

## Mobile Anwendung zur Routenplanung mit öffentlichen Verkehrsmitteln basierend auf OpenStreetMap-Daten

Markus Duchon, Andreas Paulus, Martin Werner

### Kurzfassung:

Das Netz öffentlicher Verkehrsmittel bestehend aus Bus, Tram-, U- und S-Bahn ist in vielen Städten im Laufe der Zeit stark gewachsen und sehr komplex geworden. Dabei ist es vor allem bei einer erstmaligen Reise zu einem bestimmten Ort unklar, wie dieser am schnellsten zu erreichen ist. Insbesondere wenn Abweichungen vom Fahrplan, die durch hohes oder niedriges Verkehrsaufkommen oder andere Zwischenfälle entstehen, dazu führen, dass mögliche Anschlussverbindungen nicht rechtzeitig erreicht werden können. Unter Berücksichtigung entsprechender Schwankungen besteht die Möglichkeit das Ziel auf einem anderen, schnelleren Weg zu erreichen. Die folgende Arbeit präsentiert eine mobile Anwendung für die Android Plattform in Anlehnung an [1], die mittels GPS-Daten und einem vom Benutzer eingegebenen Ziel die zum aktuellen Aufenthaltsort und Zeitpunkt ideale Route mit öffentlichen Verkehrsmitteln berechnet und gegebenenfalls aktualisiert. Die dafür notwendigen Strecken- und Haltestelleninformationen wurden aus dem OpenStreetMap Projekt extrahiert, nach bearbeitet und in ein geeignetes Datenformat überführt, welches die Anbindung weiterer Ressourcen, wie Straßennamen, Abfahrtszeiten, Haltestellenbeschreibung, etc. ermöglicht. Die dadurch entstandenen Wegnetze werden als gerichtete Graphen (Straßen-, ÖPNV-Netz) in unterschiedlichen BSP-Bäumen gespeichert. Die Übergänge zwischen den Graphen wurden durch zusätzliche Kanten modelliert und ermöglichen so eine multimodale Routenführung. Zu Testzwecken wurde der Großraum München verwendet, wobei das eingesetzte Routing-Verfahren bei der Ermittlung des kürzesten Weges die aktuellen Live-Abfahrtszeiten des Betreibers berücksichtigt.

## 1. Einleitung

In Zukunft wird Mobilität nicht mehr nur über einen einzigen Transportanbieter oder den gängigen Individualverkehr auf effiziente Art und Weise bewerkstelligt werden, sondern vielmehr durch eine intelligente Kombination unterschiedlicher Verkehrsmittel. Damit der wachsende Bedarf von der zur Verfügung stehenden Infrastruktur gedeckt werden kann, muss die Effizienz des Gesamtsystems optimiert werden. In diesem Zusammenhang sollte auch die Attraktivität des öffentlichen Nahverkehrs verbessert werden, damit die vorgehaltene Kapazität besser ausgenutzt werden kann. Nach einer bereits 1998 in Bremen durchgeführten Studie [2] erachtet die überwiegende Mehrheit der Nutzer so genannte dynamische Fahrgastinformationssysteme als sinnvoll. Diese wirken sich insgesamt positiv auf den empfundenen Komfort aus. Diese Informationen werden mehr und mehr auch über das Internet zur Verfügung gestellt. Neben den minutengenauen Abfahrtszeiten an den in der Nähe befindlichen Haltestellen bieten diese zusätzlich die Möglichkeit an, die verfügbaren Verbindungen zwischen einem Start- und Zielpunkt zu ermitteln. Leider beschränken sich diese Möglichkeiten immer auf das Angebot des jeweiligen Anbieters und integrieren weder Individualverkehr noch Fußwege. In der Regel basieren solche Anwendungen auf einer Client-Server Architektur. Dies impliziert neben Skalierbarkeits- und Kompatibilitätsproblemen auch diverse Probleme in Bezug auf den Schutz der Privatsphäre der Nutzer.

Im Vergleich zum Straßennetz wirken die möglichen Routen des öffentlichen Verkehrs wesentlich übersichtlicher. Hierbei handelt es sich allerdings um ein getaktetes, asymmetrisches Netz. Deshalb hängt die Länge eines Weges auch wesentlich von der Ankunftszeit an

jedem einzelnen Wegpunkt ab und die Routenbestimmung wird wesentlich komplexer.

Im Rahmen dieser Arbeit wird eine mobile Anwendung zur multimodalen Routenplanung basierend auf OpenStreetMap-Daten vorgestellt. Diese arbeitet im Gegensatz zu zahlreichen anderen Vertretern dieser Art autonom und bestimmt die ideale Route direkt auf dem Endgerät. Dabei werden neben Fußwegen auch Strecken des öffentlichen Nahverkehrs verwendet. Abbildung 1 zeigt ein typisches Beispiel, in welchem zwei Routen eingezeichnet sind, die jeweils zu einem bestimmten Zeitpunkt die schnellste Verbindung zwischen A und Z beschreiben.

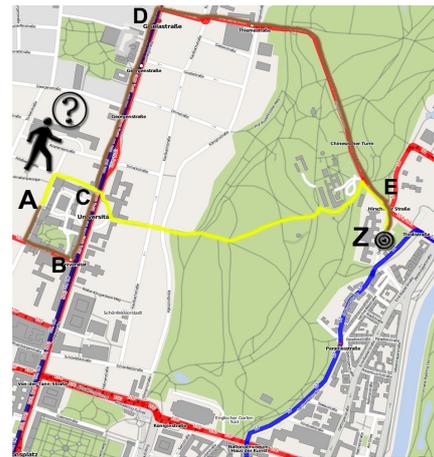


Abbildung 1: Szenario

## 2. Verwandte Arbeiten

In den letzten Jahren gewinnen mobile Systeme und Anwendungen für den Bereich der Fahrgastinformation immer mehr an Bedeutung. Neben den von zahlreichen Nahverkehrsbetreibern angebotenen Reiseplanern [3]-[5] erschienen in der letzten Zeit auch einige freie Anwendungen. Viele dieser Anwendungen verwenden das Global Positioning System (GPS) um die aktuelle Position des Benutzers zu bestimmen und liefern die Abfahrtszeiten an den in der Nähe befindlichen Haltestellen zurück. Ferris et. al. stellen in ihrer Arbeit [6] eine solche Anwendung namens „OneBusAway“ vor. Diese Anwendung kommuniziert dazu mit einem Server, der die Informationen für die aktuelle Umgebung des Benutzers bereitstellt. Dieser kann aber auch den angezeigten Kartenausschnitt verschieben und erhält die entsprechenden Informationen für den gewählten Ausschnitt. Allerdings bietet diese Anwendung keine Routingfunktionalität. Das Forschungsprojekt „Der orientierte Mensch“ [7] untersuchte die Personalisierung und Speicherung von öffentlichen Nahverkehrsrouten bzw. Fahrzeugrouten auf mobilen Endgeräten. In einer „Reisemappe“ werden die geplanten Routen abgespeichert und können über einen WAP-Service bei Bedarf aktualisiert werden. Diese Anwendung bietet dabei eine statische Navigationsunterstützung entweder für den PKW oder den öffentlichen Nahverkehr an. In der Arbeit von Becirovic [8] wurde basierend auf einer Reihe von bestehenden Web-Services eine mobile Anwendung für Java-fähige Handys entwickelt. Die vorgestellte Anwendung war prinzipiell deutschlandweit einsetzbar, wobei jedoch nur die Abfahrtszeiten im Münchner Nahverkehrsnetz implementiert wurden. Der Benutzer hatte dabei die Möglichkeit über die Anwendung einen Start sowie einen Zielpunkt zu wählen, wobei die Routenberechnung selbst von einem Web-Service übernommen und nicht ohne eine Internetanbindung verwendet werden konnte. Rehr et. al. fassen in ihrer Arbeit [9] Anforderungen an mobile, multimodale Navigationsanwendungen zusammen. Die entstandenen Prototypen erweitern ein Fahrzeugnavigationssystem so, dass Abfahrtszeiten von öffentlichen Nahverkehrsmitteln an Park-and-Ride Plätzen auf einem zentralen Server gespeichert und zur späteren Verwendung auf einem Smartphone genutzt werden können.

Insgesamt kann festgestellt werden, dass fast alle Anwendungen in diesem Bereich auf einer Client-Server Architektur basieren und lediglich eine Routenplanung vor Reiseantritt ermöglichen. Eine automatische Routenaktualisierung bei Verspätungen oder beim Verlassen der geplanten Route wurde in diesem Bereich bisher nicht berücksichtigt.

## 3. ÖPNV Routenplaner

### 3.1. Anforderungen

Damit eine entsprechende Anwendung umgesetzt werden kann, müssen einige Anforderungen erfüllt werden, um eine multimodale Wegfindung zu ermöglichen, die autonom auf dem Endgerät stattfinden soll. Zunächst werden geeignete Informationen über das zur Verfügung stehende Straßen- bzw. Wegenetz – sowohl für die Abschnitte die zu Fuß zurückgelegt werden, als auch für Teilstrecken des öffentlichen Verkehrsnetzes – inklusive zusätzlicher Informationen (z.B. Distanzen, Straßennamen, Hausnummern, etc.) benötigt. Damit basierend auf der aktuellen Uhrzeit auch die optimalen Routen im öffentlichen Verkehrsnetz gefunden werden können, müssen zusätzlich die Abfahrtszeiten in Form eines Fahrplans zur Verfügung stehen. Um eventuelle Verspätungen bei der Routenplanung zu berücksichtigen, sollten auch Live-Abfahrtszeiten verwendet werden. Da die Anwendung mit den auf einem Mobiltelefon verfügbaren Ressourcen auskommen soll, müssen die notwendigen Informationen in einer kompakten Datenstruktur abgelegt sowie ein effizientes Routingverfahren verwendet werden, um einen geeigneten Weg zu berechnen.

### 3.2. Systemüberblick

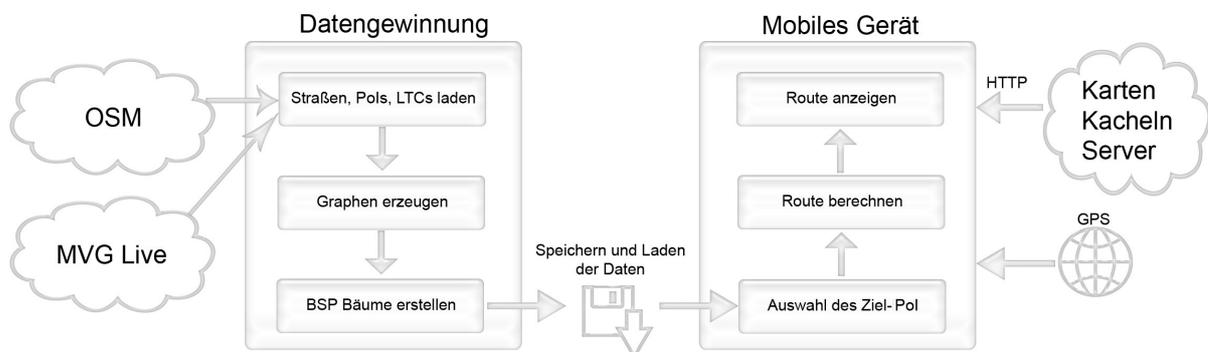


Abbildung 2: Systemarchitektur

Insgesamt kann das System in zwei Bereiche unterteilt werden, deren Architektur in Abbildung 2 veranschaulicht ist. Einerseits die Datengewinnung, bei der die notwendigen Information extrahiert, gefiltert und mit zusätzlichen Metadaten angereichert werden. Andererseits die mobile Komponente auf dem Handy, die diese Informationen zur Routenberechnung verwendet.

#### 3.2.1. Datengewinnung

In dieser Arbeit wurde exemplarisch der Großraum München modelliert. Dabei wurden die notwendigen Informationen aus dem OpenStreetMap Projekt (OSM) [10] extrahiert. Die für die Anwendung relevanten Daten bestehen aus dem Straßennetz (Straßen), wobei zu beachten ist, dass manche Typen von Straßen nicht über einen Bürgersteig verfügen und daher

nicht für unsere Anwendung in Frage kommen. Des Weiteren wurde das Streckennetz des öffentlichen Nahverkehrs (*Local Traffic Connections*, LTCs) verwendet. Um ein zuverlässiges Routing zu gewährleisten und Inkonsistenzen aufzulösen, wurde dieses mit Hilfe der über den MVG Live Service verfügbaren Informationen ergänzt. Einerseits handelt es sich dabei um eine eindeutige Bezeichnung der Haltestellen sowie die dort abfahrenden Linien und andererseits um Fahrplaninformationen bzw. die Zeitkosten zwischen zwei Stationen in Abhängigkeit von der Linie und der Zeit. Zusätzlich sollen bei der Zieleingabe so genannte *Points of Interest* (Pols) wie z.B. Apotheken, Kinos oder Theater berücksichtigt werden. Daneben enthalten die Pols auch Straßen- und Haltestellennamen.

Die ausgewählten, georeferenzierten Daten werden im folgenden Schritt in eine Graphenstruktur überführt. Dabei bilden alle mehrfach referenzierten Punkte der extrahierten Geodaten des Straßennetzes einen Knoten des Graphen. Zu diesem Zweck werden die Relation-, Way- und Node-Elemente der OSM Daten analysiert und entsprechend verarbeitet. Die Pols werden als Knoten modelliert, die der Einfachheit halber jeweils durch eine neu eingeführte Transferkante mit der nächstgelegenen Kante des Straßennetzes verbunden werden. Dabei wird zusätzlich ein neuer Knoten in die betroffene Kante eingefügt. Dadurch ist gewährleistet, dass das Routingverfahren auch Pols als Ziel akzeptiert und ein entsprechendes Ergebnis liefert. Der Graph für die LTCs wird auf die gleiche Weise erzeugt, wobei die Knoten hierbei eine Haltestellen und die Kanten die jeweiligen Streckenabschnitte der einzelnen Linien repräsentieren. Zusätzlich existiert im Gegensatz zum Straßengraphen jeweils eine Kante für jede Richtung jeder Linie, da deren Gewicht von der jeweiligen Richtung und zusätzlich von der aktuellen Uhrzeit abhängt. Auch dieser Graph wird durch Transferkanten von den Haltestellen aus mit dem Straßengraphen verbunden, wodurch der Suchalgorithmus beide Netze untersuchen kann. Des Weiteren werden so genannte Superkanten eingeführt, die die Route einer Linie in Form ihrer Haltestellen repräsentieren und Informationen über die Start- bzw. Zielstation sowie die Fahrplaninformationen enthalten. Dadurch kann zum einen Speicherplatz gespart werden und bei entsprechender Referenzierung muss zum anderen nicht der gesamte LTC-Graph im Speicher vorliegen um beispielsweise die Richtung einer Linie zu bestimmen. Insgesamt entstehen dadurch also drei unabhängige, georeferenzierte Graphen die mit Hilfe von Transferkanten verbunden sind.

Eine wichtige Technik zur Reduzierung der Menge der Navigationsdaten, die gleichzeitig im stark limitierten Arbeitsspeicher des Zielgeräts bereit liegen müssen, ist die räumliche Abgrenzung verschiedener Kartenbereiche. Zu diesem Zweck wird das Binary Space Partitioning (BSP) Verfahren eingesetzt, wobei bezüglich der Funktionsweise und Erstellung eines BSP-Baumes auf [11] verwiesen sei. Im Wesentlichen wird dabei der Raum mit Hilfe von Geraden sukzessive in Halbebenen unterteilt, die eine möglichst ausgeglichene Anzahl an Knoten beinhalten sollten. Zusätzlich wird in unserem Fall darauf geachtet, dass auch die Anzahl der Kanten, die durch eine solche Gerade geschnitten werden, minimal ist. Die Geraden werden dabei für den Straßengraphen ermittelt und für die anderen entsprechend übernommen. Diese Technik hat den Vorteil, dass nur so viele Knoten nachgeladen werden müssen, bis der Start- und Zielpunkt im Speicher vorliegen, wodurch in vielen Fällen ein wesentlich kleinerer Suchraum entsteht. Anschließend werden diese Daten in einem kompakten Format gespeichert. Das entwickelte Format unterstützt dabei die Anbindung zusätzlicher Ressourcen, wodurch beispielsweise auch Informationen über die Barrierefreiheit von der Routenplanung berücksichtigt werden könnten.

### 3.2.2. Mobile Anwendung

Die wesentlichen Aufgaben der mobilen Anwendungen beziehen sich zum einen auf die Interaktion mit dem Benutzer und zum anderen auf die Suche nach einer idealen Route. Die aktuelle Position des Benutzers wird mittels Global Positioning System (GPS) ermittelt und auf entsprechenden Kartenausschnitten angezeigt. Zusätzlich dient diese als Startpunkt der zu ermittelnden Route. Sobald die Position festgestellt wurde, wird der unterste Knoten im BSP-Baum geladen, der diese einschließt. Die Interaktion mit dem Benutzer beschränkt sich im Wesentlichen auf die Eingabe des Ziels, wobei entweder ein Straßename eingegeben oder das Ziel aus den Pols ausgewählt werden kann. Basierend auf dieser Information wird der BSP-Baum so lange expandiert, bis das eingegebene Ziel aufgelöst wurde und somit im navigierbaren Bereich liegt. In unserem Fokus liegt zunächst die Berechnung des schnellsten Weges, wobei basierend auf dem Datenformat auch alternative Einstellungen (nur Busse, Barrierefreiheit, etc.) denkbar wären. Zur Routenberechnung haben wir uns, wegen der guten durchschnittlichen Performance, für den A\* Algorithmus entschieden, dessen Funktionsweise in [12] beschrieben ist. Auf Grund der Charakteristik des LTC Graphen, bei dem schon eine leichte Abweichung von der geplanten Ankunftszeit zu erheblichen Wartezeiten führen kann, kann es vorkommen, dass die Route während einer Reise öfters neu berechnet werden muss. Ein Vergleich unterschiedlicher Verfahren bezüglich der Laufzeit findet sich unter anderem in der Arbeit von Cherkassky et. al. [13].

## 4. Implementierung und Evaluierung

### 4.1. Datengewinnung

Die Routen der öffentlichen Verkehrsmittellinien liegen als Relationen in den OSM Daten vor. Mehrere Vergleiche mit den tatsächlichen Fahrplänen der MVG zeigten, dass die Reihenfolge der Stationen darin inkonsistent ist und manche gar nicht vorhanden sind. Zur Aufbereitung der Reihenfolge wurde ein Algorithmus entwickelt, der von den Haltestellen und den Wegstücken dazwischen ausgehend die Route so zusammensetzt, dass alle auf dem Weg liegende Stationen eingebunden werden konnten, auch wenn diese nicht modelliert worden sind. Die dadurch erzeugte Reihenfolge ist bis auf wenige Ausnahmen identisch mit den Fahrplänen der MVG. Zusätzlich zu den in der Relation referenzierten Stationen wurden sämtliche in den OSM Daten vorkommenden Haltestellen eingebunden, wenn sie zu der jeweiligen Linie passen, im Sinne dass das „ref“-Tag den Namen der Linie enthält. Darüber hinaus werden auch Stationen ohne „ref“-Tag zu einer Linie hinzugefügt, wenn laut MVG Daten die jeweilige Linie dort abfährt. Die Daten für die Fahrpläne des öffentlichen Nahverkehrs wurden über den MVG Live Service gewonnen und mit den Haltestellen verbunden.

In weiteren Schritten werden die Routen der Linien in Hin- und Rückrichtung aufgeteilt, die jeweils relevanten Abfahrtszeiten zugewiesen und darauf die Verzögerung zwischen den Stationen berechnet, welche später als Kantengewicht verwendet wird. Viele der bis dahin gespeicherten Relationen werden verworfen, wenn keine Abfahrtszeiten verfügbar sind. Dabei handelte es sich im Wesentlichen um selten fahrende Verkehrsmittel wie Regionalbusse oder Fernverkehrszüge.

Im Folgenden wird das so gewonnene Straßen- und Verkehrsmittelnetz sowie die Liste der Pols mit dem Binary Space Partitioning Algorithmus räumlich aufgeteilt. Die somit erzeugten

Daten werden schließlich in einem binären Format gespeichert, welches die für multimodales Routing im Raum München nötigen Informationen in ca. 12 MB kompakt zusammenfasst.

## 4.2. Mobile Anwendung

Diese Daten werden von einer prototypischen mobilen Anwendung für die multimodale Navigation verwendet. Hierbei kann der Benutzer den Namen seines Ziels oder dessen Anfangsbuchstaben eingeben und aus entsprechenden Vorschlägen wählen. Dabei ist der Ausgangspunkt der Suche der Knoten, dem das zuletzt empfangene GPS-Signal am Nächsten liegt. Daraufhin wird der Straßen- und Verkehrsmittelgraph soweit geladen, dass Ursprung und Ziel im Speicher vorliegen. Anschließend beginnt der A\* Algorithmus mit der Suche nach der schnellsten Route. Die darin enthaltenen Kanten werden mit den zugeordneten Metadaten (Zeiten, Distanzen, Namen, Linien, etc.) angezeigt. Sobald eine wesentliche räumliche oder zeitliche Abweichung von der geplanten Strecke erkannt wird, muss die Route vollständig neu berechnet werden. Dadurch können eventuell auftretende Verspätungen oder das Verpassen einer Anschlussverbindung berücksichtigt werden.

## 4.3. Evaluierung

Insgesamt lässt sich bezüglich der Vollständigkeit im ausgewählten Bereich festhalten, dass derzeit etwa 850 von 1090 uns bekannten Haltestellen (ca. 78%) erfasst sind und die daraus entstandenen Inkonsistenzen automatisiert behoben werden konnten. Die prototypische Anwendung wurde an vielen Orten im Großraum München erfolgreich getestet, wobei im Wesentlichen zwei Mängel auffielen: Erstens wird zwar immer eine Route zum Ziel gefunden, jedoch besteht diese fast immer nur aus Fußwegen – öffentliche Verkehrsmittel wurden kaum benutzt. Dies begründet sich in der Verwendung des Suchalgorithmus, der nur dann eine optimale Lösung garantiert, wenn auf dem Graphen die Dreiecksungleichung gilt. Diese ist aber in multimodalen Verkehrsnetzen nicht überall gegeben, weshalb Haltestellen, die in der Gegenrichtung des Ziels liegen, nicht berücksichtigt werden, obwohl von dort aus eine viel schnellere Verbindung besteht. Der für multimodales Routing besser geeignete Dijkstra Algorithmus ist jedoch wesentlich weniger performant und dadurch nur für kleinere Suchräume geeignet. Zweitens muss die Anwendung im ungünstigsten Fall – nämlich dann, dass die Start- und Endpunkt auf anderen Seiten des Wurzelknotens im BSP-Baum liegen – die gesamte Datenbank laden. Das Routingverfahren trägt im Vergleich dazu nur mit einem Bruchteil zur Ausführungszeit bei.

## 5. Zusammenfassung und Ausblick

Insgesamt haben wir mit dieser Arbeit gezeigt, dass multimodale Routenplanung auf den OpenStreetMap-Daten möglich ist und auch Methoden aufgezeigt, mit denen die Navigations-Qualität der Rohdaten automatisiert entscheidend erhöht werden konnte. So waren wir in der Lage, fast alle Inkonsistenzen in der Modellierung des öffentlichen Nahverkehrs durch Fusion mit öffentlich zugänglichen Daten der Betreiber zu klären und ein für die Navigation ausreichendes Nahverkehrswegenetz zu konstruieren. Damit solche Lösungen nicht auf einzelne Städte beschränkt bleiben, muss die Berechnung der Routen lokal auf dem Endgerät erfolgen. So kann durch die Auswahl der benötigten Daten Skalierbarkeit und Zuverlässigkeit im Vergleich zu reinen Client-Server Architekturen erhöht werden. Darüber hinaus ergeben

sich Vorteile im Bezug auf den effektiven Schutz der Privatsphäre der Nutzer und damit auf die Akzeptanz des Systems.

Die algorithmischen Schwachstellen betreffen hauptsächlich das Verfahren zur Suche nach dem schnellsten Weg. Hierbei kann eine spatiale Organisation der Daten je nach verwendbarem Algorithmus und benötigter Umgebungsgröße zu enorm verbesserten Berechnungszeiten führen. So sollten grundsätzlich die Zugangspunkte zum öffentlichen Nahverkehr in der Nähe des Startpunktes und Endpunktes – nach der Entfernung sortiert – betrachtet werden und zwischen allen in einer zumutbaren Entfernung zum Start- und Endpunkt liegenden Punkten mit dem Dijkstra-Algorithmus die insgesamt beste Verbindung unter Berücksichtigung der Taktung der Verkehrsmittel errechnet werden. Diese (oder auch die k Besten, die sich wesentlich unterscheiden) müssen dann durch Fußgängernavigation mit dem A\*-Algorithmus zu einer vollständigen Route ergänzt werden und mit einer direkten Fußgängerverbindung verglichen werden.

Durch die Unterstützung von verschiedensten Fortbewegungsmodalitäten ist es ohne Weiteres möglich, beliebige Einschränkungen in der Wegwahl (Tickets, körperliche Einschränkungen, persönliche Präferenzen) zu modellieren. Hierzu muss lediglich die ohnehin nicht konstante Bewertung des Graphen entsprechend beeinflusst werden. Leider liegen auch im OpenStreetMap-Projekt und insbesondere auch in den Live-Daten der Verkehrsbetriebe noch keine ausreichenden Informationen über behindertengerechte Zugänge und Transportmittel vor.

Bei der Definition der spatialen Ordnungsstruktur für die Daten sollte die Möglichkeit, die Daten sequentiell und partiell zu laden, bestehen bleiben. So kann dann gegebenenfalls erst dann eine Datenzelle nachgeladen werden, wenn Sie auch vom Suchalgorithmus benötigt wird. Hierbei kann auch eine Anpassung des Dijkstra-Suchalgorithmus den Zeitpunkt des Ladens neuer Zellen möglichst weit hinauszögern. Die Integration von anderen Live-Daten im Zusammenhang mit dem Weg (Landmarken, interessante Punkte, Tarifinformationen und gegebenenfalls auch digitaler Ticketkauf) ist für die Akzeptanz bei den Nutzern wichtig, eine systematische Untersuchung des Bedarfs und eine technische Realisierung würde aber den Rahmen dieses Beitrages sprengen.

Kontakt zu den Autoren:

Markus Duchon	Andreas Paulus	Martin Werner
Siemens AG, CT T DE IT1	LMU München	LMU München
Otto-Hahn-Ring 6	Oettingenstraße 67	Oettingenstraße 67
D-80200 München	D-80538 München	D-80538 München
markus.duchon.ext@siemens.com	paulus.andreas@gmx.net	martin.werner@ifi.lmu.de

Literatur

- [1] *Duchon, M.*: A Context-Aware Mobile Traveller Information System based on OpenStreetMap, GI/KuVS Fachgespräch Berlin, 2010.
- [2] *Riebl, C.*; Büro für Verkehrsökologie: Akzeptanzuntersuchung zu Fahrgastinformationssystemen der Bremer Straßenbahn AG, [online] [http://www.durbit.se/Archives/ExternalPDF/Deliverables/PhaseII/Domain4/P2D\\_A41.3\\_STUDY.pdf](http://www.durbit.se/Archives/ExternalPDF/Deliverables/PhaseII/Domain4/P2D_A41.3_STUDY.pdf), (aufgerufen am 05.02.2011)
- [3] Barcelona Local Transport Train, Metro, and Bus <http://www.tmb.cat> (aufgerufen am 05.02.2011)
- [4] Massachusetts Bay Transportation Authority, <http://www.mbta.com> (aufgerufen am 05.02.2011)
- [5] Münchner Verkehrsgesellschaft mbH, <http://www.mvg-mobil.de> (aufgerufen am 05.02.2011)
- [6] *Ferris, B.; Watkins, K.; Borning, A.*: Location-Aware Tools for Improving Public Transit Usability. IEEE Pervasive Computing, 9:13–19, 2009.
- [7] *Esters, D.*; PTV AG: Schlussbericht zu DOM – Der Orientierte Mensch, [online] [http://www.tuvpt.de/fileadmin/pdf/Mobidienste/DOM\\_Abschlussbericht\\_PTV.pdf](http://www.tuvpt.de/fileadmin/pdf/Mobidienste/DOM_Abschlussbericht_PTV.pdf), (aufgerufen am 05.02.2011)
- [8] *Becirovic, E.*: Location Based Services - Entwicklung eines Prototypen einer mobilen Fahrplanauskunft. Diplomarbeit, Institut für Informatik, Ludwigs-Maximilians-Universität, 2010.
- [9] *K. Rehrl, K.; Brunsch, S.; Mentz, H.J.*: Assisting multimodal travelers: Design and prototypical implementation of a personal travel companion. IEEE Transactions on Intelligent Transportation Systems, 8(1):31–42, 2007
- [10] [http://wiki.openstreetmap.org/wiki/Main\\_Page](http://wiki.openstreetmap.org/wiki/Main_Page) (aufgerufen am 05.02.2011)
- [11] *de Berg, M.; Cheong, O.; Van Kreveld, M.; Overmars, M.*: Computational Geometry – Algorithms and Applications:259-279, Springer 2000
- [12] *Hart, P. E.; Nilsson, N. J.; Raphael, B.*: A Formal Basis for the Heuristic Determination of Minimum Cost Paths, IEEE Transactions on Systems Science and Cybernetics 4(2):100-107, 1968
- [13] *Cherkassky, B.V.; Goldberg, A.V.; Radzik, T.*: Shortest paths algorithms: theory and experimental evaluation. Mathematical programming, 73(2):129–174, 1996.