A simulation based forecast algorithm for public transportation

Master of Science Thesis in the Programme Computer Science – Algorithms, Languages and Logic

ERIK LARKÖ
Jonathan Daugaard

Chalmers University of Technology
University of Gothenburg
Department of Computer Science and Engineering
Göteborg, Sweden, September 2014

A simulation based forecast algorithm for public transportation.

Erik Larkö
Jonathan Daugaard

Examiner: Dag Wedelin

Chalmers University of Technology
University of Gothenburg
Department of Computer Science and Engineering
SE-412 96 Göteborg
Sweden
Telephone + 46 (0)31-772 1000

Department of Computer Science and Engineering
Göteborg, Sweden September 2014

Abstract

This thesis describes a simulation based system for forecasting arrival and departure times in complex public transit systems. The variables included in the calculations are: time of day, vehicle schedule adherence, estimated passenger count on stop, interactions with other vehicles and vehicle position from both historical and contemporary data. The use of a simulation based algorithm simplifies the implementation of a complex model with a great number of different dependencies. Additionally, creating a forecast using historical data while looking at different data features to find similarities gives a more accurate forecast than just looking at a moving average using data from the current day.

Sammanfattning

Denna rapport beskriver ett simuleringsbaserat prognossystem för ankomst- och avgångstider i ett komplext kollektivtrafiksnätverk. Prognoser görs med hänsyn till tid på dygnet, fordonets schemaföljsamhet, prognostiserad passagerarräkning på hållplats, interaktion med andra fordon samt fordonets position. Systemet använder sig av historisk samt dagsaktuell data för att göra prognoserna. Med en simuleringsbaserad algoritm är det enkelt att implementera en mer komplex modell med flera olika beroenden. Att använda historisk data samtidigt som att se till olika egenskaper i data för hitta likheter med de aktuella förutsättningarna ger mer korrekta prognoser jämfört med en genomsnittsmetod som enbart använder sig av data från samma dag.

Contents

1

Introduction

Intelligent transportation systems (ITS) where advanced traveler information systems (ATIS) play a big role are becoming more and more common. One important part of ATIS is predicting when public transport vehicles will arrive or depart. Passengers may look for other, less environmentally-friendly, means of transportation if there are excessive waiting times at stops. Relevant and correct arrival and departure time forecasts increases passenger experience and may thus increase the use of public transport systems. There has been a lot of research done on the subject of predicting arrival times using e.g. regression models, machine learning techniques and simple historical averages. There has not been as much research done on analyzing what actually causes the arrival and departure times to change, nor on how the public transportation vehicles affect each other.

This thesis will therefore look at the possibility to model a public transportation network by answering the following questions:

- Are simulation based approaches useful when forecasting arrival and departure times in public transport networks?
- Can the simulation based approach give better results than a commercial system based on a moving average?
- Will taking inter-vehicle dependencies into account improve the accuracy of the forecasts?
- Is the schedule poor enough to warrant the effort of implementing a complex forecast system?

This thesis will only take into consideration public transportation systems that resemble the public transportation system in Gothenburg, Sweden and that have access to historical data and real-time vehicle location reports[1] (AVL). This thesis will only focus on making good forecasts for the entire public transport network in normal situations. Thus it will not focus on situations such as accidents or construction work.

1.1 Literature study

There is not an extensive literature in this area, but we here summarize the differerent approaches that we have found.

1.1.1 Historical models

A simple way to estimate how long it takes to travel between two points is to look at how long it usually takes to travel the distance. If it took two minutes for a bus do drive between stops A and B yesterday, it seems a good bet that it should take two minutes today as well. However, if there was construction work being done yesterday but not today then yesterday's time will probably be a bit off. The effects of the fluctuations in travel times can be mitigated if more samples where used and aggregated using e.g. the mean. A sample taken when there was a temporary obstacle will not affect the estimation if it is just one of many samples. It is important that the samples used reflect the situation which is being estimated as much as possible. Samples taken just a few minutes ago will e.g. likely have the same weather and traffic conditions as the situation being estimated making them good samples. Unfortunately there is bound to be few available samples that are that fresh, and as said before, many samples must be used to avoid the effects of temporary obstacles. Choosing which samples to aggregate can be hard, but Chen et al. (2011) investigate the periodicy of travel times and show that travel times...

- ... change greatly during different times of the day
- ... are different on weekdays and weekends
- ... are similar on the same day of week

These findings can be used to try to find the best possible grouping of previously recorded travel time to apply the aggregate function to.

As mentioned briefly earlier a common aggregation function used in historical models is the average, or rather the moving average. A moving average differs from a normal

[1]Real-time may be a bit of an overstatement depending on your definition of the phrase. A vehicle location report is sent to a server when the vehicle departs from a stop or at least every two minutes.

average only when it comes to handling new numbers. A normal average of three numbers could look like

$$\frac{1+2+3}{3} = 2 \tag{1.1}$$

If a new number is added, the denominator needs to be updated to reflect the increase in number of numbers

$$\frac{1+2+3+4}{4} = 2.5 \tag{1.2}$$

In a moving average the "oldest" number is removed from the summation in the nominator and the denominator is kept constant.

$$1+\frac{2+3+4}{3} = 3 \tag{1.3}$$

1.1.2 Time estimation methods

There are several methods which can be used to do time estimation. Many of these methods can be used for a lot of other things and are often referred to within the area of artificial intelligence and machine learning because of there ability to adjust and learn.

Artificial Neural Networks or ANNs The idea behind ANNs is to build a calculation structure which is similar to a brain, with neurons and nerves. This structure is simplified so that a neuron is nothing more than a calculation step and a nerve is just a relation taking the result from one neuron's calculation as input to another neuron's calculation. A nerve can also increase or decrease the output. Just like a nerve in the brain can be strong or weak. The input to a neuron can also be information from some external sensor, e.g. traffic density on a road, time during the day or time needed for last vehicle to drive a distance. The thought is then to train the network to do good estimates. This means to adjust the nerves and neurons capabilities. One way to use ANN technique to do forecasts for a public transportation system would be to create one ANN unit for each sub-time one would like to estimate, e.g. one unit will be independently used to estimate the driving time for a distance. This unit should have a lot of different inputs and output a single value, preferable the time needed. This could then be scaled to estimate the full public transportation system by creating one ANN unit for each activity which needs to be estimated. In contrast to Shalaby & Farhan (2004), Jeong & Rilett (2004) argue that ANNs can give good results, given enough training data. The variables influencing public transport travel times tend to depend on each other non-linearly, which is something that ANNs tend to be good at discovering and handling. They too differentiate dwell time from link travel time, but also look at schedule adherence. They argue that separating link travel and dwell time yields good results, but the inclusion of schedule adherence did not have much effect.

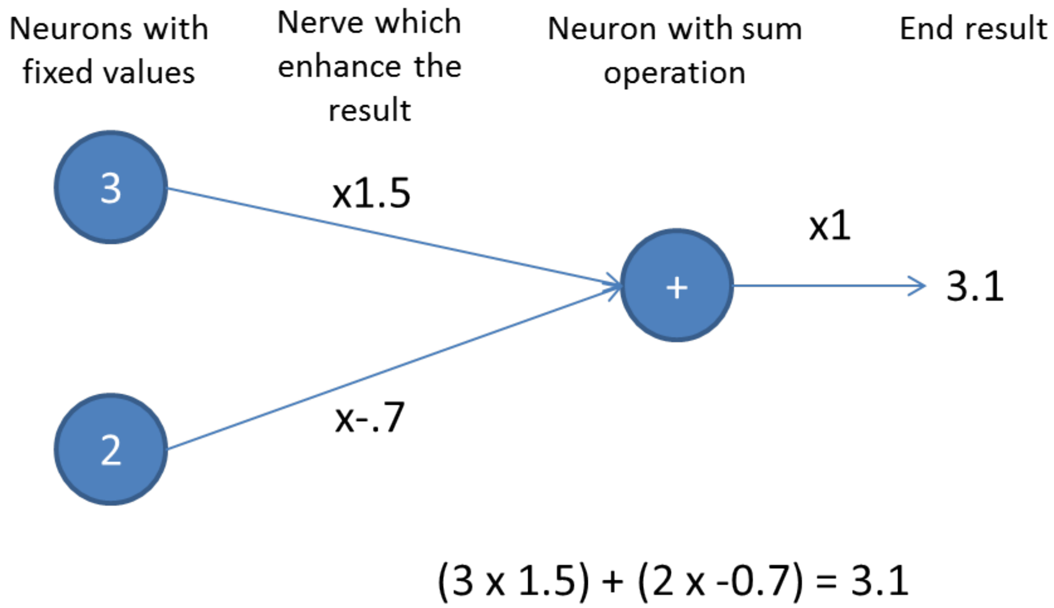


Figure 1.1: ANN with 3 neurons, 2 are fixed value and one are the summarize operation. Also nerves which enhance the result.

Kalman filter , also known as *linear quadratic estimation* or *LQE*, is an algorithm that uses a series of measurements observed over time, containing noise and other inaccuracies, and produces estimates of unknown variables that tend to be more precise than those based on a single measurement alone, according to Kalman filters are commonly used in radio receivers, e.g. mobile phones, TV etc., to reduce signal noise. The technique has several properties which can be useful for estimating arrival- and departure times in a public transportation system. An estimation method could be similar to the one described in the Artificial Neural Network section above. A divide and conquer method, where each sub part of the system can be estimated independent and all sub parts' estimates can be merged into one big estimate. The great thing with the Kalman filter approach is the built in technique for handling the unexpected, or noise. Noise in a public transportation system could be for example be more dense traffic, varying time needed to load/unload passengers, etc. An example of a Kalman filter used in public transportation estimation is Shalaby & Farhan (2004). They use VISSIM to simulate bus movements and use the simulation to make predictions, focusing mainly on the morning peak. With the simulated data they show that Kalman filters are good at responding to changes in conditions such as a temporary increase in the number of passengers or a lane being closed. They estimate dwell times and link travel times separately and argue that the algorithm reacts better to anomalies with these times separate than when they are grouped into one variable. Historical models and artificial neural networks (ANNs) do not give as good results as the Kalman filters under their circumstances.

Linear regression models is a means to understand how a value changes in regards to some other value, e.g. how the travel time between two stops changes with the time of day. More specifically, linear regression tries to find the line that minimizes some function on the data. A common approach is the *Least Squares* which minimizes the square of the distance of each point to the line. Figure 1.2 shows an example. Linear regression models is a common prediction model but it has a few drawbacks. E.g. the linearity of the model makes it difficult to model the non-linearity of some estimation problems, and the model does not easily adapt to change.

Yu et al. (2009) implement a very interesting enhanced regression model. It's based on linear regression which doesn't respond well to changes. To combat this they created a more adaptable function with the linear regression model and its error as input with very good results. However, they only estimate travel times for one hour in the morning for one line in one direction. These simplifications make their network much more homogeneous and easier to optimize. They say that a vehicle's speed is what really matters, and that this speed is what should be estimated - not travel time. This might seem reasonable but it in essence boils down to the same thing as the length between two stops is constant.

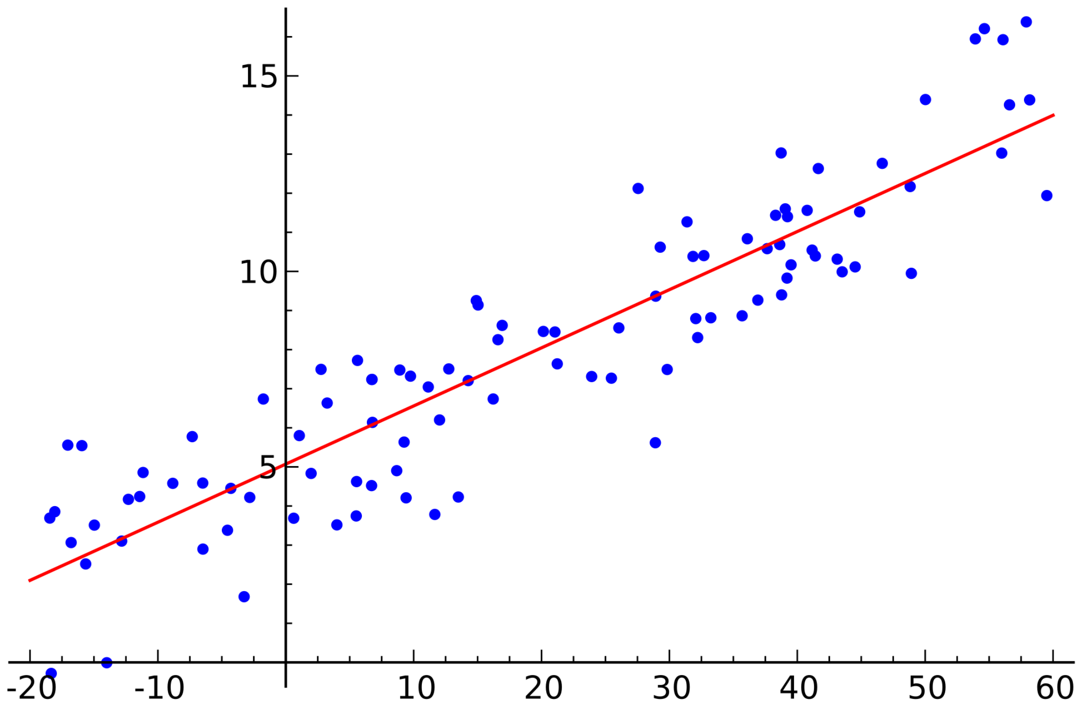


Figure 1.2: Linear regression finds the line which best "fits" the data.

1.1.3 Other reports on subject without going into too much details on time estimation method

Chen et al. (2011) show that it can be better to include many stops and links in one estimation than to calculate each on its own and then sum the estimates up. That is; given a path of four stops;

$$A \rightarrow B \rightarrow C \rightarrow D$$

it could be better to estimate the travel time of the whole path

$$\text{Estimated travel time} = TT(A,D) \tag{1.4}$$

than to sum each segment up as

$$\text{Estimated travel time} = TT(A,B) + TT(B,C) + TT(C,D) \tag{1.5}$$

where TT(X,Y) is a function aggregating previously recording travel times from stop X to stop Y.

Their results may not be as accurate as those based on more refined models such as ANNs, but their findings are still very useful.

Yu et al. (2009)

Jeong & Rilett (2004)

1.1.4 Earlier written text - Should be moved to appropriate place

As mentioned earlier, there has been quite a lot of research using a variety of methods done in this field. However; none of the read reports have tried their algorithms on complex public transportation networks, i.e. networks with many vehicles interacting with each other. Nor have any reports tried to reason about what actually causes the changes in travel times, instead they are based on models where events that alter the travel times disappear into the numbers.

Chen et al. (2011) use an historical model, i.e. a model where a simple function such as the average is applied to previously recorded travel times. They investigate the periodicy of travel times and show that travel times...

- ... change greatly during different times of the day
- ... are different on weekdays and weekends
- ... are similar on the same day of week

These finding are then used to try to find the best possible grouping of previously recorded travel time to apply their aggregate functions to. They also show that it can be better to include many stops and links in one estimation than to calculate each on its own and then sum the estimates up. That is; given a path of four stops;

$$A \rightarrow B \rightarrow C \rightarrow D$$

it could be better to estimate the travel time of the whole path

$$\text{Estimated travel time} = TT(A{,}D) \tag{1.6}$$

than to sum each segment up as

$$\text{Estimated travel time} = TT(A{,}B) + TT(B{,}C) + TT(C{,}D) \tag{1.7}$$

where TT(X,Y) is a function aggregating previously recording travel times from stop X to stop Y.

Their results may not be as accurate as those based on more refined models such as ANNs, but their findings are still very useful.

Regarding regression models, Yu et al. (2009) implement a very interesting enhanced version of this model. It's based on a linear regression model which doesn't respond well to changes. To combat this they created a more adaptable function with the linear regression model and its error as input with very good results. However, they only estimate travel times for one hour in the morning for one line in one direction. These simplifications make their network much more homogeneous and easier to optimize. They say that a vehicle's speed is what really matters, and that this speed is what should be estimated - not travel time. This might seem reasonable but it in essence boils down to the same thing as the length between two stops is constant.

2

Main system components and development process

This project is about creating a good method that can be implemented in software which can do forecasts on arrival and departure times in public transportation systems. To be able to evaluate if the method is good, a method for benchmarking the produced forecasts will also be implemented. To allow for the possibility of comparing different versions of the implemented forecast system as well as comparing it to already implemented forecast methods, the forecasting and the benchmarking system must be independent of each other.

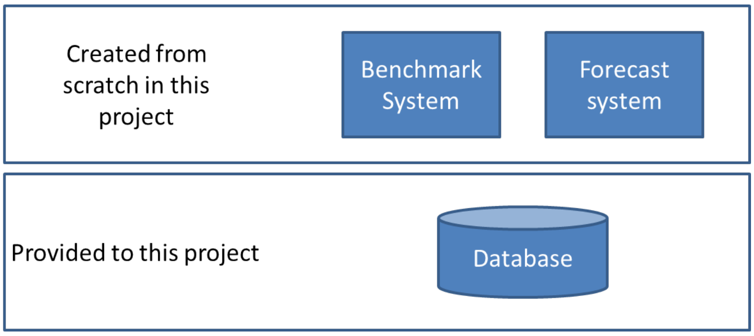


Figure 2.1: Systems

One of the preconditions of this work is the availability of data that describe the traffic in the transportation system, both planned and actual, and the produced forecasts from the industrial forecast system that is currently in use. This information will be retrieved from an existing database. Phase one will produce the benchmarking system and also

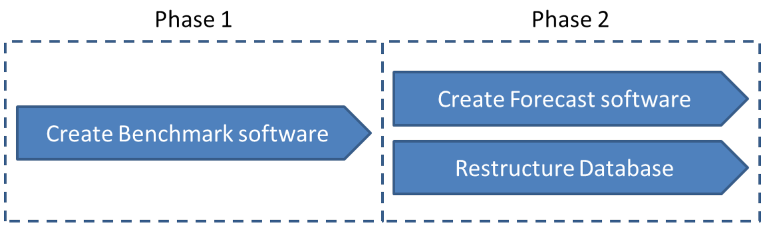


Figure 2.2: Work process for creating software

give a good basis for creating the forecast system in phase two because of the database and domain experience gained.

2.1 Benchmarking system

The benchmarking system is built by three components, (1) a model describing how to evaluate the quality of a single forecast, (2) an algorithm that will use this model to assess the quality of all forecasts generated and (3) aggregation algorithms for some measurements of quality, e.g. the Mean Absolute Error. These three components can be created separately and should be able to be modified during the project to gain a better measurement of quality or performance. In Figure 2.3 *The benchmark method* is

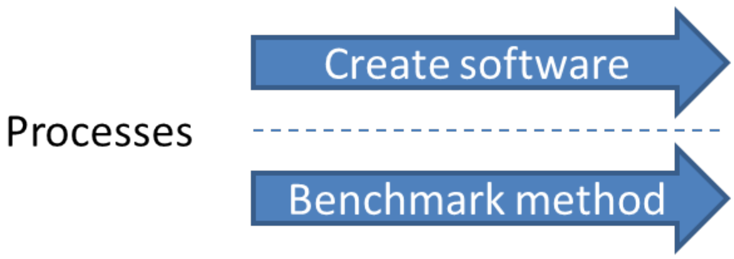


Figure 2.3: There are two separate work processes for the benchmarking system.

both the model for calculating the quality of a single forecast, and the model for how to aggregate all measurements of quality. This separated design will allow the system to be easily redesigned.

2.2 Forecasting system

The process of creating a forecasting system will be iterative, beginning with a pre-study of the data that describes the public transport system - studying the timetables, maps and historical driving times etc. The next step is to create a public transportation model and a method to simulate traffic in it. The iterative process begin by creating a very

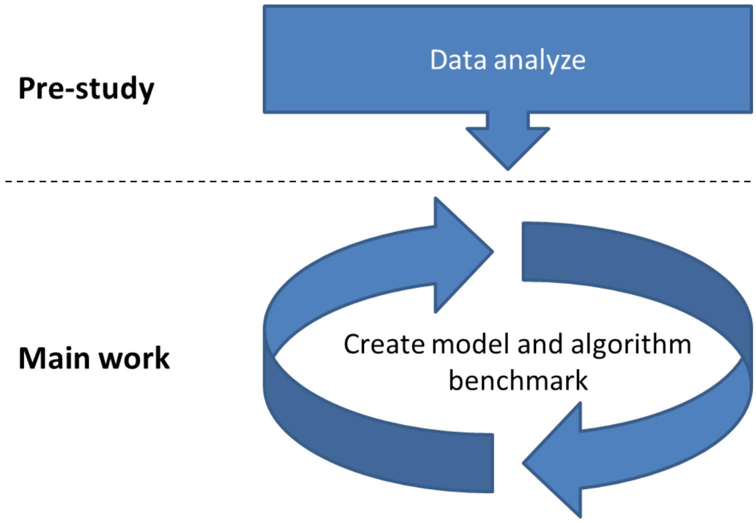


Figure 2.4: There are two phases when creating the forecasting system.

simple constant model, and an algorithm that can simulate the traffic using this model. Then more advanced models and algorithms will follow. The process of creating new algorithms will be driven by the model's new behaviors. The process of creating models will be driven by the quality of the forecast system and the conclusions drawn from analyzes of data that describe the public transport system. This process will continue until no further improvements can be done, or more likely that the project has reached its deadline.

3

Problem description

A public transportation system can be seen as a graph where the nodes are stops and the edges are physical connections between stops. Vehicles such as buses, trams and boats travel in this system on paths based on which line the vehicle is currently operating as. Figure 3.1 shows an example of a very simple public transportation system with five stops and three lines.

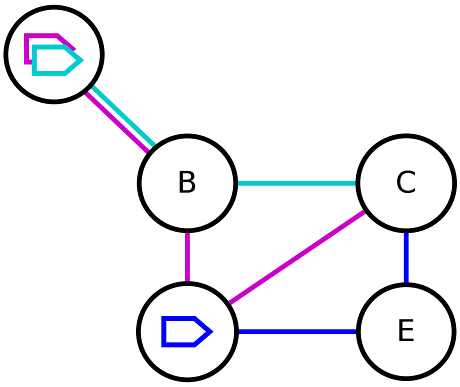


Figure 3.1: A simple public transportation system with five stops (A, B, C, D, E) and three lines (Cyan, Magenta and Blue).

The lines' paths through the system may differ on different days or even on different times of the day. To encapsulate this we introduce the concepts of *journeys* and *departures* and say that a vehicle's path through the system is determined by its journey. A journey is a list of *Departures* which in turn are triples of a line, a stop and the time when the vehicle is to leave (or depart from) the stop. A departure could be e.g.

"Tram number 6, scheduled to depart from Brunnsparken Site A at 14:05 on November

11th 2012”

Also; associated with each departure is a list of forecasts, or estimates of when a vehicle will actually depart from the stop. E.g.

”Tram number 6, scheduled to depart from Brunnsparken Site A at 14:05 on November 11th 2012 *will probably depart at 14:06, 14:07 or 14:10*”

A journey can have any number of departures; the only limitation is if it feasible for the driver to use it as driving plan. The times of the departures in the journey must be ordered ascending and the stops must be of types that the vehicle can use, e.g. there is rarely a vehicle which can use both a street stop and ferry stations. In Figure 3.2 there is an example of two journeys. One physical vehicle has several journeys during a day.

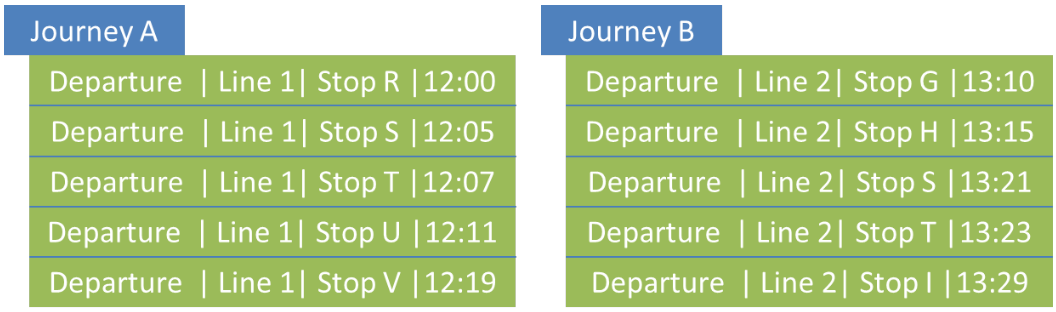


Figure 3.2: Two different journeys with departures.

The only limitation in having several journeys is similar to the limitation of departures in a single journey; feasible times and types of stops, hence in this report will vehicles only be considered to have one aggregated journey.

3.1 Stops

Stops are places where the vehicle will stop for letting passengers get on and off. A stop is a predefined and announced location; often with a sign or some other kind of marking so the passengers can easily find them. Figure 3.3 is a typical bus stop in Gothenburg. Stops can and are often used several times during a day and can even be used by vehicles that drive different lines. The time needed for loading and unloading passengers is called *loading time*. If a vehicle would stop on any other place than where the timetable states that the vehicle will stop, the time standing still will not be a part of the *loading time*, e.g. stopping for a traffic light or for pedestrian crossing.

Figure 3.3: A typical bus stop.

3.1.1 Queues into stops

On many stops, there can be only one vehicle at any given time. This is of practical reasons, e.g that the road is single laned, or that there is only enough room for one ferry to load passengers at a time. This means that when several vehicles arrive at the same stop at the same time there will be a queue to get into the stop. This also means that any two vehicles must leave the stop in the same order as they arrived. On some stops is it possible for the vehicle to skip stopping at the stop or do partly simultaneous boarding. This depends strictly on the possibility for the driver to do this in a safe way. It is quite rare that a vehicle skips a stop. This happens mostly when there are two vehicles that are on the same line and the first one is late and the second one catches up with the first one.

3.1.2 Timing points

Certain stops are so called timing points. These stop are like normal stops, except that their schedules includes a waiting time. This means that a late vehicle can make up time on these stops by skipping parts of this scheduled waiting time. There is however one caveat; imagine a vehicle waiting at a timing point when a late vehicle approaches the stop. The waiting vehicle does not want to force the late vehicle to wait for it, making it even tardier, and drives away before its scheduled departure time. Table 3.1 shows a concrete example.

Table 3.1: Vehicle A is waiting at a timing point. Vehicle B is late and arrives before A is scheduled to depart. Vehicle A will then depart at 13:57 instead of 14:00.

	Actual arrival time	Scheduled departure time	Actual departure time
Vehicle A	13:50	14:00	13:57
Vehicle B	13:57	13:52	13:57 + dwell time

3.2 Links

A link is the physical path between two stops. If e.g. the vehicle is a bus, the link can be a combination of different streets in a particular order. If the vehicle is a tram the link can be a section of rail. The time it takes to travel a link is known as the *link travel time*. Links may alsp have a maximum capacity, there are many cases where public transportation vehicles are unable to overtake each other. Trams rarely have separate rails and buses often have special single file lanes. This means that a vehicle can't leave a link before the vehicle in front of it does.

Table 3.2: Two vehicles on a single lane link. A slow driving vehicle enters the link as number one. A super-fast vehicle enters the link as number two.

	Start driving time	Ordinary driving time on link	Time when leaving the link
Slow driving Vehicle	13:00	00:30	13:30
Super-fast vehicle	13:01	00:15	13:30 + short delay

Table 3.2 is an example of this. Even if the *super fast vehicle* could pass the link by 13:16 it will have to wait until 13:30 before leaving the link because there is no possibility to overtake the *slow driving vehicle*.

3.3 Timetable

The timetable is a predefined plan on which stops vehicles should stop at and at which times. The timetable can be printed and used by passengers or drivers of the vehicle. Table 3.3 is an example of a timetable used by the driver of a specific vehicle or the passengers that intend to travel with it. Table 3.4 is an example of a timetable that could be posted on a stop to inform the passengers on when the different vehicles will arrive and depart from the stop. Note that the difference between arriving time and departure time is not necessarily the *loading time*.

Table 3.3: A example of a timetable for a vehicle

Order	Stop	Arrive at	Departure from
1	A	12:00	12:01
2	B	12:20	12:22
3	C	12:34	12:34

Table 3.4: A example of a timetable for a stop

Order	Vehicle	Arrive at	Departure from
1	A	12:00	12:01
2	B	12:20	12:22
3	C	12:34	12:34

4

Time estimation model

The model of the transportation system will be divided into two parts in this report. The reason for this is to simplify the creation of its properties as well as arguing for them. The first part is the physical representation. This part describes all stops, links, vehicles and their paths in the network, similar to a snapshot of a city from a satellite perspective. The picture will only say what is connected to what. The second part is the time representation. This part describes how much time is needed for a vehicle to pass a physical position, e.g. a stop or a link. If the first part was satellite photo then would the second part be a *time needed* table of how much time is required for a vehicle to pass them. Please note that both parts can contain more variables then a satellite photo or a simple *time needed* table.

Figure 4.1: The model can be divided into two independent parts. Each part only describe one dimension of a public transportation system. One for how to travel between places and one for how much time is needed for each position.

Physical representation of a basic model of the transportation system The physical representation in the model will describe the physical elements of the real world which have been considered in the model, e.g. streets, stops and their connections. The physical representation can be seen as a graph where the nodes are the places where a vehicle transition from be on one physical place to another, e.g. going from driving on a street to stop at a stop. The edges will be the physical places. The full graph can then look like Figure 4.2. Constraints can be added to each edge. Constraints such as how many vehicle which can be on the edge at the same time, a bus stop can only be used by a fixed number of buses at the same time, or entry and exit order, in a section of rail is it not possible for one tram to overtake another one etc.

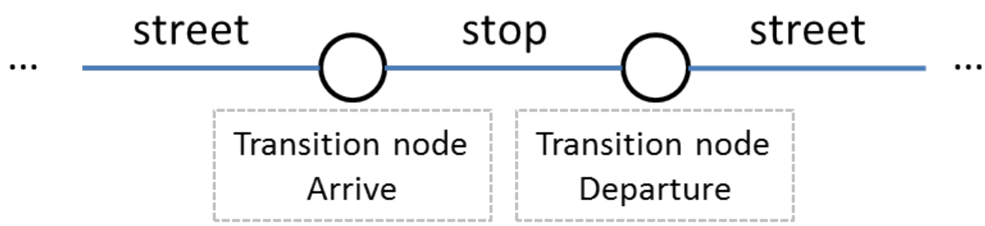


Figure 4.2: The edges are physical elements of the real world while nodes are transitions points. Several constraints can be placed on each edge.

The physical representation can also be refined to have even more details and constraints on it. An edge which represents a long street with several crossings can be divided into several edges and nodes where each edge represents a section of the street or a crossing. Some section of street might be single lane while other parts are not. Timing on traffics light is described through the time representation which you can read more about in next section.

Time representation of a basic model of the transportation system Each position in the physical representation can have a time connected to it, if there is no time then it should be equal to the time 00:00:00. The times can be estimated in any way and depend on anything. This will allow times to be constant, deterministically calculated, stochastically calculated or depend on other times in the network. A stoplight for leaving a stop that depending on the driver signaling will always need a constant time before it changes to *green*. The *travel time* on a link can be estimated by the average of historical times. The waiting time at a stoplight depends on the when it last was green, which could be unknown and can thus be anything between 0 seconds and 1 min 15 seconds. Hence a good approach might be to randomly pick a time between 0 seconds and 75 seconds as *travel time* for this position. If the position has a capacity constraint then the time will include the waiting time which depends on some other vehicle.

Historical recorded pass times will be used as a basis for link travel times and dwell times. These times are calculated based on time stamped location reports that vehicles send whenever they start loading passengers at a stop (known as *Arrival reports*) and whenever they leave a stop (known as *Departure reports*). These reports are known as *Automatic Vehicle Location Reports*, or AVL reports.

Link travel times are the times between a *departure report* and the consecutive *arrival report*,

Dwell times are the times between an *arrival report* and the consecutive *departure report*.

Aggregated view of physical- and time representation of a basic model of the transportation system The model can be used for simulations when the two parts, the time- and physical representations, are connected. The connection is made from the edges in the physical representation to the time needed to pass the edge in the time representation. Thanks to this is it easy to estimate the time needed to travel a path, i.e. a set of edges. The travel time for an edge can of course depend on which time during the day and which day it is. So when estimating the time needed one must start by estimating when the vehicle will start driving (or load passengers) on each edge. This is simply done by selecting the start time for the first edge and then iterating each edge in the correct order according to the path and summing up the time. After adding the last time one will also know the total time needed to drive the path. Figure 4.3 shows how this can be done.

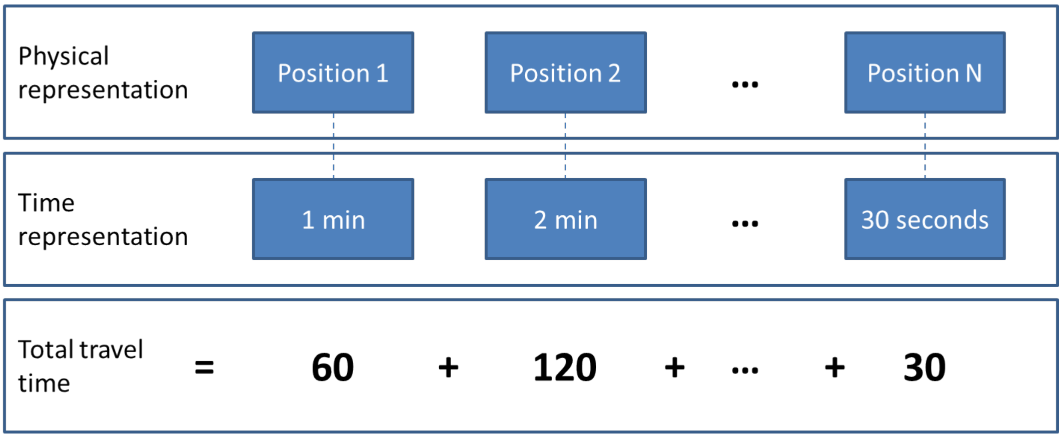


Figure 4.3: The total travel time for driving from *Position 1* to *Position N*.

Note that there is no problem if the physical representation contains a position which is not present in the time representation, because the time for that position will only be estimated as 00:00:00. Likewise if the time representation can describe more positions than the physical representation contains. This can happen because the aggregation model only uses the positions described by the physical representation. This is one of the benefits of dividing the model into two separate independent parts.

4.1 Constant time model

A constant model for describing a public transportation system can only work if the network can be forecasted by another system or if the system has the possibility to follow a predefined plan. This plan would be the time table. In both cases, there are no actual benefits to the forecast system. The benefit of a constant model in this project is only the possibility to focus on building the physical representation. The time representation will return same time for every position.

Physical representation of a basic model of the transportation system The goal of this model is too have a good and accurate physical representation of the public transportation network. The representation must contain all stops, links and pseudo locations. Constraints on positions will be excluded in order to simplify the model. The pseudo locations will be bus garages, tram depots or the berths ferries are moored to when they are not in traffic.

Time representation of a basic model of the transportation system This model will not consider which day of week, time of day or which month it is when it estimates the time needed to pass a position. It will always give the same time for the same position.
Note that with a very small change, this model can be an actual time table model. Instead of just always returning the same time for a position no matter what the conditions are, it can be refined to give different times depending on date and time on day according to the time table. The refinement would simply be to implement the time table matrix, which describes how much time is needed to pass a position for a given day and a given time on that day. Please note that a time table is often created weeks or even months in advance and can only consider a subset of all the conditions that occur during each day.

4.2 Average time model

This model is an extension to the model in section 4.1. The main differences between the models are the *time representation*. This model is also an attempt to mimic the forecast system used in Gothenburg today. The model in the forecast system used today will be referred to as the *industry model*. The industry model is an average model which also has some filtering and optimization features.

Physical representation of a basic model of the transportation system Physical representation will be same as in the model in section 4.1 with the exception that some constraints been added. The added constraint is that a vehicle should never leave the first stop in a journey before its departure time for this stop according to the time table. This constraint holds for all journeys. Note that if the vehicle is late to the first journey, this constraint will not have any affect.

Time representation of a basic model of the transportation system The time for a position depends only on the last N known recorded times for that position. The time will be the average of the sum of all times, see Equation 4.1.

$$Average\ time(x) = \frac{1}{N}\sum_{i=0}^{N} recorded\ time_i \tag{4.1}$$

This means that the model will give the same travel time for a link forecasted 10 minutes in advance as it would four hours in advance. The reason for this is that the amount of recorded information has not changed.

4.3 Proposed time model

The extended model is the main model proposed by this thesis. This model is built with the intention to encapsulate more aspects of the real world. Aspects that both have impact on constraints in the physical representation and the time needed to pass a position. The aim is to model the aspects that have the most significant effect on the quality of the forecasts. These aspects are decided through analysis of recorded times and forecast quality, both measure the quality of the timetable and the industry produced forecast.

4.3.1 Physical representation of a basic model of the transportation system

The physical model has been extended with several behaviors that can be seen in the traffic. The behaviors can be connected to different activities that the vehicle performs during its drive, e.g. driving between stops or the actual boarding of passengers. In this project, every sub task that the vehicle does while boarding passengers will be part of the stop activity. These sub tasks can be e.g. waiting to access the stop because of queues, or adjusting to the timetable before leaving the stop. This splitting is necessary because of how we use the recorded times.

Overtake limitations

Many of the roads that buses use are single lane. It is very rare that trams have the possibility to overtake each other. Because of this, the model has been constrained for all physical locations which are streets or sections of rail to not allow vehicles to overtake each other. This is simply done by applying a first-in first-out approach for each physical position, street or section of rail. No constraint has been added for how many vehicles can be on the same street or section of rail even if there is such constraints in the real world. The added constraint will also make all streets and sections of rail have similar features as an ordinary queue with the exception that each vehicle still need to pass it before it can leave. Please see Table 3.2 for an example of this.

Queues into stops

In the job description for the drivers in the studied public transportation network is it stated that the driver should stop at the most forward position at the stop to load and unload passengers. This does not stop the driver from unloading passengers at a position that is a bit further back on the stop but she should always drive to the front and make another stop. Also, there are a lot of stops that only have the length of supporting one vehicle at the time. This often creates queues into the stops during peak hours. The model has been constrained to only support one vehicle at a time. This means that a vehicle cannot overtake another vehicle at the stop because the second vehicle is not allowed to enter the stop before the first one has left. This also creates a queue into the stop. This property is part of the model, even though it isn't always true in the real world. Most of these exception stops can be ignored due to the fact that there are rarely two vehicles at the same time at these stops, but there is room for improvement here, see section 7.5 for more details.

The time a vehicle spends in the queue into the stop is estimated by looking at when the previous vehicle arrived at the stop. The queue time is the difference between this previous vehicle's departure time and the current vehicle's arrival time and can never be negative. See table Table 4.1, Table 4.2 and Table 4.3 for an example.

4.3.2 Time representation

The time representation gets a lot more complex with the physical constraints *Queues into stops* and *Overtake*. These constraints depend on other vehicles and thus on the

Table 4.1: This tables shows how vehicle B waits on vehicle A.

Vehicle	Arrival time	Departure time	Queue time
A	12:00	12:02	
B	12:01		1

Table 4.2: If there is no vehicle on the stop there will be no queue time into the stop.

Vehicle	Arrival time	Departure time	Queue time
C	13:00	13:02	
D	13:02		0

Table 4.3: This shows how the queue time can build up when more vehicles arrive at stop than departs.

Vehicle	Arrival time	Departure time	Queue time
E	14:00	14:04	
F	14:02	14:06	2
G	14:03	14:08	3

other vehicles' forecasts. This introduces a new level of time calculations, looking at other vehicles expected behavior. One difficulty with this is the complex dependencies that occur between vehicles.

Scheduled waiting time

Scheduled waiting time arises when a vehicle is at a timing point. If the stop is not a timing point stop or the vehicle is behind schedule the scheduled waiting time will be zero. Otherwise, the schedule waiting time is estimated by picking the smaller of two times. The first time is the difference between the time when the vehicle is done loading passengers and its departure time according to timetable. The second time is the difference between the time when the vehicle is done loading passengers and the arrival time of the vehicle just behind it. Thus, scheduled waiting times depend on other forecasts and on the time table. See Table 3.1 for example.

Loading time

The loading time in the real world depends on the number of passengers, their speed and the driver's patience. None of these parameters are available or easy to measure as the network is too big. Instead, the model uses other parameters to estimate the time needed to load passengers. The estimate is calculated by selecting a set of historical times that are as similar as possible to the situation to be forecast, and do an average of them.

Stop and Line The number of passengers greatly varies from stop to stop; just think about the inner city, the country side and the big connections points. Some stops are used by many different lines, lines which vary in popularity. This aspect is used, along with other aspects, to describe the number of passengers getting on and of the vehicle.

Route frequency The number of passengers getting on and off a vehicle at a stop greatly affects its dwell time. We don't have access to passenger counters, so we cannot know these numbers exactly. But we know that the number of passengers getting on a vehicle partly depends on how long ago a vehicle travelling a similar route picked up passengers at the stop. To clarify, passengers don't care which vehicle they get on as long as it's stopping where (s)he wants to get off - passengers get on the first suitable vehicle to arrive at the stop. This leads to two scenarios that may change the dwell time:

1. If there are fewer vehicles with similar routes passing the stop than planned, there will be a buildup of passengers leading to a longer loading time.
2. If there are more vehicles with similar routes passing the stop than planned, then the passengers will spread out over the vehicles - resulting in shorter loading times.

Schedule adherence The driver of the vehicle will try to follow the time table, thus if a vehicle arrives late to a stop, the driver may try to decrease the dwell time by e.g. not waiting for passengers running towards the vehicle. If the vehicle arrives at a stop too early, the driver may do the opposite to try to increase the dwell time.

Time of day The time of day has a great impact on the number of passengers on a stop, e.g. people commute to and from work and school on mornings and evenings. But the speed of the passengers also varies depending on the time of day. Just think about the night traffic on Friday and Saturday and compare it with the morning traffic on weekdays. By using reference days to capture cyclic behavior the model takes time of day as an aspect. For this assumption to hold, the reference days must be truly similar to the day the forecast is made.

Black box There are many aspects that we have not explicitly taken into consideration, e.g. temporary construction work and the weather. The effects of these aspects are assumed to be present in previously recorded times, it is just a matter of finding in which ones. We assume that these black box aspects affect the location on which they occur in a similar way every time they occur. Chen et al. (2011) show that aspects are likely to have similar effects on the same time of day and type of day. These assumptions are used when selecting which previously recorded times the estimated loading time is to be similar to.

Method for estimating the loading time The goal is to capture all the previously described aspect which can have an impact on the volatility in the loading time. The idea is to use historical samples which are as similar as possible to these aspects and then do an average. It is expected is that these historical samples have a low standard deviation and thus the estimate will be accurate. This also means that an estimate will never be lower than the lowest historical time used nor longer than the longest historical time used. The method for selecting the historical samples are rule based and samples can be categorized by samples from same day, similarity on time of day or other similarities.

The first set of samples is the last known loading time needed for vehicles of same stop of same line. The second set, similarity on time of day, are samples picked from the reference days. 10 sampled loading times from same stop and vehicle of same line which started loading immediately before or same time as the time to do forecast for. 10 sample loading time from same stop and vehicle of same line which started loading immediately after the time to do forecast for. The last set, other similarities, are also picked from the reference days, but these samples differ more in time on day but are picked by their similarity on the other aspects. This is done by adding up the weighted difference for each aspect and then picks the samples with lowest difference. The last set is made from 3 samples from each reference day. Note that this is a subject for improvements and an area for future work.

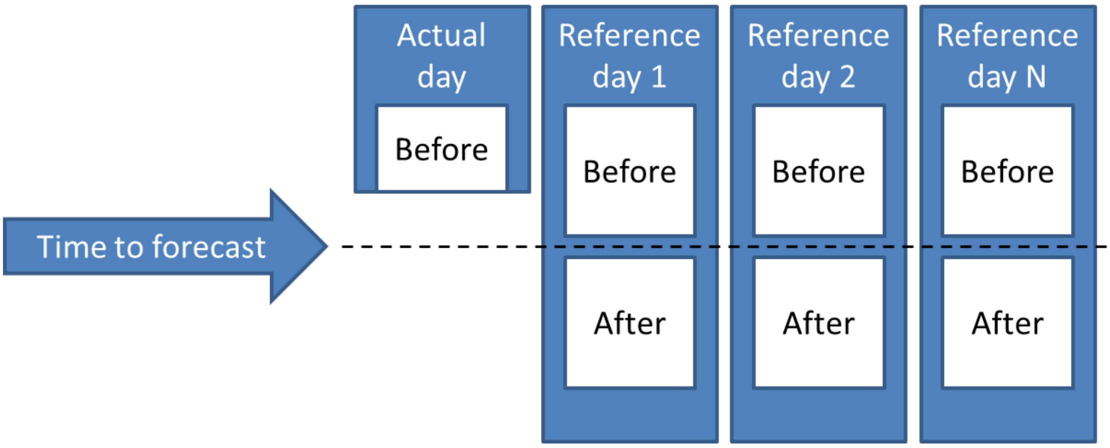


Figure 4.4: Illustrates which historical samples are used when estimating the loading time. Historical samples in the white boxes are used and the most like current conditions from each blue box is also used for the average.

Driving time

Naturally, links differ widely from one another. They, for example, have different lengths, red lights, junctions and speed limits. However, there is only one of all these link properties for which we have data; link length. So, the model uses previously recorded link travel times to estimate the combined effects of all circumstances on the link. It is however, possible to take the vehicle's schedule adherence into account. Thus, the driving time encapsulates two aspects;

Schedule adherence Drivers try to adjust their speed so as to be on schedule. A late vehicle will increase its speed slightly while the opposite is true if the vehicle is running ahead of schedule.

Black box As with the black box aspect in the loading time calculation, link travel times previously recorded on the same location at similar times as the link travel time to be estimated is used as an assumed catch-all for aspects which we have not explicitly considered.

Driving time calculation Driving times are estimated to be similar to previously recorded link travel times recorded on days similar to the DTF around the same time as the TTF, and times recorded on the DTF. The calculation is an inverse distance weighting function where the distance is the difference in schedule adherence, see Equation 4.2; if the vehicle whose driving time is being forecasted is three minutes late, previously recorded link travel times recorded when vehicles was around three minutes late will have a greater impact on the resulting driving time than those where the vehicles were not.

Given a set of previously recorded link travel times $(x, y)_1, (x, y)_2, ..., (x, y)_n$ where x is how many seconds off schedule the vehicle was and y is the link travel time, then

$$Driving\ time(x) = \frac{\sum_{i=0}^{N} w_i(x, x_i) * y_i}{t} \tag{4.2}$$

where

$$w(x, x_i) = \frac{1}{distance(x, x_i) * p} \tag{4.3}$$

$$distance(x, x_i) = |x - x_i| \tag{4.4}$$

$$t = \sum_{i=0}^{N} w(x_i) \tag{4.5}$$

The p in Equation 4.3 is a scaling parameter which is usually set to a value around the number of dimensions in the distance, one in this case.

Example Predicting the driving time between stops A and B on 2012-11-12 13:00. All previously recorded link travel times recorded on the link AB around 13:00 on the specified reference days and earlier on 2012-11-12 are plotted in a graph where the Y-axis is the recorded link travel time and the X-axis is how many seconds off schedule the vehicle was, as can be seen in Figure 4.5

From the data represented in Figure 4.5 we calculate the IDW-function and arrive at Figure 4.6. The driving time is then estimated by simply applying the IDW-function on the current forecast's schedule adherence.

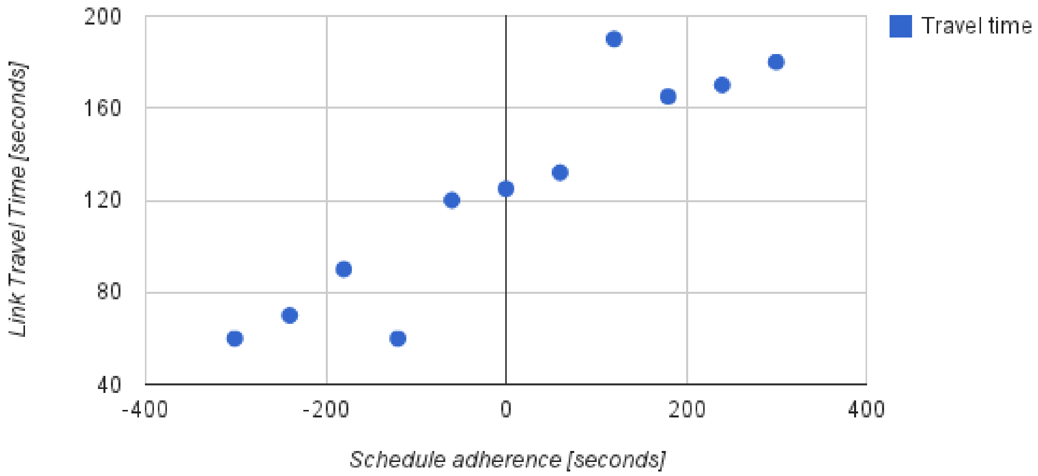


Figure 4.5: A scatter plot of previously recorded link travel times. The link travel time is plotted against the Y-axis, the number of seconds late or early the vehicle was when the Y-value was recorded is plotted against the X-axis. A negative X-value means that the vehicle arrived ahead of schedule, a positive means it arrived late.

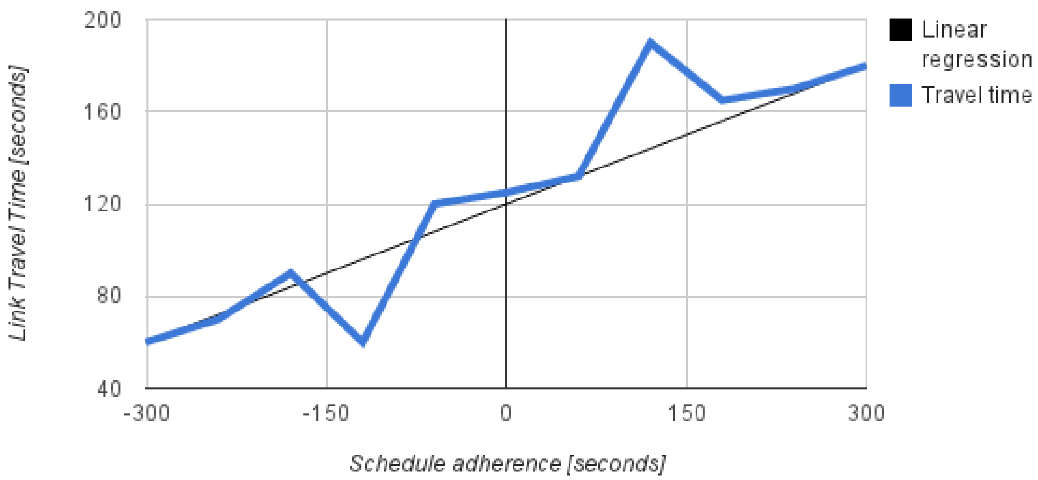


Figure 4.6: A plot of the IDW-function from example data. The simple linear regression is shown as a reference indicating that the IDW-function is more adaptable than regression.

4.3.3 Aggregated view of physical- and time representation

This model introduces a new level of complexity for time estimation. The estimate must consider and comply with other forecasts. This is done without adding any other information about the real world, e.g. traffic lights, intersections or speed bumps. Instead, new constraints are added to the already modeled elements and how the vehicles should interact with these new constraints. Constraints added to the physical representation is not in the form of *possible* or *impossible* to pass the object, but it will be *possible* but only when Vehicle X has left the stop.
This model more clearly separates between *Dwell time* and *Link travel time*. The reason for the separation is mainly because the difference in aspects used in the picking historical samples for in the time estimation method.

Dwell time

The dwell time calculation is split to three different steps. Each step resembles the actual actions that a vehicle performs when stopping at a stop. This split allows for a model that resembles the reality in a more accurate way and complex dwell time estimation. The different actions in order that a vehicle will perform at a stop are: 1. stand in queue into stop, 2. load passengers and 3. wait for the schedule , as visualized in Figure 4.7. Each action takes time, and the manner in which this time is estimated differ between types of action. The total estimated dwell time on a stop is a summation of all estimated times for each action.

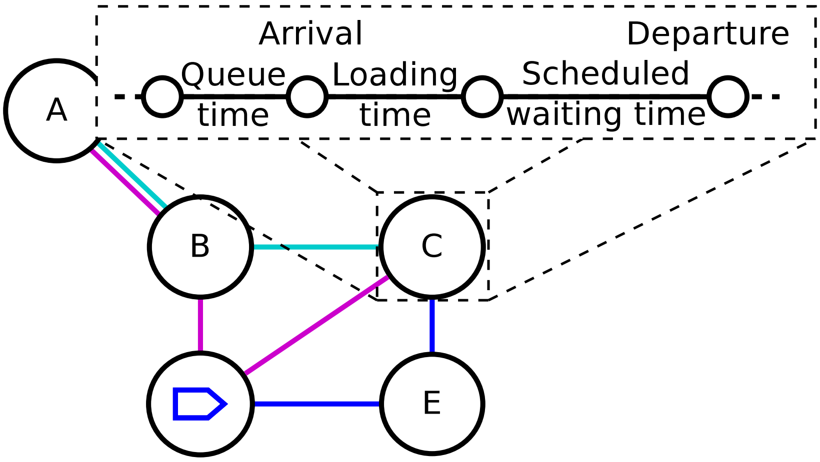


Figure 4.7: Dwell time is estimated by summing up the queue time, loading time and the scheduled waiting time. All these times are estimated separately.

Link travel time

The link travel time is based on three aspects; driving time (based on previously recorded link travel times and schedule adherence) and overtake limitations. Links are harder to model using the graph analogy than stops as e.g. queues can happen anywhere along the link. Therefore, we do not model the links as we do the stops. The link travel time is instead a function of its aspects, much like the *Loading time* of the stops.

5

Using a simulation based model in a forecast system

A real world application for giving traffic information to commuters should use live information and continuously update the displayed information. This means that the forecast system must continuously recalculate its forecasts to be up to date. This gives the forecast system two main responsibilities:

The forecast system which will handle the continuous inflow of vehicle reports and re-create the forecasts at specific intervals.

The forecast model to simulate all the traffic of the network and produce the time estimates of all forecasts.

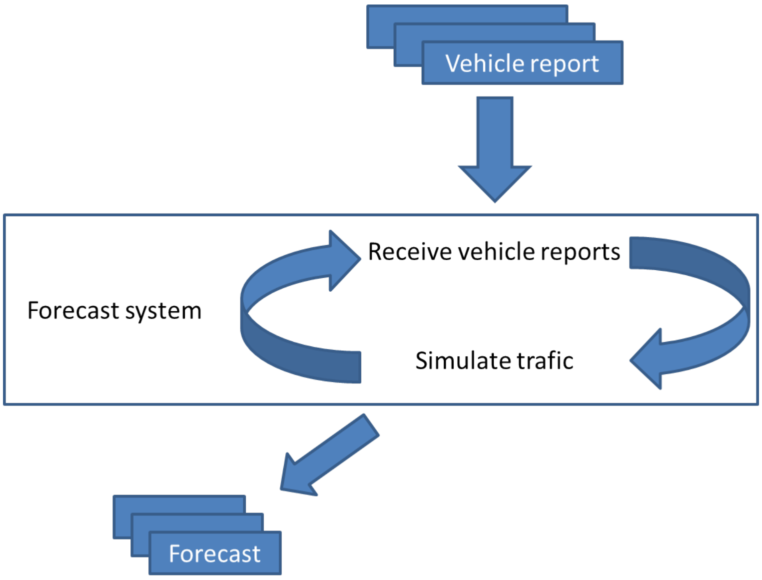


Figure 5.1: Process for creating forecast by the system

5.1 Algorithm for running the Forecast system

The forecast system needs to update its forecasts when it receives new information which differs from the previously made forecasts. This is to keep up with the unexpected. But, from a performance perspective it is interesting to limit the number of times the system creates forecasts for all future departures for all vehicles. It's also interesting how this can be limited; only forecast a subset of all departures and all vehicles. But this is not a part of this project. The opposite is to redo all forecasts every time new information is available. The quality of a newer forecast should never be lower than the quality of the older forecast. The reason for this is that the model knows more about the reality and can use this information when it estimates the future.

The model proposed by this project will recreate all forecasts for departures which are about to happen. No need to do any forecasts for events which has already happened. The model will repeatedly do this recreation with a fixed frequency.

The forecast system algorithm will,

1. collect vehicle reports in fixed time interval,
2. do forecasts for the complete system (only for future departures).
3. return to step 1.

5.2 Algorithm for the forecast model

The forecast model is simply a traffic model and a simulation algorithm which uses the traffic model to estimate how the traffic situation will be in the future given a specified state. The state should be what we know of the current situation in the real world; here we use all vehicle reports, all assumptions about the future and the time table which is also the information for the vehicles' paths.

One of the main arguments for simulating the traffic instead of using other non-interacting time estimation methods is the possibility to estimate times based on actual interactions. The problem of forecasting public transportation lies in the outer interference, e.g. traffic situation and passengers, but also inner interference. By doing a simulation the inner interference can be modelled, something that could not easily be done by other methods. In Table 5.1 you can see examples of how vehicles can interact. The value of using a simulation method compared to the non-interacting methods increase when the application is to forecast something which has low outer interference, as a layman guess would that might be airplanes to busy airports or train traffic which is not affected by passengers but only other trains in a limited infrastructure etc.

Table 5.1: Example of two vehicles' interaction at a stop.

Vehicle	Time	Performed action	Time of next action	Comment
C	13:00	Load passengers	13:02	
D	13:01	Arrive at stop	13:01	
D	13:01	Stand in queue	Unknown	Depends on C
C	13:02	Schedule waiting	13:02	Leaves because D stands in queue
C	13:02	Leave stop	-	D's Load passengers now has a time; 13:02
D	13:02	Load passengers	-	

5.2.1 Simulation model

The model used for the simulation is built from a few basic entities and relations which can describe a complete public transportation network. The entities are described in subsection 5.2.1 and Figure 5.2 describes the relations between the entities.

Vehicle The vehicle entity describes the moving vehicles of the model. This can be a bus, tram, boat etc. Every vehicle is always at some position and performing some action.

Position This is a stop or a driving distance (section of street, rail or water). Positions can have several vehicles which performs actions on it.

Action Every vehicle performs an action when it is on a position. This action can be drive, load/unload passengers, stand in queue, waiting for timetable etc. After a vehicle has finished an action the vehicle can either start with a new action on the same position or on a new position. A bus can, when it stops loading passengers at a stop, e.g start driving on the street towards next stop or wait for its scheduled departure time.

Aspect Used to estimate the time needed to perform a particular action. E.g. the driving time estimated by looking at historical data samples from vehicles travelling the same route. An aspect can be independent or dependent. The independent aspects do not use the simulated model for doing the time estimation. Estimating driving time, for example, only uses historical times. Dependent aspect uses information from the simulated model for doing time estimation, for example estimating the queue time for accessing a stop by looking on when the previously arrived vehicle will leave the stop.

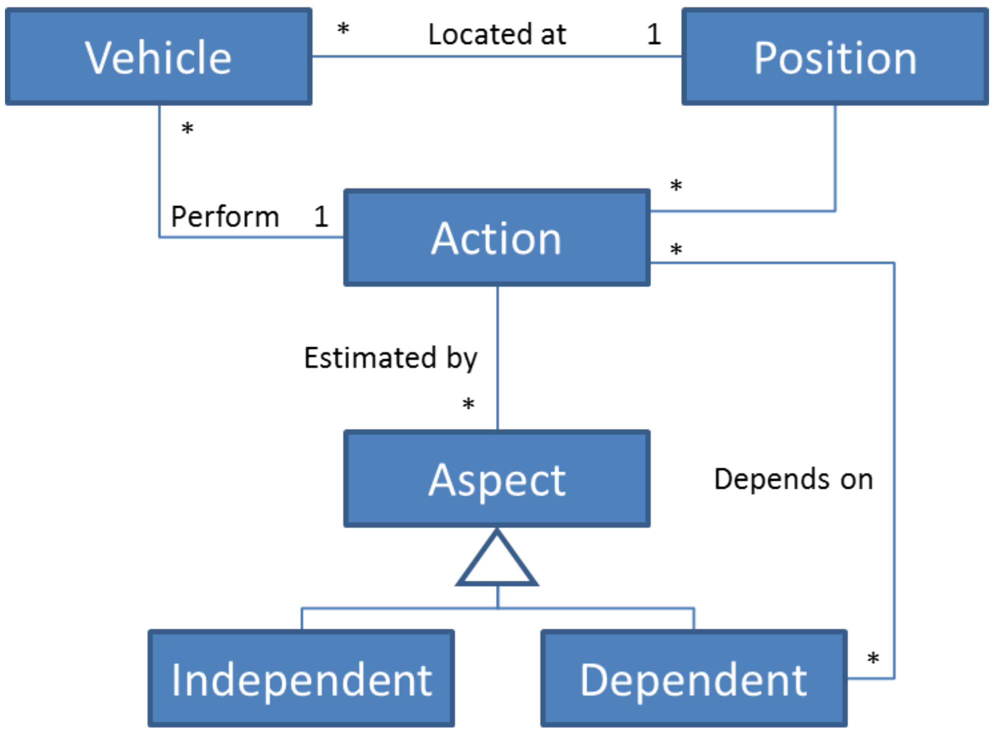


Figure 5.2: Model of the simulated world.

Limiting the dynamics by predefining vehicles' paths The setup used in this project is that only the vehicles' paths are predefined while the actions performed at each position are not. This means that the actions can be decided by the simulation model during the simulation, hence not limiting the simulation to a static predefined plan. The paths could have been determined during simulation but this is not the case in this report. The reason for this is that it is very hard to first estimate when and where there will be a deviation from the defined path and what the new path would be. The time needed for this deviation would also be very hard to estimate, especially for the first vehicle, since there is likely no data for the path. The forecast system in place today solves this by simply not producing forecasts for vehicles which makes deviations and are off track. Table 5.4 and Table 5.1 show how the simulator works with actions. The simulator is only aware of the next action for each vehicle even though it is aware of the complete path for each vehicle.

5.2.2 A simulation run

It is the result of a simulation run which will be used as forecast. The result is not only the state of the model when the simulation run ends but also everything that happened from the start to the very end. The length of a run is simply defined by how far in the future the forecast system wishes to create forecast for, e.g. 30 minutes, one hour or one day.

It is the vehicles movements and actions that will be simulated in the public transportation network. Some of the time stamps for actions will be used as forecast times. Every vehicle will move from position to position according to the path defined in the simulation model according to the time table for as long as the simulation runs. At each position the vehicle performs a set of actions defined by the position, e.g. at a stop the vehicle will stand in queue to enter the stop, load passengers and wait the scheduled waiting time. The time needed for these actions are defined in the aspect entity. The time estimation method can be found in chapter 4.

The simulation run is divided into two stages.

Stage 1 Initialize the logical representation of the public transportation network.

Stage 2 Simulate the vehicles' movement by using the aspect entities to estimate how long it takes a vehicle to perform the current action, and then move on to the next action or the next position.

Stage 1 The Initialize step will create a logical representation of the public transportation network which contains as much information as possible about what has happened in the network and what can safely be presumed. Information about what has happened is collected from AVL reports and information that can be safely presumed is the vehicles' paths. This also means that each vehicle will get a position in the network. The position will be their "last known location" based on their last AVL report. If a vehicle has not started driving yet, its position will be the same as the first stop according to its time table and it will depart from this stop according to said time table.

Stage 2 simulates how the vehicles move in the model by: (1) performing the action, e.g. load passengers from the stop the vehicle just arrived at, (2) add the next action with a starting time. (3) repeating these steps until nothing more should happen . Stage 2 ends when e.g. the simulated time has reached a specified end time of the simulation.

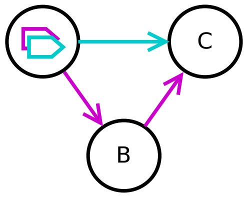
Following is an example run of the simulation involving two vehicles, Cyan and Magenta, on a network with three stops; A, B and C. The vehicles operate according to Table 5.2.

Table 5.2: The schedule used in this example.

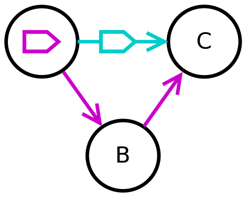
Vehicle	Path	To depart from first stop at
Cyan	A → C	T = 1
Magenta	A → B → C	T = 2

Table 5.3: A run of the simulation. The actions are handled in increasing time order. An action affects the start time of other actions

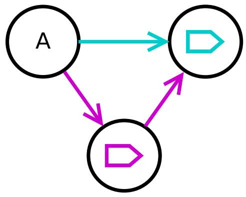
	Action	*Effect*
T = 1	Cyan departs from A	Add next action *Arrive at C* start time $T = 3$
T = 2	Magenta departs from A	Add next action *Arrive at B* start time $T = 4$
T = 3	Cyan arrives at C	
T = 4	Magenta arrives at B	Add next action *Depart from B* start time $T = 5$
T = 5	Magenta departs from B	Add next action *Arrive at C* start time $T = 6$
T = 6	Magenta arrives at C	



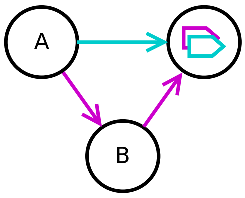
(a) T = 0. Both vehicles are at their first stop. Cyan is about to depart.



(b) T = 1. Simulation has begun by moving Cyan. It will arrive at C at T = 3



(c) T = 4. Simulation continues by moving Magenta which will depart from B at = 5



(d) T = 6. Simulation has moved both vehicles to their end stops. The simulation is finished.

Figure 5.3: A simulation run with both vehicles starting at their first stop.

Table 5.4: Example of one vehicle arriving at a stop and the different actions it perform before leaving the stop.

Vehicle	Time	Performed action	Time of next action	Comment
A	12:00	Arrive	12:00	
A	12:00	Stand in queue	12:00	No other vehicle at stop
A	12:00	Load passengers	12:02	
A	12:02	Scheduled waiting	12:03	

6

Results

In the process of creating a good forecast system it is important to benchmark ones results, but there are no standards on how to do this. Most other reports on the subject usually forecast and benchmark on arrival times instead of departure times as this thesis does, and they do this for multiple links and stops. This makes it very hard to compare the different models. This thesis' benchmarking has only been performed on a small period of time. The time period contains all of the days of the week. Benchmarking has been done on all the departures for the days in the time period. We selected a small time period for practical reasons, namely that the large data quantity took a long time to process. We are satisfied with the size of the time period because the traffic is cyclic and the results are as expected between the different days. Further, the forecast system is built to forecast the complete public transport network. This is one of the purposes of this thesis; hence it is reasonable to benchmark the complete network and not just selected clicks.

6.1 Assessing simulation accuracy

In order to assess the accuracy of the forecast system we needed a way to benchmark the generated forecasts. What being "accurate" means is quite naturally specified as: The closer to the actual arrival time a prediction is the more accurate it is. We define the operation of calculating how close to reality a forecast is as

$$\Delta t(\text{Actual departure time, Forecasted departure time}) = \text{Actual departure time - Forecasted departure time} \quad (6.1)$$

Example The forecast system says that a vehicle will depart from a stop at 14:00, but it actually departed at 13:55, then

$$\Delta t(13{:}55{,}14{:}00) = 13{:}55 - 14{:}00 = \text{-5 minutes} \tag{6.2}$$

If a forecast says that a vehicle will leave a stop after the vehicle actually left, then anyone trusting the forecast system will miss the vehicle. This is bad, and thus the delta value is negative. A forecast with a positive delta value will make a passenger wait longer than necessary, but at least she won't miss the vehicle.

Table 6.1: Sign of the delta function.

Actual departure time	Forecasted departure time	Δt	Sign
13:55	14:00	-5	Negative
14:05	14:00	5	Positive

To be able to grasp the performance of the proposed forecast system, the errors for all forecasts generated by the system are aggregated and shown as a mean absolute error (MAE). The lower the MAE, the better the forecasts. See Equation 6.3 for the complete definition.

Let F be the set of forecasts to benchmark, n be the number of these forecasts ($|F|$), Δt_f be the forecast error as defined in Equation 6.1 for forecast f, then

$$MAE = \frac{1}{n} * \sum_{f \in F} |\Delta t_f| \tag{6.3}$$

is the average accumulated absolute forecast error.

When evaluating the accuracy of a forecast we will, for fairness, consider how long before the actual departure time the forecast was made. We will denote this "generation **t**ime **b**efore actual **d**eparture" as the forecast's TBD value. If tram number 6 departs from Brunnsparken Site A at 14:16 on November 11th 2012, then a forecast made on the same day at 14:06 will have a TBD-value of 10 minutes, regardless of when it was scheduled to depart.

6.2 Benchmark process

The forecast system handles data from real-world sensors which aren't perfect and can thus measure erroneously. These errors in measurement can lead to forecasts that aren't indicative of the forecast system as a whole. We have implemented and used Peirce's Criterion according to Ross (2003) to identify these so called outliers.

To be able to compare the quality of the forecasts from the proposed forecast system, the quality of other forecast systems will also be listed in this chapter;

- Proposed model - The model that has been produced by this project
- Time table model - All forecasts come from the time table, no dynamics or adaptability is present in this model.
- Proposed model without vehicle dependencies - This model has similar logic to the proposed model but with some of the physical constraints removed, namely queues into stops and overtake at links.
- Proposed model without reference days - This model has limited access to older data, e.g. recorded times from the before. This model has the vehicle dependencies.
- Industry model - This model is the one currently in use to produce forecasts for the public transportation system in Gothenburg.

The conditions have been equal for every model; all forecast systems have had access to the same recorded information and the same contemporary information. Forecasts have been created for the entire network. If no real data for a departure's arrival or departure time exist, the departure will be excluded from the benchmarking. This happens when a vehicle has been taken out of service or due to technical issues. The same benchmark model has been used to measure the quality for all models.

To, again, ensure fairness, all models have had access to exactly the same data and is thus run on the same time period. This has been accomplished by a system that pushes *vehicle reports* to the forecast system at a predefined pace, see Figure 6.1 for a graphical representation of this system.

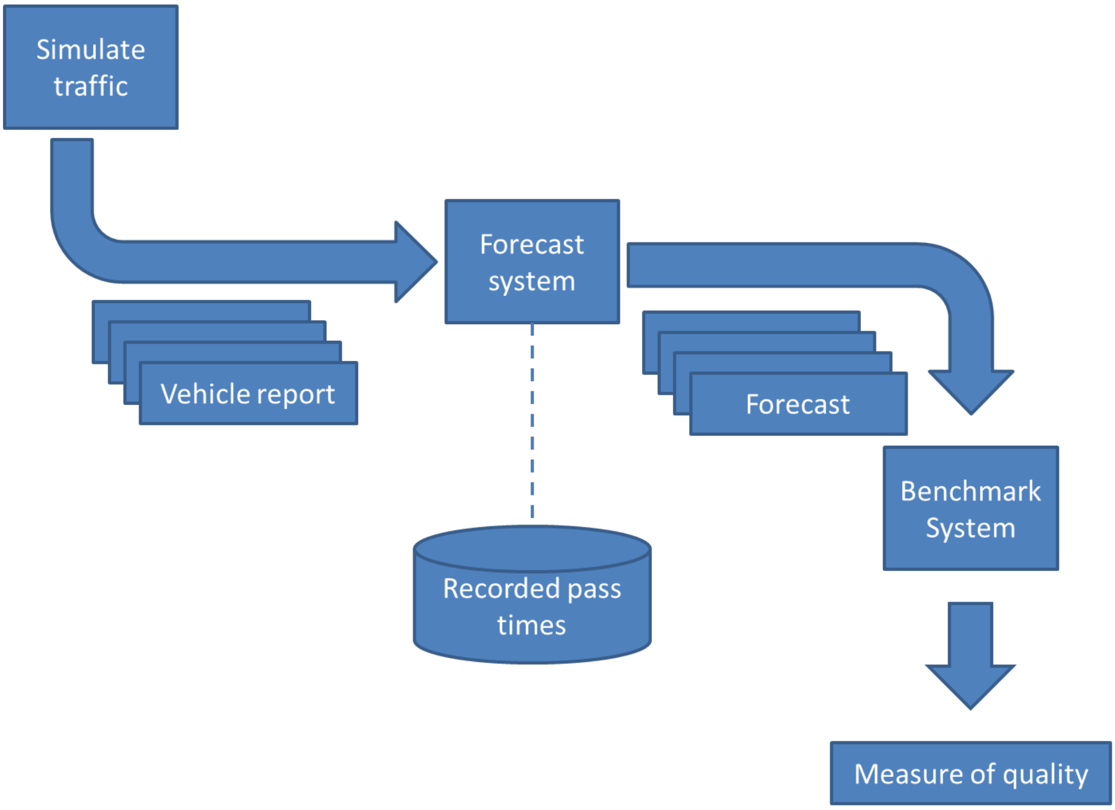


Figure 6.1: The process of benchmarking a model.

6.3 Benchmarked models

Each model has its own properties. These properties can be seen by looking at its inner model and its representation. It is expected that models with differing inner models produce forecast of different quality. It is also expected that a model can be very good at forecasting a small segment of the complete network while it unsuitable to use for the rest of the network. Below is a list of different models with a short description, key properties and results.

6.3.1 Time table model

This model represent a forecast system that produces forecast which is strictly same as the time table. When performing the benchmark for this model is there no actual need to run the simulation of *vehicle reports* because these reports won't affect the results. Likewise, this model will only create one forecast for each departure, compared to the other models that creates new forecasts when they have new relevant information.

6.3.2 Industry model

The model works with a moving average on stops for dwell time and on links for travel time. When a vehicle sends new information about its location and activity, the model will update the corresponding internal information. The model will update the forecast for vehicles affected by this new information. This model does not take interactions between vehicles into consideration.

Example A bus leaves a stop after standing still for 30 seconds. The bus that sent the message will re-estimate its future forecasts with its new departure time, as will all other buses on routes that pass by this stop.

6.3.3 Proposed model

This is the model proposed in section 4.3. Just as the *Industry model*, this model might create new forecasts when receiving new *vehicle reports*. This feature comes with the problem that these models creates a lot of forecasts for the same departure. This is quite natural, but it creates an extra dimension to the quality measure of a forecast. This dimension is not a problem for models like *Time table model*. Because it only creates one single forecast for each departure. The time table does not change or get updated during the same traffic day.

6.3.4 Proposed model without vehicle dependencies

This model is simply the *Proposed model* with a slight modification. The difference between the models is mainly in the physical representation. This model has no constraints for stops nor for links. This means that any number of vehicles can, according to this model, use a stop at the same time. Likewise can vehicles overtake each other. This also applies to trams that only has single rail per direction. The purpose of this model is only to be used as a reference on how big impact these constraints have. In all other ways are these two models identical.

6.3.5 Proposed model with full vehicle dependencies, but without reference days

This model is like the *Proposed model without vehicle dependencies*, only a version of the *Proposed model*. The purpose of this model is the same as *Proposed model without vehicle dependencies*. This model differs from the *Proposed model* only in the time representation. When estimating *loading time* and *link travel time* this model cannot use any recorded times except the ones the system receives in the form of a *vehicle report*. This can be illustrated by Figure 6.1 when the connection between *Forecast system* and database *Recorded pass times* has been broken. In all other ways are these two models identical.

6.4 Graphs

The performance of each modeled is measured in three different ways,

MAE

MAE at different TBDs

Variance

The main difference between these measurements is that *MAE* and *MAE at different TBDs* measure absolute error while *Variance* measure how volatile the quality of forecasts is. Both of these should be as close as possible to 0. But if *Variance* is 0 but not the other then there is a constant error in all forecast.

6.4.1 MAE

As can be seen in Figure 6.2, the model proposed by this thesis outperforms the commercial moving average on all days measured. It also shows that most of the accuracy comes from the reference days, as without them the model performs very poorly. The vehicle dependencies did not have as great effect, but it does increase the accuracy.

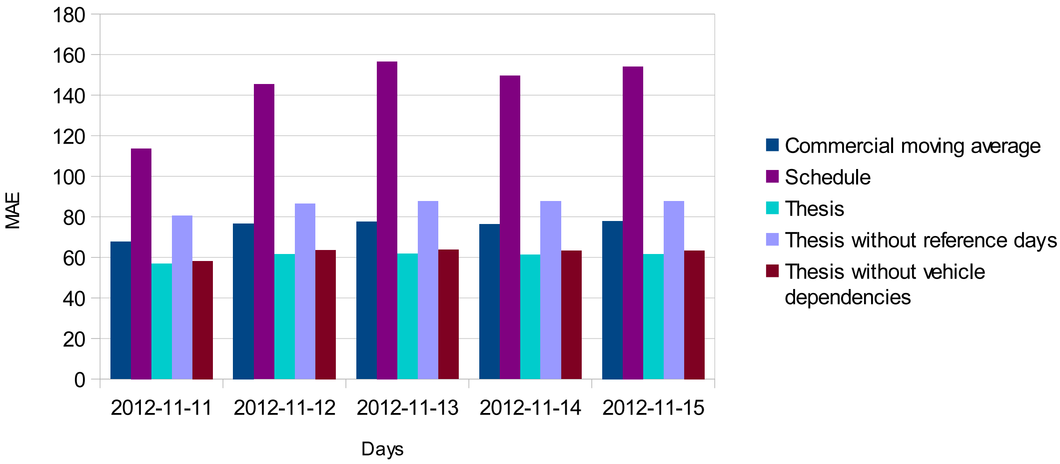


Figure 6.2: The MAE of the benchmarked models at TBD = 10 minutes.

6.4.2 MAE at different TBDs

The proposed model also performs well at forecasts generated more than 10 minutes before actual departure. However, it is less accurate than the commercial moving average at $TDB \leq 3$. The commercial moving average produces forecasts less accurate than the schedule at $TBD \geq 40$. This is illustrated in Figure 6.3.

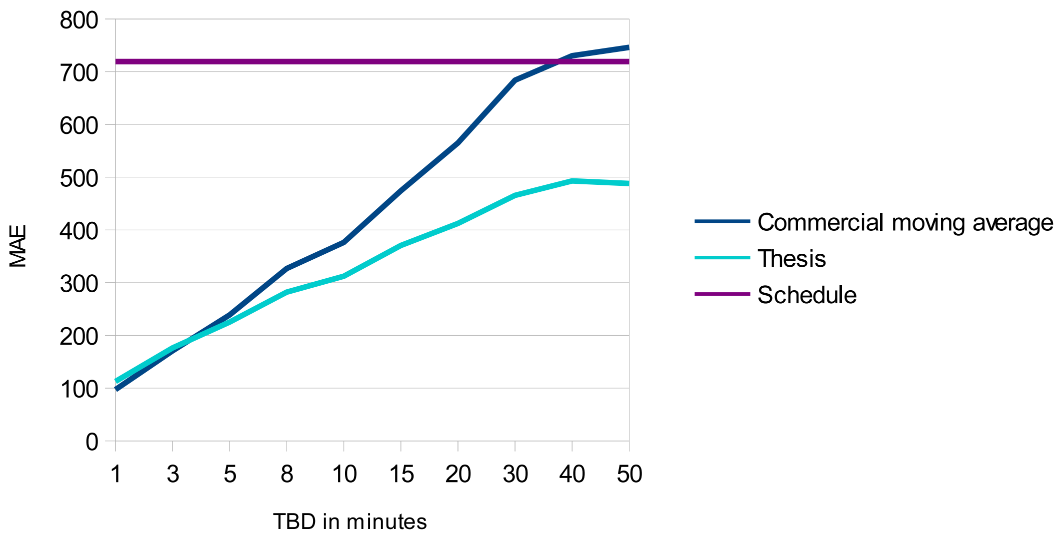


Figure 6.3: The MAE of forecasts generated at different times before actual departure.

6.4.3 Variance

Figure 6.4 and Figure 6.5 show the proposed model's and the commercial moving average's variance with outliers included and excluded. The commercial moving average deliberately moves their average away from zero in the positive direction, while the proposed algorithm strives towards a zero mean. This means that the commercial moving average will inherently get a lower variance. With the outliers removed, the variance plummets while the number of samples doesn't - and the proposed model has the lower variance, showing that the proposed model produces even results.

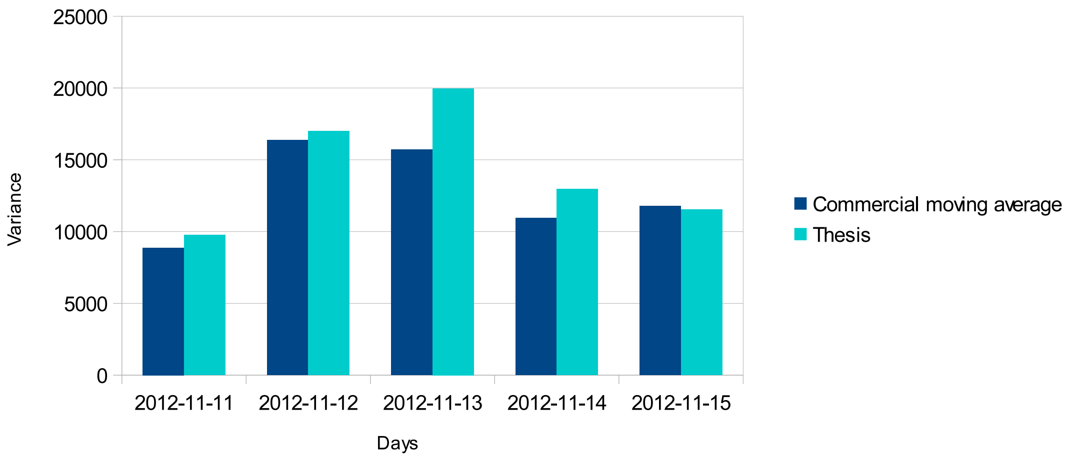


Figure 6.4: The variance including outliers. All forecasts are generated 10 minutes before actual departure.

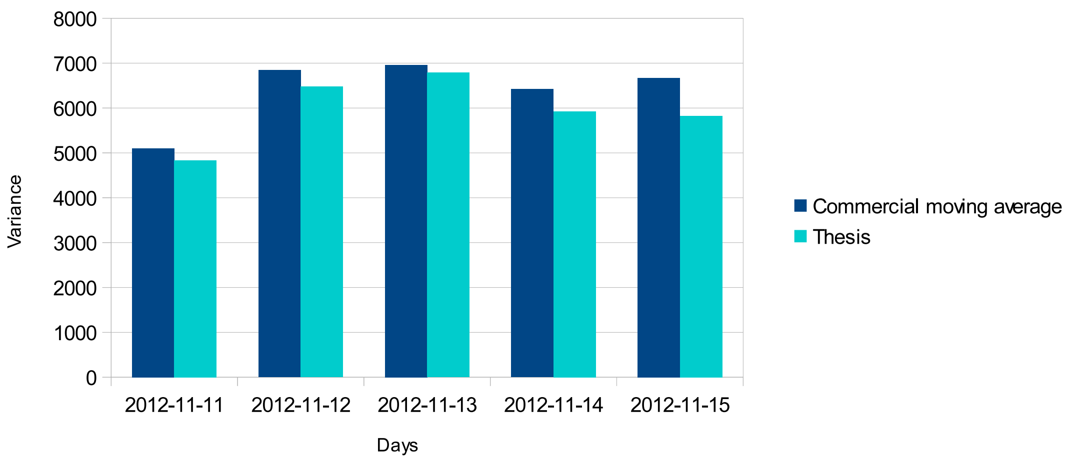


Figure 6.5: The variance excluding outliers. All forecasts are generated 10 minutes before actual departure.

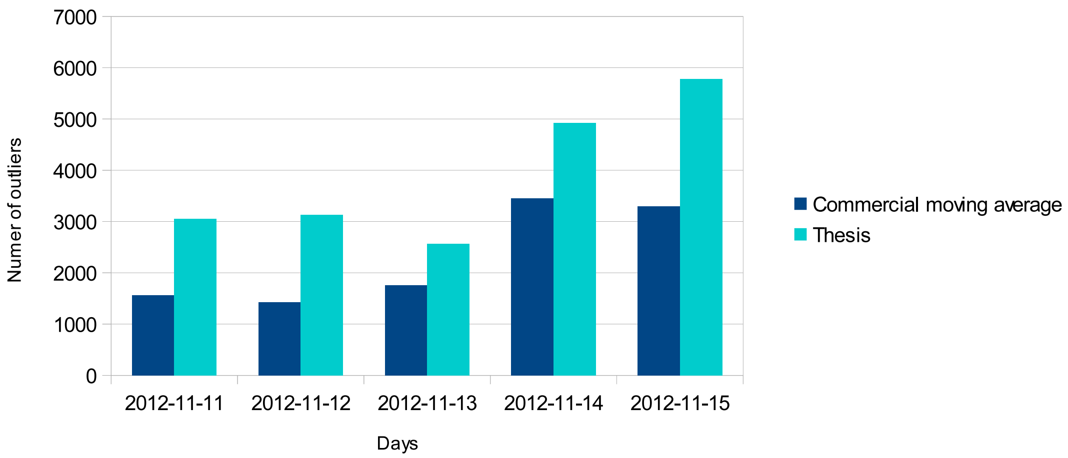


Figure 6.6: The number of samples removed by outlier elimination.

7

Conclusions and discussion

The purpose of a forecast system is to fill the gap between the time table and reality; if the buses always arrive on schedule then there is no need for a forecast system. The purpose of the time table is to describe the traffic in a general way; capacity, frequency, paths and time, but it does not consider every possible scenario and each vehicle's special conditions for each day. Thus there is a need for a forecast system to be able to supply the passengers with accurate information in advance when the vehicle will depart and when it will arrive.

The driver's job assignment is to drive safely and on time according to the time table. Because of the driver's job assignment the forecast will depend on the time table. The difficulty with creating a forecast system is which aspects affect the time that the vehicle uses to perform an action and how the different aspects interact with each other.

The simulation based rational model is a viable option which allows modelling of complex inter-vehicle dependencies and driver and passenger behavioral patterns. These dependencies and behaviors greatly affect travel times and are often too complex to be accurately accounted for in a purely statistical model.

7.1 Metrics

This thesis presents the results in MAE values. This is because MAE is one of the methods used by other theses on the subject and because it gives a good representation of the average quality without reacting too much to extreme cases but still use them for calculation. There is a problem with using a strict average for measuring quality; "two opposite errors cancels out each other". This situation is bound to happen when the quality of a forecast goes towards zero. To make sure that this will not happen the error will always be positive when considered for MAE calculation.

MAE shows in the average error a good way, but it does not say much about the expected interval that the error will be in - especially when all errors are positive. We use variance to represent this. Here, the extreme cases will dominate the value. If the variance is zero then the quality is perfect, but with a constant error. In contrast to MAE, using absolute values of errors can only be 0 if the quality is perfect.

It is possible to include all forecasts quality measure of the system, but it is impractical. There are simply too many forecasts. This thesis has only one forecast for each departure been taken into account when measuring the quality. How would one compare to different forecast for the same departure? One can use relative measurements depending on how much in advance the forecast was made. Then the big question will be if this scaling function really should be linear and at what slope. Instead we pick the youngest forecast that is at least a fixed time old for each departure. The youngest can be a lot older than the fixed time. This happens when the system has not generated a new forecast for the given departure in a long while. The reason why the system does not generate new forecasts is because it has not got any new relevant information that would contribute to another forecast, hence the older forecast should be considered to have the same quality as a forecast that is exactly as old as the fixed time. The fixed time used is 10 minutes. The fixed time was chosen in a dialogue with Consat. One of the main arguments for this is that passengers often look for and need accurate forecasts 10 minutes before they would like to depart. We did not look closer at the possibility of choosing a benchmark that took passenger benefit into consideration since what defines passenger benefit is a thesis all in its own.

7.2 Results

The small effect of the vehicle dependencies might be because of the relatively few departures in urban areas, where the queues have great effect, to the many departures in rural areas where traffic is rarely a problem. Or, it could be because many stops where vehicles interact with each other demands a more advanced model than the one introduced in this thesis. In this thesis all stops have been modelled to only allow one vehicle standing still at it at any time. This is correct for many stops, but not for all. There is also a problem in how to interpret the historical data; e.g. if a stop in reality has capacity 2 and it has two vehicles on it loading passengers. After a while the first vehicle will leave the stop, but the second one will only drive to the forward position on the stop to re-open the doors and load passengers again. Only after this second loading is complete will the vehicle leave the stop. If this happens over time the average vehicle's stop time will be double, disregarding loadings of passengers that are done in parallel. Our model has problem with handling these situation for these stops.

The fact that the commercial moving average produces more accurate forecasts than the proposed model at $TBD \leq 3$ could be because;

- The method used in this master thesis uses information from both the day to do forecasts on and a number of reference days. If these reference days are not similar to the day to do forecast on there will be irrelevant data used when calculating loading time and link travel time which may lead to poor forecast accuracy.
- Since the implemented method places focus on vehicle interactions, it becomes more vulnerable to something called "ghost buses", buses that are scheduled for traffic but never leaves its starting stop. The simulator thinks that these buses are driving around and standing in queues which causes the overtake limitation, queue time and scheduled waiting time to be erroneously estimated.
- The proposed model is more sensitive to imperfections in the data than the moving average. If an AVL report is a few seconds off many minutes before actual departure, the error won't affect the forecast as much as it is just one report of many used in the calculation - and the errors may cancel out each other. The lower number of reports influencing the forecast at lower TBDs gives each error more room to affect the forecast.

The moving average does not look at any of these aspects, thus when the aspects' or the reference days' data are invalid - the moving average method will be better to use.

The graphs in Results - Graphs clearly show that the scheduled departure time is not a good forecast and thus a complex forecasting system is indeed warranted.

7.3 Simulation based forecast model

The proposed forecast model and its algorithm for this thesis is quite different compared to the once we have seen in similar reports. This model is more extensive than the others. The model proposed have great potential for improvements, both when it comes to the actual estimation of time but also the possibility the model the actual problem with great resolution to details. It is even so good that this models allows the model to have big difference between the level of details used to describe the problem, e.g. a section of road can every traffic light, every pedestrian crossing and speed bumps be modeled while another section is only modeled to have a constant time.

7.3.1 Simulation based model

The forecast model used in this thesis is simulation based. Thanks to this, have it been simple to model most of the behaviors in traffic, e.g. vehicles, passengers, infrastructure etc. The model allows for complex dependencies and good debugging. The choice of having a simulation based model was made when we needed to describe a behavior that both rely on actions that had happen and on actions that was about to happen. The simulation based model allowed vehicles to have uncertain times when to end an ongoing action and thereby can dependencies which looks both back and forth in time exist, example *the schedule waiting time from stops*. This model allows the model designer to make big mistakes, e.g. dead locks, but more importantly it allows the model designer to model everything as she want it to be. If the problem is modeled in a correct way then will there be no deadlocks if it is not it the real world as well. This also adds another application. The application of testing a public transportation infrastructure before running it in real world, e.g. changes to the time table or re-constructing the road infrastructure.

When creating the entities in the model have we tried to think of *single responsibility principle* and *separation of concern*. This has allowed us to model the problem quite nicely where with have separation between concrete objects, what objects can do and how long time is needed to do it for the concrete object. It has also made it quite easy to create constraints to the model, e.g. capacity constraint on stops and overtake constraint on streets. This requires a lot of work for implementing the model and if the features of this model are not used will the overhead be big compared to what you get from a none-simulation based forecast model. But, you will get a model which allows for continuously improvement.

7.4 Time estimation model

It is not enough to create a great method for modeling the problem to succeed to create forecast of great quality, meaning the simulation base approach with the possibility of dividing the model to a set of different entities. To create good forecast one must create a good method for estimating the actual times it takes to perform the different actions. Even if this thesis has succeeded in outperforming the commercial forecast system one must say that there is still room for improvements. A simple grouping of time estimating methods can be does methods that try to model pure constraints and does which are not. The group which is based on constraints are dependent of other simulated vehicle and thereby the other simulated vehicles' forecasts.

Not based on constraints

1. Driving time - Time needed to drive between to stops
2. Loading time - Time needed to load passengers
3. Waiting for time table* - Time to when time table state that vehicle will departure

Based on constraints

1. Queue to stop - Time needed for queue to be first in queue and stop is free
2. Overtake limitation - Time needed for vehicle in front to leave the travel distance (street, rail etc.)
3. Waiting for time table* - Time to when next coming vehicle wish to arrive at stop

** Please note that this is part of both because it is estimated by two aspects and then the time lowest time is picked.*

There are a lot of things that could have been done differently when it comes to time estimation and actions. Some things would probably increase the quality of the forecasts, but most things would have increased the complexity of the model. This model is made to describe a set of different actions/behaviors from the real world. The model is made to be simple and clean but still model the real world and showing that this model could quite simple changed to model any actions from the real world with appropriate estimation method without changing the core infrastructure.

7.4.1 Not based on constraints

The time estimate produced in the context of *not based on constraints* will not rely on any other action in the model. This is a part of the separation of concern in the model. The exception to this is when to estimate how long time a vehicle will wait for its scheduled departure time. If looking at Figure 5.2 one can see that the case is that some action rely on several aspects and as in this case there are two aspects and the rule is to use the lower time as the time estimate.

Link travel time This time estimate is done by doing an average on using historical times for vehicle which traveled the same link under similar conditions. Two big questions are "are the right conditions used?" and "Do we use relevant historical data?". The intent is of course to answer "yes" to both questions. The method used to do the estimate has been to rather limit the quantity of historical data. In this way one might say that conditions which are periodic at a similar way as the filtering are considered. For instance by only using data from same type of week day and around same time on the day, hence it should have similar traffic condition. By using data from same day and only few days before will the weather conditions be similar. An improvement to the model could be by separate all of these constraint and minimize the filtering when picking historical data. Other conditions used are if the vehicle is on time according to time table. This condition is quite hard to say anything about because there can be several reasons why a vehicle is late, example it got stuck behind a slow moving vehicle or because it has a lot of passengers onboard. In one case should it be possible for the driver to catch up a little bit of time while on the other case should it be the opposite. So preferable should this condition be divided up to just those two. The intent is that if this is a cyclic behavior then will it be correctly forecast else is it either too complex for this model or an exception.

Loading time The time estimate is done by doing an average on historical times for vehicles of same line on the same stop for similar conditions. This is similar as *link travel time* and much about the reasoning is same here. The differences are which conditions that are used for the average. The conditions are how the vehicle follows its time table and how long time has elapsed since last vehicle of same line was at the stop. The case here is similar as in *link travel time*, it would be nice to model the passenger individual speed and amount of passengers but this is not possible. Instead are we using the hope of cyclic behavior and trying to make a clean model.

Waiting for time table This time estimate is very simple to do because it is just to calculate the difference between when the vehicle is done loading and the schedule departure time. One problem can be that not all drivers follow this instruction, even not the instruction about leaving when another vehicle approaches. One way to model this could be to look for patterns for specific stops or even go one step further and try to model specific driver behavior. The latter option is definitely out of scope for this thesis. Most cases are that vehicle not that early to the stops except the first stop of a journey and when they are early they are quite good at departure from it according to schedule.

7.4.2 Based on constraints

These *constraint based estimate* rely on two things. (1) They rely on the quality of other forecasts. A constrained based forecast will use other forecast as reference for estimating the time needed to perform its action. If the constraint is that it has to stand in queue then will the time estimate use information from the forecast of vehicle blocking it. (2) The t is modeled correct. In this thesis all constraints are constant over time, e.g. no street will get extra lanes because of peak hours. Also has it been simplified that all position of same type have same constraints, e.g. all travel distance are on single lane.

Queue to stop There are some stops which it is impossible for two vehicles to load and unload passengers at the same time. But is not the case for all stops, in fact many of the busses' stops can have more than one vehicle loading/unloading at the same time but there is some more extra actions the vehicle performs. So instead of creating a bigger model, which was needed to be custom fit to each stop, was this simplified by saying the capacity is only 1. Also to implement this change require a lot of transformation of the historical data. Here is room for improvements.

Overtake limitation This is similar as for *Queue to stop*, we simplified it by using the same constraint for all links between stops. In reality are there many links which have a more than 1 lane. The reasoning by saying that all should be 1 was of course to simplify but also the fact that the places which have high frequency of vehicles often have a separate lane. The rest of the places are it less likely that one vehicle will catch up with the vehicle in front of it, but it can happen and this is a place for making the model more accurate to real world.

Waiting for time table This is part of a verbal agreement among the drivers. If every driver honor it or not is not possible to say. The interesting thing about estimating this time is that it depends on two separate concerns. As *constrained estimate* will it only depend on the next arriving vehicle.

7.5 Future work

The results from our model is promising, but can be improved. Following are a few areas in which we would recommend more research:

Specialized time estimation models The implemented time estimation model supports the use of different time estimation models on each segment in the journey graph. This means that we can have completely different pass time estimation models in different geographical areas and on different times of the day; Time estimation models may be specialized for rural or urban areas, or they may be more effective during low traffic hours, while another may be more effective during high traffic hours. The vehicle type, e.g. bus, tram or boat, may also require different models. We have used only one model for links and one for stops, having tried to make these general enough to generate accurate forecasts for all different circumstances.

Other aspects There are many more aspects to include, e.g. weather, accidents, concerts, sport events, the color of red lights, etc.

Similarity calculation The ”similar circumstance” approach is also to a pure historical model. This hints at the possibility to use more refined algorithms such as a Kalman filter to find these ”similar” times.

Model capacity constraints In our proposed model we sets all stops to have a capacity of 1 vehicle loading/unloading passengers at a time and that no links allows for overtake. This is not true in the real world. A more correct model would use the correct capacity for all stops and links.

Table 7.1: Suggested simulation settings for a distributed network.

Node	Frequency	Simulation length
A	45 sec	60 min
B	10 sec	12 min
C	120 sec	120 min

7.5.1 Performance

This is not a question for this thesis. But our simulation based algorithm is slower than the commercial moving average algorithm. But with improvements to implementation and cleaver choices on when to run simulation and for how long would be sufficient for running the system in production for the network, see Table 7.1 for example. Also could the simulation be distributed on several nodes to create new forecast more often. Node A in Table 7.1 is the setting mostly used for our results. This setting was able to run and create forecast faster than real time.

Bibliography

Chen, G., Yang, X., Zhang, D. & Teng, J. (2011), 'Historical travel time based bus-arrival-time prediction model', *ICCTP* **Towards Sustainable Transportation Systems**, 1493 – 1504.
URL: *http://books.google.se/books?id=WL8ZPiwXFdoC*

Jeong, R. & Rilett, L. R. (2004), Bus arrival time prediction using artificial neural network model, Technical report, Washingtong D.C.,.

Ross, S. M. (2003), 'Peirce's criterion for the elimination of suspect experimental data', *Journal of Engineering Technology,* **Fall**.
URL: *http://classes.engineering.wustl.edu/2009/fall/che473/handouts/OutlierRejection.pdf*

Shalaby, A. & Farhan, A. (2004), 'Prediction model of bus arrival and departure times using avl and apc data', *Journal of Public Transportation* **7**.

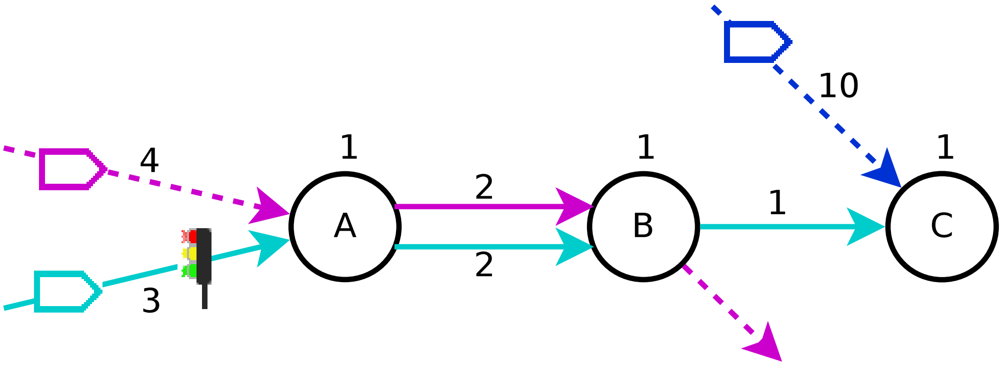
Yu, B., Lu, J., Yu, B. & Yang, Z. (2009), 'An adaptive bus arrival time prediction model', *Proceedings of the Eastern Asia Society for Transportation Studies* .
URL: *http://www.easts.info/publications/journal_proceedings/journal2010/100064.pdf*

A

Examples

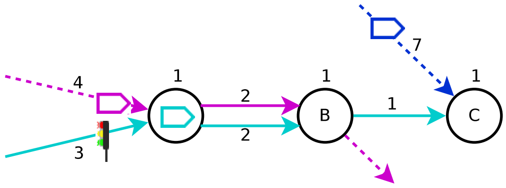
Graphical example of complex dependencies

Following is an example showing how the color of one red light can have nontrivial consequences. The example is from cyan's point of view.

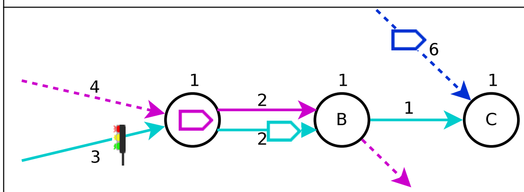


Cyan approaches a red light.

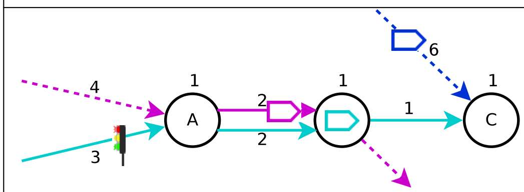
The light is green



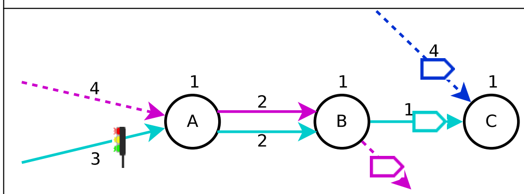
Cyan passes the green light, Magenta is forced to wait.



Cyan is ahead of Magenta.

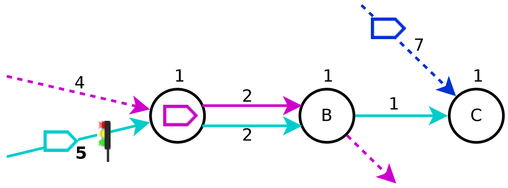


Cyan arrives at B before Magenta who is forced to wait.

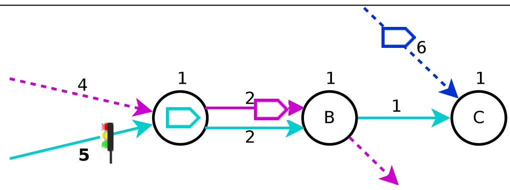


Cyan arrives at C before Blue who is forced to wait. Cyan's total travel time to C is 8 seconds.

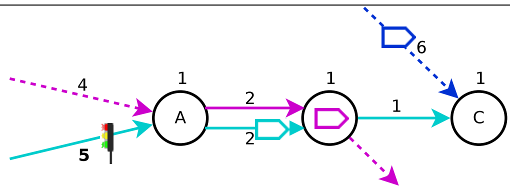
The light is red



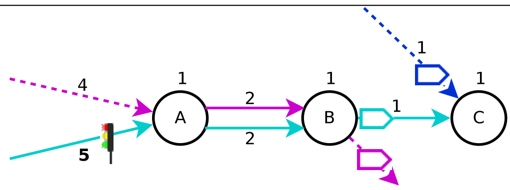
Cyan passes the green light, Magenta is forced to wait.



Cyan passes the green light, Magenta is forced to wait.



Cyan passes the green light, Magenta is forced to wait.



Cyan passes the green light, Magenta is forced to wait.

Moving average

A moving average differs from a normal average only when it comes to handling new numbers. A normal average of three numbers could look like

$$\frac{1+2+3}{3} = 2 \tag{A.1}$$

If a new number is added, the denominator needs to be updated to reflect the increase in number of numbers

$$\frac{1+2+3+4}{4} = 2.5 \tag{A.2}$$

In a moving average the ”oldest” number is removed from the summation in the nominator and the denominator is kept constant.

$$\cancel{1+}\frac{2+3+4}{3} = 3 \tag{A.3}$$