Simulation of South African minibus taxis

Masterarbeit im Studiengang "Planung und Betrieb im Verkehrswesen"

am:
FG Verkehrssystemplanung und Verkehrstelematik
(Institut für Land- und Seeverkehr, Fakultät V)
Technische Universität Berlin

Autor:
Daniel Röder, B. Sc.
Matr.-Nr.: 221 801

Betreuer:
Prof. Dr. Kai Nagel
Dipl.-Ing. Andreas Neumannn

Berlin, Februar 2013

Abstract

This work deals with the micro-simulation of South African minibus taxis. Those are the most important mode in the public transport of this country. Especially for low-income citizens, living in so called town ships, this mode is irreplaceable for their daily commute, as they are not able to buy or even maintain a car. Irrespective of the great importance the knowledge about this mode is very little. Routes, as well as fares, are not officially announced. For that, this work aims on creating a close-to-reality minibus network. The used software (MATSim), as well as an integrated evolutionary algorithm, are described in detail. The algorithm identifies highly profitable demand relations and generates a supply fitting the demand. The software is extended to fit the south african context. The extensions are tested in detail. Thereafter the model is applied to a South African scenario, namely the Nelson Mandela Bay Municipality (NMBM). The result of this work is a stable minibus-network that identifies the main corridors, as well as areas of low demand. The resulting network may be used for further investigations.

Eidesstattliche Erklärung

Die selbständige und eigenhändige Anfertigung versichere ich an Eides statt.

Berlin, den 25. Februar 2013

Daniel Röder

Contents

List of Figures

List of Tables

Abreviations

BRT . . . . Bus Rapid Transit

ETH . . . . Eidgenössiche Technische Hochschule Zürich/Switzerland

FIFO . . . . First-in-First-out

MATSim . . . . Multi-Agent Transport Simulation Toolkit

mobsim . . . . Mobility Simulation

NMBM . . . . Nelson Mandela Bay Municipal

NTPS . . . . National Transport Planning Study

pkm . . . . passenger-kilometer

pt . . . . public transport

RSA . . . . Republic of South Africa

SA . . . . South Africa

SABTA . . . . Black South African Taxi Association

TUB . . . . Technische Universität Berlin/Germany

USA . . . . United States of America

VMT . . . . vehicle miles travelled

1. Introduction

The minibus taxi in South Africa is the most important mode in the public transport and the second most important – after the private car – at all, regarding to commuter trips to work [1]. That is, 40% of all commuter trips to work are performed by car followed by the minibus with 30%. Based on the public transport only, 60% of all trips are performed by minibus. Based on all trips in South Africa irrespective of the purpose of the trip approximately 30% of all trips are performed by minibus in "typical metropolitan areas", compared to 25% performed by car.

The reasons for the popularity of the minibus date back to the days of Apartheid, as shortly described in the following paragraphs, according to Joubert [2], Joubert and Woolf [3] and Fourie [4]. Informal and semi-formal settlements were found at the outskirts of almost all big cities, the so called townships. Typically, these neighbourhoods were characterized by low household incomes. Although, the structure changed a little bit since the end of the Apartheid, the financial background in these areas is still weak [3]. Thus, the residents are not able to maintain or even buy a car. Therefore they are forced to use public transport for their daily commute. Even here they spend often more than 20% of their income for fares [1]. Admittedly the quality of the official public transport systems is unsatisfactory. During the days of Apartheid it was even worse. For that the residents of the townships started giving lifts to waiting people for money. A bolt hole in law allowed the transportation of up to 8 passengers without being affected by strong regulations of the public transport [3]. From this interim solution a whole (unofficial) industry emerged, the so called "second economy", where drivers are organized in associations. However, the internal as well as the external structures of this economy are farely unknown. Fares, schedules and routes are not officialy announced, instead informations are provided from user to user word by mouth [2, 5]. Nonetheless routes are typically fixed, but vehicles do not depart unless they are not fully occupied. Thus, using the minibus is a difficult venture, especially to foreigners. Of course, the high crime rate in the country does not make it easier [5].

At the latest with the announcement of the Soccer World Cup 2010 the public transport moved into public focus. Bus Rapid Transit (BRT) Systems and highspeed commuter rails – like the Gautrain – were build. In contrast to the minibus the new systems are strictly organized and mostly protected by security services. Although the formalization of the minibus industry was driven further as well [2, 6]. However, the new systems often do not connect the townships mentioned above. Furthermore the collected fares are higher in consquence to the higher quality and security level. Thus, the new systems either do not fit the requirements of the typical minibus user or work in concurrence to the minibus. Both could not be intended as it will lead to a loss on both sides or rather end up in violence ("taxi wars", see e.g. Dugard [7]) .

Although modern mass transport systems are necessary to offer a safe and environmentally friendly, high-quality service, especially on high frequented relations, a complete move away from the minibus seems not to be possible or rather intended. Firstly, the mode is deeply embedded in the South African culture. Secondly, in periods of low demand or in areas of low demand mass transportion is not able to operate profitable [5]. However, to combine mass transportation and minibusses (paratransit) a deeper understanding of the mode is neccessary. Hence, the served routes and the fare structures on operators side, the reasons for choosing a mode on users side and the impacts to the single modes on the systems side. For a detailed analysis of the mutual impacts a detailed traffic model is necessary, as there is currently none. Models like they are presented by Quadrifoglio et al. [8] or Fu [9] aim on the simulation of so called demand responsive transit systems (DRT) that are highly engineered and use dynamic route assignment. Often they are coordinated by an operations central. Thus, they are not suitable in the South African context.

This work presents a first attempt on the simulation of minibusses as they are known in South Africa. Because the routes of minibus taxis are farely unknown an evolutionary algorithm [10, 11] is used to create the minibus system. The work is structured as follows. Firstly, a brief description of paratransit in general and the paratransit (minibus) in South Africa in special is given. The next section will give a detailed description of the used simulation framework, namely MATSim [12], its public transit module [13] and the paratransit module [10, 11]. Subsequently the model extensions done for this work are described. The extensions are tested with a minimal example. The results are subjected to an detailed analysis. Thereafter the used real-world scenario (Joubert et al, unpublished) and the configuration for the simulation is presented. Again, the output will be subjected to a detailed analysis and compared to real world data. The work ends with a conclusion and possible future directions for the model.

2. Paratransit

Paratransit was defined first by Orski [14], Roos and Alschuler [15] and Cervero [16] as a new system within the public transport of the developed world, which serves a new type of demand resulting from the forthcoming suburbanisation. Paratransit as main part of the public transport of the developing world is for example described by Cervero [17] and Cervero and Golub [18] in general and for south africa in special by Fourie [4], Joubert [2] and Joubert and Woolf [3].

2.1. Definition

One of the first works about paratransit was done by Orski [14], but from a view of the developed world. He mentioned transit is commonly describing “public mass transportation”, i.e. systems on fixed routes with fixed schedules. As he described, the traditional systems are not able to serve the new upcomig demand, generated by the novel settlement structure – the so called suburbs – in the United States of America (USA). These suburbs never generated a demand that was strong enough to allow a public mass transportation running profitable, because many people used their own car. The upcoming transport systems – serving the new type of demand – were generally called *paratransit*, as a global description for systems like *jitney, dial-a-ride, van-pool* and so on.

These paratransit systems are described by Roos and Alschuler [15] as well. They classified the systems within two dimensions – time and space – which can be both, fixed and variable. Orski [14] describes paratransit not as flexible in both dimensions, but in at least one. Furthermore Roos and Alschuler [15] give a more detailed description of used service strategies of paratransit-services in terms of space and time as well. They identified four typical types of spatial supply, as shown in Tab. 2.1 and five typical types of temporal supply as shown in Tab. 2.2. According to Roos and Alschuler [15] both dimensions may be combined as shown in Fig. 2.3. Although this definition aims for the paratransit-systems of the developed world one will find systems working like that in the developing world as well.

Table 2.1: Types of spatial paratransit supply [15].

Jitney/ Route deviation services	basic fixed routes, with only small deviations on request, boarding and alighting is possible anywhere along the route
Point deviation services	serve only certain stops at fixed times, leaves the route on customers request
Limited area-wide services	starts anywhere within a specified area (e.g. a suburb), serves only limited destinations (e.g. factories or train stations), normally return trips are offered
Full area-wide services	fully flexible choice of origin and destination

Table 2.2: Types of temporal paratransit supply [15].

Fixed Schedules	allows boarding at fixed times
Vehicle Hail	boarding is only possible when vehicles are stopped by hail
Immediate Request	service requests for single trips that served as soon as possible.
Advanced Request	requests for scheduled, but single trips.
Advanced Standing Request	requests in advance for repetitive trips

Potential benefits, resulting from the above shown paratransit-systems, are described by Cervero [16]. Firstly, he mentioned the increasing of travel choices by a higher diversity of means of transport and a higher quality in general as well. This may result in higher acceptance of public transport overall and a possible abnegation of private cars. Secondly, benefits to the mobility of people – especially for poor neighborhoods – is assumend, thus without increasing the transport perfomance or even with a decrease of the vehicle miles travelled (VMT). Thirdly, resulting from the stagnating individual traffic he expects benefits to the environment in terms of lower pollutant emissions. Fourthly, because of probably higher competition in the public transport, Cervero expects a higher effectiveness, as result of higher flexibilty of small paratransit-companies. Although, he mentioned a complete deregulation is not possible or even desirable, because certain standards of safety and security must be respected.

Table 2.3: Combination of temporal and spatial paratransit supply types according to Roos and Alschuler [15].

	Fixed Schedule	Vehicle Hail	Immediate Request	Advanced Request	advanced Standing Request
Jitney/ Route deviation service	X	X	X		
Point deviation services	X		X		
Limited area-wide services		X	X	X	X
Full area-wide services	X	X	X	X	X

By virtue of the literature reviewed earlier it might look like paratransit has its origination within the developed world of the northern hemisphere. As described by Cervero [17] and Cervero and Golub [18], this is not the case, as a lot of public transport systems in the developing world work paratransit-like. Due to the informal structure of these systems no earlier research was done about the functionallity. However, research about the industrial background is done by Khosa [19]. Furthermore Schalekamp and Behrens [20] researched influence to society itself. The main difference between paratransit in developed and developing countries is the degree of organization. In developed countries paratransit is managed by central control centers, as described by Roos and Alschuler [15]. Against that, in developing countries there is often no central or even computerized organisation. Typically it works completely informal [2, 3, 17, 18].

Looking at Tab. 2.3 and 2.4 it becomes clear that a comprehensive description for all countries is impossible. The variety of used vehicles, offered services and covered areas differs from system to system and country to country. Because the focus of this work is on South Africa the following section will depict the paratransit mode as it occurs there. For further descriptions concerning other countries the reader is referred to Cervero [17] and Cervero and Golub [18].

Table 2.4: Paratransit Vehicle Classification according to Cervero [17, p. 15]

CLASS	Service Features		Passenger Capacity	Service Niche	Service Coverage
	Routes	**Schedules**			
I : Conventional Bus	Fixed	Fixed	25 - 60	Line-Haul	Region/Subregion
II : Minibus/Jitney	Fixed	Semi-Fixed	12 - 24	Mixed	Subregion
III : Microbus/Pick-Up	Fixed	Semi-Fixed	4 - 11	Distribution	Subregion
IV : 3-Wheeler/Motorcycle	Variable	Variable	1 - 4	Feeder	Neighborhood
V : Pedicab/Horse-cart	Variable	Variable	1 - 6	Feeder	Neighborhood

2.2. Paratransit in South Africa

Paratransit in South Africa – generally called minibus or taxi (industry) – is one of the most important modes of transport in this country, with almost the same significance like private motorized traffic. For “typical metropolitan areas” the “South African National Household Travel Survey” [1, p. 113 ff.] specifies that approximately 40% of all commuter trips are performed by car, but almost 30% are performed with minibusses. Referring commuter trips within the public transport almost 60% of all of this trips are performed by minibusses. That is, official public transport services commuter rails or busses do not have the importance as it is known from industrial countries of the northern hemisphere.

To answer the question why is that, a view back to the days of Apartheid gives the answer. Fourie [4] describes that before 1977 only Sedan vehicles played an important role, while serving the demand within black townships. These townships are big informal and semi-formal settlements at the outskirts of almost all bigger cities in South Africa. They are mainly inhabitat by black and coloured citizens. The residents in these areas were and still are part of the lower income groups. As described by Joubert and Woolf [3], these groups were forced to use public transport. Namely, this were PUTCO (Public Utility Transport Corporation) buses and all kinds of municipality buses. Using trains was possible but very unefficient as there was no appropriate schedule and the offered capacity was too low. This serves to explain why the acceptance of mass public transport is not that good. Furthermore Fourie [4] found two more reasons for the acceptance of the minibus in the ”non-white“ community. Firstly, users assumed they support a community-based service and secondly, it was a form of rebellion against the Apartheid regime. For further description of the history of the minibus industry the reader is referred to Joubert [2], Joubert and Woolf [3] and Fourie [4], as the previous paragraph is mainly an extract of their work.

At the latest since the day it was known that the Soccer World Cup will take place in South Africa, the government started to formalize the minibus industry. Officially a driver needs a permission, which defines the start and end point of his allowed route. Normally these points are located at so called ranks [2]. Certainly a high number of so called “pirate drivers” exists. E.g., for the area of Nelson Mandela Bay (NMBM). E.g., 2,347 minibuses are registered, but only 1,304 permits are issued [6]. For Johannesburg a rate of approximatelly 81% illegal operated minibus taxis is found [5]. Typically single drivers are organised in so called associations. The drivers/operators are assigned to the routes by the associations. Drivers rotate between the routes, because the routes diversify in there effectiveness [5].

From the users perspective the system is very intransparent. There is no schedule available, i.e. one need to know about the routes. From the licenses only start and end point may be generated reliable. Vehicles will not depart unless the last seat is occupied [5]. Passengers desiring to travel by minibus will find two ways into the system. Firstly, there a the so called ranks (hubs/terminals). Here a so called marshall will lead the user to the correct minibus. Secondly, vehicles may be stopped at the street by hail. Although, to do this one need to know the correct hand sign. The correct hand sign will show the driver the desired destination, i.e. only minibusses with the correct destination will stop [2].

Another problem is the fare structure, because the fares are not announced publicly. They are set by the associations. Typically passengers pay per boarding regardless of the travelled distance. The Transport Plan of Johannesburg [5, p. 99] specifies fares that may be calculated with a base fare of R2.35 and additionaly R0.064 per kilometer. When travelling a route with total length of 10 km, a fare of R2.99 should be charged. From the authors personal experience fares are much higher, at least in Tshwane (Pretoria) and Durban. Travelling the route that connects Mamelodi (Township east of Tshwane) and the city center of Tshwane a fare of R12 was charged in September

2012. The length of this route is calculated with approximately 26 km.

According to the classification of Roos and Alschuler [15] the typical South African minibus service is a “Route deviation service”, whose vehicles are stopped by hail. The used vehicles fit “Class II” according to Cervero [17].

3. Simulation Framework

The simulation framework for this work is called MATSim [12]. It is mainly developed at the “Technische Universität Berlin/Germany” and the “Eidgenössische Technische Hochschule Zürich/Switzerland”. The following sections will give a short description of the basic simulation and the principles behind it. For a more detailed description the reader is referred to Balmer et al. [21] and Raney and Nagel [22]. After that the public transport extension [13, 23] and the paratransit extension [10, 11] – which this work is based on – are described.

3.1. MATSim

MATSim is a microscopic traffic simulation. Every single part of a traffic system may be simulated separately and is represented by a so called agent. For example, every (synthetic) traveller is represented by an agent. These agents typically have at least one full daily plan, each containing desired activities and desired transport modes. All agents compete among each other for system-resources like the *FlowCapacity* of a street or the offered space in public transport vehicles. MATSim allows to simulate different traffic modes at the same time, but currently only individual motorized traffic [22] and public transport [13] are simulated physically. Other modes are teleported, thus the travel time depends only on beeline distance and average speed per mode[1].

The basic principle of a typical MATSim-simulation is shown in Fig. 3.1. First, the initial data is preprocessed, i.e. in most cases this is the network and the initial demand (daily plans). For a detailed description of how the population generation could be done the reader is referred to Balmer [25]. The iterative loop, thus the three main steps of a typical MATSim-simulation – the *physical simulation*, the *scoring* and the *replanning* – are explained in the following sections.

3.1.1. Physical simulation (execution)

MATSim is a microscopic, agent-based traffic simulation. The physical principle is based on the model of Gawron [27]. A directed graph is used to describe the street network. Intersections are represented by *nodes*, *links*/streets by edges. For every link the knowledge about *FlowCapacity* C, length l_{link} and freespeed v_0 is compulsorily.

[1] The physical simulation of mixed modes is currently work in progress [24].

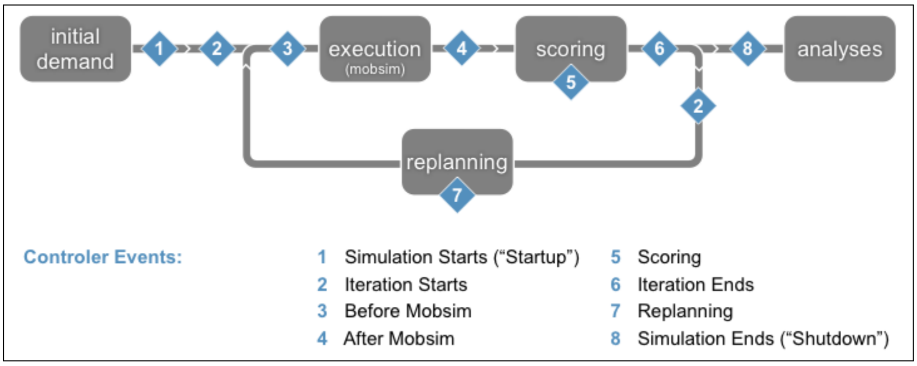


Figure 3.1: MATSim controler structure [26].
The simulation starts (1) with the first iteration (2), the **physical simulation** is executed (3 & 4) and the executed plans are **scored** (5). After the iteration ends (6), another iteration starts (2) and agents are allowed to **replan** (7) or the simulation ends (8) after a certain number of iterations.

Resulting from that the *FreeLinkTravelTime* t_0 and the *StorageCapacity* C_{max} is calculated as follows:

$$C_{max} = \frac{n_{lanes} \cdot l_{link}}{l_{veh}} \tag{3.1}$$

$$t_0 = \frac{l_{link}}{v_0} \tag{3.2}$$

Every *link* implements a First-in-First-out (FIFO)-queue with a maximum capacity of C_{max}. Thus, for every simulation step (normally 1s) the following is done. Firstly, all links are checked for vehicles trying to leave the link. Secondly, they will be moved to the next link taking the following rules into account:

- The vehicle was long enough on the link. Thus, it remains at least for the *FreeLinkTravelTime* t_0 on the link.
- The *FlowCapacity* C has not been exceeded in this timestep.
- The desired link offers free *StorageCapacity* C_{max}.

An agent moved to the the next link is provided with a time-stamp and dropped to the links queue. Thus, there are no further calculations necessary for this agent until it is on top of the queue, as only the first agent will be moved to the next link – following the rules described above – if possible. Note, normally overtaking on a link is not possible. Certainly agents will be moved to the next link, until the *FlowCapacity* has not been exceeded.

3.1.2. Scoring

The second part of MATSim's iterative loop is the scoring, based on the definition of Charypar and Nagel [28]. After every iteration the executed plans are scored. As all agents are in competition for the limited network-resources, the executed may deviate from the intended plan, e.g. because the congestion on a route is very high the realized traveltime might be higher than the expected and an agent arrives late. The score U is the sum (eq. 3.3) of the utility for performing activities $U_{act,i}$ (eq. 3.4) and the (dis-)utility for travelling $U_{trav,i}$ (eq. 3.5). Typically the utility of performing activities is positive. An exception is the "pt interaction", thus the time an agent is waiting for a pt-vehicle (compare Sec. 3.2), which is (normally) scored negative. For a more detailed description the reader is referred to the work of Charypar and Nagel [28].

$$U = \sum_{i=0} U_{act,i} + \sum_{i=1} U_{trav,i} \tag{3.3}$$

$$U_{act,i} = U_{dur,i} + U_{wait,i} + U_{late.ar,i} + U_{early.dp,i} + U_{short.dur,i} \tag{3.4}$$

$$U_{trav,i} = \beta_{mode} \cdot t_{trav,i} \tag{3.5}$$

3.1.3. Replanning

The third part of MATSim's iterative loop is the replanning. Thus, normally a certain number of agents is allowed to modify their plans. The others will randomly choose a plan from their choice-set. For the replanning an existing plan is a) copied and b) modified. This might be done in different degrees of freedom. E.g., agents might modify their planned route (path) through the network, trying to find the route with lowest generalized costs or the so called *shortest path*. Often this search is based on the travel time only. A common way to identify this shortest path is described by Dijkstra [29]. The *shortest path* might differ over iterations as in the first iteration only t_0 is known. During the iterations the system will move towards a Nash equilibrium, i.e. no agent gains any profit unilaterally when changing its strategy. Normally not all agents are allowed to replan in every iteration, as this may lead to a flip-flop-effect. Other possibilities for replanning are the shifting of departure-times or a substitution of the planned transport mode.

3.2. Transit-module

MATSim's physical public transport simulation was introduced and described by Rieser [13][2]. The module implements a schedule-based simulation of public transport. In the following the *TransitSchedule*, the adaption of the physical simulation and the routing-process (used for the replanning) are explained.

3.2.1. The TransitSchedule

MATSim's *TransitSchedule* consists of the parts described below. The xml-format is illustrated in Fig. 3.2. The example shows the implementation of 3 stations (S1, S2, S3) located at the corresponding links (L1, L2, L3). These stations are served by one busline (bus) that operates one route (bus1). The route departs 3 times in a 10 minute interval, starting at 6:00 at S1. It will depart at the second station 5 minutes and arrive at the third station 10 minutes after the departure.

TransitStopFacilities A *TransitStopFacility* is the implementation of a physical transit stop. It is defined by its *id* and its coordinates. But, idenpendently from the coordinates it is assigned to a *link*. Thus, for vehicles the *TransitStopFacility* is accesible from this link only. Traveller trying to find a stop while search stops based on the coordinates. Additionaly the option *isBlocking* defines, if a *TransitVehicle* waiting at this stop blocks the queue of the whole *link*.

TransitLines A *TransitLine* is a container for *TransitRoutes* without any further informations.

TransitRoutes A *TransitRoute* contains the information about the *RouteProfile*, the *Route* and the *Departures*.

RouteProfile The *RouteProfile* defines the sequence of served stops on this *TransitRoute*. These are the so called *TransitRouteStops*.

Route The *Route* is the physical route through the network, thus it is a sequence of *link*-ids, equal to a car-route.

[2] See also Rieser and Nagel [23]

TransitRouteStops *TransitRouteStops* are used to define the *RouteProfile*. Each *TransitRouteStop* is assigned to a *TransitStopFacility*. For all but the last *TransitRouteStop* a defined *DepartureOffset* is compulsory. This is the time a *TransitVehicle* is expected at this stop after the departure at the first stop of the route. Additionaly it is possible to force vehicles for planned departure times at this stop (awaitDeparture).

Departures For each *TransitRoute* a certain number of departures may be defined. For that the departure time and the used *TransitVehicle* are assigned. Note, this is only the plan, but may differ in the (simulated) reality.

3.2.2. Physical Transit Simulation

The transit-module is integrated into the existing MATSim-Framework. Thus, *TransitVehicles* are handled the same way as ordinary cars. They interact with each other, compete for the same resources and stuck in the same traffic jam. This may result in delays compared to the planned *TransitSchedule*. Vehicles arriving earlier than expected will wait for the planned departure (awaitDeparture == true) or otherwise depart immediately after arrival. In any case the vehicle will allow all waiting persons to enter until no more space is left, even if this results in a deviation from schedule.

The handling of users of the transit-system may be explained best with a simple example, using the *TransitSchedule* shown in Fig. 3.2. Person A – trying to execute the plan shown in Fig. 3.3 – plans to leave home at 06:00:01 and will walk to the next *TransitStopFacility* S1, located at *link* L1. There a so called “pt interaction” starts. As the mode “transit_walk” is teleported and the distance between home and S1 is 0, the person arrives immediately after departing at 06:00:01, i.e one second after the first departure of bus1. As explained above, vehicles will not wait longer than planned. Now, the agent will enter the next departing bus – that offers free space – heading forward to station S3. There the person will alight , start a “pt interaction” and will proceed with a “transit_walk” to the planned destination (L3). ”pt interaction“ and “transit_walk” are necessary to implement a pt-trip in MATSim, because an alternating sequence of activities and legs is compulsory. Thus, every pt-trip starts and ends with a “transit_walk” and every real pt-leg is enclosed by “pt interactions”.

```xml
<transitSchedule>
  <transitStops>
    <stopFacility id="S1" x="0.0" y="0.0" linkRefId="L1" isBlocking="false"/>
    <stopFacility id="S2" x="100.0" y="0.0" linkRefId="L2" isBlocking="false"/>
    <stopFacility id="S2" x="200.0" y="0.0" linkRefId="L3" isBlocking="false"/>
  </transitStops>
  <transitLine id="bus">
    <transitRoute id="bus1">
      <transportMode>bus</transportMode>
      <routeProfile>
        <stop refId="S1" departureOffset="00:00:00" awaitDeparture="true"/>
        <stop refId="S2" departureOffset="00:05:00" awaitDeparture="true"/>
        <stop refId="S3" arrivalOffset="00:10:00" awaitDeparture="true"/>
      </routeProfile>
      <route>
        <link refId="L1"/>
        <link refId="L2"/>
        <link refId="L3"/>
      </route>
      <departures>
        <departure id="0" departureTime="06:00:00" vehicleRefId="bus0"/>
        <departure id="1" departureTime="06:10:00" vehicleRefId="bus1"/>
        <departure id="2" departureTime="06:20:00" vehicleRefId="bus2"/>
      </departures>
    </transitRoute>
  </transitLine>
</transitSchedule>
```

Figure 3.2: The xml-format of a MATSim-TransitSchedule.
The schedule contains one bus-line with one bus-route. This route serves 3 stops in a 10 minute interval.

```
<person id="A" >
  <plan score="100.0" selected="yes">
    <act type="home" link="L1" x="0.0" y="0.0" end_time="06:00:01" />
    <leg mode="transit_walk" trav_time="00:00:00">
      <route type="generic"></route>
    </leg>
    <act type="pt interaction" link="L1" x="0.0" y="0.0" max_dur="00:00:00" />
    <leg mode="bus" trav_time="00:10:00">
      <route type="experimentalPt1">PT1===S1===bus===bus1===S3</route>
    </leg>
    <act type="pt interaction" link=L3 x="200.0" y="0.0" max_dur="00:00:00" />
    <leg mode="transit_walk" trav_time="00:00:00">
      <route type="generic"></route>
    </leg>
    <act type="work" link="L3" x="200.0" y="0.0" end_time="16:00:00" />
  </plan>
</person>
```

Figure 3.3: The xml-format of a Simple MATSim pt-plan.
Person A is planning to travel from home to work. For that the person will walk to the closest bus station, access the next departing bus and walk from the sation to the work location.

3.2.3. Pt-routing

The pt-routing used in MATSim [13, 23] is based on the algorithm of Dijkstra [29]. For that the so called *TransitRouterNetwork* is generated, based on the given *TransitSchedule*. For each *TransitRouteStop* a *TransitRouterNetworkNode* is created. According to the *RouteProfile* the nodes are connected with *TransitRouterNetworkLinks*. The resulting graph is unusable for algorithm of Dijkstra [29] as the various *TransitRoutes* are not connected among each other. That should be explained with the following example.
Fig. 3.4a shows a simple public transit network with 3 *TransitLines* and 6 *TransitStopFacilities*. Each line consist of 2 *TransitRoutes* (one per direction). Line 1 connects the stops C and E, Line 2 connects A, B and C and Line 3 connects A, F and D. In Fig. 3.4b now the *TransitRouterNetwork* is shown, consisting of one node per *TransitRouteStop* and links for the corresponding connections. Looking closer at the resulting network, e.g. one will find no connection between D and C or rather E, because there is no

TransitRoute connecting them directly. For that it is necessary to create so called *TransitLinks*, which are created between all *TransitRouteStops* within a certain distance, the so called *maxBeelineWalkConnectionDistance*. Of course transfer links are created at *TransitRouteStops* pointing to the *TransitStopFacilities* A, C, D and E because they are all at the same location. But, between B and F transfer links are created as well, because they are close enough to each other. Now the shortest path between all stops may be found using the algorithm of Dijkstra [29][3].

After the transit router is set up an agents plan can be processed. Before the first routing typically it consists of an activity chain like

Act(home) – Leg(pt) – Act(work).

That is, no detailed information is available about the used route. Processing this plan in the router, the coordinates of both activities are used to find a path through the network. Assuming the home-activity is close to E and the work activity is close to B and the traveltime for walk is longer than the traveltime by pt, the agent is forced to change the used *TransitRoute*, because the only opportunity is travelling from E to C and from C to B. The resulting travel pattern looks like this:

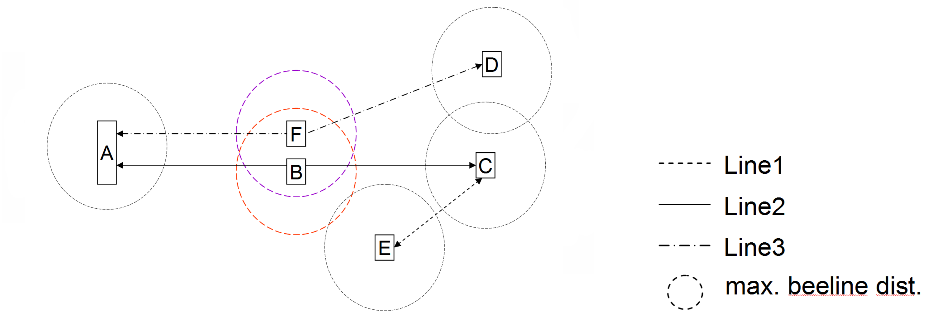
Act(home) – Leg(transit_walk) – Act(pt interaction)
– Leg(pt, Line1) – Act(pt interaction) – Leg(pt, Line2) –
Act(pt interaction) – Leg(transit_walk) – Act(work).

Before every subsequent routing the travel pattern is reduced to the first pattern again, thus the routing starts without no further informations than the used mode. The public transit router will always return the fastest (best) option.

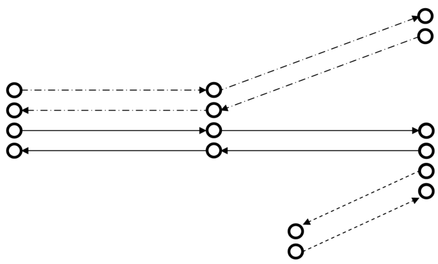
3.3. Paratransit-module

The paratransit-module as described by Neumann and Nagel [10, 11] is an evolutionary algorithm, based on MATSim's public transport simulation, inspired by the idea of paratransit. It aims on identifying profitable *TransitRoutes* through a network, based on a given demand. For that so called *Cooperatives* offer a transport-service. The service changes regularly after an iteration with the attempt to fit the offered supply to the given demand. Profitable opportunities will prevail, unprofitable "die out".

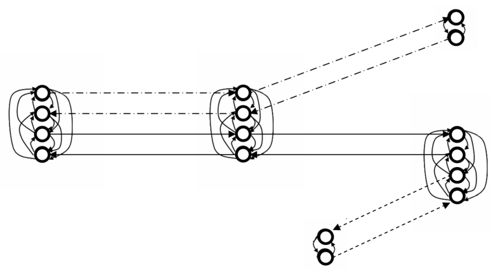
[3] Increasing the *maxBeelineWalkConnectionDistance* will increase the number of transfer links extremly, thus the number of possible pathes and the computing time. Especially in the african context longer walking connections seem to be very usual. Thus, a well balanced level between computational speed and a setup close to reality need to be identified.



(a) Lines operated



(b) Resulting TransitRouterNetwork



(c) TransitRouterNetwork with transfer links

Figure 3.4: Creation of the TransitRouterNetwork.
The first picture shows the public transport system, for that a TransitRouterNetwork will be created. The second picture shows the resulting TransitRouterNetwork. The third picture shows the resulting TransitRouterNetwork with transfer links.

3.3.1. Initialization

Initially the simulation starts with a certain number of *Cooperatives* (operators), each running one *TransitLine* with one *TransitRoute* and a configurable number of vehicles. The number of *TransitRoutes* per *Cooperative* will increase with the iterative process (see Sec. 3.3.4).

For each new *TransitRoute* two *TransitStopFacilities* are drawn from the given set of stops. This is done by a weighted random drawing, based on the activities close to this stop, i.e. a stop with more activities in its surrounding has a higher probability to be drawn. The drawn stops are the so called *stopsToBeServed*.

Now, the shortest path between the *stopsToBeServed* is calculated, resulting in a circular route from the first stop to the second and backwards. This is the physical route that the vehicles will use. All *TransitStopFacilities* on this route will be served by the *TransitRoute*. The service-time is randomly set between 0 and 24h, but with a configurable minimal length. Departures are calculated for the complete time of operation, i.e. vehicles will start again immediately after they reached their destination.

3.3.2. Physical Paratransit Simulation

The *Cooperatives* and the *TransitRoutes* mentioned before are integrated into MATSim's default *TransitSchedule*. Departures are calculated in such a way that a vehicle will never arrive earlier than expected, i.e. the assigned *DepartureOffset* is calculated for the empty network.

Vehicles arriving at a *TransitStopFacility* will allow all waiting persons to enter until no more space is left. The vehicle departs immediately after the last person entered. Thus, the behaviour is equal to ordinary public transport, but the *awaitDeparture*-option is set to false. One should notice, all planned departures will be performed, unless the vehicle is on time or not, i.e. the service ends with the last planned departure that is based on the time of operation.
A very important difference should be noticed for the users behaviour. With the default implementation users will only enter the planned *TransitRoute*. Running the simulation with this module results in a different users behaviour, i.e. users enter every vehicle simulated in the pt considering the following rules:

- The planned route does not exist anymore (may occure because of unprofitable cooperatives).

- The vehicle coming along promise an equal or better traveltime than the planned *TransitRoute*.

3.3.3. Scoring

For each *Cooperative* – more specifically for each of their *TransitRoutes* – a score is calculated after each iteration. This score is the difference of revenue and expenses generated by the routes.

The revenue results from collecting fares. Either a fixed sum is accounted for each person entering a vehicle or a distance-based fare is charged. A combination of both is possible. The expenses are generated by fixed sums and distance-based costs as well. The fixed sums may be used to depict general costs like for licenses or the drivers salary. The distance-based implement expenses like costs for fuel-consumption or mechanical wear.

The described scoring is based on monetary values. Thus, it might be seen as the *operators* cash flow. Because of that the score of each iteration is added to the operators budget.

3.3.4. Optimization

As described in the previous section each *Operator* is scored after every iteration and the sum of the scores depict the *operators* budget. Before the next iteration starts *operators* with a negative budget are forced to balance it. These *operators* are forced to sell their vehicles for a lump sum. Always the vehicles of the worst plan are sold. They have to do this until, either the budget is balanced or no more vehicles are left, i.e. the *cooperative* is bankrupt and terminated. Note, for a certain number of iterations after the initialization *cooperatives* may be allowed to continue with a negative budget, as the demand needs some time to find the new supply. Operators with a positive budget are allowed to buy vehicles for a lump sum. They will buy as much as possible. After the *operators* balanced their budget or bought new vehicles and if they own more than one vehicle, they are allowed to modify their supply. For that one of the existing *TransitRoutes* is drawn randomly. Routes with a higher number of vehicles are drawn with a higher probability. The drawn route is copied. The copy is modified in the dimension of time or space as follows.

Increase of operation time Operators may increase their time of operation. That is, the time of the first/last departure is set randomly to a time earlier/later than the current.

Fit operational time to demand The number of transported passengers is evaluated for a certain number of time slices. The length of the time slices is configurable. Now, the standard deviation of the transported passengers is calculated per slice. All slots with a demand below the standard deviation multiplied by a scaling factor are dropped. From the remaining the earliest and the latest slices generate the new time of operation.

Expanding the served route An operator can decide to modify its route. For that an additional stop is drawn from a set of unserved stops. The chosen stop is added behind the nearest stop from the *stopsToBeServed* of the route. Now the shortest path between all *stopsToBeServed* is calculated again. All stops on this path are served by the *TransitRoute*. For this work a few improvements are implement. An exemplary description for one module may be found in Sec. 4.4. A profound description of all implemented modules is given in Appendix F.

Reducing the served route The served route may be reduced by the reduction of the *stopsToBeServed*. For that, all relations of the existing demand are identified and the number of passengers per relation is summed up. Now the standard deviation of the number of passengers for all relations is calculated. All stops are removed that belong to a relation with number of trips below the standard deviation multiplied with a scaling factor. Note, the route may be the same, when the shortest path still includes the deleted stop. Furthermore the number of stops to be served will never decrease below 2.

The vehicle for the new plan is first taken from the best plan, which necessarily includes 2 vehicles. In case another plan generated a negative score in the last iteration one vehicle is moved from this plan to the best plan. All plans with no vehicles left are deleted.

Bankrupt *cooperatives* will be replaced with new ones, i.e. they will be initialized as described above. Furhtermore the *share of cooperatives with profit* may be defined. That is described best with the following example. Assuming the share is 50 % and the inital number of *cooperatives* is 20. Until 10 or less *cooperatives* are scored positively nothing will happen, but when the 11th *cooperative* is scored positively the total number will increase to 22, because of the *share of cooperatives with profit*. The smaller this value is, the higher the mutation rate is, because more *cooperatives* are founded.

4. Extension of the model

This section describes the extensions applied to the model introduced in Sec. 3. The default model allows to Re-Route only for within the whole public transport system, regardless of any detailed information. For three reasons this is unintended. Firstly, the used survey-data (see Sec. 5.1) is very detailed, i.e. for each person a very detailed daily plan is available, including the used travel modes. Secondly, neither the scoring-function, nor the parameters for routing are known or certainly calibrated, but doing this exceeds the scope of this work. Especially because the financial structures of the minibus industry are not public accessible, the fare structure is only superficial analyzed (see. Sec. 2.2) and the value of time (VoT) of the population is assumend to be very discontinuous, because of the high rate of poor people (e.g. [6, p. 18]). Thirdly, as the paratransit-module (Sec. 3.3) is an evolutionary algorithm a fixed demand will lead to a more stable result, as shown in Sec. 4.5.

To adapt the necessary modifications described before, changes in three parts of the default MATSim-implementation and in the paratransit module are necessary. That is, because the traveller routing needs to be manipulated to force rerouting within the desired pt (sub-)mode. The traveller replanning needs to be extended to use the new routing infrastructure. Furthermore, the mobility simulation (mobsim) needs to be extended to apply the desired agents behaviour that implements the new routing and replanning results. For that interfaces provided by MATSim are used as shown in Appendix A. The functionality and the necessary implementation are explained in the following. Further the operators route replanning – used for the optimization-process (Sec. 3.3.4) – is extended for this work to improve the computational speed.

The section closes with a test, showing that the new implementations will work and lead to the expected results.

4.1. Extension of the traveller replanning

Currently MATSim allows to replan within different degrees of freedom. That is for example agents might change their route. In terms of car routing this will be used to apply changing traveltimes over the iterations. In terms of pt routing the (desired) result will be static – i.e. always at the best option – assuming the network is stable. Agents choose always their optimal path through the network. In terms of this work this is not intended, because a) the minibus-network will not be stable over the iterations and b) the best option is not naturally the one persons reported in the survey (see Sec. 5.1), as personal experience may lead to a different behaviour. Problem a) will be solved in the following section as it is more part of the routing process. Problem b) is part of both, replanning and routing.

The default is doing what is displayed in Fig. 4.1. The input plan (Fig. 4.1a) is to be replanned. The information about the leg mode is very detailed. It is not only known which mode (car, pt etc.) is used, but the detailed means of transport within the pt (bus, train, minibus etc) as well. Now, Fig. 4.1b shows the output as one will receive it currently. The leg mode is set to pt. For that, the whole pt is used for the search. Fig. 4.1c shows the desired output, namely the detailed means of transport are still known and the path that leads through the network will contain only legs that use this mode or a "transit_walk".

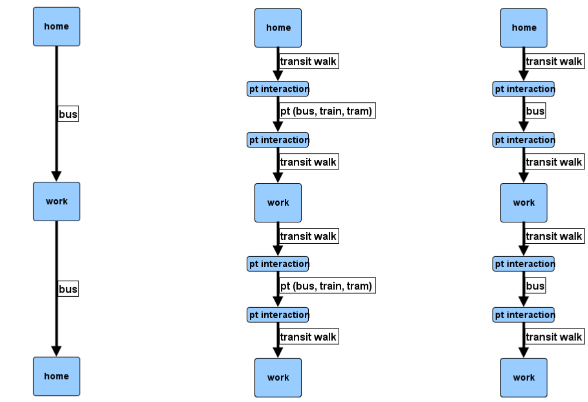
Implementation To obtain this output the implementation of an own replanning strategy is necessary, because existing strategies will overwrite any detailed information about the means of transport within the pt. For that an own MATSim-`PlanStrategy` – the so called (`PlanStrategyReRoutePtFixedSubMode`) – consisting of the following 3 modules is implemented:

PtSubModePtInteractionRemover The class is doing almost the same as the default `PtInteractionRemover`, but with a minor change for computational reasons. It uses detailed means of transport, e.g. bus/train/minibus, instead of "pt" that was hardcoded in JAVA before. Thus, the main scope of this class is not to delete detailed travel information before routing, but reduce the plan to the intial Activity-Leg-Activity-pattern.

ReturnToOldModesStrategy It might occur, that a "transit_walk" is cheaper/shorter than travelling pt. Once this occurs a former pt leg is transformed to a "transit_walk"-leg by the router. That is, the `PtSubModePtInteractionRemover` was not able to find anything but the "transit_walk", i.e. the detailed information is lost. For that this class stores the initial transport mode and resets it in case of loss.

ReRoutePtSubModeStrategy This class is technically necessary to integrate the new routing (described in the following) into the new `PlanStrategy`.

Each of the modules described before is an implementation of the class `AbstractMultiThreadedModule`. This will increase the computational speed (compare Appendix B). The described modules are used in the `PlanStrategyReRoutePtFixedSubMode` in the order as shown above.



(a) Input (b) current output (c) Desired output

Figure 4.1: Default and desired routing output.
In Fig. 4.1a a typical input plan is displayed. The leg mode is detailed (bus). In Fig. 4.1b MATSim's default output is shown. The detailing about the mode of transport is lost, i.e. the mode is now pt but not bus anymore. In Fig. 4.1c each leg is provided with the detailed mode of transport again.

4.2. Extension of the traveller routing

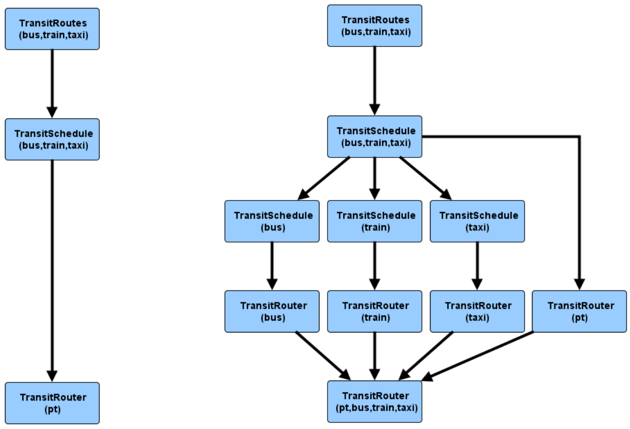
The current MATSim-router (as described in Sec. 3.2.3) allows to find paths through the whole pt-system. That is, assuming the system consists of bus-, train- and minibus-routes an agent is searching for a path through the pt-network regardless of the detailed mode. Because the provided informations about the used means of transport are very detailed, it seems to be useful to route the agents within the specified (sub-)mode. Especially because the knowledge about necessary informations for scoring and routing (costs, travel behaviour, etc.) is farely unknown. That is, finding "correct" (close to reality) routes through the network – searching independently from the detailed means of transport – seems to be very improbable and exceeds the scope of this work. Thus, the routing is done within the specified (sub-)system only. For that, the existing router structure (Fig. 4.2a) is divided according to the known sub-systems (Fig. 4.2b). Hence, one transitrouter per subsystem and one for the complete system is created. Calling the transitrouter will provide a route pattern within the desired subsystem in case it exists. Otherwise (the subsystem does not exist) the router falls back to the default (complete) router. This is done, assuming that persons rather will travel with an undesired mean that found no route.

Implementation

The standard MATSim `Controler` is extended by the `PtSubModeControler`. In the constructor of this class the `PtSubModeTripRouterFactory`, the `PtSubModeRouterFactory` and the `TransitSubModeQSimFactory` are set. The run method is overwritten to add the `PtSubModeRouterFactory` as last `ControlerListener`. That is, the listener should be called after the public transport system for the following iteration is loaded (or founded, in terms of paratransit).

The `PtSubModeTripRouterFactory` is an implementation of the interface `TripRouterFactory`. Here the method `createTripRouter()` is overwritten, where for each mode a different routing module is set. That is, here for every mode that is set with the parameter *transitModes* in the `TransitConfigGroup` an own instance of the `TransitRouterImpl` is set, containing only routes belonging to this mode.

The routers mentioned before are generated by the `PtSubModeRouterFactory`, which is an implementation of the interface `PTransitRouterFactory` and an extension of the abstract classes `IterationStartsListener` and `StartUpListener`. The extension of the listeners is necessary because the certain router need to be updated before every iteration, as the TransitSchedule changes regularly.



(a) old (b) new

Figure 4.2: Pt-Router architecture.

4.3. Extension of the mobility simulation

The extension of the mobility simulation is necessary to realize an agents behaviour according to the new routing and replanning algorithm. Furthermore it is necessary to allow agents to use the physically simulated pt, even when their desired mode is not “pt”, but a submode of the pt (e.g. bus or paratransit). The agents behaviour is changed because the paratransit module allows to enter all vehicles in the pt that offer a better or at least equal in-vehicle-time. The desired behaviour is to enter when the following conditions are true:

- the route is serving the desired stop.
- the offered travel is better or equal than the planned travel time.
- the mode is correct (i.e. do not enter a bus, when it was planned to enter a minibus)

The first two items are equal to the behaviour of the agents created by the default paratransit-module. The third point is very important as otherwise the results of the routing are not necessarily executed. Of course, from the original scope of the paratransit-module this is desired. Considering the scope of this work, this behaviour needs to be changed.

Implementation

MATSim provides the interface `MobsimFactory`. For this work it is implemented by the `TransitSubModeQSimFactory`. This is necessary to overwrite the `create-Mobsim(...)`-method. it is copied from the `QSimFactory`, which is the default MATSim implementation. Overwritting this method is necessary for 2 reasons. The `TransitQSimEngine` and the `AgentFactory` need to be replaced, to allow the desired agents behaviour. For that the `TransitSubModeAgentFactory` is impemented to create the so called `TransitSubModeAgent`. Here the method `get-EnterTransitRoute(...)` is overwritten to implement the desired behaviour described above. Furthermore the `TransitSubModeQSimEngine` with the overwritten method `handleDeparture(...)` is implemented. This is necessary for technical reasons, because departures in the physical simulated pt should be allowed for all submodes of the pt. The default allow agents only to depart when their desired mode is named “pt”. Now, it allows agents to depart when their desired mode is set with the parameter *transitModes* in the `TransitConfigGroup` of the default MATSim configuration file, otherwise they are teleported.

4.4. Extension of the operators route optimization

This section exemplary describes an improvement of the route optimization outlined in Sec. 3.3.4. It seemed to be necessary because the original implementation is drawing stops completely random-based from all allowed stops. For small and artificial networks this seems to be not a serious problem, but for larger networks the probability of finding good/optimal solutions will decrease. As the operators route search is not the main scope of this work, only an exemplary description for two algorithms is given here. Further algorithms and informations are provided in Appendix F.
The modules described here extend the cooperatives route at the beginning or at the end. Due to the fact that all used cooperatives are circle-lines (compare Sec. 3.3) the turning-point is not defined yet. One might say the start- and the end-point are equal but this seems to be problematic because normally a transitline serves a route along a corridor. This corridor is mostly defined by start- and an end-point. Thus, for the extension of the route at the end a turning point has to be defined. Beyond this the computation of the turning-point is necessary for the invented algorithms, because the corridor – the cooperative operates within – is defined by a vector $\overline{r_1}$ (Fig. 4.3) between the first stop which has to be served and the one in the greatest distance. This definition

is equal for both modules and results in the equations shown below.

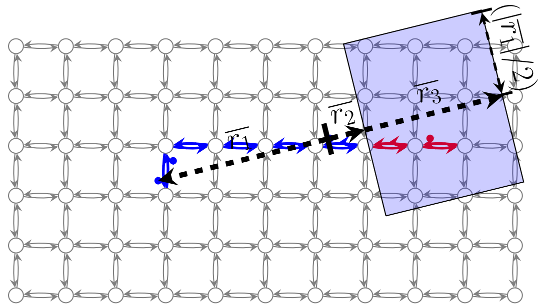
The factor d describes the length $\overline{r_2}$ of the space between first/last stop and the border of the searchspace as well, as the length $\overline{r_3}$ of the searchspace. The length of the corridor $|\overline{r_1}|$ is equal to the sum of $|\overline{r_3}|$ and $|\overline{r_2}|$.

$$|\overline{r_2}| = (1 - d) \cdot |\overline{r_1}| \tag{4.1}$$

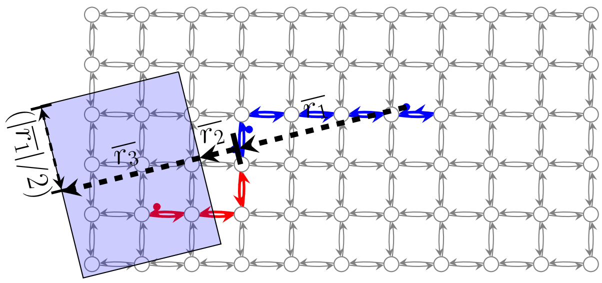
$$|\overline{r_3}| = d \cdot |\overline{r_1}| \tag{4.2}$$

$$|\overline{r_1}| = |\overline{r_2}| + |\overline{r_3}| \tag{4.3}$$

Possible solutions after the execution the modules are shown in Fig. 4.3a and 4.3b. Within the rectangle (blue area) a stop f is drawn by random, until it is unserved yet. The probability for all stops in this area is equal. In case of the *RandomRouteStartExtension* the found stop is inserted before the first stop to be served. In case of the *RandomRouteEndExtension* the stop is inserted after the stop to be served within the largest distance.



(a) Possible route after replanning with RamdomRouteStartExtension



(b) Possible route after replanning with RamdomRouteEndExtension

Figure 4.3: Examplary extension of the operators route optimization. Colored dots indicate the stopsToBeServed. Black arrows indicate links. All blue elements depict the original route. All red elements depict (one) of the possible extensions. Blue areas indicate the searchspaces within the algorithms try find a new stops.

4.5. Testcase

The implemented model extension is tested with a very simple network, consisting of 4 links and 3 nodes as shown in Fig. 4.4. All links have a length of 2,000 m and a flow capacity of 4,000 veh/h. The freespeed is set to 10 m/s.

The demand used for this test consists of 1,200 agents trying to travel by bus and 1,200 trying to travel by paratransit initially. All agents try to travel from their home-activity to work and backwards. Home activities are located at A, work activites are located at C. Their departures for the first leg are equally distributed between 6:00 and 8:00. For the travel to their home location the departures are equally distributed between 14:00 and 16:00.

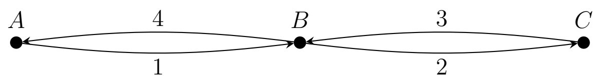


Figure 4.4: Test network.
Arrows indicate links. All links are open for paratransit and busses. A circular bus line (Tab. 4.1) operates between A and C.).

Furthermore a transit-schedule is implemented. A stop-facility is assigned to each link. Each stop is named according to the link corresponding link. Additionaly a busline is implemented. The characteristics are shown in Tab. 4.1. The bus runs with a headway of ten minutes between 6:00 and 9:00 and 14:00 and 17:00. The vehicles offer a capacity of 200 seats each. That is, the offered capacity is able to serve the complete demand, independently of their desired mode. The departure offsets are calculated with the freespeeds of the links. At each stop the vehicles will wait for 10 seconds, unless they are on time. That is, the "await departure" is set true. This results in vehicles waiting for the planned departure or go on immediately when they are late.

Table 4.1: Characteristics of the test busschedule.

stopId	arr. offset [hh:mm:ss]	dep. offset [hh:mm:ss]
4	-	00:00:10
1	00:03:30	00:03:40
2	00:07:00	00:07:10
3	00:10:30	00:10:40
4	00:14:00	-

The paratransit module is intially set up with 4 cooperatives, 5 vehicles per operator and a share of positives cooperatives of 50%. New founded operators are allowed to operate for 2 iterations with a deficit. After that they are eliminated and replaced by a new one. Operators earn 1.7 monetary units per passenger entering, but each kilometer is charged with 0.9 units per kilometer. Furthermore a lump sum of 10 monetary units per iteration and vehicle is deducted from the operators budget. The price structure is calcuted to allow an operater to offer his service in profit with a a seat occupancy of 50%. That is, for the new implementation about 27 vehicles per slot should be expected for the optimal solution. Vehicles may be bought and sold for a sum of 500 monetary units. They offer space for 10 passengers. The creation of new cooperatives is disabled in iteration 1,000. Agents are allowed to search for new routes up to the end of the simulation. The simulation ends with iteration 1,500. For the operators route search the modules RandomRouteStartExtension, RandomRouteEndExtension, RectangleHullRouteExtension, RandomEndTimeAllocator, RandomStartTimeAllocator, ReduceStopsToBeServed and ReduceTimeServed are used. The simulation is executed with the old and the new implementation. In both cases 10 sensitivity runs are conducted. That is, the random seed is changed to analyze the stability of the results.

Results of the old implementation The outcome of the simulations started with the old features is shown in Tab. 4.2. All surviving routes use the shortest possible path, namely this is either the link sequence

2-3-4-1-2 or 4-1-2-3-4.

In all simulations the morning and the evenning peak is identified and served by at least one operator. However, some did not find the complete peak. E.g., in run 3 the last vehicle in the morning departs at 7:56 am. Agents that desire to travel by paratransit are forced to use the bus here.

Though, one should notice the number of passengers transported in the morning peak as well as in the afternoon peak. For both the numbers fluctuate between approximately 2,200 and 2,300 passengers. That is, passengers that planned to use the bus (in their initial plans) switch to the paratransit mode. Only the remaining use the bus as desired.

Although both modes travel with freespeed – the schedule is calculated like this and the flow capacities are high enough – agents are not attracted by this the bus. This happens mainly for 5 reasons:

- Busses will wait for their planned departure, i.e. each stop will add at least 10 seconds waiting time to the minimum travel time.
- Paratransit operators offer a smaller headway, when offering the same transport capacity. Namely, because the vehicle capacities are smaller and thus the frequency is higher (20 paratransit vehicles offer the same capacity as 1 bus in 10 minutes).

- The public transit router will only route agents on busses when the sum of expected utility (for waiting and travelling) is lower as it is expected for travelling paratransit.

- Assuming agents are prepared to travel bus, i.e. the routing request gives back a bus route, they will enter paratransit vehicles when they came along, as the promised travel time is always shorter than the expected travel time by bus.

- Assuming agents are prepared to travel paratransit, i.e. the routing request gives back a paratransit route, they will not enter busses as the promised travel time will be always longer (waiting at stops) as the promised.

Results of the new Implementation The new implementation delivers good results (Tab 4.3). All agents desiring to travel by bus perform their trip without any trouble. Paratransit operators find the shortest path as described before. That is, the used routes consist of either the link sequence

2-3-4-1-2 or 4-1-2-3-4.

Operators find the morning and the evenning peak, mainly complete. As mentioned with the results for the default implementation some operators finish their service a few minutes before the last agent starts. This happens 4 times.

In run 4 only 1,110 agents are transported in the morning peak, although the operator serve the complete peak. But, the operator runs 2 insignificant routes with only 2 or rather 4 vehicles for the complete time. The major route runs from 6:00 to 7:30 only. The is, the systems capacity shrinks after 7:30 seriously. The both remaining routes offer not enough capacity to carry the remaining 90 passengers. Of course, the 90 agents will not travel in the afternoon, as they never reached their work location.

In run 6 the afternoon service end at 15:42. Only 1,043 passengers are transported. The same is for run 8 where only 1,116 passengers are transported.

In run 10 the afternoon peak ends at 15:53 according to the transitschedule, but 1,199 passengers are transported. Although, only 1,130 passengers should have been started until that time. Analyzing the output of the simulation leads to the result, the vehicles are late, because of the passengers entering/boarding. However, vehicles not on time will depart until the last planned departure is performed, as described in Sec. 3.3.

4.6. Conclusion

The tests described above deliver the expected results. The default implementation of the paratransit shows the desired – for the former objective – behaviour. That is, the module will identify gaps in the existing public transport supply. These gaps are filled by the new founded operators. Agents move to the new supply. Although this behaviour is intended by the module, it is not desirable for the simulation of a real exisiting paratransit systems, which operate in co-existence with other public transport systems. Especially in South Africa the decision which mode is used is not only based on costs (time and money), but on personal preferences, cultural requirements and historical experiences. Of course, these factors may be depicted by a generalized cost function. For a lack of missing data this is not possible, respectively it exceeds the scope of this work. For that the new implementation is done.

The new implementation deliver good results. Agents are forced to use their desired mode. That is, a desired bus-trip will be performed by bus or the agent walks, when the expected travel time is shorter. The same is for the paratransit (or any other pt mode). The paratransit operators are able to identify the best pathes as well as the profitable time slots. Certainly, not for all sensitivity runs the complete time slots are found and a few minutes at the beginning or end are missing, i.e. agents may stuck in the system. But, for the majority of the sensitivity runs all agents are transported at the desired time. In the majority of the runs the final operators are founded before iteration 100. Some of their routes are modified in iterations later than 900. Typically the time of operation is modified here, because this is the major task in this test.

Table 4.2: Testcase – Resulting cooperatives with the original implementation.

run	founded [iter]	modified [iter]	start [hh:mm]	end [hh:mm]	veh [#]	pax [#] morning	pax [#] afternoon
1	36	101	06:00	08:13	52	2,348	-
	27	109	13:58	16:10	51	-	2,238
2	0	109	06:00	08:13	48	2,190	-
	0	849	05:59	08:07	4	174	-
	14	111	14:00	16:08	49	-	2,107
	14	892	14:00	16:01	2	-	102
3	480	691	06:00	07:56	55	2,198	-
	480	722	06:00	07:50	3	90	-
	480	751	06:00	07:49	1	80	-
	39	122	14:00	16:18	48	-	2,194
4	0	11	06:00	08:45	39	2,119	-
	0	974	06:00	08:15	5	214	-
	10	107	14:00	16:08	48	-	2,096
	10	691	14:00	16:12	3	-	111
5	48	113	06:00	08:15	49	2,204	-
	48	789	06:00	08:07	3	132	-
	49	112	14:00	16:00	50	-	2,091
	49	574	13:59	15:54	4	-	117
6	172	202	06:00	08:15	50	2,222	-
	172	890	06:00	08:08	2	128	-
	49	145	14:00	15:55	53	-	2,087
	49	875	14:00	15:53	3	-	127
7	157	158	06:00	08:15	51	2,280	-
	157	935	06:00	08:16	1	77	-
	76	157	13:55	15:55	48	-	1,998
	76	461	13:54	15:55	6	-	192
	76	890	13:55	15:55	3	-	137
8	0	100	06:00	08:25	42	2,017	-
	0	107	06:00	08:09	4	167	-
	0	836	05:59	08:09	4	188	-
	151	203	14:00	16:00	52	-	2,117
	984	996	14:00	15:30	3	-	91
9	15	329	06:03	07:56	54	2,105	-
	15	844	06:03	07:50	2	77	-
	113	947	13:59	16:15	46	-	2,109
	113	975	13:59	16:11	2	-	48
	113	977	13:59	16:10	1	-	76
10	13	104	06:00	08:16	49	2,226	-
	13	637	05:59	08:16	3	130	-
	12	119	14:00	16:24	45	-	2,183

Table 4.3: Testcase – Resulting cooperatives with the new implementation.

run	founded [iter]	modified [iter]	start [hh:mm]	end [hh:mm]	veh [#]	pax [#] morning	pax [#] afternoon
1	21	27	06:00	08:11	26	1,121	-
	21	960	06:00	08:15	1	79	-
	17	18	14:00	16:15	26	-	1,200
2	35	37	06:00	08:45	22	1,174	-
	60	102	05:30	06:30	1	26	-
	14	28	14:00	16:03	26	-	1,075
	14	998	14:00	15:59	3	-	125
3	24	25	06:00	09:00	18	1,040	-
	24	993	06:00	08:04	4	160	-
	57	62	14:00	16:15	26	-	1,200
4	51	102	06:00	07:30	25	817	-
	51	532	06:00	08:00	2	107	-
	51	952	06:00	08:15	4	186	-
	42	114	14:00	16:15	25	-	1,168
	935	998	14:00	15:21	3	-	94
5	61	557	06:00	07:56	31	1,200	-
	46	641	14:00	15:54	31	-	1,200
6	0	102	06:00	08:15	27	1,200	-
	3	102	13:55	15:42	28	-	1,014
	3	931	13:55	15:42	1	-	29
7	12	14	06:00	08:45	21	1,120	-
	12	981	06:00	08:15	1	80	-
	6	32	13:55	16:09	25	-	1,100
	6	990	13:55	16:04	2	-	100
8	17	25	06:00	08:15	27	1,200	-
	77	121	13:57	15:45	25	-	955
	77	876	13:57	15:43	3	-	70
	77	923	13:57	15:40	2	-	91
9	0	11	06:00	08:15	25	1,108	-
	925	996	06:00	08:00	2	92	-
	16	47	13:58	16:30	23	-	1,200
10	9	11	06:00	08:15	26	1,200	-
	58	426	14:00	15:53	31	-	1,199

5. Scenario and Setup

This section begins with a brief description of the used scenario, followed by the description of the setup of the simulation. Because of the local context paratransit is referred to as *minibus* or *minibus taxi*.

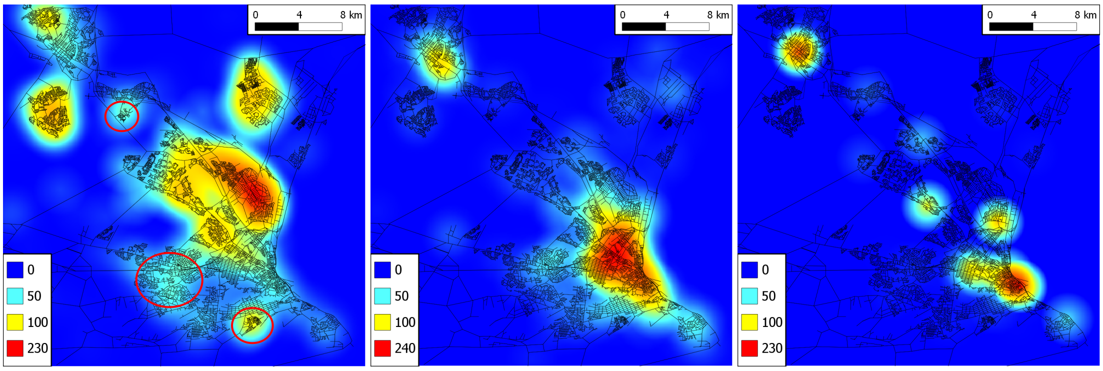
5.1. Scenario

The used scenario is generated and kindly provided by the "Optimisation Group & Centre of Transport Development" from the University of Pretorias "Department of Industrial & Systems Engineering"[4]. For computational reasons a few changes are applied to the provided scenario. These are marked clearly.

The scenario covers the city of Port Elizabeth/RSA, also known as the Nelson Mandela Bay Municipal (NMBM). It was selected for two reasons. Firstly, the number of inhabitants is with approximately 1.1 milion very comprehensible. Secondly and more important, the city and its agglomeration is very "isolated", i.e. no bigger cities are in the surrounding. Thus, possible side-effects because of national traffic should be minimized.

The spatial diversification of landuse between working, living and shopping places is very strict, as one may see in Fig. 5.1. Leisure and educational activities are mostly located in living areas (Appendix C).

[4] The description about the generation of the population and the conversion of the osm-data describe the work and methodology of this group. It is unpublisched yet. The description has been don, as far as it is known to the author. For an in-detail description of the population and network generation the reader is kindly referred to the group of Johan W. Joubert.



(a) Home-activities (b) Work-activities (c) Shopping-activities

Figure 5.1: Spatial diversification of activities.
The figures show heatmaps for the different activity-types performed in the NMBM-area. They should be read as follows. An Activity has got an influence of 1 at its location up to 0 in a distance of r = 2,500 m. In between it is linear interpolated. This has been done for every activity. The shown heatmaps are the results of the accumulation of all activities of one type. Red demarcated areas in Fig. 5.1a depict neighbourhoods of less densitiy of home activities.

5.1.1. Population generation

The synthetic population (Joubert et al, unpublished) is based on the Census of 2001 and generated in two steps. Firstly, an iterative proportional fitting (IPF) – similar to the work of Müller and Axhausen [30] – is used. The source data is fitted on individual level, but on household level as well. Secondly, the 2004 travel survey is used. It contains 24-hour trip diaries of approximately 1% of the inhabitants of this area. For each surveyed househould the size, the employment status, the number of available cars and the overall household income are evaluated. Furthermore each household member filled in a detailed questionaire about performed activities and used means of transport. The generated activity chains are assigned to the individuals characteristics. Now, for each individual of the synthetic population an activity chain is selected so that the characteristics of the synthetic person fit the ones of the real person.

For this work a randomly drawn 1%-sample of the complete synthetic population is used. The characteristics are shown in Tab. 5.1. The resulting relative deviations are within an acceptable range.

Table 5.1: Characteristics of the generated population and the used sample.

		Synthetic population	1 % Sample
Individuals	[#]	1,164,150	11,498
Share of Females	[%]	52.5	52.2
Share of Males	[%]	47.5	47.8
Average Age	[years]	27.7	27.1
Car-trips	[#]	356,208	3,574
Minibus-trips	[#]	920,722	9,261

5.1.2. Network generation

The given network (Joubert et al, unpublished) is generated from OpenStreetMap [31] (OSM) data. The OSM-data is converted to a MATSim-readable format with the parameters shown in Tab. 5.2. The length of the links is calculated to the euclidian distance.

For this work the network is simplified according to the algorithm of Balmer et al. [32]. All nodes from the type “one way pass” and “two way pass” are removed, i.e. links pointing to the same direction, connected by a node of one of this types and with the same characteristics are merged. That is the length is accumulated. The simplyfing is done to reduce the number of links, thus the number of possible paratransit-stops. This again is meaningful for computational reasons, as each link/stop will increase the number of possible pathes through the minibus-network, what increases the computation time exponentially (see Appendix B). However, the resulting network consists of 39,507 links and is shown in Fig. 5.2. Especially the capacities (shown in Fig. 5.2b) are very low except for a certain number of arterial roads.

(a) Speed [km/h]

(b) Capacities [Veh/h]

Figure 5.2: Used NMBM network.

Table 5.2: (Default) parameters to convert OSM-data (column 1 & 2) into MATSim links.

hierarchy-layer	highway-type	freespeed [km/h]	FlowCapacity [Veh/h/lane]	one way
1	motorway	120	2,000	true
1	motorway_link	80	1,500	true
2	trunk	80	2,000	false
2	trunk_link	50	1,500	false
3	primary	80	1,500	false
3	primary_link	60	1,500	false
4	secondary	60	1,000	false
5	tertiary	45	600	false
6	minor	45	600	false
6	unclassified	45	600	false
6	residential	30	600	false
6	living_street	15	300	false

5.2. Simulation setup

The simulation is started with the 1% population sample described before. Thus the given capacities, namely the *StorageCapacity* and the *FlowCapacity*, are reduced as well. The *FlowCapacity* is reduced to 5%. The *StorageCapacity* is reduced to 10%. One may notice that the capacities are not scaled to 1%. The main reason for that is “expert knowledge”, but might be explained by the example given in the excurse at the end of this section.

Initially the minibus model is started with 35 *minibus operators* offering their service with one route and 21 minibuses each. Operators are allowed to buy new minibuses for 1,000 monetary units, but can sell them for 250 monetary units each only. This will lead faster to an equilibrium as good solutions will be slowly increased, but bad opportunities die out very fast. Operating a minibus is accounted with 10 monetary units per iteration and 0.25 per km. Operators collect a fare of 3 monetary units per passenger boarding, regardless of the travelled distance. According to the price structure derived from Transport Plan of Johannesburg, this should be the price for a route with a length of 10 km (see Sec. 2.2). A distance-based fare is explicitly not used as it do not depict the fare-structure in the reality.

Newly found operators are allowed to operate with a negative budget/score during their first 4 iterations without any consequences. The used cost-structure is explicitly not depicting the real structure, as it is farely unknown. For example fares are not published, but known to frequent users by word-of-mouth advertising only. Operational costs are completely unpublished. For that the the used cost-structure was choosen to run the evolutionary algorithm efficiently as possible. Operators are allowed to create new routes until the end of iteration 150. Buying and selling vehicles is allowed until the simulation ends. Minibus stops are located on all links with a speed limit of 80 km/h or below. That is, only the strong red links in Fig. 5.2a – namely the motorway – are excluded. This has been done according to the assumption that it is not possible to stop there. Note, minibuses are allowed to stop on ramps. For the operators route search the modules RandomRouteStartExtension, RandomRouteEndExtension, RectangleHullRouteExtension, RandomEndTimeAllocator, RandomStartTimeAllocator, ReduceStopsToBeServed and ReduceTimeServed are used.

Agents store three plans in their choice-set. Until the end of iteration 200, a randomly chosen set of 40% of the agents is allowed to search for an alternative route within their chosen transport mode after each iteration. The plan definitely remains in the choice set until the agent is selected for rerouting again. Now the plan, which generated the worst score, is dropped when the new plan generates a better score.

The transitrouter is set up with a radius of 2,500 m for the search of possible boarding/alighting stops. From a european point of view this may seem to high, but in the (south) african context [1] its very usual to walk longer distances to access public infrastructure. The radius may be extended for another 1,500 m in case no stops are found. Especially in the beginning (when not that much supply exists) this is useful. The *maxBeelineWalkConnectionDistance* is set to 300 m. That is, agents will try to walk only 300 m to change the vehicle. Of course, this value is very far from the 2,500 m mentioned above. However, it is necessary for computational reasons. The resulting network will be very dense, especially in the city center. That is, the resulting TransitRouterNetwork will increase exponentially (compare Appendix B) with the number of possible transfer connections. For the scoring/routing only one, but very important, changing is done. The disutility of line switch is increased from -0.3 to -0.6 to prevent a high number of agents travelling very short subroutes.

For the simulation of cars the default settings of MATSim are used, as the main focus of this work is on the paratransit mode and the simulation of car traffic is only necessary to obtain more realistic travel times on the links. For this scenario all pt-modes – except the minibus – are teleported, because the accesible real-world schedule is flawed and finally not free to use

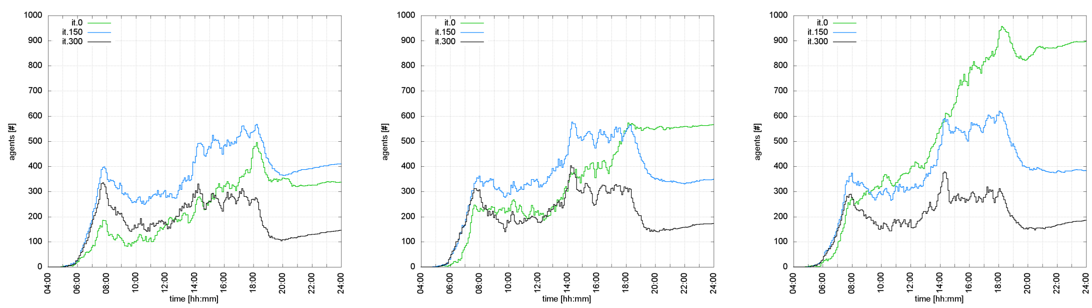
Excurse: Scaling the Queue-Model Assuming a link offers a *StorageCapacity* of 10 vehicles (70 m linklength divided by 7 m, the average length of a car). The reduction to 1% leads to a *StorageCapacity* of 0.1 vehicles, i.e. it will not be possible for any vehicle to enter. It is the same with the *FlowCapacity*. A link with a capacity of 3,600 Veh/h allows that 1 Veh/s passes. That is, scaled to 1% only 0.01 vehicle is allowed to pass. In other words, first after 100 s a vehicle is allowed to leave the link. A similar problem occurs looking at the capacity of minibusses, because the underlying traffic simulation simulates only whole passenger. Typically a South African minibus offers space for 16 people. Reducing the capacity to 1 % would lead to unusable vehicles, as there would be only space for 0.16 persons. The consequence is, a capacity of 1 is the absoulute minimum. But, this will lead to a problem as well. Because all agents planning to use the best opportunity very low vehicle-capacities will lead to something like a flip-flop-effect, as the best opportunity is fully loaded with the first agent entering. For that the capacity is set to 2, aware of the resulting consequences to the produced data (for example the expected frequency will be lower).

6. Analysis and Validation

In the following a detailed analysis of the proposed model, applied to the NMBM scenario described before, is given. The analysis starts with a detailed view on the evolution of the minibus system. Thereafter the output of the simulation is validated against real-world data and examined for plausibility. For validation reasons, a certain number of sensitivity runs is executed. All of the runs with the same configuration parameters, but with different random seeds.

6.1. Evolution of the minibus system

Before the simulation starts there is no minibus system. The supply is randomly drawn. That is, the founded minibus operators will not naturally fit the given demand, although stops with more activities in their surrounding will be drawn with a higher probability. Fig. 6.1 shows the evolution of the minibus system. More specifically the agents en route are displayed for iteration 0, 150 and 300 and that for three different runs.



(a) run 3 (b) run 5 (c) run 8

Figure 6.1: Agents en route in the minibus system.
The figures show the agents en Route during the day for different runs. That is, the same config parameters but different random seeds are used fo the simulation.

In iteration 0 (green line) the supply is drawn by a weighted a random. As might be expected the divergations between the 3 different runs are stronger than in later iterations. Although a morning and afternoon peak are already noticeable in all 3 runs the forms are very different. In run 3 the morning peak is less distinctive compared to run 5 and 8. Anyway, for iteration 0 it can be stated that substantially less agents stuck in the end of run 3, compared to run 5 and 8.

Even though the forms of iteration 0 are very different, the forms of iteration 150 (blue line) are much more similar. All 3 runs show a morning peak between 7:00 and 8:00 with approximately 350 and 400 agents en route. The same may be noticed for the afternoon/evening peaks between 14:00 and 19:00, as the number of agents en route deviates around 550. The number of agents stucking is brought more into line with approximately 350 to 400 agents.

In iteration 300 (black line) – after disabling the innovative minibus modules and agents rerouting – the forms of all 3 runs are very similar. All peaks are clearly distinctive. Even smaller peaks, e.g. around 10:00, are noticeable. Of course the results are not completely similar, as the the supply is generated by an evolutionary process and an optimal solution is hardly to identify. That is, a good solution is maybe not the best possible, but good enough to dominate newly founded operators.

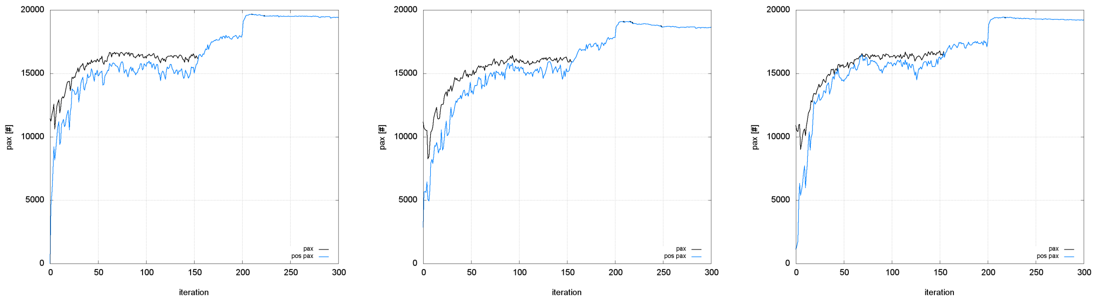
There are several reasons why agents do not not reach their destination, when using the minibus. Tab. 6.1 shows a detailed analysis for the 10 sensitivity runs performed. 3 main reasons are identified as described in the following.

- Agents try to travel using an invalid route, i.e. the route does not exist anymore. This may happen because the agents route search is disabled in iteration 200. Before that only 40 % search for new routes in every iteration. Although operators stop replanning in iteration 150 some may become bankrupt, because the budget still needs to be balanced. Consequently agents may store invalid routes.
- Agents missed the last departure. This may happen because they get delayed on previous trips.
- The boarding was not possible. This may happen because of fully occupied vehicles.

Table 6.1: Analysis of agents stucked in the last iteration.

run	route exists not	missed last departure	boarding not possible	total
1	169	13	9	191
2	131	13	2	146
3	124	21	4	149
4	169	14	5	188
5	165	9	2	176
6	194	19	7	220
7	213	26	9	248
8	165	12	12	189
9	122	11	4	137
10	256	21	28	305
min.	122	9	2	137
max.	256	26	28	305
avg.	170.8	15.9	8.2	194.9
std. dev.	39.8	5.2	7.3	48.9

Although the histogramms described before look quite similar, little deviations may be noticed. Furthermore the number of agents en route say nothing about the total number of passengers transported, rather the number of performed legs. Now, in Fig. 6.2 the change of the number of executed minubus legs is shown. On the first look all three runs show quite similar trends. Up to approximately iteration 150 (replanning and founding of the minibus operators is disabled) the number of transported passengers and positive scored passengers deviates. That results from the fact that operators running services that may be not profitable respectively able to fit the demand. Now, these operators die out after a short time and no new operators are founded. Existing operators may increase there service. Agents are allowed to adapt to the services up to iteration 200. After that in all 3 runs a discontinuity occurs. This happens because from now on agents are only allowed to choose from their existing plans, with the highest probability for the best plan. That is, no more disadvantageous plans are created and all agents can switch to the best opportunity, witch adapts by buing/selling vehicles as necessary. The average of performed legs for the 10 sensitivity runs is given with 18,916. The standard deviation is given by 670 (see Tab. 6.2). According to this, the average minibus trip consists of 2.2 legs.



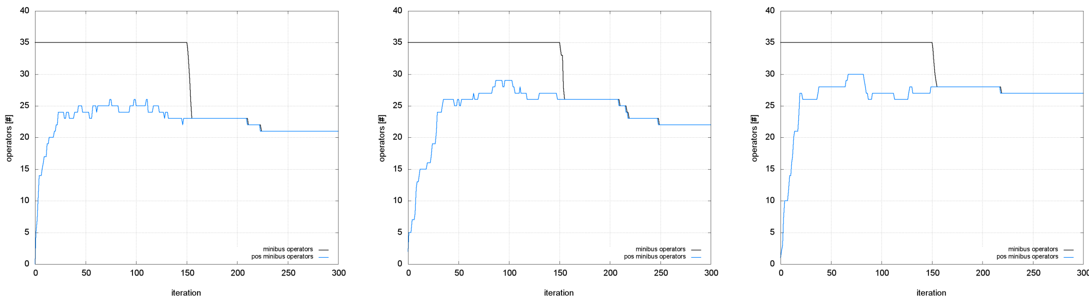
(a) run3 (b) run5 (c) run8

Figure 6.2: Evolution of minibus passengers.
Each agent entering a minibus is counted as a passenger.

The growth of operating minibus provider is shown in Fig. 6.3. As already seen among the evolution of number of transported passengers discontinuities occur short after iteration 150 and 200. After iteration 150 unprofitable operators die out and the number of total operators approaches the number of operators running profitable. The discontinuity after iteration 200 may be explained as follows. Agents are no longer allowed to create and test new routes. They will use their existing opportunities and will choose the best with the highest probability. As operators are still allowed to buy/sell vehicles the best opportunities increase their service frequency. Hence, worse alternatives lose demand and die out. That, of course, is the main explanation for the stucking agents displayed in Fig. 6.1.

The deviation of the number of cooperatives results from the fact that each operator is allowed to operate as many routes as possible. Namely, the 21 operators in run 3 operate 79 routes with 392 minibuses and a total length of 3,923 km, the 22 operators in run 5 operate 61 routes with 335 minibuses and a total length of 3,274 km and the 27 operators in run 8 operate 74 routes with 394 minibuses and a total length of 3,382 km. A comparison of the number of vehicles in the simulation and in reality seems not to be very useful, as the number of performed trips deviates seriously. Thus, the number of necessary vehicles deviates as well.

The manner of the evolution of the minibus system as described before is similar. In all analyzed dimensions the qualitative growth is comparable. Key points like peaks in the histogramm or discontinuties occur in the same iterations. Merely the detailed qualitative analyses show different results. This is not suprising, as the underlying evolutionary algorithm promise to find good, but not necessarily the best solution within a certain amount of time (Sec. 4.5). However, the quality of the system is comparable for all runs as shown in Sec. 6.4.



(a) run3 (b) run5 (c) run8

Figure 6.3: Evolution of the number minibus operators.

6.2. Accessibility of the minibus system

After the evolution of the minibus system was researched before, the fitting of demand and supply should be investigated now. That is, the accessibility of the minibus system is compared to numbers derived from the "South African National Household Travel Survey" [1, p. 38]. Fig. 6.4 shows the simulated values of the accessibility of minibus stops from homelocations within certain distances. The majority of the households (approx. 90%) can access the minibus system with a walk of less than 1,000 m. Compared to reality the simulation overestimates marginal. 95% of all households may access the system within a distance of 2,000 m. This applies for the simulation as well as for the reality. This distance is specified as comon in South Africa to access public infrastructure by the "South African National Household Travel Survey" [1] as well as by local traffic experts.

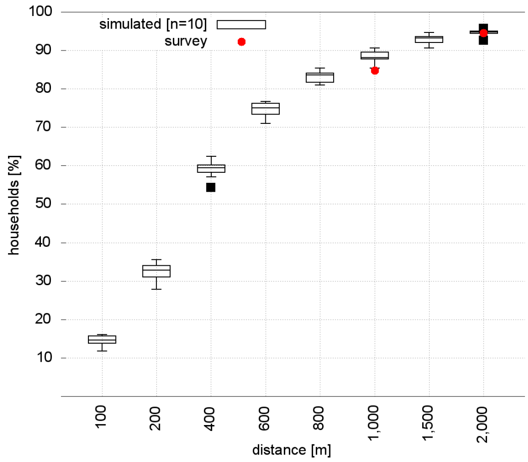


Figure 6.4: Accessibility of the minibus system from household locations. The red dots show numbers from the “South African National Household Travel Survey” for a “typical South African metropolitan Area” [1, p. 38] based on a walking speed of 4 km/h. The black boxplots show the values of 10 sensitivity runs.

Another important question concerning the accessibility of the system is, will the system identify and connect neighborhoods with a low demand (low density of home activities) to the system. Fig. 6.5 shows the catchment areas of the resulting minibus systems (after iteration 300). Remembering Fig. 5.1a 3 areas with a low density of home activities – located at the outskirts of the city – are identified. All of these areas are connected to the system with walking distances less than 500 m to the next minibus stop. This applies for the 3 mainly investigated runs as well as for the remaining 7. Particularly impressive is the shape of the displayed catchment areas, it is almost equal for all runs. From all places in the city center the system is accessible within 500 m. The same may be noticed for the beach front (south east) and Kwanobuhle (north west).

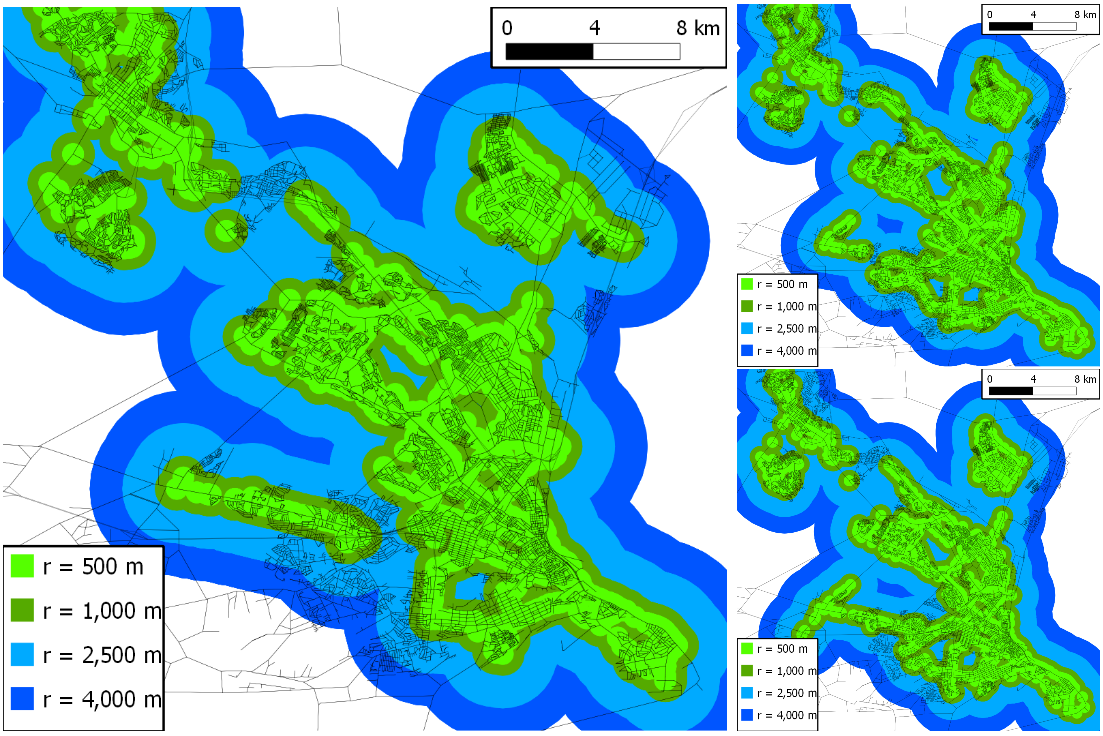


Figure 6.5: Catchment area of the minibus system for run 3.
Shown are the distances within a served minibus stop is accessible. Smaller figures show the results of sensitivity runs 5 and 8.

6.3. Minibus traveltime and trip statistics

The resulting travel times (Tab. 6.2) and trip statistics show a good quality. There are no serious discontinuities between the sensitivity runs and the deviations from the average/median are acceptable. Merely the total number of trips performed deviates from the number of trips planned (9,261 see Tab. 5.1). Of course, this is a problem. But, the explanation may be found in the number of agents that stuck during the simulation (Sec. 6.1). Although the number of stucking agents is smaller than the difference between planned and executed trips. Agents that stuck in their first trip will generate one stuck only, but all of their remaining trips will not be performed.

However, the evaluated travel times, shown in Tab. 6.2, fit the expectations. The “South African National Household Travel Survey” [1, p. 121] for example specifies travel times for work trips in “typical metropolitan areas” in South Africa as follows. About 50% of all trips have a duration up to 30 minutes, about 35% take approximately 30 to 60 minutes and the remaining 15% are specified with a travel time over 60 minutes. That is, the simulated average travel time of approximately 40 minutes is within the expectations.

Table 6.2: Minibus traveltime and trip statistics (n=10). Details may be found in Appendix D.

		avg.	median	std. dev.	min.	max.
avg. trips	[#]	8,466.7	8,491.0	94.4	8,271.0	8,596.0
avg. legs	[#]	18,916.4	19,138.5	669.9	17,083.0	19,431.0
avg. boardings / trip	[#]	2.2	2.3	0.1	2.1	2.3
avg. traveltime / trip	[min]	39.8	39.7	1.5	37.7	43.2
avg. access walk / trip	[min]	10.5	10.3	0.2	10.2	10.8
avg. in vehicle time / trip	[min]	9.3	9.3	0.2	9.1	9.6
avg. egress walk /trip	[min]	9.2	9.2	0.3	8.7	9.7

6.4. Passenger volumes

The passenger volumes and the number of performed trips are some of the main aspects to validate a (public) transport model. For that, the number of transported passengers should be analyzed.

Hence, data from the Transport Plan of the NMBM [6] is derived. Firstly their is the number of total trips. Although, this number is not clearly pointed out, as there are two. The number of "passengers boarded taxis daily" is given with 119,000 [6, p. 21]. Secondly, the number of person trips (minib) per day for the year 2009 [6, p. 19] is accounted with 372,866. Admittedly, this number is calculated by a VISUM-model. Compared to the 920,722 minibus-trips executed by the synthetic population (100%, Tab. 5.1) a serious deviation is observable. Reasons for that are given at the end of the section.

Furthermmore the data for 7 count stations is derived from the Transport Plan of the NMBM [6]. The locations of the stations is described on p. 35 in the Transport Plan. Following this description the stations are located in the used network as shown the map in Appendix E. The given numbers are scaled to 1% (according to the used population sample) and compared to the outcome of the simulation in Fig. 6.6. Analyzing the diagramm serious inconsistencies are observable, for the real-world and the simulated data.

But, there are serious deviatons for the simulated numbers as well. For station B a mean of 1,022.7 passengers and standard deviation of 365.9 is measured. For station C a mean of 1,371.0 and a standard deviation of 269.3 is evaluated (Appendix E). This, of course, is a major problem but can be explained as follows. In Fig. 6.7 the passenger volumes for the time from 6:00 to 18:00 for each link in the city center are displayed. Furthermore the locations of the count stations are marked. Looking at station B and C it becomes clear that both point to the same direction, namely from the northwest to the southeast. Comparing the figures for run 3 and 5 both look quite similar. Remembering Fig. 6.6 this was expectable as the simulated numbers are quite similar. Comparing both to run 8 a serious deviation at station B and a minor deviation at C is observable. The demand that was using the streets crossing station B before, moved in run 8 a little bit northwards, using the street along the coast.

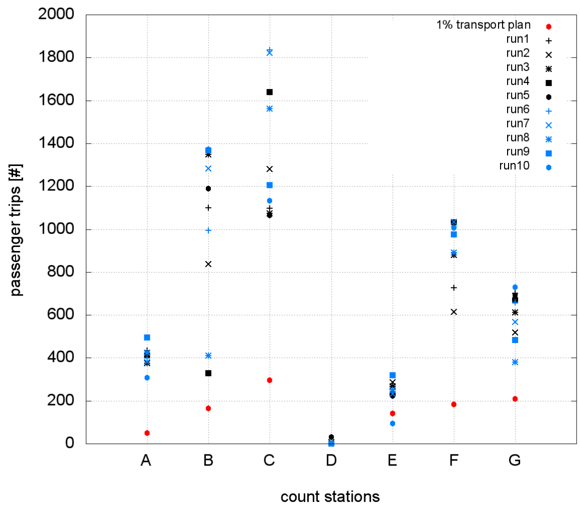
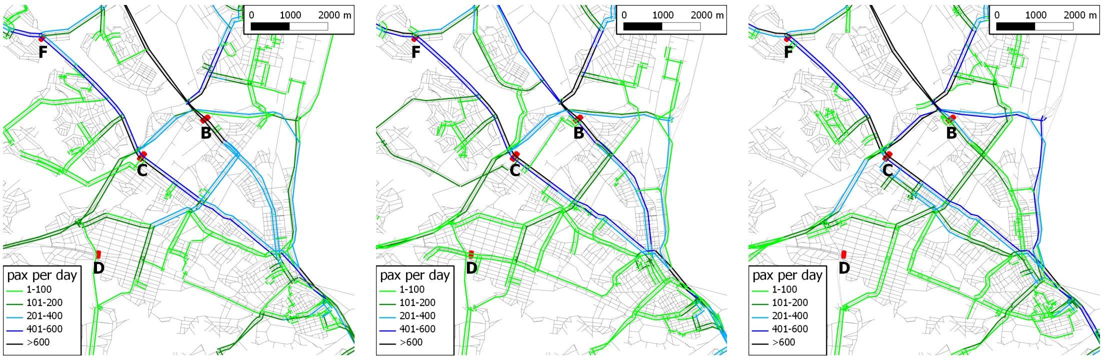


Figure 6.6: Minibus passenger trip counts from 6:00 to 18:00.
Shown are the results from ten different runs (same config parameters, but different random seeds) and the evaluated numbers from the NMBM transport plan [6, p. 23].

Another inconsistency is observable at station D. The simulation will not find this corridor in every run, but only in 2 out of 10. That is because the demand desiring to use this corridor is very low. For a detailed analysis of this problem the reader is referred to Sec. 6.6.

Considering the complete research area (Fig. 6.8) the outcome of all 3 runs – qualitative and quantitative – is comparable. That is, especially the very high demand in the city centre is measured for all simulations, which results from a certain amount of transfer stops. For that the boarding and alighting activities are analysed in the following section. Although the resulting demand is comparable for all sensitivity runs the resulting routes are not stable. That is; because the algorithm do not necessarily find the best but good options, as mentioned before. As long as the number of collected fares is high enough to run a service an operator will offer it. Agents use that service as long as they do not experience a loss in travel time. Tab. 6.2 shows that the resulting routes do not affect the agents result.

Summarizing the results pointed out in this section the accessible data seems not very suitable for validation reasons. Counting minibus trips is very difficult as there are no fixed stations (except the ranks) in reality. That is, passengers can board and alight everywhere they want to. Because of that, because of the high reliability of the underlying dataset of the synthetic population, because of the high number of "pirate taxis" [5, 6] and because of the information given by local experts the numbers provided by the transport plan are classified as not very reliable.



(a) run 3 (b) run 5 (c) run 8

Figure 6.7: Minibus passenger counts from 6:00 to 18:00 – city center.

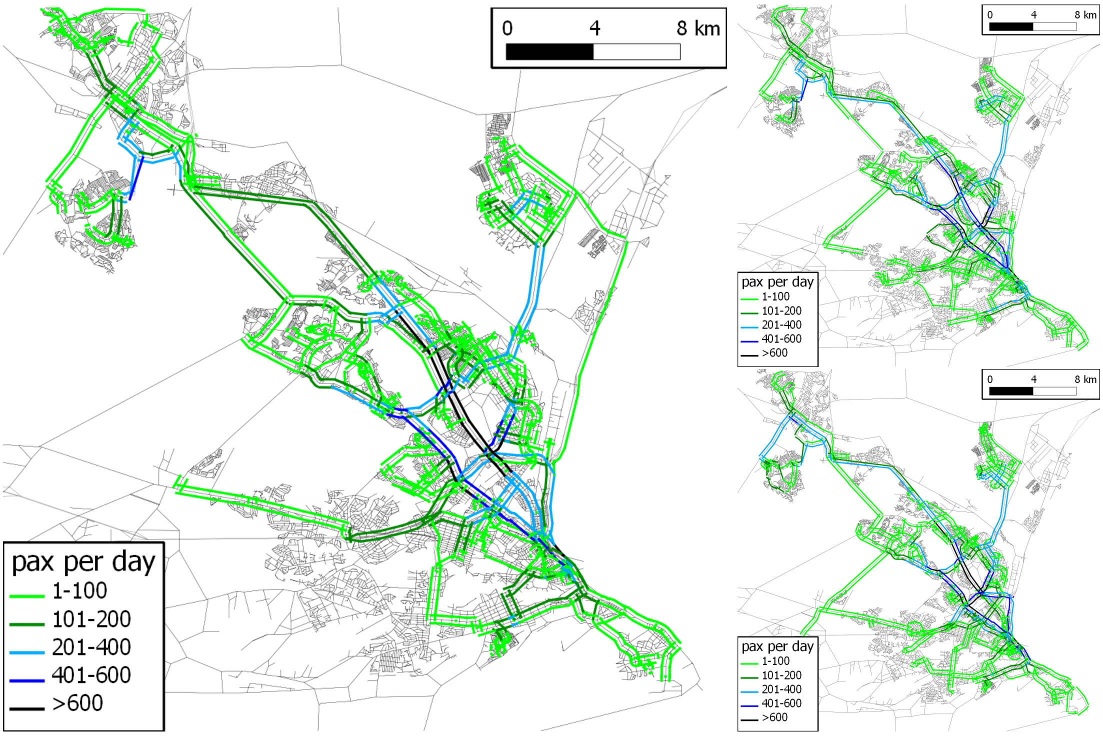


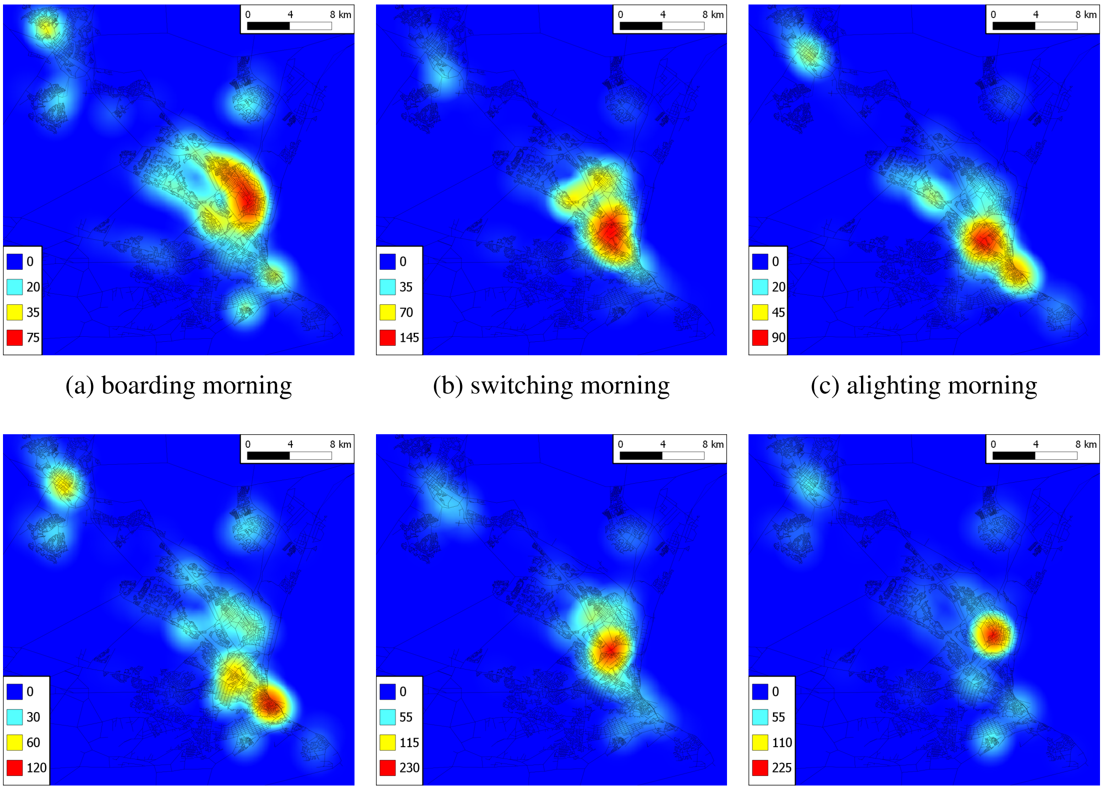
Figure 6.8: Minibus passenger volumes.
Shown are the numbers of transported passengers per link for a whole day.
Smaller figures show the results of sensitivity runs.

6.5. Boarding & alighting

To validate the correctness or at least plausibility of the resulting minibus traffic the locations where people live may be used. Individuals will start their way to work in the morning and will leave their working place in the afternoon. That is, the places where minibus trips start in the morning should be close to the areas of living. Places where people finish their trips in the morning should be close to working facilities. Remembering the spatial distribution of activities (Fig. 5.1) one should expect the distribution of boardings and alightings to be equal. For that, two time slices are analyzed. The morning peak between 7:00 and 8:00 and the afternoon peak between 15:00 an 16:00. To analyze the activities, again a heatmap-scheme is used. Red indicate areas with the highest measured numbers. Blue indicate areas with no or less activity measured.

Looking closer to the results for the morning peak (Fig. 6.9a, b and c) one will notice the highest numbers for boarding north of the city center within a fairly large area, which also shows the highest number of living activities. Furthermore there are five areas with less numbers of boardings, in the north-west, the north-east and the south-west. All but the one close to cost are areas of living as well. Certainly the number of home activites is lower there and thus, the number of boarding activities is smaller. The switching activities bundle at the city center. Furthermore a hotspot at the Kwanobuhle Access (north-west) may be noticed.The travellers mainly alight close to the harbour, where the highest number of working activities is located.

For the afternoon peak (Fig. 6.9d, e and f) the highest number of boarding activities is performed where the most traveller alight in the morning. Switchings again take place in the city center. However, the alighting bundles not at the places where the highest number of home activities is measured, but where the highest number of leisure activities takes place (compare Appendix C).



(a) boarding morning (b) switching morning (c) alighting morning

(d) boarding afternoon (e) switching afternoon (f) alighting afternoon

Figure 6.9: Passengers boarding & alighting minibuses.
The figures show where passengers board, switch and alight minibuses. The morning peak is measured between 7:00 and 8:00, the afternoon peak between 15:00 and 16:00.

6.6. Countstation D

As already found in Sec. 6.4 the corridor crossing Countstation D is not served with satisfactory steadiness, because of the low demand. Now the question is, if these users experience a serious loss that results from the missing connection. First of all potential users need to be identified. For this purpose two areas, one at the eastbound and one at the westbound of the countstation, need to be defined (compare Fig. 6.10). Now, the potential users are filtered according to the following rules. The trip starts either in the first or in the second area. Assuming it starts in the first area, the destination must be in the second area. The desired mode of the trip is the minibus. All travellers are taken into account, whose daily plan contain at least one trip fitting this scheme.

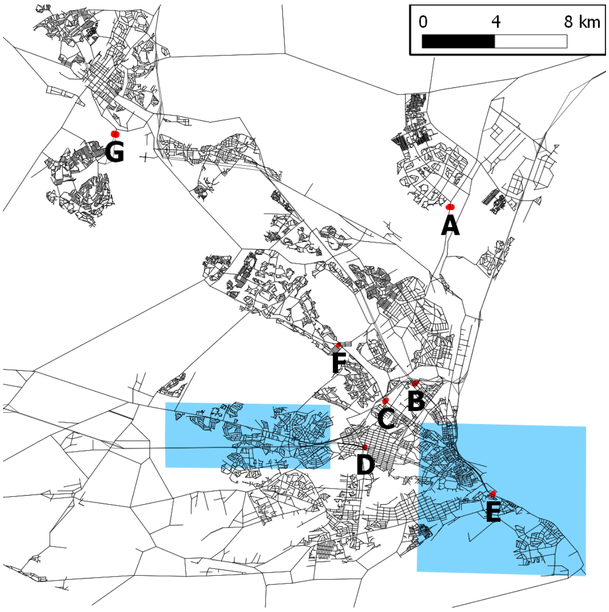


Figure 6.10: Potential living and working areas (blue) that affect count D.

Following the rules described above only 22 travellers, with a maximum of 58 desired minibus trips, are identified. Now, the question is will travellers experience a loss, when no direct connection is found between those areas. Looking closer to the evaluated numbers in Tab. 6.3, the question may be answered with no.

In 2 runs (2 and 5) a direct connection between both areas is established. In run 5 the best solution for this group of users is found, with low travel times and less boardings per trip. Against that, evaluated numbers for run 5 are above the averages. Furthermore the number of successful performed trips (53) is lower. One would expect a better performance for those agents when a direct connection is established. However, the attraction to offer a service along this corridor is very low, as there are only 22 potential users (0.2 % of the complete population) which perform only 0.6 % of all minibus trips in the complete research area. Thus, the impulse to offer a service for this corridor is very small for the operators.

Table 6.3: Detailed traveltime and trip statistics for potential users of Count D.

		run				
		1	2	3	4	5
trips	[#]	57.0	53.0	54.0	57.0	57.0
legs	[#]	176.0	167.0	165.0	166.0	166.0
avg. boardings / trip	[#]	3.1	3.2	3.1	2.9	2.9
avg. traveltime / trip	[min]	54.9	54.3	54.8	44.8	44.8
avg. access walk / trip	[min]	14.6	12.9	16.1	12.0	12.0
avg. in-vehicle-time / trip	[min]	11.0	11.3	10.4	9.8	9.8
avg. egress walk / trip	[min]	12.0	10.7	12.7	10.6	10.6

		run				
		6	7	8	9	10
trips	[#]	55.0	57.0	55.0	58.0	50.0
legs	[#]	141.0	185.0	180.0	176.0	141.0
avg. boardings / trip	[#]	2.6	3.2	3.3	3.0	2.8
avg. traveltime / trip	[min]	49.7	55.6	55.5	47.1	47.9
avg. access walk / trip	[min]	13.1	10.4	12.8	11.8	13.1
avg. in-vehicle-time / trip	[min]	10.2	12.1	11.3	11.5	10.3
avg. egress walk / trip	[min]	11.4	10.4	11.9	11.1	12.0

7. Summary & Conclusion

This work aims on creating a South African minibus network. For that, only the initial demand and a street-network is used. Relations are identified by an evolutionary algorithm. The work starts with a brief description of paratransit in general and South African minibusses in particular. The used model (MATSim) with a focus to the public transport and the paratransit module is described in detail. Afterwards a detailed description of the model extension – implemented for this work – is given. The extension is tested in detail. Afterwards the used real world scenario – the Nelson Mandela Bay Municipality (NMBM) – is introduced and the simulation setup is explained. A detailed analyis is applied to the NMBM-scenario.

The implemented extension allows a detailed routing of public transport (pt) users for an arbitrary number of modes in the public transport. In case a very detailed knowledge about the used (pt) travel modes is available agents are no longer routed through the whole pt-network, but only on the desired mode. For example, agents that desire to travel minibus will not be forced to travel with a commuter rail, even when the expected utility is better. That is, the implicite reroute in the public transport is disabled. One may argue this is a step back, as the routing should be based on a generalized utility function. However, a utility function need to be calibrated to deliver reliable results. Especially in the South African context the decision making of public transport users is not only based on travel time, but more on intrinsic motivation and personal income. Thus, the calculation of utilities becomes more complicated. Furthermore a calibration of the model is more difficult, because the available data about (e.g. trips performed or vehicles counted) is neither very detailed nor reliable, in terms of public transport in general and minibusses in special. The logical conclusion is, to use the detailed knowledge about used means, thus route a desired minibus trip only on the minibus network.

The extension is first tested with a minimal scenario and delivers the expected results. Agents are forced to use the mode they planned to use before the routing. The paratransit module identifies always the best route. Certainly, the algorithm do not find the perfect time of operation in every simulation. For some of the sensitivity runs a small slot at the end is missing. Thus not all agents are transported. However, the found solutions are close to the expected (optimal) solutions.

The NMBM scenario is tested without a detailed transit schedule, because the license did not allow a publishing. However, the resulting minibus network is close to reality, as far as reliable validition data is available. For that “The First South African National Household Travel Survey” [1] and the “NMBM Transport Plan”[6] are used. The number of performed minubus trips deviates seriously between the initial plans and the derived data for validation. Although, the coverage of the service area and the realized travel times fit the expectations. Furthermore the stability of the results is sufficient. Areas of low demand are connected to the network for all sensitivity runs. Merely one relation is not distinct in all sensitivity runs, as the according demand covers only less than 1% of the complete demand. Thus, the attraction for operators is to small.

The proposed model delivers good results in the South African context. Admittedly a few typical characteristics of the system may be enhanced for future research. Namely, these are:

- Typically vehicles depart only when they are fully loaded. A within-day-decision-model for the drivers could solve this problem.
- The average houshold income is very low. That is, the decision how much money is spend for transport is very important. An routing process that implements fares seems to be necessary for future implementations. According to this a feedback in the scoring function is necessary.
- With a routing and scoring that implements fares a detailed and maybe personalized decision-model may be implemented, based on intrinsic motivation and personal income.
- According to the previous item the fare structure should be enhanced. Currently only a flat fare is collected. It is indenpently of the length of the route. That is, long- and short-distance routes charge the same fare. Although, the model offers a posibility to implement distance-based fares for the operators it is useless without an implementation in the agents routing and scoring process. Without that users will choose the fastest option regardless of the fare. This will lead to longer routes, maybe with useless detours, as the operators will gain more profit.

References

[1] DoT. The First South African National Household Travel Survey 2003. Technical report, Department of Transport, Pretoria, South Africa, 8 2005.

[2] Johan W. Joubert. Paratransit: perpetual pain or potent potential? ! Technical report, University of Pretoria, unpublished, 2012.

[3] Johan W. Joubert and Susan Woolf. When paratransit evolves to become the answer, taxi hand signs point the way! Technical report, University of Pretoria and University of Witwatersrand, unpublished, 2012.

[4] L. J. Fourie. Rethinking the formalisation of the minibus-taxi industry in south africa. Master's thesis, University of Pretoria, 2003.

[5] City of Johannesburg. *Integrated Transport Plan 2003 / 2008*. City of Johannesburg, 2004.

[6] SSI Engineers and Environmental Consultants. *Comprehensive Integrated Transport Plan 2011/12*. Infrastructure and Engineering - Directorate Nelson Mandela Bay Municipality, 2011.

[7] Jackie Dugard. From Low Intensity War to Mafia War: Taxi violence in South Africa (1987 - 2000). 2001.

[8] L. Quadrifoglio, M.M. Dessouky, and F. Ordóñez. A simulation study of demand responsive transit system design. *Transportation Research Part A*, 42:718–737, 2008.

[9] Liping Fu. A simulation model for evaluating advanced dial-a-ride paratransit systems. *Transportation Research Part A: Policy and Practice*, 36(4):291 – 307, 2002. ISSN 0965-8564. doi: DOI:10.1016/S0965-8564(01)00002-7. URL `http://www.sciencedirect.com/science/article/B6VG7-44N9N9D-1/2/511466d3831a5f5a8ffe99885b7adb37`.

[10] A. Neumann and K. Nagel. A paratransit-inspired evolutionary process for public transit network design. Annual Meeting Preprint 12-0716, Transportation Research Board, Washington D.C., 2012. Also VSP WP 11-15, see www.vsp.tu-berlin.de/publications.

[11] A. Neumann and K. Nagel. Passenger agent and paratransit operator reaction to changes of service frequency of a fixed train line. In *Proceedings of the 13th Conference of the International Association for Travel Behaviour Research (IATBR)*, Toronto, Canada, 2012. Also VSP WP 12-15, see www.vsp.tu-berlin.de/publications.

[12] MATSim. Multi-Agent Transportation Simulation Toolkit. last visited: Feb 20th, 2012, 2012. URL `http://www.matsim.org`.

[13] M. Rieser. *Adding transit to an agent-based transportation simulation concepts and implementation*. PhD thesis, TU Berlin, 2010. Also VSP WP 10-05, see www.vsp.tu-berlin.de/publications.

[14] C. Kenneth Orski. Paratransit: The coming of age of a transportation concept. *Transportation*, 4:329–334, 1975. ISSN 0049-4488. URL `http://dx.doi.org/10.1007/BF00174734`. 10.1007/BF00174734.

[15] Daniel Roos and David Alschuler. Paratransit - existing issues and future directions. *Transportation*, 4:335–350, 1975. ISSN 0049-4488. URL `http://dx.doi.org/10.1007/BF00174735`. 10.1007/BF00174735.

[16] Robert Cervero. *Paratransit in america: Redefining mass transportation*. Greenwood Publishing Group, 1997. URL `http://books.google.co.za/books/about/Paratransit_in_America.html?id=6VrXI4qi6mAC&redir_esc=y`.

[17] Robert Cervero. *Informal Transport in the Developing World*. Number HS/593/00E. UN-HABITAT, 2000.

[18] Robert Cervero and Aaron Golub. Informal transport: A global perspective. *Transport Policy*, 14(6):445 – 457, 2007. ISSN 0967-070X. doi: DOI: 10.1016/j.tranpol.2007.04.011. URL `http://www.sciencedirect.com/science/article/pii/S0967070X07000546`.

[19] M. Khosa. Accumulation and labour relations in the taxi industry. *Transformation*, 24:55 – 71, 1994.

[20] H. Schalekamp and R. Behrens. An international review of paratransit regulation and integration experiences: lessons for public transport system rationalisation and improvement in south african cities. Technical report, Centre for Transport Studies, Department of Civil Engineering, University of Capetown., 2009.

[21] M. Balmer, B. Raney, and K. Nagel. Adjustment of activity timing and duration in an agent-based traffic flow simulation. In H.J.P. Timmermans, editor, *Progress in activity-based analysis*, pages 91–114. Elsevier, Oxford, UK, 2005.

[22] B. Raney and K. Nagel. An improved framework for large-scale multi-agent simulations of travel behaviour. In P. Rietveld, B. Jourquin, and K. Westin, editors, *Towards better performing European Transportation Systems*, pages 305–347. Routledge, London, 2006.

[23] M. Rieser and K. Nagel. Combined agent-based simulation of private car traffic and transit. In *Proceedings of The 12th Conference of the International Association for Travel Behaviour Research (IATBR)*, Jaipur, India, 2009. Also VSP WP 09-11, see www.vsp.tu-berlin.de/publications.

[24] A. Agarwal, M. Zilske, K.R. Rao, and K. Nagel. Person-based dynamic traffic assignment for mixed traffic conditions. VSP working paper 12-11, TU Berlin, Transport Systems Planning and Transport Telematics, 2012. URL `https://svn.vsp.tu-berlin.de/repos/public-svn/publications/vspwp/2012/12-11/`. See www.vsp.tu-berlin.de/publications.

[25] M. Balmer. *Travel demand modeling for multi-agent transport simulations: Algorithms and systems*. PhD thesis, Swiss Federal Institute of Technology (ETH) Zürich, Switzerland, 2007.

[26] Marcel Rieser. Controler structure, 07 2008. URL `http://matsim.org/docs/controler`.

[27] C. Gawron. An iterative algorithm to determine the dynamic user equilibrium in a traffic simulation model. *International Journal of Modern Physics C*, 9(3): 393–407, 1998.

[28] D. Charypar and K. Nagel. Generating complete all-day activity plans with genetic algorithms. *Transportation*, 32(4):369–397, 2005.

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

[30] K. Müller and K. W. Axhausen. Hierarchical IPF: Generating a synthetic population for switzerland. In *51st ERSA Conference*, 2011. doi: 10.3929/ethz-a-006620748.

[31] OpenStreetMap. Map data © OpenStreetMap contributors, CC BY-SA. http://www.openstreetmap.or, 2012. URL `http://www.openstreetmap.org`.

[32] M. Balmer, M. Bernard, and K.W. Axhausen. Matching geo-coded graphs. 2005.

A. Implementation

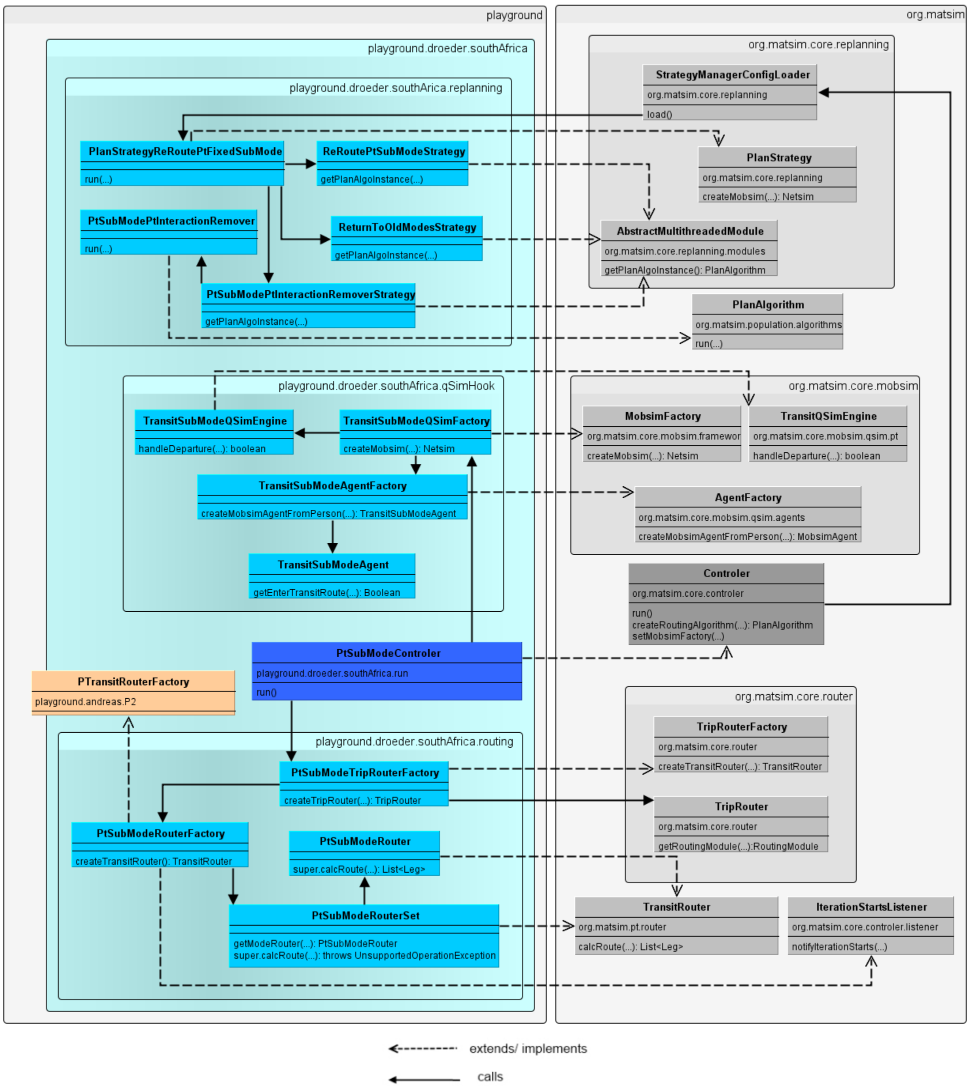


Figure A.1: Implementation of the model extension.
The figure shows the Java implementation of the applied model. On the right side MATSim is shown, so to say the interfaces used for the implementation. On the left side the own implementation is shown as it can be found in revision 22067 at `https://matsim.svn.sourceforge.net/svnroot/matsim/playgrounds/trunk/droeder/`.

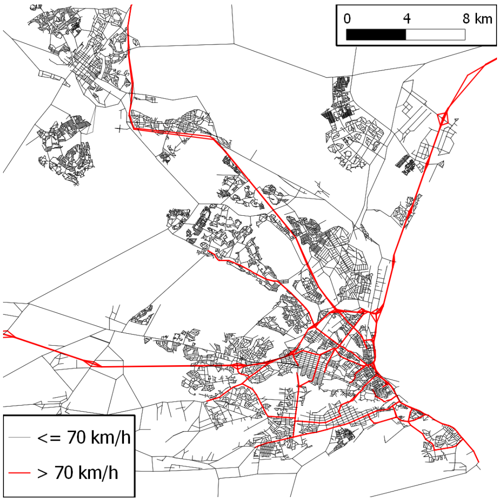
B. Duration of the routing-process

The duration of the routing when using MATSim's pt-router (see Sec. 3.2.3) highly depends on the number of implemented *TransitRoutes*, as every route is represented by own nodes and links in the graph. That is, two routes serving the same corridor with the same stops will not use the same infrastructure in the routing-network. Instead so called *TransferLinks* are added, connecting both routes (compare Fig. 3.4). The same will be done for all routes crossing each other at the intersection and for all stops within a certain distance. That is, the so called *TransitRouterNetwork* increases exponentatially when the density of a network increases. That might be explained with the example shown in Fig. B.1. The network of NMBM is used. In scenario a) displayed in Fig. B.1a stops are created on all links with a freespeed below 70 km/h. In scenario b) displayed in Fig. B.1b for all links with a freespeed below 81 km/h. This results in 35,289 possible stops in scenario a) and 36,435 stops in scenario b).

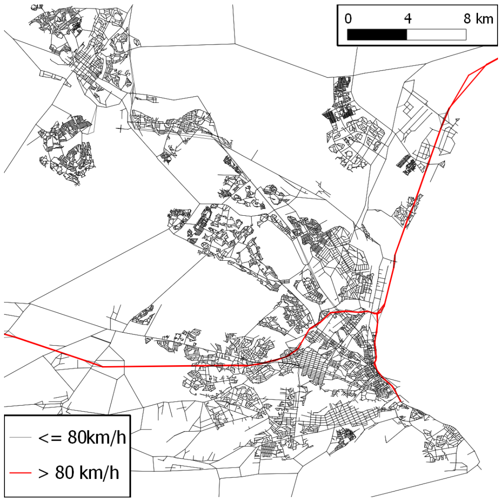
Using the network described above the following is done. 10 Cooperatives, each running one route with 10 vehicles, are founded and added to transitschedule. For that the initialization-procedure described in Sec. 3.3 is used. From that the *TransitRouter-Networks* and the *TransitRouters* are created. Then 5,000 routing-requests are executed, the same for both scenarios. Now the number of routes is increased by one, a new router is build and the same 5,000 requests are done. Fig. B.1c) shows the trend of the required time for the routing process. Although the number of possible stops from scenario a) to b) was increased for 3% only, the required routing time increased expontially. E.g. with 250 routes the routing in sceanrio a) takes approximately 400 s, in scenario amount of routing requests takes approximately 1,200 s.

The reason for that is the founded routes will always use the shortest/fastest path between start- and turning-point, which is are randomly drawn. The added links are the high capacitive and very fast links. That is, they are used more often then other links. This is the same in both scenarios. But because it is possible to stop there in scenario b) only, the routing takes much longer, because of much more alternatives. and thus more *TransferLinks*.

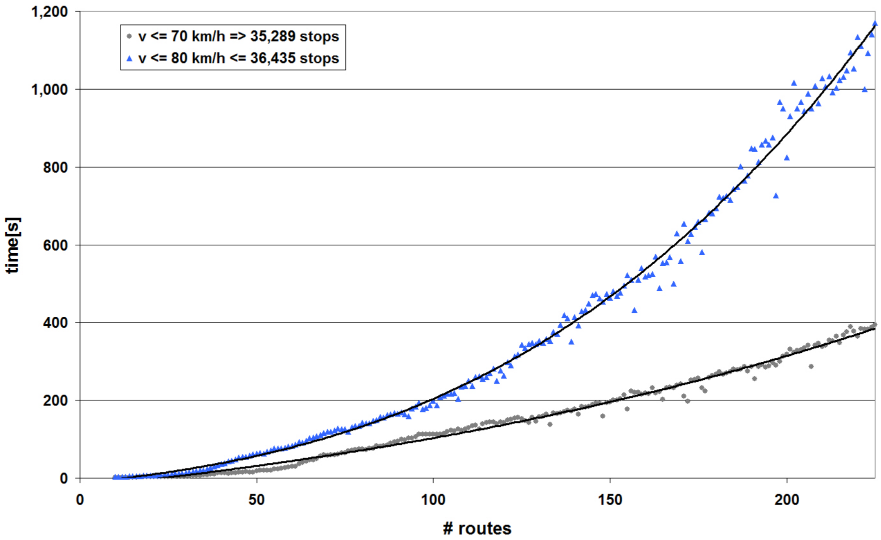
The benchmark was done on a Intel Xeon E7540 with 2GHz, using one thread and 10 GB RAM.



(a) Speedlimit 70 km/h



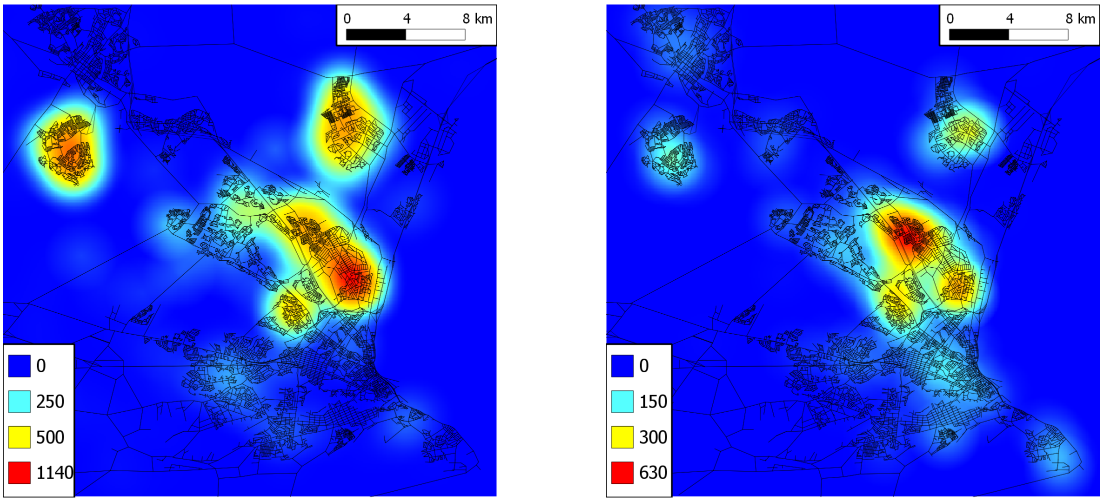
(b) Speedlimit 80 km/h



(c)

Figure B.1: Routing performance as function of the number of routes and possible stops. Red links are links with a freespeed above the speed limit for minibus stops.

C. Activity locations



(a) Leisure-activities

(b) Education-activities

Figure C.1: Spatial diversification of activities.
The figures show heatmaps for the different activity-types performed in the NMBM-area. They should be read as follows. An Activity has got an influence of 1 at its location up to 0 in a distance of r = 2,500 m. In between it is linear interpolated. This has been done for every activity. The shown heatmaps are the results of the accumulation of all activities of one type.

D. Traveltime and trip statistics

Table D.1: Detailed traveltime and trip statistics.

		run				
		1	2	3	4	5
trips	[#]	8,500.0	8,530.0	8,506.0	8,482.0	8,481.0
legs	[#]	19,348.0	19,186.0	19,288.0	19,415.0	18,486.0
avg. boardings / trip	[#]	2.3	2.2	2.3	2.3	2.2
total trip traveltime	[h]	5,665.0	5,890.5	5,440.3	5,554.1	5,654.4
total acces walk	[h]	1,497.1	1,525.4	1,451.3	1,456.4	1,467.9
total in vehicle	[h]	1,322.2	1,367.2	1,302.2	1,308.1	1,343.0
total egress walk	[h]	1,234.6	1,375.6	1,277.6	1,288.2	1,333.5
avg. traveltime / trip	[min]	40.0	41.4	38.4	39.3	40.0
avg. access walk / trip	[min]	10.6	10.7	10.2	10.3	10.4
avg. in-vehicle-time / trip	[min]	9.3	9.6	9.2	9.3	9.5
avg. egress walk / trip	[min]	8.7	9.7	9.0	9.1	9.4

		run				
		6	7	8	9	10
trips	[#]	8,342.0	8,405.0	8,554.0	8,596.0	8,271.0
legs	[#]	18,922.0	18,914.0	19,091.0	19,431.0	17,083.0
avg. boardings / trip	[#]	2.3	2.3	2.2	2.3	2.1
total trip traveltime	[h]	5,359.9	5,510.5	5,759.8	5,396.8	5,961.6
total acces walk	[h]	1,431.3	1,432.0	1,527.8	1,464.9	1,494.9
total in vehicle	[h]	1,319.9	1,275.0	1,351.8	1,339.6	1,255.8
total egress walk	[h]	1,262.0	1,293.9	1,347.9	1,309.9	1,289.5
avg. traveltime / trip	[min]	38.6	39.3	40.4	37.7	43.2
avg. access walk / trip	[min]	10.3	10.2	10.7	10.2	10.8
avg. in-vehicle-time / trip	[min]	9.5	9.1	9.5	9.4	9.1
avg. egress walk / trip	[min]	9.1	9.2	9.5	9.1	9.4

E. Count Stations

Table E.1: Simulated Minibus passenger trip counts from 6:00 am to 18:00 and data derived from the transport plan [6].

station\run	1	2	3	4	5
A Dibanisa Rd (Motherwell)	434	422	375	413	417
B Commercial Rd	1,100	838	1,348	328	1,188
C Standford Rd (Livingston Hospital)	1,098	1,281	1,073	1,638	1,064
D CapeRd	0	8	0	0	31
E Humewood / Beach Rd	262	285	268	231	222
F Standford Rd (Bethelsdorp)	728	615	879	1,031	1,030
G Kwanobuhle Access	700	519	612	669	691

station\run	6	7	8	9	10
A Dibanisa Rd (Motherwell)	398	380	424	494	306
B Commercial Rd	993	1,283	411	1,367	1,371
C Standford Rd (Livingston Hospital)	1,837	1,821	1,562	1,204	1,132
D CapeRd	0	0	0	0	0
E Humewood / Beach Rd	318	232	251	319	93
F Standford Rd (Bethelsdorp)	1,010	1,035	892	975	1,006
G Kwanobuhle Access	658	567	381	484	729

station\statistics	avg.	median	stdDev	transport plan (1 %)
A Dibanisa Rd (Motherwell)	406.3	415.0	45.8	49
B Commercial Rd	1,022.7	1,144.0	365.9	165
C Standford Rd (Livingston Hospital)	1,371.0	1,242.5	296.3	295
D CapeRd	3.9	0.0	9.3	30
E Humewood / Beach Rd	248.1	256.5	61.0	140
F Standford Rd (Bethelsdorp)	920.1	990.5	137.3	184
G Kwanobuhle Access	601.0	635.0	105.9	208

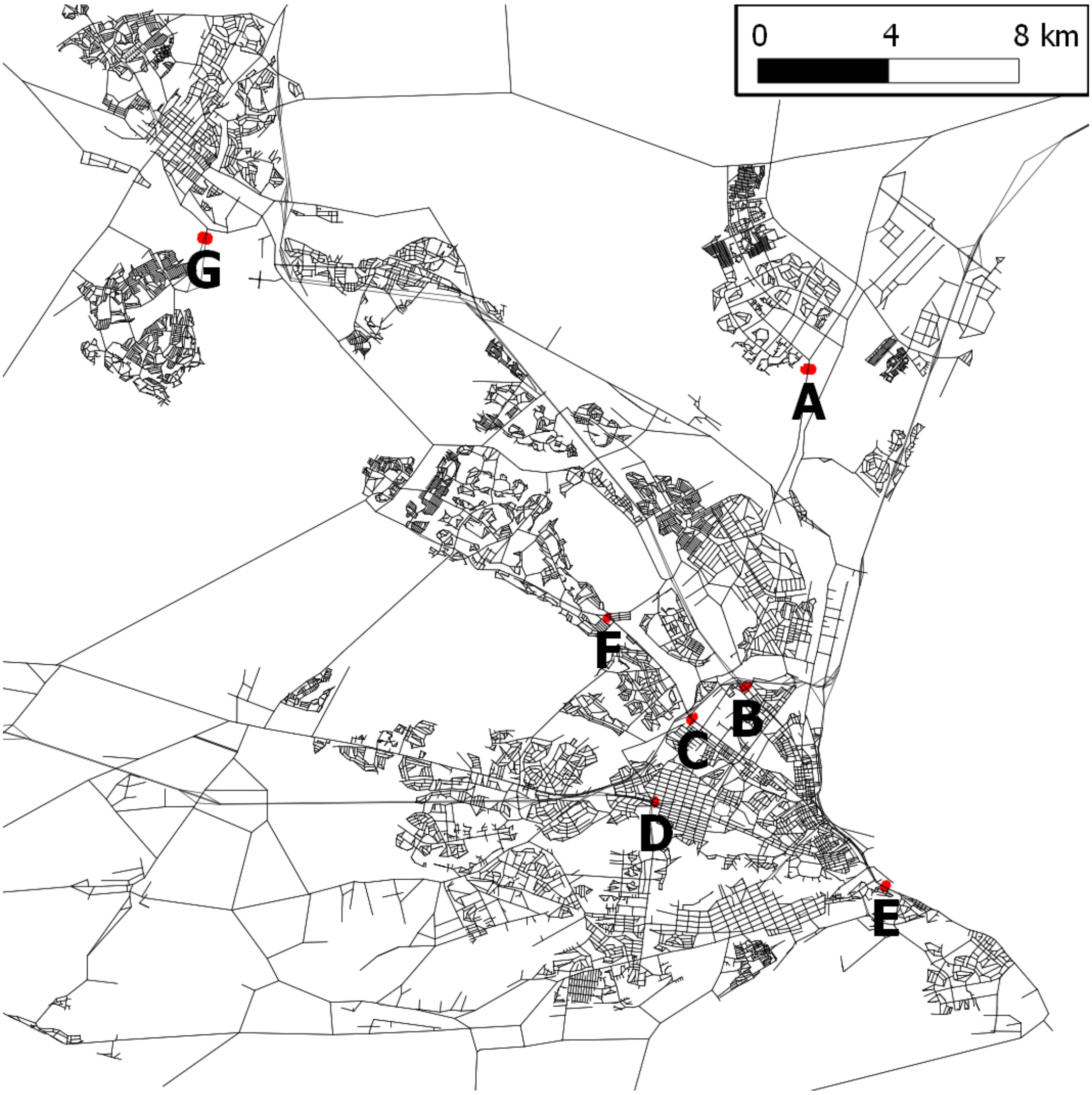


Figure E.1: Locations of minibus counts.

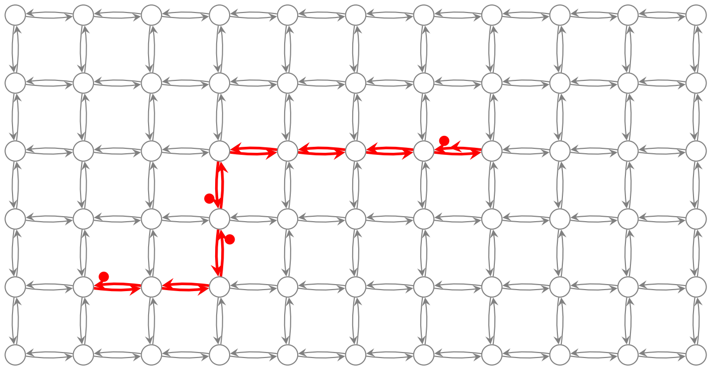
F. Paratransit route-replanning

The replanning of the cooperatives was more or less random-based. A new cooperative started with a origin-destination pair, the shortest route in between and a randomly set operationtime as described in Sec. 3.3. . Before every iteration a profitable cooperation can extend or reduce the operationtime, but is not able to extend or change the number or the order of served facilities in a systematicly way. This results in a suspected high number of iterations until a cooperative is generated which serves an optimal or even better route as the ones created before. Therefore the following replanning-modules are invented and described.

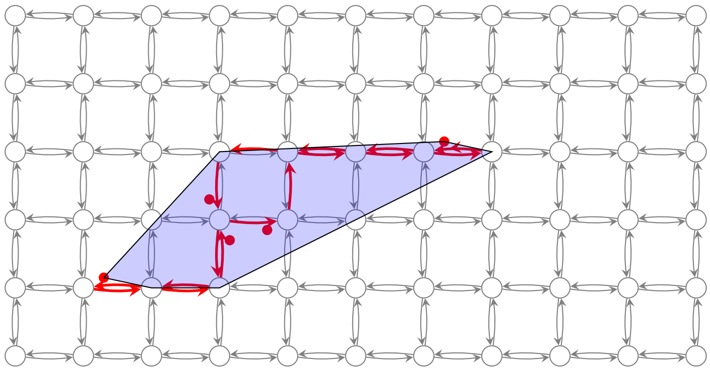
ConvexHullRouteExtension The *ConvexHullRouteExtension* spread out an convex area covering all TransitStopFacilities served by the processed cooperative. A set of facilities located within this area (blue area in Fig. F.1b) and at this time unserved by the cooperative is build. Afterwards a facility f out of this set is drawn randomly. Now this facility has to be added to the list of stops to be served. To minimize the risk of unecessary loops and to provide the risk of routes serving unecessary detours through the network, a bit more computation is necessary. Therefore the route is divided into a set of subroutes j containing all stops between 2 stops to be served. Then the average distances d_j from f to all stops of a subroute are calculated for every subroute.

$$d_j = \frac{\sum_{i=1}^{n} \sqrt{(x_i - x_n)^2 + (y_i - x_n)^2}}{n} \tag{F.1}$$

Now f is inserted to the stops to be served list between the two stops of subroute j with the smallest average distance d_j. Fig. F.1b shows a possible outcome of the module processing the route shown in Fig. F.1a.



(a) Route before replanning



(b) Possible route after replanning and the area to find a new Stop

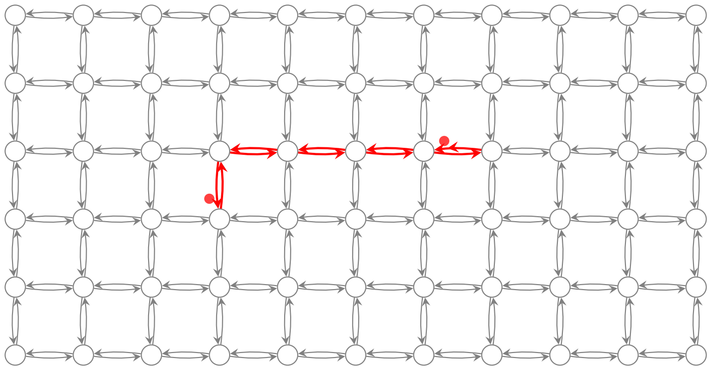
Figure F.1: Example for ConvexHullRouteExtension. Red Dots indicate the stopsToBeServed. Red arrows indicate the route. Double arrows indicate termini. Blue areas indicate the searchspaces within the algorithms try find a new stop.

RectangleHullRouteExtension This module is used for extension somewhere between the beginning and the end of a route. For this an rectangular area between start- and endstop is spreaded out. The length l of the rectangle is given by the length of the corridor $\overline{r1}$ (compare Sec. 4.4). The width w of the rectangle is set relatively to l by using the parameter p defined in the configfile.

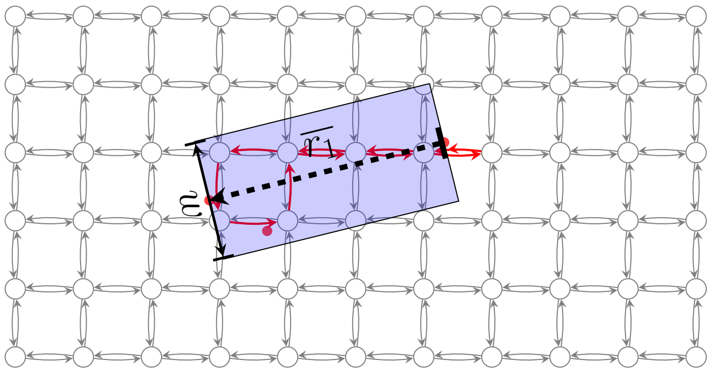
$$l = |\overline{r1}| \tag{F.2}$$

$$w = p \cdot l \tag{F.3}$$

The resulting geometry for the route shown in Fig. F.2a is shown in Fig. F.2b. A stopfacility f – located within this area and currently unused – is drawn randomly. The route is divided into a set of subroutes j containing all stops between 2 stops to be served. The average distances d_j from f to all stops of a subroute are calculated for every subroute, as shown in equation F.1. Then f is inserted to the stops to be served list between the two stops of subroute j with the smallest average distance d_j.



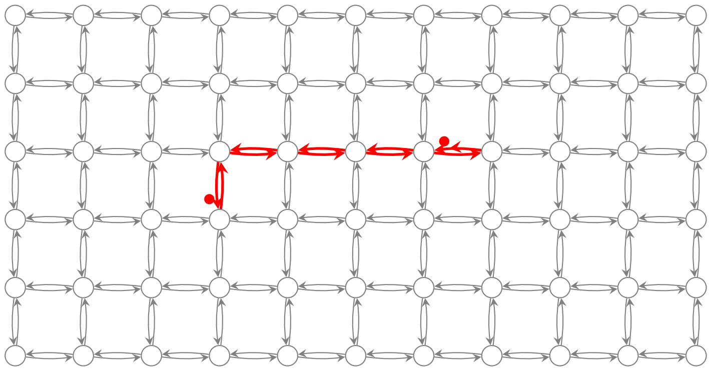
(a) Route before replanning



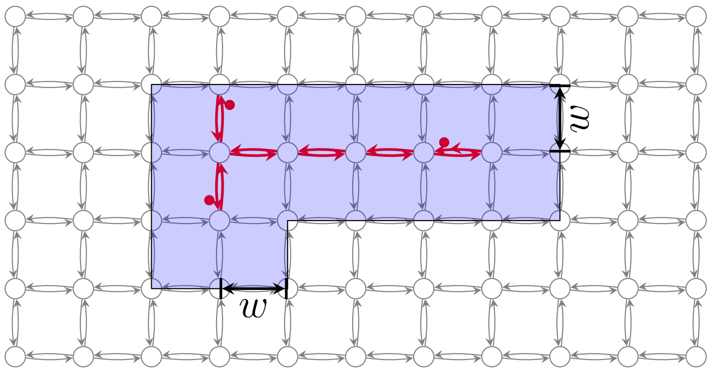
(b) Possible route after replanning and the area to find a new Stop

Figure F.2: Example for RectangleHullRouteExtension. Red Dots indicate the stopsToBeServed. Red arrows indicate the route. Double arrows indicate termini. Blue areas indicate the searchspaces within the algorithms try find a new stop.

RouteEnvelopeExtension This replanning-module spread out an envelope around the route within a certaindistance w. The resulting geometry for the route shown in Fig. F.3a is shown in Fig. F.3b. A stopfacility f – located within this area and currently unused – is drawn randomly. The route is divided into a set of subroutes j containing all stops between 2 stops to be served. The average distances d_j from f to all stops of a subroute are calculated for every subroute, as shown in equation F.1. Then f is inserted to the stops to be served list between the two stops of subroute j with the smallest average distance d_j.



(a) Route before replanning



(b) Route after replanning and the area to find a new Stop

Figure F.3: Example for RouteEnvelopeExtension. Red Dots indicate the stopsToBe-Served. Red arrows indicate the route. Double arrows indicate termini. Blue areas indicate the searchspaces within the algorithms try find a new stop.

G. German Summary

Die vorliegende Arbeit beschäftigt sich mit der Mikro-Simulation südafrikanischer Minibus Taxis (paratransit). Die verwendete Software (MATSim) [12], sowie ein auf ihr aufbauender evolutionärer Algorithmus [10, 11] werden detailiert vorgestellt. Der vorgestellte Algorithmus wird einigen Erweiterungen unterzogen. Diese werden ausführlich auf ihre Funktionalität getestet. Das vorgestellte und erweiterte Modell wird auf ein südafrikanisches Szenario – die Nelson Mandela Bay Municipality (NMBM), auch bekannt als Port Elizabeth – angewandt. Die Ergebnisse werden einer detailierten Analyse unterzogen und mit Daten aus dem lokalen Nahverkehrsplan [6] und einer Haushaltsbefragung [1] verglichen. Im Folgenden werden die Inhalte der Arbeit in stark verkürzter Form wiedergegeben.

Paratransit ist ein Begriff, der ursprünglich zur Beschreibung flexibler Nahverkehrsangebote, z. B. in den Vereinigten Staaten von Amerika (USA), verwendet wurde. Unter anderem beschreiben Roos und Alschuler[15], Cervero [16] und Orski [14] derartige Systeme, die zur Bedienung einer neuen Art von Nachfrage (resultierend aus der fortschreitenden Suburbanisierung) Anwendung finden. Im Gegensatz zu herkömmlichen Verkehrsmitteln des öffentlichen Nahverkehrs sind diese zeitlich und räumlich flexibel und können auch schwache Nachfragen ohne fianzielle Verluste bedienen.

Diese flexiblen Verkehrsmittel weisen eine grosse Ähnlichkeit zu (meist informellen) Verkehrsmitteln im afrikansichen, asiatischen oder südamerikanischen Raum auf [17].

Südafrikanische Minibus Taxis zeichnen sich weniger durch ihre räumliche , sondern mehr durch ihre zeitliche Flexibilität aus. Typischer Weise werden feste Routen bedient, auch wenn diese nicht öffentlich bekannt gegeben werden. Ortskundige wissen um den Verlauf der Routen, ortsfremden ist der Zugang zum System oft nur durch “erfragen” möglich. Abfahrtszeiten hingegen sind generell nicht bekannt. Vielmehr legen die Fahrer ihre Abfahrten spontan fest. Sie starten üblicher Weise erst, wenn ihr Fahrzeug voll ist. Diese informelle Struktur resultiert aus der Geschichte des Landes, so transportierten Autobesitzer zu Apartheitszeiten andere Menschen gegen eine kleine Gebühr, da die Qualität der herkömmlichen Nahverkehrsmittel nicht ausreichend war [2–4].

MATSim [12] ist eine Verkehrssimulation basierend auf dem Modell von Gawron [27]. Sie wird hauptsächlich an der Technischen Universität Berlin und der Eidgenössischen Technischen Hochschule Zürich entwickelt und basiert auf dem Multi-Agenten-Prinzip und der Warteschlangen-Theorie. Sowohl individueller Autoverkehr, als auch öffentlicher Nahverkehr [13, 23] können physisch simuliert werden. Ein, in diese Software integrierter, evolutionärer Algorithmus [10, 11] adaptiert das Prinzip des Paratransit. Sogenannte Kooperativen können ihr Nahverkehrsangebot im iterativen Prozess an die Nachfrage anpassen. Gute Lösungen überleben, schlechte werden durch neue ersetzt.

Modell-Erweiterungen waren notwendig, um die zuvor beschriebene Software für die Simulation eines realen Szenarios zu verwenden. Der gegenwärtigen Routensuchalgorithmus für Nutzer des öffentlichen Verkehrs sucht immer die schnellsten Alternative. Im südafrikansichen Kontext, wo intrinsische Motivation und finanzielle Lage eine bedeutendere Rolle spielen, ist das (zunächst) nicht erwünscht. Vielmehr soll der hohe detailierungsgrad der zu Grunde liegen synthetischen Bevölkerung genutzt werden, um ein realistisches Verkehrsbild zu erhalten. Das heisst, die Routensuche der Nutzer wird im Rahmen dieser Arbeit auf zuvor gewählten Verkehrsträger beschränkt. Über das Problem der Routenwahlentscheidung hinaus, zielt der zugrunde liegende evolutionäre Algorithmus auf die Identifikation von Schwachstellen im Nahverkehrssystem ab. Das heisst, das Angebot wird intensiviert, um Nachfrage von anderen Systemen abzuziehen, falls dies profitabel ist. Wie die durchgeführten Tests zeigen geschieht das auch. Die Reimplementierung ermöglicht es die Nachfrage auf einen zuvor festgelegten Verkehrsträger zu limitieren und so ein abwandern (wie beim eigentlichen Paratransit-Modul gewünscht) zu verhindern.

Szenario Das verwendete Szenario wurde durch die Forschungsgruppe von Johan W. Joubert an der Universität von Pretoria entwickelt und zur Verfügung gestellt. Es umfasst die sog. Nelson Mandela Bay Municipality (NMBM), auch bekannt als Port Elizabeth. Eine detailierte Beschreibung befindet sich im englischsprachigen Teil dieser Arbeit.

Analysis Die Ergebnisse für das zugrunde liegende Szenario sind zufriedenstellend. Im laufe des iterativen Prozess entwickelt sich ein stabiles Minibus-Netzwerk. Einige der simulierten Reisenden finden zum Ende der Simulation keinen Weg an ihr Ziel. Dies resultiert aus der Tatsache, dass jeder Agent 3 Pläne mit sich führt und sich das System ständig ändert. Das heisst, alte Pläne können invalide Routen enthalten.

Die Netzabdeckung entspricht der in einer Haushaltserhebung dargestellten [1]. Selbiges gilt für die resultierenden Fahrzeiten der Minibusnutzer. Ein Problem stellt die Anzhal der durchgeführten Fahrten mit dem Minibus dar, da die Werte für die synthetische Bevölkerung im Vergleich zum Nahverkehrsplan [6] stark abweichen. Auf Grund einer hohen Dunklziffer, die durch lokale Experten bestätigt wurde, wird dieses Manko zuächst hingenommen.

Zusammenfassung Der verwendete und erweiterte Algorithmus liefert stabile Ergebnisse. Die Validierung der Eingangsdaten gestaltet sich mitunter schwierig. Dies ist auf hohe Dunkelziffern und schwierige Erhebunsgbedingungen für den Minibus zurückzuführen. Trotzdessen ist das resultierende Netzwerk nah an der Realität, soweit valide Vergleichsdaten vorliegen.