

Heuristiken zur Berechnung effizienter Mobilitätsketten und zu beachtende Herausforderungen

Konstantin Gaarz
Technische Hochschule Koeln
Gummersbach, NRW
konstantin.gaarz@smail.th-koeln.de

Julian Arera
Technische Hochschule Koeln
Gummersbach, NRW
julian_alexander.arera@smail.th-koeln.de

ABSTRACT

Das Studienprojekt *Trabit* befasst sich mit der Verknüpfung von Mobilitätsmöglichkeiten. Mit Hilfe von Heuristiken werden effiziente Mobilitätsketten gebildet, die zu einer möglichst optimalen Route führen und dabei die Laufzeit des Algorithmus, sowie die Nutzung externer APIs, schonen. Die Implementation der Heuristiken basiert auf dem theoretischen Prinzip des *MVC-Paradigmas*, welches Software-Code in die Komponenten der *model*, *view* und *controller* aufteilt. Abschließend werden entstandene Herausforderungen dargestellt und diskutiert. Diese beziehen sich auf die Automatisierung und Verknüpfung des Bezahlvorganges, die Unvorhersehbarkeit der Verkehrslage und die Verknüpfung des Fahrrads mit öffentlichen Verkehrsmitteln.

KEYWORDS

Mobilität, Heuristiken, Routengenerierung, Vernetzung

ACM Reference Format:

Konstantin Gaarz and Julian Arera. 2019. Heuristiken zur Berechnung effizienter Mobilitätsketten und zu beachtende Herausforderungen. In *Proceedings of (Technische Hochschule Koeln)*. Vertiefungsworkshop 2 - Veröffentlichung und Förderung, Gummersbach, NRW, Deutschland, 6 pages.

1 KONTEXT

Das Schlagwort „Mobilität“ ist durch den Bewusstseinwandel in Deutschland, zu einer umweltbewussteren und nachhaltigeren Nutzung von Mobilitätsmöglichkeiten, in den letzten Jahren immer präsenter geworden. Das *Bundesministerium für Verkehr und digitale Infrastruktur* (BMVI) stellt in ihrem jährlichen Bericht „Verkehr in Zahlen“ einen Rekordwert an zurückgelegten Wegen in Deutschland fest und das die transportierten Personen im Öffentlichen Personennahverkehr (ÖPNV) ähnliche Werte erreicht haben [2]. Hinzu kommt eine rückläufige Autonutzung der jungen Generation zwischen 18 und 25 Jahren, die offener für alternative Angebote werden [1]. Ein großes Problem bei der Nutzung verschiedener Mobilitätsangebote ist jedoch die mangelhafte Qualität und Umfang des Informationsangebotes der Anbieter [6]. Dem Fahrgast fehlt somit die Um- bzw. Weitsicht, verschiedene Angebote wahrzunehmen und als Alternative in Betracht ziehen zu können. Außerdem

würde sich durch ein hochwertiges Informationsangebot eine Verknüpfung dieser Informationen anbieten. Effiziente Mobilitätsketten könnten geschaffen werden, die den Fahrgast bei seiner Reise unterstützen und im Störfall auf Alternativen aufmerksam machen.

Dieser Problematik wurde sich im Studienprojekt *Trabit* angenähert. Das Projekt ist im Medieninformatik Master der Technischen Hochschule Köln entstanden und beschäftigt sich seit 2018 mit der Konzeptualisierung, Entwicklung und Evaluation einer Softwarelösung für das zuvor beschriebene Problem. Der *Trabit*-Algorithmus generiert auf Basis verschiedenster Mobilitätsmöglichkeiten effiziente Mobilitätsketten und Routen. Außerdem bietet es die Möglichkeit Alternativrouten im Störfall zu generieren. Die Anreicherung der Störungsdaten wird über die eigens benutzergenerierten Störungsdaten umgesetzt, die mit Hilfe eines Glaubwürdigkeitsindex verifiziert werden.

Der Fokus dieser Arbeit liegt auf der Berechnung der effizienten Mobilitätsketten. In diesem Zusammenhang werden Heuristiken dargestellt, die zur Generierung der Routen eingesetzt werden, um gewissen Problematiken entgegen zu wirken und zu vereinfachen. Zu Beginn werden dazu verschiedene im Zusammenhang stehende Arbeiten betrachtet, die in die generelle Thematik einleiten und einen Überblick über die Herausforderungen und Problematiken im Kontext der Mobilität schaffen. Dieser Überblick soll anschließend als Basis für die identifizierten Heuristiken dienen, um deren Bedeutsamkeit herauszustellen. Angesichts der angewandten Heuristiken in *Trabit* werden zu erst die Problematiken erläutert und anschließend die Lösungsstrategien erläutert. Im Anschluss wird die technische Implementierung der Heuristiken dargestellt. Dafür werden zu Beginn des Abschnittes die Grundlagen des Codes erläutert, die auf Basis des *Model-View-Controller-Paradigmas* entwickelt wurden. Auf die theoretische Darstellung des Paradigmas folgt die praktische Implementierung der Heuristiken in den Code. Zum Abschluss werden die generellen Herausforderungen identifiziert, die auf den *Trabit*-Algorithmus einwirken. Hierbei werden hauptsächlich Aspekte erläutert, die bisher weniger Betrachtung bei der Entwicklung der Heuristiken erhalten haben und im Ansatz Lösungsstrategien dargestellt und im Fazit diskutiert.

2 RELATED WORKS

In diesem Abschnitt werden die Werke von *Wolter*, *Bratzel* und *Hamöller and Bäsch* bezüglich ihrer relevanten Aspekte für die Heuristiken und Herausforderungen effizienter Mobilitätsketten vorgestellt und zusammengefasst.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

Technische Hochschule Koeln, Vertiefungsworkshop 2 - Veröffentlichung und Förderung, WS2019/20

© Copyright held by the owner/author(s).

2.1 Smart Mobility - Intelligente Vernetzung der Verkehrsangebote in Großstädten

Wolter beschreibt in seiner Arbeit eine Informations- und Kommunikationstechnologie (IKT), die bestehende Angebote in der Mobilitätswelt aufgreift und miteinander in Verbindung setzt. Die Vision, die *Wolter* mit Smart Mobility hat, weist viele Parallelen zu der von *Trabit* auf. Durch die Bündelung der Informationen soll ein System geschaffen werden, das auf Basis dieser Daten auf Gefahren und Verkehrsstörungen hinweist und intelligent eine Alternativroute berechnet. Die Vision von Smart Mobility umfasst außerdem den automatisierten Bezahlungsprozess von öffentlichen Verkehrsmitteln, sodass hier das Spektrum der Idee über das von *Trabit* hinaus geht.

Die Automatisierung des Bezahlvorgangs wird von *Wolter* in drei Entwicklungsstufen unterteilt. Die erste Stufe bezieht sich auf das bargeldlose Bezahlen, Stufe zwei auf das einfache elektronische Ticketing (Chipkarte o.Ä.) und die letzte Entwicklungsstufe stellt einen vollwertigen elektronischen Ticketbezahlprozess dar. Die Vollwertigkeit umfasst hierbei den An- bzw. Abmeldevorgang und die automatische Fahrpreiserkennung [vgl. 6, S.531]. Die Automatisierung des Bezahlvorganges soll primär zu einer direkten und schnelleren Alternative zum motorisierten Individualverkehr (bspw. Auto) führen. *Wolter* beschreibt die Wichtigkeit der Automatisierung wie folgt: „Ein Autofahrer wird sich nur dann mit Alternativen befassen, wenn ihm diese mündgerecht dargeboten werden und wenn die Nutzung genauso einfach, sicher und kostengünstig ist, wie das Einsteigen in das vor dem Haus parkende Auto, das Fahren, ggf. mit Navigationssystem und das anschließende Parken.“ [6, S.534]. Der zum Teil schwerfällige Bezahlvorgang im öffentlichen Personennahverkehr würde durch die Automatisierung des Bezahlvorganges somit als ernstere Alternative in Betracht gezogen werden können.

Anschließend greift *Wolter* die bestehenden Abfahrtsinformationen auf und beschreibt den Status quo des Informationsangebotes. Der Kunde wird in den meisten Verkehrsverbünden noch nicht über den aktuellen Zustand von Bus und Bahn informiert. Falls Informationen gesammelt werden, werden diese, laut *Wolter*, direkt an die dynamischen Fahrplananzeigen weitergeleitet. Jedoch sind die dynamischen Fahrplananzeigen nicht an allen Haltestellen installiert und nur die Personen vor Ort profitieren von diesem Angebot. Außerdem ist die Qualität dieser Informationen meist mangelhaft. Die angezeigte Abfahrtszeit wird durch das Passieren von Meldepunkten berechnet. Kommt es zwischen zwei Meldepunkten zu einer Störung, wird die Zeit durch diese nicht beeinflusst, so lange bis die Mobilitätsmöglichkeit den nächsten Meldepunkt erreicht. Für eine detailliertere Verkehrslage schlägt *Wolter* die Nutzung von herkömmlichen Detektoren, das Geschwindigkeit messen oder die Nutzung von Floating Car bzw. Floating Car Observer Daten vor [6, S.535]. Als weitere Alternative führt *Wolter* die Nutzung von GPS-Trackern auf und verweist hierbei auf den Einsatz dieser in Helsinki oder amerikanischen Großstädten. Dadurch wird den Fahrgästen der aktuelle Standort der Mobilitätsmöglichkeit in Echtzeit zur Verfügung gestellt. Eine kostengünstigere Alternative, neben der Installation von GPS-Trackern, bietet die Lokalisierung durch

die Fahrgäste selbst. Per Smartphone-GPS werden Informationen über den Standort der Fahrgäste mit den Abfahrtszeiten abgeglichen und Daten über die Verkehrslage erhoben. Im Ausland wird diese Möglichkeit zu meist mit Hilfe eines kostenlosen Internetzuganges beworben [siehe 6, S.537].

Im Anschluss geht *Wolter* auf die Verknüpfung des Öffentlichen Verkehrs (ÖV) mit dem (Leih-)Fahrrad bzw. Car Sharing ein. Hinsichtlich der Verbindung mit dem Leihfahrrad stellt *Wolter* klar, Deutschland weist eine hohe private Fahrradverfügbarkeit auf und daher wird häufiger eine Kombination zwischen Leihfahrrad und ÖV festgestellt (Stuttgart 47% aller Fahrten kombiniert, Barcelona 20% [6, S.538]), da für „normale Strecken“ meist das eigene Fahrrad verwendet wird. Strategisch betrachtet sieht *Wolter* im Radverkehr das größte Substitutionspotenzial, im Hinblick auf die Verlagerung des PKW-Verkehrs. *Wolter* begründet diese These folgendermaßen: „Etwa die Hälfte der Pkw-Fahrten in Städten ist kürzer als 5 km - hier ist das Fahrrad nicht nur das sauberere und kostengünstigere, sondern oftmals auch flexiblere und schnellere Verkehrsmittel. Deshalb wird gerade im innerstädtischen Verkehr im Fahrrad das Fortbewegungsmittel der Zukunft gesehen – auch vor dem Hintergrund wachsenden Umwelt- und Gesundheitsbewusstseins sowie des demografischen und städtebaulichen Wandels.“ [6, S.539]. Hinsichtlich der Verknüpfung mit dem Car-Sharing stellt *Wolter* dar, dass verschiedene Untersuchungen zu dem Ergebnis geführt haben, dass Car-Sharing-Nutzer ihre Kurzstrecken vermehrt mit umweltschonenderen Verkehrsmitteln zurücklegen und eine deutlich geringere PKW-Fahrleistung aufweisen [vgl. 6, S.540]. Außerdem besitzen Car-Sharing-Nutzer überdurchschnittlich viele Jahreskarten im ÖPNV, so dass auch hier die Verbindung leichter hergestellt werden kann.

Wolter schlussfolgert, dass ein deutliches Informationsbedürfnis auf Seiten des Kunden besteht, dieses Bedürfnis jedoch mit nicht ausreichender Qualität gedeckt wird. Qualität und Umfang des Informationsangebotes sind im ÖPNV, wie auch im MIV unzureichend und reichen aktuell nicht aus um die Vision der Smart Mobility ausreichend umsetzen zu können.

2.2 Die junge Generation und das Automobil – Neue Kundenanforderungen an das Auto der Zukunft?

Bratzel stellt in seiner Arbeit die Verbindung zwischen dem Automobil und der jungen Generation (18-25 Jahre) dar und welche Anforderungen diese Generation an das Automobil hat. Der Autor beginnt mit einem historischen Abriss des Wandels der Automobilmuster der jungen Generation. Nach dem zweiten Weltkrieg entwickelte sich das Automobil zum Symbol des sozialen Aufstiegs und war mit Eigenschaften wie Unabhängigkeit und Anerkennung verbunden. Die heutige junge Generation ist, laut *Bratzel*, nicht weniger mobil, jedoch weist sie eine deutlich geringere Autoaffinität auf. Dieser Trend ist speziell in den Großstädten in größerem Ausmaß zu erkennen. So ging die tägliche Autonutzung in Großstädten (>100.000 Einwohner) zwischen 2002-2008 um rund 12% zurück [vgl. 1, S.96]. Neben der Nutzung hat sich jedoch auch die Einstellung der jungen Generation gegenüber dem Automobil verändert. Die

junge Generation sieht das Automobil immer pragmatischer und stellt eine geringere emotionale Bindung bzw. Mehrwert zum Automobil her. *Bratzel* stellt fest, dass das Automobil als Statussymbol keine wichtige Rolle mehr spielt. Trotzdem ist festzuhalten, dass auch *Bratzel* eine weiterhin positive Einstellung zum Automobil darstellt. So sehen drei Viertel der Befragten einer Umfrage von *Bratzel* das Automobil weiterhin als wichtig bzw. sehr wichtig an. Diese Wichtigkeit hat jedoch vermehrt pragmatische Gründe und das Automobil wird grundsätzlich als Gebrauchsgegenstand angesehen.

Als Einflussfaktoren für den Wandel der jungen Generation hat *Bratzel* folgende Punkte identifiziert: Die steigenden Kosten für Kauf und Unterhalt von Automobilen, die zunehmenden Stau- und Parkplatzprobleme und das Smartphone als neues Statussymbol der jungen Generation. Des Weiteren begründet *Bratzel* den Wandel damit, dass die junge Generation den ÖV als ernsthafte Mobilitätsalternative wahrnimmt und attraktiver wird. Als wichtigste Faktoren für die Nutzung der öffentlichen Verkehrsmittel ermittelt *Bratzel* die Kosten, Fahrzeit und die Bequemlichkeit [vgl. 1, S.102].

Als Handlungskonsequenz für die Automobilhersteller sieht *Bratzel* hauptsächlich die Vernetzung des Fahrzeuges mit dem Internet und der generellen Erweiterung der digitalen Angebote. Durch die intensive Nutzung der digitalen Medien hat die junge Generation einen sehr hohen Anspruch an die neu aufkommenden Angebote.

2.3 RadSTRATEGIE Baden-Württemberg

Ein Gutachten aus dem Jahr 2016 vom Ministerium für Verkehr und Infrastruktur Baden-Württemberg hat sich mit der Verknüpfung des ÖV mit dem Fahrradverkehr beschäftigt. Aus diesem Gutachten ergibt sich die *RadSTRATEGIE* [vgl. 3, S.1], welche ein Förderungsprogramm für den Fahrradverkehr ist, das die Landesregierung Baden-Württemberg bis 2025 für das Bundesland umsetzen möchte. Konkret ist hier als Ziel anvisiert worden, den Radverkehrsanteil von 8 % zu verdoppeln [vgl. 3, S.1]. Den Hauptansatzpunkt bildet die Förderung der Verknüpfung von ÖV mit Fahrrad, also dem *B+R* (Bike And Ride). Dies soll sich um einen durch die Vorteile des Fahrrads rechtfertigen, welches ressourcenschonender und klimaschonender als das Auto ist. Zum anderen ergeben sich durch die Verknüpfung zusätzliche Vorteile für den ÖV; die Anreise an dessen Haltestellen kann beschleunigt werden, die Haltestelleneinzugsbereiche können ausgeweitet werden und nicht erschlossene Fahrziele können erreichbar gemacht werden. Durch die Hinzunahme von Leihfahrradsystemen sollen außerdem die Fahrradmitnahmekapazität des ÖV entlastet werden.

Im Rahmen des Gutachtens wurden außerdem Informationen hinsichtlich des sicheren Parkens des Fahrrads am Bahnhof und des erlaubten Mitnehmens des Fahrrads im ÖV zusammengestellt. So gibt es zum Abstellen des Fahrrads am Bahnhof einfache Fahrradstationen, größere Fahrradparkhäuser oder abschließbare Fahrradboxen. Kostentabellen des Gutachtens zeigen, dass „wegen des geringen Flächenbedarfs des Fahrrades [...] die Investitions- und Betriebskosten für die Erschließung von Bike+Ride-Anlagen in der Regel geringer als für Park+Ride-Anlagen“ [3, S.9] sind. Durch das Verhältnis von 1:9 hinsichtlich des benötigten Platzes eines Fahrrads

gegenüber eines Autos ergibt sich, dass ebenerdige Abstellplätze, Parkpaletten, Tiefgaragen und dessen monatlichen Betriebskosten für Fahrräder deutlich günstiger sind als für Autos [vgl. 3, S.9].

Hinsichtlich der Fahrradmitnahme in ÖV hat das Gutachten gezeigt, dass hier einige Rahmenbedingungen gibt, die zu beachten sind [vgl. 3, S.29]. Dazu zählt zunächst einmal die Barrierefreiheit des Bahnhofs, da es ohne die entsprechende Bebauung für Fahrräder zum Betreten der Gleise nicht möglich ist sein Fahrrad in den ÖV mitzuführen. Ist man mit seinem Fahrrad bis hierhin gekommen, ist die Mitnahmekapazität des jeweiligen ÖV entscheidend. Im Idealfall handelt es sich um einen Zug, welcher ein Mehrzweckabteil enthält. Des Weiteren kommt das Tarifsystem hinzu, welches dazu führen kann, dass bei Fahrradmitnahme zusätzliche Kosten entstehen können. Dies bezieht sich auch auf die Mitnahmezeiten für Fahrräder, da je nach Verkehrsverbund und je nach Uhrzeit eine kostenlose oder kostenpflichtige möglich ist oder gar keine Fahrradmitnahme möglich ist, da eine Sperrzeit aktiv ist [vgl. 3, S.33].

Schließlich kommen noch die Beförderungsbedingungen hinzu, die auch verbundsübergreifend sein können. So ist in vielen Bussen die Fahrradkapazität auf maximal zwei beschränkt. Dabei kommt die Problematik hinzu, dass hierzu auch Rollstühle und Kinderwagen gehören. In Mehrzweckabteilen liegt darüber hinaus die Beförderungspriorität bei Rollstuhlfahrenden und Kinderwagen [vgl. 3, S.32]. Außerdem stellt das Gutachten heraus, dass es keine Mitnahmegarantie für die Fahrräder gibt und im Nahverkehr auch keine Reservierungen zur Fahrradmitnahme möglich sind. Man erfährt also erst vor Ort, ob man sein Fahrrad wirklich mitnehmen darf.

3 ANGEWANDTE HEURISTIKEN IN TRABIT

Hinter dem System von *Trabit* steckt ein komplexer Algorithmus, welcher auf dem Server für jede Routenabfrage des Benutzers ausgeführt wird. Dieser Algorithmus ist dafür zuständig, die ideale Mobilitätskette für den Benutzer von seinen Startkoordinaten zu seinen Zielkoordinaten zu generieren. Eine Mobilitätskette besteht dabei aus einer Verknüpfung von Teilrouten, welche von einer externen API abgerufen werden. Je länger die Mobilitätskette ist, desto höher ist die Menge an Verkettungen von Teilrouten.

3.1 Problematik

Die Problematik, welche sich aus dem Algorithmus hinter *Trabit* ergibt ist vergleichbar mit der Problematik des *Traveling Salesman*. Hierbei gilt es, in einem Netzwerk den kürzesten Weg von A nach B zu finden. Durchläuft man mithilfe eines Algorithmus jedoch alle möglichen Wege, stellt man diesen vor eine unlösbare Aufgabe, da es schlichtweg zu viele mögliche Kombinationen an Wegen gibt. Da für jede Kreuzung oder Abzweigung, die man wählt, eine neue Auswahlmöglichkeit hinzukommt, ergibt sich ein Laufzeitverhalten von n^n .

Wenn man einen Algorithmus nutzt, der alle möglichen Kombinationen miteinander vergleicht, gilt ähnliches auch für *Trabit*. Eine Teilroute lässt sich mit verschiedenen Mobilitätsmöglichkeiten zurücklegen. Es handelt sich also um Verknüpfungen von Teilrouten, die jeweils mit verschiedenen Mobilitätsmöglichkeiten befahren

werden können, mit der jeweiligen darauffolgenden Teilroute, welche ebenfalls mit verschiedenen Mobilitätsmöglichkeiten befahren werden kann.

Zunächst soll es in *Trabit* möglich sein, zu Fuß, mit dem Fahrrad, Auto, Bike-Sharing, Car-Sharing oder öffentlichen Verkehrsmitteln zu reisen. Es handelt sich also um eine Basis von sechs verschiedenen Mobilitätsmöglichkeiten, die im Algorithmus berücksichtigt werden müssen. Daraus folgt ein Laufzeitverhalten von 6^n , welches schnell zu einer zu großen Menge an Kombinationsmöglichkeiten der Teilrouten führen kann. Daraus würde eine zu hohe Last resultieren, als dass sie die Rechenkapazität des Servers tragen könnte. Darüber hinaus folgt aus einer entsprechenden Menge an Kombinationsmöglichkeiten ein zu hohes Maß Anfragen für die jeweiligen Teilrouten, welche an externe APIs gesendet werden müssen. Dies kann zum einen zu Fehlern führen und zum anderen würde es zu hohe Kosten verursachen, welche an den Betreiber der API zu zahlen wären. Außerdem gibt es bei manchen APIs Limits. Diese Limits bestimmen, wie oft in der Minute der Server maximal eine Anfrage an die API senden darf.

3.2 Lösungsstrategien

Um dem Benutzer trotz dieser Problematik eine adäquate und effiziente Mobilitätskette präsentieren zu können, soll sich der idealen Lösung mit Hilfe von Heuristiken angenähert werden. Konkret sollen im Algorithmus drei Heuristiken implementiert werden. Diese wurden eigens für *Trabit* entwickelt und es handelt sich dabei um die *Heuristik der quantitativen Routengenerierung*, die *Heuristik der logischen Effizienzabfolge* und die *Heuristik der kleinsten Radian*.

Heuristik der quantitativen Routengenerierung

Als Grundlage für den gesamten Prozess der Bildung der Mobilitätsketten soll die *Heuristik der quantitativen Routengenerierung* dienen. Anhand dieses quantitativen Ansatzes soll auf Basis eine möglichst große Menge an unterschiedlichen Mobilitätsketten generiert werden. Die Menge soll jedoch so kontrolliert und klein genug gehalten werden, dass der Server nicht überlastet und die API-Kosten nicht zu hoch ausfallen. Dazu werden möglichst viele Teilstrecken, die mit verschiedenen Mobilitätsmöglichkeiten befahren werden, miteinander kombiniert. Je mehr verschiedene Teilrouten und verschiedene Umstiegsorte miteinander verknüpft werden, desto größer ist die Chance, dass die optimale Verkettung erfasst wurde. Neben den heterogenen Mobilitätsketten, die mindestens zwei verschiedene Mobilitätsmöglichkeiten beinhalten, gehören zu jeder Mobilitätskettenmenge auch alle verfügbaren homogenen Routen, also Routen, die nur eine Mobilitätsmöglichkeit beinhalten. Final wird aus der Gesamtmenge an Mobilitätsketten die Mobilitätskette mit der kürzesten Dauer identifiziert und an den Benutzer zurückgegeben. Diese Heuristik soll der Problematik des Laufzeitverhaltens und der zu intensiven Nutzung der externen APIs entgegenwirken. Die Kontrolle Mobilitätskettenmenge soll mit Hilfe der anderen beiden Heuristiken realisiert werden.

Heuristik der logischen Effizienzabfolge

Die *Heuristik der logischen Effizienzabfolge* dient vor allem dazu, die Laufzeit des Algorithmus und die API-Limits zu kontrollieren.

Dies funktioniert, indem die Menge der Kombinationsmöglichkeiten dahingehend reduziert wird, dass bestimmte Verknüpfung nicht abgefragt und somit auch nicht kombiniert werden. Allgemein lässt sich hier sagen, dass nie von einer Teilroute, die mit einer schnelleren Mobilitätsmöglichkeit zurückgelegt wird, zu einer Teilroute kombiniert wird, die mit einer langsameren Mobilitätsmöglichkeit zurückgelegt wird. Eine Ausnahme jedoch ist hier die Verknüpfung mit den öffentlichen Verkehrsmitteln, falls dabei eine lange Teilroute vorliegt, die mit dem Zug zurückgelegt wird. Normalerweise geht der Algorithmus davon aus, dass eine direkte Autofahrt über eine Teilroute effizienter ist, als mit einer Mobilitätsmöglichkeit zu einem Bahnhof zu gelangen und von dort mit dem Bus oder Zug die Teilroute zurückzulegen. Jedoch gibt es bestimmte Zugstrecken, gerade auf überregionalen Fahrten, die effizienter als die restlichen Mobilitätsmöglichkeiten sind und somit als Kombinationsmöglichkeit in die Menge der Mobilitätskettenmenge mit einfließen. Solche Zugstrecken werden identifiziert und zusätzlich mit den entsprechenden anderen Teilrouten verglichen. Schlussendlich sagt die *Heuristik der logischen Effizienzabfolge* aus, dass die folgende absteigende Reihenfolge die effizientesten bis ineffizientesten Mobilitätsmöglichkeiten darstellt.

Längste Teilzugstrecke < Auto < Carsharing < Öffentliche Verkehrsmittel < Fahrrad < Bikesharing < zu Fuß

Also wird außer von Teilrouten, die mittels der längsten Teilzugstrecke zurückgelegt werden, nie zu Teilrouten kombiniert, welche mit Mobilitätsmöglichkeiten zurückgelegt werden, die weiter rechts in der Reihenfolge liegen.

Heuristik der kleinsten Radian

Die *Heuristik der kleinsten Radian* bezieht sich konkret auf die Position von Bahnhöfen, sowie die Position der Angebote für Bike-Sharing und Car-Sharing. Über die jeweilige API lässt sich abfragen, an welchen Koordinaten sich die Objekte befinden. Diese Abfrage wird mittels eines Radius und Koordinaten, um die der Radius verläuft, welche beide vom Server bestimmt werden, vollzogen. Um zu ermitteln, welcher Bahnhof oder welches Sharing-Objekt bezüglich seiner Entfernung zum Benutzer und seiner Position bezüglich des Fahrtziels am effizientesten ist, müssten für alle sich im abgefragten Radius befindlichen Objekte Mobilitätsketten generiert und miteinander verglichen werden. Die *Heuristik der kleinsten Radian* soll jedoch die zu hohe daraus resultierende Laufzeit kontrollieren. Außerdem würden auch hier API-Limits überschritten werden, weshalb die Menge an Abfrage möglichst klein gehalten werden soll. Dazu werden am Startpunkt oder am Ende von Teilrouten die umliegenden Bahnhöfe und Sharing-Angebote mittels jeweils drei größer werdenden Radian nacheinander abgefragt. Sobald innerhalb eines Radius ein oder mehrere Objekte gefunden wurden, wird die Suche abgebrochen. Also werden für alle größeren Radian gar nicht erst Abfragen an die API gesendet. Anschließend müssen nur für die sich im kleinsten Radius befindlichen Objekte Mobilitätsketten generiert werden. Die gewählten Radian entsprechen außerdem jeweils dem für die Mobilitätskette entsprechenden Mehrwert des Sharing-Angebots oder Bahnhofs. Die Radian für die Angebote des Bike-Sharing sind also deutlich kleiner gewählt als für die Angebote des Car-Sharing oder der Bahnhöfe.

Abschließend muss mit Bezug auf die Verwendung der Heuristiken jedoch genannt werden, dass diese noch keine ideale Lösungsstrategie sind. Problematisch bei der Verwendung der Heuristiken ist, dass nicht alle Kombinationen berechnet werden, sondern nur eine, der Schätzung nach, sinnvollste Teilmenge. Es ist also möglich, dass effizientere Verkettungen gibt, die nicht entdeckt werden.

4 IMPLEMENTIERUNG DER HEURISTIKEN

Das *Model-View-Controller-Paradigma* (kurz MVC-Paradigma) wurde erstmals von Krasner *et al.* in der objektorientierten Programmiersprache Smalltalk-80 verwendet [5]. Ziel war es ein Entwicklungsmuster zu schaffen, das Systemkomponenten schafft, die den hochgradig interaktiven Softwareentwicklungsprozess unterstützen können. Außerdem sollten die Komponentensätze generalisiert werden, um Programmierern portable, wiederverwendbare Inhalte zur Verfügung zu stellen [vgl. 5, S.2]. Krasner *et al.* stellen außerdem heraus, dass die Modularität bei der Entwicklung von Software eine prägende Rolle spielt. Funktionalitäten sollen hierbei so gut wie möglich voneinander isoliert werden. Daraus ergeben sich die Vorteile, dass einzelne Komponenten einfacher modifiziert werden können, ohne das gesamte System zu gefährden. Des Weiteren brauchen Programmierer kein Wissen über das Gesamtsystem, sondern das Wissen über die Komponente bzw. zu bearbeitende Einheit genügt.

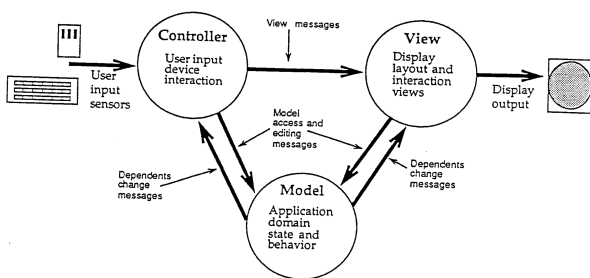


Abbildung 1: MVC-Abbildung nach Krasner *et al.* [5]

Das Paradigma besteht aus den Bestandteilen *model*, *view* und *controller* (siehe Abbildung 1). Das *model* übernimmt die Operationen, die sich auf die Anwendungsdomäne beziehen. Krasner *et al.* beschreiben ein *model* als Bestandteile einer Systemapplikation, die die Arbeit tatsächlich ausführen. Hartmann versteht unter diesem Aspekt „[...] dass mit dem Model eine wiederverwendbare Komponente geschaffen wird, in der das Domain-Spezifische einer Anwendung codiert ist“ [4, S.343]. Unter der *view* wird die Anzeige des Anwendungstatus verstanden. Darunter fallen alle grafischen Elemente des *model* und fordert dessen Daten, um diese visuell darstellen zu können. Neben den visuellen Aspekten können, laut Hartmann, jedoch auch die taktilen und auditiven Ausgaben eines Systems als *view* verstanden werden. Als letzten Bestandteil greifen Krasner *et al.* die *controller* auf. Diese beinhalten die Benutzerinteraktion und die daraus resultierende Aufforderung an das System, die an das *model* bzw. *view* weitergeleitet wird. Nachrichten werden an das *model* übermittelt und der *controller* bietet die Schnittstelle

zwischen dem *model* und den dazugehörigen *views* und deren Interaktionsobjekten (bspw. Maus, Tastatur, Touchscreen). *Controller* und *view* arbeiten dabei eng zusammen und ergeben ein sogenanntes *Controller-View-Tupel* [4, S.343]. Dieses Tupel steht genau zu einem *model* in Verbindung - ein *model* kann jedoch auch mehrere Beziehungen zu unterschiedlichen Tupeln haben (1:n-Beziehung).

Um nach dem Vorbild des MVC-Paradigmas auf dem Server von Trabit einen übersichtlichen und wiederverwendbaren Code zu erreichen, wurde eine Struktur geschaffen, in der jede JavaScript-Datei nur eine Verantwortung hat, auf andere Dateien referenziert und auch von anderen referenzierbar ist. Für die Kommunikation zwischen dem Client und dem Server wurde eine REST-Schnittstelle gewählt und implementiert. Für jede der fünf REST-Ressourcen gibt es eine *Controller*-Datei. Diese nehmen die mitgelieferten Daten des Clients entgegen, interagieren mit der Datenbank, oder rufen jeweils im Kontext benötigte Funktionen auf. Des Weiteren gibt es *Helper*-Dateien, welche Funktionen enthalten, die von den Controllern oder anderen *Helpern* verwendet werden können.

Der *Controller* namens *directionSelectionController* ist konkret für die Generierung der Mobilitätsketten verantwortlich. Von diesem wird dementsprechend der *Helper* aufgerufen, welcher unter Beachtung der dem Benutzer zur Verfügung stehenden Mobilitätsmöglichkeiten Teilrouten kombinieren sollen. Die *Heuristik der quantitativen Routengenerierung* wird also vom *directionSelectionController* angewandt. Denn dieser delegiert die Aufgabe, die möglichst große Menge an Mobilitätsketten zu generieren, und gibt anschließend die effizienteste von diesen an den Client zurück.

Die *Heuristik der logischen Effizienzabfolge* wird zum einen in jedem einzelnen *Helper* angewandt, ist aber auch schon dadurch angewandt, dass es *Helper* für bestimmte Kombinationen von Mobilitätsmöglichkeiten nicht gibt. So gibt es beispielsweise keinen *Helper* für die Kombination von dem Auto des Benutzers mit Car-Sharing-Objekten, da eine solche Kombination nach dieser Heuristik nicht effizient wäre. Darüber hinaus ist die Heuristik auch im Code der jeweiligen *Helper* angewandt. Hier ist festgelegt, dass bei der Generierung der Mobilitätsketten die in 3.2 erläuterte Reihenfolge der Effizienz von Mobilitätsmöglichkeiten eingehalten wird, sodass viele Kombinationen von Teilrouten ausgelassen werden können.

Die *Heuristik der kleinsten Radien* ist ebenfalls bei einigen, aber nicht allen, *Helpern* angewandt worden. Jeder *Helper*, der entweder eine homogene Route aus öffentlichen Verkehrsmitteln, Bike-Sharing oder Car-Sharing generiert oder eine Kombination einer beliebigen Mobilitätsmöglichkeit mit öffentlichen Verkehrsmitteln, Bike-Sharing oder Car-Sharing erstellt, verwendet die Heuristik. Denn hier müssen zu Beginn stets zuerst als Grundlage die jeweiligen Bahnhöfe und Sharing-Objekte im näheren Umkreis identifiziert werden. Die Heuristik findet sich also im ersten Teil des Codes all dieser *Helper* wieder.

5 HERAUSFORDERUNGEN

Im Folgenden werden Herausforderungen identifiziert, die auf den *Trabit*-Algorithmus einwirken und durch die gewonnenen Erkenntnisse der Recherche neu hinzugekommen sind. Außerdem werden teilweise Lösungsansätze beschrieben, die die Herausforderungen eingrenzen können.

Ein Aspekt, der bisher in *Trabit* weniger beachtet wurde ist die Automatisierung und Verknüpfung des Bezahlvorganges. Die Verknüpfung bezieht sich aktuell auf die Mobilitätsmöglichkeiten. Eine Vereinfachung des Bezahlvorganges könnte als weiterer Grund fungieren, um offener gegenüber Mobilitätsalternativen zu sein. Diese zusätzliche Herausforderung baut hauptsächlich auf den Überlegungen von *Wolter* auf. Wie auch schon bei der Verknüpfung der Mobilitätsmöglichkeiten wird auch beim Bezahlvorgang die Informationsqualität und die Zahlungsangebote nicht ausreichend sein, um eine Automatisierung vollständig umzusetzen. Durch die monetäre Komponente bei der Automatisierung sollte hierbei auf die Vollständigkeit geachtet werden und eine Implementierung von Bezahlvorgang-Heuristiken wäre unangebracht.

Eine weitere Herausforderung die den *Trabit*-Algorithmus beeinträchtigt ist die Unvorhersehbarkeit der Verkehrslage des ÖPNV. Dies führt dazu, dass Prognose schwerer umzusetzen sind. Die Einflussfaktoren, die auf die Verkehrslage einwirken sind dabei vielfältig und können aktuell im Algorithmus nur durch die Heuristiken eingegrenzt werden.

Aus dem Gutachten der Landesregierung Baden-Württembergs (siehe 2.3) gehen auch Probleme und Hindernisse bezüglich des ÖV hervor, die konkret die Funktionalität von *Trabit* betreffen. So geht dort hervor, dass noch nicht vom ÖV erschlossene Fahrtziele erreichbar gemacht werden müssen. Dabei stehen die Verknüpfungen des ÖV mit den eigenen Mobilitätsmöglichkeiten der Reisenden im Fokus. *Trabit* setzt hier konkret an und sieht bisher die Verknüpfung des ÖV mit dem Fahrrad oder Auto des Benutzers als Optionen vor. Auch die in der *RadSTRATEGIE* anvisierten Verknüpfungen mit Sharing-Angeboten ist in *Trabit* bereits ebenfalls eine Option. Problematischer ist allerdings, dass aus dem Gutachten hervorgeht, dass die Fahrradmitnahmekapazitäten des ÖV entlastet werden müssen. Denn die verschiedenen Mittel des ÖV bieten nicht immer die notwendigen Kapazitäten, sodass ein Benutzer nicht immer mit seinem Fahrrad einsteigen kann. Gerade bei Bussen ist die Situation hier kritisch. Hinzu kommen einige allgemeine Probleme und Hindernisse bei der Fahrradmitnahme im ÖV. So kann es gerade zu Uhrzeiten, zu denen viele zur Arbeit fahren, Sperrzeiten geben, in denen es allgemein nicht erlaubt ist, sein Fahrrad mit in den Bus zu nehmen. Des Weiteren können diese Sperrzeiten, wie auch zusätzliche Kosten für die Fahrradmitnahme, je nach Verkehrsverbund variieren. Mit Blick auf *Trabit* erschwert dies die Anpassung des Systems an die erlaubte Fahrradmitnahme und die damit verbundenen preislichen Auswirkungen auf den Benutzer nochmals. Da auch keine Reservierungen für die Fahrradmitnahme möglich sind, gibt es keine Mitnahmegarantie, die *Trabit* gegenüber seinen Benutzern aussprechen kann, wodurch dieser Aspekt des Systems aktuell als unzuverlässig einzustufen ist. Um zu einer sichereren Lösung

zu gelangen, müsste der Algorithmus zur Bildung der Mobilitätsketten von *Trabit* bei der Verknüpfung mit ÖV also vermehrt auf Leihfahrradsysteme setzen. So könnte man zu einer Entlastung der Fahrradmitnahmekapazitäten des ÖV gelangen und gleichzeitig den weiteren genannten Hindernissen, welche die Fahrradmitnahme mit sich bringt, aus dem Weg zu gehen.

6 FAZIT

Die *Related Works* zeigen, dass die Problematiken, denen *Trabit* entgegenzuwirken versucht, vorhanden sind und dass es bereits Ideen zur Lösung dieser gibt. Gleichzeitig stellen diese einige Hindernisse heraus, welche es dem Algorithmus von *Trabit* erschweren, zuverlässige Verknüpfungen mit dem ÖV herzustellen und dabei deutschlandweit einen adäquaten Bezahlvorgang zu implementieren. Gerade bei Verknüpfungen zwischen dem ÖV und dem Fahrrad gibt es bezüglich der Mitnahmegarantie noch Schwierigkeiten, weshalb vor allem die Option des Bike-Sharing bei der Bildung der Mobilitätsketten an Bedeutung gewinnt.

Die ursprüngliche Problematik des Lauzeitverhaltens des Algorithmus und der Nutzung der externen APIs kann mit Hilfe der drei Heuristiken erfolgreich gegengewirkt werden. Durch eine Orientierung an dem MVC-Paradigma konnte eine Struktur geschaffen werden, in welche die Heuristiken implementiert werden konnten. So wird eine, der Schätzung durch die Heuristiken nach, sinnvolle Teilmenge der möglichen Mobilitätsketten generiert, bei der jedoch beachtet werden muss, dass es möglicherweise effizientere Verkettungen gibt, die nicht entdeckt werden.

LITERATUR

- [1] Stefan Bratzel. 2014. *Die junge Generation und das Automobil – Neue Kundenanforderungen an das Auto der Zukunft?* Springer Berlin Heidelberg, Berlin, Heidelberg, 93–108. https://doi.org/10.1007/978-3-642-34068-0_6
- [2] BMVI Bundesministerium für Verkehr und digitale Infrastruktur. 2018. Verkehr in Zahlen 2018/2019. <https://www.bmvi.de/SharedDocs/DE/Publikationen/G/verkehr-in-zahlen%5F2018-pdf.pdf?%5F%5Fblob=publicationFile>
- [3] Gerald Hamöller and Christine Bäsch. 2016. *RadSTRATEGIE Baden-Württemberg - Verknüpfung ÖV und Fahrrad. Ministerium für Verkehr und Infrastruktur Baden-Württemberg 1* (2016).
- [4] Gerhard Hartmann. 2016. Vorlesungsbegleitende Materialien zum Modul Mensch-Computer Interaktion. (Feb. 2016). <https://th-koeln.sciebo.de/s/fidbhi268BBOYt9> unpublished.
- [5] Glenn E Krasner, Stephen T Pope, et al. 1988. A description of the model-view-controller user interface paradigm in the smalltalk-80 system. *Journal of object oriented programming* 1, 3 (1988), 26–49.
- [6] Stefan Wolter. 2012. *Smart Mobility- Intelligente Vernetzung der Verkehrsangebote in Großstädten*. Gabler Verlag, Wiesbaden, 527–548. https://doi.org/10.1007/978-3-8349-7117-3_42