

Titel der Arbeit:
Multi-criteria decision support
for the planning of multimodal itineraries

Schriftliche Promotionsleistung
zur Erlangung des akademischen Grades
Doctor rerum politicarum
vorgelegt und angenommen
an der Fakultät für Wirtschaftswissenschaft
der Otto-von-Guericke Universität Magdeburg

Verfasser: Thomas Horstmannshoff

Geburtsdatum und -ort: 11.04.1989, Glandorf

Arbeit eingereicht am: 21.08.2023

Gutachter der schriftlichen Promotionsleistung:

Prof. Dr. Marlin Wolf Ulmer

Prof. Dr. Jan Fabian Ehmke

Prof. Dr. Dirk C. Mattfeld

Datum der Disputation: 27.10.2023

Publications

- Horstmannshoff, T., Ehmke, J. F. & Ulmer, M. W. (2023). **Dynamic learning-based search for multi-criteria itinerary planning** *Submitted to Omega. 1st round*
- Horstmannshoff, T. & Ehmke, J. F. (2023). **Predicting Pareto fronts to guide smart sampling of multimodal itineraries** *Submitted to OR Spectrum. 2nd round (major revision)*
- Horstmannshoff, T. & Redmond, M. F. (2023). **Identifying alternative stops for first and last mile urban travel planning** *Submitted to Public Transport. 3rd round (major revision)*
- Horstmannshoff, T. & Ehmke, J. F. (2022). **Traveler-oriented multi-criteria decision support for multimodal itineraries**. *Transportation Research Part C: Emerging Technologies*, 141, 103741. doi: 10.1016/j.trc.2022.103741
- Horstmannshoff, T. (2022). **Mobility-as-a-Service-Plattformen – Berücksichtigung von komplexen Reisendenanforderungen mittels nutzerorientierter Algorithmen**. In M. Bruhn & K. Hadwich (Eds.), *Smart Services* (pp. 523–546). [S.l.]: GABLER. doi: 10.1007/978-3-658-37346-7 19

List of presentations

- Horstmannshoff, T., Ehmke, J. F. & Ulmer, M. W. **Learning and predicting Pareto fronts of multi-criteria itineraries** Transportation Science and Logistics Society Conference (TSL 2023), Chicago, USA, July 2023
- Horstmannshoff, T., Ehmke, J. F. & Ulmer, M. W. **Traveler-oriented multi-criteria decision support for multimodal itineraries** Doktorandenworkshop Nordost (DoWoNo 2023), Wittenberg, Germany, May 2023
- Horstmannshoff, T. & Ehmke, J. F. **Multi-criteria decision support system for multimodal travel itineraries** 15th International Conference on Advanced Systems in Public Transport (CASPT 2022), Tel Aviv, Israel, November 2022
- Horstmannshoff, T. & Ehmke, J. F. **Smart sampling for traveler-oriented multi-criteria itineraries** 32nd European Conference On Operational Research (EURO 2022), Espoo, Finland, July 2022
- Horstmannshoff, T. & Ehmke, J. F. **Solution-based sampling of multimodal travel itineraries** INFORMS Annual Meeting (INFORMS 2020), Digital conference, November 2020
- Horstmannshoff, T. & Ehmke, J. F. **Considering multiple preferences in searching multimodal travel itineraries** Symposium on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2020), Digital conference, September 2020
- Horstmannshoff, T. & Ehmke, J. F. **Creation of individual sets of multimodal travel itineraries** Faculty Research Seminar, Magdeburg, Germany, January 2020
- Horstmannshoff, T. & Ehmke, J. F. **Creation of individual sets of multimodal travel itineraries** KUNO-Workshop, Braunschweig, Germany, December 2019
- Horstmannshoff, T. & Ehmke, J. F. **Creation of individual sets of multimodal travel itineraries** 22nd EURO Working Group on Transportation Meeting (EWGT 2019), Barcelona, Spain, September 2019
- Horstmannshoff, T. & Ehmke, J. F. **Considering complex customer preferences in multimodal travel itineraries** Doktorandenworkshop Nordost (DoWoNo 2019), Wittenberg, Germany, May 2019

Abstract

The importance of planning multimodal itineraries has grown in recent years. To adequately support travelers in their decision-making process and encourage the usage of multimodal mobility, it is crucial to fulfill their high expectations concerning state-of-the-art routing platforms. They expect that multiple preferences such as travel time, price, and number of transfers are incorporated into the search. As it is challenging for the traveler to assess the relevance of individual preferences on the complex multimodal solution space, they desire to choose from a set of diverse options provided by an integrated and multimodal mobility platform.

Satisfying these expectations is challenging for mobility platform providers. To offer a seamless multimodal door-to-door mobility experience to travelers, a wide range of services have to be integrated into a multimodal setting. Moreover, taking multiple traveler preferences with competing characteristics into account requires multi-criteria decision support.

In this thesis, we contribute to the field of multi-criteria decision support for planning multimodal itineraries while considering complex traveler requirements. First, we conduct a systematic literature review to gain comprehensive insights into the complex traveler requirements concerning mobility platforms. Subsequently, we propose multi-criteria decision support frameworks to incorporate specific traveler requirements while planning multimodal itineraries. We provide additional value for the traveler by identifying a set of alternative stops within walking distance by taking route and stop-based information into account. Additionally, we develop an effective scalable framework that enables the integration of a vast variety of individual preferences into the search and propose two approaches to enhance this framework further. By embedding the proposed frameworks into state-of-the-art mobility platforms, higher traveler orientation can be achieved.

Zusammenfassung

Die Planung von multimodalen Reiserouten hat in den letzten Jahren an Bedeutung gewonnen. Um die Nutzung multimodaler Mobilität zu fördern sowie Reisende angemessen bei ihrer Entscheidungsfindung zu unterstützen, ist es entscheidend, die hohen Erwartungen der Reisenden an aktuelle Mobilitätsplattformen zu erfüllen. Sie erwarten, dass verschiedene Präferenzen wie Fahrzeit, Preis und Anzahl der Umstiege in die Suche einbezogen werden. Da es für den Reisenden herausfordernd ist, die Auswirkungen individueller Präferenzen auf den komplexen multimodalen Lösungsraum zu bewerten, erwarten Reisende aus einer Vielzahl von Optionen wählen zu können, die ihm von einer integrierten und multimodalen Mobilitätsplattform zur Verfügung gestellt werden.

Die Erfüllung dieser Erwartungen stellt eine Herausforderung für Anbieter von Mobilitätsplattformen dar. Um den Reisenden ein nahtloses multimodales Tür-zu-Tür-Mobilitätsenerlebnis zu bieten, gilt es eine Vielzahl von Mobilitätsdiensten in eine multimodale Umgebung zu integrieren. Darüber hinaus erfordert die Berücksichtigung mehrerer Reisendenpräferenzen mit konkurrierenden Eigenschaften die Notwendigkeit einer multikriteriellen Entscheidungsunterstützung.

In dieser Arbeit tragen wir zur multikriteriellen Entscheidungsunterstützung für die Planung multimodaler Reiserouten bei gleichzeitiger Berücksichtigung komplexer Anforderungen der Reisenden bei. Zunächst führen wir eine systematische Literaturrecherche durch mit dem Ziel, umfassende Einblicke in die komplexen Anforderungen der Reisenden an Mobilitätsplattformen zu gewinnen. Anschließend führen wir multikriterielle Frameworks zur Entscheidungsunterstützung ein, um spezifische Anforderungen der Reisenden bei der Planung multimodaler Reiserouten zu berücksichtigen. Ein Mehrwert für die Reisenden ergibt sich, indem wir eine Reihe von alternativen Haltestellen in Gehdistanz identifizieren. Hierbei werden sowohl Routen als auch Haltestelleninformationen integriert. Weiterhin entwickeln wir ein effektives skalierbares Framework, das die Integration einer Vielzahl individueller Präferenzen in die Suche ermöglicht und führen zwei Ansätze zur weiteren Verbesserung dieses Frameworks ein. Durch die Integration der vorgeschlagenen Frameworks in aktuelle Mobilitätsplattformen wird eine höhere Ausrichtung auf die Bedürfnisse der Reisenden erreicht.

Contents

List of Figures	11
List of Tables	15
1 Introduction	17
1.1 Traveler requirements	19
1.2 Challenges for mobility platform providers	20
1.3 Contribution	22
2 Mobility-as-a-Service-Plattformen: Berücksichtigung von komplexen Reisendenanforderungen mittels nutzerorientierter Algorithmen	27
2.1 Einleitung	29
2.2 Reisendenanforderungen an MaaS-Plattformen	31
2.3 Berücksichtigung von Reisendenanforderungen durch multimodale Routingalgorithmen	43
2.4 Fallstudie zur Berücksichtigung von Reisendenanforderungen	48
2.5 Forschungsbedarf zur Bereitstellung von MaaS-Plattformen	51
3 Identifying alternative stops for first and last mile urban travel planning	55
3.1 Introduction	57
3.2 Related work	58
3.3 Framework for identifying relevant first and last mile stops	63
3.4 Experimental results	67
3.5 Discussion	75
3.6 Conclusion	77
Appendix	78

4	Traveler-oriented multi-criteria decision support for multimodal itineraries	79
4.1	Introduction	81
4.2	Related literature	83
4.3	Solution space sampling	88
4.4	Computational design	105
4.5	Results	110
4.6	Conclusion	122
	Appendices	125
5	Predicting Pareto fronts to guide smart sampling of multimodal itineraries	133
5.1	Introduction	135
5.2	Related literature	137
5.3	Smart sampling framework	140
5.4	Computational design	151
5.5	Computational results	157
5.6	Conclusion	170
	Appendices	172
6	Dynamic learning-based search for multi-criteria itinerary planning	175
6.1	Introduction	177
6.2	Related literature	179
6.3	Gaussian process regression sampling framework	182
6.4	Computational design	191
6.5	Computational results	195
6.6	Conclusion	203
	Appendix	205
7	Conclusion and outlook	207
	Global references	211

List of Figures

1.1	Exemplary Pareto-optimal itineraries	20
1.2	Overview of the structure of this thesis	23
2.1	Anzahl an relevanten Beiträgen	32
2.2	Jelbi-Servicekombinationen	49
3.1	Districts for experiments	68
3.2	Example identification of nearby stops for a traveler	69
3.3	Additional time for stops in the Pareto set	71
3.4	Relationship between preferences	73
3.5	Percent change by district	74
3.6	Parallel coordinates plot of different Pareto front parameters	75
3.7	Percentual change by district	78
4.1	Representation of ϵ -constraint method	87
4.2	Solution space sampling framework	91
4.3	Identification of min-max-intervals for two preferences	92
4.4	Example for two-dimensional systematic sampling	94
4.5	Exemplary multimodal network model	97
4.6	Exemplary finite automaton	99
4.7	Iteration skipping in two-dimensional systematic sampling	103
4.8	Performance metrics	109
4.9	Impact of preferences by mode choice variant and sampling density	113
4.10	Number of Pareto-optimal solutions by mode choice variant and sampling density	118
4.11	Exemplary relationships between preferences	119
4.12	Exemplary aggregated relationships for different ODs	120

4.13	Parallel coordinates for the request from Berlin to Hamburg	121
4.14	Parallel coordinates for the request from Berlin to Hamburg	122
4.15	Finite Automata: All	129
4.16	Finite Automata: Public transportation only	129
4.17	Finite Automata: No flights	129
4.18	Relationship for different traveler preferences	132
5.1	Systematic sampling framework	142
5.2	Smart sampling for each two-dimensional set	143
5.3	Wasserstein distances	145
5.4	Prediction approach: Greedy nearest neighbor	146
5.5	Prediction approach: Clustered attributes	146
5.6	Prediction approach: Decision tree	147
5.7	Prediction approach: Random forest	148
5.8	Transformation techniques	149
5.9	Identification of a reasonable number of clusters	157
5.10	Predicted Pareto front for the request from Essen to Osnabrueck	158
5.11	Parallel coordinates for the request from Essen to Osnabrueck	158
5.12	Density plot for prediction approaches	161
5.13	Pareto-optimal and improved solutions for two-dimensional sampling	166
5.14	Pareto-optimal and improved solutions for three-dimensional sampling	168
5.15	Wasserstein distance and its impact on improved solutions	170
5.16	Exemplary finite automata	172
5.17	Finite automata: All	172
5.18	Finite automata: Public transportation only	172
5.19	Finite automata: No flights	172
5.20	Wasserstein distance and its impact on improved solutions	173
6.1	Exemplary Pareto front	177
6.2	Individual traveler request to an integrated mobility application	183
6.3	Main steps in sampling framework	184
6.4	Gaussian Process Regression (GPR) sampling for set $S_{travelTime,price}$	189
6.5	Systematic sampling for set $S_{price,numberOfTransfers}$	190
6.6	Improved solutions in comparison to the baseline	198

6.7	Accumulated average iteration to retrieve Pareto-optimal solutions . . .	201
6.8	GPR for set $S_{travelTime,price}$	202
6.9	Radar plot for the request from Stuttgart to Erfurt	203
6.10	Accumulated average iteration to retrieve Pareto-optimal solutions . . .	205

List of Tables

2.1	Berücksichtigung von Reisendenanforderungen durch Routingalgorithmen	45
3.1	Traveler preferences	61
3.2	Savings potential with respect to different parameters	70
3.3	Advantages for accessible stops when implementing new accessibility parameter	72
3.4	Average differences between scooter and non-scooter experiments . .	73
4.1	Comparison of maximum runs for different sampling densities	102
4.2	Summary of results	111
4.3	Comparison of two and three-dimensional experiments	111
4.4	Performance metrics by preference sets	116
4.5	Performance metrics by mode choice variant and sampling density for $P := \{tt, pr, nt, wd, wt\}$	117
4.6	Performance metrics by fa and k for $P := \{tt, pr\}$	130
4.7	Performance metrics by fa and k for $P := \{tt, pr, nt\}$	130
4.8	Performance metrics by fa and k for $P := \{tt, pr, nt, wd\}$	131
5.1	Exemplary transformation for Euclidean and Product distance	149
5.2	Summary of results: Prediction of Pareto front structures	160
5.3	Summary of results: Transformation techniques	162
5.4	Summary results for random forest prediction with Euclidean distance transformation	163
6.1	Examined combinations of traveler preferences	193
6.2	Summary results for GPR sampling	196

List of Tables

6.3	Baseline solutions enhanced by improved solutions differentiated by preference	199
6.4	Avg. iteration to retrieve Pareto-optimal solutions: Two-dimensional setting	200

Chapter 1

Introduction

The importance of multimodal mobility planning has increased in recent years. According to the EU-27 transport report by the European Commission, passenger transport in the 27 member states of the European Union increased significantly by approximately 34% between 1995 and 2019. A similar trend can also be observed in the United States, with the number of passenger kilometers rising from 6626.5 billion in 1995 to 9804.2 billion in 2019, representing an increase of about 48% (European Commission & Directorate-General for Mobility and Transport, 2022). In addition, a tendency to combine multiple mobility services such as public transport, biking and sharing services according to individual needs can be observed. These combinations of multiple mobility services are referred to as multimodal mobility (Willing et al., 2017). The study “Millennials and Mobility” by the American Public Transportation Association highlights that close to 70% of the millennial generation utilize various forms of mobility service several times or more per week, further emphasizing the growing acceptance of multimodal mobility (American Public Transportation Association, 2013). Promoting the attractiveness of multimodal mobility supports the shift away from motorized individual transportation toward public transport. This shift not only helps alleviate traffic congestion, pollution, and noise, particularly in urban areas, but also contributes to more sustainable transportation systems (Grotenhuis et al., 2007; Jittrapirom et al., 2018).

The surge in travel and the growing willingness to use multimodal mobility underline the high relevance of Mobility as a Service (MaaS). MaaS entails decision support for the traveler through digital mobility platforms. These MaaS platforms integrate a wide range of available mobility services and offer assistance to the traveler throughout the

entire planning process, from searching for mobility options to booking and payment (Alyavina et al., 2022; Esztergár-Kiss et al., 2020).

An essential key toward seamless MaaS and hence fostering multimodal travel is to fulfill the high expectations of travelers toward integrated mobility platforms. While searching and planning individual itineraries, travelers provide basic information such as the individual origin and the destination as well as the earliest departure time. Given this information, travelers expect swift provision of a diverse set of relevant door-to-door options to choose from, enriched with additional information about the available options. Furthermore, they expect that multiple traveler preferences, among others travel time, price, and number of transfers, are integrated into the search (Lyons et al., 2020; Spickermann et al., 2014).

The need for integrated multimodal itinerary planning and the high expectations of travelers pose new challenges for mobility platform providers such as *GoogleMaps* and *Rome2Rio*. Mobility platform providers integrate multiple available mobility services into one platform aiming to offer a seamless multimodal door-to-door mobility experience. Alongside challenges associated with establishing business and cooperation models between integrated mobility service providers and integrating data from different mobility services, additional challenges arise from a route planning algorithm perspective. To address these challenges, mobility platform providers require a multi-criteria decision support framework capable of integrating multiple traveler preferences within a multimodal context and that retrieves relevant door-to-door options in efficient run time (Bast et al., 2015).

In this thesis, we analyze traveler requirements for MaaS platforms in detail by conducting a literature review and propose two novel multi-criteria decision support frameworks to enhance the level of traveler orientation when planning multimodal itineraries. Our first framework focuses on identifying alternative stops within walking distance, enabling travelers to explore additional options for their itineraries. The second framework addresses the challenge of scalability. We propose an effective scalable framework to enable the incorporation of a multitude of traveler preferences into the search without significant impact on the framework’s run time. The developed frameworks can be integrated into state-of-the-art mobility platforms to foster MaaS.

In Section 1.1, we discuss traveler requirements toward integrated multimodal mobility platforms in detail. Then, in Section 1.2, we present resulting challenges

for platform providers and discuss examples of state-of-the-art mobility platforms. Finally, in Section 1.3, we introduce the contribution of this thesis toward traveler orientation in multi-criteria decision support for multimodal traveler planning and thus seamless MaaS.

1.1 Traveler requirements

Traveler orientation is an essential part of state-of-the-art itinerary planning. First of all, travelers face significant cognitive effort and time expenditure when they have to manually combine various routing platforms to access different mobility services. For instance, when a traveler intends to travel to the United States and has several relevant airports within an acceptable distance, it becomes highly demanding and time-consuming to separately research and compare the options to each airport, along with the respective available flights from each airport. Therefore, the traveler requires that all available mobility services applicable to the individual request are integrated in one multimodal mobility platform. This integration allows travelers to access comprehensive options, reducing the burden of searching across multiple platforms and facilitating a more seamless itinerary planning experience (Grotenhuis et al., 2007; Willing et al., 2017).

Furthermore, multiple individual preferences have to be integrated into the platform. While most state-of-the-art mobility platforms provide only options considering travel time, price, and number of transfers as relevant traveler preferences, further preferences such as walking distance, waiting time, and reliability can be of individual importance for the traveler as well. These preferences often compete against each other, resulting in trade-offs in the set of available itineraries. For instance, an itinerary could be either very fast but expensive, while another itinerary may be slower but more affordable. As common for such a posteriori planning problem settings like multimodal itinerary planning, it is challenging for the traveler to assess the relevance of individual preferences on the choice set. Hence, the traveler expects a reasonable-sized set of diverse Pareto-optimal options to choose from (Lyons et al., 2020; Spickermann et al., 2014). An illustration of Pareto-optimal itineraries is shown in Figure 1.1 with travel time and price as exemplary preferences. The cheapest but slowest choice is represented by itinerary A, whereas the fastest but most expensive option is depicted

by itinerary D. Itineraries B and C are Pareto-optimal trade-offs between travel time and price.

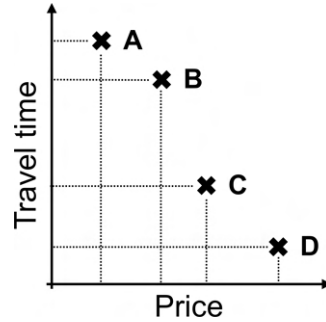


Figure 1.1: Exemplary Pareto-optimal itineraries

Given the set of multimodal options, travelers often seek assistance in selecting one of these (Alt et al., 2019; Grotenhuis et al., 2007; Meske et al., 2020). For instance, many integrated mobility platforms offer travelers a filtering feature to refine the selection of relevant service combinations. However, travelers require additional information about the complex multimodal solution space to effectively use these filtering features. For instance, restricting the possible options based on criteria such as travel time and price may result in an empty set. Offering adequate support reduces the cognitive and time burden associated with service selection, thus making the utilization of MaaS platforms more traveler-oriented. Presenting a set of alternative nearby stops with their respective characteristics to the traveler can further empower the traveler to take informed decisions (Bucher et al., 2017; Nasibov et al., 2016).

1.2 Challenges for mobility platform providers

Mobility service providers should meet the needs of the traveler to enable seamless MaaS and consequently promote multimodal travel. Advances in digitization and greater computing power have led to significant advances in route planning algorithms when considering a single mobility service at a time. Modern algorithms, combined with preprocessing techniques, can calculate itineraries on large unimodal networks in negligible run time (Bast et al., 2015; Pajor, 2009). However, the integration of all available mobility services into an integrated mobility platform increases the complexity of planning multimodal itineraries. This complexity arises from the

combination of both scheduled-based services and unrestricted services. In particular, scheduled-based services make the planning problem more complex due to their inherent characteristic of time dependency (Bast et al., 2015).

Incorporating multiple traveler preferences and hence dealing with a multi-criteria decision making problem increases the complexity even further. In recent years, several approaches for multi-criteria decision support in a multimodal setting have been proposed (Delling et al., 2013a; Dib et al., 2017; Potthoff & Sauer, 2022). While these approaches are capable of identifying the full Pareto-optimal set of itineraries in efficient run time if two or three preferences are integrated into the search, they perform significantly worse when considering more than three traveler preferences simultaneously.

Recent state-of-the-art mobility platforms still have weaknesses in meeting these challenges despite progress in functionality and traveler orientation. Let us consider the following examples:

- *Google Maps* is one of the most well-known mobility platforms globally. It enables travelers to choose between driving, using public transport, walking, cycling, and flying. According to Zipper (2017), Google has established a partnership with over 800 public transportation providers to make their schedules available by integrating their General Transit Feed Specification (GTFS) data. However, Google Maps does not fully combine the wide range of integrated mobility services to provide a seamless multimodal experience for the traveler. For example, searching for available itineraries from Magdeburg, Germany, to Chicago, United States, results in an empty solution set, as there is no combination of itineraries to the airport with the respective flights integrated. Regarding traveler preferences, Google Maps enables the traveler to specify their mode choice, minimize transfers, prioritize short walking distances, and opt for accessible itineraries suitable for travelers with disabilities. However, price information is partly missing, leaving travelers with an incomplete comprehension of the prices associated with the available itineraries. Another limitation of Google Maps is the lack of explanation provided to travelers regarding the resulting choice set. For instance, when the option “Best Route” is chosen as the preferred itineraries, there is no explanation given to the traveler why the displayed options are considered as the best choices.

- *Rome2Rio* is a multimodal mobility platform providing mobility services globally. It is owned by the German company *Omio*. Rome2Rio has a vast majority of available mobility services integrated including recent car- and bike-sharing services. As a result, the traveler is provided with a complete multimodal travel experience. For example, unlike Google Maps, it enables the traveler to plan an itinerary from Magdeburg to Chicago. A significant drawback of Rome2Rio is that only travel time, price, and the number of transfers are considered as relevant traveler preferences. Further individual relevant preferences such as total walking distance and waiting time are missing. The importance of these preferences has been highlighted and emphasized in recent studies (Grotenhuis et al., 2007; He & Csiszár, 2020). The traveler is presented with a set of possible options. In addition to the fastest and cheapest options, the best option is also highlighted. However, there is no detailed explanation why this itinerary is labeled as the best option. Moreover, providing a comprehensive overview of the characteristics of all available options would further empower the traveler and lead to a seamless MaaS experience (Alt et al., 2019; Meske et al., 2020).

These limitations in current state-of-the-art mobility platforms to find a reasonable-sized set of Pareto-optimal options in a large multimodal network motivated us to develop multi-criteria decision support frameworks for complex multimodal travel planning in this thesis and contribute to the development of seamless MaaS.

1.3 Contribution

The goal of this thesis is to contribute to traveler orientation in multi-criteria decision support for multimodal traveler planning. This research is divided into two main parts that complement one another. Figure 1.2 presents an outline of this thesis.

In the first part, the key focus is on understanding traveler requirements for MaaS platforms and identifying resulting research gaps for traveler-oriented multi-criteria itinerary planning in a multimodal context. Based on a systematic literature review, we aim at understanding individually prevalent traveler preferences while planning multimodal travel itineraries. These preferences can be situational (e.g., minimizing transfers when traveling with small children) as well as habitual (e.g., preferring using a sharing service over taking a bus in an urban area). Furthermore, we identify

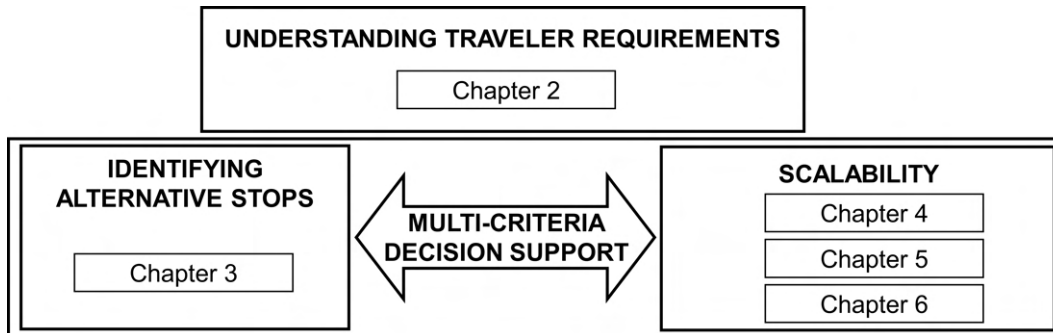


Figure 1.2: Overview of the structure of this thesis

functional requirements and requirements for supporting the decision-making process by the traveler to integrated mobility platforms. Next, we discuss to what extent the identified traveler requirements are covered by state-of-the-art multimodal routing algorithms. Finally, we present a case study based on the mobility platform *Jelbi* (www.jelbi.de) and analyze to what extent it fulfills the identified traveler requirements. Jelbi has been deployed in the Berlin metropolitan area since 2019 and integrates multiple public transport services as well as sharing services. Details of understanding traveler requirements are comprised in the following publication, which was created in single authorship.

Chapter 2

Horstmannshoff, T. (2022). **Mobility-as-a-service-plattformen – Berücksichtigung von komplexen Reisendenanforderungen mittels nutzerorientierter Algorithmen.** In M. Bruhn & K. Hadwich (Eds.), **Smart Services** (pp. 523–546). [S.l.]: GABLER. doi: 10.1007/978-3-658-37346-7 19

In the second part of this thesis, we introduce multiple multi-criteria decision support frameworks for planning traveler-oriented multimodal itineraries. All publications assigned to the second part of this thesis have been created in co-authorship.

First, we aim at providing the traveler with a set of alternative stops within walking distance combining route and stop-based information. Besides common route-based preferences such as travel time and price, we integrate additional relevant stop-based parameters such as frequency and number of lines into the search. This provides the traveler with additional value in their decision-making process to decide between alternative stops for the first and last mile in their itinerary planning. This framework

is evaluated on the public urban bus network of Göttingen, Germany. Furthermore, we extend this framework to a multimodal setting by incorporating free-floating scooters and analyze changes in itinerary and stop characteristics. Chapter 3 introduces the framework for identifying alternative stops in detail and consists of the following paper.

Chapter 3

Horstmannshoff, T. & Redmond, M. F. (2023). **Identifying alternative stops for first and last mile urban travel planning** *Submitted to Public Transport. 3rd round (major revision)*

Next, we focus on effective scalable multi-criteria decision support for the planning of multimodal itineraries. In Chapter 4, we propose a sampling framework to approximate the Pareto-optimal set of itineraries, which scales well in terms of the number of considered traveler preferences. The core idea is to break down a high-dimensional multi-criteria problem setting into multiple problems of smaller dimensions, which are much easier to compute. Then, we sample the respective low-dimensional problem settings simultaneously by sampling uniformly distributed across the complex multimodal solution space and present the traveler the merged set removed from dominated itineraries. This effective scaling framework is evaluated by analyzing itineraries between major cities in Germany integrating a large amount of real-world data on mobility services.

In addition, to improve the quality of the retrieved Pareto-optimal set further, we introduce two approaches to guide the sampling process during the search. On the one hand, we learn and predict the Pareto-front structure of the multimodal solution space before starting the sampling process with predictive modeling from historical search data. The predicted structure will then be used to guide the sampling framework to more promising areas of the multimodal solution space. Chapter 5 presents details on predicting Pareto fronts to guide smart sampling of multimodal itineraries. On the other hand, we dynamically guide the sampling framework during the search to relevant areas of the solution space with high uncertainty by applying Gaussian process regression. Hereby, it is dynamically decided which area of the complex multimodal solution space is to be sampled next. More information about multi-criteria itinerary Planning with Gaussian process regression is provided in

Chapter 6. The following publications are dealing with effective scalable multi-criteria decision support.

Chapter 4

Horstmannshoff, T. & Ehmke, J. F. (2022). **Traveler-oriented multi-criteria decision support for multimodal itineraries**. *Transportation Research Part C: Emerging Technologies*, 141, 103741. doi: 10.1016/j.trc.2022.103741

Chapter 5

Horstmannshoff, T. & Ehmke, J. F. (2023). **Predicting Pareto fronts to guide smart sampling of multimodal itineraries** *Submitted to OR Spectrum. 2nd round (major revision)*

Chapter 6

Horstmannshoff, T., Ehmke, J. F. & Ulmer, M. W. (2023). **Dynamic learning-based search for multi-criteria itinerary planning** *Submitted to Omega. 1st round*

All papers have been reformatted with font sizes and styles aligned to maintain uniformity in presentation. Figures and tables may be moved within the text to eliminate needless page breaks and are continuously numbered across all publications. At the beginning of this thesis, a list of all figures and tables is provided. Furthermore, a coherent citation style is applied and global references are stated at the end of this thesis.

Chapter 2

Mobility-as-a-Service-Plattformen: Berücksichtigung von komplexen Reisendenanforderungen mittels nutzerorientierter Algorithmen

Abstract

Mobility-as-a-Service-Plattformen (MaaS-Plattformen) ermöglichen Reisenden eine anbieter- und verkehrsmittelübergreifende Mobilität von Tür-zu-Tür. In diesem Beitrag werden mittels einer systematischen Literaturrecherche komplexe Reisendenanforderungen an MaaS-Plattformen identifiziert. Anschließend wird die Berücksichtigung dieser Reisendenanforderungen durch aktuelle multimodalen Routingalgorithmen untersucht. Schließlich erfolgt eine Diskussion des MaaS-Konzeptes im Kontext der multimodalen Mobilität anhand der MaaS-Plattform Jelbi. Außerdem wird zukünftiger Forschungsbedarf an der Schnittstelle von Plattform- und Algorithmen-Design diskutiert.

Keywords Mobility-as-a-Service, Multimodale Mobilität, Reisendenanforderungen

Contents

2.1	Einleitung	29
2.2	Reisendenanforderungen an MaaS-Plattformen	31
2.2.1	Gewünschte Informationen für MaaS	33
2.2.2	Reisendenanforderungen an plattformbasierte Mobilitätsdienstleistungen	37
2.2.3	Forschungsbedarf	43
2.3	Berücksichtigung von Reisendenanforderungen durch multimodale Routingalgorithmen	43
2.4	Fallstudie zur Berücksichtigung von Reisendenanforderungen	48
2.4.1	MaaS-Plattform Jelbi	48
2.4.2	Evaluation der MaaS-Plattform Jelbi	50
2.5	Forschungsbedarf zur Bereitstellung von MaaS-Plattformen	51

2.1 Einleitung

Reisenden steht heutzutage eine Vielzahl heterogener Mobilitätsservices zur Verfügung, um ihren Reisewunsch individuell zu gestalten. Neben traditionellen Mobilitätsservices wie öffentliche Nah- und Fernverkehre, zum Beispiel der Deutschen Bahn, umfasst dies immer mehr auch neu aufkommende Sharing-Services wie bspw. Ridesharing (Esztergár-Kiss & Kerényi, 2020). Bedingt durch die Digitalisierung ist der Zugriff auf diese und die Kombination dieser Mobilitätsservices inzwischen deutlich einfacher geworden. Mobility-as-a-Service-Plattformen (MaaS-Plattformen) wie Google Maps (www.google.com/maps) und Rome2Rio (www.rome2rio.com) integrieren eine Vielzahl unterschiedlicher Mobilitätsservices und ermöglichen Reisenden die Auswahl und Kombination dieser Services in einer integrierten Plattform Lyons et al. (2020).

Während den Reisenden eine Vielzahl unterschiedlicher Mobilitätsservices zur Auswahl steht, besteht eine wesentliche Herausforderung in der Berücksichtigung individueller, komplexer Anforderungen der Reisenden in allen Phasen des Mobilitätsserviceprozesses. Dieser Prozess umfasst die Suche und die Bereitstellung der Information zu multimodalen Mobilitätsangeboten (im Folgenden als Servicekombination bezeichnet) bis zur Buchung und Bezahlung dieser (Albrecht & Ehmke, 2016). Diese Serviceanforderungen sollten integrierte MaaS-Plattformen bei der Planung der individuellen Reise berücksichtigen. Dies umfasst neben klassischen Präferenzen wie Fahrzeit, Preis und Anzahl der Umstiege auch komplexere Präferenzen wie eine begrenzte Wartezeit innerhalb der Reisketten (Habermann et al., 2016).

MaaS-Plattformen erleichtern Reisenden den Zugang zu den Mobilitätsservices, indem sie Reisende bei der Suche, Auswahl und Buchung von Mobilitätsservices unterstützen (Alt et al., 2019; Stopka et al., 2018). Ein Beispiel hierfür ist die Bereitstellung einer Auswahl an alternativen möglichen Servicekombinationen auf Basis einer individuellen Anfrage, aus welchen Reisende anschließend selbstständig auswählen können (Alt et al., 2019; Stopka, 2014). Hierbei ist zu beachten, dass die Auswahl von Servicekombinationen geeignet ist, d.h. dass Reisenden einerseits keine Auswahl angeboten werden, z. B. aufgrund sich widersprechender Anforderungen, andererseits die Menge an Servicekombinationen auch nicht zu groß ist, was wiederum zu einer Überforderung bei der Auswahl führen könnte (Spickermann et al., 2014).

Der Fokus auf die Integration von komplexen Reisenanforderungen kann Anreize zur Nutzung von MaaS-Plattformen setzen und damit auch eine Veränderung des

Mobilitätsverhaltens induzieren. Ein geändertes Mobilitätsverhalten ist ein Baustein, um anstehende Herausforderungen hinsichtlich der zunehmenden Urbanisierung zu mildern. Dies umfasst unter anderem zunehmender Stau, weniger Parkraum sowie die Luft- und Lärmverschmutzung in den Städten bedingt durch den hohen Anteil an privaten Autos von ca. 75 %. Zusätzlich unterstützt ein zunehmendes multimodales Mobilitätsverhalten die Nachhaltigkeitsziele (Willing et al., 2017).

Um komplexe Serviceanforderungen reisendenorientiert berücksichtigen und damit eine Auswahl an geeigneten Servicekombinationen bereitstellen zu können, sind MaaS-Plattformen benutzerfreundlich zu gestalten (z. B. durch eine nutzerfreundliche Präsentation der Auswahl) (Meske et al., 2020), und die Anforderungen sind durch einen zugrundeliegenden Routingalgorithmus zu berücksichtigen. Während aus algorithmischer Perspektive in den letzten Jahren erhebliche Fortschritte hinsichtlich eines effizienten multimodalen Routings erzielt worden sind (Bast et al., 2015), besteht weiterhin die Herausforderung einige komplexe Serviceanforderungen in diese Algorithmen einzubeziehen und damit die Auswahl an Servicekombinationen an den Bedürfnissen von Reisenden auszurichten. Beispielsweise kann die vollständige Menge der Pareto-optimalen Servicekombinationen bisher nicht effizient bestimmt werden, sofern eine Vielzahl individueller Servicewünsche des Reisenden zu berücksichtigen sind (Delling et al., 2013a; Dib et al., 2017). Damit wird gegebenenfalls eine nur unzureichende Auswahl von Servicekombinationen bereitgestellt.

In diesem Beitrag werden daher zunächst in Kapitel 2.2 die Anforderungen an integrierte MaaS-Plattformen systematisch erfasst. Die Analyse erfolgt dabei vorwiegend aus Reisendenperspektive; weitergehende Aspekte wie bspw. Geschäfts- und Kooperationsmodelle zwischen den Mobilitätsserviceanbietern sowie Datenintegration der unterschiedlichen Mobilitätsservices werden im Rahmen dieses Beitrages nicht betrachtet. Stattdessen werden bestehende quantitative und qualitative Forschungsarbeiten zu Reisendenanforderungen an MaaS-Plattformen strukturiert zusammengeführt. Die resultierende Forschungsfrage F_1 lautet:

F_1 : Welche Anforderungen weisen Reisende an MaaS-Plattformen auf?

Im nächsten Schritt wird der Fokus auf die algorithmische Perspektive gelegt. Dazu wird in Kapitel 2.3 ein Überblick über aktuelle, multimodale Routingalgorithmen gegeben. Hierbei wird insbesondere dargestellt, inwieweit diese die identifizierten Reisendenanforderungen an integrierte MaaS-Plattform (F_1) aus algorithmischer

Perspektive berücksichtigen. Die Auswahl der dargestellten Routingalgorithmen wird auf solche begrenzt, welche Teilanforderungen an MaaS-Plattformen bereits konzeptionell berücksichtigen. Zusammengefasst wird dies in Forschungsfrage F_2 :

F_2 : Wie weit werden identifizierte Reisendenanforderungen aus F_1 durch aktuelle Algorithmen des multimodalen Routings berücksichtigt?

Sodann erfolgt in Kapitel 2.4 eine Präsentation einer Fallstudie zur Berücksichtigung des MaaS-Konzeptes im Kontext der multimodalen Mobilität anhand der MaaS-Plattform Jelbi. Abschließend werden in Kapitel 2.5 Forschungs- und Optimierungsbedarf zur besseren Berücksichtigung von Reisendenanforderungen bei der Bereitstellung von MaaS-Plattformen identifiziert:

F_3 : Identifikation von Forschungsbedarf zur Bereitstellung von MaaS-Plattformen.

2.2 Reisendenanforderungen an MaaS-Plattformen

Im Folgenden werden die Reisendenanforderungen an eine MaaS-Plattform auf Basis einer Literatursuche nach vom Brocke et al. (2009) systematisch analysiert (F_1). Um eine möglichst umfassende Suche zu gewährleisten, verwenden wir die Literaturdatenbank Scopus als größte Abstrakt- und Zitationsdatenbank für wissenschaftlich einschlägig begutachtete Literatur. Zur Eingrenzung der Suche auf Reisendenanforderungen an MaaS-Plattformen verwenden wir die folgenden Suchbegriffe in den jeweiligen Titeln, Abstracts und Stichwörtern der Beiträge:

(TITLE-ABS-KEY ("customer") OR TITLE-ABS-KEY ("User") OR TITLE-ABS-KEY ("Traveler") OR TITLE-ABS-KEY ("Traveller")) AND (TITLE-ABS-KEY ("Mobility as a Service") OR TITLE-ABS-KEY ("Mobility-as-a-Service") OR TITLE-ABS-KEY ("Mobility Platform") OR TITLE-ABS-KEY ("MaaS") OR TITLE-ABS-KEY ("Multimodal Mobility") OR TITLE-ABS-KEY ("Intermodal Mobility") OR TITLE-ABS-KEY ("Mobility Service"))

Weiterhin grenzen wir die Ergebnisse auf final veröffentlichte, deutsch- und englischsprachige Beiträge ab dem Jahr 2007 ein. Die Einführung des Smartphones in diesem Jahr hat wesentlich zur Entwicklung von integrierten MaaS-Plattformen beigetragen, sodass Reisende von überall einen einfachen Zugang zu diesen haben.

Insgesamt produziert diese Suche 860 Ergebnisse (Stand 24.07.2021). Die Analyse wird nach Vorwärts- und Rückwärtssuche auf Beiträge eingegrenzt, welche sich explizit auf Reisendenanforderungen an integrierte MaaS-Plattformen fokussieren,

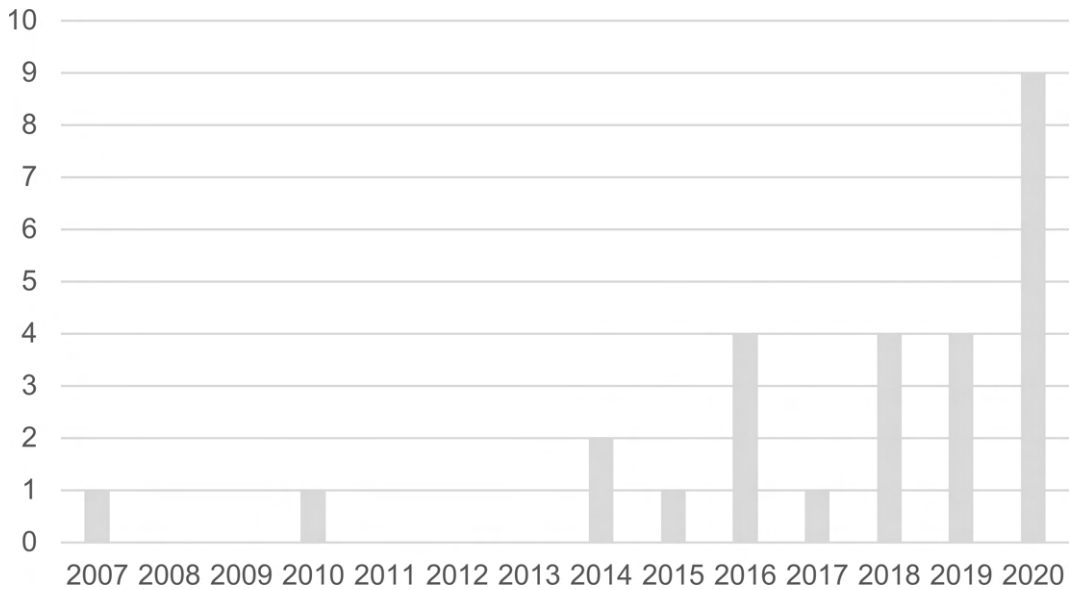


Figure 2.1: Anzahl an relevanten Beiträgen

sodass auf Basis der jeweiligen Abstracts 27 Beiträge zur detaillierten Sichtung verbleiben. Abbildung 2.1 zeigt die Anzahl der veröffentlichten Beiträge pro Jahr. Es ist ersichtlich, dass ab dem Jahr 2014 die Forschung zur reisendenorientierten Gestaltung von MaaS an Relevanz gewonnen hat. Lyons et al. (2020) betonen, dass ungefähr jede zweite vorhandene MaaS-Plattform ab 2014 entstanden ist. Weiterhin haben sich seitdem einige neue Mobilitätsservices am Markt etabliert sowie sind erhebliche Fortschritte in der Digitalisierung erzielt worden, sodass die Bedeutung von integrierten MaaS-Plattformen zugenommen hat.

Im Folgenden werden die identifizierten Beiträge zunächst hinsichtlich möglicher Reisendenanforderungen an eine MaaS-Plattform analysiert. Dies beinhaltet auch komplexe Präferenzen, welche bisher nicht in MaaS-Plattformen berücksichtigt werden. Ein ausführlicher Überblick über die entsprechenden Anforderungen bzw. gewünschte Serviceinformationen seitens des Reisenden erfolgt in Kapitel 2.2.1. Anschließend werden in Kapitel 2.2.2 die Anforderungen an MaaS-Plattformen analysiert.

2.2.1 Gewünschte Informationen für MaaS

Reisende haben individuelle Servicewünsche, welche sowohl situativ (z. B. wird bei einem anstehenden Geschäftstermin eine zuverlässige Ankunftszeit priorisiert), sowie gewohnheitsmäßig (z. B. Gewohnheit, dass das private Auto als komfortabler angesehen wird) bedingt sein können (Lyons et al., 2020; Schwinger & Krempels, 2019). Eine Berücksichtigung dieser Servicewünsche bei der Recherche nach relevanten Servicekombinationen in einer MaaS-Plattform ist ein wesentlicher Beitrag zur reisendenorientierten Planung.

Reisende benötigen eine *multikriterielle* Entscheidungsunterstützung bei der Suche und Auswahl nach möglichen Servicekombinationen. Überwiegend ist nicht nur eine einzige Serviceinformation allein von Relevanz für den Reisenden (Lyons et al., 2020; Spickermann et al., 2014), sondern die gleichzeitige Berücksichtigung einer Vielzahl von Servicewünschen ist erforderlich. Beispielsweise können Reisende gewillt sein etwas mehr Fahrzeit zu investieren, um einen deutlich günstigeren Preis zu erhalten. Hierdurch ergibt sich eine Menge an relevanten Pareto-optimalen Servicekombinationen zur Auswahl für den Reisenden. Servicewünsche können entweder als Bestandteil einer multikriteriellen Zielfunktion (z. B. Trade-off zwischen Fahrzeit und Preis) oder als Nebenbedingung (z. B. maximale Fahrzeit) im Routingalgorithmus berücksichtigt werden (Bast et al., 2015). Im Folgenden wird ein Überblick über relevante Serviceinformationen seitens des Reisenden gegeben, welche im Zuge der multikriteriellen Entscheidungsunterstützung integriert werden können:

Fahrzeit, Preis sowie Anzahl der Umstiege: Sehr häufig werden in aktuellen MaaS-Plattformen Servicewünsche wie maximale Fahrzeit, Preis sowie Anzahl der Umstiege berücksichtigt. Die hohe Relevanz dieser Serviceinformationen wird ebenso durch die identifizierte Literatur zu Reisendenanforderungen an MaaS-Plattformen bestätigt (Arias-Molinares & García-Palomares, 2020; Casady, 2020; He & Csiszár, 2020; Lyons et al., 2020). Arias-Molinares & García-Palomares (2020) heben auf Basis eines ausführlichen Literaturüberblicks zu diversen MaaS-Aspekten hervor, dass insbesondere junge Reisende ein begrenztes Budget haben. Folglich ist der Preis für diese Benutzergruppe vermehrt von wesentlicher Bedeutung (Arias-Molinares & García-Palomares, 2020; Casady, 2020). He & Csiszár (2020) untersuchen die wahrgenommene Servicequalität von MaaS-Plattformen. Hinsichtlich der Serviceinformation Preis betonen die Autoren, dass vorhandene Rabattkarten (bspw. Bahncard)

sowie Wochen-, Monats- oder Jahreskarten von einzelnen Mobilitätsservices einmalig in den integrierten MaaS-Plattformen hinterlegt und anschließend automatisiert bei der Preiskalkulation mitberücksichtigt werden sollten. Dadurch wird der Suchaufwand für den Reisenden erheblich reduziert.

Zusätzlich lassen sich viele weitere Servicewünsche seitens des Reisenden identifizieren, welche in aktuellen MaaS-Plattformen überwiegend nicht berücksichtigt werden. Dazu gehören die Sicherheit während der Reise, eine hohe Zuverlässigkeit hinsichtlich der Fahrzeit, die aggregierte Gehdistanz, eine begrenzte Wartezeit, die Verfügbarkeit, Sauberkeit und der Komfort der jeweiligen Mobilitätsservices, die Umweltverträglichkeit der Servicekombination sowie eine einfache Eingabe der Servicewünsche.

Sicherheit: Casadó et al. (2020) untersuchen die Bedürfnisse und die Wahrnehmung von integrierten MaaS-Plattformen von Kindern und Jugendlichen in Newcastle-upon-Tyne. Mitglieder dieser Altersgruppe weisen besondere Anforderungen an das Sicherheitsgefühl während einer Reise auf. Beispielsweise äußert ein Mitglied aus der Befragungsgruppe der 16- bis 18-Jährigen, dass es sich in einem Bus sicherer fühlt als in der Metro, da im Bus der Fahrer unmittelbar bei Gefahrensituationen eingreifen kann. Hinsichtlich der Verwendung von Ridesharing- und Taxis als Mobilitätsservices wird ein erhöhtes Sicherheitsgefühl erzeugt, wenn einer dritten Person (bspw. den Erziehungsberechtigten) die Kontaktdaten des Fahrers leicht zugänglich gemacht werden können. Ebenso kann die Integration eines Bewertungssystems für Fahrer beim Ridesharing das Sicherheitsgefühl erhöhen. Gilibert & Ribas (2019) fokussieren sich in ihrem Beitrag auf Reisendenanforderungen an geteilte Ridesharing-Dienste. Auf Basis einer Pilotstudie in Barcelona unterstützen sie die Aussage, dass ein Bewertungssystem einen wesentlichen Beitrag u.a. hinsichtlich der wahrgenommenen Sicherheit leistet. Das Teilen der Fahrerdaten mit Dritten sowie ein integriertes Bewertungssystem können eine zusätzliche Funktion einer integrierten MaaS-Plattform insbesondere für Kinder und Jugendliche darstellen (Casadó et al., 2020; Gilibert & Ribas, 2019; Spickermann et al., 2014). Ebenso kann die Sicherheit hinsichtlich der Ansteckungsgefahr im Zuge von pandemischen Entwicklungen eine relevante Rolle bei der Wahl des Mobilitätsservice sein.

Zuverlässigkeit: Weiterhin ist eine hohe Zuverlässigkeit hinsichtlich der Fahrzeit von hoher Relevanz für Reisende bei der Wahl geeigneter Servicekombinationen. Ein

Beispiel hierfür kann eine Geschäftsreise sein, in welcher der Reisende gewillt ist extra Zeit zu investieren um eine höhere geschätzte Zuverlässigkeit der Reisekette zu erzielen (Bell, 2019; Redmond et al., 2020; Spickermann et al., 2014; Stopka et al., 2015). Dies betrifft sowohl die Verlässlichkeit von Umsteigevorgängen als auch die Ankunftszeit am Zielort. Spickermann et al. (2014) stützen sich auf empirische Erkenntnisse aus drei parallelen Delphi-Studien und mehreren Fokusgruppen-Workshops, um strategische Implikationen für Unternehmen, Behörden und Reisenden aufzuzeigen und zeigen die Relevanz von zuverlässigen Servicekombinationen auf. Stopka et al. (2015) belegen ebenfalls die Relevanz einer hohen Zuverlässigkeit bei der Wahl von Servicekombinationen durch qualitative und quantitative Interviews im Zuge des MaaS-Forschungsprojekts “Dynamic seamless mobility information”.

Gehdistanz: Die aggregierte Gehdistanz für die erste und letzte Meile sowie im Zuge des Transfers zwischen nah beieinanderliegenden Haltestellen kann für einige Reisende von Relevanz sein (Casadó et al., 2020; Gilibert & Ribas, 2019; Grotenhuis et al., 2007; He & Csiszár, 2020; Stopka, 2014). Die aggregierte Gehdistanz ist bspw. von hoher Bedeutung für mobilitätseingeschränkte Reisende oder Reisende mit viel Gepäck. Diese Reisende präferieren oft Servicekombinationen mit geringer aggregierter Gehdistanz.

Wartezeit: In einigen Beiträgen wird eine hohe Relevanz einer begrenzten Wartezeit innerhalb der Reisekette betont (Grotenhuis et al., 2007; Habermann et al., 2016; He & Csiszár, 2020). Habermann et al. (2016) präsentieren eine empirische Studie zur Bewertung von Reisendenanforderungen an MaaS-Plattformen. Diese Studie ist im interdisziplinären Forschungsprojekt “Mobility Broker” eingebettet. Hierbei haben ca. 40 % der befragten Teilnehmer geäußert, dass eine lange Wartezeit innerhalb der Reisekette gegen eine Auswahl dieser spricht. Für die Gruppe der befragten Studierenden stellt eine lange Wartezeit innerhalb der Reisekette der zweitwichtigste Grund gegen die Wahl eines spezifischen Mobilitätsservice dar.

Verfügbarkeit: Insbesondere bei innovativen Mobilitätsservices wie Bike-, Car- und Ridesharing-Services ist die Verfügbarkeit von größerer Bedeutung für die Gruppe der befragten Studierenden (Habermann et al., 2016). Unter Verfügbarkeit wird hierbei verstanden, dass der jeweilige Mobilitätsservice zum Zeitpunkt des Bedarfs für die Benutzung tatsächlich vorhanden ist. Beispielsweise können insbesondere zu Stoßzeiten einzelne Bikesharing-Stationen keine Räder anbieten. Gilibert & Ribas

(2019) und He & Csiszár (2020) unterstützten die hohe Wichtigkeit der Verfügbarkeit bei der Wahl von Ridesharing-Services in ihren jeweiligen Beiträgen. Gilibert & Ribas (2019) betonen im Zuge ihrer Pilotstudie in Barcelona, dass eine hohe Verfügbarkeit von Ridesharing-Services einer der entscheidenden Faktoren ist, dass Reisende ein regelmäßiger Benutzer dieser wird.

Komfort: Ein hoher Reisekomfort ist für eine Vielzahl von Reisenden ebenso eine wichtige Serviceinformation (Casadó et al., 2020; He & Csiszár, 2020; Schwinger & Krempels, 2019; Spickermann et al., 2014; Stopka, 2014). Hierbei umfasst Komfort neben der Ausstattung innerhalb des jeweiligen Mobilitätsservices (bspw. Art der Sitze) auch den Komfort beim Umsteigen sowie die Betreuung der Reisenden (Handhabung von Beschwerden, Unterstützung von mobilitätseingeschränkten Personen, etc.) (He & Csiszár, 2020). Casadó et al. (2020) betonen, dass die befragten jüngeren Reisenden den Wunsch äußern, dass die Qualität von öffentlichen Mobilitätsservices erhöht werden muss, um weitere Anreize für die Verwendung dieser zu schaffen. Spickermann et al. (2014) identifizieren, dass spezifische Reisegruppen gewillt sind für mehr Komfort einen höheren Preis zu bezahlen. Ein Beispiel hierfür ist die Wahl der 1. Klasse in der Bahn, um einen höheren Reisekomfort zu haben. Schwinger & Krempels (2019) betonen, dass der Komfort einer der Hauptgründe für den motorisierten Individualverkehr ist, sodass multimodales Reiseverhalten durch höheren Komfort gefördert werden kann. Weiterhin trägt die Sauberkeit des jeweiligen Mobilitätsservice zum empfundenen Komfort bei (Casady, 2020; Gilibert & Ribas, 2019).

Umweltverträglichkeit: Insbesondere in den letzten Jahren hat die Bedeutung umweltverträglich zu reisen erheblich an Bedeutung gewonnen. In ihrer Untersuchung zu Bedürfnissen von Kindern und Jugendlichen an integrierten MaaS-Plattformen wird die Verringerung des motorisierten Individualverkehrs hin zu umweltverträglicheren Mobilitätsservices angegeben (Casadó et al., 2020). Im Allgemeinen kann gezeigt werden, dass ein Wandel hin zu multimodalem Mobilitätsverhalten die Einhaltung der Nachhaltigkeitsziele wesentlich unterstützt (Casady, 2020; Habermann et al., 2016; Kyamakya & Mitrea, 2010).

Einfache Eingabe der Servicewünsche: Die einfache und als unkompliziert wahrgenommene Eingabe der individuellen Servicewünsche in die MaaS-Plattform ist ein wesentlicher Faktor, dass Reisende die Benutzung dieser als angenehm empfinden. Meske et al. (2020) untersuchen wie die Benutzeroberfläche von MaaS-Plattformen mit

dem Ziel einer besseren Entscheidungsunterstützung für den Reisenden gestaltet werden sollen. Eine Herausforderung hierbei ist, dass uneinheitliche und kompliziert zu bedienbare Eingabemechanismen (z. B. durch die Kombination von unterschiedlichen Servicewünschen) den kognitiven und zeitlichen Aufwand für den Reisenden erhöhen. Neben der Wahl der relevanten Servicewünsche, welche bei der Suche der Servicekombinationen berücksichtigt werden sollen, sind weitere optionale Angaben wie Laufgeschwindigkeit sowie Angabe, ob z. B. ein Fahrrad mitgenommen wird, relevant, um realistische Umsteigezeiten zu berechnen (Stopka et al., 2016).

2.2.2 Reisendenanforderungen an plattformbasierte Mobilitätsdienstleistungen

Nachdem in Kapitel 2.2.1 die gewünschten Serviceinformationen seitens des Reisenden identifiziert worden sind, werden im Folgenden Anforderungen an integrierte MaaS-Plattformen identifiziert. Die Berücksichtigung dieser Anforderungen trägt wesentlich dazu bei, dass Reisende einen einfacheren Zugang zu MaaS-Plattformen haben und damit Änderungen des Mobilitätsverhaltens induziert werden können. Hierbei wird nach funktionalen Anforderungen der Reisenden an integrierte MaaS-Plattformen (Kapitel 2.2.2.1) und Anforderungen zur Unterstützung der Serviceauswahl (Kapitel 2.2.2.2) differenziert.

2.2.2.1 Funktionale Anforderungen

Im Folgenden werden zunächst funktionale Anforderungen an integrierte MaaS-Plattformen identifiziert.

Verkehrsmittelübergreifend (Multimodal): Die Förderung der multimodalen Mobilität stellt ein wesentliches Merkmal von MaaS dar (Albrecht & Ehmke, 2016; Casady, 2020; Esztergár-Kiss & Kerényi, 2020; Li et al., 2019; Lyons et al., 2020). Reisenden sollen möglichst alle am Markt verfügbaren Mobilitätsservices angeboten werden, um eine individuelle Auswahl und Kombination dieser zu ermöglichen. Dies umfasst neben traditionellen Mobilitätsservices wie öffentliche Nah- und Fernverkehre, zum Beispiel der Deutschen Bahn, vermehrt auch neu aufkommende Services wie bspw. Ridesharing-Services (Esztergár-Kiss & Kerényi, 2020). Hierbei besteht eine wesentliche Herausforderung in der Berücksichtigung der heterogenen Eigenschaften der Mobilitätsservices. Während beispielsweise der öffentliche Nahverkehr auf Basis

eines Fahrplans agiert, weisen viele Ridesharing-Anbieter ein dynamisches Angebot auf, welches anfrageaktuell evaluiert werden muss. Ebenso sollten Reisenden z. B. Bikesharing-Services nur angezeigt werden, sofern diese (voraussichtlich) verfügbar sind. Li et al. (2019) geben einen Überblick über aktuelle MaaS-Konzepte und entwickeln eine generalisierte Übersicht über die Benutzererfahrung auf der Grundlage des Multi-Level-Service-Design-Modells. In ihrer Analyse hinsichtlich der Anzahl an integrierten Mobilitätsservices ist ersichtlich, dass 9 von den 15 MaaS-Plattformen ein vollständiges multimodales Angebot berücksichtigen. Dies umfasst sowohl Nah- und Fernverkehr, Taxifahrten, Car-, Bike- und Ridesharing-Services. Bei drei weiteren MaaS-Plattformen ist lediglich ein Mobilitätsservice nicht integriert. Dies ist überwiegend dadurch bedingt, dass die jeweilige MaaS-Plattform speziell für das städtische Gebiet entwickelt wurde. UbiGo, für die Stadträume Göteborg und Stockholm eingesetzt, bietet z. B. keine Fernverkehrsverbindungen an. Albrecht & Ehmke (2016) geben einen Überblick über die Funktionalität multimodaler Mobilitätsmanager und analysieren Berücksichtigung von spezifischen Reisendenanforderungen. In ihrer Analyse hinsichtlich des Grads der Multimodalität (Stand 2016) gehen sie feingranularer vor als Li et al. (2019) und differenzieren zwischen 14 unterschiedlichen Mobilitätsservices. Die betrachteten MaaS-Plattformen integrieren hierbei zwischen 3 (GoEuro, heute Omio) und 12 (Qixxit, 2019 von der Deutschen Bahn an Lastminute.com verkauft) unterschiedliche Mobilitätsservices.

Tür-zu-Tür-Mobilität: Die Integration einer Vielzahl von Mobilitätsservices in eine MaaS-Plattform ermöglicht den Reisenden eine Tür-zu-Tür-Mobilität. Dies bedeutet, dass Reisenden Servicekombinationen angeboten werden, welche die komplette Reisekette abdecken. Insbesondere die Integration von Bike- und Ridesharing-Services ermöglicht die Berücksichtigung der ersten und letzten Meile der jeweiligen Reisekette (Casady, 2020; Lyons et al., 2020; Stopka, 2014). Ebenso ist die Anzeige des Gehweges z. B. von der Haustür zur nächstgelegenen relevanten Haltestelle des Nah- und Fernverkehrs gewünscht (Stopka, 2014).

Integrierte MaaS-Plattform (anbieterübergreifend): In allen analysierten Beiträgen ist eine integrierte MaaS-Plattform eines der zentralen Merkmale zur Vermittlung von MaaS (Casady, 2020; Grotenhuis et al., 2007; Spickermann et al., 2014; Stopka, 2014). Die verkehrs- und anbieterübergreifende Recherche und Auswahl von Servicekombinationen wird für Reisende erleichtert, sofern diese eine einzige MaaS-Plattform

verwenden können. Dadurch wird vermieden, dass Reisende die Ergebnisse unterschiedlicher MaaS-Plattformen manuell kombinieren und vergleichen müssen. Dies würde zu einem erheblichen kognitiven und zeitlichen Mehraufwand führen (Grotenhuis et al., 2007).

Integrierte Buchung, Bezahlung und Fahrkarten: Eine weitere elementare Reisendenanforderung an integrierte MaaS-Plattformen ist ein integriertes Buchungs-, Zahlungs- und Ticketingsystem (He & Csiszár, 2020; Lyons et al., 2020; Stopka et al., 2018). Hierdurch wird Reisenden der Aufwand genommen bei unterschiedlichen Mobilitätsservices — insbesondere von unterschiedlichen Anbietern — Fahrkarten getrennt buchen und bezahlen zu müssen. Reisende wünschen die Buchung und Bezahlung einer einzelnen Fahrkarte für die gesamte Reisekette, sodass z. B. keine mehrfache Registrierung bei jedem verwendeten Mobilitätsserviceanbieter notwendig ist (Lyons et al., 2020). Um den Aufwand weiter für Reisende zu minimieren und die Verwendung von MaaS-Plattformen möglichst einfach zu gestalten, sollten Rabatt- und vorhandene Wochen-, Monats- oder Jahreskarten von einzelnen Mobilitätsservices bei der Buchung und Bezahlung integriert werden (He & Csiszár, 2020; Stopka et al., 2018).

Echtzeitinformationen: Die Zuverlässigkeit und Echtzeit der bereitgestellten Informationen zu Servicekombinationen sind ebenfalls entscheidend für Reisende (Casady, 2020; Spickermann et al., 2014; Stopka, 2014). Dies umfasst neben der prognostizierten Verspätung bei der Verwendung fahrplanbasierter Mobilitätsservices wie Nah- und Fernverkehr ebenso eine Echtzeitprognose zur Fahrzeit bei der Verwendung von Ridesharing-Services. Diese wird z. B. durch die aktuelle Verkehrssituation oder Verfügbarkeit von Verkehrsmitteln beeinflusst (Spickermann et al., 2014). Stopka (2014) präsentiert Ergebnisse einer empirischen Studie der TU Dresden hinsichtlich Reisendenanforderungen und Servicewünschen. Ein Ergebnis ist, dass Reisende Informationen über den Grund der Verspätung sowie eine Angabe zu möglichen Alternativen von Servicekombinationen wünschen. Hierbei sind möglichst zuverlässige Aussagen von hoher Bedeutung für Reisende. Durch die Bereitstellung von zuverlässigen Echtzeitinformationen und die Angabe des Verspätungsgrundes erhalten Reisende die Möglichkeit selbstständig aus der Wahl von alternativen Servicekombinationen zu wählen.

Skalierbarkeit: Die Skalierbarkeit des multimodalen Routingalgorithmus der MaaS-Plattform ist weiterhin wichtig (Horstmannshoff & Ehmke, 2020). Sofern weitere Mobilitätsservices und/oder Servicewünsche in der Suche integriert werden, sollte Reisenden weiterhin eine multikriterielle Entscheidungsunterstützung in effizienter Laufzeit ermöglicht werden.

2.2.2.2 Unterstützung der Serviceauswahl

Reisende benötigen adäquate Unterstützung bei der eigenständigen Serviceauswahl aus einer Menge an möglichen Servicekombinationen. Die Berücksichtigung der folgenden Anforderungen trägt dazu bei, den Aufwand für die Serviceauswahl zu minimieren, sodass Reisende die Benutzung von MaaS-Plattform als angenehm empfinden.

Auswahl aus einer Menge an Servicekombinationen: Reisenden stehen eine Vielzahl von heterogenen Mobilitätsservices zur Verfügung, welche in Abhängigkeit der individuellen Servicewünsche kombiniert werden können. Hierdurch ergibt sich eine Menge an möglichen Pareto-optimalen Servicekombinationen, aus welchen Reisende auswählen können. Ein Beispiel für Pareto-optimale Servicekombinationen ist eine günstige, aber langsame Servicekombination vs. eine schnelle, aber teure Servicekombination. Stopka et al. (2016) analysieren in ihrem Beitrag den Prototyp der MaaS-Plattform “Dynamic Seamless Mobility Information (Dynamo)” u.a. hinsichtlich der relevanten Reisendenanforderungen mittels einer quantitativen Studie. Hierbei wird der Wunsch nach einer geeigneten Auswahl aus einer Menge von Servicekombinationen betont. Grotenhuis et al. (2007) sowie Spickermann et al. (2014) heben hervor, dass sich die Mobilitätsservicewahl des Reisenden in Abhängigkeit unterschiedlicher angebotener Servicekombinationen ändern kann. Es ist wichtig zu betonen, dass die Anzeige einer einzigen Servicekombination in der Regel nicht ausreichend ist, da situative und gewohnheitsmäßige bedingte Servicewünsche des Reisenden nicht vollständig berücksichtigt werden können.

Angemessene Anzahl an Servicekombinationen: Zusätzliche Servicewünsche von Reisenden sowie zusätzlich integrierte Mobilitätsservices können zu einer größeren Anzahl an möglichen Pareto-optimalen Servicekombinationen führen. Zur Reduktion des zeitlichen und kognitiven Aufwands bei der Serviceauswahl sollte die Anzahl an Servicekombinationen angemessen sein. Dies bedeutet, dass Reisenden einerseits überhaupt eine Auswahl angeboten wird, denn aufgrund sich widersprechender

Anforderungen könnte eventuell gar keine Servicekombination gefunden werden. Andererseits sollte die Menge an angebotenen Servicekombinationen auch nicht zu groß werden, um eine Überforderung bei der Auswahl zu vermeiden (Ehmke & Horstmannshoff, 2020; Grotenhuis et al., 2007; Lyons et al., 2020; Ulloa et al., 2018).

Diversität der Servicekombinationen: Sofern die Menge an Servicekombinationen sehr groß ist, können Servicekombinationen, welche für den Reisenden vernachlässigbare Differenzen zu weiteren Servicekombinationen aufweisen, entfernt werden. Beispielsweise sollte eine Servicekombination, welche nur wenige Minuten schneller, aber wesentlich teurer im Vergleich zu einer anderen Servicekombination ist, Reisenden nicht notwendigerweise präsentiert werden (Horstmannshoff & Ehmke, 2020). Dies stellt eine ausreichende Diversität der präsentierten Servicekombinationen sicher. Dies ermöglicht Reisenden bei der Serviceauswahl auch auf situativ bedingte Servicewünsche eingehen zu können, welche der MaaS-Plattform nicht vollständig bekannt und die schwer formalisierbar sind (Casady, 2020; Lyons et al., 2020). Casady (2020) nennt als Beispiel Reisende unter Zeitdruck, welche in dieser Situation gewillt sind für ein Premium-Produkt zu bezahlen, um schneller dem individuellen Reisewunsch gerecht zu werden.

Unterstützung bei der Serviceauswahl: Reisende benötigen bei der eigenständigen Recherche und Auswahl aus der Menge an möglichen Servicekombinationen Unterstützung. Hierdurch wird der zeitliche und kognitive Aufwand bei der Serviceauswahl reduziert und die Verwendung von MaaS-Plattformen wird reisendenorientierter gestaltet (Alt et al., 2019; Ehmke & Horstmannshoff, 2020; Grotenhuis et al., 2007; Lyons et al., 2020; Wienken & Krömker, 2018). Viele MaaS-Plattformen bieten Reisenden Filteroptionen an, um die Auswahl an relevanten Servicekombinationen einzuzugrenzen. Reisende benötigen hierbei zusätzliche Informationen über den multimodalen Lösungsraum, um die Filteroptionen zielgerecht zu bedienen. Beispielsweise kann das Eingrenzen der möglichen Servicekombinationen durch gesetzte Einschränkungen hinsichtlich Fahrzeit und Preis zu einer leeren Menge führen (Ehmke & Horstmannshoff, 2020; Meske et al., 2020). Ehmke & Horstmannshoff (2020) stellen einen Ansatz vor, um den komplexen multimodalen Lösungsraum an Servicekombinationen für den Reisenden erklärbarer zu gestalten. Die Präsentation von Zusammenhängen zwischen unterschiedlichen Servicewünschen (z. B. wie viel mehr Fahrzeit muss investiert werden, bis erhebliche Kosteneinsparungen erzielt werden) kann Reisende in

der Verfeinerung individuell adäquater Servicekombination unterstützen. Zusätzlich können relevante Servicekombinationen außerhalb der gesetzten Filtereinschränkungen existieren, welche für Reisenden eventuell trotzdem relevant sind. Beispielsweise können durch die Akzeptanz von ein wenig mehr Fahrzeit wesentlich geringere Preise erzielt werden (Meske et al., 2020). Casadó et al. (2020) heben hervor, dass bei der Integration von Sharing-Services ebenso eine Information notwendig ist, ob z. B. das Abstellen von Leihfahrrädern für die letzte Meile am Zielort möglich ist. Zusätzlich ist eine Information wünschenswert, ob z. B. ein mobilitätseingeschränkter Umstieg an einer Haltestelle möglich ist (Bell, 2019; Stopka et al., 2015).

Alternative Zugangspunkte insbesondere für die erste und letzte Meile: Informationen über mögliche Zugangspunkte vom jeweiligen Start- bzw. Zielpunkt des Reisenden können ebenfalls bei der Serviceauswahl unterstützen (Bucher et al., 2017; Nasibov et al., 2016). Beispielsweise kann der Reisende zu einem Zugangspunkt (z. B. eine Haltestelle) eine etwas größere Gehdistanz zurücklegen im Vergleich zu einem nähergelegenen Zugangspunkt, um bedingt durch eine höhere Anzahl an möglichen Linien, welche den Zugangspunkt bedienen, im Falle eines kurzfristigen Ausfalls der Verbindung mehrere Alternativen zu haben. Diese Informationen sind insbesondere für die erste und letzte Meile der Reisekette von Relevanz, da im urbanen Raum meistens mehrere Zugangspunkte benutzt werden können.

Integration von Benutzerprofilen: Die Integration von Benutzerprofilen in einer MaaS-Plattform ist ein weiterer Baustein, welcher zur Aufwandsreduktion für den Reisenden bei der Serviceauswahl beiträgt. Basierend auf vorherigen Suchanfragen und Buchungen können spezifische gewohnheitsmäßige Eigenschaften wie präferierte Servicewünsche gelernt werden. Diese individuellen Servicewünsche sollten bei der Auswahl an präsentierten Servicekombinationen berücksichtigt werden (Dolinayova et al., 2018; Lyons et al., 2020; Stopka et al., 2016). Stopka et al. (2016) betonen, dass das Hinterlegen von Dokumenten wie Rabattkarten weiter zur Aufwandsreduktion beiträgt. Ebenso ist die Suchhistorie insbesondere für Pendler von hoher Relevanz, da diese regelmäßig die gleiche Anfrage hinsichtlich Tür-zu-Tür-Mobilität stellen und damit auf vergangene Anfragen zugreifen können sollten (Schwinger & Krempels, 2019; Stopka et al., 2016).

2.2.3 Forschungsbedarf

Jeder Reisende hat individuelle Servicewünsche, welche u.a. situativ und gewohnheitsmäßig bedingt sind. Diese sollten von der jeweiligen MaaS-Plattform individuell berücksichtigt werden können. Dies ist bisher nicht vollständig gegeben. Beispielsweise werden schwierig quantifizierbare Servicewünsche wie Komfort und Umweltverträglichkeit bisher überwiegend nicht berücksichtigt.

Weiterhin sollte eine MaaS-Plattform diverse funktionale Anforderungen erfüllen, um den zeitlichen und kognitiven Suchaufwand für Reisende zu reduzieren. Dies umfasst eine vollständige Integration aller verfügbaren Mobilitätsservices in einer MaaS-Plattform, sodass eine anbieter- und verkehrsmittelübergreifende Suche, Buchung und Bezahlung von Tür-zu-Tür-Mobilität unter Einbezug von Echtzeitinformationen ermöglicht wird. Diesbezüglich sind in den letzten Jahren erhebliche Fortschritte erzielt worden, wie in der Fallstudie in Kapitel 2.4 gezeigt wird.

Ein wesentlicher Forschungsbedarf besteht in der Aufwandsreduktion für Reisende bei der Serviceauswahl aus einer Menge an verfügbaren Servicekombinationen. Die Anzahl an Servicekombinationen sollte angemessen und Reisendenindividuell veränderbar sein. Zusätzlich ist eine ausreichende Diversität innerhalb der präsentierten Servicekombinationen in Abhängigkeit der individuellen Servicewünsche sicherzustellen. Die Reduzierung der “black-box“-Eigenschaft von MaaS-Plattform im Zuge der Unterstützung bei der Serviceauswahl kann als ein wesentlicher Schritt hin zu reisendenorientierten MaaS-Plattformen identifiziert werden (Alt et al., 2019). Weiterhin werden die diversen Vor- und Nachteile von Haltestellen insbesondere für die erste und letzte Meile bisher nicht vollständig in der Suche und Auswahl von Servicekombinationen berücksichtigt.

2.3 Berücksichtigung von Reisendenanforderungen durch multimodale Routingalgorithmen

Im Folgenden wird aufgezeigt, in welchem Umfang die zuvor identifizierten Reisendenanforderungen aus Kapitel 2.2 durch state-of-the-art multimodale Routingalgorithmen berücksichtigt werden (F_2). Diese spielen eine zentrale Rolle für MaaS-Plattformen, da die individuelle Ermittlung der Menge an Pareto-optimalen Servicekombination mittels dieser Algorithmen vorgenommen wird. Hierbei erfolgt eine Suche

in komplexen multimodalen Netzwerken bestehend aus einer Vielzahl integrierter Mobilitätsservices. Diese umfassen sowohl fahrplanbasierte Services wie Nah- und Fernverkehr sowie dynamische Sharing-Services. Einerseits sind in der Entwicklung von effizienten Routingalgorithmen enorme Fortschritte erzielt worden (Bast et al., 2015), andererseits werden die aufgezeigten Reisendenanforderungen bisher nicht vollständig berücksichtigt. Beispielsweise ist die Integration von Echtzeitinformatio- nen nicht möglich, sofern zeitaufwendige Vorberechnungen eine Grundvoraussetzung für eine hohe Effizienz des Algorithmus sind. Reisende haben in diesem Fall nicht die Möglichkeit bei unerwarteten Unterbrechungen der Reisekette eine kurzfristige Änderung unter Einbezug der aktuell verfügbaren Mobilitätsservices zu planen (Alt et al., 2019). Die Auswahl wird im Folgenden auf Algorithmen eingegrenzt, welche zumindest Teilanforderungen aus Kapitel 2.2 umfassen.

Tabelle 2.1 gibt einen Überblick über multimodale Routingalgorithmen sowie den Grad der Berücksichtigung von identifizierten Reisendenanforderungen in der Suche nach Servicekombinationen (SK in Tabelle 2.1). Hierbei werden nur solche Reisende- nanforderungen betrachtet, welche direkt von dem zugrundeliegenden multimodalen Routingalgorithmus abhängig sind. Weitere Komponenten einer MaaS-Plattform werden an dieser Stelle nicht analysiert, da der Fokus auf den Einfluss der Algorithmen liegt. Ansätze zur Unterstützung Reisender bei der Serviceauswahl werden nicht betra- chtet, da diese auf der Menge an Servicekombinationen als Ergebnis des multimodalen Routingalgorithmus arbeiten. Folglich sind diese nicht von dem Algorithmus abhängig. Weiterhin sind Anforderungen wie z. B. eine anbieter- und verkehrsmittelübergreifende Suche, Buchung und Bezahlung von Tür-zu-Tür, etc., durch geeignete Geschäfts- und Kooperationsmodelle zwischen den Mobilitätsserviceanbietern bedingt, sodass diese nicht durch den zugrundeliegenden Algorithmus einer MaaS-Plattform abhängig sind. Eine einfache Eingabe der individuellen Servicewünsche und die Integration von Benutzerprofilen sind vorwiegend vom reisendenorientierten Interface-Design der MaaS-Plattform bedingt.

Delling et al. (2013a) haben einen in der Praxis vielseitig eingesetzten multimodalen Routingalgorithmus RAPTOR (Round Based Public Transit Optimized Router) ent- wickelt. Hierbei wird die Menge an Pareto-optimalen Servicekombinationen in Runden ermittelt, wobei in jeder weiteren Runde ein zusätzlicher Umstieg ermöglicht wird. Eine Integration von Echtzeitinformatio- nen wie verspäteter Nah- und Fer-

Table 2.1: Berücksichtigung von Reisendenanforderungen durch Routingalgorithmen
(x: Anforderung erfüllt, o: Anforderung unzureichend erfüllt,
k.A.: keine Angabe); SK: Servicekombinationen

	Gewünschte Serviceinformationen (2.2)	MaaS-Plattformen: Funktionale Anforderungen (2.2.2.1)		MaaS-Plattformen: Unterstützung der Serviceauswahl (2.2.2.2)		
	Multikriterielle Entscheidungsunterstützung	Integration von Echtzeitinformationen	Skalierbarkeit	Menge an SK	Angemessene Anzahl an SK	Diversität der SK
(Mc)RAPTOR (Delling et al. 2013; Delling et al. 2015)	x	x	o	x	x	x
Xerox Trip Planner (Ulloa et al. 2018)	x	x	k.A.	x	o	x
Multi-Label Correcting Algorithm (Dib et al. 2017)	x	k.A.	o	x	o	o
Sampling-Methoden (Horstmannshoff und Ehmke 2020)	x	k.A.	x	x	x	x

nverkehr ist möglich. Zur Sicherstellung einer angemessenen Menge an diversen Servicekombinationen verwenden Delling et al. Methoden der *fuzzy logic*. Bast et al. (2013) diskutieren ein filterbasiertes Verfahren, mit welchem eine ausreichend kleine und repräsentative Menge an Servicekombinationen erstellt wird. McRAPTOR ist eine Erweiterung zu RAPTOR und berücksichtigt zusätzliche Servicewünsche bei der Suche nach multimodalen Servicekombinationen (Delling et al., 2015).

Ulloa et al. (2018) stellen das *Xerox Trip Planner System* vor. Das wesentliche Ziel ist es Reisenden eine einfache und effiziente Suche, Auswahl und Buchung

von multimodalen Servicekombinationen unter Berücksichtigung einer Vielzahl von individuellen Servicewünschen zu ermöglichen. Hierzu optimieren Ulloa et al. (2018) die Ausführung des multimodalen Routingalgorithmus RAPTOR und kombinieren diesen mit einem statischen Algorithmus zur Berechnung von kürzesten Wegen in Graphen (Dijkstra bzw. effizientere Erweiterungen zu Dijkstra (Bast et al., 2015)). Die Integration von Echtzeitinformationen ist möglich, da keine Vorberechnungen erforderlich sind. Die Berechnung einer Menge an Servicekombinationen erfolgt, indem rekursiv Teilbereiche bereits identifizierter Servicekombination (z. B. Fahrt von Berlin nach Magdeburg mit der Regionalbahn) für weitere Servicekombinationen ausgeschlossen werden. Dies stellt eine Menge an diversen Servicekombinationen sicher, welche dem Reisenden präsentiert werden können. Allerdings erfolgt hierbei keine Eingrenzung bzgl. der Anzahl an identifizierten Servicekombinationen, sodass je nach Anfrage der Reisende durch die Anzahl an präsentierten Servicekombinationen überfordert werden kann. Hinsichtlich der Skalierbarkeit des vorgestellten *Xerox Trip Planner System* liegen keine Angaben vor.

Dib et al. (2017) betonen, dass das Finden der kompletten Menge an Pareto-optimalen Servicekombinationen im multimodalen Kontext sehr zeitaufwendig ist. Auf Basis einer erweiterten Formulierung zur Modellierung des multimodalen Graphen (sie ermöglichen realistische Umsteigezeiten innerhalb eines Bahnhofes), führen sie den *Multi-Label Correcting Algorithmus* ein. Die wesentliche Idee hierbei ist, dass ein Standardverfahren zur Berechnung von kürzesten Wegen mit zusätzlichen Labeln zu jedem Knoten des multimodalen Graphen erweitert wird. In diesen Labeln werden die jeweiligen Pareto-optimalen Werte der berücksichtigten individuellen Servicewünsche gespeichert. Dies ermöglicht effiziente Abbruchkriterien der Suche in einzelnen Pfaden des multimodalen Graphen, wenn die Dominanz der identifizierten Servicekombination bereits durch vorhandene Labelinformationen bewiesen werden kann. In ihrem vorgestellten Verfahren erfolgt dabei keine Eingrenzung auf eine angemessene Anzahl an Servicekombination sowie einer ausreichenden Diversität dieser. Bezüglich der Integration von Echtzeit werden keine Angaben gemacht.

Bei den zuvor genannten multimodalen Routingalgorithmen wird die vollständige Menge aller Pareto-optimalen Servicekombinationen identifiziert. Mit jeder zusätzlich berücksichtigten Serviceinformation wächst dabei der Aufwand zur Identifikation dieser Menge stark an. Delling et al. (2013a) sowie Dib et al. (2017) betonen, dass

sich in ihren jeweiligen Ansätzen die Laufzeit des Algorithmus stark erhöht, sofern mehr als drei Servicewünsche gleichzeitig berücksichtigt werden. Folglich ist keine ausreichende Skalierung gewährleistet. Horstmannshoff & Ehmke (2020) stellen ein Framework vor, bei welchem eine effiziente Skalierung sichergestellt ist. Die Kernidee ist die Verwendung von Sampling-Methoden, um eine Reihe von angemessenen Servicekombinationen entsprechend den individuellen Servicewünschen des Reisenden zu ermitteln. Eine effiziente Skalierung wird sichergestellt, indem eine Vielzahl von zweidimensionalen Mengen parallel berechnet werden. In jeder zweidimensionalen Menge wird eine Serviceinformation als unikriterielle Zielfunktion gesetzt, während eine weitere Serviceinformation als Nebenbedingung systematisch verändert wird. Bei zusätzlich berücksichtigten Servicewünschen erhöht sich dabei lediglich die Anzahl der parallel berechneten Mengen, nicht die Komplexität zur Berechnung einer dieser zweidimensionalen Mengen. Weiterhin wird eine angemessene Anzahl an diversen Servicekombinationen sichergestellt. In ihren Experimenten untersuchen Horstmannshoff und Ehmke Fernreisen zwischen großen Städten in Deutschland. Dabei werden reale Daten mehrerer innovativer Mobilitätsservices, wie z. B. Ridesharing-Dienste, Deutsche Bahn, FlixBus, Flüge und Nahverkehr integriert. Darüber hinaus werden in einer Fallstudie bis zu fünf Servicewünsche der Reisenden gleichzeitig berücksichtigt und eine effiziente Laufzeit gewährleistet. Hinsichtlich der Integration von Echtzeitinformationen werden keine Angaben gemacht.

Zusammenfassend kann festgestellt werden, dass in den letzten Jahren multimodale Routingalgorithmen entwickelt worden sind, welche eine Vielzahl identifizierter Reisendenanforderungen berücksichtigen. Allerdings erfüllt keiner der dargestellten Algorithmen alle Anforderungen. Insbesondere besteht hinsichtlich der Entwicklung von skalierbaren Routingalgorithmen in großen multimodalen Netzwerken unter Berücksichtigung einer Vielzahl von individuellen Serviceinformation des Reisenden sowie Integration von Echtzeitinformationen Forschungsbedarf. Weiterhin erwarten Reisende eine Unterstützung bei der Serviceauswahl. Die Berücksichtigung dieser Aspekte in MaaS-Plattformen würde zu einer reisendenorientierten Entscheidungsunterstützung beitragen.

2.4 Fallstudie zur Berücksichtigung von Reisendenanforderungen

Als aktuelles Fallbeispiel zur Berücksichtigung diverser Reisendenanforderungen dient die MaaS-Plattform Jelbi (www.jelbi.com). Diese MaaS-Plattform wird seit 2019 im Großraum Berlin eingesetzt und wurde von der litauischen Firma Trafi entwickelt. Trafi ist ein Vorreiter in der Entwicklung von MaaS-Plattformen für den urbanen Bereich und ist an der Einführung von MaaS-Plattformen u.a. in Bogotá, München und Zürich beteiligt (www.trafi.com). Im Folgenden wird Jelbi im Detail vorgestellt und hinsichtlich der Berücksichtigung von Reisendenanforderungen an MaaS-Plattformen evaluiert.

2.4.1 MaaS-Plattform Jelbi

Jelbi vereint die multimodale Routenauskunft mit integrierten Buchungs-, Bezahlungs- und Fahrkartenoptionen. Ein wesentliches Merkmal von Jelbi ist ein verkehrsmittel- und anbieterübergreifendes Angebot diverser Mobilitätsservices in einer Plattform, sodass multimodale Mobilität von Tür-zu-Tür für den Reisenden ermöglicht wird. Folgende Mobilitätsservices sind integriert:

- ÖPNV: Nah- und Fernverkehr der Berliner Verkehrsbetriebe (BVG) sowie des Verkehrsverbunds Berlin Brandenburg (VBB)
- E-Scooter-Sharing: TIER, Voi., Lime
- Bikesharing: Nextbike
- E-Moped-Sharing: Emmy, TIER
- Carsharing: MILES, Mobileeee, DB Flinkster, Cambio, Greenwheels
- Ridesharing: BerlKönig
- Taxi Berlin

Durch Jelbi-Stationen wird Reisenden ein einfacher Zugang zu unterschiedlichen Mobilitätsservices ermöglicht. Diese Stationen befinden sich an zentralen S+U-Bahnhöfen im Stadtgebiet und bündeln eine Vielzahl von Sharing-Services. Insbesondere für die

erste und letzte Meile der Reisekette stehen somit eine Vielzahl von Mobilitätsservices zur Verfügung, wenn der Reisende z. B. an der U-Bahn aussteigt und die letzte Meile hin zum Zielpunkt flexibel mit einem Sharing-Service zurücklegen kann. Zum Zeitpunkt dieser Recherche existieren 11 Jelbi-Stationen im Großraum Berlin, wobei weitere geplant sind, um zukünftig ein flächendeckendes Angebot zu ermöglichen.

Jelbi bietet Reisenden die Einrichtung eines zentralen Jelbi-Benutzerkontos an. Sobald der Reisende seine Kontakt- und Zahlungsdaten sowie optional den Führerschein hinterlegt hat, werden diese automatisiert für alle verfügbaren Mobilitätsservices verwendet. Vorherige Suchanfragen und Buchungen werden in diesem gespeichert, sodass Reisende einfach auf diese zurückgreifen können, was insbesondere für Pendler von hoher Bedeutung ist.

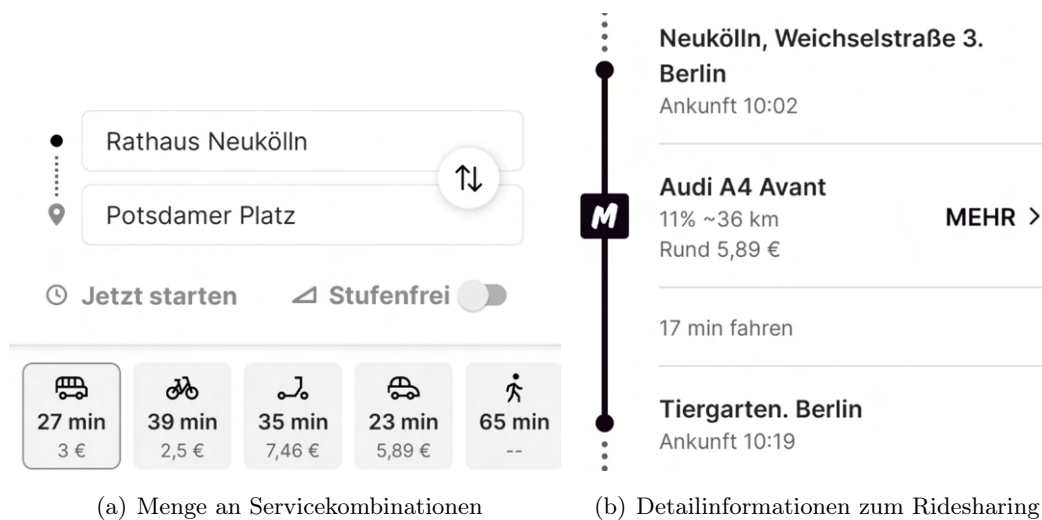


Figure 2.2: Jelbi-Servicekombinationen

Das Ergebnis einer exemplarischen Suche vom Rathaus Neukölln zum Potsdamer Platz ist in Abbildung 2.2 dargestellt. Reisende geben Start- und Zielpunkt sowie die gewünschte Abfahrtszeit an. Zusätzlich kann für mobilitätseingeschränkte Reisende Servicekombinationen, welche Stufensteigen enthalten, ausgeschlossen werden. Abbildung 2.2(a) zeigt einen Ausschnitt aus der Menge an verfügbaren Servicekombinationen. Die Verwendung des traditionellen Nahverkehrs (linkes Icon, U- und S-Bahn) dauert 27 Minuten und kostet 3 €. Detaillierte Informationen zur Servicekombination

nen mit zusätzlichen Angaben zur Anzahl der Umstiege, Gehdistanz und aggregierte Wartezeit sind verfügbar, sofern der Reisende auf das jeweilige Icon klickt. Bei allen Mobilitätsservices werden Echtzeitinformationen berücksichtigt. Beispielsweise werden aktuelle Angaben zur Verfügbarkeit von E-Scootern, Fahrrädern und verfügbaren Ridesharing-Fahrten angezeigt. Sofern der Reisende preissensitiver ist, würde sich die Verwendung eines Bikesharing-Services (39 Minuten, 2,50 €) oder das Laufen der Strecke anbieten (65 Minuten). Abbildung 2.2(b) zeigt detaillierte Informationen zur schnellsten Servicekombination via Carsharing mit weiteren relevanten Zusatzinformationen wie Fahrzeugtyp sowie Akkustand. Für Reisende, welche einen ausreichend hohen Akkustand erwarten, kann diese Information von hoher Relevanz sein (Lyons et al., 2020).

2.4.2 Evaluation der MaaS-Plattform Jelbi

Jelbi ermöglicht Reisenden im Großraum Berlin ein multimodales Reiseerlebnis von Tür-zu-Tür, in dem vielfältige Mobilitätsservices unter Berücksichtigung von Echtzeitinformationen in einer integrierten MaaS-Plattform eingebunden werden. Alle Phasen des Mobilitätsserviceprozesses von der Suche und Bereitstellung der Informationen zu Servicekombinationen bis zur Buchung und Bezahlung dieser werden abgedeckt (Albrecht & Ehmke, 2016). Jelbi erfüllt somit alle funktionalen Anforderungen an eine integrierte MaaS-Plattform (Kapitel 2.2.2.1).

Hinsichtlich der gewünschten Serviceinformationen (Kapitel 2.2) werden eine Vielzahl von Serviceinformationen des Reisenden berücksichtigt (Fahrzeit, Preis, Anzahl der Umstiege, Gehdistanz, Wartezeit sowie Verfügbarkeit). Die Integration von weiteren Servicewünschen wie geschätzte Zuverlässigkeit auf Basis vorheriger Fahrten sowie die Umweltverträglichkeit der jeweiligen Servicekombinationen würden zu einer noch umfangreicheren, multikriteriellen Entscheidungsunterstützung für den Reisenden beitragen. Eine einheitliche und einfache Eingabe der Servicewünsche reduziert den zeitlichen und kognitiven Aufwand für den Reisenden.

Hinsichtlich der Anforderungen zur Unterstützung der Serviceauswahl (Kapitel 2.2.2.2) steht Reisenden eine Menge an möglichen Servicekombinationen zur Auswahl. Diese wird ihnen strukturiert unter Angabe wesentlicher Informationen wie Fahrzeit und Preis dargestellt. Zu jeder Servicekombination können sich Reisende Detailinformationen zur konkreten Servicekombination, Gehdistanz, Wartezeit sowie

dem Akkustand von E-Scootern und Leihfahrrädern/Autos anzeigen lassen. Hierbei wäre wünschenswert, dass diese Informationen bereits in der Übersicht der verfügbaren Servicekombinationen angezeigt werden, sodass Reisende besser bei der Serviceauswahl unterstützt werden. Beispielsweise wird in Abbildung 2.2(a) für den Reisenden der Eindruck erweckt, dass die Verwendung von E-Scootern (drittes Icon von links) im Vergleich zur Wahl des traditionellen Nahverkehrs (erstes Icon von links) schlechter ist, da ersteres sowohl länger dauert, sowie teurer ist. Dabei werden die Vorteile der Verwendung des E-Scooters im Vergleich zum traditionellen Nahverkehr wie geringere Gehdistanz und Wartezeit sowie keine Notwendigkeit zum Umstieg für den Reisenden erst nach einem manuellen Vergleich offensichtlich. Dies führt zu einem erhöhten zeitlichen und kognitiven Aufwand bei der Auswahl aus einer Menge an verfügbaren Servicekombinationen. In den zentralen Benutzerprofilen zur Speicherung vorheriger Suchanfragen und Buchungen sowie zur Hinterlegung von Kontakt- und Zahlungsdaten lassen sich derzeit keine vorhandenen Rabattkarten wie z. B. vorhandene Jahreskarten für die BVG hinterlegen. Hierdurch wird die Verwendung dieser MaaS-Plattform für solche Reisende mit vorhandenen Rabattkarten unpraktisch.

Zusammenfassend können vereinzelte Optimierungspotentiale bei der MaaS-Plattform Jelbi identifiziert werden, um diese noch reisendenorientierter zu gestalten. Einige Aspekte sind hierbei durch den zugrundeliegenden multimodalen Routingalgorithmus bedingt, wie z.B. die Berücksichtigung zusätzlicher Serviceinformationen bei der Suche und Auswahl von Servicekombinationen. Ebenso können hinsichtlich des Interface-Designs der MaaS-Plattform Verbesserungsvorschläge angemerkt werden, um den zeitlichen und kognitiven Suchaufwand für Reisende zu minimieren.

2.5 Forschungsbedarf zur Bereitstellung von MaaS-Plattformen

In diesem Beitrag sind individuelle, komplexe Reisendenanforderungen an integrierte MaaS-Plattformen systematisch erfasst worden. Eine Berücksichtigung dieser im Design von MaaS-Plattformen kann eine Änderung des Mobilitätsverhaltens hin zu multimodaler Mobilität induzieren. Im Folgenden werden die vielfältigen

Reisendenanforderungen zusammengefasst und Forschungs- und Optimierungsbedarf zur Bereitstellung von MaaS-Plattformen identifiziert (F_3).

Reisende wünschen eine multikriterielle Entscheidungsunterstützung bei der Suche, Auswahl und Buchung von Mobilitätsservices. Neben bereits in vielen MaaS-Plattformen berücksichtigten Servicewünschen wie Fahrzeit, Preis und Anzahl der Umstiege, sollten auch komplexere Servicewünsche von Reisenden wie z. B. Sicherheit, Zuverlässigkeit und Umweltverträglichkeit integriert werden. Hierdurch wird ermöglicht, dass individuelle Servicewünsche Reisender, welche situativ sowie gewohnheitsmäßig bedingt sein können, umfassend berücksichtigt werden und damit eine reisendenorientierte Auswahl an Servicekombinationen zur Verfügung steht. Hinsichtlich der Integration weiterer komplexer Servicewünsche von Reisenden in MaaS-Plattformen besteht Forschungsbedarf.

Weiterhin weisen Reisende diverse funktionale Anforderungen an plattformbasierte Mobilitätsdienstleistungen auf. Eine MaaS-Plattform soll Reisenden eine anbieter- und verkehrsmittelübergreifende Mobilität von Tür-zu-Tür ermöglichen. Eine integrierte Buchung, Bezahlung und das Ausstellen der Fahrkarten in einer integrierten Plattform stellt sicher, dass Reisende ausschließlich eine integrierte Plattform verwenden müssen, um auf eine Vielzahl verfügbarer Mobilitätsservices zuzugreifen. Zusätzlich sind Echtzeitinformationen für Reisende von hoher Relevanz. Wesentlicher Forschungsbedarf besteht in der zeitlichen und kognitiven Aufwandsreduktion für Reisende bei der Suche und Auswahl von Servicekombinationen. Reisende wünschen oftmals eine eigenständige Auswahl aus einer angemessenen und diversen Menge an Servicekombinationen. Zusätzlich sind Informationen über den komplexen multimodalen Lösungsraum wichtig, um die "black-box"-Eigenschaft der verfügbaren Servicekombinationen bei der Suche und Auswahl zu reduzieren.

Für eine Vielzahl der identifizierten komplexen Reisendenanforderungen an MaaS-Plattformen ist der zugrundeliegende multimodale Routingalgorithmus von zentraler Bedeutung. In den letzten Jahren sind einige Algorithmen veröffentlicht worden, welche Teilaspekte der identifizierten komplexen Reisendenanforderungen berücksichtigen. Kapitel 2.3 gibt einen Überblick über diese multimodalen Routingalgorithmen und den Grad der Berücksichtigung der Reisendenanforderungen. Da keiner der untersuchten Algorithmen alle Anforderungen vollständig erfüllt, besteht diesbezüglich weiterhin Forschungs- und Optimierungsbedarf. Eine umfassende skalierbare und

multikriterielle Entscheidungsunterstützung, welche dem Reisenden unter Einbezug von Echtzeitinformationen eine angemessene Menge an diversen Servicekombinationen zur eigenständigen Auswahl ermöglicht, unterstützt den Reisenden wesentlich bei der Verwendung von MaaS-Plattformen.

Jelbi ist eine aktuelle multimodale MaaS-Plattform für den Großraum Berlin. Bei der Analyse zur Integration der identifizierten Reisedenfordernung kann festgestellt werden, dass Reisende in allen Phasen des Mobilitätsserviceprozesses weitgehend unterstützt werden. Hierbei lassen sich hinsichtlich der Berücksichtigung komplexer Servicewünsche Reisender, der fehlenden Integration von Rabattkarten, sowie der Unterstützung des Reisenden bei der Serviceauswahl Verbesserungspotentiale identifizieren.

Chapter 3

Identifying alternative stops for first and last mile urban travel planning

Abstract

Urban travelers today are seeking increasingly more information to plan their optimal itinerary, based on additional factors other than scheduled departure times. Still some route planning applications provide a simple approach with a few parameter settings (e.g. to minimize travel time between two specific places at a certain time) and without any multimodal solutions. Our approach provides travelers with a set of Pareto-optimal nearby stops that presents a number of traveler preferences in an easily comprehensible and quickly calculable manner. We display first and last mile stops that fall on a Pareto front based on multiple criteria such as travel time, number of transfers, and frequency of service. Our algorithm combines stop and route-based information to quickly present the traveler with numerous nearby quality options for their itinerary decision making. We expand this algorithm to include multimodal itineraries with the incorporation of free-floating scooters to investigate the change in stop and itinerary characteristics. We then analyze the results on the star-shaped public transport network of Göttingen, Germany, to show what advantages stops on the Pareto front have as well as demonstrate the increased effect on frequency and service lines when incorporating a broadened multimodal approach.

Keywords Route Planning, Pareto-optimal, Stop and Route Optimization, Multimodal

Contents

3.1	Introduction	57
3.2	Related work	58
3.2.1	Multimodal routing	59
3.2.2	Traveler preferences	60
3.3	Framework for identifying relevant first and last mile stops	63
3.3.1	Stop-based methodology	64
3.4	Experimental results	67
3.4.1	Design of experiment	68
3.4.2	Stop-relevant results	69
3.4.3	Results from scooter implementation	72
3.4.4	Visualizing results of comparative scooter implementation	75
3.5	Discussion	75
3.6	Conclusion	77
	Appendix	78
A	Further percentual changes by districts	78

3.1 Introduction

Modern public transport travelers expect a high quality of service and have varying priorities when creating their individual itineraries. Currently, several widespread urban route planning algorithms are focused on using time-dependent, route-based optimization to minimize preferences including the traveler’s travel time. However, while this itinerary may be optimal at a given moment, this may change with the time of day or with traveler itinerary preferences. While applications like GoogleMaps, City Mapper, and others have made large strides in recent years of developing their navigation tools to be traveler-friendly, there still needs to be a way to make information about relevant nearby stops for the first and last mile more transparent to the traveler considering a multitude of traveler preferences. Our stop-based optimization (SBO) framework aggregates detailed information from public transport route-based information and stop characteristics to give the traveler a simplified overview of multiple criteria for their route planning.

Current public transportation literature primarily revolves around route-based preferences, such as walking time or total travel time that are calculated in accordance with the traveler’s available routes (Mulley et al., 2018). However, there is also research into other important information revolving around a traveler’s nearby stops either at the origin or destination (Nasibov et al., 2016; Yang et al., 2020). Currently, only a few researchers propose to integrate both route-based criteria with first and last mile stop information, like accessibility, number of lines and frequency of transport services at that stop. By focusing on this unique combination of criteria, we can offer a comprehensive decision tool for the traveler for more informed travel planning.

Our SBO approach incorporates a mixture of quickly calculable route and pre-computed stop-based information to provide a Pareto-optimal set of nearby stops for the first and last mile of the traveler’s itinerary. For instance, besides travel time, price, and number of transfers, travelers care about stop-based information like frequency, accessibility, and number of public transport lines. Additionally, the overall walking distance can be of high importance for the traveler. When the route and stop-based information are taken into account, several Pareto-optimal stops result due to the conflicting objectives of these preferences. For example, there may be a stop that is operated at a high frequency. This results in backup options for the traveler. On the other hand, the itinerary starting at this stop may be more expensive compared to the

itinerary starting at another stop within walking distance from the origin. Presenting these diversified solutions in a multimodal setting to the traveler is important since it broadens a traveler’s decision making according to personal preferences and context, like personal mobility or time of day (Lyons et al., 2020).

In the following, we will analyze the potential of combining route-based and stop-based information to better inform the traveler of the characteristics of their first and last mile decisions. Our experiments are based on the public transport bus network of Göttingen. In addition to the bus network with walking edges, we will consider unscheduled, innovative modes of transportation, such as electric scooters. The novelty of this approach differs from route-based planning to focus more on the choice of stops for the first and last mile of the itinerary by including stop-based information into the decision-making process. We incorporate multiple traveler preferences and allow for various modes of transportation within our model to build upon recent work in travel planning.

Section 3.2 focuses on how our work contributes and builds upon current urban route planning literature. Section 3.3 highlights the problem structure, our stop-based methodology, and the algorithm we use to identify the Pareto-optimal stops. Section 3.4 outlines our experiments that analyze the quality of stops on the public transport network of Göttingen, Germany, and also incorporates scooters as a comparative example of how our approach can expand to multimodal networks. Section 3.5 discusses limitations of this study, offers further research avenues for expansion of this approach and its contributions on the long term of travel planning policies. Finally, Section 3.6 summarizes our approach and its impact on multimodal public transport.

3.2 Related work

Urban route planning research has markedly expanded in recent years as it becomes easier to incorporate into travelers’ decision making. In this section, we review how traveler preferences can help expand classic route-based optimization to help multi-preference travelers navigate complex multimodal networks. Section 3.2.1 highlights various multi-criteria and multimodal optimization research that motivated the development of our algorithm for finding high quality, Pareto-optimal first and last mile stops. Section 3.2.2 explores current work on incorporating traveler preferences on itinerary decision making.

3.2.1 Multimodal routing

Traditional route-based optimization typically requires a fixed origin, destination, and start time. However, recent research has expanded this route-based optimization view (Willing et al., 2017). Delling et al. (2013a), Delling et al. (2013b) use public transport route planning techniques to propose a bi-criteria itinerary planning algorithm. The authors use optimization rounds of the multimodal network to produce a Pareto-optimal set while limiting the computational time. Dib et al. (2017) introduce a label-based multi-criteria routing algorithm considering travel time, number of transfers and the total walking time as traveler preferences. Bozyigit et al. (2017) extend Dijkstra’s algorithm to enable taking walking distance as well as number of transfers as additional relevant traveler preferences into account. Therefore, they introduce a penalty rule set integrated into Dijkstra’s algorithm. This research utilizes both stop-based information as well as aggregated route data to form a multi-criteria objective. We explore the Pareto front and how travel options change based on this stop and route-based approach.

Redmond et al. (2020) limit the computational time of optimizing on multimodal driving and flight networks by focusing on a set of nearby first and last mile airports for the traveler’s decision. This focus on selecting nearby airports showed that always myopically choosing the closest or largest nearby airport can result in less reliable itineraries. Ge et al. (2021) highlight the importance that multimodal itinerary applications have in integrating all available mobility services and data sources into one framework to support the traveler in their decision-making process. Bucher et al. (2017) propose to precompute candidate stops for the first and the last mile in a preprocessing step of the actual routing. Based on the candidate solutions, the routing algorithm focuses primarily on these. Therefore, the computational effort can be significantly reduced by considering a select set of nearby first and last mile stops.

Nykl et al. (2015) integrate multiple traveler preferences using a meta-graph that is able to incorporate multimodal itineraries. They also use a multi-criteria approach with time, distance, emissions, physical effort, and price as their parameters. Their approach is defined by a two-stage algorithm that capitalizes on using existing itinerary planning meta-data to set the weights on their graph. Horstmannshoff & Ehmke (2022) propose a sampling framework to approximate the Pareto-optimal set of itineraries. In particular, they focus on efficient scalability with respect to

the number of considered traveler preferences in a large multimodal network. In addition, they present insights into itinerary characteristics which can be embedded into decision tools for the traveler.

McKenzie (2019) examines scooter and bike-share usage in the United States capital of Washington, D.C. The author focuses on the spatial and temporal distributions of scooter-sharing itineraries in the area. Zou et al. (2020) also look at how e-scooters compete against and complement public transport and bike share in Washington D.C., and show other reasons for choosing included public health priorities and access in underserved areas. Esztergár-Kiss & Lizarraga (2021) utilize surveys across five European cities to discover that its popularity is driven by flexibility and speed, despite safety and road-sharing concerns. Further surveys are done by Jie et al. (2021) to show factors associated with shared mobility usage including gender, employment status, and income. Smith (2020) demonstrates the time saving and accessibility benefits of incorporating scooters into multimodal itineraries in Chicago. Shokouhyar et al. (2021) examines the impact COVID-19 had on shared mobility, and the authors highlight the need to consider social and environmental factors when considering shared mobility implementation. Our research integrates shared mobility in multimodal networks with traveler preferences for an easily comprehensible and quickly computed tool for route planning.

3.2.2 Traveler preferences

Understanding what is important to a traveler while navigating a public transport network is key to developing route planning tools. Javadian Sabet et al. (2021) highlight that the individual context of the traveler is of high importance to be able to take traveler preferences into account. Studies like Sharples (2017) focus qualitatively on what is needed to educate travelers in order to increase traveler competence to be able to make better use of available transport options. They present a model (context dimension tree) which allows the real-time integration of traveler requirements, the individual traveler preferences as well as the individual profile into the decision-making progress of the traveler. A considerable amount of literature has been published to identify traveler preferences for multimodal mobility by mainly analyzing traveler surveys. Grotenhuis et al. (2007) outline how integrated multimodal information can affect a traveler's choice. The authors highlight what types of information are

necessary and the importance that travelers place on travel time and minimizing effort in route planning. Table 3.1 gives an overview of the preferences considered in the literature differentiated according to route and stop-based information.

Table 3.1: Traveler preferences

	Route-based Preferences					Stop-based Preferences		
	Travel time	Price	# transfers	Walking time	Waiting time	freq.	# lines	Accessibility
Grotenhuis et al. (2007)	x	x	x	x	x	-	-	-
Spickermann et al. (2014)	x	x	x	-	x	-	-	-
Stopka (2014)	x	x	x	x	-	-	-	-
Stopka et al. (2015)	x	x	x	-	-	-	-	-
Gilibert & Ribas (2019)	x	x	-	x	-	-	-	-
He & Csiszár (2020)	x	x	x	x	x	-	-	-
Mulley et al. (2018)	x	-	-	x	-	x	-	-
Yang et al. (2020)	x	x	x	-	-	-	x	-
Wu et al. (2018)	x	x	x	-	-	-	x	-
Nasibov et al. (2016)	x	-	x	x	-	-	x	-
Fatima & Moridpour (2019)	x	x	x	x	-	x	-	x
Esztergár-Kiss (2019)	x	x	x	x	x	-	-	x
Mandžuka (2021)	x	x	x	x	x	-	-	x
SBO approach	x	-	x	x	-	x	x	x

Travel time, price, and number of transfers are the most prevalent preferences for route planning. In addition, the consideration of the overall walking time during an

itinerary and the overall waiting time is of high importance as well (Gilibert & Ribas, 2019; Grotenhuis et al., 2007; He & Csiszár, 2020; Spickermann et al., 2014; Stopka, 2014; Stopka et al., 2015). Please note that further route-based preferences such as comfort are not included in this overview as these are rarely mentioned. We refer to Horstmannshoff (2022) for a detailed overview of relevant traveler preferences in a multimodal setting.

Further work integrate stop-based characteristics into the planning of multimodal itineraries. Yan et al. (2019) show the significance that low quality last mile stops have in deterring travelers from using public transport options. Thus, there is a need to incorporate additional preferences about stop-based information into the search to increase the option quality of first and last mile stops in route planning. Recent research has attempted to model these preferences in traveler decision making. Mulley et al. (2018) demonstrate through stated choice experiments that travelers are generally willing to walk further for a more frequent public transport service as well as to achieve travel time savings. Yang et al. (2020) develop a Markov game to sequence travelers' interactive public transport mode choices based on a set of features. The authors highlight that besides common preferences such as travel time, price and number of transfers, the number of choices available is of relevance as well. Wu et al. (2018) use a preference-learning algorithm to predict travelers' decisions when evaluating a new public transportation plan. The goal of this paper is to integrate both route-based information and stop-based information into a comprehensive decision tool for travelers trying to navigate a multimodal urban public transportation network. Nasibov et al. (2016) examine route planning from a perspective of stop-based preference degrees. The authors develop a fuzzy preference model that factors in the stop's activity, the count of the public transport lines that run through that stop, the travel time, the number of transfers and the walking distance to the stop. Fatima & Moridpour (2019) emphasize that due to the aging population further challenges in the planning of multimodal mobility arise. In particular, mobility applications should provide information whether the itinerary can be completed in a handicapped accessible manner. Esztergár-Kiss (2019) compares multiple European route planning applications from a traveler perspective. A differentiation is made between different user groups with individual requirements. In addition to the high relevance of integrating a variety of route-based preferences into the search, the

inclusion of handicapped routes especially for elderly people is mentioned. Mandžuka (2021) discusses the challenge of multimodal routing, particularly between different countries. The author highlights that travelers have multiple parameters such as travel time, price, number of transfers, walking distance and waiting time as examples for route-based preferences. Furthermore, accessibility information describing whether the access to the respective mode of transportation is e.g. step free and wheelchair-accessible has to be provided. We abstain from further stop-based information such as safety information and the simplicity to find the right stop into the search in this overview. We envision to embed this information in the future within the proposed framework.

In this paper, we utilize both route and stop-based information to enhance the quality of the Pareto-optimal set of relevant stops and respective itineraries, which can be presented to the traveler and form the choice set for the traveler. In our SBO approach, we use travel time, number of transfers and walking distance as route-based information. For stop-based preferences, we integrate accessibility for disabled and handicapped travelers, the frequency as well as the number of lines. The set of considered preferences can be extended beyond this proof-of-concept study.

3.3 Framework for identifying relevant first and last mile stops

We propose a new framework to identify request-specific stops for the first and the last mile for travelers. As shown in Section 3.2.1, enormous progress has been made in multimodal routing in recent years. As these approaches merely focus on route-based information, they neglect taking information about relevant nearby stops into account. Hence, we integrate both route and stop-based information into the search while forming the choice set for the traveler.

As mentioned in Section 3.2, there is extensive research on the benefits of incorporating unscheduled modes into an itinerary that takes advantage of popular trends in bike-sharing and scooter-sharing. We address how this would look in our algorithm by showing how relevant stops for the first and last mile can change based on the availability of these modes. We model them based on simulated and schedule-based data and see in our experiments how this could affect the traveler’s decision criteria

and the Pareto-optimal set. In this context, we assume that the current location and availability of the unscheduled services are provided in real-time in an integrated mobility platform. This mobility platform also includes the data of the scheduled network. Therefore, we can model the environment as a static network at the time of a traveler request. Aggregating all mobility service data into one platform enables traveler-oriented multimodal transportation planning.

3.3.1 Stop-based methodology

Travelers expect quick identification of relevant nearby stops for their individual itinerary from their specific origin O to their destination D . As shown in Section 3.2, most route planning algorithms merely consider route-based information to enable door-to-door mobility for the traveler. Our approach incorporates stop-based information as additional parameters, and thereby enriches existing route-based information with stop-based information. In the following section, we identify relevant stops for travelers based on their respective requests on an undirected network graph (3.3.1.1), which has been supplemented by stop-based information (3.3.1.2). This sets the framework for discussion of our algorithm for identifying and presenting these stops in Section 3.3.1.3.

3.3.1.1 Network graph

We define a public transportation network of an undirected graph $G = (V, A)$ where V represents all possible stops in the transportation network. The set of edges A represents legs between these stops. Each leg $a \in A$ is defined by a deterministic travel time, either using the existing bus network or a deterministic walking or scooter time.

By running a standard Dijkstra's algorithm (Dijkstra, 1959) on this network optimized by overall travel time, we are able to calculate the following route-based information quickly:

- Overall travel time: This parameter provides information on the travel time to get from O to D . The overall travel time includes the time from origin O to the first transfer stop, the cumulatively summed travel times of all modes

used in public transport, and from the final transfer stop of the itinerary to destination D .

- Overall walking time: This parameter provides information on the required combined walking time for the specific itinerary. Hereby, we assume a predefined walking speed. Walked distances, which occur during the transfer at the same stop, are not taken into account.
- Number of transfers: This parameter provides information on the minimum times the traveler has to transfer from one service to another.

3.3.1.2 Stop-based information

We enrich the discussed route-based parameters with additional stop-based information for each stop $v \in V$ to have a more sophisticated multi-criteria decision-making approach identifying relevant nearby stops for the traveler. This stop-based information can be easily precomputed using the timetable for the respective public transportation network. As additional stop-based parameters, we consider the following:

- Frequency (headway): This parameter provides information on how often a bus is scheduled on average to access a specific stop. This information gives insight into how long a traveler has to wait in case of missing a bus or if a bus fails on short notice. A stop with a smaller frequency in average minutes between bus lines is more ideal for a traveler than a stop with a larger, more infrequent average time between service. Thus for example, a higher frequency of 20 minutes is worse in comparison to a lower frequency of 10 minutes.
- Number of bus lines: This parameter provides information on how many bus lines service a stop. As more bus lines service a stop, the more alternatives the traveler has available. Thus, a higher number of bus lines is advantageous for the traveler in comparison to a lower number of bus lines servicing a bus stop.
- Accessibility: This parameter is a binary variable indicating if a stop is handicap accessible for the traveler. This can be of importance to travelers and can be extended to include sheltered stops or well-lit areas for nighttime travel.

For route-based information as well as for stop-based information an extension with further parameters is possible. For instance, additional route-based information can be the overall waiting time. As additional stop-based information safety information and the simplicity to find the right stop can be integrated in future work.

3.3.1.3 Framework for identification of relevant nearby stops

Based on the network graph and additional stop-based information, we present the framework for identifying a set $S_{traveler}$ of traveler-oriented nearby stops to achieve door-to-door mobility. As several conflicting objectives have to be addressed while identifying this set, we are dealing with a multi-criteria decision-making problem. In general, we aim at minimizing a vector S_O^{Choice} of n objectives such as $\min_{s \in S_O^{Choice}} (f_1(s), f_2(s), \dots, f_n(s))$ (Ehrgott, 2005). S_O^{Choice} describes the set of nearby stops and n the set of considered route and stop-based information. The components $s \in S_O^{Choice}$ are mostly competing against each other.

Algorithm 1 shows the basic components of the framework. Given O and D , we identify a set of stops nearby the origin S_O^{Choice} , which are in walking distance (line 1). For each stop $s \in S_O^{Choice}$, several route and stop-based information are taken into account. The overall travel time s_{dijk} as well as the optimal path from s to D are calculated by solving a standard Dijkstra's algorithm minimizing the overall travel time (line 3) (Dijkstra, 1959). This optimal path contains all information about the itinerary, the departure and arrival time at which stop, and the respective transfers.

Algorithm 1 Stop-based optimization framework

- 1: $S_O^{Choice} \leftarrow \text{IdentificationOfStopsInWalkingDistance}(O, D)$
 - 2: **for** $s \in S_O^{Choice}$ **do**
 - 3: $s_{dijk}, path \leftarrow \text{Dijkstra}(s, D)$
 - 4: $s_{\#transfers}, s_{walkingTime} \leftarrow \text{FurtherRouteBasedInformation}(s, path)$
 - 5: $s_{freq}, s_{\#lines}, s_{accessibility} \leftarrow \text{StopBasedInformation}(s)$
 - 6: **end for**
 - 7: $S_{traveler} \leftarrow \text{RemovalOfDominatedStops}(S_O^{Choice})$
-

The parameters for number of transfers $s_{\#transfers}$ as well as walking time $s_{walkingTime}$ are derived easily in a subsequent step after applying Dijkstra's algorithm using $path$ information retrieved in the preceding step (line 4). The walking time can be calculated by taking into account the individual traveler's origin and destination, the

first and last mile stop of the respective path, as well as the overall walking time at transfer stops.

In the next step, based on available scheduled network data, precomputed information about the frequency s_{freq} , the number of bus lines $s_{\#lines}$ and accessibility information $s_{accessibility}$ are added as additional stop-based information for each stop $s \in S_O^{Choice}$ (line 5). This stop-based data needs to be precomputed based on the public transportation network details to ensure a quick runtime of the algorithm.

Finally, after all parameters for each stop $s \in S_O^{Choice}$ have been quickly calculated, dominated stops are removed (line 7). This results in a set of Pareto-optimal stops $S_{traveler}$, which can then be presented to the traveler with all relevant information. A stop s_1 dominates a stop s_2 if s_1 is superior to s_2 according to at least one parameter and not inferior regarding all other parameters (Delling et al., 2013a). It is worth mentioning that we apply a minimization objective in this multi-criteria decision-making setting. Therefore, $s_{\#lines}$ has to be transformed for a minimization setting before it is considered in any domination rules. Remaining stops build up the Pareto front.

3.4 Experimental results

In this section, we present experimental results applying our framework in a medium-sized public transportation network in the city of Göttingen, Germany. This is a university town with a star-shaped structure with the city center and train station at the center, similar to many other European cities. Göttingen’s urban area covers approximately 11,699 hectares, in which about 134,000 residents live (Stadt Göttingen, 2022). The public transportation network comprises 20 daytime lines, 8 night lines and includes about 500 stops (Göttinger Verkehrsbetriebe GmbH, 2022). Section 3.4.1 outlines the experiments run with our dataset to provide varied results from different areas of the city. We demonstrate in Section 3.4.2 the benefit and effect that considering stop and route-based information simultaneously can have in expanding the traveler’s options. Sections 3.4.3 and 3.4.4 detail the differences that arise when scooters are added to the network.

3.4.1 Design of experiment

To discover the effects that our SBO approach has on public transportation networks, we consider all 18 districts of the city of Göttingen as shown in Figure 3.1. Our experiments run Algorithm 1 from each of the 18 districts to every other district for a total of 306 Origin-Destination combinations. The origin and destination for each experiment is located at the center of the district, and nearby stops (within 0.5 kilometers) are potential relevant stops for the first and last mile.

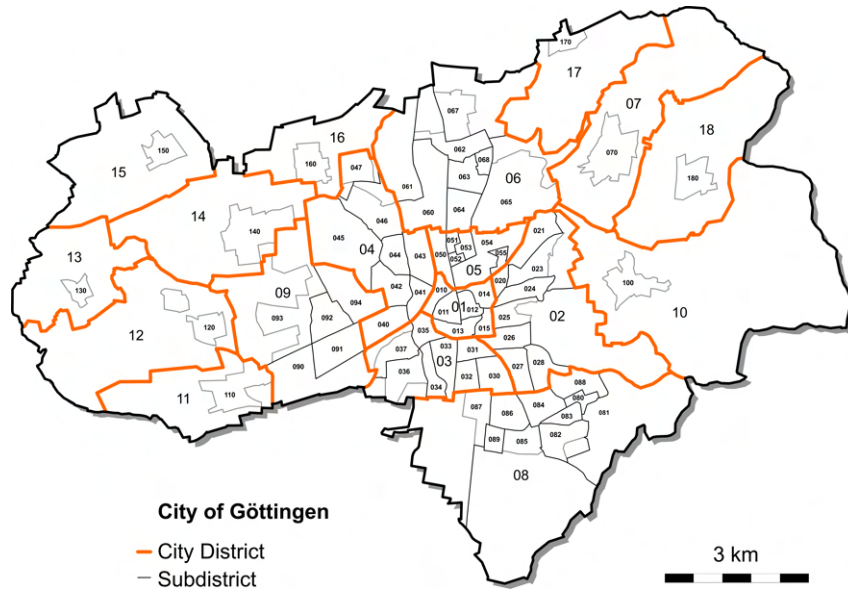


Figure 3.1: Districts for experiments: Adapted by Klatt & Walter (2011)

The bus network is based on the real-world schedule of Göttingen reduced to one day of scheduled operations. We limit the maximum walking distance between two stops to 500 meters, but this could be expanded later to see the effect on experimental results. We assume a walking speed of $5\text{km}/\text{h}$.

Figure 3.2 demonstrates an example output of Algorithm 1 of stops in Göttingen, which are in walking distance. Here, the traveler’s origin is marked in gray. The nearby stops that are dominated are displayed in red and would not be shown to the traveler as these do not offer any added value for the traveler. Each stop on the Pareto front is shown in blue. These are the stops which form the choice set for the traveler. Their characteristics are displayed with bubbles to represent how each

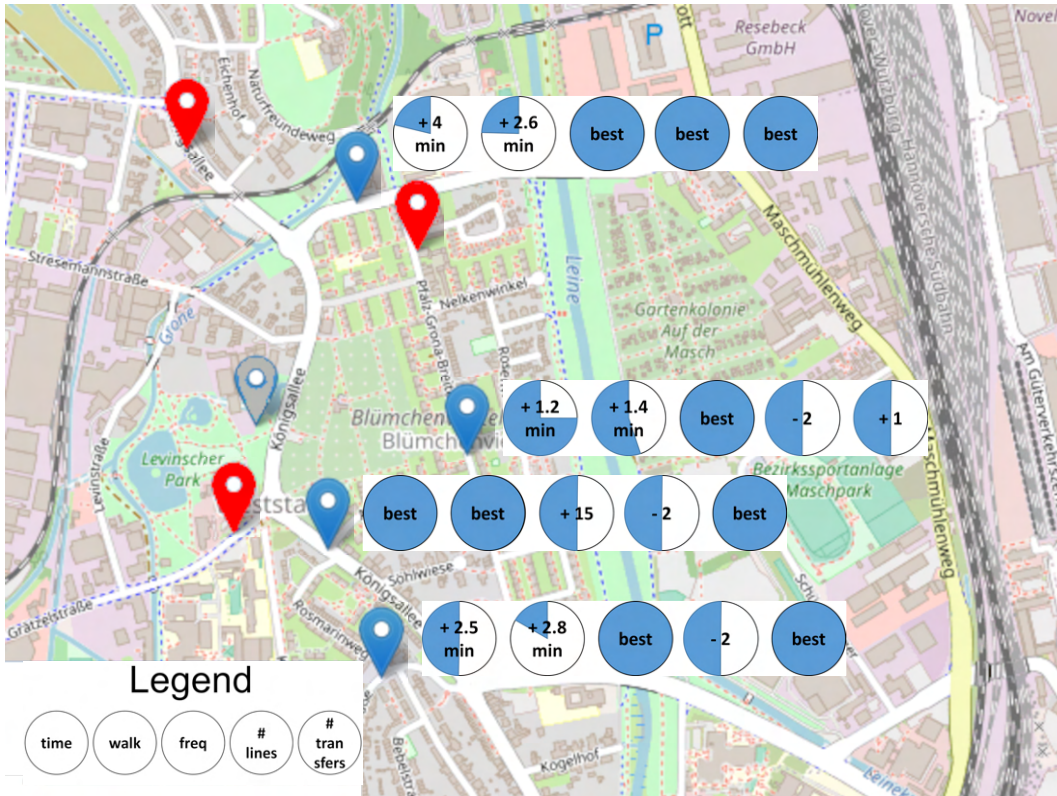


Figure 3.2: Example identification of nearby stops for a traveler

stop compares to others on the Pareto front. For example, the optimal travel time would be displayed as a full circle with “best” while an alternative stop choice may be partially shaded and have +1.2 minutes in the label. This algorithm output gives the traveler a complete picture of the benefits and drawbacks of all nearby stops. Further information such as the underlying itinerary and the actual values for each respective preference can be presented as more in-depth information for each relevant stop. This supports the traveler in their decision-making process.

3.4.2 Stop-relevant results

To investigate the impact that Algorithm 1 had on identifying relevant stops, we performed experiments between Origin-Destination (OD) centroids of each district only considering walking and bus edges for $v \in V$. We found that there were on

average 10 stops within walking distance of both the origin or destination. However, when using Algorithm 1, there were only a quarter (2.4) of these stops on the Pareto front. Additionally, the average travel time between origin and destination was approximately 23 minutes with average headway between buses of 24.5 minutes.

Table 3.2 presents a comparison between relevant stops on the Pareto front to dominated stops. While on average around 2% overall travel time savings and 4% walking time can be seen, relevant stops have a 21% more frequent schedule in comparison to stops not presented to the traveler. Thus, the largest savings for travelers using this method arise in the frequency, number of lines, and number of transfers. These key savings in the frequency, lines, and number of transfers are substantial given that it shows that the Pareto front results can yield savings in areas other than traditional route-based optimization, which focuses on time savings. By expanding the definition of optimal beyond fastest transport service, travelers can experience more frequent public transport options, more lines servicing the stop, and less number of transfers to their destination. This research highlights expansion of the Pareto front to lesser utilized, but important categories that can give the traveler options not displayed by strictly time-optimized techniques.

Table 3.2: Savings potential with respect to different parameters

	Time (min)	Walk (min)	Freq. (min)	Lines	Transfers
Relevant Stops	22.7	6.6	24.5	2.6	1.4
Dominated Stops	23.1	6.9	31.0	2.1	1.5
Savings Potential	2%	4%	21%	20%	12%

Further examining the non-dominated stops yields the closeness to optimality for stops on the Pareto front in each category as shown in Figure 3.3. Here, we can see that 75% of the stops on the Pareto front add an additional 2-3 minutes of overall time and walking time to the traveler’s itinerary. Thus, most stops on the Pareto front reveal first and last mile stops that do not add unreasonable amounts of time to the itinerary.

These results indicate that by evaluating multiple preferences when considering nearby stops, we can identify high quality stops with a number of advantages. The Pareto front stops give much more frequent service and number of lines while displaying options that are usually adding only a few minutes to travel and walking time. This

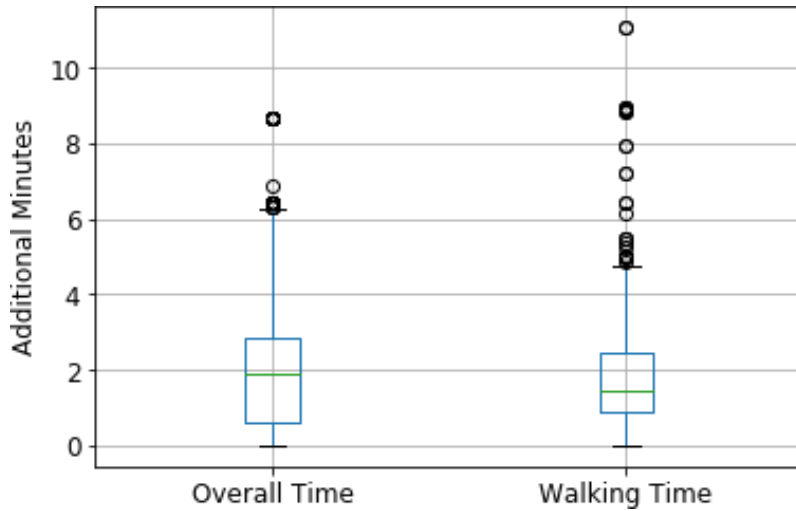


Figure 3.3: Additional time for stops in the Pareto set

approach can help travelers focus on these non-dominated stops and evaluate the preferences that are important in their route planning.

3.4.2.1 Accessibility

One additional example of a preference that could be important to travelers is accessibility for disabled and handicapped travelers (Fatima & Moridpour, 2019). We have included this additional parameter in Table 3.3. Here, when comparing results with and without the binary accessibility parameter, the number of stops on the Pareto front increases on average from 2.4 to 2.9, while the overall travel time decreases by over 80 seconds. Average frequency of arrivals does not change, but both number of lines per stop and transfers per itinerary increase when considering accessible stops. When we examine accessible-only stops, we can see a drastic decrease in the travel time as well as an increase in the number of lines accessed per stop. This type of analysis for parameters that can be of significant importance to certain travelers is essential to provide the traveler with routes and stops that will fit their preferences. We can expand on this with additional parameters or incorporate alternate transportation methods into the model.

Table 3.3: Advantages for accessible stops when implementing new accessibility parameter

	Stops on Pareto Front	Time (min)	Freq. (min)	Lines	Transfers
Without Accessibility	2.4	22.7	24.5	2.6	1.4
Accessible Stops	2.9	21.5	24.4	3.0	1.2
% Improvement	22%	5%	0%	18%	9%

3.4.3 Results from scooter implementation

Following the initial experiments that tracked how stops were chosen based on the parameters, we investigated the effect that incorporating an additional mode of transportation had on the results. Specifically, we focused on how positioning scooter nodes close to bus stops in each region of the city would expand and alter the Pareto-optimal stops shown to the traveler.

To achieve this, in each region we assume there are scooter nodes located near the region center and also scooter edges that connect any two bus nodes within 1.5 kilometers of each other. If the two bus nodes are within 0.5 kilometers of each other, then a walking edge supersedes this scooter edge and is added to the network instead. The results of these added free-floating scooter edges are displayed in Table 3.4.

Table 3.4 demonstrates that by adding scooters as first and last mile modes to the network, the options for travelers are expanded to more than twice that of the original network. While the average time of the shortest path slightly increases, the traveler is presented with stops that have a number of attractive qualities. In addition, slightly less walking is required in case scooters are considered. The stops considered have more frequent service, are serviced by two more bus lines on average, and have less transfers on the traveler’s itinerary. This demonstrates that increasing the range of nearby stops by adding scooters can provide more options that may more closely suit travelers’ preferences.

This benefit is further illustrated in Figure 3.4(a). Here, the average number of transfers as well as the average frequency between buses in seconds is shown for each of the 18 districts. The relationship intuitively indicates an increasing number of minimum transfers as the stop becomes less frequent. It can be seen that considering scooters (blue crosses) yields a lower number of transfers in comparison to merely

Table 3.4: Average differences between scooter and non-scooter experiments

Mode	Stops on Pareto Front	Time (min)	Walk (min)	Freq. (min)	Lines	Transfers
No Scooters	2.4 (29.6%)	22.7	6.6	24.5	2.6	1.4
Scooters	5.4 (16%)	23.2	6.3	18	4.4	0.9

considering buses as an available mobility service (red circles). Figure 3.4(b) compares the average travel time and the average walking time in seconds for scooters against non-scooter integration for each of the 18 districts. The figure shows that a higher travel time results in a higher walking time as well. The required walking time can be reduced by adding scooters into the network as a first and last mile sharing service (blue crosses).

The blue marks and line show that on average implementing scooter access results in the usage of bus stops that have more frequent service as well as less transfers for the traveler.

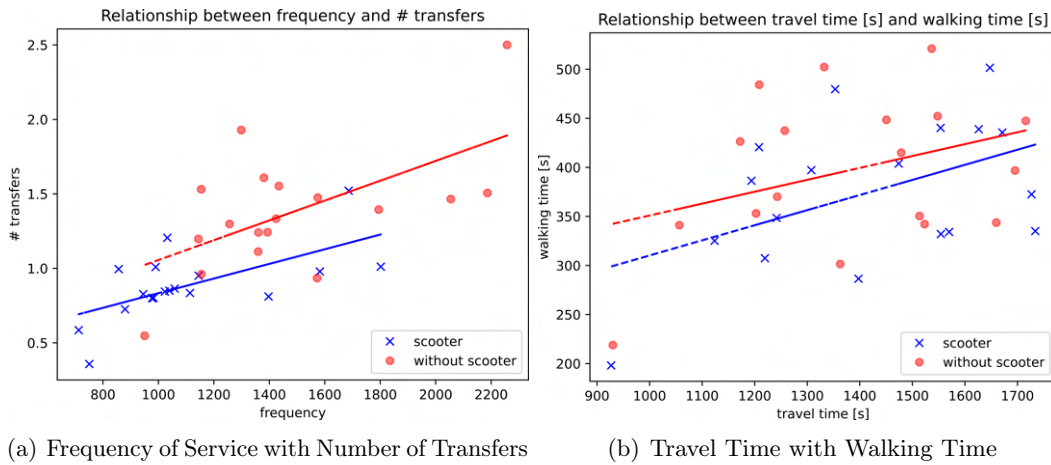


Figure 3.4: Relationship between preferences

Figure 3.5 shows the average percent change of the experiments with scooter integration against the non-scooter experiments by districts for each respective route and stop-based preference. The non-scooter experiments serve as a baseline. Following, districts highlighted in red indicate that, on average, the value of the

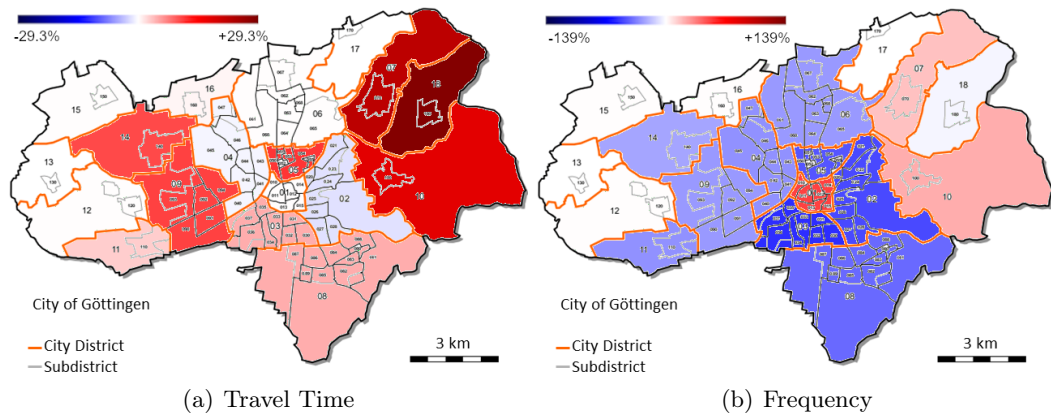


Figure 3.5: Percent change by district

respective preference has worsened in that district. Districts marked in blue indicate that the value of the respective preference has improved, whereas white highlighted districts mean no significant difference in comparison to the non-scooter results. Please note that the percent change indicated by dark blue and dark blue, respectively, is determined by the respective maximum change, and therefore differs for each preference.

Figure 3.5(a) shows the change for the overall travel time. A slightly worse average travel time can be observed in particular for the eastern districts (7, 10 and 18) as well as for the southwest districts. For district 18 (Roringen) the average travel time decreases by 29%. Conversely, if the traveler's origin is in district 2 (Oststadt), minor improvements of approximately 2% with respect to the average overall travel time can be observed. As can be seen in Figure 3.5(b), integrating scooters for the first and last mile enables the traveler to reach additional stops, which have a more traveler-oriented frequency in comparison to stops more accessible by walking to them. A significant deterioration in terms of frequency can be observed in the city center (district 1). It can be assumed that scooter integration also leads to the consideration of less frequented stops outside the city center, which are Pareto-optimal with regard to one of the preferences taken into account.

Further analysis for percent changes by district for the preferences walking time, number of lines and number of transfers can be found in Figure 3.7 in Appendix A.

3.4.4 Visualizing results of comparative scooter implementation

For a specific comparison of the effect of scooter usage by region, we examine Figure 3.6. These two regions are Innenstadt, represented in blue and near the city center, and Weststadt, represented in orange and away from the city center. The solid line represents the average across categories when scooters are utilized. While the average travel time is comparable between the modes, slightly longer runtime is necessary if scooters are considered as an additional service. Additionally, buses arrive more frequently for stops accessed by scooters. The expanded stop options also service more bus lines and require fewer transfers. These averages vary across the regions, but the benefit of including scooters into a multimodal network persists throughout. Incorporating a first or last mile on-demand option, such as scooters, can identify stops with more frequent and varied service and less transfers that can expand the traveler’s information availability and decision making.

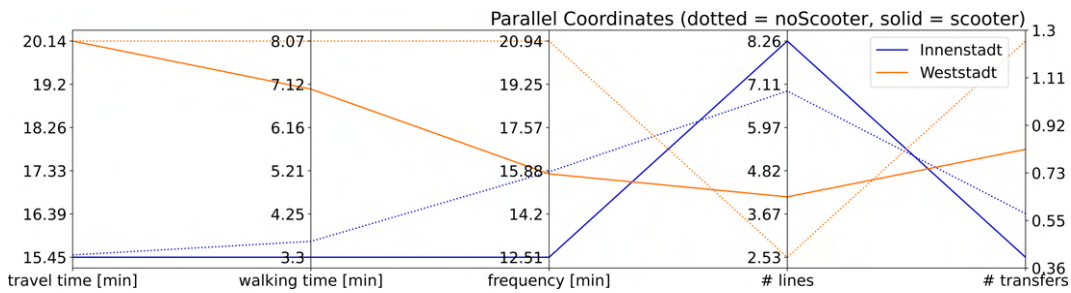


Figure 3.6: Parallel coordinates plot of different Pareto front parameters

3.5 Discussion

While this work contributes to the existing literature through a stop-based optimization that takes into account multiple stop and route-based preferences for the traveler, there are some areas that could expand the reach of this work. For example, additional important parameters, such as safety of a stop’s area and ease of access to other public transport modes could be included. In addition to not including these parameters that may be important, this work does not explore the potential large amount of options that may show up on the Pareto front for large-scale country or regional multimodal networks.

For multimodal networks, travelers expect dynamic, updated results to know the availability and updated schedule of travel options. In reality, an additional dynamic map - updated each minute or more frequently - could be integrated to refresh the scooter nodes and availability for the traveler. This would give even more additional stops on the Pareto front for the traveler and knowledge if scooter nodes would be a viable option to begin or end their itinerary.

In future work, further stop and route-based information can be integrated into the search. For instance, the price can be integrated as an additional route-based preference. As further stop-based information, we can envision, for example, the integration of safety information and the simplicity to find the right stop into the search. The integration of further preferences lead to a larger Pareto front. Thus, techniques to limit the set of Pareto-optimal stops for the first and last mile of the traveler's itinerary are required. One way would be to enable the traveler to set weights for individual preferences in an expert menu within a mobility platform to allow travelers to prioritise their preferences individually. This should be aided by advanced visualization techniques to support the traveler in their decision-making process. In addition, further experiments applying the proposed SBO approach to more cities with different characteristics ensures generalization of the results.

In real-world settings, travelers want to use one multimodal application that integrates all available mobility services. Our framework can easily be adapted to an extended multimodal setting. The integration of additional mobility services can increase the number of relevant nearby stops, which is an exciting potential for future research.

To adequately assist the traveler in selecting the most appropriate nearby stop, a simple presentation of the Pareto-optimal options is necessary. In this work, we have focused on the technical perspective of identifying the set of Pareto-optimal nearby stops and presented a first approach to present this information to the traveler. Further work can additionally present this information into an integrated multimodal routing application in a traveler-oriented way.

Travel policy implications of this traveler-centered approach include an analysis of stop location to see if certain stops should be included or excluded for traveler convenience or lack of use. Additionally, timetable policies can use this Pareto front analysis to see if certain stops should be frequented more or less. With shared mobility

policies in a city, the placement and replacement of shared bikes, scooters, and other transportation modes could utilize this tool to maximize demand for the service along with existing public transport networks.

3.6 Conclusion

In recent years, large strides have been made in creating multimodal door-to-door itineraries. However, significant challenges remain while identifying these options in a traveler-oriented way. Travelers expect information about relevant first and last mile stops and their characteristics in a transparent way using up-to-date mobility applications. In this work, we combine stop and route-based information in the decision-making process. In particular, we consider the overall travel time, the overall walking time as well as the number of transfers as route-based preferences, and frequency, accessibility and number of bus lines as stop-based information into the search. The set of relevant nearby stops considering this information can then be presented to the traveler. This enables travelers to make better-informed decisions.

The proposed framework for identifying alternative stops for first and last mile urban travel planning has been evaluated using a medium-sized public transportation network of Göttingen, Germany. In addition to the public transport bus network based on real-world data, we integrate unscheduled mobility services such as electric scooters.

We show that the traveler has several non-dominated nearby stops with different characteristics available to choose from. Stops on the Pareto front have on average more public transport lines and more frequent service than dominated stops. Furthermore, the traveler saves both travel and walking time. This trend is also true for incorporation of scooter nodes that expand the traveler's nearby stop options. In addition, we have introduced a novel idea on how to present the Pareto-optimal set of nearby stops to the traveler. We envision this framework of identifying relevant nearby stops being implemented in the future as the demand for integrated multimodal transportation information increases. Providing this information to the traveler allows for better decision making while planning individual multimodal itineraries.

Appendix

A Further percentual changes by districts

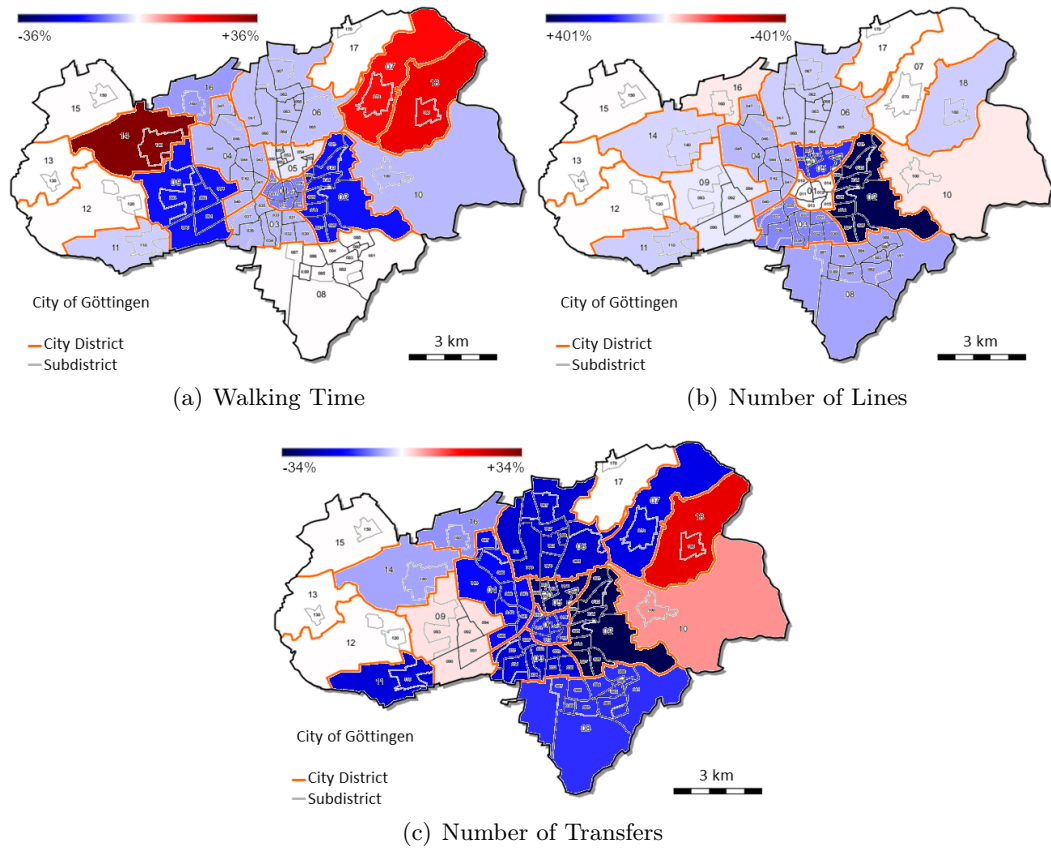


Figure 3.7: Percentual change by district

Chapter 4

Traveler-oriented multi-criteria decision support for multimodal itineraries

Abstract

In recent years, the variety of mobility services has increased strongly. Travelers expect a diverse set of combined mobility services according to multiple individual preferences. Due to the competing characteristics of these preferences (e.g. travel time, price, and the number of transfers), several Pareto-optimal itineraries representing trade-offs arise. While there are efficient approaches for finding multimodal shortest paths, the full set of Pareto-optimal itineraries cannot be determined efficiently when multiple traveler preferences are considered in a large multimodal network. However, this would be required to provide travelers with more relevant choices in the light of available options and complex solution space characteristics.

In this work, we propose smart ways to approximate the Pareto front of multimodal itineraries efficiently. The core idea is to apply solution space sampling systematically. We focus on the scalability of the sampling framework with respect to multiple traveler preferences as well as identifying interesting characteristics of itineraries to enable travelers to take well-informed decisions. The framework is evaluated with a large amount of real-world data of mobility services. To this end, we analyze long-distance trips between major cities in Germany, taking up to five most prevalent traveler preferences into account. In addition, we examine the Pareto-optimal solutions and derive characteristics of potential interest for the traveler that can help to make the search more transparent and explainable and thus shape the traveler's choice set.

Keywords Routing, Multi-Criteria Decision Support, Multimodal Mobility, Pareto Front Approximation

Contents

4.1	Introduction	81
4.2	Related literature	83
4.2.1	Traveler-oriented multimodal travel planning	83
4.2.2	MCDM methods	84
4.2.3	Multimodal routing algorithms	87
4.3	Solution space sampling	88
4.3.1	Problem description	89
4.3.2	Framework	90
4.3.3	Approximation of three-dimensional sets of solutions	95
4.3.4	Multimodal network model	96
4.3.5	Resource-constrained multimodal routing	99
4.3.6	Complexity analysis	101
4.3.7	Speed ups	102
4.4	Computational design	105
4.4.1	Experimental setup	105
4.4.2	Metrics	107
4.5	Results	110
4.5.1	Summary results	111
4.5.2	Technical results	111
4.5.3	Traveler perspective	118
4.6	Conclusion	122
	Appendices	125
A	Approximation of three-dimensional sets of multimodal solutions	125
B	Approximation of two-dimensional sets of multimodal solutions with speed-up techniques	127
C	Approximation of three-dimensional sets of multimodal solutions with speed-up techniques	127
D	Mode choice options	129
E	Performance metrics	130
F	Aggregated relationships for different traveler preferences	132

4.1 Introduction

In recent years, innovative mobility services such as car, bike and ridesharing services have emerged. These services contribute to the set of itineraries available for personal mobility planning. The combination of several mobility services is known as multimodal mobility (Lyons et al., 2020; Willing et al., 2017). Integrated mobility platforms such as *Rome2rio*, *fromAtoB* and *omio* promise travelers to create door-to-door itineraries considering their preferences as well as a comprehensive quantity of these services (Esztergár-Kiss et al., 2020; Molenbruch et al., 2021).

There are significant challenges when creating a set of multimodal itineraries for travelers to choose from. Given new business models and big data handling, travelers expect a reasonable choice of door-to-door itineraries considering their individual preferences (Schulz et al., 2020). Many studies have revealed that besides travel time, price, and number of transfers, which are considered by nearly all mobility platforms, also additional preferences are of importance such as mode choice, walking distance, waiting time, reliability, and sustainability (Abdullahi et al., 2021; Alt et al., 2019; Esztergár-Kiss & Csiszár, 2015; Grotenhuis et al., 2007). For the most part, the simultaneous consideration of several traveler preferences is required to shape a traveler’s choice set (Esztergár-Kiss et al., 2020; Lyons et al., 2020). Furthermore, travelers want to be well-informed about the importance of and trade-off between their preferences. For example, travelers may be willing to invest a little more travel time in order to achieve a significantly lower price. This results in a set of relevant Pareto-optimal itineraries for the traveler to choose from.

While finding optimal paths given a set of constraints has been investigated thoroughly in the area of Multi-Criteria Decision Making (MCDM), finding the full Pareto-optimal set of itineraries with multiple traveler preferences in a multimodal setting remains a significant challenge. Delling et al. (2013a) and Dib et al. (2017) highlight that taking more than three traveler preferences simultaneously into account has an enormous impact on the run time of finding the full Pareto-optimal set. Integrating multiple mobility services in a large area network increases the complexity further. As run time is of high importance for mobility platforms, a multimodal search algorithm should scale efficiently with respect to the number of considered traveler preferences. Furthermore, the search should provide a diverse set of itineraries that best supports the traveler in decision making (Bast et al., 2015).

In addition to acceptable run times, communicating search results and adapting them to travelers' preferences is needed. We address situations where travelers either have only a vague idea of their preferences and/or the complexity of the multimodal solution space is high and its characteristics are hence not transparent to an individual traveler ("black-box"). The high complexity results from the impact of numerous traveler preferences and service features. Consequently, it is hard for a traveler to assess the choice set as well as the impact of individual preferences on further refinement of this set. Giving the traveler more insight into the diverse set of itineraries and the impact of preferences on this set would make the search more transparent and enable travelers to adjust their preferences in light of available options and solution space characteristics (Alt et al., 2019; Ehmke & Horstmannshoff, 2020).

In this work, we propose smart ways to approximate the Pareto front of multimodal itineraries efficiently. Based on general information given by the traveler about the origin, destination, earliest departure time as well as further individually relevant preferences, we approximate a set of Pareto-optimal itineraries. We assume that the traveler initially formulates the relevant preferences. We do not make any further basic assumptions about these preferences, e.g. whether a longer travel time is preferred to a longer waiting time or not. The resulting choice set can be presented to the traveler including additional information about the set's characteristics, which enables the traveler to make well-informed decisions. We use solution space sampling to approximate the Pareto front of multimodal itineraries efficiently. We especially focus on the scalability of our framework with respect to multiple traveler preferences in a large multimodal network. Furthermore, we present ideas to make the multimodal search space more transparent and explainable to travelers to help them identify and set decisive preferences. Consequently, in the long run, our framework should provide input for a scalable multimodal mobility platform and help travelers understand and form their choice sets through additional information about the complex multimodal solution space. These characteristics about the choice set can be embedded as an expert menu in mobility platforms, for example. As we focus on the technical capabilities of multimodal network search, we do not model the choice behavior of travelers in this paper.

Our approach is evaluated through a proof-of-concept study using a large amount of real-world data of mobility services. In particular, we analyze long-distance trips between major cities in Germany, taking up to five preferences into account. We identify characteristics of resulting itineraries to support the traveler in multimodal decision making. With our experiments, we demonstrate the efficient scaling of the framework for a larger number of traveler preferences. Integrating further preferences into the search results in additional Pareto-optimal itineraries that could not be found before: with five preferences in parallel, we can derive one more relevant itinerary on average compared to considering only two preferences.

An overview of related work on multimodal route planning is given in Section 4.2. Then, in Section 4.3, we formally introduce our solution space sampling framework and how it ensures scalability with respect to the number of considered traveler preferences. Section 4.4 introduces the considered data on mobility services, long-distance trips, and traveler preferences, the framework’s settings as well as relevant metrics for evaluation. In Section 4.5, we analyze results both from a technical point of view and from a traveler perspective. Finally, we give a brief summary and identify areas for further research in Section 4.6.

4.2 Related literature

MCDM is a well-established research field with a large number of publications and practical relevance. In Section 4.2.1, we highlight related research on traveler requirements to online multimodal routing platforms. Then, we present a classification of MCDM methods regarding the interaction with the decision-maker as well as an introduction into MCDM methods (Section 4.2.2). Numerous studies have proposed multimodal routing algorithms in recent years; in Section 4.2.3, a relevant subset of these is introduced.

4.2.1 Traveler-oriented multimodal travel planning

Many studies have emphasized that traveler orientation is a key factor for the acceptance of multimodal mobility. Schulz et al. (2020) and Valderas et al. (2020) argue that all mobility services have to be accessible through an integrated mobility platform. Travelers expect that sufficient private and public mobility services are

available to achieve a one-stop search for door-to-door mobility. This includes innovative services such as bike and ride-sharing for the first and last mile as well as long-distance services like going by train and flying (Lyons et al., 2020; Stopka, 2014).

In addition to comprehensiveness, numerous studies have found that travelers expect a diverse set of alternative itineraries, i.e. travelers want to make an independent choice from different mobility options (Esztergár-Kiss & Csiszár, 2015; Lyons et al., 2020; Stopka, 2014). Grotenhuis et al. (2007) highlight the importance of presented alternatives to achieve door-to-door mobility for the traveler. This enables the traveler to compare various mobility options with additional information about the respective alternatives. Ulloa et al. (2018) argue that diversified alternatives enable travelers to adapt their travel planning according to their individual preferences and context.

Alt et al. (2019) and Ehmke & Horstmannshoff (2020) emphasize that most platforms that combine several services such as mobility services follow a “black-box” paradigm. This limits the possibilities of travelers to assess and filter the presented choice set. The authors identify requirements towards empowering travelers in their decision-making process by giving them transparent information about the offered services as well as solution space characteristics. This reduces the time and cognitive effort required for service selection and makes the use of mobility platforms more traveler-oriented (Grotenhuis et al., 2007; Lyons et al., 2020).

In this paper, we incorporate the most prevalent traveler preferences to create a diverse set of alternative door-to-door itineraries in efficient run time. Furthermore, we make solution space characteristics more transparent to the traveler by presenting insights into the complex multimodal solution space.

4.2.2 MCDM methods

As many individual traveler preferences have to be considered simultaneously for multimodal itinerary planning, we are dealing with multi-criteria decision support and methods of multi-criteria decision making (MCDM). MCDM aims at minimizing a vector X of n objectives such as $\min_{\vec{x} \in X} (f_1(\vec{x}), f_2(\vec{x}), \dots, f_n(\vec{x}))$ (Ehrgott, 2005). In the context of multimodal routing, X describes the set of feasible itineraries and n the set of traveler preferences. The components $\vec{x} \in X$ are mostly competing against each other. For example, using a fast train implies a high price of travel, whereas a

slow train is much cheaper. Hence, there is no unique solution to this problem, but rather a set of Pareto-optimal solutions.

Hwang & Masud (1979) and Miettinen (2013) survey MCDM methods with respect to interaction with the decision-maker (who is the traveler in our case). The following method classes are investigated:

In *a priori methods*, the decision-maker sets preferences, and the underlying search algorithm finds the closest solution according to the respective preferences. As common for complex problem settings such as multimodal routing, it is hard for the decision-maker to express proper preferences without knowing the problem well. As the impact of the individual preferences on the multimodal solution space is not transparent for the decision-maker (esp. for unknown itineraries) due to the complex characteristics of the integrated mobility services, expectations may be too optimistic or pessimistic (Alt et al., 2019). For instance, the traveler could set a filter to constrain the search to five hours of maximum travel time. As a result, an itinerary that would result in significant price savings but takes a bit longer would not be presented to the traveler.

Interactive methods include an iterative search process, where the solution pattern is formed iteratively. First, the decision-maker gives some high-level information about his or her preferences. Based on these, the search algorithm identifies valid solutions, which are then presented to the decision-maker again, who refines the preferences. This procedure is applied for some iterations.

In *a posteriori methods*, based on basic information given by the decision-maker, a set of Pareto-optimal solutions is created and enriched with additional information about the diverse set of Pareto-optimal solutions. However, this calculation can be computationally expensive as many solutions have to be examined. Furthermore, it is difficult to represent the Pareto-optimal set if more than two dimensions are considered simultaneously. As multimodal routing platforms often imply a black-box paradigm (Alt et al., 2019), a posteriori methods are well applicable in this context. Based on fundamental information given by the traveler such as origin, destination, mode choice, relevant preferences, and desired departure time, we aim at identifying a set of Pareto-optimal door-to-door itineraries to help shape the traveler's choice set.

A variety of MCDM methods has been proposed (Ehrgott, 2005; Hillier et al., 2002; Yannis et al., 2020). Yannis et al. (2020) give a comprehensive overview of these

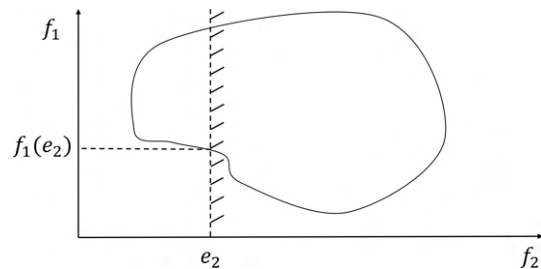
methods for the transportation and mobility sector. In general, MCDM methods can be divided into methods for which a number of available solutions is known a priori, and methods for which no detailed information about the solutions is available at the beginning of the decision-making process (a posteriori). The latter can easily create a large number of alternative solutions (Yannis et al., 2020). Typical solution approaches for such problems include additive weighting, analytical hierarchy process (AHP) (Saaty, R.W., 1987), elimination and choice translating reality (ELECTRE), and preference ranking organization method for enriching evaluation (PROMETHEE) (Omman, 2004).

The ϵ -constraint method is a well-known solution technique for MCDM and applied in this paper (Deb, 2011; Ehrgott, 2005). Hereby, one particular preference is considered as a single objective, and all other preferences are transformed into constraints. According to Mavrotas (2009), the generalized MCDM formulation above can be transformed as follows applying ϵ -constraints:

$$\begin{aligned}
 \min \quad & f_1(\vec{x}) \\
 \text{s.t.} \quad & f_2(\vec{x}) \leq e_2 \\
 & \dots \\
 & f_n(\vec{x}) \leq e_n \\
 & \vec{x} \in X.
 \end{aligned} \tag{4.1}$$

Then, by systematically altering the right-hand side of the constraints (e_i), several feasible solutions are obtained. Figure 4.1 shows a two-dimensional representation of the ϵ -constraint method for a problem setting with two objectives. While objective f_1 is set as a single objective, objective f_2 is transformed into a constraint with its right-hand side constraint set to e_2 . The circle indicates the feasible solution space.

Our solution space sampling framework is inspired by the presented ϵ -constraint method (Mavrotas, 2009), which is assigned to the class of a posteriori methods. In particular, we tackle the issue of its increased run time for problem settings with more than two considered preferences in the objective function by approximating multiple two and three-dimensional sets of solutions simultaneously. Details can be found in Section 4.3.

Figure 4.1: Representation of ϵ -constraint method

4.2.3 Multimodal routing algorithms

Many studies have proposed network search algorithms to identify multimodal door-to-door itineraries. Multimodal routing integrates both route planning in scheduled-based services and unrestricted services. Bast et al. (2015) present an overview of fundamental algorithms for these services and how to combine them into a multimodal setting. Multimodal route planning can be structured by graph-based and timetable-based approaches. While the first extend the well-known shortest path algorithm to a multimodal setting, the latter operate directly on the timetable of the integrated mobility services. In the following, we provide an overview of relevant multimodal routing algorithms. We focus on the capability of the algorithms to include a variety of traveler preferences.

Relevant for timetable-based approaches are e.g. the Connection Scan Algorithm (Dibbelt et al., 2018), Trip-Based Routing (Witt, 2015), UCCH (Dibbelt et al., 2015), and RAPTOR (Delling et al., 2012). Bast et al. (2013) and Delling et al. (2013a) optimize public transit routes using dynamic programming operating on timetables and iteratively increasing the number of possible transfers when calculating all relevant itineraries. They extend the RAPTOR algorithm by Delling et al. (2012) to a multimodal setting (McRAPTOR). They produce a Pareto-optimal set while limiting the computational time considering travel time, traveler convenience, and costs as preferences. They increase the performance of their algorithm by preprocessing the network graph using Contraction Hierarchies (CHs). However, the run time of their approach deteriorates significantly when further preferences are taken into account. Recently, ULTRA (UnLimited TRAnsfers) has been introduced to compute one-to-one-queries in multimodal networks (Baum et al., 2019). ULTRA resolves some limitations of Trip-Based Routing and McRAPTOR and is hence capable

of computing all Pareto-optimal itineraries considering travel time as well as the number of transfers. However, scaling issues remain when simultaneously integrating numerous individual preferences (Sauer et al., 2020).

Examples for graph-based approaches include Giannakopoulou et al. (2018), Kirchler (2013) and Dib et al. (2017). Dib et al. (2017) introduce a multimodal transport network model and a multi-label algorithm considering travel time, number of transfers, and total walking time as traveler preferences. While identifying the full Pareto-optimal set of multimodal itineraries, they encounter the same scaling issues already described by Delling et al. (2013a) and Bast et al. (2013) when taking into account additional preferences.

A major challenge in multimodal route planning is to avoid unrealistic sequences of mobility services. For instance, taking a car between two train rides would be an unfeasible option for the traveler. Barrett et al. (2008), Barrett et al. (2000) tackle this problem with a generalization of the shortest-path problem. Their label-constrained multimodal shortest-path algorithm uses regular languages to constrain the shortest-path problem, which can be modeled with non-deterministic finite automata (Pajor, 2009). A detailed introduction into finite automata as well as how they can be applied to a given multimodal network is shown in Section 4.3.4. In addition, Pajor (2009) presents a comprehensible theoretical introduction into languages, regular languages and finite automata. Using regular languages enables the consideration of travelers mode choice and ensures realistic itineraries (Barrett et al., 2008; Pajor, 2009). Barrett et al. (2000) show that solving the shortest-path problem using regular languages to optimality can be achieved in deterministic polynomial time.

In our solution space sampling framework, we will systematically apply the generalization of the shortest path problem with regular languages proposed (Barrett et al., 2008; Barrett et al., 2000) to identify a set of multimodal itineraries. While we approximate the set of Pareto-optimal solutions, we ensure high scalability with respect to multiple traveler preferences in efficient run time.

4.3 Solution space sampling

In the following, we present an overview of our solution space sampling framework. The overall goal is to approximate the Pareto front as efficiently as possible such that travelers can make well-informed decisions when searching for multimodal

itineraries. To this end, we take multiple traveler preferences as well as a variety of mobility services into account. We will refer to a multimodal itinerary as a solution. Based on the problem description (Section 4.3.1), we present the framework for the approximation of two-dimensional sets of solutions in Section 4.3.2. We extend this to three dimensions in Section 4.3.3 to identify additional solutions, which cannot be found in a two-dimensional setting. Next, we introduce the multimodal network model (Section 4.3.4) and the applied multimodal routing algorithm (Section 4.3.5). The scalability of the proposed framework is ensured independent of the used multimodal routing algorithm as well as the underlying network. In Section 4.3.6, we analyze the complexity of the solution space sampling framework and conclude with speed-up techniques in Section 4.3.7.

4.3.1 Problem description

For multimodal itinerary planning, we require the specific traveler origin O , the destination D , the earliest departure time t_{dep} , and a set of traveler preferences P as input parameters. The set $P := \{P_1, \dots, P_p\}$ consists of p traveler preferences, e.g. travel time, price and number of transfers. Knowledge of these parameters is a common assumption for a posteriori methods. Additionally, we consider a set of mobility services MS . The set of mobility services $MS := \{MS_1, \dots, MS_{ms}\}$ includes ms mobility services such as railway, long-distance bus, and local transit. Details of the solution space sampling framework will be presented in Section 4.3.2. The traveler can restrict the considered mobility services, e.g. exclude flying. Such restrictions are modeled by finite automata fa . MS and fa are integrated into the multimodal network model, which is described in detail in Section 4.3.4.

Given this information, we approximate the Pareto front by generating a set of Pareto-optimal multimodal solutions $S_{traveler}$ according to traveler preferences P by systematically rerunning a multimodal routing algorithm SPM (introduced in Section 4.3.5). Each run of the SPM returns an optimal solution Sol , which contains information about the solution as well as its preference values. Hereby, Sol_i represents the solution value for traveler preference $i \in P$. The set $S_{traveler}$ is composed of different Pareto-optimal solutions Sol . The traveler can individually select from this set of multimodal solutions. While each run of SPM is solved to optimality (Barrett

et al., 2000; Pajor, 2009), we cannot ensure that the set $S_{traveler}$ builds the true Pareto front. Following, we approximate the Pareto front in this set.

We create several sets of solutions in the sampling framework for two and three dimensions. Here, the term dimension refers to how many traveler preferences are taken into account simultaneously when creating the respective sets. For instance, two-dimensional sampling means that we consider two preferences at the same time when approximating the Pareto front in the respective set, whereas in three-dimensional sampling, we consider three preferences at the same time. Details can be found in Section 4.3.2.2. The number of considered preferences in set P is independent of this. We will repeat sampling pairwise (for two-dimensional sampling) or threefold (for three-dimensional sampling), respectively, to approximate sets of solutions.

4.3.2 Framework

We begin with the presentation of our solution space sampling framework approximating the Pareto front for a two-dimensional setting. Algorithm 2 shows the basic components of the framework, and Figure 4.2 depicts each step in case three preferences $P := \{\text{travel time, price, number of transfers}\}$ are considered. Each of these four components is discussed in detail in the following subsections.

Algorithm 2 Solution space sampling framework algorithm

- 1: $I \leftarrow \text{IdentificationOfMinMaxIntervals}(O, D, t_{dep}, P)$ (details in 4.3.2.1)
 - 2: $S_{all} \leftarrow \text{SystematicSamplingWithinIntervals}(O, D, t_{dep}, I, P, k)$ (details in 4.3.2.2)
 - 3: $S_{all}^{opt} \leftarrow \text{RemovalOfDominatedSolutions}(S_{all})$ (details in 4.3.2.3)
 - 4: $S_{traveler} \leftarrow \text{ReductionOfSolutionSet}(S_{all}^{opt})$
-

First, based on given O , D , t_{dep} and P , we identify a min-max-interval $[l_i, u_i]$ for each considered preference $i \in P$ resulting in a set of intervals I . This ensures that only relevant solutions are investigated, and that the following sampling step is efficient. Figure 4.2(a) shows an example for the set of intervals I .

In the next step, we systematically sample the multimodal solution space by rerunning SPM multiple times to approximate the Pareto front. As highlighted by Delling et al. (2013a) and Dib et al. (2017), not all possible solutions can be found in appropriate run time if more than three preferences are considered simultaneously. Therefore, in this step, we limit the search to two traveler preferences $i, j | i \neq j \in P$ at

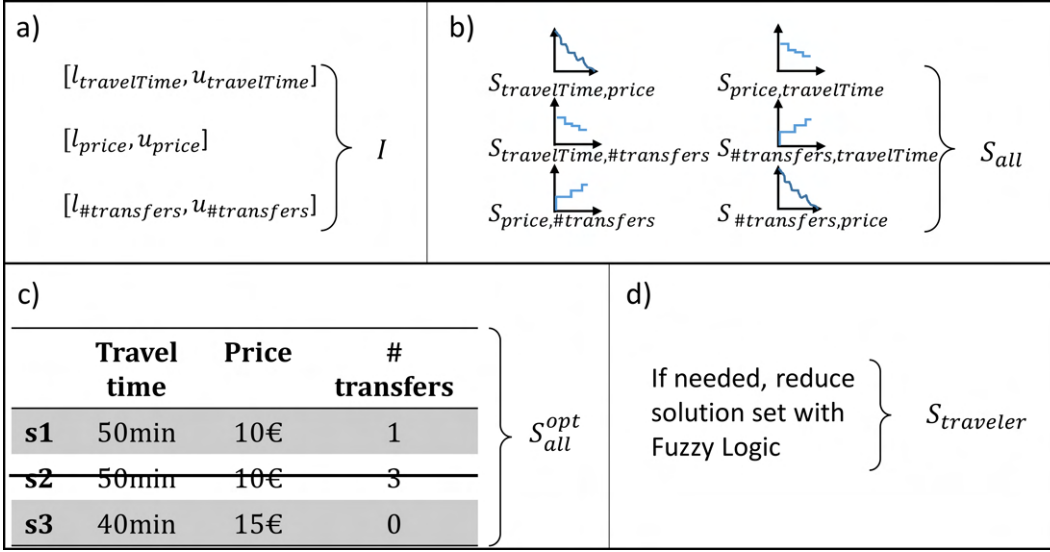


Figure 4.2: Solution space sampling framework

the same time, which results in $|P| * (|P| - 1)$ sets of solutions $S_{i,j}$. The computation of the respective sets can be parallelized, thereby limiting the run time with respect to the number of considered preferences $|P|$. This ensures scalability even for a larger number of preferences, as only the number of the parallelly determined sets increases, but not the complexity for the calculation of these two-dimensional sets. The two-dimensional sets $S_{i,j}$ are merged together in a joint set $S_{all} := \bigcup_{i,j|i \neq j \in P} S_{i,j}$. Figure 4.2(b) shows the six build two-dimensional sets.

Third, dominated solutions are removed from S_{all} as they are irrelevant to the traveler (Deb, 2011). The remaining solutions form the set of Pareto-optimal solutions S_{all}^{opt} (Figure 4.2(c)).

Finally, since the number of Pareto-optimal solutions can become very large with increasing number of simultaneously considered preferences, S_{all}^{opt} may need to be reduced to create an informative choice set $S_{traveler}$ which is finally presented to the traveler (Figure 4.2(d)). Reducing the choice set by leaving out obviously unattractive choices can help reduce the time and cognitive effort required for itinerary selection (Grotenhuis et al., 2007; Lyons et al., 2020; Ulloa et al., 2018). For instance, fuzzy logic approaches as discussed by (Delling et al., 2013a) and concepts of Types aNd Thresholds (Bast et al., 2013) can be implemented in case of very large choice sets.

4.3.2.1 Identification of min-max-intervals for each preference

Algorithm 3 Identification of min-max-intervals

```

1: function IDENTIFICATIONOFMINMAXINTERVALS( $O, D, t_{dep}, P$ )
2:   for  $i \in P$  do
3:      $Sol \leftarrow SPM(O, D, t_{dep}, i, C = \{\emptyset\})$   $\triangleright i$  as objective
4:      $l_i = Sol_i$ 
5:     for  $j \in P$  do
6:       if  $u_j < Sol_j$  then
7:          $u_j = Sol_j$ 
8:       end if
9:     end for
10:  end for
11:  return  $I = \{[l_i, u_i] \mid i \in P\}$ 
12: end function

```

In the first step, we identify min-max-intervals for each traveler preference: $I = \{[l_i, u_i] \mid i \in P\}$. As shown in Algorithm 3, given O, D, t_{dep} and P , a multimodal routing algorithm SPM is run for every traveler preference $i \in P$ setting i as the single objective. The lower bound l_i for the preference $i \in P$ is set as the respective solution value Sol_i of running SPM with objective i (line 4) without considering any constraints C . The upper bound u_i for the preference $i \in P$ is set as the maximum value for the respective preference of all single-objective optimization runs over all preferences (line 5 to line 9).

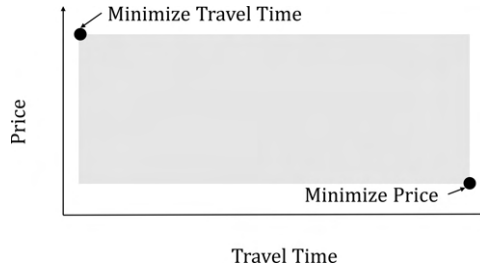


Figure 4.3: Identification of min-max-intervals for two preferences

An example is visualized in Figure 4.3, where SPM is run twice, setting travel time and price as the respective single objectives. The resulting extreme solutions are used to define the respective min-max-intervals and thereby span a two-dimensional

solution space (grey area). Note that travel time and price is only an example of a preference combination for two-dimensional sampling. For three-dimensional sampling, the degree of dimensionality would increase by one.

4.3.2.2 Approximation of two-dimensional solution sets

In the next step, we systematically sample the multimodal solution space within identified intervals I to approximate the Pareto front. Following the idea of the ϵ -constraint method (Deb, 2011), we convert the MCDM problem into a single-objective optimization problem by representing one of the preferences as a single objective and the rest of the preferences as constraints resulting in several sets $S_{i,j}$ for each preference $i, j \in P$. Preference i is set as the single objective, preference j is systematically altered and set as an upper-bound constraint $restrValue_j$. All other preferences $l \in P \mid l \neq i, j$ are set as upper-bound constraints according to their respective upper bound u_l as shown in Equation 4.2. The set $S_{i,j}$ contains several feasible solutions \vec{x} which are mostly competing against each other.

$$\begin{aligned}
 & \min f_i(\vec{x}) \\
 & \text{s.t. } f_j(\vec{x}) \leq restrValue_j \\
 & \quad f_l(\vec{x}) \leq u_l \quad \forall l \mid l \neq i, j \\
 & \quad \vec{x} \in S_{i,j}.
 \end{aligned} \tag{4.2}$$

While each run of the multimodal routing algorithm SPM returns an optimal solution (Pajor, 2009), we cannot ensure identifying all Pareto-optimal solutions (the true Pareto set). Therefore, we approximate the Pareto front for a two-dimensional setting. The approximation is due to the reduction of the search to two dimensions when creating the respective sets as well as limiting the number of set constraint values based on the preset sampling density.

Figure 4.4 demonstrates this for the example of price and travel time $S_{i,j} := \{S_{price,travelTime}\}$, with the price set as the objective i and travel time as the systematically altered preference j . For approximating the Pareto front, we divide the min-max-interval $[i_{travelTime}, u_{travelTime}]$ into k equally sized buckets. The predefined sampling density value k controls the granularity of the sampling. Then, we obtain

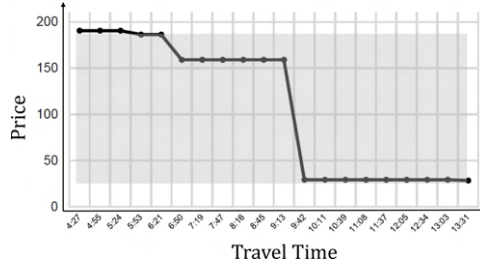


Figure 4.4: Example for two-dimensional systematic sampling

several solutions (each with price as a single objective) considering the respective upper-bound constraint for each bucket. By softening the upper-bound constraint for travel time iteratively and rerunning *SPM* subject to the respective constraint, we can identify many solutions with different travel times which are all optimal with respect to the price objective under the given maximum travel time constraint.

Algorithm 4 defines the process of approximating two-dimensional sets of multimodal solutions more formally. For each set $S_{i,j}$ (line 2), the traveler preference i is set as the objective, and the preference j serves as an upper-bound constraint for *SPM*. The value of j is systematically altered according to the predefined sampling density value k . By slightly alternating the constraint, several optimal solutions for this two-dimensional set can be identified. The increasing rate for the respective constraint j is denoted as $j_{IncreasingRate}$. Whether we are able to sample k times depends on the respective preference data type. For preferences represented by integers (e.g. number of transfers), we alter the respective constraint value by 1 (line 4). For preferences represented by floating point numbers (e.g. price), $j_{IncreasingRate}$ is calculated by dividing the min-max-interval $[i_j, u_j]$ into k equally sized buckets (line 7). Then, for each sampling iteration (line 10), the following steps are executed. First, the current value of the upper-bound constraint $restrValue_j$ is updated (line 11). Next, we retrieve the solution by running *SPM* with given O , D , t_{dep} , the respective objective i , as well as with a constraint set C (line 12). Set C consists of the respective $restrValue_j$ and upper intervals u_r for all remaining traveler preferences $r \in P$ as additional upper-bound constraints. This creates solutions which respect all constraints defined by I . Finally, the merged set $S_{all} = \bigcup_{i,j|i \neq j \in P} S_{i,j}$ is built.

Algorithm 4 Approximation of two-dimensional sets of multimodal solutions

```

1: function SYSTEMATICSAMPLINGWITHININTERVALS( $O, D, t_{dep}, I, P, k$ )
2:   for all  $i, j | i \neq j \in P$  do ▷ Parallelized Execution
3:     if  $j$  s.t. integer condition then
4:        $j_{IncreasingRate} = 1$ 
5:        $\hat{j} = u_j - l_j$ 
6:     else
7:        $j_{IncreasingRate} = (u_j - l_j)/k$ 
8:        $\hat{j} = k$ 
9:     end if
10:    while  $\hat{j} \geq 0$  do
11:       $restrValue_j = l_j + \hat{j} * j_{IncreasingRate}$ 
12:       $Sol \leftarrow SPM(O, D, t_{dep}, i, C)$ 
13:       $S_{i,j} = S_{i,j} \cup Sol$ 
14:       $\hat{j} = \hat{j} - 1$ 
15:    end while
16:  end for
17:  return  $S_{all} = \bigcup_{i,j | i \neq j \in P} S_{i,j}$ 
18: end function

```

4.3.2.3 Removal of dominated solutions

In the next step, we follow a standard approach of MCDM for reducing the identified solutions to Pareto-optimal solutions (Deb, 2011). Having identified several multimodal itineraries, the set S_{all} can contain some dominated solutions. This is explained by merely considering two preferences simultaneously while creating $S_{all} = \bigcup_{i,j | i \neq j \in P} S_{i,j}$. For instance, the set $S_{travelTime,price}$ might contain a solution which dominates some solutions in the set $S_{price,numberOfTransfers}$. Hence, S_{all} needs to be reduced to a set of Pareto-optimal solutions S_{all}^{opt} . A solution s_1 dominates a solution s_2 if s_1 is strictly superior to s_2 according to at least one traveler preference $i \in P$ and not inferior regarding all other preferences (Delling et al., 2013a). The remaining solutions build up the (approximated) Pareto front. We use a standard package in our implementation to reduce the set efficiently (Blank & Deb, 2020).

4.3.3 Approximation of three-dimensional sets of solutions

In addition to the determination of two-dimensional solution sets $S_{i,j}$ (see Section 4.3.2.2), we now examine taking three traveler preferences into account simulta-

neously. This results in three-dimensional sets $S_{i,j,h}^3$ for each considered preference $i, j, h | i \neq j \wedge i \neq h \wedge j < h \in P$.

Algorithm 9 in Appendix A shows the details for the creation of three-dimensional solution sets $S_{i,j,h}^3$, which again can be computed in parallel (line 2). Hereby, traveler preference i is set as the objective, and preferences j and h are set as upper-bound constraints for SPM . By systematically altering these upper-bound constraints according to a predefined sampling density value k , several optimal solutions for the three-dimensional set can be identified as follows. In the first step, we create a sampling list $SL := (SL_1, \dots, SL_n)$ with $SL_1 = (j_1, h_1), \dots, SL_n = (j_n, h_n)$ containing tuples with different upper-bound constraint values for preferences j and h , respectively. As for two-dimensional sampling, the number of tuples to be sampled depends on the data types of j and h as well as the sampling density value k (line 3 to line 26).

In the next step, we run SPM for every sampling setting in SL (line 28). $SL_{current} \in SL$ identifies the next entry to be examined. As the highest upper-bound constraint combination is stored at the first position of SL_1 , we set $SL_{current}$ to that entry and remove it from SL (line 27). For $SL_{current}$, we run SPM with given O, D, t_{dep} , the respective objective i , and with constraint set C (line 29 to line 32). The set C consists of the respective upper-bound constraint for the preferences j and h (as $SL_{current_j}$ and $SL_{current_h}$, respectively) and the previously identified upper intervals u_r for all remaining traveler preferences $r \in P$ as additional upper-bound constraints. Given C , it is possible that no feasible solution can be found. Hence, we check whether a valid solution has been retrieved (line 30) before adding that solution to $S_{i,j,h}^3$ (line 31). Finally, we update the next upper-bound constraint combination $SL_{current} \in SL$ to be investigated (line 33 to line 38). To ensure a maximum impact of the tuning technique of skipping entries (see Section 4.3.7), we set $SL_{current}$ to a non-dominated entry in SL (line 33). In case SL contains more than one non-dominated entry, we take that entry found furthest away from the last solution Sol to diversify the sampling progress (line 37).

4.3.4 Multimodal network model

As a foundation for our multimodal routing algorithm SPM , a network model G representing and integrating multiple mobility services is needed. Following Pajor (2009), for each considered mobility service $ms \in MS$, we generate a unimodal

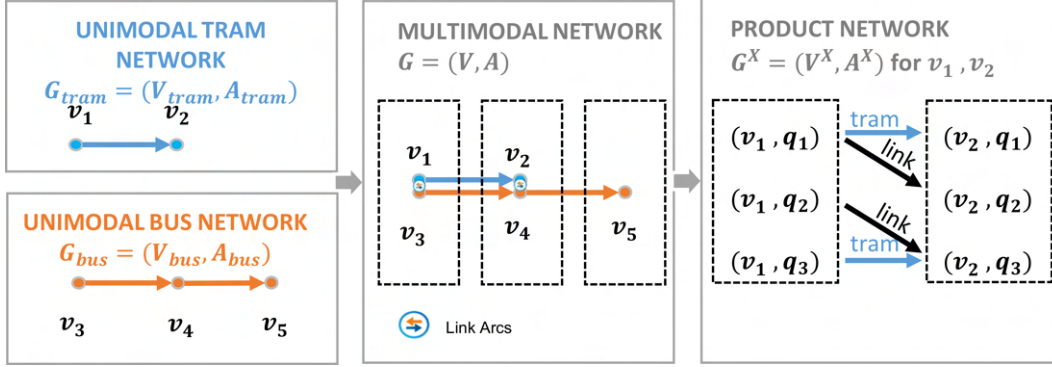


Figure 4.5: Exemplary multimodal network model

network $G_{ms} = (V_{ms}, A_{ms})$. As common for time-expanded network modeling, the set of vertices V_{ms} represents arrival and departure events, assigned to a stop location and a trip. Each stop location contains a label referring to $ms \in MS$ and information about the stop's name, coordinates, timezone, and further useful attributes such as wheelchair accessibility. A trip is a sequence of two or more vertices that occur during a specific time period. The set A_{ms} contains all arcs modeling a valid trip/subsequence of a trip between two vertices. Each arc $a \in A$ is characterized by $a = (v_1, v_2, label_{ms}, \varphi)$. The vertices $v_1, v_2 \in V_{ms}$ identify the respective successive events (departure and arrival vertices) which are assigned to the same trip. The assigned mobility service label $ms \in MS$ is represented by the $label_{ms}$ attribute. φ contains information about the cost values such as price, travel time, etc. assigned to that specific arc. Note that the set A_{ms} contains only those arcs which result in a valid trip. Figure 4.5 (left box) depicts an example for two unimodal networks G_{tram} and G_{bus} for one trip with two and three vertices, respectively. For the sake of simplicity, we present arrival and departure events as one vertex.

To consider a variety of mobility services simultaneously, we merge all unimodal networks into one multimodal network $G = (V, A)$. The set of vertices (arcs) is created by merging all unimodal sets V_{ms} (A_{ms}) with $ms \in MS$ into one set V (A). In addition, a transfer arc $a = (v_1, v_2, 'transfer', \varphi)$ is added to A in case two vertices v_1 and v_2 of the same mobility service are at the same stop location, but represent different arrival and departure events, and no time violation occurs. Transferring between different mobility services is provided by adding a link arc $a = (v_1, v_2, 'link', \varphi)$ to

set A between vertices v_1 and v_2 of different mobility services. These links will only be created if the services are in a given walking distance wd^{max} . For link and transfer arcs, we assume a predefined walking speed ws . Additionally, we integrate car usage into the multimodal network model. Assuming that a car is available at the traveler origin O , and an itinerary cannot only be partially driven by car, we add an arc $a = (O, D, 'car', \varphi)$ from O to D . Figure 4.5 presents the unimodal networks merged into one multimodal network G containing five vertices. The dashed boxes represent vertices at the same stop location. Link arcs to transfer between different mobility services are added between v_1 and v_3 as well as v_2 and v_4 , respectively. Transfer arcs are not integrated in the exemplary multimodal network as merely one trip is shown for each mobility service.

To reduce the size of the resulting multimodal network G and thus speed up the optimization, we apply the concept of CH. CHs introduce shortcuts in the network between relevant vertices (Geisberger et al., 2008). We are examining merely long-distance trips. Therefore, we model transferring between two long-distance vertices (assigned to e.g. railway or flight) using local transit mobility services as one arc. We abstract thereby from non-relevant details from local transit. Furthermore, we limit the usage of local transit CHs to only those local transit zones that either the origin and/or the destination vertex are assigned to. Hereby, expensive computational analysis of local transit usage of other local transit zones is avoided.

As highlighted by Pajor (2009) and Bast et al. (2015), solving a multimodal routing algorithm on G can yield unrealistic solutions. For instance, optimizing for travel time might lead to taking a train in between two car sections in case a highspeed train service is available. The avoidance of these unrealistic solutions is ensured by non-deterministic finite automata fa , which represent the travelers' mode choice in the network model. Following Pajor (2009), $fa := (Q, \Sigma, \delta, Q_I, Q_F)$ consists of a set Q of states, an alphabet Σ , a transition function $\delta : Q \times \Sigma \rightarrow P(Q)$, $Q_I \subseteq Q$ as a set of initial states and $Q_F \subseteq Q$ as a set of final states. The main idea is that all solutions retrieved by SPM have to fulfill the solution structure as defined by fa , i.e. it is guaranteed that there is a path in the transition graph from an initial state $q_i \in Q_I$ to a final state $q_f \in Q_F$. Figure 4.6 visualizes an exemplary fa by its transition graph, whereas each state $q \in Q$ is represented as a vertex. We insert an arc from q to q' (labeled by $\sigma \in \Sigma$) for each state $q \in Q$ and every label $\sigma \in \Sigma$ if $q' \in \delta(q, \sigma)$.

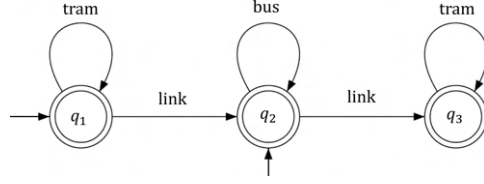


Figure 4.6: Exemplary finite automaton

This exemplary finite automaton consists of a set of states $Q = Q_I = Q_F := \{q_1, q_2, q_3\}$, an alphabet $\Sigma := \{tram, link, bus\}$, and an exemplary transition function δ as visualized in Figure 4.6. According to this finite automaton, merely walking and using railway is allowed.

Given the multimodal network $G = (V, A)$ and finite automata $fa := (Q, \Sigma, \delta, Q_I, Q_F)$, we build the product network $G^\times = (V^\times, A^\times)$. The set of vertices V^\times in the product network consists of vertices $(v, q) \in V^\times$ with $v \in V$ and $q \in Q$. The set of arcs A^\times consists of arcs (v_1^\times, v_2^\times) with $v_1^\times = (v_1, q_1)$, $v_2^\times = (v_2, q_2) \in V^\times$. An arc is inserted into A^\times only if $a = (v_1, v_2) \in A$ and an existing label $\sigma \in \Sigma$ is included in the transition function δ such that $q_2 \in \delta(q_1, \sigma)$. The cost φ and the respective label of the new arc is set according to the arc $a = (v_1, v_2) \in A$. For all retrieved solutions by *SPM* on G^\times , it is thereby ensured that they follow the solution structure as defined by fa . Finally, the right box in Figure 4.5 depicts the product network G^\times for v_1 and v_2 and the finite automata with its transition graph visualized in Figure 4.6. Product network vertices are inserted for each vertex and each state $q \in Q$. Arcs are inserted according to the transition graph. This product network can be extended for all vertices of the multimodal network G .

4.3.5 Resource-constrained multimodal routing

We build on the label-constrained multimodal shortest-path algorithm *SPM* as presented by Pajor (2009). This is a variant of Dijkstra's shortest path algorithm integrating finite automata. We extend the given algorithm so that we can consider several constraints in parallel. We systematically rerun *SPM* on G^\times , which is implicitly computed based on the multimodal network G . Additionally, we consider the given finite automaton $fa := (Q, \Sigma, \delta, Q_I, Q_F)$. Algorithm 5 shows the resource-constrained multimodal routing algorithm.

Algorithm 5 Resource-constrained multimodal routing algorithm

```

1: function SPM( $O, D, t_{dep}, obj, C$ )
2:   Open PriorityQueue PQ
3:   for all  $q_i \in Q_I$  do
4:     PQ.insert( $(O, q_i), 0$ )
5:   end for
6:   settled-targets  $\leftarrow \emptyset$ 
7:   while not PQ.isEmpty() do
8:      $(v, q) \leftarrow pop(PQ_1)$ 
9:     if  $v \in D \wedge q \in Q_F$  then
10:      settled-targets  $\leftarrow$  settled-targets  $\cup v$ 
11:      if settled-targets =  $D$  then
12:        return  $Sol \leftarrow returnBestFoundPath()$ 
13:      end if
14:    end if
15:    for all outgoing arcs  $a = (v, w)$  do
16:      for all states  $q' \in \delta(q, label(a))$  do
17:        if  $(w, q')$  has not been examined before then
18:          if noRestrictionViolationOccurs( $(w, q'), C$ ) then
19:            PQ.insert( $(w, q'), cost_s^{obj}((v, q)) + \varphi(a)^{obj}$ )
20:          end if
21:          else if  $cost_s^{obj}((v, q)) + \varphi(a)^{obj} < cost_s^{obj}((w, q'))$  then
22:            if noRestrictionViolationOccurs( $(w, q'), C$ ) then
23:              PQ.updateCost( $(w, q') \leftarrow cost_s^{obj}((v, q))$ )
24:            end if
25:          end if
26:        end for
27:      end for
28:      sort(PQ)
29:    end while
30: end function

```

Following the main idea of Dijkstra's algorithm, we insert all product vertices (O, q_i) with origin O and any initial states of the applied finite automata $q_i \in Q_i$ with no assigned cost to the priority queue PQ (line 4). While PQ is not empty (line 7), all neighboring product vertices are examined (line 15) according to the underlying transition function (line 16). In case no restriction violation occurs, the examined neighboring product vertex is either inserted into PQ – if it has not been examined

before –, or the respective cost value is updated if the examined neighboring product vertex has been examined already and an improved path has been identified (line 17 to line 25). Finally, PQ is sorted in ascending order by its cost values (line 28). The optimal path is identified once all product network vertices $(v, q) \in V^X$ with $v \in D$ and $q \in Q_F$ (indicating the final states of the applied finite automata) have been set as the current priority queue entry under investigation (line 9 to line 14).

The function $cost_s^{obj}((v, q))$ indicates the respective cost value dependent on the objective obj of entry (v, q) in the priority queue PQ . For an easier and comprehensible illustration, we have abstracted from some details in the algorithm, esp. from time-feasibility checks in the network. Furthermore, information for each entry in $(v, q) \in PQ$ about its cost values with respect to all considered preferences P as well as information to its predecessor are available, but not explicitly shown. Additionally, we copy priority queue entries with all its associated information in specific cases (e.g. better objective value found, but worse restriction value) to ensure optimality.

4.3.6 Complexity analysis

We compare the complexity of approximating all two-dimensional and three-dimensional sets of multimodal itineraries by analyzing the maximum number of runs of the multimodal routing algorithm SPM . These can be estimated as follows, with $CO(n, r) = \frac{n!}{(r! * (n-r)!)}$ representing the number of combinations of a set of size n resulting in entries of size r :

$$\text{two-dimensional:} \quad |P| * (|P| - 1) * k \quad (4.3)$$

$$\text{three-dimensional:} \quad |P| * CO(|P| - 1, 2) * k^2. \quad (4.4)$$

Table 4.1 compares the maximum number of runs of SPM dependent on different sampling density values k and the number of considered traveler preferences $|P|$. For $k = 8$ and $|P| = 3$, creating S_{all} requires at most 48 runs of SPM , whereas creating S_{all}^3 requires four times as many runs. This ratio increases significantly if the problem setting becomes more complex. Considering five traveler preferences in the framework increases the ratio to 12. Additionally, with more fine-grained sampling, the search becomes more computationally expensive. For instance, for $k = 16$ and $|P| = 5$, the

ratio increases up to 24 due to the higher sampling density value k . Since we are able to parallelize the computation of the sets, we achieve a run time dependent on k (two-dimensional) and k^2 (three-dimensional), respectively.

Table 4.1: Comparison of maximum runs for different sampling densities

Maximum runs of SPM	# of Considered Preferences $ P $	Two-Dim. Analysis S_{all}	Three-Dim. Analysis S_{all}^3	Ratio
Sampling density $k = 8$	3	48	192	4
	4	96	768	8
	5	160	1920	12
Sampling density $k = 16$	3	96	768	8
	4	192	3072	16
	5	320	7680	24

4.3.7 Speed ups

We improve the scalability of our solution space sampling framework by applying the following speed-up techniques.

Skipping iterations We can skip some iterations in the process of systematic sampling by analyzing information on solution dominance gained through recent sampling steps. This contributes to a significant performance improvement of the framework. Figure 4.7 demonstrates the main idea for the example of $S_{price,travelTime}$. While systematically altering one preference, a major advantage by starting at the upper bound of the min-max-interval for travel time $u_{travelTime}$ in the sampling process is that numerous runs of SPM can be skipped. For instance, if we run SPM with an upper-bound constraint for a travel time of 13:31 optimized by price, which results in a solution with a travel time of 9:30, we know that SPM would return the same solution if configured to an upper-bound constraint for travel time of 13:03 optimized by price. Skippable iterations are highlighted.

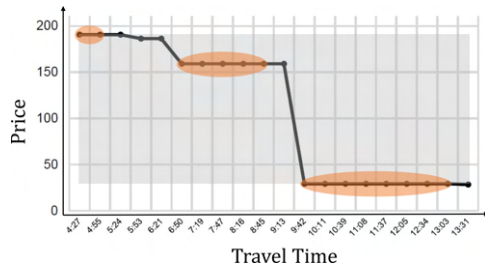


Figure 4.7: Iteration skipping in two-dimensional systematic sampling

This speed-up technique is formally represented in Algorithm 6. Here, Sol_j is the solution value found in the last iteration for preference j . This algorithm will only be used after the first iteration, as skipping of iterations is not possible in the first iteration.

Algorithm 6 Skipping of iterations in two-dimensional sampling

- 1: **function** ITERATIONCANBESKIPPED($Sol_j, restrValue_j$)
 - 2: **if** $Sol_j \leq restrValue_j$ **then**
 - 3: **return** True
 - 4: **else**
 - 5: **return** False
 - 6: **end if**
 - 7: **end function**
-

This approach can be extended to the three-dimensional setting as shown in Algorithm 7. Hereby, for all entries in the sampling list $sl \in SL$, it is checked if their respective preference values are greater or equal than in the recently found solution Sol as well as smaller or equal than in the currently examined setting $SL_{current}$. If this applies, we remove that entry from SL , since we do not gain any new information about the solution space.

Reuse of solutions from interval identification Solutions used for the determination of an interval boundary can be reused in the process of systematic sampling. This reduces the number of necessary runs of the multimodal routing algorithm SPM slightly. In particular, the set of solutions S_{lower} contains all solutions used to identify lower bounds l_i as described in Section 4.3.2.1. On the contrary, the set of solutions S_{upper} contains all solutions used to identify upper bounds u_i . Each solution

Algorithm 7 Skipping of iterations in three-dimensional sampling

```

1: function ITERATIONCANBESKIPPED( $Sol, SL_{currentj}, SL_{currenth}, SL$ )
2:   for all  $sl \in SL$  do
3:     if  $sl_j \geq Sol_j \wedge sl_h \geq Sol_h \wedge sl_j \leq SL_{currentj} \wedge sl_h \leq SL_{currenth}$  then
4:        $SL.remove(sl)$ 
5:     end if
6:   end for
7:   return  $SL$ 
8: end function

```

Sol stored in the sets S_{lower} or S_{upper} , respectively, has two additional attributes: Sol_{obj} comprises the single-objective set while identifying intervals. $Sol_{restriction}$ represents the preference $i \in P$ for which the min-max-interval $[l_i, u_i]$ has been updated. These two additional attributes are relevant to determine the respective solutions from S_{lower} and S_{upper} . Algorithm 8 shows the procedure more formally for the two-dimensional setting. Note that the usage of solutions from interval identification is merely applicable for the first (line 8) and the last iteration (line 2) of the sampling process.

Algorithm 8 Reuse of solution from interval identification

```

1: function REUSESOLFROMINTERVALIDENTIFICATION( $k, \hat{j}, i, j$ )
2:   if  $\hat{j} = 0$  then
3:     for all  $Sol \in S_{lower}$  do
4:       if  $Sol_{obj} = i \wedge Sol_{restriction} = j$  then
5:         return  $s$ 
6:       end if
7:     end for
8:   else if  $\hat{j} = k$  then
9:     for all  $Sol \in S_{lower}$  do
10:      if  $Sol_{obj} = i \wedge Sol_{restriction} = j$  then
11:        return  $s$ 
12:      end if
13:    end for
14:   end if
15:   return  $\emptyset$ 
16: end function

```

This procedure can be easily adapted to the three-dimensional setting. The minor difference is that both systematically altered restrictions $j, h \in P$ have to match the $Sol_{restriction1}$ and $Sol_{restriction2}$ attributes of the solution. Algorithm 10 (11) in Appendix B (C) show the approximation of two-dimensional (three-dimensional) multimodal itineraries with applied speed-up techniques, respectively.

4.4 Computational design

In the following, we introduce the experimental setup for evaluating the solution space sampling framework, normalization techniques, and metrics applied for evaluation.

4.4.1 Experimental setup

Our framework is applied to find Pareto front approximations for long-distance, multimodal itineraries between major cities in Germany. We examine 300 different origin-destination-combinations (ODs) between major cities assuming the earliest departure time t_{dep} at 9 am on October 8, 2018. For this date, we could collect data for all relevant mobility services. Furthermore, this date is quite sufficient for analyzing itineraries as it is a regular working day (Monday) as well as it is not a public holiday. For this one day of operation, our resulting multimodal network $G = (V, A)$ consists of about 40,000 vertices and 10 million arcs. We assume a maximum walking distance wd^{max} of $0.5km$. We set the walking speed ws to $5\frac{km}{h}$ for adding the link and transfer arcs as described in Section 4.3.4.

The set of mobility services MS includes a large amount of real-world data of services such as German Railways, Flixbus, and local transit services using published General Transit Feed Specification (GTFS) data. Furthermore, we consider flights and long-distance ridesharing services like “BlaBlaCar” based on real-world data collected from publicly available service data. We consider major airports with more than 50,000 aircraft movements per year (Berlin-Schönefeld, Cologne/Bonn, Berlin-Tegel, Dusseldorf, Frankfurt am Main, Munich, Stuttgart, Hamburg, Hanover, Leipzig/Halle, Nuremberg). We integrate information on the road network using the open-source routing library GraphHopper (www.graphhopper.com). For a more realistic estimation of travel times during peak hours, we multiply the retrieved travel times by 1.25.

As price information is not given for German Railways and Flixbus in the GTFS data, we estimate this as follows. For German Railways, the estimation is based on the train type chosen as well as the distance covered with the respective train type. We consider three different types of trains, namely *regional trains* (slowest train), *intercity trains* as well as *intercity express trains* (fastest train). We assume that intercity trains are 50% more expensive and intercity express trains are 100% more expensive in comparison to regional trains. Based on preliminary empirical investigations on *www.bahn.de*, the price for regional trains will be €17 per 100-kilometer distance traveled. For Flixbus, we assume €10 per 100-kilometer distance traveled. For individual road mobility, we assume 30 cents per kilometer following the flat-rate depreciation allowance in the German tax system. Note that all these values are only rough estimates. While integrating more realistic price information would make our study more impactful, we have abstracted from this in our proof-of-concept study. In reality, prices are dynamic and dependent on the OD combination, time of day, advance booking period, and flexibility on the days of travel (Randelhoff, 2022).

The set of traveler preferences P consists of: *travel time* (tt), *price* (pr), *number of transfers* (nt), *overall walking distance* (wd), and the overall *waiting time* (wt). We examine the following combinations:

- tt, pr
 $P := \{\text{travel time, price}\}$
- tt, pr, nt
 $P := \{\text{travel time, price, number of transfers}\}$
- tt, pr, nt, wd
 $P := \{\text{travel time, price, number of transfers, walking distance}\}$
- tt, pr, nt, wd, wt
 $P := \{\text{travel time, price, number of transfers, walking distance, waiting time}\}$

In the first set, we only consider travel time and price. These two preferences are perceived as essential decision criteria for the traveler (Grotenhuis et al., 2007). Then, we integrate additional preferences of high importance for the traveler into the search (Alt et al., 2019; Esztergár-Kiss & Csiszár, 2015; Grotenhuis et al., 2007). These are prevalent exemplary preferences as highlighted by the above studies. In general,

additional preferences which can be modeled as cost values of an arc in the multimodal network can be easily integrated into the search.

In addition, we examine the impact of different travelers' mode choice options presented in Appendix D. We consider three cases: (1) no mode choice restrictions, (2) using public transportation only, (3) allowing all services except flights. These mode choice options are integrated by nondeterministic finite automata fa as introduced in Section 4.3.4. These help avoid unrealistic solutions. The first finite automaton fa_{all} allows the usage of all integrated mobility services. The corresponding transition graph is shown in Figure 4.15. The finite automaton fa_{ptOnly} shown in Figure 4.16 limits the mobility offers to public transportation services including the walk to the first visited public transportation vertex and from the last visited public transportation vertex. Transferring between two public transportation services is modeled through arcs labeled as *link* and *transfer*, respectively. The finite automaton shown in Figure 4.17 forbids flights. In comparison to fa_{ptOnly} , car usage is allowed here.

We analyze the impact of different granularities of sampling on the scalability and quality of Pareto front approximation as follows. The granularity of the sampling is controlled by the sampling density parameter k . We evaluate the effect of setting k to 8, 16, 32, 64 and 128, respectively. Based thereon, we can compare whether investing time in more runs pays off. We also evaluate the impact of extending the search from two to three dimensions, as we expect that considering three dimensions at once improves the solution quality, but also results in larger computational effort.

The framework has been implemented in Java 12. The experiments are run on a multi-core environment with 16 core processors (AMD Epyc 7351 Processors) and 256GB of DDR4-2666 RAM.

4.4.2 Metrics

We use the following metrics to examine the results:

Run time [s] The run time in seconds provides information about the total run time the solution space sampling framework requires to approximate the set of Pareto-optimal itineraries $S_{traveler}$.

SPM-iterations per setting This metric shows the total number of iterations *SPM* runs across all sets $S_{i,j}$ for two-dimensional and $S_{i,j,h}^3$ for three-dimensional sampling, respectively, while identifying set $S_{traveler}$. This is reflected by the loop in line 7 of Algorithm 5. Note that this is different from a specific sampling iteration to identify a solution for a specific i, j in set $S_{i,j}$ (line 10 in Algorithm 4, line 28 in Algorithm 9, respectively).

Skipped iterations [%] shows the proportion of the iterations that can be skipped (Section 4.3.7).

Reused solutions [%] shows the proportion of iterations that can be reused from interval boundary determination (Section 4.3.7).

of Pareto-optimal solutions The size of set $S_{traveler}$ reflects the number of Pareto-optimal solutions retrieved by the solution space sampling framework.

MCDM metrics To evaluate the quality of our Pareto front approximation, we use several metrics proposed for MCDM such as **Inverted Generational Distance (IGD)** and **spread metric** (Deb, 2011; Riquelme et al., 2015). We apply 0-1-normalization before calculating these metrics. These metrics are used to get some technical insights into the structure of the solution set using different configurations (number of considered preferences, sampling density, and mode choice variant).

IGD can be assigned to the group of convergence-based methods. These measure the closeness of the identified solutions (here: set $S_{traveler}$) to the true Pareto front $S_{traveler}^{true}$. As $S_{traveler}^{true}$ is unknown in our setting, we build this set artificially by merging all sets across all applied sampling densities k per OD and fa into one set. Consequently, $S_{traveler}^{true}$ consists of all Pareto-optimal solutions across all applied sampling densities k . This serves for relative comparison to $S_{traveler}$.

Figure 4.8(a) visualizes the idea of IGD: the average distances from the solutions in $S_{traveler}^{true}$ (indicated in black) to $S_{traveler}$ (indicated in orange) are measured. Thus, a small value for IGD indicates a good convergence. More formally, IGD is calculated as:

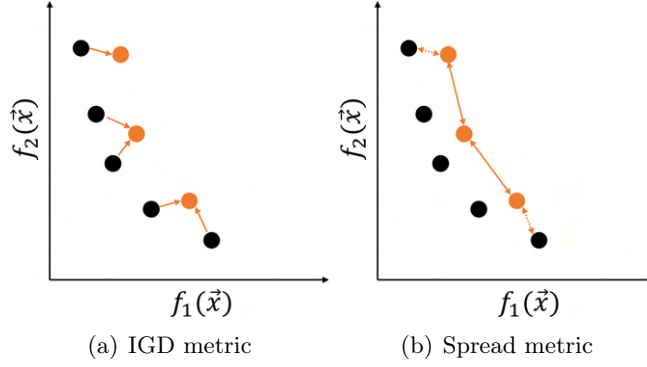


Figure 4.8: Performance metrics

$$IGD(S_{traveler}^{true}, S_{traveler}) = \frac{(\sum_{i=1}^{|S_{traveler}^{true}|} d_i^q)^{1/q}}{|S_{traveler}^{true}|}. \quad (4.5)$$

For $q = 2$, the parameter d_i represents the Euclidean distance between the solution $i \in S_{traveler}^{true}$ and the nearest solution in set $S_{traveler}$ (Deb, 2011). $f_j(\vec{x})^i$ is the j -th objective function value of $i \in S_{traveler}^{true}$ and $f_j(\vec{x})^k$ is the j -th objective function value of $k \in S_{traveler}$, with

$$d_i = \min_{k \in S_{traveler}} \sqrt{\sum_{j=1}^n \left(f_j(\vec{x})^i - f_j(\vec{x})^k \right)^2}. \quad (4.6)$$

The spread metric Δ helps measure the diversity of the solutions found by looking at the distribution of the identified solutions in $S_{traveler}$ (Deb, 2011). Figure 4.8(b) demonstrates the basic principle of this metric. For each solution $i \in S_{traveler}$ (indicated by orange circles), the Euclidean distances to neighboring solutions d_i are measured. To obtain an indication of how uniformly distributed the solutions in the solution space are, the average distance \bar{d} over solutions is calculated. In addition, the distance d^e between extreme solutions of set $S_{traveler}$ and set $S_{traveler}^{true}$ is taken into account (indicated by dotted lines). The spread metric Δ is zero for an equal

distribution of solutions in set $S_{traveler}$ in the solution space. The smaller the value of Δ is, the more diverse the solutions in $S_{traveler}$ are. Formally, Δ is calculated as:

$$\Delta = \frac{\sum_{j=1}^n d_j^e + \sum_{i=1}^{|S_{traveler}|} |d_i - \bar{d}|}{\sum_{j=1}^n d_j^e + |S_{traveler}| \bar{d}}. \quad (4.7)$$

Some of the metrics we use to examine the results are not free from the scaling of the considered traveler preferences (Deb, 2011). To address this issue and to ensure comparability across different traveler preferences and across different ODs, we normalize the preference values between 0 and 1 or by their mean:

- For 0-1-normalization, the minimum value for the respective preference is 0 and the maximum value equals 1. The normalized value \vec{x}'_i of preference value of \vec{x}_i for preference $\vec{x} \in P$ with \vec{x}_{min} as the minimum value for \vec{x} and \vec{x}_{max} as the maximum value for \vec{x} is calculated as:

$$\vec{x}'_i = \frac{\vec{x}_i - \vec{x}_{min}}{\vec{x}_{max} - \vec{x}_{min}}. \quad (4.8)$$

- To analyze the relationship between different traveler preferences, a normalization of preference values \vec{x}_i according to their mean values $mean(\vec{x})$ is performed. This indicates the deviation from the respective average. The mean-normalized preference \vec{x}^*_i is calculated as:

$$\vec{x}^*_i = \frac{\vec{x}_i}{mean(\vec{x})}. \quad (4.9)$$

4.5 Results

In this section, we analyze the results based on an aggregated view across 300 OD pairs. We then break the results down by number of dimensions, impact of the number of considered preferences, computational performance, and traveler-related metrics.

4.5.1 Summary results

Table 4.2: Summary of results

Dimensions	avg. run time [s]	avg. # Pareto-opt. solutions	avg. # skipped iterations [%]
two-dim.	10.86	5.13	85.22
three-dim.	40.56	5.81	97.31

Table 4.2 shows the results aggregated across all analyzed ODs, the number of considered traveler preferences P , applied mode choice variant fa , and sampling densities k . These are analyzed further in the following subsections. We can see that the creation of two-dimensional sets requires about 11 seconds on average and our framework finds 5.1 Pareto-optimal solutions per request. Hereby, on average, about 85% of the iteration steps can be skipped. The approximation of three-dimensional sets creates more Pareto-optimal solutions – 5.8 on average, i.e. about 14% more. However, this requires much more run time (about 373%), and more iterations can be skipped. This indicates that taking more traveler preferences into account simultaneously has an enormous impact on the run time, but not so much on the solution quality. In the following, we will analyze these results in detail.

4.5.2 Technical results

Table 4.3: Comparison of two and three-dimensional experiments

	Increase in comparison to two-dimensional sampling [%]	
	Run time	# Pareto-optimal solutions
tt, pr, nt	56.4	3.2
tt, pr, nt, wd	212.3	9.0
tt, pr, nt, wd, wt	343.2	17.0

Number of dimensions In the first step, we analyze the impact of the number of dimensions by comparing the number of Pareto-optimal solutions and the run time of a two-dimensional setting and a three-dimensional setting (Table 4.3). The presented run time as well as the number of Pareto-optimal solutions are relative to the two-dimensional sampling. Note that we do not examine the results for $P :=$

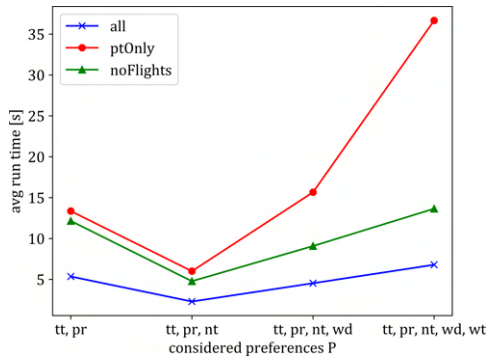
{travel time, price}, as at least three preferences have to be considered simultaneously for the approximation of three-dimensional solution sets.

For $P := \{\text{travel time, price, number of transfers}\}$, the run time increases by 56.4%. However, merely additional 3.2% Pareto-optimal solutions can be identified. This increase in run time can mainly be explained by the higher complexity of identifying three-dimensional solution sets as discussed in Section 4.3.6. The average increase in the number of skipped iterations (also see Table 4.2) cannot compensate for the increased complexity of the sampling process. In case four (five) traveler preferences are being considered simultaneously, the complexity of the sampling process increases exponentially. Thus, the average increase in run time is on average 212.3% (343.2%) higher for three-dimensional sampling. However, the number of additionally identified non-dominated solutions is 9.0% (17.0%).

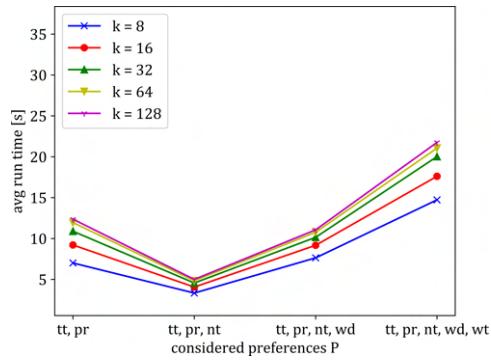
We conclude that integrating additional dimensions above two dimensions into the search has a severe impact on the run time. However, we can find a few additional solutions which may be of interest to the traveler. These findings are consistent with related research, which encountered scaling issues when taking into account additional preferences (Bast et al., 2013; Delling et al., 2013a; Dib et al., 2017).

Number of considered preferences We analyze the impact of the number of considered traveler preferences in the set P while approximating two-dimensional sets. Results are shown in Figure 4.9, with preference sets of an increasing number along the x -axis. The results for different metrics aggregated by different mode choice variants (left plots) as well as by different sampling densities (right plots) are presented on the respective y -axis.

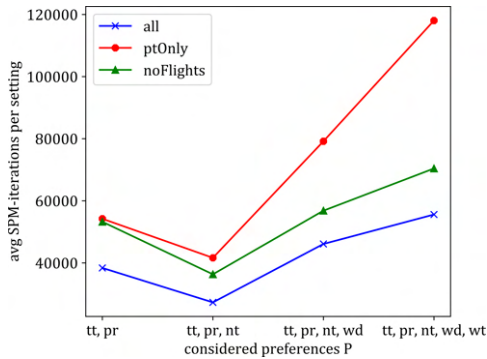
Number of considered preferences by finite automata In Figure 4.9(a), it can be seen that applying different mode choice variants has a significant impact on the run time. Allowing the usage of all mobility services (fa_{all} , blue line) requires the shortest run time. Limiting the search to public transportation services (fa_{ptOnly} , red line) results in the longest run time. Furthermore, forbidding flights ($fa_{noFlights}$, green line) requires slightly more run time on average than fa_{all} . For instance, with two traveler preferences, applying fa_{all} takes about 5 seconds on average, whereas applying fa_{ptOnly} requires about 13 seconds on average.



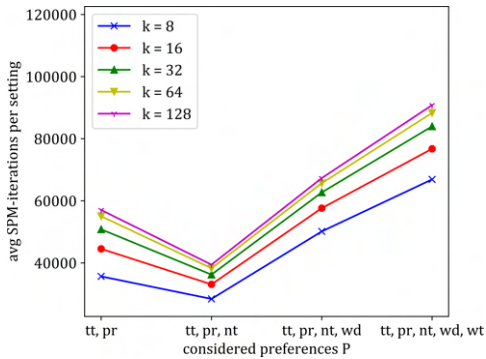
(a) avg. run time by mode choice variant



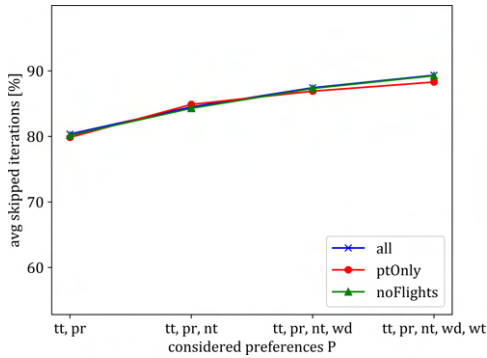
(b) avg. run time by sampling density



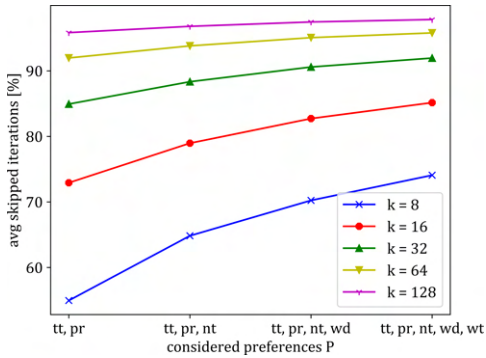
(c) avg. sum iterations of *SPM* by mode choice variant



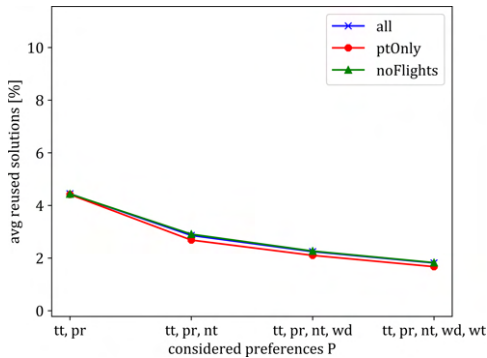
(d) avg. sum iterations of *SPM* by sampling density



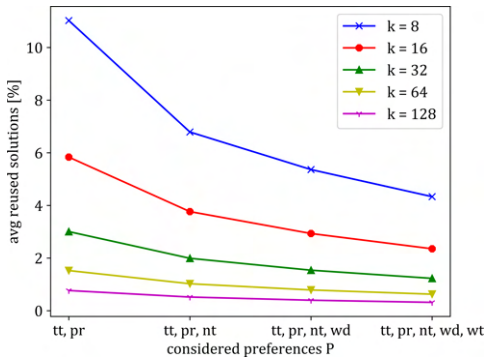
(e) avg. skipped iterations by mode choice variant



(f) avg. skipped iterations by sampling density



(g) avg. reused solutions by mode choice variant



(h) avg. reused solutions by sampling density

Figure 4.9: Impact of preferences by mode choice variant and sampling density

The average run time decreases slightly if a third preference is taken into account; e.g., for the number of transfers, run times are about 2 and 6 seconds, respectively. Adding a fourth and fifth traveler preference to P increases the run time for $f_{a_{all}}$ and $f_{a_{noFlights}}$ constantly. Contrary to this, for public transportation mode choice, the average run time more than doubles, resulting in values about 16 and 37 seconds, respectively. The overall number of SPM -iterations (Figure 4.9(c)) follows the overall trend, i.e., more iterations come with higher run times.

Figure 4.9(e) presents the average proportion of skipped iterations. Regarding mode choice variants, no significant differences can be observed with respect to the average proportion of skipped iterations. The more preferences are considered at the same time, the more iterations can be skipped proportionally. While for $P := \{\text{travel time, price}\}$ roughly 80% of the iterations can be skipped, this metric increases up to approximately 89% for five traveler preferences. In summary, when additional preferences are considered in terms of constraints, there is an increase in run time, but the relative increase decreases with each additional preference.

Finally, we compare the results with respect to the average proportion of reused solutions from interval identification in Figure 4.9(g). No differences for mode choice variants can be observed with this speed-up technique. The respective proportion decreases as the size of set P increases iteratively.

Number of considered preferences by sampling density Next, we present results for different sampling densities k (right plots in Figure 4.9). For the average run time as shown in (Figure 4.9(b)), the lowest sampling density with $k = 8$ (blue line) requires the least run time on average. Doubling the sampling density leads to a small increase of the average run time. If we apply a sampling density 16 times as high with $k = 128$ (purple line), the average run time is only 140-180% compared to $k = 8$. For instance, in case merely travel time and price (tt, pr) are taken into account, the framework requires on average about 7 seconds for $k = 8$ and about 12 seconds for $k = 128$, respectively. In case five traveler preferences are considered, we need about 15 and 22 seconds on average, respectively. As already seen in the analysis of the mode choice variants, a comparable development of the run time with regard to the number of preferences can be observed here.

Figure 4.9(d) presents the results for the overall number of SPM -iterations across different sampling densities k . Obviously, applying a higher sampling density results

in more runs. The curve shape is similar to the ones observed for the run time analysis under different preference combinations.

With respect to the average proportion of skipped iterations, major differences can be seen for different sampling densities (Figure 4.9(f)). While for $P := \{\text{travel time, price}\}$ and $k = 8$ about 55% of overall iterations can be skipped, this proportion increases up to 96% for $k = 128$. Hence, a higher sampling rate results in a higher proportion of skipped iterations: the more traveler preferences are taken into account simultaneously, the larger the proportion of skipped iterations. For five traveler preferences, on average, about 74% and 98% of iterations are skipped, respectively. The increase diminishes with each additional preference.

Finally, we compare the results with respect to the average proportion of reused solutions from interval identification, see Figure 4.9(h). While for $P := \{\text{travel time, price}\}$ and $k = 8$ about 11% in the solutions from interval identification can be reused, this value drops to under 1% for $k = 128$. When more restrictions are considered, this proportion becomes smaller.

Number of considered preferences: Insights The analysis of the above metrics gives us the following insights on the scalability of the solution space sampling framework. In particular, we observe an efficient scaling with respect to the number of considered preferences. The scalability effect is guaranteed regardless of the multimodal routing algorithm and the underlying network. However, if $f_{a_{ptOnly}}$ is in the focus, the run time increases exponentially if four or five preferences are considered. This is due to decreasing flexibility of the solution space when excluding more flexible mode choice variants such as car usage. Consequently, many more iterations are needed than with $f_{a_{all}}$ and $f_{a_{noFlights}}$, which is demonstrated in Figure 4.9(c). Hence, focusing on timetable-based services can become much more complex compared to situations where car usage is allowed. Furthermore, forbidding flying ($f_{a_{noFlights}}$) increases the run time in comparison to applying no mode choice restrictions ($f_{a_{all}}$) although the multimodal network decreases in size. If arcs with flight labels exist in the network, examining these arcs enables covering long distances in the network, and thus destination vertices are faster selected from the priority queue as the currently examined node. Figure 4.9(c) supports the assumption that flight restrictions require a higher average number of SPM-iterations than no mode choice restrictions.

Table 4.4: Performance metrics by preference sets

Preference set	IGD	Spread Δ
tt, pr	0.0020	0.4030
tt, pr, nt	0.0025	0.3432
tt, pr, nt, wd	0.0034	0.3439
tt, pr, nt, wd, wt	0.0049	0.3600

Efficient scaling is also independent of the applied sampling density k (Figure 4.9(b)). As demonstrated, the run time increases slightly when the sampling density is doubled. This can be explained by examining the proportion of skipped iterations in the sampling framework in Figure 4.9(f): the more preferences are considered simultaneously, the more iterations can be skipped. The usage of speed-up techniques significantly alleviates the increase in run time for higher sampling densities. Furthermore, it is apparent that the average run time decreases slightly even if a third preference (here: number of transfers) is taken into account in comparison to $P := \{\text{travel time, price}\}$. Limiting the number of transfers seems to avoid searching for unpromising paths and enables faster termination of a *SPM* run.

From Figure 4.9(g) and Figure 4.9(h), it is apparent that taking more traveler preferences into account simultaneously results in a smaller proportion of reused solutions from interval identification in the sampling process. As more traveler preferences are considered, more different two-dimensional sets of solutions are created. As solutions from interval identification can merely be reused for interval boundaries, calculating more two-dimensional sets results in a smaller proportion of solutions reused from interval identification. With respect to the applied sampling density k , a smaller sampling density results in a higher proportion of reused solutions from interval identification as more sampling outside of the interval boundaries occurs. Consequently, relatively fewer solutions from interval identification can be reused.

Performance metrics Next, we analyze the performance of the sampling framework regarding Pareto front approximation using the IGD and spread metrics. Table 4.4 presents the metric values averaged over all ODs, mode choice variants, and sampling densities by different preference sets. It can be seen that the value for the IGD metric increases as more preferences are taken into account. For

Table 4.5: Performance metrics by mode choice variant and sampling density for $P := \{\text{tt, pr, nt, wd, wt}\}$

Mode choice variant	Metric	Sampling density k				
		8	16	32	64	128
all	IGD	0.0097	0.0036	0.0013	0.0003	0
	Spread	0.3106	0.3231	0.3324	0.3340	0.3347
ptOnly	IGD	0.0208	0.0096	0.0024	0.0014	0.0008
	Spread	0.3976	0.4124	0.4251	0.4318	0.4334
noFlights	IGD	0.1230	0.0047	0.0017	0.0013	0.0002
	Spread	0.3147	0.3306	0.3364	0.3409	0.3416

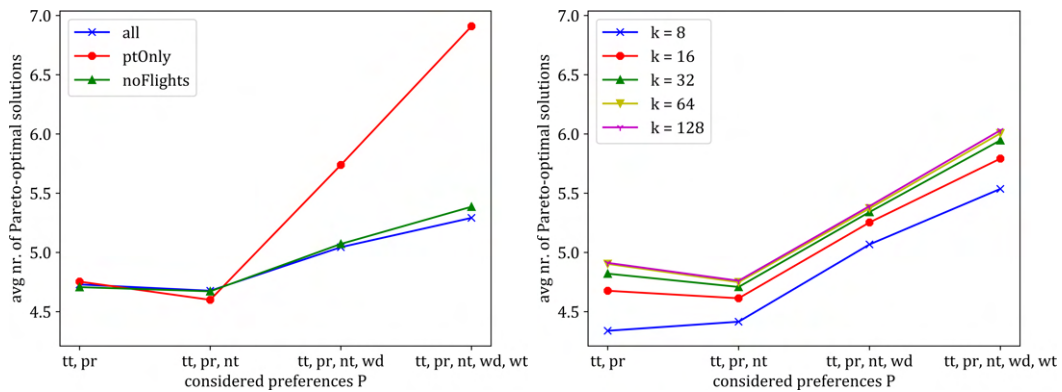
$P := \{\text{travel time, price}\}$, the IGD metric is 0.0020. This value increases up to 0.0049 in case five preferences are considered. Consequently, the sampling process converges slower if more preferences are taken into account; any additional preference makes the multimodal solution space more complex and leads to a higher overall distance to the true Pareto front. Regarding the diversity and the distribution of the solutions in the solution space, no clear conclusions can be drawn on the basis of the spread metric. While a spread value of $\Delta = 0.4030$ indicates a more unbalanced distribution of the solutions in the solution space for $P := \{\text{travel time, price}\}$, this diversity diminishes if more traveler preferences are considered.

We assess details of the MCDM metrics considering different sampling densities and mode choice variants in Table 4.5. These values are aggregated across all ODs. A higher sampling density leads to a better convergence indicated by IGD. This is as the true Pareto front is being built using merged sets across all investigated sampling densities. The true Pareto front is used for relative comparison against the set of identified solutions $S_{traveler}$. In particular, the doubling of the sampling density from $k = 8$ to $k = 16$ results in a significantly better convergence of the Pareto-optimal solution set. Regarding the spread metric, it can be seen that the retrieved non-dominated solutions are – on average – more diverse the higher the sampling density is. Once again, a major increase in diversity can be examined from $k = 8$ to $k = 16$. Thus, doubling the sampling density has a significant effect on the performance of the sampling process. This is also underlined by Figure 4.9 and 4.10. Differences regarding mode choice variants are similar to the analysis of Figure 4.9; both metrics show comparable values for all mode choice variants except public transit.

Further results for the IGD and spread performance metric by mode choice variant and sampling density for two, three or four preferences can be found in Table 4.6, Table 4.7, and Table 4.8 in Appendix E. The analysis regarding the performance metrics yields comparable findings as shown for five preferences in Table 4.5.

4.5.3 Traveler perspective

In the following, we analyze results from the traveler’s perspective. To this end, we examine the Pareto-optimal solutions and derive characteristics of potential interest for the traveler to make the search more transparent and explainable.



(a) Pareto-optimal solutions by mode choice variant (b) Pareto-optimal solutions by sampling density

Figure 4.10: Number of Pareto-optimal solutions by mode choice variant and sampling density

Pareto-optimal solutions The total number of retrieved Pareto-optimal solutions is of high importance for the traveler and can be seen in Figure 4.10 by increasing number of considered preferences. First, we examine the results aggregated by mode choice variant (Figure 4.10(a)). For two or three traveler preferences, 4.6 to 4.8 non-dominated multimodal solutions are retrieved on average. This number is independent of the mode choice variant. The average number of non-dominated solutions increases to 5.7 and 6.9 for four and five traveler preferences, respectively. While the increase is comparable for $f_{a_{all}}$ and $f_{a_{noFlights}}$, many more solutions can be

identified for $f_{a_{pt}Only}$. Enabling car usage often leads to solutions that have a lower travel time and have a lower price compared to public transit solutions. Therefore, these solutions dominate public transit solutions and thus lead to a lower number of total Pareto-optimal solutions.

Next, we analyze the results aggregated by different sampling densities (Figure 4.10(b)). It is apparent that with a higher number of preferences, more Pareto-optimal solutions can be found, as more criteria have an impact on Pareto optimality. As already observed in the run time analysis, a more fine-grained search has only a minor impact on the number of Pareto-optimal solutions. Even if we apply a sampling density 16 times as high with $k = 128$ (purple line), the average total number is merely 0.5 higher compared to $k = 8$ (blue line).

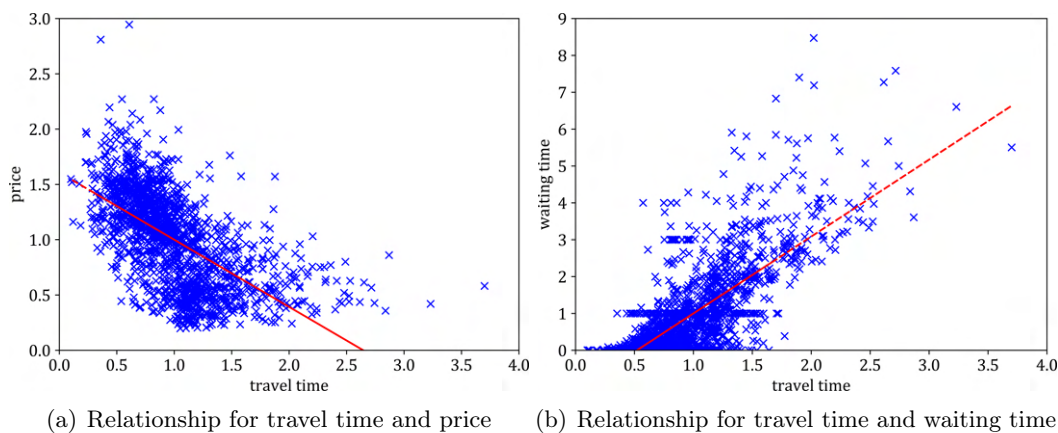
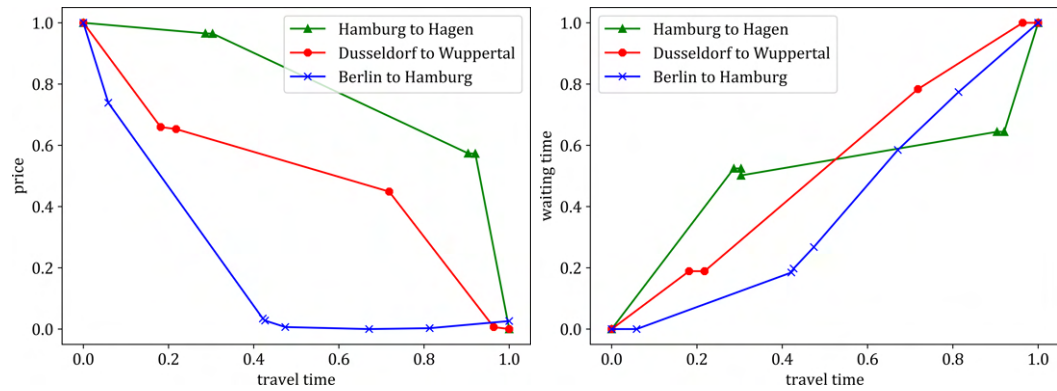


Figure 4.11: Exemplary relationships between preferences

Relationships between preferences To make the search of multimodal itineraries more transparent and explainable for the traveler in the light of available options, we analyze characteristics about the multimodal solution space. These can be presented to the traveler in a simplified and aggregated way. Figure 4.11 presents relationships between (mean-normalized) travel time and price as well as between travel time and waiting time for all ODs, sets of traveler preferences, mode choice variants, and sampling densities for two-dimensional solution sets. Each blue cross represents a solution value. The red line shows an approximated linear trend line.

As shown in the left figure, the more travel time the traveler is willing to invest, the more affordable the itinerary becomes. An opposing relationship can be observed in the right figure with respect to the relationship between travel time and waiting time. In general, an increase in travel time results in an overall higher waiting time. Further figures providing additional relationships between traveler preferences are shown in Figure 4.18 in Appendix F.



(a) Aggregated relationships for travel time and price (b) Aggregated relationships for travel time and waiting time

Figure 4.12: Exemplary aggregated relationships for different ODs

OD-specific relationships between different preferences Figure 4.12 illustrates the relationships between travel time and price as well as travel time and waiting time for three exemplary ODs. We present long-distance multimodal itineraries between the cities of Hamburg and Hagen (indicated by green crosses), Dusseldorf and Wuppertal (indicated by red circles), as well as Berlin and Hamburg (indicated by blue triangles) with five traveler preferences considered, public transit mode choice, and a sampling density of $k = 128$ applied. To ensure comparability, the preference values are 0-1-normalized.

Significant differences between travel time and price can be observed. For the request from Hamburg to Hagen, the traveler has to invest quite a significant amount of time until major cost savings occur. In contrast, the traveler has to invest only slightly more time in comparison to the best available travel time until large cost savings can be realized for traveling between Berlin and Hamburg. A balanced

ratio between travel time and price can be found for the request from Dusseldorf to Wuppertal. Hence, only minor differences can be seen regarding the relationship between travel time and waiting time. Thus, it can be concluded that the relationship between different traveler preferences is highly OD-dependent. Consequently, an OD-specific analysis is required to empower the traveler with additional information about solution space characteristics. This information could assist travelers in understanding and shaping their choice set and could support them in their decision-making process.

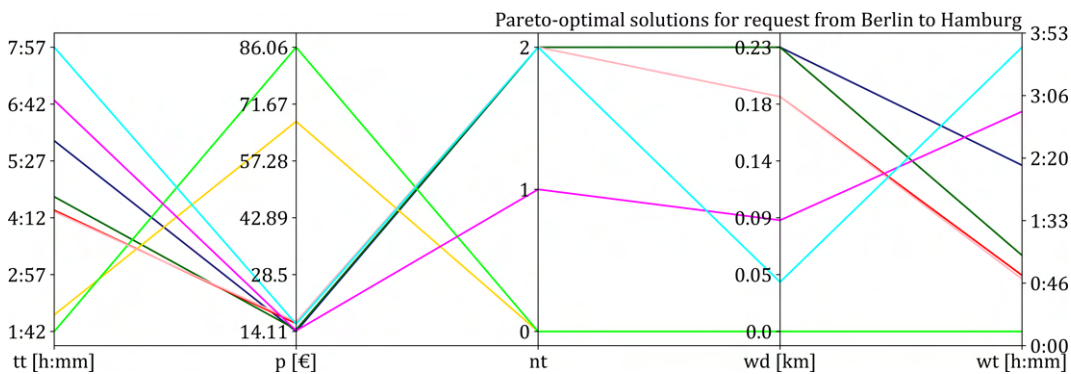


Figure 4.13: Parallel coordinates for the request from Berlin to Hamburg

We further analyze the identified itineraries for the request from Berlin to Hamburg in detail. Figure 4.13 displays the characteristics of the Pareto-optimal itineraries. High diversity in terms of the impact of individual preferences is apparent. The itinerary highlighted in light blue has the highest travel time, number of transfers as well as waiting time, and comparably small values for price and walking distance. Contrary to this, the itinerary highlighted in green has inverse properties except for walking distance. Presenting this information to the traveler could be the first step to making the available options more transparent and explainable. For instance, a traveler who assesses travel time as the most important individual preference receives valuable information about relevant itineraries and their relative quality regarding travelers' preferences.

Impact of three-dimensional sampling on the identified itineraries Finally, we present the identified itineraries for the request from Berlin to Hamburg for three-dimensional sampling in detail. Figure 4.14 displays the characteristics of Pareto-optimal itineraries. The itineraries highlighted in grey and light blue are the

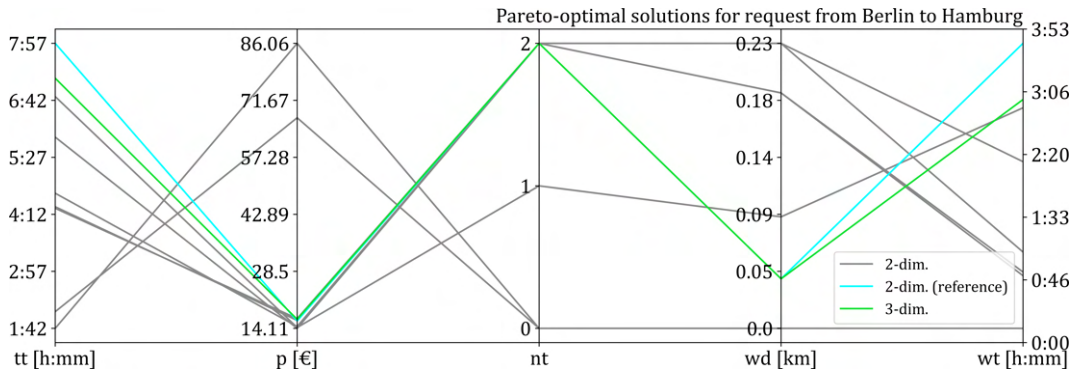


Figure 4.14: Parallel coordinates for the request from Berlin to Hamburg

same itineraries as identified for two-dimensional sampling in Figure 4.13, whereas the itinerary highlighted in green could not be found for two-dimensional sampling. The light blue, as well as the green itineraries, have similar characteristics. While the newly found itinerary is superior in terms of travel time and waiting relative to the blue one, the price increases only a bit. Consequently, expanding the dimensions of the sampling framework can lead to new relevant itineraries. However, these relevant itineraries can only be found with a comparatively large effort (see Section 4.5.2).

4.6 Conclusion

In recent years, there has been enormous progress in creating multimodal door-to-door itineraries. However, significant challenges remain when combining mobility services according to travelers' preferences. Travelers expect a choice of itineraries according to their individual preferences. Nevertheless, recent state-of-the-art MCDM approaches for multimodal routing struggle to scale efficiently when more than three traveler preferences are taken into account simultaneously. In this work, we have proposed an efficient multi-criteria decision support system for shaping a reasonable set of multimodal itineraries. In particular, we ensure scaling of the MCDM framework when additional traveler preferences are considered in a large multimodal network. The core idea is to apply solution space sampling on several two-dimensional sets in parallel to approximate the Pareto front. In addition, we have introduced speed-up techniques to improve the run time of the framework.

The solution space sampling framework has been evaluated by analyzing itineraries between large cities in Germany. We have included a large amount of real-world data of mobility services into the search, both scheduled as well as unscheduled services such as ridesharing. We have considered up to five prevalent traveler preferences – travel time, price, number of transfers, walking distance, and waiting time – to examine the scaling of the framework and to evaluate the outcome from a traveler’s perspective.

From a technical perspective, it can be concluded that the proposed framework scales well with a larger number of traveler preferences. When analyzing the impact of different mode choice variants, focusing on timetable-based services only can become much more complex compared to settings where car usage is allowed. Comparing our results with the approximation of three-dimensional sets in parallel reveals a significant increase in run time while merely a few additional non-dominated itineraries can be retrieved.

From a traveler’s perspective, systematic relationships between different preferences can be observed which are highly OD-dependent. Embedding these characteristics in existing mobility platforms as an expert menu, for example, could make the choice of appropriate itineraries more transparent and explainable for the traveler and help soften the black-box character of the complex multimodal solution space. This could provide the required input for research on design science addressing how a mobility platform should be designed in order to improve travelers’ decision making (e.g. Meske et al. (2020)).

In future work, we plan to extend the presented framework to enable the reduction of the Pareto-optimal solution set and examine the applicability of variety of published methods for this purpose such as Fuzzy Logic approaches (Delling et al., 2013a) and concepts of Types and Thresholds (Bast et al., 2013). Furthermore, we plan to evaluate the proposed framework against routing algorithms that identify the true Pareto-optimal set of multimodal itineraries such as McRAPTOR. As stated in Section 4.2.3, the run time of this approach deteriorates when multiple preferences are taken into account simultaneously. Applying the proposed framework ensures efficient scaling with respect to the number of traveler preferences. Regarding the number of identified Pareto-optimal solutions, it would be interesting to examine how

many solutions we are able to identify in comparison to algorithms that identify the true Pareto-optimal set of solutions.

In addition, we plan to integrate reinforcement learning techniques into the sampling framework. Thereby, information gained during the sampling process can be used to guide the sampling process into more promising parts of the solution space. This would allow identifying further relevant itineraries in efficient run time. Further OD-specific analysis can help to identify whether a more profound analysis of the multimodal solution space is promising for a given request and set of preferences. Furthermore, the experimental setting could be extended by integrating more complex traveler preferences such as sustainability and reliability. Moreover, integrating the complex choice behavior of travelers could make the framework more useful. This also holds for pricing, as the price values used in the paper – esp. for Deutsche Bahn and Flixbus – are merely rough estimations of actual service prices.

Appendices

A Approximation of three-dimensional sets of multimodal solutions

In the following, we highlight the differences to the framework described in Algorithm 2. Merely line 2 has to be altered in order to calculate a merged set S_{all}^3 . The input parameter in line 3 of Algorithm 2 has to be changed to S_{all}^3 . For $S_{i,j,h}^3$, we set traveler preference i as the objective and systematically alter preferences j and h according to a predefined sampling density value k .

Algorithm 9 Approximation of three-dimensional sets of multimodal solutions

```

1: function SYSTEMATICSAMPLINGWITHININTERVALS( $O, D, t_{dep}, I, P, k$ )
2:   for all  $i, j, h | i \neq j \wedge i \neq h \wedge j < h \in P$  do ▷ Parallelized Execution
3:     Open SamplingList  $SL$ 
4:     if  $j$  s.t. integer condition then
5:        $j_{IncreasingRate} = 1$ 
6:        $\hat{j} = u_j - l_j$ 
7:     else
8:        $j_{IncreasingRate} = (u_j - l_j)/k$ 
9:        $\hat{j} = k$ 
10:    end if
11:    if  $h$  s.t. integer condition then
12:       $h_{IncreasingRate} = 1$ 
13:       $\hat{h} = u_h - l_h$ 
14:    else
15:       $h_{IncreasingRate} = (u_h - l_h)/k$ 
16:       $\hat{h} = k$ 
17:    end if
18:    while  $\hat{j} \geq 0$  do
19:      while  $\hat{h} \geq 0$  do
20:         $restrValue_j = l_j + \hat{j} * j_{IncreasingRate}$ 
21:         $restrValue_h = l_h + \hat{h} * h_{IncreasingRate}$ 
22:         $SL = SL \cup (restrValue_j, restrValue_h)$ 
23:         $\hat{h} = \hat{h} - 1$ 
24:      end while
25:       $\hat{j} = \hat{j} - 1$ 
26:    end while
27:     $SL_{current} = pop(SL_1)$ 
28:    while  $SL \neq \emptyset$  do
29:       $Sol \leftarrow SPM(O, D, t_{dep}, i, C)$ 
30:      if  $Sol$  is valid then
31:         $S_{i,j,h} = S_{i,j,h} \cup Sol$ 
32:      end if
33:       $SL_{candidates} \leftarrow removeDominatedSolutions(SL)$ 
34:      if  $|SL_{candidates}| = 1$  then
35:         $SL_{current} \leftarrow pop(SL_{candidates}_1)$ 
36:      else
37:         $SL_{current} \leftarrow pop(entryWithMaxNormEuclDistanceFromSol(SL_{candidates}))$ 
38:      end if
39:    end while
40:  end for
41:  return  $S_{all}^3 = \bigcup_{i,j,h | i \neq j \wedge i \neq h \wedge j < h \in P} S_{i,j,h}^3$ 
42: end function

```

B Approximation of two-dimensional sets of multimodal solutions with speed-up techniques

Algorithm 10 Approximation of two-dimensional sets of multimodal solutions with speed-up techniques

```

1: function SYSTEMATICSAMPLINGWITHININTERVALS( $O, D, t_{dep}, I, P, k$ )
2:   for all  $i, j | i \neq j \in P$  do ▷ Parallelized Execution
3:     if  $j$  s.t. integer condition then
4:        $\hat{j}_{IncreasingRate} = 1$ 
5:        $\hat{j} = u_j - l_j$ 
6:     else
7:        $\hat{j}_{IncreasingRate} = (u_j - l_j)/k$ 
8:        $\hat{j} = k$ 
9:     end if
10:    while  $\hat{j} \geq 0$  do
11:       $restrValue_j = l_j + \hat{j} * \hat{j}_{IncreasingRate}$ 
12:      if ReuseSolFromIntervalIdentification( $k, \hat{j}, i, j$ )  $\neq \emptyset$  then
13:         $Sol \leftarrow$  ReuseSolFromIntervalIdentification( $k, \hat{j}, i, j$ )
14:         $S_{i,j} = S_{i,j} \cup Sol$ 
15:      else if IterationCanBeSkipped( $Sol_j, restrValue_j$ ) then
16:        ▷ Iteration can be skipped
17:      else
18:         $Sol \leftarrow$  SPM( $O, D, t_{dep}, i, C$ )
19:         $S_{i,j} = S_{i,j} \cup Sol$ 
20:      end if
21:       $\hat{j} = \hat{j} - 1$ 
22:    end while
23:  end for
24:  return  $S_{all} = \bigcup_{i,j | i \neq j \in P} S_{i,j}$ 
25: end function

```

C Approximation of three-dimensional sets of multimodal solutions with speed-up techniques

Algorithm 11 Approx. of 3-dim. sets of multimodal sol. with speed-up techniques

```

1: function SYSTEMATICSAMPLINGWITHININTERVALS( $O, D, t_{dep}, I, P, k$ )
2:   for all  $i, j, h | i \neq j \wedge i \neq h \wedge j < h \in P$  do ▷ Parallelized Execution
3:     Open SamplingList  $SL$ 
4:     if  $j$  s.t. integer condition then
5:        $\hat{j}_{IncreasingRate} = 1$ 
6:        $\hat{j} = u_j - l_j$ 
7:     else
8:        $\hat{j}_{IncreasingRate} = (u_j - l_j)/k$ 
9:        $\hat{j} = k$ 
10:    end if
11:    if  $h$  s.t. integer condition then
12:       $\hat{h}_{IncreasingRate} = 1$ 
13:       $\hat{h} = u_h - l_h$ 
14:    else
15:       $\hat{h}_{IncreasingRate} = (u_h - l_h)/k$ 
16:       $\hat{h} = k$ 
17:    end if
18:    while  $\hat{j} \geq 0$  do
19:      while  $\hat{h} \geq 0$  do
20:         $restrValue_j = l_j + \hat{j} * \hat{j}_{IncreasingRate}$ 
21:         $restrValue_h = l_h + \hat{h} * \hat{h}_{IncreasingRate}$ 
22:         $SL = SL \cup (restrValue_j, restrValue_h)$ 
23:         $\hat{h} = \hat{h} - 1$ 
24:      end while
25:       $\hat{j} = \hat{j} - 1$ 
26:    end while
27:     $SL_{current} = pop(SL_1)$ 
28:    while  $SL \neq \emptyset$  do
29:      if ReuseSolFromIntervalIdentification( $k, \hat{j}, \hat{h}, i, j, h$ )  $\neq \emptyset$  then
30:         $Sol \leftarrow$  ReuseSolFromIntervalIdentification( $k, \hat{j}, \hat{h}, i, j, h$ )
31:         $S_{i,j,h} = S_{i,j,h} \cup Sol$ 
32:      else
33:         $Sol \leftarrow SPM(O, D, t_{dep}, i, C)$ 
34:        if  $Sol$  is valid then
35:           $S_{i,j,h} = S_{i,j,h} \cup Sol$ 
36:        end if
37:      end if
38:      if  $Sol$  is valid then
39:         $SL \leftarrow$  IterationCanBeSkipped( $Sol, SL_{current_j}, SL_{current_h}, SL$ )
40:      end if
41:       $SL_{candidates} \leftarrow$  removeDominatedSolutions( $SL$ )
42:      if  $|SL_{candidates}| = 1$  then
43:         $SL_{current} \leftarrow pop(SL_{candidates}_1)$ 
44:      else
45:         $SL_{current} \leftarrow pop(entryWithMaxNormEuclDistanceFromSol(SL_{candidates}))$ 
46:      end if
47:    end while
48:  end for
49:  return  $S_{i,j,h}^3 = \bigcup_{i,j,h | i \neq j \wedge i \neq h \wedge j < h \in P} S_{i,j,h}$ 
50: end function

```

D Mode choice options



Figure 4.15: Finite Automata: All

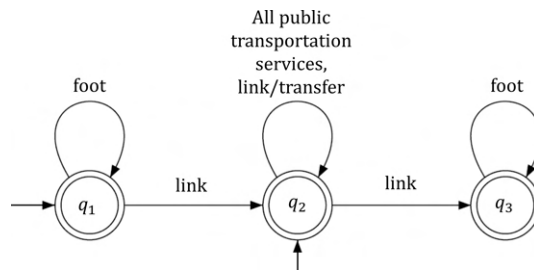


Figure 4.16: Finite Automata: Public transportation only

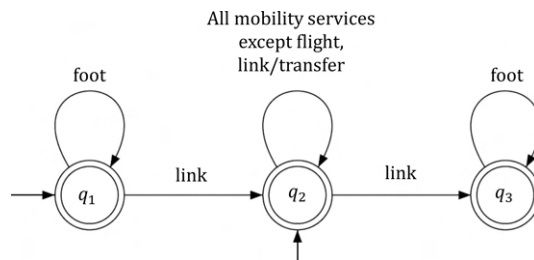


Figure 4.17: Finite Automata: No flights

E Performance metrics

Table 4.6: Performance metrics by fa and k for $P := \{tt, pr\}$

Mode choice fa	Metric	Sampling density k				
		8	16	32	64	128
all	IGD	0.0059	0.0018	0.0004	0	0
	Spread	0.3754	0.3996	0.4069	0.4102	0.4101
ptOnly	IGD	0.0085	0.0023	0.0007	0.0002	0
	Spread	0.3834	0.4063	0.4161	0.4220	0.4212
noFlights	IGD	0.0069	0.0022	0.0006	0.0001	0
	Spread	0.3695	0.3961	0.4072	0.4112	0.4105

Table 4.7: Performance metrics by fa and k for $P := \{tt, pr, nt\}$

Mode choice fa	Metric	Sampling density k				
		8	16	32	64	128
all	IGD	0.0078	0.0019	0.0006	0.0002	0.0002
	Spread	0.3148	0.3239	0.3347	0.3360	0.3373
ptOnly	IGD	0.0124	0.0023	0.0006	0.0001	0
	Spread	0.3468	0.3665	0.3732	0.3777	0.3789
noFlights	IGD	0.0083	0.0021	0.0006	0.0001	0
	Spread	0.3152	0.3274	0.3372	0.3387	0.3400

Table 4.8: Performance metrics by fa and k for $P := \{tt, pr, nt, wd\}$

Mode choice fa	Metric	Sampling density k				
		8	16	32	64	128
all	IGD	0.0082	0.0025	0.0010	0.0005	0.0002
	Spread	0.3156	0.3246	0.3309	0.3335	0.3346
ptOnly	IGD	0.0140	0.0049	0.0013	0.0008	0.0005
	Spread	0.3662	0.3780	0.3853	0.3893	0.3926
noFlights	IGD	0.0107	0.0031	0.0014	0.0014	0.0011
	Spread	0.3047	0.3192	0.3261	0.3283	0.3290

F Aggregated relationships for different traveler preferences

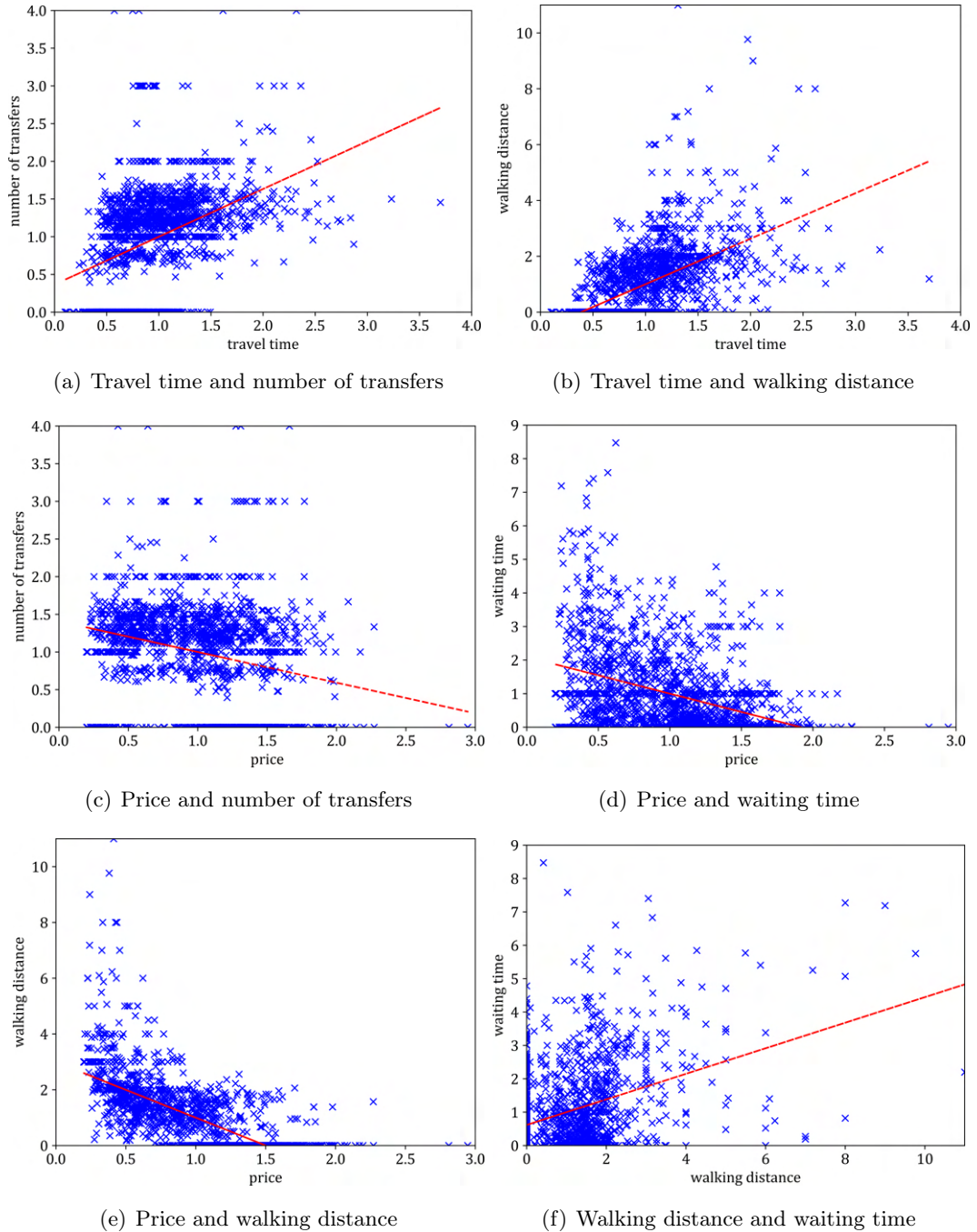


Figure 4.18: Relationship for different traveler preferences

Chapter 5

Predicting Pareto fronts to guide smart sampling of multimodal itineraries

Abstract

In recent years, the significance of decision support for the planning of multimodal itineraries has grown. Travelers expect the integration of multiple individual preferences such as travel time, price, and the number of transfers into the search. As these preferences feature competing characteristics, multiple Pareto-optimal itineraries arise that represent trade-offs. A major challenge is to find reasonable, Pareto-optimal sets of itineraries in efficient run time when multiple traveler preferences are taken into account in a large multimodal network.

In this work, we approximate the set of Pareto-optimal itineraries. In particular, we learn and predict structures of the multimodal Pareto front from search data collected by integrated mobility platforms. The predicted structure of the Pareto front then serves to guide the approximation of the set of Pareto-optimal itineraries by focusing on more relevant areas of the multimodal solution space. To this end, we compare four approaches to predict Pareto front structures as well as three different approaches to apply these to Pareto front approximation in our smart sampling framework. The framework is evaluated with a substantial amount of real-world data on mobility services. We examine long-distance trips between major cities in Germany, taking up to five most prevalent traveler preferences into account. We observe that integrating Pareto front structures into the search accurately guides sampling, while the efficiency of the search is highly dependent on origin and destination.

Keywords Routing, Multi-Criteria Decision Support, Multimodal Mobility, Pareto Front Approximation, Pareto Front Prediction

Contents

5.1	Introduction	135
5.2	Related literature	137
5.2.1	Traveler requirements at multimodal route planning	137
5.2.2	Algorithms for multimodal route planning	138
5.2.3	Comparison of Pareto-optimal sets	139
5.3	Smart sampling framework	140
5.3.1	Problem description	140
5.3.2	Framework	141
5.3.3	Prediction of Pareto front structures	144
5.3.4	Transformation techniques	148
5.3.5	Multimodal network model and routing	150
5.4	Computational design	151
5.4.1	Experimental setup	151
5.4.2	Metrics	155
5.4.3	Configuration of prediction approaches	156
5.5	Computational results	157
5.5.1	OD-specific example	157
5.5.2	Prediction of structures of the Pareto front	159
5.5.3	Transformation techniques	162
5.5.4	Random forest prediction with Euclidean distance transformation	163
5.6	Conclusion	170
	Appendices	172
A	Mode choice options	172
B	Impact of Wasserstein distance on improved solutions	172

5.1 Introduction

The combination of several mobility services is known as multimodal mobility (Lyons et al., 2020; Willing et al., 2017). The importance of decision support for the planning of multimodal itineraries has grown in recent years. Integrated mobility platforms like *GoogleMaps*, *omio*, and *Rome2rio* promise that travelers can plan door-to-door itineraries taking into account their individual preferences (Esztergár-Kiss et al., 2020; Molenbruch et al., 2021). When multiple traveler preferences are taken into account simultaneously, several Pareto-optimal travel itineraries result due to conflicting preferences. For instance, travelers may be willing to accept a slightly higher fare to achieve a significantly reduced travel time. These Pareto-optimal itineraries form the choice set that can be presented to the traveler to enable informed decision-making (Esztergár-Kiss et al., 2020; Lyons et al., 2020).

Finding the full Pareto-optimal set of itineraries integrating multiple traveler preferences in a multimodal context is still challenging. Modeled as a Multi-Criteria Decision Making (MCDM) problem, Delling et al. (2013a) and Dib et al. (2017) highlight that taking more than three traveler preferences simultaneously into account has an enormous impact on the run time. Combining multiple mobility services in a large area network increases the level of complexity further. Since run time is critical for mobility platforms, multimodal search algorithms should scale efficiently in relation to the number of traveler preferences integrated (Bast et al., 2015; He et al., 2022).

To achieve a scalable algorithm, we have introduced a systematic sampling framework approximating the Pareto-optimal set of itineraries in our previous paper Horstmannshoff & Ehmke (2022). We assumed that the traveler expresses individually relevant preferences, and all preferences are equally important. Based on general information provided by the traveler about the origin, destination, earliest departure time, and further individually relevant preferences, a set of Pareto-optimal itineraries is approximated through many iterations of uniform solution sampling. The systematic sampling framework scales well when additional preferences are integrated into the search. We examined five traveler preferences simultaneously while creating sets of multimodal itineraries. We have observed efficient scaling for two-dimensional sampling. However, extending the search to three-dimensional sampling increased the quality of the Pareto-optimal set of multimodal itineraries at the cost of a considerable

increase in run time. We also found similarities between the different preferences but figured out that these depend highly on the underlying origin-destination-combinations (OD) combination. While the traveler has to invest quite a significant amount of time before major cost savings occur for some ODs, these arise swiftly for other ODs.

In this work, we aim to make our systematic sampling framework smarter by exploring relationships between OD pair characteristics and Pareto front structures. The core idea is to learn and predict the structures of multidimensional Pareto fronts from search data collected by integrated mobility platforms using quickly calculable attributes of a new traveler request (e.g. maximum travel time, Haversine distance, etc.). To this end, we propose and compare different approaches for predicting relationships between OD pair characteristics and Pareto front structures for a new traveler request. The predicted relationships are integrated into the sampling framework to enhance the approximation of Pareto-optimal itineraries by focusing on more relevant areas of the multimodal solution space. We compare four different approaches to predict relationships between OD pair characteristics and Pareto front structures for a new traveler request as well as three different transformation techniques to integrate the predicted relationships into the sampling framework. The resulting choice set can be presented to the traveler as an expert menu in the integrated mobility platform, including additional information about the choice set's characteristics and predicted relationships between OD pair characteristics and Pareto front structures. Hereby, we enable the traveler to take well-informed decisions. The integration of predicted relationships into the sampling framework is evaluated in a proof-of-concept study using a large amount of real-world data of multiple mobility services for long-distance trips between major cities in Germany, taking up to five preferences into account. This includes a comparison with the results of our prior paper.

An overview of related work on traveler-oriented route planning, multimodal route planning, and approaches to compare Pareto-optimal sets is presented in Section 5.2. Then, in Section 5.3, we introduce the smart sampling framework and discuss how we integrate the predicted relationships into the process of sampling. Section 5.4 introduces the considered data on mobility services, traveler preferences, the framework's settings as well as the relevant metrics for evaluation. In Section 5.5, we analyze the results in terms of prediction accuracy and the impact of the integration

on the efficiency and informativeness of the sampling framework. Finally, we identify areas for further research in Section 5.6.

5.2 Related literature

MCDM is a well-established research field with numerous papers and practical applications. First, we discuss traveler requirements when planning multimodal itineraries in Section 5.2.1. Then, we present an overview of routing algorithms for multimodal route planning in Section 5.2.2. As we take advantage of predicted Pareto front structures in our smart sampling framework, we require approaches to compare Pareto-optimal sets with each other. Related literature is presented in Section 5.2.3.

5.2.1 Traveler requirements at multimodal route planning

Multiple research studies have revealed that traveler orientation is an essential part in the acceptance of multimodal mobility. To enable the traveler a one-stop search for door-to-door travel, all available private and public mobility services have to be accessible (Esztergár-Kiss & Kerényi, 2020; Schulz et al., 2020; Valderas et al., 2020). Grotenhuis et al. (2007) highlight that the search and selection of multimodal itineraries across several mobility services and providers is simplified for travelers if they can use an integrated mobility platform. This prevents travelers from having to manually combine and compare the results of different mobility platforms, which would lead to a considerable additional cognitive and time effort. The integration of multiple available mobility services includes information on well-established services such as scheduled trains and buses, as well as the integration of innovative sharing services such as car, bike, and ride-sharing. In particular, the integration of sharing services allows the first and last mile of the respective itinerary to be taken into account (Lyons et al., 2020; Stopka, 2014). A major challenge is the consideration of the heterogeneous characteristics of mobility services. While public transport, for example, operates on timetables, many ride-sharing providers work in a dynamic environment that has to be evaluated on demand.

Furthermore, travelers require multi-criteria decision support in the search for and selection of possible multimodal itineraries. For the most part, not only one preference is relevant for the traveler (Lyons et al., 2020; Musolino et al., 2023; Spickermann

et al., 2014), but the simultaneous consideration of multiple competing preferences is required. For example, travelers may be willing to invest a little more travel time in order to obtain a significantly lower fare. This results in a set of relevant Pareto-optimal itineraries which form the choice set for the traveler. Horstmannshoff (2022) presents a thorough literature overview of relevant traveler preferences. Among others, these are travel time, price, number of transfers, waiting time, walking distance, reliability, and sustainability.

In this work, we consider prevalent traveler preferences to approximate a set of alternative Pareto-optimal door-to-door itineraries. In addition, by providing insights into the complex multimodal solution space, we make solution space characteristics more transparent to the traveler.

5.2.2 Algorithms for multimodal route planning

As we integrate multiple traveler preferences into the search for multimodal itineraries, we are dealing with methods of MCDM, in particular those for multimodal route planning. For a detailed overview of algorithms for route planning and how to extend these into a multimodal setting, we refer to Bast et al. (2015). In the following, we present details of the most related algorithmic approaches.

Delling et al. (2013a) propose a label-based itinerary planning algorithm using public transportation route planning approaches and iteratively increase the number of possible transfers when calculating all relevant itineraries. They transfer the RAPTOR algorithm by Delling et al. (2012) to a multimodal context (MCR) and retrieve a Pareto-optimal set while reducing the computational time by considering preferences such as travel time, price, and traveler convenience. In addition, they improve the efficiency of their algorithm by utilizing Contraction Hierarchies (CHs) to preprocess the network graph. Dib et al. (2017) provide a label-based multi-criteria routing algorithm which takes travel time, total walking time and number of transfers into account. Bozyigit et al. (2017) extend Dijkstra’s method to consider walking distance as well as the number of transfers. As a result, they include a penalty rule in Dijkstra’s algorithm. Potthoff & Sauer (2022) extend existing efficient bimodal algorithms by excluding undesirable itineraries. Further relevant multimodal route planning approaches, which operate directly on the integrated mobility services’

timetable are, e.g., user-constrained CHs (UCCH) (Dibbelt et al., 2015), trip-based routing (Witt, 2015), and the connection scan algorithm (Dibbelt et al., 2018).

Most of the algorithms introduced above account for only two or three preferences at the same time. Delling et al. (2013a), Dib et al. (2017), and Potthoff & Sauer (2022) highlight that the run time of their approaches deteriorates significantly when additional preferences are taken into account. They encounter scaling issues when considering an additional traveler preference while identifying the full Pareto-optimal set of itineraries.

Delling et al. (2019) introduce the concept of restricted Pareto sets. They limit the number of examined Pareto-optimal itineraries during the search by proposing rules to avoid retrieving undesirable itineraries for the traveler. This results in faster run times of the applied routing algorithm as well as limiting the size of the retrieved Pareto-optimal set of itineraries in their multi-criteria problem setting applied in an unimodal network. A major issue is that they limit the search to itineraries which are close to the optimal travel time. Another approach to ensure efficient scaling algorithms with respect to the number of traveler preferences is the approximating of the set of Pareto-optimal itineraries. While this enables efficient scaling, it cannot guarantee that the full Pareto-optimal set can be found. Herzel et al. (2021) state in their survey of approximation methods for multi-objective problems that the majority of the algorithms studied have underlying assumptions of at most three objectives. In our previous paper Horstmannshoff & Ehmke (2022), we have proposed a systematic sampling framework. The core idea is to decompose the high-dimensional problem setting into several problems of smaller dimension, thereby ensuring efficient sampling. In this paper, we tackle a significant shortcoming of the systematic sampling framework. Instead of sampling uniformly over the multimodal solution space, we now focus on promising areas of the solution space. These areas are identified by learning and predicting structures of multidimensional Pareto fronts from search data.

5.2.3 Comparison of Pareto-optimal sets

To predict similar Pareto fronts, our smart sampling framework requires that we measure the proximity between two sets of Pareto-optimal solutions. Multiple MCDM metrics have been proposed to evaluate the quality of Pareto fronts such as hypervolume and spread metric (Deb, 2011; Riquelme et al., 2015). Cao et al.

(2015) refine the Hypervolume metric to compare Pareto fronts. They highlight that the choice of the reference point has a major impact on the calculation of the hypervolume metric and provide rules for choosing appropriate reference points. Candelieri et al. (2022) emphasize that the Wasserstein distance, also referred to as the optimal transport distance, allows to compare distributions without the need for further parameterization. In this work, we use the Wasserstein distance to measure the proximity between two Pareto-optimal sets of solutions to avoid any impact of parameterization on the results.

5.3 Smart sampling framework

In the following, we present our smart sampling framework that takes advantage of predicted Pareto fronts of multiple traveler preferences. The main objective is to enhance the sampling process by learning and predicting structures of the Pareto front for a new traveler request. Using the predicted Pareto front structures, we guide the systematic sampling framework introduced in our previous paper to more promising areas of the complex multimodal solution space. In the following, we refer to the framework outlined in the previous paper as *systematic sampling*, while we refer to the framework presented here as *smart sampling*.

Building on the problem description (Section 5.3.1), we recapitulate the systematic sampling framework of our previous paper and introduce the enhanced smart sampling framework for approximating the Pareto-optimal choice set in Section 5.3.2. In particular, we discuss several approaches to learn and predict structures of the Pareto front (Section 5.3.3) as well as transform these into the smart sampling framework (Section 5.3.4). Finally, we introduce the multimodal network model and the multimodal routing algorithm (Section 5.3.5).

5.3.1 Problem description

In our a posteriori smart sampling framework, we require the individual traveler origin O and destination D , the earliest departure time t_{dep} , and a set of traveler preferences P as input parameters stated by the traveler. The set $P := \{P_1, \dots, P_p\}$ contains p traveler preferences such as travel time, price, number of transfers, etc. Knowledge of these parameters is a common assumption for a posteriori settings.

Furthermore, a set of ms integrated mobility services $MS := \{MS_1, \dots, MS_{ms}\}$ such as railway and flights is considered. The traveler might restrict the set of considered mobility services, e.g. exclude driving by car. These mode choice restrictions are modeled by finite automata fa . The set of mobility services MS and the applied finite automata fa are integrated into the multimodal network model, which is described in detail in Section 5.3.5.

Given this information, we approximate the corresponding Pareto front by identifying the set of Pareto-optimal itineraries S_{all}^{opt} . In detail, we predict the structure of the Pareto front and transform the prediction into a sampling instruction. During the sampling procedure, we rerun a multimodal routing algorithm SPM multiple times. We create several sets of solutions for two and three dimensions. Here, the term dimension refers to how many traveler preferences are taken into account simultaneously when creating the respective sets. For instance, two-dimensional sampling means that we consider two preferences at the same time when approximating the Pareto front in the respective set, whereas, in three-dimensional sampling, we consider three preferences at the same time. The number of considered preferences in set P is independent of this. We will repeat sampling pairwise (for two-dimensional sampling) or threefold (for three-dimensional sampling), respectively, to approximate sets of solutions.

5.3.2 Framework

Algorithm 12 shows the basic procedure and components of our smart sampling framework. The lines 3 and 4 indicate the new features of predicting Pareto front structures and transforming these into a sampling instruction, while the features in lines 1, 2, 5, 7 and 8 follow the structure of our systematic sampling framework.

First, we introduce the *systematic sampling framework*. Figure 5.1 outlines the basic principle in case three preferences $P := \{\text{travel time, price, number of transfers}\}$ are considered. Based on the given information by the traveler, we identify a min-max-interval $[l_i, u_i]$ for each preference $i \in P$ resulting in a set of intervals I (line 1 in Algorithm 12). The intervals are determined by a single-objective run of the multimodal routing algorithm SPM for each traveler preference and then setting the obtained minimum and maximum values of the respective preferences as the interval limits. This ensures that only reasonable parts of the multimodal solution

Algorithm 12 Smart solution space sampling framework algorithm

```

1:  $I \leftarrow \text{IdentificationOfMinMaxIntervals}(O, D, t_{dep}, P)$ 
2: for all  $i, j | i \neq j \in P$  do ▷ Parallelized Execution
3:    $PPF \leftarrow \text{PredictionOfParetoFrontStructures}(O, D, t_{dep}, i, j)$  (details in 5.3.3)
4:    $TF \leftarrow \text{TransformationOfPredictedParetoFrontIntoSamplingProcedure}(PPF)$ 
     (details in 5.3.4)
5:    $S_{i,j} \leftarrow \text{SamplingWithinMinMaxIntervals}(O, D, t_{dep}, I, P, k, i, j, TF)$ 
6: end for
7:  $S_{all} = \bigcup_{i,j | i \neq j \in P} S_{i,j}$ 
8:  $S_{all}^{opt} \leftarrow \text{RemovalOfDominatedSolutions}(S_{all})$ 
    
```

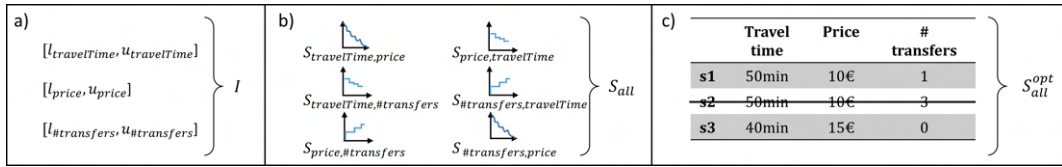


Figure 5.1: Systematic sampling framework

space are investigated. Figure 5.1(a) presents an example for the set of intervals I . Next, we approximate the set of Pareto-optimal itineraries by many iterations of sampling uniformly over the solution space. To this end, we compute several two-dimensional sets $S_{i,j}$ simultaneously (line 2), thereby reducing the computational complexity of calculating each of these sets. Following the ϵ -constraint method, we set one preference $i \in P$ as a single objective while the upper-bound constraint of another preference $j \in P$ is systematically altered (line 5). The computation of the respective sets can be parallelized. This guarantees scalability even for a larger number of preferences since only the number of parallelly determined sets grows, but not the complexity of calculating these two-dimensional sets. Then, the two-dimensional sets $S_{i,j}$ are merged together in a joint set $S_{all} := \bigcup_{i,j | i \neq j \in P} S_{i,j}$ (line 7 in Algorithm 12). Figure 5.1(b) shows the six created two-dimensional sets. Finally, as depicted in Figure 5.1(c), dominated solutions are removed from S_{all} as they are irrelevant to the traveler (line 8). The remaining solutions form the set of Pareto-optimal solutions S_{all}^{opt} , which shape the travelers' choice set. For a comprehensive introduction to the systematic sampling framework (without Pareto front prediction) including detailed pseudo codes, we refer to our previous paper Horstmannshoff & Ehmke (2022).

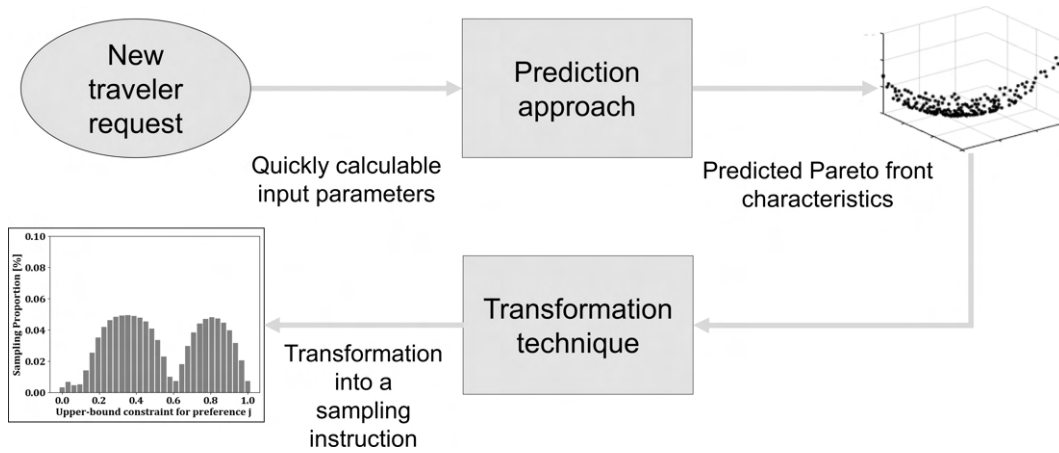


Figure 5.2: Smart sampling for each two-dimensional set

We have observed that uniformly distributed sampling across the multimodal solution space can create inefficiencies as there is no focus on relevant regions. Hence, we propose learning and predicting Pareto fronts from OD pair characteristics and transforming these into a sampling instruction to guide the search to these promising areas and thereby making our sampling framework more effective. This applies especially to preferences that are not subject to an integer constraint, e.g., price, travel time, waiting time. Figure 5.2 depicts the core idea of our *smart sampling framework*. For each two-dimensional set $S_{i,j}$, we use quickly calculable attributes of a new traveler request as input parameters for a prediction model. These input parameters are 1) the respective min-max-interval boundaries, 2) the Haversine distance between the origin and destination, and 3) the airport availability at the respective origin and destination. The objective of the prediction model is to predict the true – but actually unknown – Pareto front as accurately as possible. Therefore, we predict a Pareto front known from OD pair characteristics and historical search data (line 3 in Algorithm 12). For instance, in case we want to predict the Pareto front for a request from Berlin to Hamburg, our prediction model indicates – based on OD pair characteristics – that the Pareto front approximation already known for Leipzig to Munich is a promising candidate. A detailed discussion of the prediction approaches is given in Section 5.3.3.

Next, we take advantage of the predicted Pareto front structures and transform these into a sampling instruction to guide the sampling procedure (line 4). Thereby,

we guide the sampling framework to more promising areas of the multimodal solution space and hence enhance the approximation of Pareto-optimal solutions. A detailed discussion of the investigated transformation techniques is presented in Section 5.3.4.

We also investigate our smart sampling framework for three dimensions aiming at identifying additional solutions which cannot be found in a two-dimensional setting. This results in multiple three-dimensional sets $S_{i,j,h}^3$ for each considered preference $i, j, h | i \neq j \wedge i \neq h \wedge j < h \in P$. In this case, we set traveler preference i as the objective, and systematically alter the preferences j and h . The extension of the smart sampling framework to a three-dimensional setting works analogously to the two-dimensional setting. The prediction of Pareto front structures is conducted in three dimensions and then transformed into sampling instructions for a guided search on preferences j and h , respectively, whose upper-bound constraints are altered during the search.

5.3.3 Prediction of Pareto front structures

In the following, we introduce four approaches to predict Pareto front structures. Given the traveler’s origin O and destination D , the earliest departure time t_{dep} as well as information about the examined two-dimensional set $S_{i,j}$ (i indicates the preference set as single objective, j the preference whose upper-bound constraint is altered during sampling), we predict the Pareto front PPF as accurately as possible.

In general, we distinguish two types of approaches for predicting Pareto front structures. For the first, we use only the structure of the input parameters for prediction (“Greedy nearest neighbor” and “Clustered attributes”). For the second, we cluster similar Pareto fronts. To predict these clusters with their representative medoid, we build a prediction model that utilizes the input parameters for learning and prediction (“Decision tree” and “Random forest”). In the following figures, we depict the input parameters by a cross, while Pareto fronts are depicted by a cube. We present historical search data in grey, the information of the traveler’s new request in black, and the predicted information in blue.

For clustering Pareto fronts, we require a metric to quantify the proximity of two Pareto fronts. We use the Wasserstein distance, which works as follows. The smaller this value, the “closer” the respective sets are to each other. Figure 5.3 presents examples for the Wasserstein distance. The two sets in Figure 5.3(a) are identical

and thus have a Wasserstein distance value of 0. The two sets shown in Figure 5.3(b) differ only slightly from each other. The respective Wasserstein distance value is 0.01. Finally, when two sets differ significantly from each other, the respective Wasserstein distance value increases significantly as shown in Figure 5.3(c) (1.55).

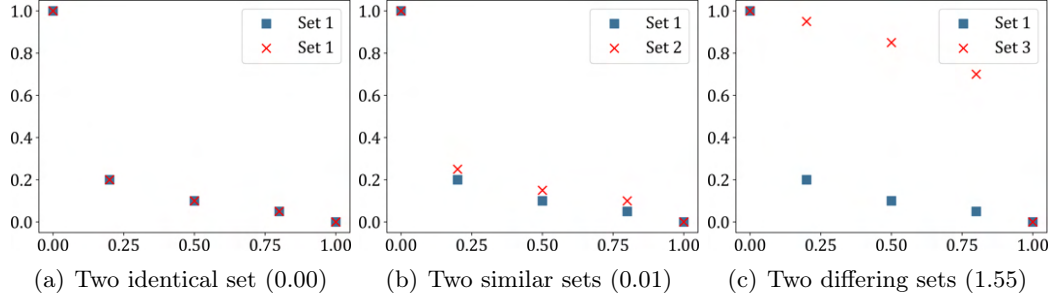


Figure 5.3: Wasserstein distances

To ensure comparability between the input parameters, we apply 0-1-normalization. For 0-1-normalization, the minimum value for the respective parameter is 0 and the maximum value equals 1. The normalized value x'_i of parameter value of x_i with x_{min} as the minimum value for x and x_{max} as the maximum value for x is calculated as:

$$x'_i = \frac{x_i - x_{min}}{x_{max} - x_{min}}. \quad (5.1)$$

For some approaches, we work on the closeness between normalized input parameters of the historical search data using the Euclidean distance. The Euclidean distance d_{ab} between the n input parameters of data set $a = (a'_1, \dots, a'_n)$ and $b = (b'_1, \dots, b'_n)$ is calculated as:

$$d_{ab} = \sqrt{\sum_{i=1}^n (a'_i - b'_i)^2}. \quad (5.2)$$

5.3.3.1 Greedy nearest neighbor

In the “Greedy nearest neighbor” approach, we use the structure of the input parameters for predicting Pareto front structures. As shown in Figure 5.4, based on

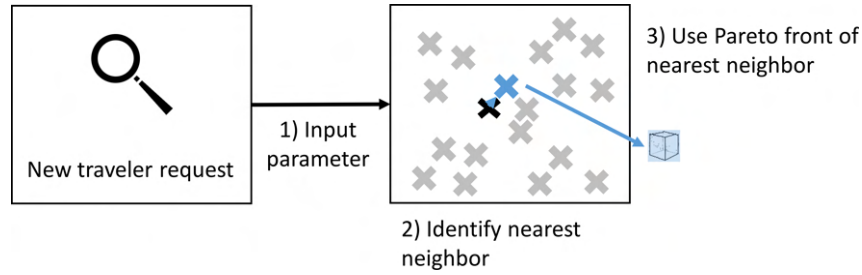


Figure 5.4: Prediction approach: Greedy nearest neighbor

the normalized input parameters of the new traveler request, we identify the nearest neighbor out of the set of historical search data with its minimum Euclidean distance in terms of the normalized input parameters (indicated by a blue cross in Figure 5.4). Then, we take the respective Pareto front of the identified nearest neighbor as the predicted Pareto front.

5.3.3.2 Clustered attributes

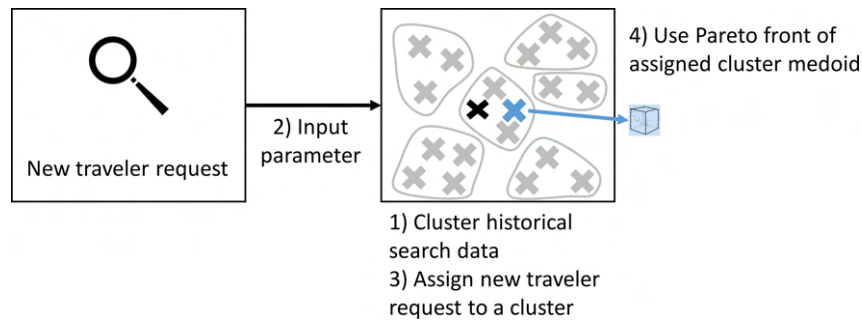


Figure 5.5: Prediction approach: Clustered attributes

Similar to the previous approach, the “Clustered attributes” approach uses the input parameters for prediction. The key difference is that we now create clusters on the set of historical search data (training set) based on their normalized input parameters (Step 1 in Figure 5.5). For clustering, we apply the k -medoids approach as a classical partitioning technique to determine a medoid included in the training data set for each identified cluster. In the next step, based on the normalized input parameters for the new traveler request, we identify the “closest” medoid of all clusters

in terms of the Euclidean distance (Steps 2 and 3 in Figure 5.5) and use the medoid's Pareto front as the predicted Pareto front (Step 4).

5.3.3.3 Decision tree

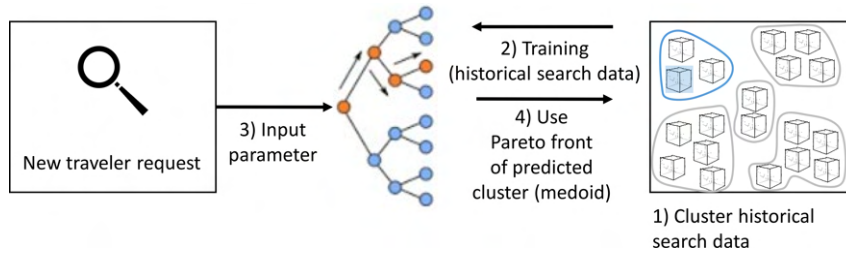


Figure 5.6: Prediction approach: Decision tree

Next, we introduce the “Decision tree” approach. Compared to the two approaches discussed above, we now focus on the Pareto front structures of historical search data to predict a Pareto front, see Figure 5.6. In the first step, we cluster Pareto fronts of the historical search data using the Wasserstein distance as the similarity measure unit (Step 1). We create clusters using the k -medoid partition technique to identify representative Pareto front structures (Kaufman & Rousseeuw, 2009). A reasonable number of clusters is determined during a preceding parameter tuning phase as discussed in Section 5.4.3. Then, we train a decision tree as a prediction model (Step 2 in Figure 5.6) to predict these clusters with their representative medoids using the normalized input parameters for learning and prediction (Pochiraju & Seshadri, 2019). Finally, based on this input, we predict a cluster using the trained decision tree. The medoid of the predicted cluster will then be used to guide the sampling procedure (Steps 3 and 4).

5.3.3.4 Random forest

Finally, we use “Random forest” to predict the structures of the Pareto front. The use of a random forest model generally results in higher accuracy compared to a decision tree model, but its overall interpretability is more difficult (Pochiraju & Seshadri, 2019). The overall procedure is depicted in Figure 5.7. This approach works essentially analogous to the “Decision tree” approach. The main difference is that we use a random forest model to predict the cluster of similar Pareto fronts instead of a

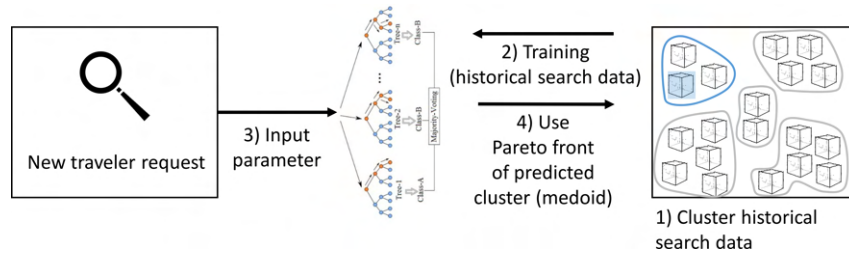


Figure 5.7: Prediction approach: Random forest

decision tree. Based on the clustering of similar Pareto fronts by their Wasserstein distance, we predict the appropriate cluster using the normalized input parameters of a new traveler request. The sampling process will then be guided by the medoid’s Pareto front of the predicted cluster.

5.3.4 Transformation techniques

In the next step, we introduce three approaches to transform the predicted Pareto front structures for a new traveler request PPF into the smart sampling framework. Applying the analyzed transformation techniques yields a sampling instruction TF , which is then used to guide the sampling procedure to promising parts of the multimodal solution space. All approaches are illustrated using the following exemplary coordinates of the predicted Pareto front: $\{(0, 1), (0.1, 0.5), (0.6, 0.1), (1, 0)\}$.

5.3.4.1 Euclidean distance

First, we introduce the “Euclidean distance” approach. As depicted in Figure 5.8(a), we calculate for each area the respective Euclidean distance between two predicted Pareto-optimal solutions. Table 5.1 presents the Euclidean distance for the exemplary coordinates. For instance, for the first area, a distance of $\sqrt{(0 - 0.1)^2 + (1 - 0.5)^2} = 0.5099$ is calculated. Subsequently, the proportional share of the interval ranges to the total sum of the respective Euclidean distances is determined, e.g. $\frac{0.5099}{0.5099 + 0.6403 + 0.4123} = 0.326$ for the first area. This determines that 32.6% of the sampling steps will occur uniformly distributed across the first area.

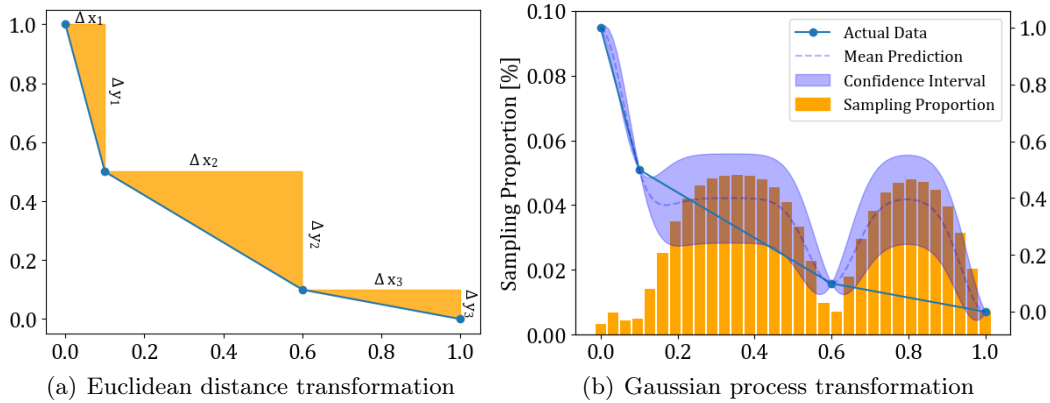


Figure 5.8: Transformation techniques

Table 5.1: Exemplary transformation for Euclidean and Product distance

	Euclidean distance transformation		Product distance transformation	
	Euclidean Distance	Proportion Sampling [%]	Product Distance	Proportion Sampling [%]
Area 1	0.5099	32.6	0.05	17.2
Area 2	0.6403	41.0	0.20	69.0
Area 3	0.4123	26.4	0.04	13.8

5.3.4.2 Product distance

Applying the “Product distance” approach, we calculate the respective distance for each area by the absolute value of the respective coordinate product as presented in Table 5.1. As an example, the product distance for the first area is calculated by $\left| \left(0 - 0.1 \right) * \left(1 - 0.5 \right) \right| = 0.05$. Afterwards, the proportional share of the interval ranges to the total sum of the respective product distances is calculated, which determines the respective proportion of sampling steps in each area. For instance, for the first area, we guide the sampling to process $\frac{0.05}{0.05 + 0.2 + 0.04} = 0.172 \rightarrow 17.2\%$ of the overall sampling steps at that area. Compared to the “Euclidean distance” approach, the “Product distance” focuses on areas that are characterized by large leaps in the x and/or y-domain.

5.3.4.3 Gaussian process

Finally, we present the “Gaussian process” transformation technique (Rasmussen & Williams, 2006; Schulz et al., 2018). Applying the Euclidean distance and the Product distance yield uniformly distributed sampling within the areas between two neighboring predicted Pareto-optimal solutions. Using Gaussian process regression, we aim to account for different levels of uncertainty even within an area between two neighboring solutions. The core idea is to learn the complex structure of the function describing the Pareto front taking the given observations by the predicted Pareto front into account. We learn possible functions which explain the structure. Then, areas with the highest uncertainty in the multidimensional solution space are identified. We focus on sampling in these areas. As shown in Figure 5.8(b), we first learn a function for the current observations (blue dotted line) and then determine the confidence interval (shaded blue area) of the learned function. Finally, we guide the sampling procedure to areas with the highest uncertainty as indicated by the orange highlighted bars.

5.3.5 Multimodal network model and routing

Following Pajor (2009), we create a multimodal network model G integrating multiple mobility services. First, we generate a unimodal network $G_{ms} = (V_{ms}, A_{ms})$ for each considered mobility service $ms \in MS$ modeled as a time-expanded network. The set of vertices V_{ms} represents arrival and departure events, whereas the set of arcs A_{ms} models a valid trip/subsequence of a trip between two vertices, which are assigned to the same trip.

Next, we merge all unimodal networks into one multimodal network $G = (V, A)$. The set of vertices (arcs) is created by merging all unimodal sets V_{ms} (A_{ms}) with $ms \in MS$ into one set V (A). In addition, we add transfer and link arcs. Transfer arcs represent feasible transfers between the same mobility service at the same stop location at different arrival and departure events. Link arcs provide transfers between different mobility services, which are in a given walking distance wd^{max} assuming a predefined walking speed ws . To model shortcuts in the network between relevant vertices and thus speed up the optimization, we apply the concept of CH (Geisberger et al., 2008). In terms of integrating car usage into the network, we assume that an

itinerary cannot be partially driven by car. Therefore, we add an arc for car usage from O to D .

Finally, to avoid unrealistic solutions, e.g., taking a train in between two car sections, we use non-deterministic finite automata fa , which represent the travelers' mode choice in the network model (Bast et al., 2015; Pajor, 2009). The core idea is that all solutions retrieved by SPM have to fulfill the solution structure as defined by fa . Building on the multimodal network as well as the finite automata, we create a product network $G^\times = (V^\times, A^\times)$.

For identifying a traveler's choice set, we rerun a label-constrained multimodal shortest-path algorithm SPM as presented by Pajor (2009) multiple times. This algorithm is a variant of Dijkstra's shortest path algorithm integrating finite automata. We extend the given algorithm so that we can consider several constraints in parallel given a product network G^\times and a finite automata fa . The proposed framework's scalability is guaranteed independently of the multimodal routing algorithm implemented or the underlying network. For an in-depth explanation of creating the multimodal network and the resource-constrained multimodal shortest-path algorithm including a detailed pseudo code, we refer to our previous paper Horstmannshoff & Ehmke (2022).

5.4 Computational design

In the following, we introduce the computational design for the evaluation of the presented smart sampling framework. First, we provide details about the experimental setup (Section 5.4.1). Then, we introduce the metrics applied for evaluation (Section 5.4.2). Finally, we discuss the configuration of prediction approaches (Section 5.4.3).

5.4.1 Experimental setup

For evaluation, we examine 300 ODs between the main railway stations of major cities in Germany. For an individual traveler request, we assume an earliest departure time t_{dep} at 9am on October 8, 2018. We have chosen this regular work day since data for all mobility services were available for this day. The resulting multimodal network $G = (V, A)$ consists of roughly 40,000 vertices and 10,000,000 arcs.

The set of mobility services MS is built on a large amount of real-world data. We use available General Transit Feed Specification (GTFS) data for German Railways, Flixbus, and local transit services. Furthermore, we integrate flights and long-distance ridesharing services like “BlaBlaCar” based on real-world data collected from publicly available service data. Hereby, we consider major airports with more than 50,000 aircraft movements per year (Berlin-Schönefeld, Cologne/Bonn, Berlin-Tegel, Dusseldorf, Frankfurt am Main, Munich, Stuttgart, Hamburg, Hanover, Leipzig/Halle, Nuremberg). In addition, we presume that the traveler has to arrive at the designated airport an hour before the flight’s departure and that they require 15 minutes after arrival to leave the airport. We integrate information on the road network using the open-source routing library GraphHopper (www.graphhopper.com). For a more realistic estimation of travel times during peak hours, we multiply the retrieved travel times by 1.25. We assume a walking speed ws of $5 \frac{km}{h}$ and set the maximum walking distance wd^{max} to $0.5km$.

We estimate the fare information for German Railways and Flixbus as these are not included in the GTFS data. For German Railways, the estimation is based on the train type chosen as well as the distance covered with the respective train type. We consider three different types of trains, namely *regional trains* (slowest train), *intercity trains* as well as *intercity express trains* (fastest train). We assume that intercity trains are 50% more expensive and intercity express trains are 100% more expensive in comparison to regional trains. Based on preliminary empirical investigations on www.bahn.de, the price for regional trains will be €17 per 100-kilometer distance traveled. For Flixbus, we assume €10 per 100-kilometer distance traveled. For individual road mobility, we assume 30 cents per kilometer following the flat-rate depreciation allowance in the German tax system. Note that all these values are only rough estimates and can be adapted as needed in a real-world scenario to, e.g., incorporate tariff fare structures and discount tickets (Schöbel & Urban, 2022; Timm & Storandt, 2020). The prices for flights are based on real data that has been sampled for the respective day of the experiment.

To evaluate the impact of integrating predicted relationships into the smart sampling framework, we compare four different approaches which have been introduced in Section 5.3.3:

- *Greedy nearest neighbor*: Use the Pareto front structure of the nearest neighbor in terms of the normalized input parameters (details in Section 5.3.3.1).
- *Clustered attributes*: Use the Pareto front structure of the closest medoid out of clustered historical search data (details in Section 5.3.3.2).
- *Decision tree*: Train a decision tree model to predict clustered Pareto front structures of historical search data by their Wasserstein distance. Use the model to assign a cluster for a new traveler request and use the cluster's medoid as the predicted Pareto front structure (details in Section 5.3.3.3).
- *Random forest*: Train a random forest model to predict clustered Pareto front structures of historical search data by their Wasserstein distance. Use the model to assign a cluster for a new traveler request and use the cluster's medoid as the predicted Pareto front structure (details in Section 5.3.3.4).

In addition, we analyze the impact of the following techniques to transform the predicted structure of the Pareto front into the smart sampling framework introduced in detail in Section 5.3.4:

- *Euclidean distance*: Determine the sampling rate at the respective part of the predicted Pareto front by means of the proportional share of the Euclidean distance (details in Section 5.3.4.1).
- *Product distance*: Determine the sampling rate at the respective part of the predicted Pareto front by means of the proportional share of the Product distance (details in Section 5.3.4.2).
- *Gaussian process*: Determine the sampling rate based on the uncertainty retrieved by Gaussian process regression (details in Section 5.3.4.3).

We will analyze the impact of the parameters *set of considered traveler preferences*, *sampling density*, as well as the *travelers' mode choice* for all combinations of prediction and transformation techniques as follows. The different sets of traveler preferences P are defined by *travel time* (tt), *price* (pr), *number of transfers* (nt), *overall walking distance* (wd), and the overall *waiting time* (wt). We examine the following combinations:

- tt, pr
 $P := \{\text{travel time, price}\}$
- tt, pr, nt
 $P := \{\text{travel time, price, number of transfers}\}$
- tt, pr, nt, wd
 $P := \{\text{travel time, price, number of transfers, walking distance}\}$
- tt, pr, nt, wd, wt
 $P := \{\text{travel time, price, number of transfers, walking distance, waiting time}\}$

In the first set, we only consider travel time and price, which are perceived as essential decision criteria for the traveler (Grotenhuis et al., 2007). Then, we analyze the impact of integrating additional preferences, which are of high importance for the traveler, into the search (Lyons et al., 2020; Musolino et al., 2023; Spickermann et al., 2014). Further preferences which can be modeled as cost values of an arc in the underlying multimodal network can be integrated easily into the search.

In addition, we evaluate the effect of different granularities of sampling on the quality of the Pareto front approximation, which is controlled by the sampling density parameter k . We examine the impact of setting k to 8, 16, 32, 64 and 128, respectively. This provides information about whether investing additional effort into the search pays off.

Furthermore, we examine the impact of different travelers' mode choice options to avoid unrealistic solutions. We evaluate three cases: (1) no mode choice restrictions, (2) using public transportation only, (3) allowing all services except flights. These mode choice options are integrated by non-deterministic finite automata fa as introduced in Section 5.3.5. The first finite automaton fa_{all} allows the usage of all integrated mobility services. The corresponding transition graph is shown in the Appendix in Figure 5.17. The finite automaton fa_{ptOnly} shown in the Appendix, Figure 5.18 limits the mobility offers to public transportation services including the walk to the first visited public transportation vertex and from the last visited public transportation vertex. Transferring between two public transportation services is modeled through arcs labeled as *link* and *transfer*, respectively. The finite automaton shown in the Appendix, Figure 5.19 forbids flights. In comparison to fa_{ptOnly} , car usage is allowed here.

The framework has been implemented in Java 12. The experiments are run on a multi-core environment with 16 core processors (AMD Epyc 7351 Processors) and 256GB of DDR4-2666 RAM.

5.4.2 Metrics

The results are examined using the following metrics:

Run time [s] This is the total run time in seconds the smart sampling framework requires to predict structures of the Pareto front, transform these into instructions for the framework as well as approximate the set of Pareto-optimal itineraries S_{all}^{opt} . Hereby, the time required to train the predictive model is not taken into account as the training occurs prior to and separately from the individual traveler request.

Wasserstein distance The Wasserstein distance quantifies the proximity of two sets of solutions. This metric has been introduced in detail in Section 5.3.3.

of Pareto-optimal solutions The number of Pareto-optimal solutions reflects the size of set S_{all}^{opt} retrieved by the framework. This set forms the choice set for the traveler.

Improved solutions [%] This metric reports the proportion of improved solutions retrieved by the smart sampling framework in comparison to the systematic sampling framework proposed in our previous paper Horstmannshoff & Ehmke (2022), which serves as a baseline for evaluation. For instance, a value of 100% indicates that all identified solutions could be improved and therefore dominate those solutions found by systematic sampling. A value of 0% means that no solution at all could be improved by smart sampling.

Sampling iterations The number of sampling iterations provides information on how often the multimodal routing algorithm SPM is called while approximating the set of Pareto-optimal itineraries.

5.4.3 Configuration of prediction approaches

The prediction approaches introduced in Section 5.3.3 require some configurations, which are as follows.

Number of clusters with similar Pareto fronts: When we predict the Pareto front structure using the “Decision tree” or the “Random forest” approach, we cluster similar Pareto fronts of historical search data using the Wasserstein distance. Therefore, we apply the k -medoid partition technique which requires a number of clusters to be identified as input. As shown in Figure 5.9(a), we have varied the number of clusters (x -axis) during a preceding parameter tuning phase and have measured the inertia value indicating the distance between a medoid and the respective cluster elements (y -axis). To determine a reasonable number of clusters, we have applied the “knee method”. Since the inertia value increases slightly with an increase from six to seven clusters and subsequently does not decline as steeply as before, we have decided for six clusters.

When analyzing the impact of the number of clusters on the prediction accuracy using the “Decision tree” approach, it becomes apparent that using six clusters in the first step leads to the highest prediction accuracy. Figure 5.9(b) presents the average mean and variance of the respective Wasserstein distance of the examined predicted Pareto front of the new traveler request to its real – but actually unknown – Pareto front. For instance, using six clusters leads to a Wasserstein distance of 1.21, on average.

Cross-validation for prediction models: We apply 10-fold cross-validation for the “Clustered attributes”, the “Decision tree” and the “Random forest” approaches to ensure that these perform well on an independent historical search data set. Consequently, we split the data set into ten partitions and use a different fold as the test data in each iteration, while merging the other nine folds to generate the training data for clustering.

Python package: For clustering and building a prediction model, we use the Python package *scikit-learn* always with the default settings if not stated otherwise.

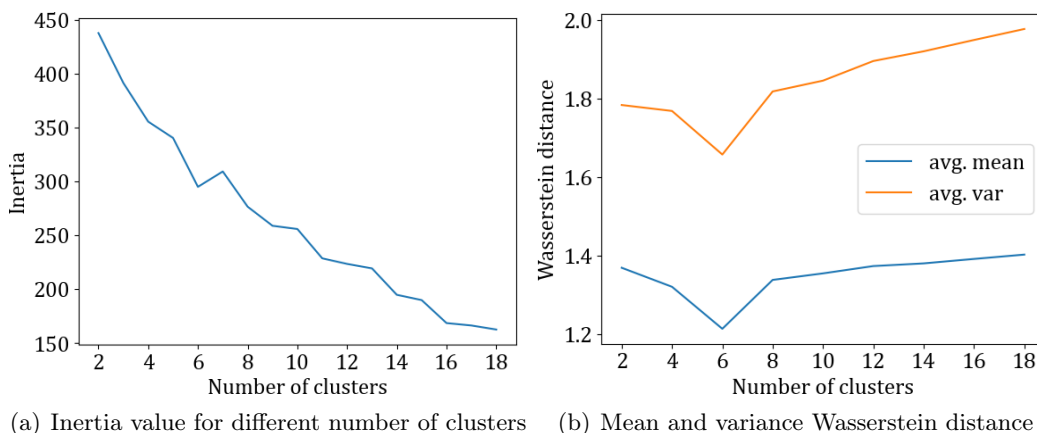


Figure 5.9: Identification of a reasonable number of clusters

5.5 Computational results

In this section, we present the results for guiding the smart sampling framework by learning and integrating structures of the Pareto front. First, we present a detailed example for a specific OD pair in Section 5.5.1. Next, we show results based on an aggregated view across 300 OD pairs. We compare the different approaches to predict structures of the Pareto front in Section 5.5.2. In Section 5.5.3, we analyze the results with respect to different transformation techniques to integrate the predicted Pareto front. Finally, in Section 5.5.4, we present comprehensive results for the best-evaluated approach from a technical as well as from a traveler perspective.

5.5.1 OD-specific example

First, we demonstrate the smart sampling framework for one OD pair, namely from Essen to Osnabrueck, Germany, in detail. In this example, we take five traveler preferences into account with a sampling density of $k = 8$ and “public transport only” as the mode choice option in a three-dimensional setting. We apply “Random forest” prediction with “Euclidean distance” transformation to guide the sampling.

Figure 5.10 shows an example for sampling the set $S_{pr,nt,wd}^3$ with 0-1-normalized preference values. While the blue highlighted Pareto front structure indicates the real – but actually unknown – Pareto front, we depict the Pareto front structure predicted from historical search data in orange. This structure is then used to guide

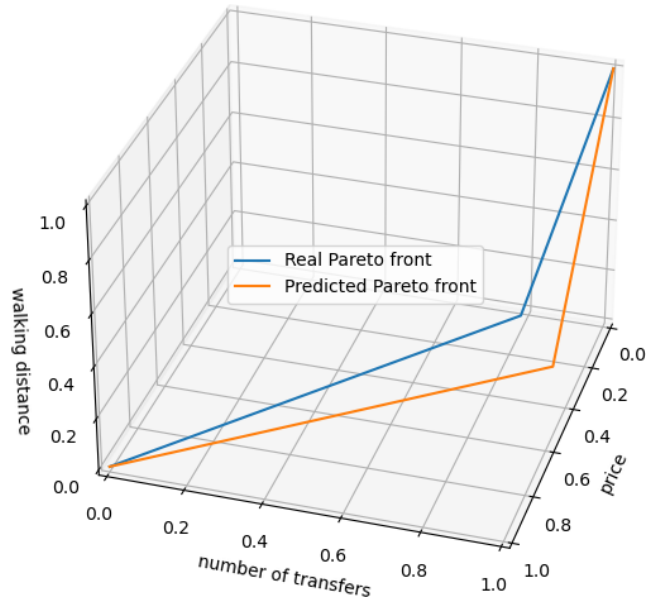


Figure 5.10: Predicted Pareto front for the request from Essen to Osnabrueck

the sampling process. Besides two extreme solutions in the respective corners of the multimodal solution space, an additional trade-off solution can be seen, which lies in both patterns in the same area of the multimodal solution space.

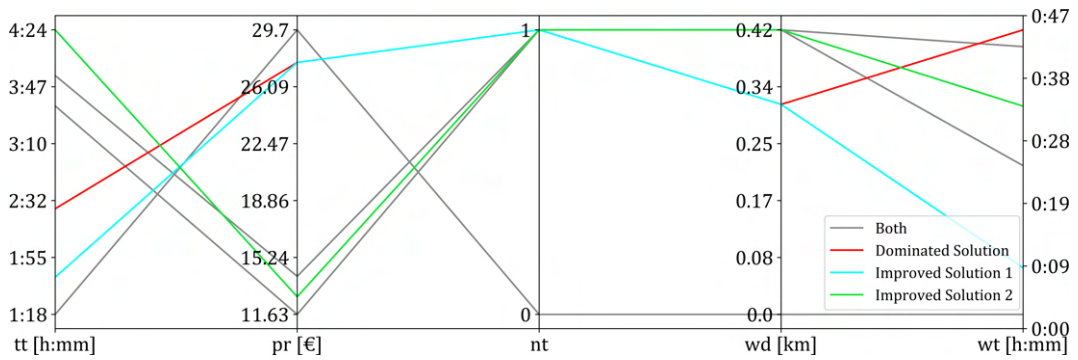


Figure 5.11: Parallel coordinates for the request from Essen to Osnabrueck

Next, we compare the Pareto-optimal itineraries identified using “Random forest” prediction with “Euclidean distance” transformation to the systematic sampling approach (baseline approach) in Figure 5.11. The itineraries highlighted in grey are included in both. In case systematic sampling is applied, the itinerary marked in red

is retrieved. This itinerary has a travel time of about 2.5 hours and costs €27.63, while one transfer is required with a total walking distance of 310 meters and a total waiting time of about 48 minutes. This red-highlighted itinerary is dominated by the one highlighted in light blue (labeled “Improved Solution 1”), which is found only by taking advantage of predicted relationships between traveler preferences. Here, improvements can be observed in terms of total travel time with a saving of 45 minutes as well as an overall reduced waiting time of about 40 minutes.

Regarding the price, the number of transfers, and the overall walking distance, the red as well as the light blue itinerary have the same values. In addition, the green-highlighted itinerary (labeled “Improved Solution 2”) is identified. While this itinerary has the highest travel time, it has a relatively low cost in comparison to other identified itineraries with €12.75. Hence, these two additionally found itineraries enrich the identified set of Pareto-optimal itineraries and give the traveler more attractive options to choose from. Presenting this set of itineraries to the traveler could be the first step to making the available options more transparent and explainable. For instance, a traveler who assesses price as the most important individual preference receives valuable information about relevant itineraries and their relative quality regarding travelers’ preferences.

5.5.2 Prediction of structures of the Pareto front

In Table 5.2, we compare four approaches introduced in Section 5.3.3 for predicting structures of the Pareto fronts. Hereby, we always transform the predicted Pareto front into the smart sampling framework by using “Euclidean distances”. As a benchmark, we present the results of the systematic sampling approach. The results are aggregated across all ODs, sampling densities, mode choice options and set of considered preferences differentiated by two and three-dimensional sampling, respectively.

Using the “Greedy nearest neighbor” approach has an average Wasserstein distance of 0.49 for the two-dimensional setting. This is comparable with the “Decision tree” approach (0.46). While the “Random forest” is the most precise approach (0.38), using “Clustered attributes” is significantly inferior (0.75). In the three-dimensional setting, the prediction accuracy becomes worse due to the higher complexity of the

Table 5.2: Summary of results: Prediction of Pareto front structures

	Wasserstein distance		Improved solutions [%]		Run time [s]		Sampling iterations	
	2-dim.	3-dim.	2-dim.	3-dim.	2-dim.	3-dim.	2-dim.	3-dim.
Systematic Sampling (baseline)	--	--	--	--	10.86	40.56	24.04	69.77
Greedy nearest neighbor	0.49	1.40	0.69	0.67	12.74	45.21	23.86	68.28
Clustered attributes	0.75	1.95	0.61	0.69	12.23	48.43	23.80	67.85
Decision tree	0.46	1.21	0.78	0.74	10.31	44.24	23.99	68.94
Random forest	0.38	1.05	0.78	0.77	10.38	43.09	23.99	68.98

multimodal solution space. Nonetheless, “Random forest” remains to provide the best prediction accuracy (1.05), while “Clustered attributes” performs worst (1.95).

In relation to the baseline, both “Random forest” and “Decision tree” perform best with 0.78% improved solutions for the two-dimensional case. As will be discussed in Section 5.5.4, these results are highly dependent on the underlying OD pair as well as the sampling density k . With respect to the three-dimensional case, “Random forest” performs slightly better with 0.77% in comparison to the “Decision tree” (0.74%). For both settings, “Greedy nearest neighbor” and “Clustered attributes” show less improved solutions than “Random forest”.

The average run time is roughly the same for “Decision tree” and “Random forest” with 10.31 (44.24) and 10.38 (43.09) seconds, respectively, for two-(three-)dimensional sampling. The average run time becomes slightly worse for “Greedy nearest neighbor” and “Clustered attributes”. This observation is also valid for the three-dimensional setting.

No significant differences with respect to the average sampling iterations required can be seen for the different approaches. Compared to the baseline, the average number of sampling iterations required is slightly lower. For instance, the “Random forest” requires on average 68.98 runs of the multimodal routing algorithm while systematic sampling needs 69.77 runs for the three-dimensional case. However,

as improved solutions are identified for “Random forest”, we conclude that taking advantage of relationships between traveler preferences makes sense and guides the smart sampling framework to more promising areas of the multimodal solution space. In particular, using “Random forest” to predict structures of the Pareto fronts is the most promising approach.

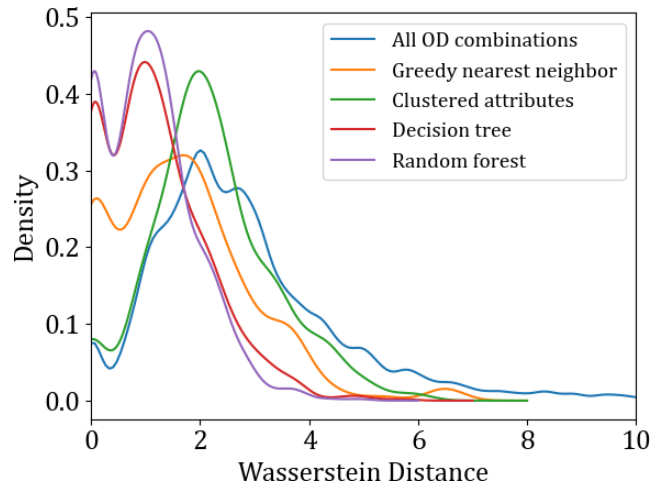


Figure 5.12: Density plot for prediction approaches

Figure 5.12 presents density plots of the Wasserstein distance taking three traveler preferences into account for $k = 16$ and allowing all available modes in a three-dimensional setting. The Wasserstein distance is shown on the x -axis. A leftward shift of the respective density plot towards small Wasserstein distance values indicates a more accurate prediction for the respective approach. The distribution of Wasserstein distances of all possible OD combinations is shown by the blue highlighted density plot with a mean value of 2.90. This distribution stands for a random selection of a predicted Pareto front. For all four examined approaches, we present the respective density plot for the Wasserstein distance of the predicted Pareto front structure to the real Pareto front. It is apparent that the “Random forest” (purple) and “Decision tree” (red) approaches perform best with a mean Wasserstein distance of 1.05 and 1.15, respectively. In comparison, using the “Greedy nearest neighbor” (orange) approach has a shallower and slightly right-shifted density plot indicating a slightly inferior prediction performance. Applying the “Clustered attributes” (green) approach yields very few Wasserstein distances close to zero, and a high accumulation around the

range of 2, falling rapidly again, indicating an average quality prediction of Pareto fronts.

5.5.3 Transformation techniques

Table 5.3: Summary of results: Transformation techniques

	Improved solutions [%]		Run time [s]		Sampling iterations	
	2-dim.	3-dim.	2-dim.	3-dim.	2-dim.	3-dim.
Systematic Sampling (baseline)	--	--	10.86	40.56	24.04	69.77
Euclidean distance	0.78	0.77	10.38	43.09	23.99	68.98
Product distance	0.72	0.64	11.07	39.24	23.22	57.21
Gaussian Process	0.78	0.71	12.11	50.81	24.04	70.96

Next, we analyze the impact of the different transformation techniques to use the predicted Pareto front for guided sampling. These have been introduced in Section 5.3.4. We use “Random forest” as the best-evaluated prediction approach. The results are aggregated across all ODs, sampling densities, mode choice options, and sets of considered preferences. They are differentiated by two- and three-dimensional sampling in Table 5.3.

For the two-dimensional setting, “Euclidean distance” and “Gaussian process” result in 0.78% improved solutions in comparison to the systematic sampling framework. Applying the product distance yields on average slightly less improved solutions with 0.72%. For the three-dimensional case, “Euclidean distance” transformation is best with 0.77% on average, followed by “Gaussian process” with 0.71% improved solutions as well as the “Product distance” transformation technique with 0.64% on average.

The average run time of baseline, “Euclidean distance”, and “Product distance” is comparable, while using “Gaussian process” has a significantly higher run time. For instance, for three-dimensional sampling, “Gaussian process” is on average approximately 7.7 seconds slower in comparison to “Euclidean distance”. The higher run

time of “Gaussian process” is due to the necessary calculations to identify the next sampling point after each sampling iteration.

The average number of required sampling iterations is almost the same for “Euclidean distance” and “Gaussian process” for both two- and three-dimensional settings. Merely using the “Product distance” transformation yields fewer runs of the multimodal routing algorithm, in particular for the three-dimensional setting. This is a possible explanation for the slightly lower number of improved solutions for the “Product distance” transformation.

5.5.4 Random forest prediction with Euclidean distance transformation

Finally, we present detailed results for the best-evaluated approach, i.e., using “Random forest” prediction with “Euclidean distance” transformation.

Table 5.4: Summary results for random forest prediction with Euclidean distance transformation

Dim.	Sampling density k	avg. run time [s]		avg. # Pareto-opt. solutions		avg. improved solutions [%]		avg. sampling iterations		avg. Wasserstein distance	
two-dim. setting	8	7.44	(116.75%)	4.78	(100.12%)	2.29	(-)	21.89	(99.84%)	0.35	(-)
	16	9.35	(115.19%)	5.02	(100.06%)	1.03	(-)	23.35	(99.85%)	0.37	(-)
	32	10.92	(113.76%)	5.14	(99.97%)	0.39	(-)	24.35	(100.03%)	0.39	(-)
	64	11.80	(111.69%)	5.19	(99.99%)	0.15	(-)	24.99	(100.05%)	0.40	(-)
	128	12.36	(111.03%)	5.22	(100.03%)	0.06	(-)	25.37	(99.94%)	0.40	(-)
	\emptyset	10.38	(113.68%)	5.07	(100.03%)	0.78	(-)	23.99	(99.94%)	0.38	(-)
three-dim. setting	8	24.89	(107.87%)	5.43	(100.68%)	2.35	(-)	55.23	(99.07%)	0.94	(-)
	16	34.02	(105.99%)	5.66	(100.17%)	1.05	(-)	62.40	(98.88%)	1.02	(-)
	32	42.45	(105.45%)	5.78	(99.97%)	0.36	(-)	69.29	(99.53%)	1.07	(-)
	64	50.89	(105.53%)	5.83	(99.96%)	0.07	(-)	75.73	(99.88%)	1.10	(-)
	128	63.12	(107.85%)	5.85	(99.98%)	0.01	(-)	85.25	(100.17%)	1.12	(-)
	\emptyset	43.09	(106.54%)	5.71	(100.15%)	0.77	(-)	68.98	(99.51%)	1.05	(-)

Summary results: Table 5.4 shows the results aggregated across all ODs, the number of considered traveler preferences P and applied mode choice variant fa

differentiated by the applied dimension and sampling densities k . These results are further analyzed in the following paragraphs and subsections. The percentages in parentheses represent the average change compared to the systematic sampling framework of our prior paper Horstmannshoff & Ehmke (2022).

The creation of two-dimensional sets requires about 10.38 seconds on average and finds 5.07 Pareto-optimal solutions per request averaged across all sampling density settings. Identifying these sets is 13.68% slower in comparison to the systematic sampling approach. The average run time depends on the respective sampling density parameter k ranging from 7.44 seconds for $k = 8$ up to 12.36 seconds for $k = 128$ and is in all cases slightly higher in comparison to the baseline. Applying a higher sampling density apparently leads to more Pareto-optimal solutions, with decreasing amounts for higher densities. These numbers are almost equal to the systematic sampling framework (baseline). Nevertheless, in particular for smaller densities, using “Random forest” prediction with “Euclidean distance” transformation leads to improved solutions due to the guided search. The more fine-grained we sample, the smaller this effect becomes. While we retrieve on average 2.29% improved solutions for $k = 8$, this percentage improvement decreases with each doubling of k down to 0.06% for $k = 128$. As we do not apply any prediction of Pareto front structures in our baseline approach, no average change is shown in parentheses for improved solutions. The average number of sampling iterations performed is comparable to the baseline at 23.99. The average Wasserstein distance is 0.38 for this setting indicating that the predicted Pareto front is close to the real – but actually unknown – Pareto front for the specific request. The prediction accuracy becomes slightly worse with higher sampling densities.

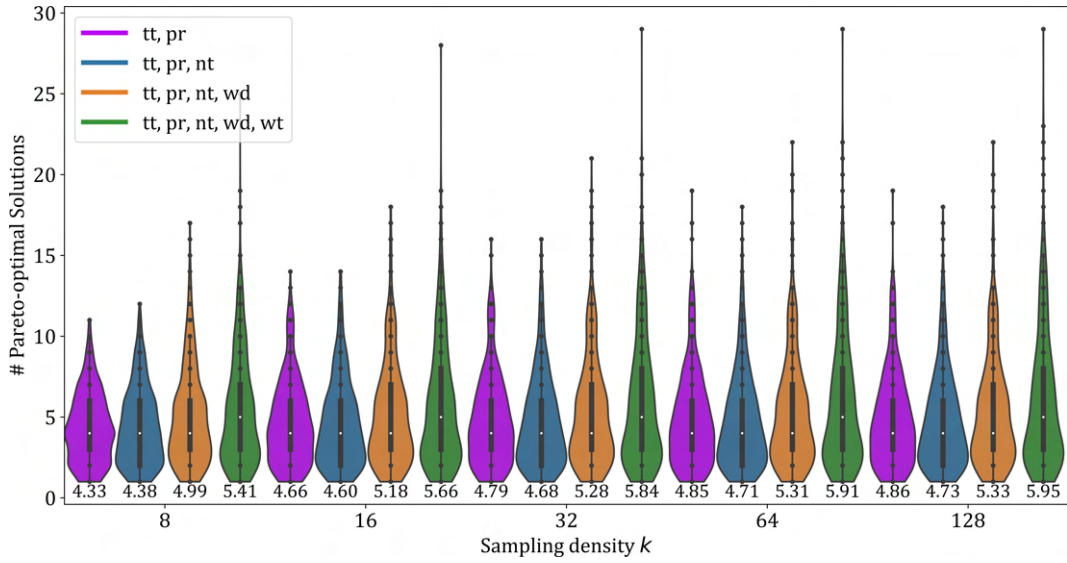
The approximation of three-dimensional sets generates more Pareto-optimal solutions with 5.71 on average. However, this requires much more run time (about 415%) compared to two-dimensional sampling. Using “Random forest” prediction with “Euclidean distance” transformation in the smart sampling framework increases the average run time by about 6.5% compared to the base case. This indicates moderately higher complexity due to the prediction and transformation process applied. While the number of retrieved Pareto-optimal solutions is on average very similar to the baseline, a slight increase in improved solutions can be seen. Again, as for two-dimensional sampling, the percentage of improved solutions is highest for small

sampling densities and decreases with every doubling of k ranging from 2.35% for $k = 8$ down to 0.01% for $k = 128$. The average Wasserstein distance of 1.05 is higher for three-dimensional sampling in comparison to two-dimensional sampling (about 276%). This indicates that the Pareto front becomes more complex and difficult to predict when an additional dimension is considered in the multimodal solution space.

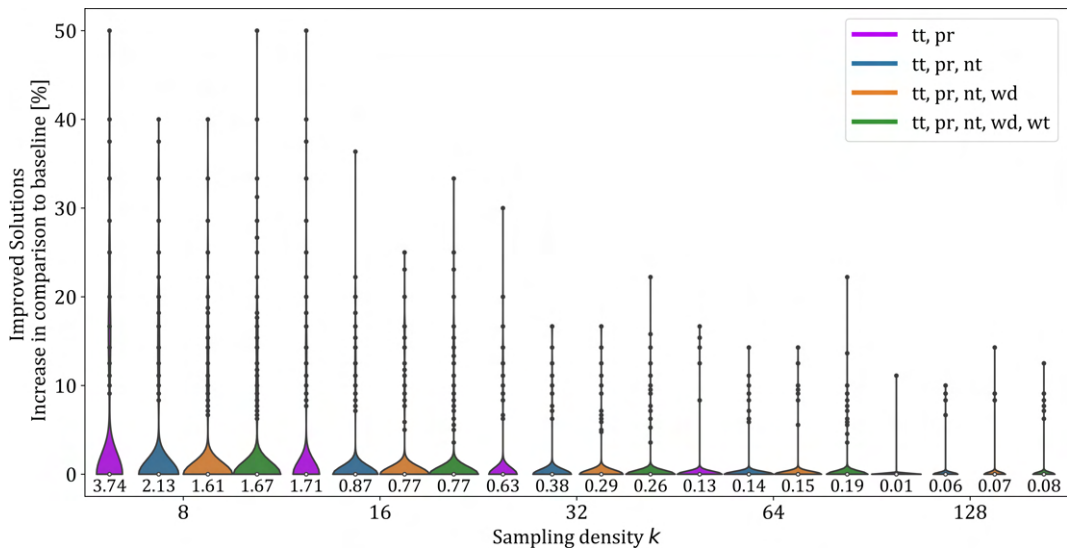
Next, we break down the summary results from different perspectives revealing that the results vary highly depending on the underlying OD and underlying setting in general.

Pareto-optimal and improved solutions: In the first step, we deepen the analysis with respect to the number of Pareto-optimal solutions in relation to improved solutions. Results are shown in Figure 5.13 for the creation of two-dimensional sets and in Figure 5.14 for the creation of three-dimensional sets with sampling density k in increasing number along the x -axis differentiated by the respective set of considered preferences P . The results are aggregated across all 300 ODs as well as across all considered mode choice options fa . The violin plots can be interpreted as follows: Each black dot represents a separate data point, while the average of all data points is represented as a white dot and is also given as a number under the respective plot. The width in the violin plot represents the number of data points for the respective y -value area.

Pareto-optimal and improved solutions: Two-dimensional setting. Figure 5.13(a) shows the number of Pareto-optimal solutions for “Random forest” prediction with “Euclidean distance” transformation. It can be seen that increasing the sampling density k leads to additional Pareto-optimal solutions being found. For instance, for $k = 8$ and considering two preferences simultaneously (tt, pr), on average 4.33 Pareto-optimal solutions are found (left purple plot). Thereby, the number of identified Pareto-optimal solution strongly depends on the underlying OD. While for some ODs only one non-dominated solution is found, at maximum up to 11 Pareto-optimal solutions are found. If we double the sampling density to $k = 16$, the average number of Pareto-optimal solutions increases up to 4.66. Even if we apply a sampling density 16 times higher with $k = 128$ compared to $k = 8$, the average number of Pareto-optimal solutions increases by only about 12% to 4.86. We conclude that a higher sampling density leads to a higher number of Pareto-optimal



(a) # Pareto-optimal solutions



(b) Improved solutions in comparison to the baseline

Figure 5.13: Pareto-optimal and improved solutions for two-dimensional sampling

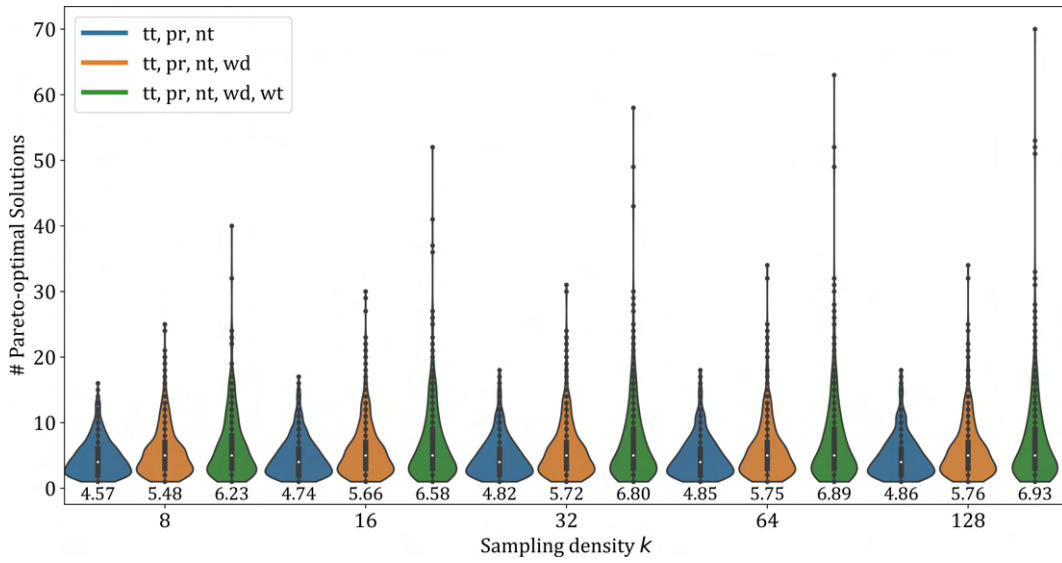
solutions, but with decreasing effect. These findings are consistent with the findings of our prior paper Horstmannshoff & Ehmke (2022).

Furthermore, it is apparent that the average number of Pareto-optimal solutions does not increase significantly (for $k = 8$) and decreases slightly (for $k = 16, 32, 64, 128$) if a third preference (here: number of transfers) is taken into account in comparison to $P := \{\text{travel time, price}\}$. Adding a fourth (here: walking distance) and fifth (here: waiting time) preference into the search has an enormous impact on the average number of retrieved Pareto-optimal solutions. For a sampling density of $k = 128$, 5.33 (orange plot) and 5.95 (green plot) Pareto-optimal solutions are found, respectively.

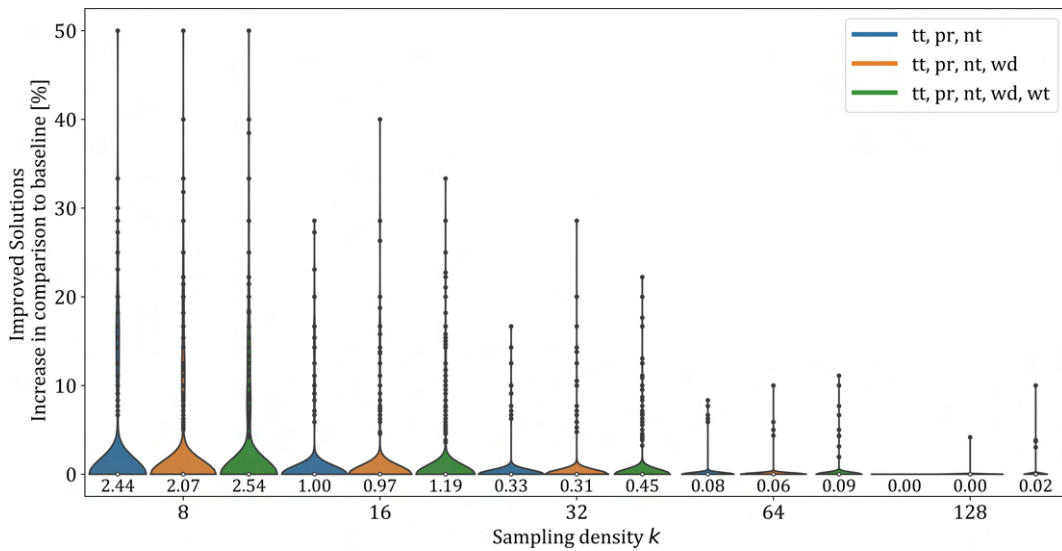
Figure 5.13(b) shows how many improved solutions have been found in comparison to systematic sampling. For instance, when applying a sampling density of $k = 8$ and considering two traveler preferences simultaneously (tt, pr), on average 3.74% of the identified solutions are dominating those found by systematic sampling (left purple plot). It is apparent that with each doubling of the sampling density k fewer improved solutions are found in percentage terms. Regarding the set of considered traveler preferences P , no clear conclusions can be drawn. While the average number of improved solutions tends to decrease for smaller sampling densities ($k = 8, 16, 32$), it increases slightly for higher sampling densities ($k = 64, 128$).

Pareto-optimal and improved solutions: Three-dimensional setting. Next, we analyze the results in terms of the number of Pareto-optimal solutions (Figure 5.14(a)) and improved solutions (Figure 5.14(b)) for a three-dimensional setting. It can be seen that a higher sampling density k yields a higher average number of Pareto-optimal solutions. When taking five preferences into account simultaneously (green plots), we identify on average 6.23 Pareto-optimal solutions for $k = 8$ and up to 6.93 solutions for $k = 128$. In addition, we observe that with each doubling of the sampling density the potential for further Pareto-optimal solutions decreases. As for the two-dimensional setting, the results are highly dependent on the underlying OD. Comparable with the analysis performed for the two-dimensional setting, adding a fourth (orange plots) and a fifth (green plots) preference has a profound effect on the average number of Pareto-optimal solutions. While for $k = 8$ and $P := \{\text{tt, pr, nt}\}$ 4.57 and up to 16 (left blue plot in Figure 5.14(a)) non-dominated solutions are found, taking five preferences into account simultaneously (left green plot) results in 6.23 and up to 40 Pareto-optimal solutions.

In the following, we discuss the proportion of improved solutions compared to the systematic sampling framework serving as a baseline in Figure 5.14(b) for a



(a) # Pareto-optimal solutions



(b) Improved solutions in comparison to the baseline

Figure 5.14: Pareto-optimal and improved solutions for three-dimensional sampling

three-dimensional setting. It can be seen that the highest average improvement can be achieved for lower sampling densities. For $k = 8$ and three preferences (tt, pr, nt) (left blue plot), 2.44% of the solutions are improved. This number drops

to 0% for the highest sampling density of $k = 128$ applied (right blue plot). In general, it can be observed that a more fine-grained sampling leads to fewer improved solutions on average with decreasing effects. In terms of the set of considered traveler preferences P , taking five preferences simultaneously into account yields slightly more improved solutions than considering three or four preferences at the same time, respectively.

Pareto-optimal and improved solutions: Insights. We observe that improved solutions in comparison to the baseline are found for small values for the sampling density parameter k . In this context, large differences can be observed depending on the respective OD. While we do not find any improving solutions at all for the majority of the analyzed ODs, up to 50% improved solutions can be examined for other ODs. We conclude that taking advantage of predicted relationships between preferences makes sense and is competitive to the baseline approach. Comparing the two-dimensional setting and the three-dimensional setting, it is evident that especially for smaller applied sampling densities k , the latter setting shows more potential for improved solutions. This potential decreases for the highest analyzed sampling density value of $k = 128$. Hence, a dense systematic sampling (baseline approach) – due to a high sampling density – already covers many solutions. However, for low sampling densities k , there is significant potential to improve systematic sampling, which is important for interactive multimodal mobility platforms with tight runtime constraints.

Impact of Wasserstein distance on improved solutions: Finally, we analyze the technical quality of the Pareto front prediction with respect to the Wasserstein distance. In Figure 5.15, we show the Wasserstein distance grouped by all mode choice options fa and sets of traveler preferences P . Figure 5.15(a) presents results for $k = 8$ in a two-dimensional setting. Each dot represents an individual request. The Wasserstein distance displayed on the x -axis indicates how close the respective predicted Pareto front is to the real – but actually unknown – Pareto front. Hence, a small value indicates a “good” prediction. The improved solutions in percentages against the baseline approach are shown on the y -axis.

It turns out that smaller values for the Wasserstein distance generally lead to a higher number of improved solutions. Therefore, we conclude that a proper prediction

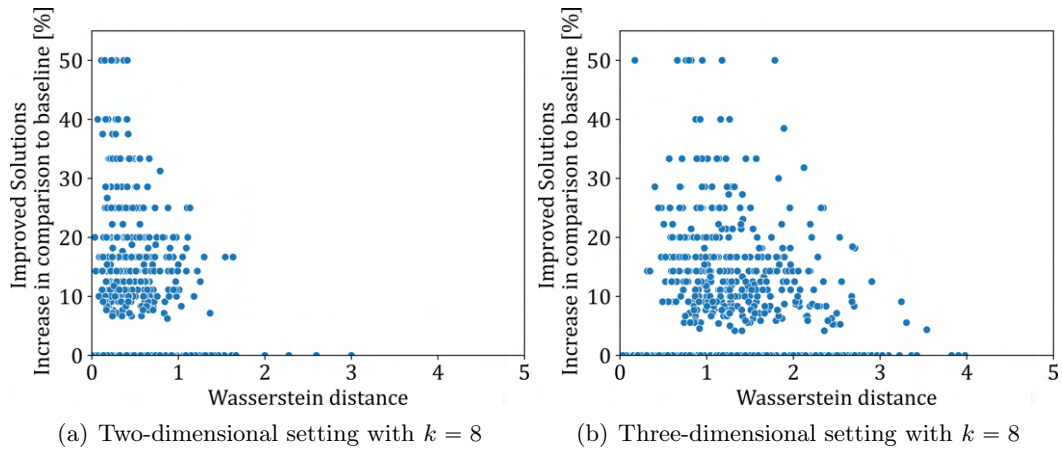


Figure 5.15: Wasserstein distance and its impact on improved solutions

of the real Pareto front and integrating these into the smart sampling framework assists the framework to focus on more relevant areas of the complex multimodal solution space. Nevertheless, as already seen in Figure 5.13, no solutions at all could be improved for plenty of ODs resulting in those blue dots at the bottom of the plot. Figure 5.15(b) presents results for a three-dimensional setting and $k = 8$. The tendency that smaller values for the Wasserstein distance yield a higher number of improved solutions is also valid for this case. Further results for $k = 16, 32, 64, 128$ are presented in Figure 5.20 in Appendix B.

5.6 Conclusion

In recent years, there has been great progress in the planning of traveler-oriented multimodal door-to-door itineraries. However, a major challenge remains when creating a set of multimodal door-to-door itineraries according to multiple individual travelers' preferences. Travelers expect a set of itineraries to choose from that satisfy their individual preferences. When more than three traveler preferences are taken into account at once, modern state-of-the-art MCDM techniques for multimodal routing fail to scale well. In this work, we have enhanced the well-scaling a posteriori multi-criteria decision support system for identifying a traveler-oriented choice set of multimodal itineraries of our previous paper Horstmannshoff & Ehmke (2022). In

particular, we learn and predict structures of the Pareto front from historical search data and OD pair characteristics. The predicted Pareto front structure is then used to guide the sampling framework to more promising areas of the multimodal solution space.

The smart sampling framework has been evaluated by examining itineraries between major cities in Germany including a large amount of real-world data on mobility services. We have analyzed four approaches to predicting structures of the Pareto front as well as three approaches to transforming the predicted Pareto front structure into a sampling procedure. To examine the scaling of the enhanced smart sampling framework, we have taken up to five prevalent traveler preferences – travel time, price, number of transfers, walking distance, and waiting time – into account. Furthermore, we have analyzed the impact of different sampling densities.

In terms of the approaches utilized for prediction and transformation, “Random forest” prediction with “Euclidean distance” transformation has emerged as the best configuration. Especially when only a few iterations (small sampling density) are conducted, significant advantages of a more guided search compared to a uniformly distributed search can be observed. However, the potential improvement because of a more guided search depends highly on the underlying OD pair. Therefore, we conclude that integrating learning and predicting Pareto front structures is of high relevance and promising.

In future work, we plan to use Gaussian process regression to guide the sampling procedure during the search. Considering already identified solutions, the area with the highest uncertainty in the multidimensional solution space can be identified and used as the next sampling point. Hereby, it is dynamically decided which area of the complex multimodal solution space is to be sampled next. Furthermore, we plan to evaluate the proposed framework against routing algorithms that identify the true Pareto-optimal set of multimodal itineraries. As stated in Section 5.2, these algorithms are not efficient for multiple traveler preferences, whereas our proposed smart sampling framework ensures efficient scaling. In addition, the computational design can be expanded to examine the impact of additional preferences such as reliability, sustainability, and more realistic pricing. Furthermore, pruning methods for multi-objective optimization to avoid extremely large sets of solutions can be integrated into the sampling framework.

Appendices

A Mode choice options

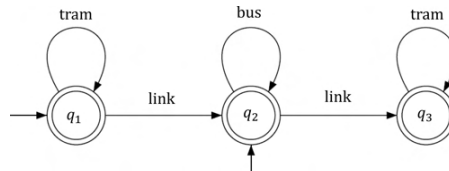


Figure 5.16: Exemplary finite automata



Figure 5.17: Finite automata: All

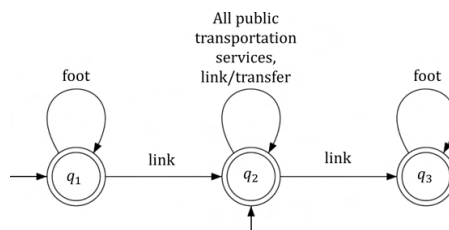


Figure 5.18: Finite automata: Public transportation only

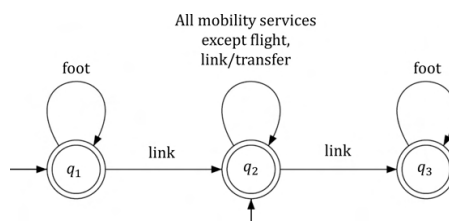
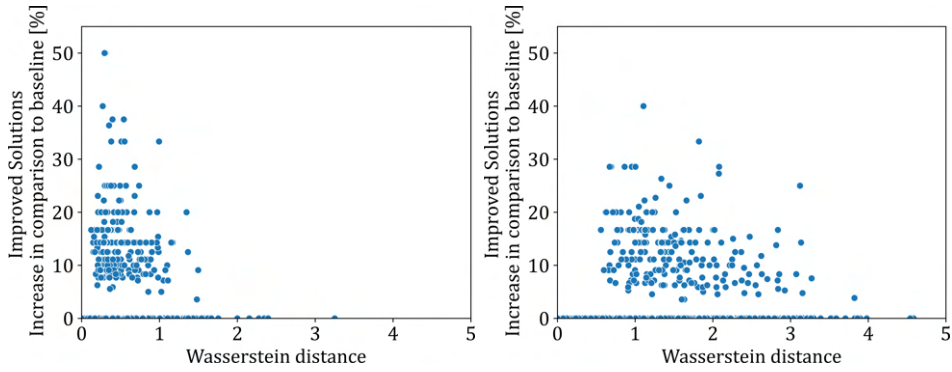
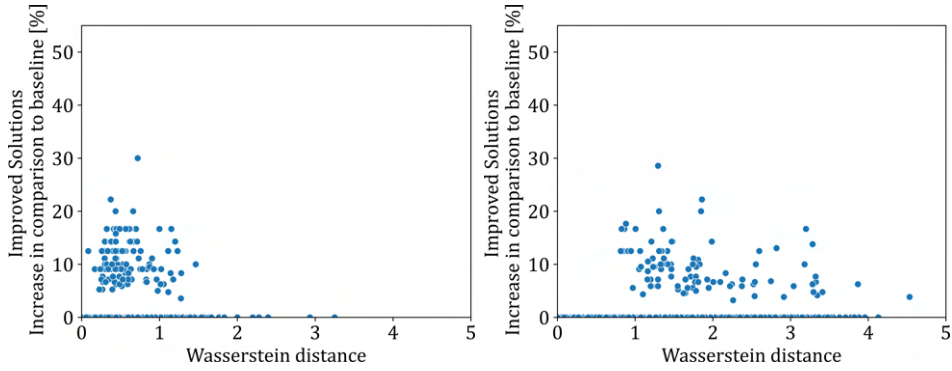


Figure 5.19: Finite automata: No flights

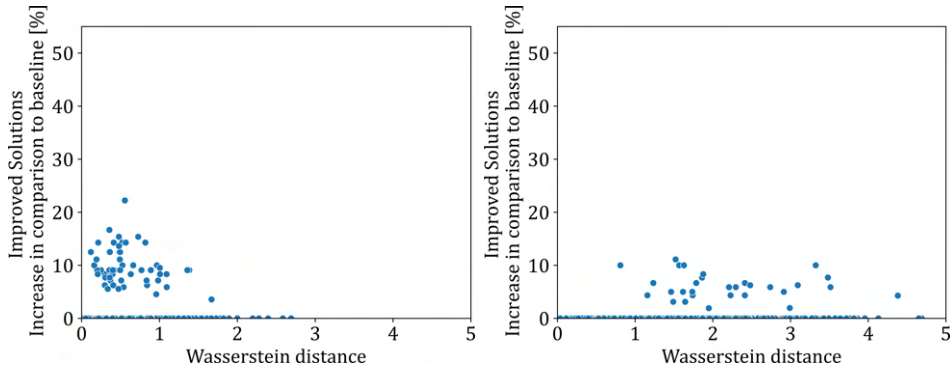
B Impact of Wasserstein distance on improved solutions



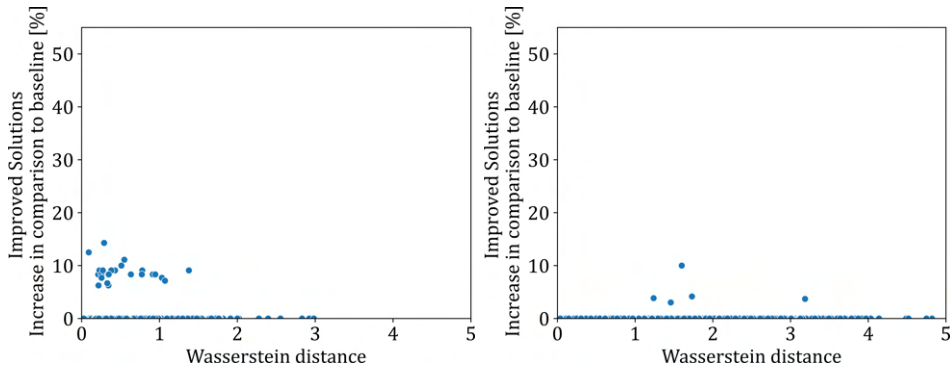
(a) Two-dimensional setting with $k = 16$ (b) Three-dimensional setting with $k = 16$



(c) Two-dimensional setting with $k = 32$ (d) Three-dimensional setting with $k = 32$



(e) Two-dimensional setting with $k = 64$ (f) Three-dimensional setting with $k = 64$



(g) Two-dimensional setting with $k = 128$ (h) Three-dimensional setting with $k = 128$

Figure 5.20: Wasserstein distance and its impact on improved solutions

Chapter 6

Dynamic learning-based search for multi-criteria itinerary planning

Abstract

Travelers expect integrated and multimodal itinerary planning while addressing their individual expectations. Besides common preferences such as travel time and price, further criteria such as walking and waiting times are of importance as well. The competing features of these preferences yield a variety of Pareto-optimal itineraries. Finding the set of Pareto-optimal multimodal travel itineraries in efficient run time remains a challenge in case multiple traveler preferences are considered.

In this work, we present a sampling framework to approximate the set of Pareto-optimal travel itineraries that scales well in terms of considered preferences. In particular, we guide the search process dynamically to uncertain areas of the complex multimodal solution space. To this end, we learn the structure of the Pareto front during the search with Gaussian Process Regression (GPR). The GPR sampling framework is evaluated integrating an extensive amount of real-world data on mobility services. We analyze long-distance trips between major cities in Germany. Furthermore, we take up to five traveler preferences into account. We observe that the framework performs well, revealing origin and destination specifics of Pareto fronts of multimodal travel itineraries.

Keywords Routing, Multi-Criteria Decision Support, Multimodal Mobility, Gaussian Process Regression

Contents

6.1	Introduction	177
6.2	Related literature	179
6.2.1	Traveler requirements to multi-criteria itinerary planning	180
6.2.2	Challenges of platform providers for multi-criteria itinerary planning	180
6.2.3	Dynamically learning Pareto front structures	181
6.3	Gaussian process regression sampling framework	182
6.3.1	Problem description	182
6.3.2	Framework	184
6.3.3	Multimodal network model and routing	190
6.4	Computational design	191
6.4.1	Experimental setup	191
6.4.2	Metrics	193
6.4.3	Configuration of Gaussian process regression	194
6.5	Computational results	195
6.5.1	Summary results	195
6.5.2	Improved solutions	197
6.5.3	Effectiveness of GPR sampling	200
6.5.4	OD-specific example	201
6.6	Conclusion	203
	Appendix	205
A	Accumulated average iteration to retrieve Pareto-optimal solutions	205

6.1 Introduction

Travel planning has always been a challenge, searching for travel modes, checking connections, booking tickets, and all this with a firm eye on time, cost, and convenience. Today's digitization automates some of the parts with single-mode planning for trains, planes, buses, and on-demand services but it does not solve the main problem that when planning, the different modes need to be considered in an integrated manner. First platforms such as *GoogleMaps* and *Rome2Rio* aim for integrated and multimodal planning of travel. For such services, the traveler is either offered only one or two, mostly time- or cost-efficient, options, or the traveler is left alone with all itineraries available. Both do not address the multiple expectations travelers usually have when they travel. Besides classical goals of cheap and fast travel, other criteria play important roles, e.g., a limited number of transfers as well as walking or waiting times. All these options are usually competing against each other, e.g. an itinerary can be either cheap but slow or very fast but expensive. Given that the platforms (and often the travelers) do not know the importance of individual preferences, the travelers desire a reasonable-sized set of options covering all different preferences to select their preferred option, and, ideally, they expect the instant provision of such a set. While this is very convenient for travelers, it poses several challenges for the platform providers. First of all, searching for itineraries over different modes is a challenge in itself. Second, after the search, the platform is confronted with thousands of possible itineraries and now needs to find a small, diverse set to capture potential preferences within a fraction of a second. How to instantly provide a reasonable-sized, diverse set of options in a multimodal context is the focus of this work.

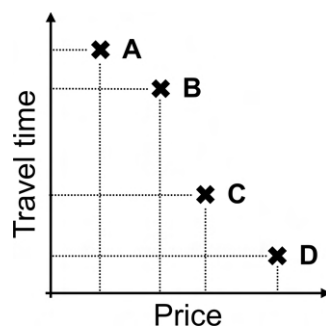


Figure 6.1: Exemplary Pareto front

Mathematically, we aim on determining a Pareto-set for multiple preferences integrating all available mobility services. Figure 6.1 depicts exemplary Pareto-optimal itineraries for two different preferences, travel time and price. Itinerary A is the cheapest, but the slowest, itinerary D is the fastest, but most expensive option. Itineraries B and C are non-dominated compromises between time and price, which differ in terms of travel time and price. Delling et al. (2013a) and Dib et al. (2017) highlight that taking more than three preferences into account at the same time results in run time issues for this Multi-Criteria Decision Making (MCDM) problem. Since run time is highly important for mobility platforms, multimodal search algorithms have to scale efficiently in terms of the number of individual traveler preferences integrated (Bast et al., 2015; He et al., 2022). To ensure scalability, approximating the Pareto-optimal set of itineraries instead of identifying the full Pareto-optimal set has been proposed in recent literature (He et al., 2022; Horstmannshoff & Ehmke, 2022). In Horstmannshoff & Ehmke (2022), we have proposed an a posteriori systematic sampling framework assuming that the traveler expresses individually relevant preferences with no underlying hierarchy. Furthermore, we require the respective origin, the destination, and the earliest departure time by the traveler. Based on this general information, we approximate the set of Pareto-optimal itineraries by computing multiple two-dimensional sets simultaneously with many iterations of solution sampling equally distributed over the solution space. We have observed efficient scaling integrating up to five traveler preferences. However, we have noticed that uniform sampling across the multimodal solution space might lead to inefficiencies with respect to the approximated Pareto front as there is no focus on relevant regions.

In this work, we enhance our systematic sampling framework by actively learning the Pareto front structure during the search and dynamically guiding the search to promising parts of the complex multimodal solution space. The core idea is to apply Gaussian Process Regression (GPR) to identify the area with the highest uncertainty in the solution space and focus the sampling process on that area. Consequently, we improve the systematic sampling framework where no dynamic guidance occurred while searching the multimodal solution space. The choice set retrieved by the GPR sampling framework can then be presented to the traveler enabling the traveler to take well-informed decisions. Furthermore, giving additional information about the

structure of the multimodal solution space in an expert menu in the integrated mobility application assists the traveler in their decision-making progress. The proposed GPR sampling framework is evaluated in a proof-of-concept study embedding a large amount of real-world data from multiple mobility services. We analyze long-distance trips between major cities in Germany, considering up to five traveler preferences. This includes a comparison with the results of the systematic sampling framework, which serves as a baseline of evaluation. With our extensive experiments, we demonstrate efficient scaling of the GPR sampling framework in terms of considered traveler preferences. We observe that learning the Pareto front structure actively is valuable, in particular, if only a limited number of runs are available. Compared to the baseline approach, improvements are observed mainly for those preferences that are continuous (esp. travel time and waiting time). Furthermore, we also demonstrate how the structure of the Pareto-optimal solutions affects these improvements.

An overview of related work on traveler-oriented route planning, multimodal route planning, and GPR is presented in Section 6.2. Then, we introduce the GPR sampling framework in detail in Section 6.3. In Section 6.4, we present the considered data on mobility services, traveler preferences, and further framework settings. In addition, we introduce relevant metrics for evaluation as well as discuss the parameterization of the GPR. Next, we present results in terms of the examined improvement and the increased effectiveness in comparison to the baseline approach in Section 6.5. Finally, we give a summary and identify areas for further research in Section 6.6.

6.2 Related literature

Multi-criteria itinerary planning is a well-established research field with a large number of publications and practical relevance. First, we discuss requirements traveler have while planning multi-criteria itineraries in Section 6.2.1. Next, we examine the resulting challenges for mobility platform providers in Section 6.2.2. In this paper, we propose an active learning sampling framework to identify multimodal traveler itineraries according to multiple individual traveler preferences by actively learning Pareto front structures. Related approaches are discussed in Section 6.2.3.

6.2.1 Traveler requirements to multi-criteria itinerary planning

Multiple studies highlight the importance of taking diverse traveler requirements into account while planning multimodal itineraries. Travelers expect a one-stop search for door-to-door travel and do not want to combine and compare results from different mobility applications manually. Consequently, information on all available mobility services has to be integrated into one mobility application (Schulz et al., 2020; Stopka, 2014; Valderas et al., 2020). Grotenhuis et al. (2007) emphasize that offering integrated multimodal itinerary planning reduces the cognitive and temporal effort for the traveler. Furthermore, travelers have multiple competing traveler preferences which need to be integrated into the search (Lyons et al., 2020; Musolino et al., 2023; Spickermann et al., 2014). Horstmannshoff (2022) presents an overview of recent studies about individually relevant preferences. Beside common traveler preferences such as travel time, price, and number of transfers, further preferences (e.g. walking distance, waiting time, reliability, and sustainability) can be of importance for the individual traveler as well (Liang et al., 2023; Lyons et al., 2020; Samaranayake et al., 2011). Integrating multiple competing preferences into the search yields a set of Pareto-optimal itineraries. For instance, while one itinerary is fast but very expensive, another is much cheaper but requires the traveler to invest more time. Presenting a diverse, reasonably sized set of Pareto-optimal itineraries enables the traveler to choose according to situational and habitual requirements (Grotenhuis et al., 2007; Spickermann et al., 2014; Stopka et al., 2016). Wu et al. (2021) propose to infer request-specific traveler preferences based on automatic fare collection data. In this work, we consider five prevalent preferences in an extensive multimodal context to approximate a set of multimodal itineraries. Furthermore, presenting insights into the complex solution space and the itinerary characteristics provides additional information to the traveler.

6.2.2 Challenges of platform providers for multi-criteria itinerary planning

Several challenges arise from multiple traveler requirements for mobility platform providers. Core of each mobility platform is a multimodal routing algorithm applicable for MCDM problems. Bast et al. (2015) present a detailed overview of route planning algorithms as well as the extension of these into a multimodal context. Herzel et al.

(2021) highlight that most methods for MCDM problem settings assume at most three objectives. Delling et al. (2013a) introduce a label-based algorithm to retrieve a Pareto-optimal set of itineraries in a multimodal context while integrating three preferences at the same time. Dib et al. (2017) proposes a multi-criteria planning algorithm taking travel time, walking time and number of transfers into account. Potthoff & Sauer (2022) extend established bi-criteria routing algorithms by focusing on relevant solutions only during the search. The vast majority of the multi-criteria multimodal routing algorithms introduced in the literature – including the above approaches – consider only two or three preferences simultaneously. Delling et al. (2013a), Dib et al. (2017), and Potthoff & Sauer (2022) emphasize that significant run time issues arise when integrating more preferences into the search while identifying the full Pareto-optimal set.

To ensure scalability and to make the solution space more accessible to travelers, approximating the set of Pareto-optimal itineraries has recently been proposed. We have introduced a systematic sampling framework in our recent paper Horstmannshoff & Ehmke (2022). By breaking down the high-dimensional problem setting into multiple problems of smaller dimensions, we ensure scalability even for a large number of considered traveler preferences. In this work, we propose a scalable multi-criteria itinerary planning framework. In particular, we actively guide the sampling process during the search by dynamically learning the structure of the Pareto front. Thereby, we tackle a shortcoming of our recent paper of sampling the complex multimodal solution space uniformly.

6.2.3 Dynamically learning Pareto front structures

Guiding the search to promising areas of the complex multimodal solution space is challenging. In our sampling framework, we aim at learning the function describing the structure of the Pareto front as accurately as possible with a limited number of iterations available, hence ensuring efficient run time. One common approach to actively explore unknown functions with high accuracy is Gaussian Process Regression (GPR) (Schulz et al., 2018). The core idea of GPR is to learn multiple possible functions according to a preset underlying kernel. Then, once we have some data points x_1, \dots, x_N , we learn a scalar function $f(x)$. Hereby, it is assumed that the function $p(f(x_1), \dots, f(x_N))$ is jointly Gaussian, with some mean $\mu(x)$ and

covariance $\Sigma(x)$ provided by $\Sigma_{ij} = k(x_i, x_j)$, where k is a positive definite kernel function. Using the kernel, we control that x_i and x_j are seen as similar to each other in case we expect these points close to each other in the output function as well (Murphy, 2012; Rasmussen & Williams, 2006). The applied configuration of the GPR kernel for our GPR sampling framework will be discussed in Section 6.4.3.

GPR is applied in many areas of MCDM. For instance, Palm et al. (2022) propose a GPR based multi-objective Bayesian optimization for designing power systems. Deringer et al. (2021) present a detailed overview of how GPR can be used in the area of computational materials science and chemistry taking multiple criteria in parallel into account. However, GPR has yet not been utilized in the area of multimodal itinerary planning.

We apply GPR in the area of multi-criteria multimodal mobility planning to steer our sampling framework to promising parts of the complex solution space.

6.3 Gaussian process regression sampling framework

In the following, we present our GPR sampling framework enhancing the systematic sampling framework introduced by Horstmannshoff & Ehmke (2022). The idea is to learn the Pareto front structure actively during the search and dynamically guide the search to promising parts of the complex multimodal solution space. We refer to the enhanced framework outlined here as *GPR sampling*. We refer to the framework presented in Horstmannshoff & Ehmke (2022), in which no dynamic guidance takes place while searching the multimodal solution space, as *systematic sampling*. Based on the problem description in Section 6.3.1, we introduce the enhanced GPR framework in Section 6.3.2. Finally, we present the multimodal network model and the multimodal shortest-path algorithm in Section 6.3.3.

6.3.1 Problem description

Figure 6.2 depicts the overall procedure for a new traveler request. As common for a posteriori settings, we assume that basic information such as the individual origin O , the destination D , and the earliest departure time t_{dep} is given by the traveler (Figure 6.2(a)). Furthermore, the traveler has an individual set of preferences in mind without being able to judge the respective importance of these. Based on the provided

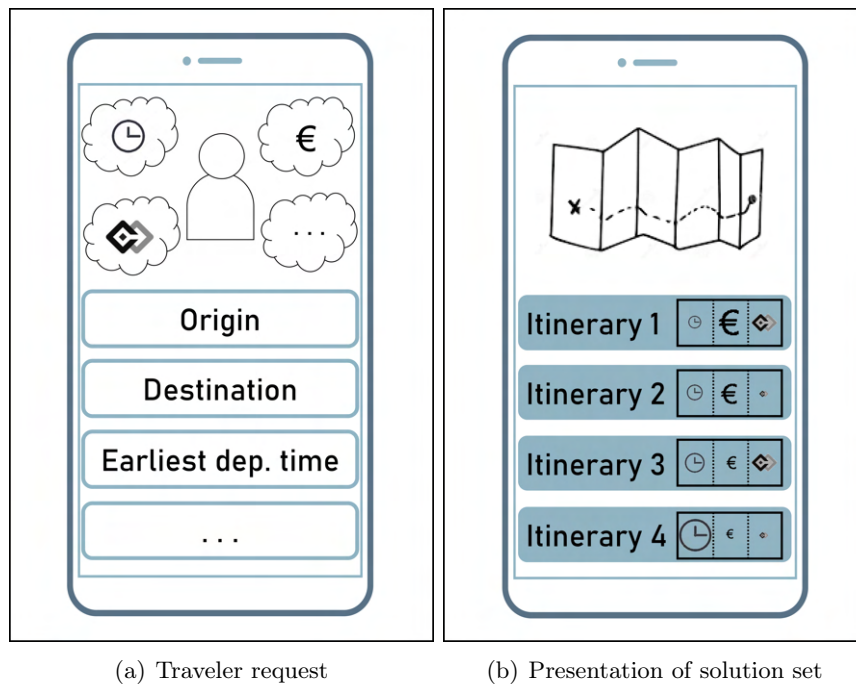


Figure 6.2: Individual traveler request to an integrated mobility application

information and using an integrated mobility application, the traveler expects a diverse set of solutions to choose from (Figure 6.2(b)). Additional information about the solution preference values assists the traveler in their decision-making progress.

To identify the choice set in reasonable run time, several challenges result for the platform provider of the integrated mobility application. First of all, the provider has to integrate a set of ms available mobility services $MS := \{MS_1, \dots, MS_{ms}\}$ such as ridesharing services and railway into the multimodal network model. Furthermore, a set of traveler preferences P has to be considered. The set $P := \{P_1, \dots, P_p\}$ contains p traveler preferences such as price, travel time, and the number of transfers. Knowledge of these parameters is a common assumption for a posteriori settings.

To solve the problem setting at hand, we propose a framework that is capable of retrieving multiple itineraries quickly while integrating multiple traveler preferences simultaneously in a large multimodal network. In the following, we will refer to an itinerary as a solution. To ensure scalability with respect to the number of considered traveler preferences – hence ensure fast run times – we approximate high-dimensional

Pareto fronts by sampling many lower-dimensional Pareto fronts in parallel resulting in a set of Pareto-optimal multimodal solutions S_{all}^{opt} . In particular, we rerun a multimodal routing algorithm SPM multiple times with different configurations for each lower-dimensional set. These configurations are actively learned during the search (details in Section 6.3.2.2). Each run of SPM returns an optimal solution Sol , which contains information about the solution as well as its preference values. Hereby, Sol_i represents the solution value for traveler preference $i \in P$. The set S_{all}^{opt} is composed of different Pareto-optimal solutions. We create several sets of solutions for two and three dimensions. Here, the term dimension refers to how many traveler preferences are taken into account simultaneously when creating the respective sets. Two-dimensional sampling means that we take two preferences into account when approximating the Pareto front for each set. For three-dimensional sampling, we extend this to three preferences considered at the same time. We will repeat sampling pairwise (for two-dimensional sampling) or threefold (for three-dimensional sampling), respectively, to approximate sets of solutions considering all preferences.

6.3.2 Framework



Figure 6.3: Main steps in sampling framework

6.3.2.1 Overview

Figure 6.3 shows the basic procedure of our GPR sampling framework. For a new traveler request, we first identify a min-max-interval $[l_i, u_i]$ for each preference $i \in P$ resulting in a set of intervals I indicating the worst and best value for the respective preference. The intervals are determined by a single-objective run of the multimodal routing algorithm for each traveler preference and setting the obtained minimum and maximum values of the respective preferences as interval boundaries. This guarantees that only reasonable parts of the multimodal solution space are searched. Figure 6.3(a)

presents an example of identifying the set of intervals I in case three preferences $P := \{\text{travel time, price, number of transfers}\}$ are considered.

Next, we approximate the set of Pareto-optimal solutions by many runs of the multimodal routing algorithm SPM . Therefore, we compute multiple two-dimensional sets $S_{i,j} | i \neq j \in P$ simultaneously. Based on the ϵ -constraint method (Deb, 2011), we set preference $i \in P$ as a single objective and iteratively alter the upper-bound constraint of preference $j \in P$ for each set $S_{i,j}$. Figure 6.3(b) depicts a simplified example for identifying six two-dimensional sets in parallel. The reduction of a high-dimensional problem setting into multiple two-dimensional problem sets reduces the computational complexity of calculating each of these sets significantly as we merely have a single-objective problem at hand, and the different upper-bound configurations are only impacted by one preference at the same time. This guarantees scalability even for a larger number of preferences since only the number of parallelly determined sets grows, but not the complexity of calculating each of these two-dimensional sets.

To ensure a focused search on promising areas of the solution space, we learn actively during the search process where to sample next to guide the search and make our sampling framework more effective by applying GPR. The overall procedure how we identify each two-dimensional set $S_{i,j}$ depends on whether preference j is subject to an integer constraint (e.g. number of transfers) or not (e.g. price). For the latter, we dynamically decide which area of the complex multimodal solution space is to be sampled next by actively learning the area with highest uncertainty. Details are presented in Section 6.3.2.2. For preferences to which the former applies, only a few possible upper-bound constraints can be set. Following, we set each possible integer value in the identified min-max-interval $[l_j, u_j]$ as an upper-bound constraint and run the multimodal routing algorithm SPM with preference i as a single-objective. Details are presented in Section 6.3.2.3. After each two-dimensional set $S_{i,j}$ has been calculated, these are merged together into a joint set $S_{all} = \bigcup_{i,j | i \neq j \in P} S_{i,j}$.

Finally, as shown in Figure 6.3(c), dominated solutions are removed from S_{all} as they are of no relevance to the traveler. A solution Sol_1 dominates a solution Sol_2 if Sol_1 is strictly superior to Sol_2 regarding at least one preference $i \in P$ and not inferior in terms all other preferences (Delling et al., 2013a). All remaining Pareto-optimal solutions shape the travelers' choice set S_{all}^{opt} . For a comprehensive introduction to

the systematic sampling framework (without GPR applied) including detailed pseudo codes, we refer to Horstmannshoff & Ehmke (2022).

Algorithm 13 GPR solution space sampling framework algorithm

```

1:  $I \leftarrow \text{IdentificationOfMinMaxIntervals}(O, D, t_{dep}, P)$ 
2: for all  $i, j | i \neq j \in P$  do ▷ Parallelized Execution
3:   if not  $j$  s.t. integer condition then
4:      $S_{i,j} \leftarrow \text{GPR}(O, D, t_{dep}, i, j, I, k)$  (details in 6.3.2.2)
5:   else
6:      $S_{i,j} \leftarrow \text{SystematicSampling}(O, D, t_{dep}, i, j, I)$  (details in 6.3.2.3)
7:   end if
8: end for
9:  $S_{all} = \bigcup_{i,j|i \neq j \in P} S_{i,j}$ 
10:  $S_{all}^{opt} \leftarrow \text{RemovalOfDominatedSolutions}(S_{all})$ 

```

Algorithm 13 presents the procedure of our sampling framework in detail. After identifying a set of min-max-intervals I in line 1, we compute multiple two-dimensional sets $S_{i,j}$ in parallel (line 2). In case preference $j \in P$ is not subject to an integer constraint, we apply the feature of GPR where to sample next and thereby focus on promising parts of the complex multimodal solution space (line 4). Otherwise, we systematically sample the solution space (line 6). After each two-dimensional set has been calculated, we merge all sets (line 9) and remove dominated solutions from the merged choice set (line 10).

6.3.2.2 Gaussian process regression sampling

Next, we introduce how to dynamically decide where to sample next by applying GPR sampling. Algorithm 14 details the overall procedure. Given basic information of the individual request, we sample k times (line 2 in Algorithm 14). In the first iteration of the sampling process, we set the next sampling point in the middle of the identified min-max-interval as we only have knowledge of the min-max-interval boundary solutions (line 4). For the further iterations, we use GPR to identify the area with the highest uncertainty in the solution space and set the next sampling point (line 6). The core idea of GPR is to actively learn the complex structure of the function describing the Pareto front taking already found solutions as well as areas marked as “blocked” into account.

Algorithm 14 GPR learning sampling

```

1: function GPRSAMPLING( $O, D, t_{dep}, i, j, I, k$ )
2:   for  $\hat{j} \in \text{range}(1, k)$  do
3:     if  $\hat{j} = 1$  then
4:        $NextSamplingPoint \leftarrow 0.5$ 
5:     else
6:        $NextSamplingPoint \leftarrow \text{GaussianProcessRegression}(S_{i,j}, Blocked_{i,j})$ 
7:     end if
8:      $C_{restrValue_j} = NextSamplingPoint * (u_j - l_j) + l_j$ 
9:      $Sol \leftarrow \text{SPM}(O, D, t_{dep}, i, C)$ 
10:     $S_{i,j} = S_{i,j} \cup Sol$ 
11:     $Blocked_{i,j} \leftarrow \text{AddBlockedArea}()$ 
12:    if not  $\text{AreaAvailable}(Blocked_{i,j})$  then
13:       $\text{terminate}()$ 
14:    end if
15:  end for
16:  return  $S_{i,j}$ 
17: end function

```

The underlying assumptions of the complex structure (e.g. linear, quadratic, and periodic) can be controlled by a positive definite kernel function, which describes the covariance of the Gaussian process random variables. Using the mean and standard deviation retrieved by GPR, we identify the area with the highest uncertainty in the complex multimodal solution space. This area is chosen as the next sampling point. Thereby, we guide the sampling process dynamically during the search. Please note that we 0-1-normalize the already found solution values to avoid scaling issues in GPR (Rasmussen & Williams, 2006; Schulz et al., 2018). Details about the configuration of GPR will be discussed in Section 6.4.3.

Next, we convert the next sampling point into the actual upper-bound constraint value for preference j (line 8) and run the multimodal routing algorithm SPM with the current setting (line 9). Then, we add the retrieved solution Sol to the set $S_{i,j}$ (line 10).

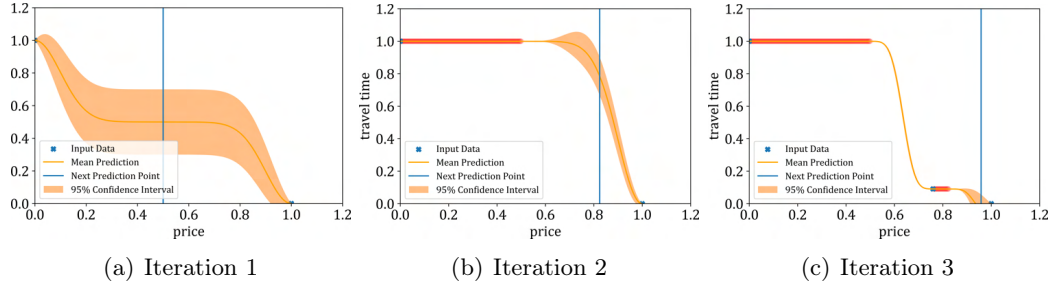
Furthermore, we mark certain areas as blocked. Blocked areas indicate that we have already retrieved the respective solution that we would retrieve when running the sampling framework with upper-bound constraints within the areas marked as blocked. Therefore, there is no added value in running the sampling framework with

the respective setting: the area between the value for preference j (for which the upper-bound constraints are dynamically adjusted) of the newly found solution Sol and the respective upper-bound constraint set for j is marked as blocked since any upper-bound constraint would lead to the same solution. Furthermore, we block the area in the solution space between the newfound solution Sol and an already identified solution in $S_{i,j}$, which both have the same objective function value. In case all solutions between the lower-interval bound l_i and the upper-interval bound u_i are marked as blocked (line 12), we know that no additional solutions will be found in the respective two-dimensional set and thus we terminate the sampling process for the respective set (line 13).

An example for the set $S_{travelTime,price}$ is presented in Figure 6.4(a). Already found solutions (here only the boundary solutions) are depicted by blue crosses. While the mean prediction retrieved by GPR is shown by the orange line, the orange highlighted area indicates the 95% confidence interval. The area with the highest uncertainty (largest confidence interval) is then used as the next sampling point and highlighted by a blue vertical line. The multimodal routing algorithm is started with the upper-bound constraint for the price set in the middle of the corresponding interval. As shown in Figure 6.4(b), the resulting solution is equal to the lower-bound solution of the previously identified min-max-interval. Consequently, this range of the multimodal solution space is marked as blocked (highlighted in red), since no new solutions can be received with upper-bound constraints in this part of the solution space. Subsequently, the next sampling point is determined in iteration 2 by means of GPR resulting in a normalized value of about 0.8. With this configuration, the multimodal routing algorithm is started again in the third iteration (Figure 6.4(c)), and the area between the set upper-bound constraint for price and the value of the solution found for price is marked as blocked. This process is repeated until k iterations have been passed or all solutions ranging from the lower-interval bound l_i to the upper-interval bound u_i are marked as blocked.

6.3.2.3 Systematic sampling for integer constraints

In the following, we introduce the systematic sampling framework applied in case preference j is subject to an integer constraint. Given the traveler's origin O and destination D , the earliest departure time t_{dep} , preference $i \in P$ set as a single-objective,


 Figure 6.4: GPR sampling for set $S_{travelTime,price}$

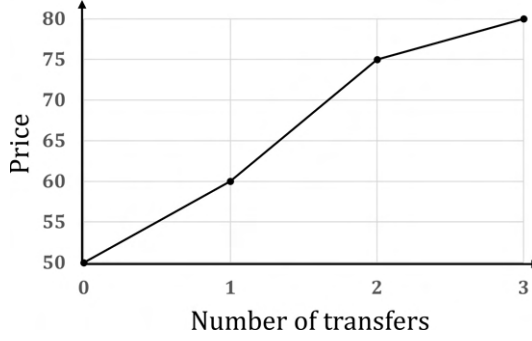
Algorithm 15 Systematic sampling for interval constraints

```

1: function SYSTEMATICSAMPLING( $O, D, t_{dep}, i, j, I$ )
2:    $\hat{j} = u_j - l_j$ 
3:   while  $\hat{j} \geq 0$  do
4:      $C_{restrValue_j} = l_j + \hat{j}$ 
5:      $Sol \leftarrow SPM(O, D, t_{dep}, i, C)$ 
6:      $S_{i,j} = S_{i,j} \cup Sol$ 
7:      $\hat{j} = \hat{j} - 1$ 
8:   end while
9:   return  $S_{i,j}$ 
10: end function
    
```

preference $j \in P$, which upper-bound constraint is systematically altered, and the interval set I , we create the two-dimensional set $S_{i,j}$ as shown in Algorithm 15. For each value in the respective min-max-interval $[l_{numberOfTransfers}, u_{numberOfTransfers}]$ (line 3 to line 8), we systematically alter the upper-bound constraint for preference j (line 4), then run the multimodal routing algorithm SPM (line 5), and finally add the retrieved solution to the two-dimensional set $S_{i,j}$ (line 6).

Figure 6.5 shows a fictitious example for determining set $S_{price,numberOfTransfers}$ with “price” set as the single-objective and “number of transfers” set as the systematically altered upper-bound constraint which is subject to an integer constraint. For each value in the interval $[l_j, u_j] := [0, 3]$ set as an upper-bound constraint, the multimodal routing algorithm is run, and different solutions are retrieved.

Figure 6.5: Systematic sampling for set $S_{price,numberOfTransfers}$

6.3.2.4 Extension to three-dimensional sampling

We also investigate our sampling framework for three dimensions aiming at identifying additional solutions which cannot be found in a two-dimensional setting. This results in multiple three-dimensional sets $S_{i,j,h}^3$ for each considered preference $i, j, h | i \neq j \wedge i \neq h \wedge j < h \in P$. In this case, we set traveler preference i as the objective and dynamically alter the upper-bound constraints for preferences j and h by our framework. The extension of the sampling framework to a three-dimensional setting works analogously to the two-dimensional setting. The identification of the next sampling point by GPR is then conducted in three dimensions.

6.3.3 Multimodal network model and routing

We require a multimodal network and a multimodal routing algorithm for our sampling framework. We build a multimodal network model G that incorporates various mobility services following Pajor (2009). First, for each considered mobility service $ms \in MS$, we create a unimodal network $G_{ms} = (V_{ms}, A_{ms})$ modeled as a time-expanded network. The set of vertices V_{ms} represents arrival and departure events. The set of arcs A_{ms} models a valid trip/subsequence of a trip between two vertices, which are assigned to the same trip. In the next step, we merge all unimodal networks into one multimodal network $G = (V, A)$. The set of vertices V (arcs A) is created by merging all unimodal sets V_{ms} (A_{ms}) with $ms \in MS$ into one set. In addition, we add transfer and link arcs. While transfer arcs provide feasible transfers between the same mobility service at the same stop location at different arrival and departure events, link arcs represent transfers between different mobility services.

These transfers have to be in a given walking distance wd^{max} assuming a predefined walking speed ws .

In addition, we enable car usage by adding an arc from O to D assuming that an itinerary cannot be partially driven by car. Furthermore, we use the concept of Contraction Hierarchy (CH) to establish shortcuts in the network between relevant vertices and therefore speed up the optimization (Geisberger et al., 2008). Finally, to avoid unrealistic solutions, e.g., using a bus service in between two car sections, we use non-deterministic finite automata fa , which represent the travelers' mode choice in the network model (Bast et al., 2015; Pajor, 2009). The core idea is that all solutions retrieved by SPM have to fulfill the solution structure as defined by fa . Building on the multimodal network as well as the finite automata, we create a product network $G^\times = (V^\times, A^\times)$.

Given the network, we rerun a multimodal shortest-path algorithm SPM as presented by Pajor (2009) multiple times with different settings as described above. For an in-depth explanation of creating the multimodal network and the resource-constrained multimodal shortest-path algorithm including a detailed pseudo code, we refer to Horstmannshoff & Ehmke (2022).

6.4 Computational design

In the following, the experimental setup for the evaluation of the GPR sampling framework (Section 6.4.1), relevant metrics (Section 6.4.2), and the configuration of the GPR (Section 6.4.3) will be introduced.

6.4.1 Experimental setup

To evaluate the GPR sampling framework, we examine 300 origin-destination-combinations (ODs) between the main railway stations of major cities in Germany assuming the earliest departure time t_{dep} at 9 am on October 8, 2018. We have chosen this date as data for all integrated mobility services are available and it represents a regular working day (Monday). The resulting multimodal network $G = (V, A)$ for this day of operation consists of about 40,000 vertices and 10,000,000 arcs.

The set of mobility services MS is based on a large amount of real-world data. We include available General Transit Feed Specification (GTFS) data for German

Railways, Flixbus, and local transit services. In addition, we use publicly available service data for flights and long-distance ridesharing service like “BlaBlaCar”. We consider major airports with more than 50,000 aircraft movements per year (Berlin-Schönefeld, Cologne/Bonn, Berlin-Tegel, Dusseldorf, Frankfurt am Main, Munich, Stuttgart, Hamburg, Hanover, Leipzig/Halle, Nuremberg). Furthermore, we assume that the traveler has to arrive at the respective airport one hour prior to the flight departure, and requires 15 minutes after landing to get out of the airport. To enable car usage, we integrate information on the road network using the open-source routing library GraphHopper (www.graphhopper.com). For a more realistic estimation of travel times during peak hours, we multiply the retrieved travel times by 1.25. Furthermore, we set the walking speed ws to $5\frac{km}{h}$ and the maximum walking distance wd^{max} to $0.5km$ for adding the link and transfer arcs as described in Section 6.3.3.

Since German Railways and Flixbus’ prices are not disclosed in the GTFS data, we estimate them as follows. For German Railways, the estimation is based on the train type chosen as well as the distance covered with the respective train type. We consider three different types of trains, namely *regional trains* (slowest train), *intercity trains* as well as *intercity express trains* (fastest train). We assume that intercity trains are 50% more expensive and intercity express trains are 100% more expensive in comparison to regional trains. Based on preliminary empirical investigations on www.bahn.de, the price for regional trains will be €17 per 100-kilometer distance traveled. For Flixbus, we assume €10 per 100-kilometer distance traveled. For individual road mobility, we assume 30 cents per kilometer following the flat-rate depreciation allowance in the German tax system. Note that all these values are only rough estimates and can be adapted as needed in a real-world scenario to, e.g., incorporate tariff fare structures and advance booking periods (Randelhoff, 2022; Schöbel & Urban, 2022). The prices for flights are based on real data, which have been sampled for the respective day of the experiment.

The set of traveler preferences P consists of: *travel time* (tt), *price* (pr), *number of transfers* (nt), *overall walking distance* (wd), and the overall *waiting time* (wt). We examine the following combinations as stated in Table 6.1:

In the first set, we only consider travel time and price. These two preferences are perceived as essential decision criteria for the traveler (Grotenhuis et al., 2007). Subsequently, we integrate additional prevalent exemplary preferences of high impor-

Table 6.1: Examined combinations of traveler preferences

Combination	Set P consists of
tt, pr	{travel time, price}
tt, pr, nt	{travel time, price, number of transfers}
tt, pr, nt, wd	{travel time, price, number of transfers, walking distance}
tt, pr, nt, wd, wt	{travel time, price, number of transfers, walking distance, waiting time}

tance for the traveler into the search (Alt et al., 2019; Esztergár-Kiss & Csiszár, 2015; Grotenhuis et al., 2007). In general, it is simple to add further preferences that can be modeled as cost values of an arc in the multimodal network.

In addition, we analyze the impact of different sampling granularities on the quality of the Pareto front approximation, which is controlled by the sampling density parameter k . We evaluate the GPR sampling framework by setting k to 8, 16, 32, 64 and 128, respectively. With this, we can assess if investing additional effort into the search is advantageous. We also evaluate the impact of extending the search from two to three dimensions, as we expect that considering three dimensions at once improves the solution quality, but also results in larger computational effort.

The framework has been implemented in Java 12. The experiments are run on a multi-core environment with 16 core processors (AMD Epyc 7351 Processors) and 256GB of DDR4-2666 RAM.

6.4.2 Metrics

The results are evaluated using the following metrics:

Run time [s]: The total run time in seconds provides information about the total run time the GPR sampling framework requires to approximate the set of Pareto-optimal itineraries S_{all}^{opt} .

of Pareto-optimal solutions: The number of Pareto-optimal solutions reflects the size of set S_{all}^{opt} retrieved by the framework to form the choice set for the traveler.

Improved solutions [%]: This metric compares the Pareto-optimal sets retrieved by our GPR sampling framework to the systematic sampling framework proposed by

Horstmannshoff & Ehmke (2022), which serves as a baseline for evaluation. While a value of 0% indicates that no solution at all could be improved by GPR sampling, a value of 100% means that all identified solutions have been improved and hence dominate those solutions found by applying systematic sampling.

Iterations to retrieve Pareto-optimal solution: This metric shows in which iteration a Pareto-optimal solution has been retrieved while sampling the respective two-dimensional set $S_{i,j}$ ($S_{i,j,h}^3$ for a three-dimensional setting). As multiple sets are computed simultaneously, a Pareto-optimal solution can be identified in more than one set. In that case, the smaller iteration is taken for this metric.

6.4.3 Configuration of Gaussian process regression

The GPR used to actively learn the Pareto front structure during the search and dynamically decide where to sample next requires specific configurations, which are as follows:

Python package: We use the Python package *scikit-learn* with default settings if not stated otherwise to apply the GPR.

Kernel settings: The underlying kernel, which describes the covariance of the Gaussian process random variables, can be customized to embed the underlying assumptions of the complex structure of the Pareto front into the search. We compose our kernel of two additive components: a radial basic function kernel and a linear kernel. The radial basic function kernel is a widely used kernel that makes the assumption that closer points in the solution space are more similar to each other than farther points (Schulz et al., 2018). Following Rasmussen & Williams (2006), the radial basic function kernel is also referred to as squared exponential kernel and is defined as follows:

$$k(x_i, x_j) = \exp\left(-\frac{1}{2}|x_i - x_j|^2\right). \quad (6.1)$$

The linear kernel takes into account that the boundary solutions of each min-max-interval are in opposite corners of the solution space:

$$k(x_i, x_j) = x_i^T \sum_p x'. \quad (6.2)$$

To ensure reproducible results across multiple runs of the GPR sampling framework, we fix the random number generation used to initialize the centers of the regression to 1234.

6.5 Computational results

In this section, we present computational results for our GPR sampling framework. First, we show aggregated results across 300 OD pairs in Section 6.5.1. Then, we perform a detailed analysis of the number of improved solutions to the systematic sampling framework, which serves as the baseline, in Section 6.5.2. Furthermore, we analyze if these improvements are dependent on the structure of the respective Pareto-optimal solutions. Next, we examine the effectiveness retrieving relevant solutions fast of the GPR sampling framework in Section 6.5.3. Finally, we conclude with a detailed example of a traveler request from Stuttgart to Erfurt in Section 6.5.4.

6.5.1 Summary results

Table 6.2 shows aggregated results across all analyzed ODs and number of considered traveler preferences P differentiated by two- and three-dimensional setting and the applied sampling density k . These are analyzed in-depth in the following subsections. The creation of two-dimensional sets requires about 5.32 seconds on average. The run time differs in terms of the applied sampling density k . Applying $k = 8$, hence sampling eight times per two-dimensional set $S_{i,j}$, requires 4.11 seconds. If we apply a sampling density 16 times as high with $k = 128$, the average run time increases only by about 47% compared to $k = 8$. This is due to the fact that the sampling framework terminates early at high sampling densities as we already have information that no additional solutions can be found with further iterations. Hence, the entire approximated Pareto front is marked as blocked when calculating the respective two-dimensional set. Compared to the systematic sampling framework, the run time increases by approximately 10% resulting from the additional computational effort

Table 6.2: Summary results for GPR sampling

Dim.	Sampling density k	avg. run time [s]		avg. # Pareto-opt. solutions		avg. improved solutions [%]	
two-dim. setting	8	4.11	(114.36%)	4.89	(104.30%)	4.05	(-)
	16	4.98	(113.84%)	5.01	(102.38%)	1.99	(-)
	32	5.57	(111.96%)	5.05	(100.95%)	0.83	(-)
	64	5.91	(111.35%)	5.05	(100.13%)	0.26	(-)
	128	6.05	(110.34%)	5.05	(99.90%)	0.10	(-)
	\emptyset	5.32	(112.37%)	5.01	(101.49%)	1.44	(-)
three-dim. setting	8	12.13	(108.57%)	5.49	(105.85%)	3.76	(-)
	16	15.87	(108.12%)	5.56	(103.07%)	1.78	(-)
	32	19.40	(107.55%)	5.58	(101.15%)	0.53	(-)
	64	22.77	(107.33%)	5.58	(100.24%)	0.09	(-)
	128	28.85	(107.12%)	5.58	(99.96%)	0.01	(-)
	\emptyset	19.80	(107.74%)	5.56	(101.99%)	1.23	(-)

of actively learning the complex structure of the Pareto front during the search. Investing the additional effort pays off compared to the baseline framework. Applying $k = 8$, with 4.89 Pareto-optimal solutions, about 4% additional solutions are found on average. Doubling the sampling density leads to a decrease of additionally found (102.38%) as well as improving solutions (1.99%). This pattern is also valid with each further doubling of the sampling density. This observation indicates that applying the GPR sampling framework is, in particular, promising if only a limited number of runs is available.

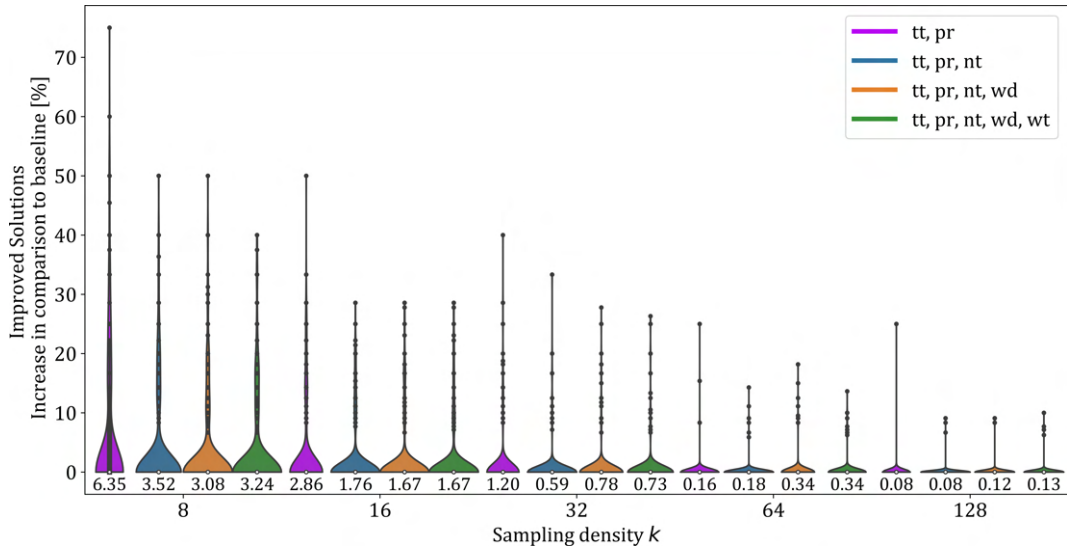
Extending the search to three dimensions results in a significant increase in run time compared to two-dimensional sampling as we now alter upper-bound constraints for two preferences j and h simultaneously when creating the three-dimensional sets $S_{i,j,h}^3$. While we require about 12 seconds on average for $k = 8$, this metric increases to about 29 seconds for a very fine-grained search with $k = 128$. In contrast to the systematic sampling framework, the required run time increases by about 7 to 8 percent. In terms of the average number of retrieved Pareto-optimal solutions and the proportional increase in improved solutions a comparable pattern to the two-dimensional analysis can be examined. Especially for smaller sampling densities k ,

these metric values increase, while almost no change at all can be observed for high $k = 128$.

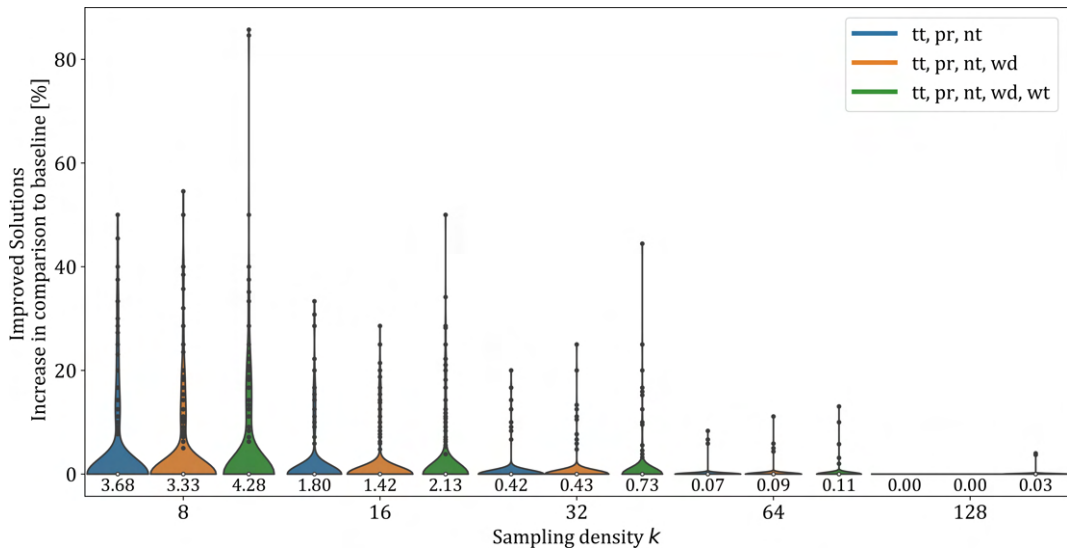
6.5.2 Improved solutions

In the following, we deepen the analysis with respect to improved solutions. Results aggregated across all 300 ODs are shown in Figure 6.6 with sampling density k in increasing number along the x -axis. In addition, we differentiate the results by the respective set of considered preferences P . The violin plots can be interpreted as follows: Each black dot represents a separate data point, while the average is given as a number under the respective plot. The width in the violin plot represents the number of data points for the respective y -value area.

Figure 6.6(a) shows how many improved solutions have been found in comparison to the systematic sampling framework for two-dimensional sampling. For instance, when setting the sampling density to $k = 8$ and considering two preferences in the search (tt, pr), on average 6.35% of the identified Pareto-optimal solutions by the GPR sampling framework are dominating those found by the systematic sampling framework (left purple plot). It is apparent that with each doubling of the sampling density k fewer improved solutions in percentage terms are found with decreasing impact. Limiting the set of considered preferences P yields a higher average number of improved solutions. While, for instance, on average 2.86% improved solutions can be seen for $k = 16$ and $P := \{\text{tt}, \text{p}\}$, merely 1.67% improved solutions occur in case five preferences are considered simultaneously (tt, pr, nt, wd, wt). Comparable observations can be made for three-dimensional sampling in Figure 6.6(b). Again, the number of analyzed improved solutions decreases with every doubling of the configured sampling density value. For instance, when taking three preferences at the same time into account (tt, pr, nt), on average 3.68% improved solutions result while no improved solutions at all occur when sampling fine-grained with $k = 128$. Furthermore, extending the sampling process to three dimensions yields slightly more improved solutions in comparison to two dimensions only as more potential configurations are analyzed in the multimodal solution space. In general, we observe that improved solutions are found in particular for small sampling densities. In this context, large differences can be observed depending on the respective OD. While we do not find any improving solutions at all for the majority of the analyzed ODs,



(a) Improved solutions: Two-dimensional setting



(b) Improved solutions: Three-dimensional setting

Figure 6.6: Improved solutions in comparison to the baseline

up to 85.71% improved solutions can be examined for other ODs. We conclude that actively learning during the search makes sense and is competitive with the baseline approach.

Table 6.3: Baseline solutions enhanced by improved solutions differentiated by preference

Dim.	Sampling density k	Travel time	Price	Number of transfers	Walking distance	Waiting time
two-dim. setting	8	11.92%	7.71%	2.91%	6.88%	9.46%
	16	7.30%	5.18%	1.17%	4.07%	5.35%
	32	3.17%	2.48%	0.86%	2.21%	2.43%
	64	1.38%	0.96%	0.73%	1.44%	1.79%
	128	0.60%	0.43%	0.55%	0.69%	0.74%
	\emptyset	4.87%	3.35%	1.24%	3.06%	3.95%
three-dim. setting	8	11.36%	6.82%	2.55%	7.38%	10.20%
	16	6.83%	4.36%	1.31%	4.68%	6.06%
	32	2.55%	1.49%	0.54%	2.00%	2.57%
	64	0.79%	0.34%	0.29%	0.82%	0.80%
	128	0.15%	0.05%	0.09%	0.12%	0.40%
	\emptyset	4.34%	2.61%	0.96%	3.00%	4.01%

Next, in Table 6.3, we analyze if the observed improved solutions enhance those solutions from the set of baseline solutions with respect to certain preferences. The results are aggregated across all analyzed OD pairs as well as considered sets of traveler preferences P and differentiated by the underlying dimension as well as the applied sampling density k . For instance, for two-dimensional sampling and $k = 8$, on average 11.92% of the baseline solutions could be improved with respect to travel time. This value decreases with every doubling of the configured sampling density. Aggregated across all parameter values for k , 4.87% of the baseline solutions could be improved in terms of travel time. On average, the second highest improvement is seen in waiting time with about 4%, followed by price and walking distance with 3.35% and 3.06%, respectively. Only for number of transfers as the solely considered traveler preference, which is subject to an integer constraint, merely 1.24% solutions could be improved. Similar trends can be observed in the three-dimensional setting. While on average 4.34% of the baseline solutions have been improved in terms of travel time, merely about 1% could be improved regarding the number of transfers. We conclude that applying the GPR sampling framework is in particular promising to retrieve improving solutions with additional value to the traveler in terms of continuous traveler preferences, esp. travel and waiting times.

6.5.3 Effectiveness of GPR sampling

Table 6.4: Avg. iteration to retrieve Pareto-optimal solutions: Two-dimensional setting

sampling density k	GPR:	Systematic sampling (baseline):
	avg. iteration to retrieve Pareto-optimal solutions	avg. iteration to retrieve Pareto-optimal solutions
8	1.81	2.38
16	2.12	2.88
32	2.31	3.66
64	2.33	5.10
128	2.33	7.93

In the following, we examine the effectiveness of the GPR sampling framework in comparison to the systematic sampling framework. Table 6.4 presents the required iterations to retrieve Pareto-optimal solutions averaged across all analyzed 300 OD pairs as well as considered traveler preferences P . For $k = 8$, we require on average 1.81 iterations to find a Pareto-optimal solution while creating two-dimensional sets $S_{i,j}$. To find the respective solution applying the systematic sampling approach, we need on average 2.38 iterations. Hence, Pareto-optimal solutions are approximated faster when actively learning the area with the highest uncertainty in the complex multimodal solution space. When increasing the sampling density to $k = 16$, the average iteration to approximate Pareto-optimal solutions increases to 2.12 when applying the GPR sampling framework. With every further doubling of k , it increases only slightly up to 2.31 (for $k = 32$) and remains equal at 2.33 for $k = 64$ and $k = 128$, respectively. Hence, we examine that putting additional effort has no value anymore, while the average iteration to approximate Pareto-optimal solutions when applying the systematic sampling framework still increases with every doubling of the sampling density k . We conclude that actively learning characteristics of the Pareto front structure enhances the systematic sampling framework significantly.

Figure 6.7 presents further insights into the average number of required iterations to approximate Pareto-optimal solutions for a sampling density of $k = 8$ applied in a two-dimensional setting. The number of iterations is shown on the x -axis. Accumulated information in which iteration proportionally how many Pareto-optimal solutions are

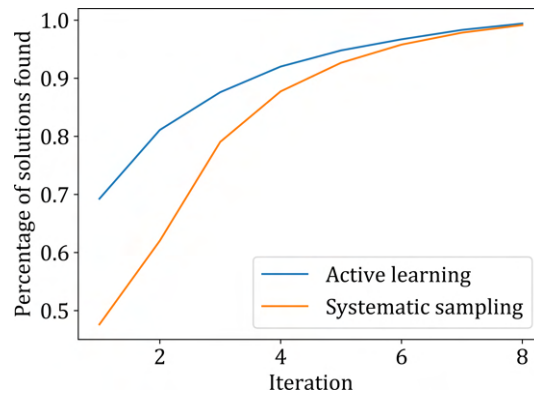


Figure 6.7: Accumulated average iteration to retrieve Pareto-optimal solutions

found are depicted on the y -axis. When applying systematic sampling (orange line), about 50% of the Pareto-optimal solutions are retrieved already in the first iteration. For the GPR sampling framework, even about 70% of the Pareto-optimal solutions are found in the first iteration (blue line). In the next iterations, we examine that applying the GPR sampling framework results in faster convergence of approximating the Pareto-optimal solutions. Comparable observations are applicable for further sampling densities of $k = 16, 32, 64, 128$ as shown in Figure 6.10 in Appendix A.

6.5.4 OD-specific example

Finally, we demonstrate the GPR sampling framework for the specific OD pair of Stuttgart to Erfurt, Germany, in detail. In this two-dimensional setting, we take five preferences into account and set the sampling density to $k = 8$.

Figure 6.8 provides visualizations for the first four iterations when calculating the set $S_{travelTime,price}$ for the exemplary OD pair. Already retrieved solutions are depicted by blue crosses. The orange line indicates the mean prediction of the GPR, while the orange highlighted area indicates the 95% confidence interval. The area with the largest confidence interval – thus the highest uncertainty – is then used as the next sampling point and marked with a blue vertical line. Areas in which no additional information is expected are marked as blocked and highlighted in red. Figure 6.8(a) presents the GPR for the first iteration. As only the boundary solutions of the min-max-interval are known, we set the next sampling point in the middle of the identified interval for price. Rerunning the multimodal routing algorithm

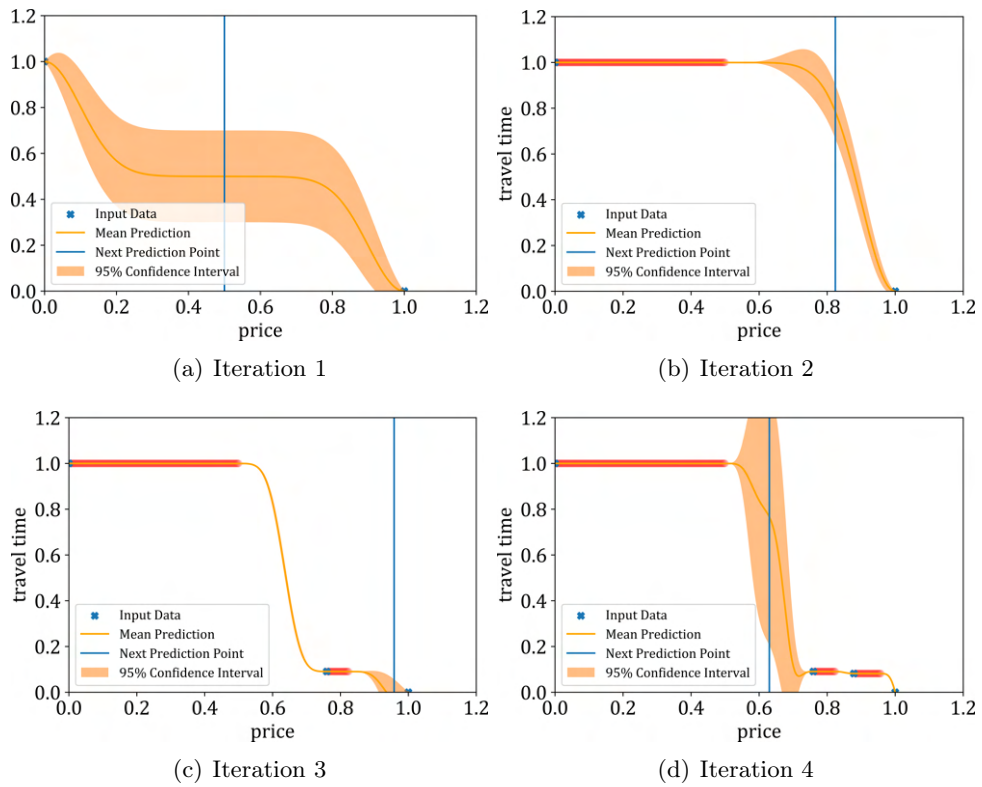


Figure 6.8: GPR for set $S_{travelTime,price}$

with this setting yields a solution that equals the left boundary solution already retrieved while identifying the min-max-interval. Hence, the area between between 0.0 and 0.5 is marked as blocked as can be seen in Figure 6.8(b). Then, GPR is used to identify the area with the highest uncertainty in the multimodal solution space. The respective mean prediction as well as the resulting 95% confidence interval are highlighted in orange in Figure 6.8(b). Following, the next sampling point is set to about 0.8 of the 0-1-normalized price interval. Subsequently, this process is repeated until either all solutions ranging from the respective lower-interval bound to the respective upper-interval bound are marked as blocked or k iterations have been run through.

Figure 6.9 presents the solutions identified using the GPR sampling framework in comparison to the systematic sampling framework. While the solution labeled “Solution 1” to “Solution 5” are retrieved by both frameworks, two additional solutions

are found by actively learning the structure of the Pareto front during the search (highlighted in light green in Figure 6.9). These two additional solutions represent a trade-off compared to the solution with the lowest travel time (highlighted in red); both solutions are slightly cheaper (up to €12.25), but require a bit more travel time and one more transfer.

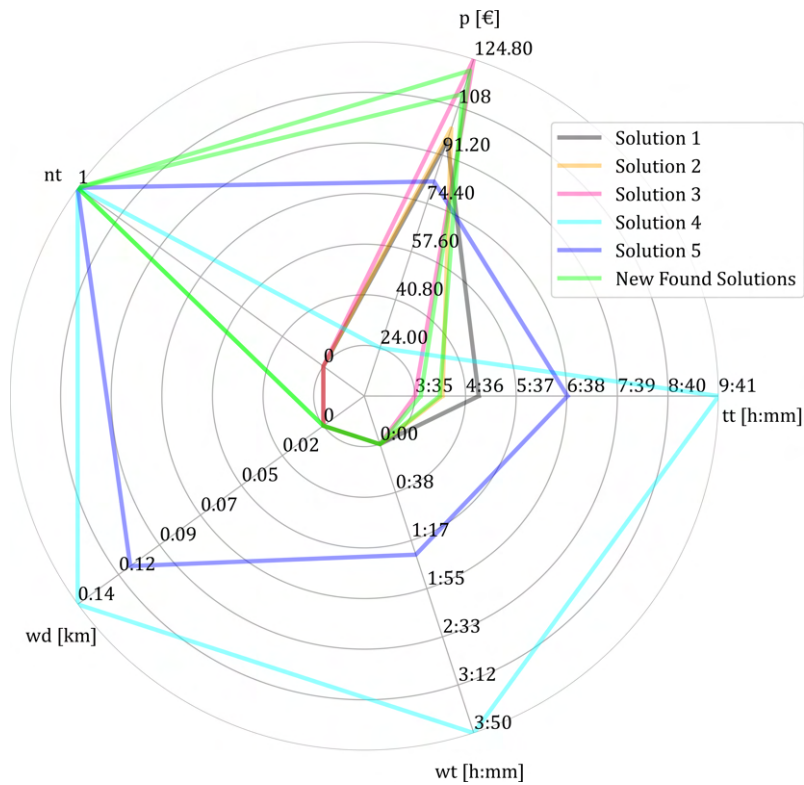


Figure 6.9: Radar plot for the request from Stuttgart to Erfurt

6.6 Conclusion

In recent years, the importance of decision support for the planning of multimodal door-to-door itineraries has increased. However, one major challenge when identifying a choice set for the traveler is the integration of multiple individual traveler preferences into the search. Modern MCDM approaches struggle to scale efficiently in terms of run time when more than three traveler preferences are considered. In this work, we

have enhanced a recent systematic sampling framework proposed by Horstmannshoff & Ehmke (2022). By sampling many lower-dimensional sets simultaneously, we approximate the high-dimensional Pareto fronts efficiently and hence ensure scalability. In particular, we learn the structure of the Pareto front actively during the search and thereby guide the search dynamically to promising parts of the complex multimodal solution space.

The GPR sampling framework has been evaluated analyzing long-distance itineraries between major cities in Germany embedding a large amount of real-world data from multiple mobility services. To evaluate the scalability of the proposed framework, we have integrated up to five traveler preferences – travel time, price, number of transfers, walking distance, and waiting time – into the search. In addition, we have analyzed the impact of different sampling densities.

We have examined significant improvements compared to the baseline framework by Horstmannshoff & Ehmke (2022). In particular, if only a few iterations (small sampling density) are available, dynamically guiding the sampling process to areas with highest uncertainty is promising. The potential improvement is thereby highly OD-dependent. Therefore, we conclude that multi-criteria itinerary planning with GPR shows added value for the traveler.

In future work, we plan to integrate more realistic price information such as tariff fare structures and advance booking periods to design the framework even more realistically. In addition, analyzing the impact of integrating additional traveler preferences such as reliability and sustainability into the framework adds additional value for the traveler. Furthermore, we plan to evaluate the proposed framework against MCDM frameworks from the literature that determine the full Pareto-optimal set of multimodal itineraries. These algorithms do not scale effectively when multiple traveler preferences are taken into account, as stated in Section 6.2, whereas our introduced GPR sampling framework ensures scalability while approximating the set of Pareto-optimal itineraries. This comparison would give us insights how close we are to the computation expensive full Pareto-optimal set.

Appendix

A Accumulated average iteration to retrieve Pareto-optimal solutions

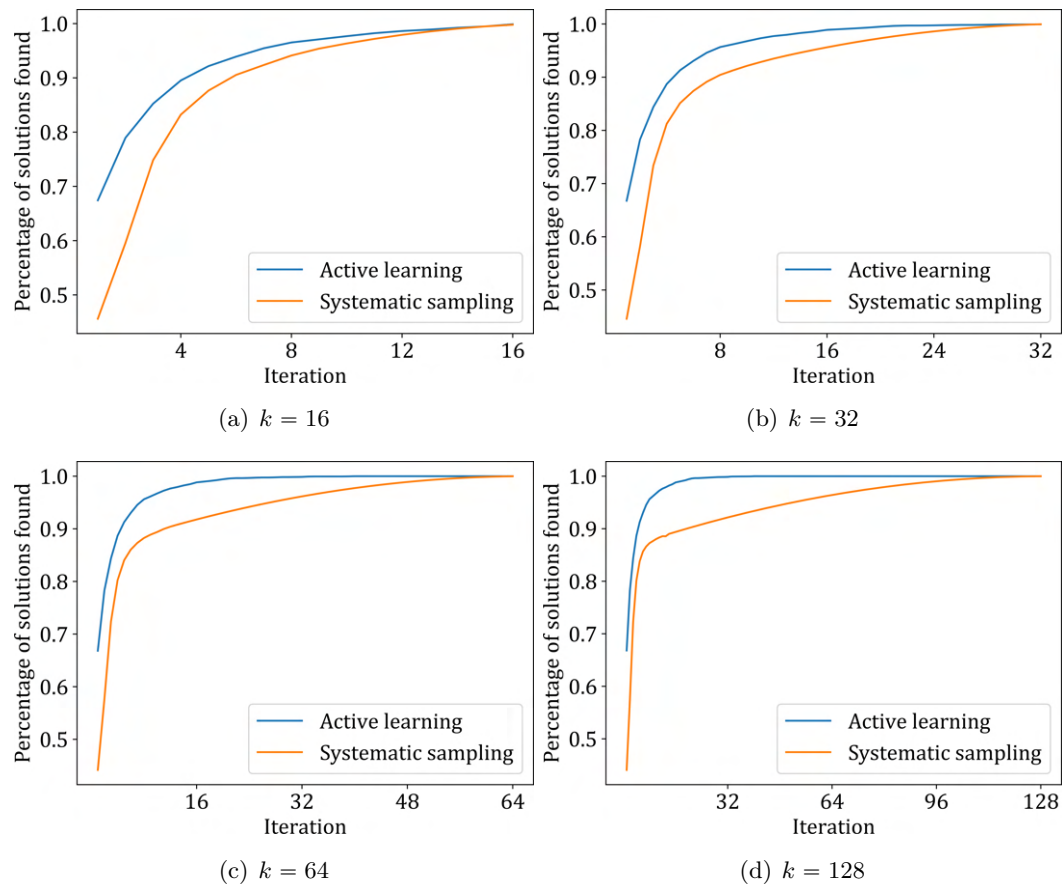


Figure 6.10: Accumulated average iteration to retrieve Pareto-optimal solutions

Chapter 7

Conclusion and outlook

The surge in travel, the growing willingness to adopt multimodal travel, and the trend toward higher sustainability have contributed significantly to the importance of Mobility as a Service (MaaS) platforms in recent years. Essential to foster seamless MaaS and multimodal mobility is the fulfillment of travelers' high expectations toward these digital mobility platforms when searching and planning individual itineraries. In this thesis, we have presented a comprehensive literature review to examine travelers' requirements when using MaaS platforms. Additionally, we have developed two novel multi-criteria decision support frameworks. These frameworks aim to enhance the traveler orientation in multimodal travel planning, thereby contributing to the advancement of MaaS platforms.

The first part of this thesis deals with understanding traveler requirements when planning multimodal itineraries in detail. To minimize the cognitive effort and time expenditure of travelers, they expect all available mobility services to be accessible through an integrated multimodal mobility platform. Furthermore, they expect that multiple prevalent traveler preferences are simultaneously considered in the search. This encompasses preferences such as travel time, price, and the number of transfers, but also preferences like walking distance and waiting time, which are not yet integrated into many state-of-the-art mobility platforms. As these preferences have competing characteristics, and as it is challenging for the traveler to assess the individual impact of these on the complex multimodal solution space, travelers need decision support to choose from a set of reasonable size consisting of Pareto-optimal itineraries. Moreover, travelers expect the presentation of additional information about the available options further alleviating the cognitive and time burden when searching and planning multimodal itineraries. We analyze to what extent the

identified traveler requirements for MaaS platforms are addressed by state-of-the-art multimodal routing algorithms and by the mobility platform *Jelbi*. Identified research gaps in providing multi-criteria decision support for the planning of multimodal itineraries motivate the second part of this thesis.

In the second part, we have proposed two multi-criteria decision support frameworks for planning traveler-oriented multimodal itineraries tackling resulting challenges for mobility platform providers while aiming to achieve seamless MaaS. The first framework focuses on providing the traveler with a reasonable choice set of alternative stops within walking distance based on the individual traveler's request. In this context, we combine route and stop-based information in the decision-making progress. The proposed framework has been evaluated using real-world data from the public transport bus network of Göttingen, Germany. Furthermore, we have added an unscheduled mobility service (electric scooters) to the search. We have observed that travelers have frequently multiple nearby stops with different characteristics in walking distance, which are all Pareto-optimal. For instance, the traveler can save travel and walking time at the cost of reduced service frequency.

The second framework ensures scalability in terms of the number of considered traveler preferences by breaking down the high-dimensional problem into multiple problems of smaller dimensions. The reduction of the underlying dimensionality ensures easier computation. The choice set, which can be presented to the traveler, is identified by systematically sampling across the complex multimodal solution space for each low-dimensional problem setting simultaneously. We have evaluated the framework by analyzing long-distance trips between major cities in Germany integrating a large amount of real-world data on mobility services and considering up to five traveler preferences. Furthermore, we assist the traveler by deriving characteristics of the Pareto-optimal solutions. In addition, we have introduced two approaches to improve the quality of the multi-criteria decision support framework further. First, we enhance the sampling framework by learning and predicting the Pareto-front structure of the complex multimodal solution space and using the predicted structure to guide the search to more promising parts of the solution space. Second, we apply Gaussian process regression to dynamically guide the framework during the search to relevant areas with high uncertainty. Both enhancements of the sampling framework result in significant improvements, which are highly OD-dependent, though.

In the future, the significance of multimodal mobility and the necessity for a seamless MaaS experience will likely continue to increase. Consequently, further developments are necessary for offering multi-criteria decision support when planning multimodal itineraries. In addition to taking up to five considered traveler preferences – travel time, price, number of transfers, walking distance, and waiting time – into account, further preferences such as sustainability and reliability can be of high importance as well. The integration of these preferences adds additional complexity. The consideration of sustainability principles requires the computation of non-linear energy consumption. The integration of reliability as an additional preference necessitates the handling of stochastic data in both the computation and presentation of multimodal itineraries. Hence, the demand for our framework for offering effective scalability can increase even further. The simultaneous integration of various preferences into the search implies that travelers require assistance while searching and planning individual multimodal itineraries. Utilizing advanced design science approaches, such as providing a comprehensible overview of how preferences impact the complex multimodal solution space, could make the search more transparent and hence empower travelers further in their decision-making progress.

Furthermore, certain assumptions that had to be made within this thesis could be alleviated. For instance, the underlying price information can be modeled more realistically by considering tariff structures and the option of early bookings. This topic is strongly connected with complex approaches to demand management known from airlines, for example. Moreover, to evaluate the performance of our Pareto-front approximation framework, a comparison to state-of-the-art multi-criteria solution techniques would be required. A fundamental challenge in this context is that the implementation of these solution techniques is not readily accessible. Furthermore, it requires the standardization of the utilized multimodal networks and the incorporation of the same set of preferences within the network. Finally, we can envision establishing an interaction among problems of smaller dimensions, which are calculated simultaneously, during the search process. For example, the information derived while solving a problem of smaller dimensions, indicating the absence of additional relevant itineraries in a specific part of the complex multimodal solution space, can contribute to enhancing search efficiency.

Global references

- Abdullahi, H., Reyes-Rubiano, L., Ouelhadj, D., Faulin, J., & Juan, A. A. (2021). Modelling and multi-criteria analysis of the sustainability dimensions for the green vehicle routing problem. *European Journal of Operational Research*, *292*(1), 143–154.
- Albrecht, L., & Ehmke, J. F. (2016). Innovative Services in der Mobilitätsbranche: Eine Marktanalyse multimodaler Mobilitätsmanager. *Multikonferenz Wirtschaftsinformatik, MKWI 2016, Vol. 3, 2016*, 1355–1366.
- Alt, R., Ehmke, J. F., Haux, R., Henke, T., Mattfeld, D. C., Oberweis, A., Paech, B., & Winter, A. (2019). Towards customer-induced service orchestration - requirements for the next step of customer orientation. *Electronic Markets*, *29*(1), 79–91.
- Alyavina, E., Nikitas, A., & Njoya, E. T. (2022). Mobility as a service (MaaS): A thematic map of challenges and opportunities. *Research in Transportation Business & Management*, 100783.
- American Public Transportation Association. (2013). *Millennials&Mobility: Understanding the Millennial Mindset*. Publications Office of the APTA.
- Arias-Molinares, D., & García-Palomares, J. C. (2020). The Ws of MaaS: Understanding mobility as a service from a literature review. *IATSS Research*, *44*(3), 253–263.
- Barrett, C., Bisset, K., Holzer, M., Konjevod, G., Marathe, M., & Wagner, D. (2008). Engineering label-constrained shortest-path algorithms. In R. Fleischer & J. Xu (Eds.), *Algorithmic Aspects in Information and Management* (pp. 27–37). Springer Berlin Heidelberg.
- Barrett, C., Jacob, R., & Marathe, M. (2000). Formal-language-constrained path problems. *SIAM Journal on Computing*, *30*(3), 809–837.
- Bast, H., Brodessaer, M., & Storandt, S. (2013). Result diversity for multi-modal route planning. *OpenAccess Series in Informatics*, *33*.

- Bast, H., Delling, D., Goldberg, A., Müller-Hannemann, M., Pajor, T., Sanders, P., Wagner, D., & Werneck, R. F. (2015). Route planning in transportation networks. *Algorithm Engineering, LNCS 9220*, 19–80.
- Baum, M., Buchhold, V., Sauer, J., Wagner, D., & Zündorf, T. (2019). UnLimited transfers for multi-modal route planning: An efficient solution.
- Bell, D. (2019). Intermodal mobility hubs and user needs. *Social Sciences*, 8(2), 65.
- Blank, J., & Deb, K. (2020). Pymoo: Multi-Objective Optimization in Python. *IEEE Access*, 8, 89497–89509.
- Bozyigit, A., Alankus, G., & Nasiboglu, E. (2017). Public transport route planning: Modified Dijkstra’s algorithm. *2017 International Conference on Computer Science and Engineering (UBMK)*, 502–505.
- Bucher, D., Jonietz, D., & Raubal, M. (2017). A heuristic for multi-modal route planning. In G. Gartner & H. Huang (Eds.), *Progress in Location-Based Services 2016* (pp. 211–229). Springer International Publishing.
- Candelieri, A., Ponti, A., & Archetti, F. (2022). Bayesian optimization in Wasserstein spaces. In D. E. Simos, V. A. Rasskazova, F. Archetti, I. Kotsireas, & P. M. Pardalos (Eds.), *Learning and Intelligent Optimization* (pp. 248–262). Springer.
- Cao, Y., Smucker, B. J., & Robinson, T. J. (2015). On using the hypervolume indicator to compare Pareto fronts: Applications to multi-criteria optimal experimental design. *Journal of Statistical Planning and Inference*, 160, 60–74.
- Casadó, R. G., Golightly, D., Laing, K., Palacin, R., & Todd, L. (2020). Children, young people and mobility as a service: Opportunities and barriers for future mobility. *Transportation Research Interdisciplinary Perspectives*, 4, 100107.
- Casady, C. B. (2020). Customer-led mobility: A research agenda for Mobility-as-a-Service (MaaS) enablement. *Case Studies on Transport Policy*, 8(4), 1451–1457.
- Deb, K. (2011). Multi-objective optimisation using evolutionary algorithms: An introduction. In L. Wang, A. H. C. Ng, & K. Deb (Eds.), *Multi-objective Evolutionary Optimisation for Product Design and Manufacturing* (pp. 3–34). Springer London.
- Delling, D., Dibbelt, J., & Pajor, T. (2019). Fast and exact public transit routing with restricted pareto sets. In S. Kobourov & H. Meyerhenke (Eds.), *2019*

-
- Proceedings of the Twenty-First Workshop on Algorithm Engineering and Experiments (ALENEX)* (pp. 54–65). Society for Industrial and Applied Mathematics.
- Delling, D., Dibbelt, J., Pajor, T., Wagner, D., & Werneck, R. F. (2013a). Computing multimodal journeys in practice. In D. Hutchison, T. Kanade, & J. Kittler (Eds.), *Experimental Algorithms* (pp. 260–271). Springer Berlin Heidelberg.
- Delling, D., Goldberg, A. V., Pajor, T., & Werneck, R. F. (2013b). Journey planning in public transportation networks [Google Patents - US Patent 8,494,771].
- Delling, D., Pajor, T., & Werneck, R. (2012). Round-based public transit routing. *Proceedings of the 14th Meeting on Algorithm Engineering and Experiments (ALENEX'12)*.
- Delling, D., Pajor, T., & Werneck, R. F. (2015). Round-based public transit routing. *Transportation Science*, 49(3), 591–604.
- Deringer, V. L., Bartók, A. P., Bernstein, N., Wilkins, D. M., Ceriotti, M., & Csányi, G. (2021). Gaussian process regression for materials and molecules. *Chemical Reviews*, 121(16), 10073–10141.
- Dib, O., Manier, M.-A., Moalic, L., & Caminada, A. (2017). A multimodal transport network model and efficient algorithms for building advanced traveler information systems. *Transportation Research Procedia*, 22, 134–143.
- Dibbelt, J., Pajor, T., Strasser, B., & Wagner, D. (2018). Connection scan algorithm. *Journal of Experimental Algorithmics*, 23, 1–56.
- Dibbelt, J., Pajor, T., & Wagner, D. (2015). User-constrained multimodal route planning. *Journal of Experimental Algorithmics*, 19, 1.1–1.19.
- Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik*, 1(1), 269–271.
- Dolinayova, A., Masek, J., Kendra, M., Čamaj, J., Grandsart, D., Marlier, E., Colzani, P., Arena, M., Paragreen, J., Navaratnam, P., Brennan, M., & Paleta, T. (2018). Research of the passenger’s preferences and requirements for the travel companion application. *Journal of Advanced Transportation*, 2018, 8092147.
- Ehmke, J. F., & Horstmannshoff, T. (2020). Explainable search of multimodal itineraries. In *Joint Proceedings of Modellierung 2020 Short Papers, Workshop Papers, and Tools & Demos Papers* (pp. 150–156).

- Ehrgott, M. (2005). *Multicriteria optimization* (Second edition). Springer Berlin Heidelberg.
- Esztergár-Kiss, D. (2019). Framework of aspects for the evaluation of multimodal journey planners. *Sustainability*, *11*, 4960.
- Esztergár-Kiss, D., & Csiszár, C. (2015). Evaluation of multimodal journey planners and definition of service levels. *International Journal of Intelligent Transportation Systems Research*, *13*(3), 154–165.
- Esztergár-Kiss, D., & Kerényi, T. (2020). Creation of mobility packages based on the maas concept. *Travel Behaviour and Society*, *21*, 307–317.
- Esztergár-Kiss, D., Kerényi, T., Mátrai, T., & Aba, A. (2020). Exploring the MaaS market with systematic analysis. *European Transport Research Review*, *12*(1).
- Esztergár-Kiss, D., & Lizarraga, J. C. L. (2021). Exploring user requirements and service features of e-micromobility in five European cities. *Case Studies on Transport Policy*, *9*(4), 1531–1541.
- European Commission & Directorate-General for Mobility and Transport. (2022). *EU transport in figures: Statistical pocketbook 2022*. Publications Office of the European Union.
- Fatima, K., & Moridpour, S. (2019). Measuring public transport accessibility for elderly. *MATEC Web Conf.*, *259*, 03006.
- Ge, L., Sarhani, M., Voß, S., & Xie, L. (2021). Review of transit data sources: Potentials, challenges and complementarity. *Sustainability*, *13*(20), 11450.
- Geisberger, R., Sanders, P., Schultes, D., & Delling, D. (2008). Contraction hierarchies: Faster and simpler hierarchical routing in road networks. In C. C. McGeoch (Ed.), *Experimental Algorithms* (pp. 319–333). Springer Berlin Heidelberg.
- Giannakopoulou, K., Paraskevopoulos, A., & Zaroliagis, C. (2018). Multimodal dynamic journey planning. *Algorithms*, *12*.
- Gilbert, M., & Ribas, I. (2019). Main design factors for shared ride-hailing services from a user perspective. *International Journal of Transport Development and Integration*, *3*(3), 195–206.
- Göttinger Verkehrsbetriebe GmbH. (2022). *Göttinger Verkehrsbetriebe GmbH*. Retrieved September 30, 2022, from <https://www.goevb.de>

- Grotenhuis, J.-W., Wiegmans, B. W., & Rietveld, P. (2007). The desired quality of integrated multimodal travel information in public transport: Customer needs for time and effort savings. *Transport Policy*, 14(1), 27–38.
- Habermann, A. L., Kasugai, K., & Ziefle, M. (2016). Mobile app for public transport: A usability and user experience perspective. In B. Mandler, J. Marquez-Barja, M. E. Mitre Campista, D. Cagáňová, H. Chaouchi, S. Zeadally, M. Badra, S. Giordano, M. Fazio, A. Somov, & R.-L. Vieriu (Eds.), *Internet of Things. IoT Infrastructures* (pp. 168–174). Springer International Publishing.
- He, P., Jiang, G., Lam, S.-K., Sun, Y., & Ning, F. (2022). Exploring public transport transfer opportunities for pareto search of multicriteria journeys. *IEEE Transactions on Intelligent Transportation Systems*, 1–14.
- He, Y., & Csiszár, C. (2020). Quality assessment method for mobility as a service. *Promet - Traffic & Transportation*, 32(5), 611–624.
- Herzel, A., Ruzika, S., & Thielen, C. (2021). Approximation methods for multiobjective optimization problems: A survey. *INFORMS Journal on Computing*.
- Hillier, F. S., Ehr Gott, M., & Gandibleux, X. (2002). *Multiple criteria optimization: State of the art annotated bibliographic surveys* (Vol. 52). Springer US.
- Horstmannshoff, T. (2022). Mobility-as-a-Service-Plattformen – Berücksichtigung von komplexen Reisendenanforderungen mittels nutzerorientierter Algorithmen. In M. Bruhn & K. Hadwich (Eds.), *Smart Services* (pp. 523–546). GABLER.
- Horstmannshoff, T., & Ehmke, J. F. (2020). Creation of individualized sets of multimodal travel itineraries. *Transportation Research Procedia*, 47, 553–560.
- Horstmannshoff, T., & Ehmke, J. F. (2022). Traveler-oriented multi-criteria decision support for multimodal itineraries. *Transportation Research Part C: Emerging Technologies*, 141, 103741.
- Hwang, C.-L., & Masud, A. S. M. (1979). *Multiple objective decision making – methods and applications: A state-of-the-art survey* (Vol. 164). Springer Berlin Heidelberg.
- Javadian Sabet, A., Rossi, M., Schreiber, F., & Tanca, L. (2021). Towards learning travelers’ preferences in a context-aware fashion. In P. Novais, G. Vercelli, J. L. Larriba-Pey, F. Herrera, & P. Chamoso (Eds.), *Ambient Intelligence – Software and Applications* (pp. 203–212). Springer International Publishing.

- Jie, F., Standing, C., Biermann, S., Standing, S., & Le, T. (2021). Factors affecting the adoption of shared mobility systems: Evidence from Australia. *Research in Transportation Business & Management*, 41, 100651.
- Jittrapirom, P., Marchau, V., van der Heijden, R., & Meurs, H. (2018). Dynamic adaptive policymaking for implementing Mobility-as-a Service (MaaS). *Research in Transportation Business & Management*, 27, 46–55.
- Kaufman, L., & Rousseeuw, P. J. (2009). *Finding groups in data: An introduction to cluster analysis* (9th ed.). John Wiley & Sons, Inc.
- Kirchler, D. (2013). *Efficient routing on multi-modal transportation networks* (PhD Thesis). Ecole Polytechnique X.
- Klatt, J., & Walter, F. (2011). Erhebungsorte. In *Entbehrliche der Bürgergesellschaft?: Sozial Benachteiligte und Engagement* (pp. 59–90). Transcript Verlag.
- Kyamakya, K., & Mitrea, O. (2010). Re-thinking urban mobility services and operations. In J. Düh, H. Hufnagl, E. Juritsch, R. Pfliegl, H.-K. Schimany, & H. Schönegger (Eds.), *Data and Mobility* (pp. 177–186). Springer Berlin Heidelberg.
- Li, Y., May, A., & Cook, S. (2019). Mobility-as-a-service: A critical review and the generalized multi-modal transport experience. In P.-L. P. Rau (Ed.), *Cross-Cultural Design* (pp. 186–206). Springer.
- Liang, J., Zang, G., Liu, H., Zheng, J., & Gao, Z. (2023). Reducing passenger waiting time in oversaturated metro lines with passenger flow control policy. *Omega*, 117(1), 102845.
- Lyons, G., Hammond, P., & Mackay, K. (2020). Reprint of: The importance of user perspective in the evolution of maas. *Transportation Research Part A: Policy and Practice*, 131, 20–34.
- Mandžuka, S. (2021). Providing multimodal traveler information cross-border journey planners approach. *International Conference “New Technologies, Development and Applications”*, 665–672.
- Mavrotas, G. (2009). Effective implementation of the ϵ -constraint method in multi-objective mathematical programming problems. *Applied Mathematics and Computation*, 213(2), 455–465.

- McKenzie, G. (2019). Spatiotemporal comparative analysis of scooter-share and bike-share usage patterns in Washington, DC. *Journal of Transport Geography*, 78, 19–28.
- Meske, C., Bunde, E., & Ehmke, J. F. (2020). Improving customers' decision making on blackboxed multimodal mobility platforms - A design science approach. *International Conference on Information Systems, ICIS 2020 - Making Digital Inclusive: Blending the Local and the Global. Association for Information Systems. 2020. (International Conference on Information Systems)*.
- Miettinen, K. (2013). *Nonlinear multiobjective optimization*. Springer Verlag.
- Molenbruch, Y., Braekers, K., Hirsch, P., & Oberscheider, M. (2021). Analyzing the benefits of an integrated mobility system using a metaheuristic routing algorithm. *European Journal of Operational Research*, 290(1), 81–98.
- Mulley, C., Ho, C., Ho, L., Hensher, D., & Rose, J. (2018). Will bus travellers walk further for a more frequent service? An international study using a stated preference approach. *Transport Policy*, 69, 88–97.
- Murphy, K. P. (2012). *Machine learning: A probabilistic perspective*. MIT Press.
- Musolino, G., Rindone, C., Vitale, A., & Vitetta, A. (2023). Pilot survey of passengers' preferences in Mobility as a Service (MaaS) scenarios: a case study. *Transportation Research Procedia*, 69, 328–335.
- Nasibov, E., Diker, A. C., & Nasibov, E. (2016). A multi-criteria route planning model based on fuzzy preference degrees of stops. *Applied Soft Computing*, 49, 13–26.
- Nykl, J., Hrnčir, J., & Jakob, M. (2015). Achieving full plan multimodality by integrating multiple incomplete journey planners. *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, 1430–1435.
- Omman, I. (2004). *Multi-criteria decision aid as an approach for sustainable development analysis and implementation* (PhD Thesis). University of Graz.
- Pajor, T. (2009). *Multi-modal route planning* (Diploma Thesis). Karlsruhe Institute of Technology. Karlsruhe, Institute of Theoretical Informatics.
- Palm, N., Landerer, M., & Palm, H. (2022). Gaussian process regression based multi-objective Bayesian optimization for power system design. *Sustainability*, 14(19), 12777.

- Pochiraju, B., & Seshadri, S. (2019). *Essentials of business analytics: An introduction to the methodology and its applications* (1st ed. 2019, Vol. volume 264). Springer.
- Potthoff, M., & Sauer, J. (2022–September 9). Efficient algorithms for fully multi-modal journey planning. *22nd Symposium on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2022)*. Ed.: *Mattia D’Emidio*, 106, 14.
- Randelhoff, M. (2022). *Innerdeutsches Reisen: Reisezeit- und Preisvergleiche zwischen Bahn, Pkw, Flugzeug und Fernbus*. Retrieved January 19, 2022, from <https://www.zukunft-mobilitaet.net/172255/analyse/preisvergleich-bahn-flugzeug-fernbus-pkw-auto-reisezeit-dauer-kosten/>
- Rasmussen, C. E., & Williams, C. K. I. (2006). *Gaussian processes for machine learning*. MIT.
- Redmond, M., Campbell, A. M., & Ehmke, J. F. (2020). Data-driven planning of reliable itineraries in multi-modal transit networks. *Public Transport*, 12(1), 171–205.
- Riquelme, N., von Lucken, C., & Baran, B. (2015). Performance metrics in multi-objective optimization. *2015 Latin American Computing Conference (CLEI)*, 1–11.
- Saaty, R.W. (1987). The analytic hierarchy process—what it is and how it is used. *Mathematical Modelling*, 9(3), 161–176.
- Samaranayake, S., Blandin, S., & Bayen, A. (2011). A tractable class of algorithms for reliable routing in stochastic networks. *Procedia - Social and Behavioral Sciences*, 17(1), 341–363.
- Sauer, J., Wagner, D., & Zündorf, T. (2020). An efficient solution for one-to-many multi-modal journey planning. *Algorithmic Approaches for Transportation Modeling, Optimization, and Systems*.
- Schöbel, A., & Urban, R. (2022). The cheapest ticket problem in public transport. *Transportation Science*, 56(6), 1432–1451.
- Schulz, E., Speekenbrink, M., & Krause, A. (2018). A tutorial on Gaussian process regression: Modelling, exploring, and exploiting functions. *Journal of Mathematical Psychology*, 85, 1–16.

- Schulz, T., Gewalt, H., Böhm, M., & Krcmar, H. (2020). Smart mobility: Contradictions in value co-creation. *Information Systems Frontiers*.
- Schwinger, F., & Krempels, K.-H. (2019). Mobility-oriented agenda planning as a value-adding feature for mobility as a service. In *Proceedings of the 11th International Conference on Agents and Artificial Intelligence* (pp. 288–294).
- Sharples, R. (2017). Travel competence: Empowering travellers. *Transportation Research Part F: Traffic Psychology and Behaviour*, 44, 63–75.
- Shokouhyar, S., Shokoohyar, S., Sobhani, A., & Gorizi, A. J. (2021). Shared mobility in post-COVID era: New challenges and opportunities. *Sustainable Cities and Society*, 67, 102714.
- Smith, C. S. (2020). E-scooter mobility: Estimates of the time-savings and accessibility benefits achieved via chicago’s 2019 e-scooter pilot program. *Chaddick Institute Policy Series*.
- Spickermann, A., Grienitz, V., & von der Gracht, H. A. (2014). Heading towards a multimodal city of the future? *Technological Forecasting and Social Change*, 89, 201–221.
- Stadt Göttingen. (2022). *Stadt im Überblick*. Retrieved September 30, 2022, from <https://www.goettingen.de/portal/seiten/stadt-im-ueberblick-900000073-25480.html>
- Stopka, U. (2014). Identification of user requirements for mobile applications to support door-to-door mobility in public transport. In D. Hutchison, T. Kanade, & J. Kittler (Eds.), *Human-Computer Interaction. Applications and Services* (pp. 513–524). Springer International Publishing.
- Stopka, U., Fischer, K., & Pessier, R. (2016). Evaluation methods and results for intermodal mobility applications in public transport. In M. Kurosu (Ed.), *Human-Computer Interaction. Novel User Experiences* (pp. 343–354). Springer International Publishing.
- Stopka, U., Pessier, R., & Fischer, K. (2015). User requirements for intermodal mobility applications and acceptance of operating concepts. In M. Kurosu (Ed.), *Human-Computer Interaction: Design and Evaluation - 17th International Conference, HCI International 2015, Los Angeles, CA, USA, August 2-7, 2015, Proceedings, Part I* (pp. 415–425). Springer.

- Stopka, U., Pessier, R., & Günther, C. (2018). Mobility as a service (MaaS) based on intermodal electronic platforms in public transport. In M. Kurosu (Ed.), *Human-Computer Interaction. Interaction in Context* (pp. 419–439). Springer International Publishing.
- Timm, M., & Störandt, S. (2020). On the multi-kind bahncard problem. In D. Huisman & C. D. Zaroliagis (Eds.), *20th Symposium on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2020)* (2:1–2:13). Schloss Dagstuhl–Leibniz-Zentrum für Informatik.
- Ulloa, L., Lehoux-Lebacque, V., & Roulland, F. (2018). Trip planning within a multimodal urban mobility. *IET Intelligent Transport Systems*, 12(2), 87–92.
- Valderas, P., Torres, V., & Pelechano, V. (2020). Towards the composition of services by end-users. *Business & Information Systems Engineering*, 62(4), 305–321.
- vom Brocke, J., Simons, A., Niehaves, B., Niehaves, B., Reimer, K., Plattfaut, R., & Cleven, A. (2009). Reconstructing the giant: On the importance of rigour in documenting the literature search process. *ECIS 2009 Proceedings*. 161.
- Wienken, T., & Krömker, H. (2018). Experience maps for mobility. In M. Kurosu (Ed.), *Human-Computer Interaction. Interaction in Context* (pp. 615–627). Springer International Publishing.
- Willing, C., Brandt, T., & Neumann, D. (2017). Intermodal mobility. *Business & Information Systems Engineering*, 59(3), 173–179.
- Witt, S. (2015). Trip-based public transit routing. In N. Bansal & I. Finocchi (Eds.), *Algorithms - ESA 2015* (pp. 1025–1036). Springer Berlin Heidelberg.
- Wu, G., Li, Y., Bao, J., Zheng, Y., Ye, J., & Luo, J. (2018). Human-centric urban transit evaluation and planning. *2018 IEEE International Conference on Data Mining (ICDM)*, 547–556.
- Wu, L., Kang, J. E., Chung, Y., & Nikolaev, A. (2021). Inferring origin-destination demand and user preferences in a multi-modal travel environment using automated fare collection data. *Omega*, 101(4), 102260.
- Yan, X., Levine, J., & Zhao, X. (2019). Integrating ridesourcing services with public transit: An evaluation of traveler responses combining revealed and stated preference data. *Transportation Research Part C: Emerging Technologies*, 105, 683–696.

- Yang, M., Li, Y., Zhou, X., Lu, H., Tian, Z., & Luo, J. (2020). Inferring passengers' interactive choices on public transits via MA-AL: Multi-agent apprenticeship learning. *Proceedings of The Web Conference 2020*, 1637–1647.
- Yannis, G., Kopsacheili, A., Dragomanovits, A., & Petraki, V. (2020). State-of-the-art review on multi-criteria decision-making in the transport sector. *Journal of Traffic and Transportation Engineering (English Edition)*, 7(4), 413–431.
- Zipper, D. (2017). *Who Owns Transit Data?* Retrieved April 9, 2017, from <https://www.bloomberg.com/news/articles/2017-04-09/the-case-for-open-transit-data>
- Zou, Z., Younes, H., Erdoğan, S., & Wu, J. (2020). Exploratory analysis of real-time e-scooter trip data in Washington, DC. *Transportation Research Record*, 2674(8), 285–299.

