter Comet links in den Kurt-Schumacher-Ring einbiegen
Hinter der Apotheke links einbiegen (Am Park)
am Reiherweg rechts einbiegen.

Route 5

Route directions used in the study:
Folge dem Mühlendamm bis zur Ampel. Auf der linken Seite liegt eine Kneipe und ein Frisör, beides lässt du links liegen.
An der Ampel biegst du nach links in die Nortorfer Heerstraße ein.
Folge dieser bis zu einer Apotheke auf der linken Seite. Biege direkt hinter der Apotheke links ab. Achtung! Dies ist eine Einbahnstraße, in die du mit dem Auto nicht hineinfahren darfst!
Folge dieser Straße (im Ulmenau) bis zum Ende. Biege dann rechts in die Parkallee ein. Achtung Baustelle!
Rechts halten, in den Hollerweg einbiegen,
und Hollerweg bis zum Ende folgen, dann links in den Fliederweg einbiegen.
Nach der Fußpflege auf der rechten Seite rechts in die Rosenstraße einbiegen
Achtung! Auch dies ist wieder eine Einbahnstraße!

The other collected directions:
Du gehst geradeaus bis zu einer T-Kreuzung.
Du kannst nicht weiter geradeaus und biegst links in die Nortorfer Heerstraße ein.
Links siehst du eine Apotheke und an der biegst du links ab.
Dann gehst du auf eine Baustelle zu und gehst an der Baustelle rechts.
Ein Stück weiter gabelt sich der Weg. Links ist eine Brücke — du gehst aber rechts in den Holunderweg, dann weiter bis zur Kirche.
An der Kirche links und dann hinter dem roten Backsteinhaus mit der Fußpflege rechts in die Rosenstraße.

geradeaus — auf der Straße bleiben bis die Straße auf die Querstraße (Nortorfer Heerstraße) stößt,
dort links abbiegen
hinter der Apotheke links rein (im Ulmenau) bis du auf die Querstraße Waldbrücke / Hollunderweg stößt — dort rechts in den Hollunderweg einbiegen
bis du auf den Fliederweg stößt dort links einbiegen
an der nächsten Gabelung rechts (neben der Fußpflege) in die Rosenstraße einfahren

B.2.3 The Route-Planner-Style Route Directions

Route 1

Folgen Sie der Rummelsburger Straße für 537m.
Biegen Sie nach rechts in den Meiersfelder Weg. Folgen Sie ihm für 46m.
Biegen Sie nach rechts in die Walddörfer Hauptstraße. Folgen Sie ihr für 2083m.
Biegen Sie nach links in die Fichtenstraße. Folgen Sie ihr für 612m.
Biegen Sie nach links in die Hartwigstraße. Folgen Sie ihr für 269m.
Biegen Sie nach links in den Mühlenweg. Sie haben Ihr Ziel erreicht.

Route 2

Folgen Sie der Straße Am Waldrand für 387m.
Biegen Sie nach rechts in die Straße Im Wiesengrund. Folgen Sie der Straße für 2710m.
Die Straße ändert ihren Namen in Kanalweg.
Biegen Sie nach links in die Kennedystraße. Folgen Sie der Straße für 473m.
Biegen Sie nach rechts in die Museumsstraße. Sie haben ihr Ziel erreicht.

Route 3

Folgen Sie der Lindenallee für 988m.
Biegen Sie rechts in die Oldenburger Straße. Folgen Sie der Straße für 3420m.
Der Name ändert sich in Oldenburger Weg
Der Name ändert sich in Delmenhorster Weg
Biegen Sie nach links in den Kurt-Schuhmacher-Ring. Folgen Sie ihm für 164m.
Biegen Sie nach links in die Straß Am Park. Folgen Sie ihr für 7m.
Biegen Sie nach rechts in den Reiherweg. Sie haben Ihr Ziel erreicht.

Route 4

Folgen Sie der Burgdorfer Hauptstraße für 1350m.
Der Name ändert sich in Burgdorfer Weg.
Biegen Sie nach rechts in den Finkenweg. Folgen Sie ihm für 89m.
Biegen Sie nach links in den Moorweg. Folgen Sie ihm für 67m.
Biegen Sie nach links in die Zuckmayer-Straße. Folgen Sie ihr für 35m.
Biegen Sie nach rechts in den Moehrricke-Weg. Sie haben Ihr Ziel erreicht.

Route 5

Folgen Sie dem Mühlendamm für 780m.
Biegen Sie nach links in die Nortorfer Heerstraße Folgen Sie ihr für 387m.
Biegen Sie nach links in die Straße Im Ulmenau. Folgen Sie ihr für 85m.
Biegen Sie nach rechts in die Parkallee. Folgen Sie ihr für 27m.
Biegen Sie nach rechts in den Hollunderweg. Folgen Sie ihm für 154m.
Biegen Sie nach links in den Fliederweg. Folgen Sie ihm für 270m.
Biegen Sie nach rechts in die Rosenstraße. Sie haben Ihr Ziel erreicht.

Route 6

Folgen Sie der Naher Ringstraße für 265m.
Geradeaus weiter in die Finkenfurt. Folgen Sie ihr für 489m.
Biegen Sie nach links in den Teichweg. Folgen Sie ihm für 43m.
Biegen Sie nach rechts in die Straße Zum Pulverturm. Folgen Sie ihr für 1876m.
Der Name ändert sich in Niendorfer Weg.
Biegen Sie nach links in die Siemensstraße. Sie haben Ihr Ziel erreicht.

Index