Geographical Information Management and Applications

Improving traffic system performance by combining tolling and intention-based prediction: an agent-based model

Nino de Maat
4136675

Universiteit Utrecht
TU Delft
University of Twente.
Wageningen UR

28-02-2020
MSc Thesis
Supervisor: dr. ir. A. Ligtenberg
Resp. Professor: dr. ir. R.J.A. van Lammeren

Improving traffic system performance by combining tolling and intention-based prediction: an agent-based model

N.B. de Maat

February 26, 2020

Summary

A traffic system in which all drivers optimise their individual efficiency does not always lead to the most efficient use of a road network. Rather, ”selfish-routing” can lead to *user equilibrium*, which optimises individual travel time. This is not necessarily equal to the *system optimum*, in which the average travel time is minimised. This study refers to this phenomenon as the *inefficiency problem*. Modern communication and navigation devices arguably increase the tendency of a network to reach user equilibrium. However, the adoption of autonomous vehicles and new technologies also offers possibilities to route traffic in a more efficient way. This study explores some of these possibilities and has examined the effects of tolling and predictive methods on the performance of a traffic system. It does so by creating and assessing an agent-based traffic model, in which Δ-tolling and intention-based prediction are combined with the aim to lower the average travel time and thus improve the efficiency of the network. In Δ-tolling, road segments are tolled according to their current congestion, and in intention-based prediction, future states of the network are predicted through the shared travel plans of drivers. These concepts have been researched separately before, but this study adds to both research fields by combining them.

Several scenarios are used to assess the outcome of the model: an as-is scenario, a predictive scenario, a system-optimising scenario, and a predictive, system-optimising scenario. The model is ran on three different networks, two of which are hypothetical test networks, and one of which is a simplified network of Sioux Falls (USA). For each of these scenarios and for each network, the study examines the average travel time. The study also examines the effect that the scenario has on the distribution of travel times (equality) and the impact the scenario has on individual drivers (fairness).

It is found that both Δ-tolling and intention-based prediction are methods capable of improving a system’s performance. Δ-tolling appears to have a larger effect on the average travel time than prediction. However, it is shown that both methods can complement each other.

The average travel time on networks with sub-optimal tolling settings can be decreased by applying intention-based prediction, too. This implies that if optimal tolling-settings are not known, the efficiency of the network can improve further through intention-based prediction. It also implies that even if optimal tolling settings are known, lower, less-intrusive toll rates can be combined with intention-based prediction in order to achieve a similar result to optimal toll settings. The study finds that both methods have only limited impact on the equality and fairness of the traffic system. Nevertheless, there are some drivers who benefit greatly from the measure, whereas others are disadvantaged.

The study concludes therefore that both Δ-tolling and intention-based prediction show potential to improve the efficiency of traffic systems, but further research needs to be conducted to examine the impact these measures have on individual drivers. Moreover, policy decisions about this impact needs to be made before measures are implemented.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 The problem of selfish routing

The advent of autonomous vehicles enables a higher degree of traffic control than ever before. The perceptive and responsive capabilities of the sensors and processors of an autonomous vehicle (AV) outperform the senses and brainpower of a human driver. If properly designed, an AV can process and react to information safer and faster than any human counterpart (Dresner & Stone, 2008). Moreover, AVs can be connected to other vehicles and systems through the Internet of Things, changing their routes in response to live traffic data. This opens opportunities for AVs to drive in ways that limit congestion (Bagloee, Tavana, Asadi, & Oliver, 2016). The implications of this behaviour and large scale implementation of AVs in traffic are yet to be fully understood, giving rise to a need for research on the behaviour and opportunities of AVs on road networks (Klein & Ben-Elia, 2018).

One possible area of application of the capabilities of AVs–and of existing navigation devices–is traffic flow optimization. In contemporary traffic, a regular driver does not drive in such a way that optimizes the flow of traffic on a network. Rather, he drives in a way most beneficial for itself, using information on the network and other drivers. This default, ”selfish” behaviour results in the User Equilibrium: a stable equilibrium where all drivers achieve the optimal route for themselves: the route taken is the quickest one available, given the network design and the behaviour of other drivers. As a result, no one can switch to another route to be better off (Liao, Dai, & Chen, 2012). However, selfish behaviour does not necessarily lead to System Optimum; the situation in which the total travel time of all agents is minimized. In other words: the road network is used inefficiently (Helbing, Schönhof, Stark, & Holyst, 2005).

Arguably, this problem has increased in recent years. Not only autonomous cars, but also regular cars connected to the internet through existing navigation devices and mobile phones now provide drivers with real-time information, enabling drivers to select the most optimal route faster than ever before. Thus, the inefficient User Equilibrium is reached quicker than compared to a situation with less information (Helbing et al., 2005). Additionally, navigation devices are only capable of responding to current congestion and other traffic situations, and are unable to act on future states of the network. This results in delayed routing advice and overreaction phenomena, both causing congestion and delay rather than avoiding it (Mahajan, Hegyi, Hoogendoorn, & van Arem, 2019).

Even though increased intelligence of autonomous vehicles and contemporary communication and navigation systems can initially hinder reaching system optimum, technological developments open up new opportunities to reach system optimum, as well. Solutions for moving a traffic system from user equilibrium to system optimum have been proposed since the 1920s (Pigou, 1920), but modern monitoring and communication technologies bring actual implementation of such solutions closer than ever (e.g., Levy, Klein, & Ben-Elia, 2018), for example by charging dynamic tolls (e.g., Sharon, Hanna, et al., 2017). Moreover, modern communication devices can—theoretically—enable drivers to respond to predicted network occupancy based on communicated travel intentions of other drivers, rather than to historical or current network occupancy, and thus increase their individual efficiency (e.g., Mahajan et al., 2019). Agent-based modelling (ABM) is often used to model system-optimising or predictive traffic systems, because it is capable of modelling complex systems with many interacting entities. However, current agent-based simulations have not yet combined the system-optimising and intention-predictive approaches: simulations are often either aimed at system-optimisation using past or present data (e.g., Dong, Mahmassani, Erdoğan, & Lu, 2011; Sharon, Hanna, et al., 2017), or predictive and aimed at individual optimisation or user equilibrium (e.g., De Juncker, Phillipson, Bruijns, & Sangers, 2018). Because the potential for combining these two approaches has been acknowledged in both system-optimising (Sharon, Hanna, et al., 2017) and predictive works (De Juncker et al., 2018), this research contributes in closing this gap.

1.2 Research objective

This research aims to contribute to system-optimising solutions and to combine predictive and system-optimizing techniques to develop a traffic ABM in which agents are supplied with current and predictive network information on which they act in order to improve system performance. In this research, a traffic system is a network on which agents travel from origins to destinations. Performance is defined as the average travel time agents experience while traversing the network. However, if a method is to be translated to a real-world application, other characteristics of a model are highly relevant as well. Therefore, the equality and fairness of the model is assessed, too. Equality refers to the distribution of travel times among agents: do all agents experience similar travel times? Fairness refers to the impact that measures have on individual agents: is every agent affected in the same way? The main objective of this research is:

To develop and assess an agent-based traffic model (ABM) in which agents are supplied with multi-temporal network data to improve insight in the effects of predictive information on system performance, specifically when compared to user equilibrium.

Because this research aims to build on previously proposed methods, it is necessary to first construct a solid conceptual model, in which the fundamental workings of the method are explained. The first research question is therefore:

- **RQ 1** What conceptual model using multi-temporal data can be used to improve the performance of a traffic system?

The next step is to operationalise this conceptual model and translate it to a functional agent based model.

- **RQ 2** How can this conceptual model be implemented in an agent based traffic model?

After this step, the sensitivity of the performance of the traffic system to different scenarios is to be assessed. The scenarios consist of an as-is scenario, a predictive scenario, a system-optimising scenario and a predictive, system optimising scenario.

- **RQ 3** To what extent is the performance of the traffic system sensitive to different scenarios?

Finally, this research evaluates the impact of the different scenarios on the equality and fairness of a traffic system.

- **RQ 4** What is the impact of the different scenarios on the equality and fairness of a traffic system when compared to user equilibrium?

1.2.1 Scope

It must be noted that this research does not aim to design a real-life, readily implementable traffic control system, or at displaying a real-life traffic situation as accurately as possible. Neither is it aimed at improving a specific, real-life traffic system. Rather, this research explores the functioning of a proposed solution for increasing the performance of a traffic system, using an agent-based model. It can thus be seen as a ”proof of concept” of using predictive methods to increase system performance on a traffic network. The methods are first applied on a simplified model, to test its workings. After that, the method is scaled to the Sioux Falls road network to examine its effect on a more complex network.

1.2.2 Relevance

Research on the optimal behaviour of AVs on a road network has a high societal relevance. In the near future, AVs will likely be a part of everyday traffic (Bagloee et al., 2016). This offers new opportunities to increase traffic safety and efficiency. The number of accidents, fuel consumption and congestion are all likely to decrease, which in turn leads to lower societal and financial costs (Dresner & Stone, 2008). It is thus required to study the effect of AV adoption on real-world road networks and traffic systems, and to find ways in which to design the behaviour of AVs in order to reap the potential benefits. However, even before the autonomous car is a common sight on the road, the insights of this study might be of value. Currently existing navigation devices can connect cars to the internet as well and thus might be capable of providing drivers with useful information. Even though autonomous cars are better capable of processing this information, a system with regular vehicles might thus benefit from multi-temporal information, too.

By providing insight in how to reap the potential benefits of autonomous and other connected vehicles, this study is also of relevance for policy makers concerned with city planning, road planning and transportation.

Besides making a contribution to achieving societal benefits, the objective of using a predictive method to increase system efficiency also adds to existing literature. As both methods for prediction and system optimisation have been proposed before, this research aims to build on these methods and to combine them, thus creating a method which is both predictive and system optimising. To the author's knowledge, such a method has not been proposed yet.

The remainder of this report is structured as follows. Chapter 2 addresses the theoretical background of the user equilibrium – system optimum dissonance and of its solutions. Chapter 3 explains the methods and models applied, after which chapter 4 discusses the results. A conclusion and discussion is provided in chapters 5 and 6, respectively.

Chapter 2

Theoretical background

This chapter provides a theoretical background of the problem presented in the introduction. For lack of a commonly accepted term in literature, the discrepancy between user equilibrium and system optimum is referred to as the *efficiency problem*. First, the foundation of the problem is explained using a numerical example. Then, different approaches to solutions are described. Besides their basic workings, the benefits and disadvantages of each method are discussed. Subsequently, the chapter provides a more in-depth discussion of tolling methods and predictive methods. Finally, the chapter presents some conclusions from the theory, and relates those to this research.

2.1 The foundation of the efficiency problem

Research on the discrepancy between User Equilibrium (UE) and System Optimum (SO) is not new. Pigou (1920) first described the situation in which individually optimising routing strategies might result in suboptimal system welfare. In 1952, Wardrop demonstrated mathematically that such selfish strategies do not necessarily result in the best performance of a traffic system. He noted that the most likely behaviour of traffic participants is to select routes in such way that all used routes between a given origin and destination have an equal travel time. If this is not the case, individual participants can change routes to select a faster route until an equilibrium is reached in which no one can switch to a faster route. This is the User Equilibrium which does not have to coincide with System Optimum (Wardrop, 1952).

2.1.1 Pigou's example

So what makes that this inefficient equilibrium arises? Roughgarden offers a clear explanation: the equilibrium is inefficient because *"Selfish users cannot resist overcongesting a route that is beneficial when used in moderation"* (2003, p. 362). The simple network below, an adapted version of Pigou's example, illustrates this reasoning.

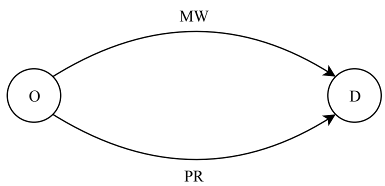


Figure 2.1: A binary network. Adapted from (Roughgarden, 2003)

Given are n drivers on a binary network with one origin (O) and one destination (D) connected through two roads: the broad, longer motorway, and a narrow but shorter provincial road. The costs to traverse each road is measured in travel time. Because of its breadth, the time needed to get from origin to destination via the motorway is always 60 minutes, regardless of the amount of traffic using it. Because the provincial road is narrow, it is load-dependent. This means the time needed to traverse the road is dependent on how many drivers use it. In this example, the travel time is linearly related to the share of the drivers using it, with a minimum of 30 minutes. If the road is empty, it only takes half an hour to arrive at the destination. If all drivers decide to use it, the provincial road takes 60 minutes. It follows that if only half of the drivers decide to take the provincial road, this would cost them 45 minutes (Roughgarden & Tardos, 2002).

Given the set of n drivers, two roads and one origin-destination pair, there are three main alternatives in which all drivers get from the origin to the destination: they all take the motorway, they all take the provincial road, or the drivers divide themselves over the two roads. It follows from the cost/time functions that if all drivers were to take the motorway, the average cost incurred by each driver would be 1. If all drivers were to take the provincial roads, the average individual cost would increase up to 1 as well. In terms of lowest average costs, the most efficient solution—the system optimum—would arise if the drivers were to split up, half of them using the motorway, and half of them using the provincial road. This way, half of the drivers would travel 60 minutes, and half of the drivers would travel 45 minutes, resulting in an average travel time of only 52,5 minutes, without any driver being worse off than in either one of the previous cases. However, this would not happen. After all, in that situation, each driver on the motorway can defect from this optimal solution, switching to the provincial road to save some time. As a result, all selfish drivers use the provincial road, and are unable to switch the other road to increase their personal efficiency: this is the user equilibrium as defined in the introduction[1]. This causes congestion on the motorway and the average travel time is higher than it could have been if the drivers were to cooperate (Helbing et al., 2005), or if some system-level intervention forced a share of the driver to take the motorway (Levy & Ben-Elia, 2016). This effect is not restricted to the given binary single-flow network; the effect occurs in complex networks with more nodes, edges and flows as well. (Roughgarden, 2003).

2.1.2 The Braess paradox

The Braess paradox (Braess, 1968) is another manifestation of the possible inefficiency of selfish routing. This paradox entails that "in unfavorable situations an extension of the road network may lead to increased travel times." (Braess, Nagurney, & Wakolbinger, 2005, p. 449). Figure 2.2 shows how. Given is a simple network on which a traffic flow travels from origin (O) to destination (D) (fig. a). Drivers can choose to travel via either vertex A or B. The costs (time) required to traverse each link is shown next to each link. X refers to the amount of traffic using the link. If ½ of the drivers use link O-A, this will cost them ½ (hour), for example. It is

[1]This numerical example is taken from an interview by the IT university of Copenhagen with Tim Roughgarden (IT University of Copenhagen, 2018, September 17)

simple to notice that in the user equilibrium, half of the drivers would travel via A, and half of the drivers would travel via B. This results in an average cost of 1½. This is also the optimal solution, since if any one driver would switch to another route, that route would increase in cost for all its users, and thus the average cost increases (Roughgarden & Tardos, 2002).

Then add an extra link A-C with cost 0 (fig. b). Intuitively, the optimal solution remains the same: there is no faster option for any driver to travel form O to D. However, now it is beneficial for an individual driver to drive to vertex A first, and then switch to vertex B, since initially, O-A and B-D have lower costs. As all individual drivers make this same decision, the links O-A and B-D get congested, resulting in a cost for 1 for all drivers. As a result, the average travel time becomes 2, with no possibility for any driver to choose a route with less costs (Roughgarden & Tardos, 2002).

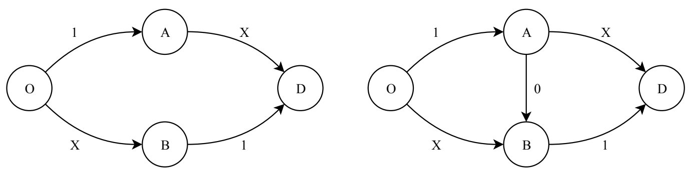


Figure 2.2: The Braess paradox. Adapted from (Roughgarden & Tardos, 2002)

2.2 The price of anarchy

In the example above, the given network allows for an average travel time of only 52,5 minutes. Yet, due to the selfish behaviour of the drivers, the network reaches an equilibrium in which the average travel time is 60 minutes. This implies that selfish, uncoordinated behaviour leads to an average efficiency loss of over 14%. The loss of efficiency due to uncoordinated behaviour is labelled the price of anarchy (PoA) (Papadimitriou, 2001). So how high can this price be? Roughgarden and Tardos (2002) prove that in principle, with limited assumptions on network cost functions (referred to as latency functions), the price can be “arbitrarily large”. However, when imposing restrictions to the type of latency functions allowed in the network, they find that the maximum PoA is bound, depending on the latency functions characterising the network. In a network with linear latency functions, in which the travel time on a link linearly increases with occupancy, like the one presented above, the maximum PoA is proven to be $\frac{4}{3}$, or 1,33. This means that the average travel time of the user equilibrium is $\frac{4}{3}$ of the average travel time in system optimum, meaning an efficiency loss of 33%. When a network is characterised with quadratic or cubic latency functions (see table 2.1 for examples), the PoA can increase up to a loss of 1,63 (63%) and 1,90 (90%), respectively (Roughgarden, 2003). The table below gives an overview of these three classes, typical functions and the PoA. A thorough discussion on the relation between the Price of Anarchy and more complex networks can be found in (Roughgarden, 2003).

Table 2.1: *Latency function classes and their price of anarchy. Adapted from (Roughgarden, 2003)*

Function class	Typical latency function	Maximum Price of Anarchy
Linear	$ax + b$	1,33
Quadtratic	$ax^2 + bx + c$	1,63
Cubic	$ax^3 + bx^2 + cx + d$	1,90

2.2.1 Real-world examples

The percentages mentioned above are maximum PoA values. Obviously, real-world networks, traffic flows and latency functions are more complex than the ones in the presented examples. This raises the question how well these figures translate to reality. Recently, authors have addressed this question (Zhang, Pourazarm, Cassandras, & Paschalidis, 2016b; Monnot, Benita, & Piliouras, 2017). The ability to gather, process and analyse large amounts of data facilitates empirical studies the performance of real-life networks. Using data on 33 thousand trips of nearly 16 thousand students, Monnot et al. (2017) demonstrate the inefficient use of the transport network in Singapore. They estimate a price of anarchy between 1,11 and 1,22 (11%-22%). A similar research was conducted by Zhang et al. (2016b; 2016a), who use data on average speed and road capacity of over 100 thousand road segments in Eastern Massachusetts. Over a 30-day period, they find an average PoA of 1,5, indicating an efficiency loss of 50% (2016a). This demonstrates that inefficient network use is not bound to theory, but occurs in real-life traffic networks as well.

2.3 How to solve for system optimum: a game theoretical approach

Given the potential impact of inefficient network use, one might wonder how to improve the workings of a network. How can a traffic system move from the inefficient user equilibrium to system optimum, or at least to a better performing system? Unfortunately, SO was not always seen as a realistic goal to achieve (Levy et al., 2018). In the examples mentioned in section 2.1, it becomes clear that a lack of cooperation and coordination leads to inefficient network use. If the agents were to cooperate, or if they were coordinated in an effective way, they would be able to achieve system optimum, and decrease their average driving time. Because of a lack of possibilities for agents to communicate with other agents, or to impose an optimal strategy on traffic participants, some authors considered SO as a mere theoretical state of a network (Adler & Blue, 2002; Roughgarden & Tardos, 2002; Levy et al., 2018).

Today, however, information technologies are capable of enabling communication between systems and the development of AVs is opening up opportunities for implementing system-level guidance. The quest for System Optimum is thus highly relevant (Levy et al., 2018). The following paragraph addresses some of the methods developed to overcome the efficiency problem. Before proceeding to these methods, however, it is useful to provide a framework in which these methods can be seen. This allows us to understand these solutions properly, and to assess their differences.

2.3.1 The route choice game

The efficiency problem is now viewed from the perspective of Game Theory. In game theoretical context, the drivers on the networks in the examples play a route choice game, in which they aim to maximise their own payoff (Helbing et al., 2005). In this game, a higher payoff corresponds to a lower travel time and thus two rational, optimising drivers (players) pick a route, in order to minimise their own travel time. Below, the example of paragraph 2.1.1 is displayed as a route choice game, in which the numbers represent travel time in minutes. In this game, the individual optimal choice for each agent is to choose the provincial road (whatever the other player picks, the provincial road is either better or equal to the motorway). Consequently, the game ends in both players choosing the provincial road. This is a Nash equilibrium: a result of a game in which no player can defect to get a higher individual payoff. Note that this is analogous to the definition of the user equilibrium given in the introduction. This Nash equilibrium, though, does not maximise the total payoff: total travel time (and thus average travel time) is lowest when the players take different roads. This means that the total payoff would be highest in either the top right or top left cell (Helbing et al., 2005). This coincides with the result of paragraph 2.1.1, in which system optimum was achieved when half of the drivers take the motorway, and half of the drivers take the provincial roads.

		Player 2	
		MW	PR
Player 1	MW	60 / 60	60 / 45
	PR	45 / 60	60 / 60

Figure 2.3: A route choice game. Source: author

So how can a game like this end up in the most efficient solution? Intuitively, this problem can be solved in two ways: by cooperation among players, or by changing the payoffs (Helbing et al., 2005). If the players were to cooperate in an iterated game, they could take turns in taking the motorway and the provincial way, and thus decrease their average driving time. If the payoffs are changed, the players could be persuaded to each take a different road. Both of these approaches are addressed below. For the sake of clarity, the following section will refer to agents and players. However, these could be translated to drivers in real-life situations.

2.3.2 Cooperation among players

In their work on the route choice game Helbing et al. (2005) argue that cooperation can lead to system optimum if players are to "take turns". They show in laboratory experiments with human participants that when the 2-player route choice game is repeated, players are likely to take turns in selecting the most beneficial road, thus effectively increasing their payoffs and decreasing the average travel time (although it was found that a cooperative strategy is not always stable: in some cases, players defected to the individually optimising strategy after finding the optimal strategy). In an agent-based model in which reinforcement learning 'taught' agents the optimal strategy, this behaviour emerges as well. However, it is noted that such cooperation "is unlikely to appear in real traffic systems" (Helbing et al., 2005, p. 24).

In both the laboratory experiment and the agent-based model cooperation only emerges after many iterations. Moreover, cooperation takes a longer time (a couple hundred of iterations) when the number of players is larger than 2. In their discussion on emergent cooperation Klein, Levy, and Ben-Elia (2018) conclude that:

"it is far more difficult for cooperation to emerge as group size increases [and] it seems that day-to-day learning, though necessary, is not sufficient to encourage players to cooperate in large numbers over a long period of time" (p. 185).

Therefore, in order to accelerate this learning process, Helbing et al. (2005) suggest using Advanced Traveller Information Systems (ATIS). An ATIS can help cooperation to emerge by suggesting system optimal routing advice to individual agents (Helbing et al., 2005). Klein et al. (2018) use an ATIS on a four-link, four-node network to near system optimum. Their system provides individual routing recommendations to 100 agents, which lead to system optimum if every agent would follow this advice. In this model earlier experiences are taken into account when an agent decides whether to follow the routing advice or not. The effect of following the recommended route on individual travel time determines whether an agent is prone to adhere to the advice again. It is important to note that in this approach, agents are still able to neglect the advice, and choose for another route instead. The agents are therefore enabled to remember the travel time for either complying with or neglecting the recommendation. This way, over time, agents learn that complying results in a lower average travel time (Klein et al., 2018). The authors stress that this effect is dependent on network design and the allocation of routing advice and that incentives—i.e. punishments and rewards—can help accelerate the emergence of system optimum. Even though this approach is effective when applied on the four-link, four-node network, Klein et al. (2018) acknowledge that when applying this solution to more complex networks, the contribution to system performance is harder to achieve, giving rise to additional, "harder" policies.

2.3.3 Changing the payoffs

Changing the payoffs can be such a harder policy. Rather than waiting for agents to discover cooperation by themselves or providing efficient routing advice, one could also change the costs agents incur when traversing a link. By changing the cost—by levying tolls, for example—in such a way that congested, overused links become more expensive to use, self-optimising agents are nudged to use a network in a more efficient way (Kaddoura & Nagel, 2019). This solution has first been proposed by Pigou since his identification of the efficiency problem (Pigou, 1920).

So how does this work? Let us revisit paragraph 2.1.1. If that situation was changed in such a way that drivers on the provincial road had to pay a toll, the outcome would be very different. Suppose that on the provincial road, a toll with a value equal to 15 travel minutes is levied. This means that for an individual driver, the value of the toll is equal to travelling 15 minutes shorter or longer. In this situation, a driver on the provincial road does not only incur costs in terms of travel time, but also in terms of a toll to be paid. Thus, the costs for traversing the provincial road is equal to the travel time plus 15 minutes. This implies that when half of the drivers is using the provincial road (leading to a travel time of 45 minutes), the costs for traversing this road is 60 minutes. For any additional driver, it would be beneficial to opt for the motorway instead. This results in a traffic distribution in which half of the drivers takes the provincial road, and the other half the motorway, thus leading to the optimal solution.[2]

Pigou notes that this approach can solve the inefficiency, given the height of the toll is "rightly chosen" (1920, p. 194). So how can a right toll be determined? Once again, consider paragraph 2.1.1. On the provincial road, additional traffic led to additional costs (travel time) to all users of the road. This means that if one extra driver decides to use provincial road, all other users experience a delay due to that one driver. The general idea behind system optimal tolling is that the individual driver not only carries the costs directly incurred when travelling on a road, but also the costs for the delay it causes to other drivers on the road. This cost is known as the marginal congestion cost. When a toll is equal to the marginal congestion cost, system optimum is achieved (Button, 2004).[3]

Here, a problem arises. Tolls imposed on links in a network can either be static or dynamic (Button, 2004). Static tolls remain stable over time ("flat tolls") and can solve the inefficiency problem in static, non-dynamic networks. Real-world networks and traffic flows are, obviously, diverse and dynamic: no two roads are identical, and the flow of traffic varies in density and destination within and between days. This means that the marginal congestion costs

[2]I.e., in terms of travel time. One can argue that due to the toll half of the drivers have to pay, the total welfare of the system does not change. Then again, the toll revenues can be redistributed among the system optimal drivers using the slower road to reverse this effect. This discussion, however, is a study on its own and beyond the scope of this thesis. For a discussion on the effects and possibilities of toll revenue distribution, see (Tsekeris & Voß, 2009).

[3]For an elaborated economical discussion on road pricing and marginal congestion cost, see (Button, 2004).

are spatially and temporally dynamic, giving rise to a need for a spatially and temporally dynamic toll (Button, 2004). This concept is known as micro-tolling (Sharon, Hanna, et al., 2017). Since determination of the marginal congestion costs of each individual driver would require future knowledge of all actions of all agents and the implications of these actions on the network, dynamically determining the marginal congestion cost in a real-world network is, seen as "unfeasible" (Sharon, Hanna, et al., 2017). Therefore, multiple tolling schemes with approximation methods for marginal costs have been proposed (Tsekeris & Voß, 2009).

In their discussion and overview of such methods, Sharon, Levin, et al. (2017) point out that all previously proposed methods have restricting assumptions which limit the practical implementation of each method. These methods assume, for example, fixed or fully known traffic demands and roadway capacity. These assumptions make a method unrealistic to implement, since full knowledge of drivers and networks might not be possible to attain, and real-work traffic does not consist of homogeneous drivers (Sharon, Levin, et al., 2017).

For this reason, Sharon, Hanna, et al. (2017) propose a novel tolling method: Δ-tolling (delta-tolling). This method is simple, effective and makes little assumptions. The fundamental workings of Δ-tolling are as follows. For each individual network link, a toll is calculated in the following way:

"Δ-tolling assigns a toll to each link proportional to the difference between its current travel time and its free-flow travel time (denoted Δ)" (Sharon, Hanna, et al., 2017, p. 828)

A parameter then defines the proportionality between the toll and Δ. As such, Δ-tolling only assumes complete knowledge of free-flow travel time and current travel time. Given that the method calculates a toll value for each individual link independently, the method is not only more practical to implement, but also well-suited to scale to a real-world network (Sharon, Hanna, et al., 2017). A similar method is proposed by Kaddoura and Nagel (2019), but rather than directly using Δ to compute a toll value, in their tolling method the toll on a link is updated with a set adjustment value if Δ surpasses a certain threshold. Using Δ directly to compute a toll value (as Δ-tolling does), however, has apparent benefits.

In their article introducing Δ-tolling, Sharon, Hanna, et al. (2017) demonstrate its effect on system performance. They prove that when applied on a macroscopic scale, Δ-tolling is equal to the marginal congestion cost and leads to system optimum, while when applied on a mesoscopic model, Δ-tolling leads to a decrease in average travel time up to 32%. The effect of

Δ-tolling on a microscopic model of an intersection was demonstrated as well (Sharon, Levin, et al., 2017). Mirzaei, Sharon, Boyles, Givargis, and Stone (2018) build further on this method by using a reinforcement learning algorithm to set the Δ-tolling parameters for each link individually. Compared to basic Δ-tolling, Mirzaei et al.'s enhanced Δ-tolling enables a decrease in average travel time of 28%.

2.3.4 Planning for the future: parameters and the past

Δ-tolling is thus a simple and effective method which enables a network to dynamically respond to changing traffic conditions. Such a tolling scheme, choosing the shortest route while responding to live traffic conditions, can be described as a "reactive" tolling scheme (Dong et al., 2011). Sharon, Hanna, et al. (2017) show that this responsiveness to changing traffic loads has one drawback: it can lead to a rapid rise or fall in tolls on specific links in the network, induced by a short but sudden change in traffic load traversing them. To illustrate this, consider paragraph 2.1.1. Assume Δ-tolling is applied in this case. Initially, all agents would opt to travel via the provincial road. This leads to a high travel time of 1 hour and a high toll. Since all agents respond to the same toll values, all agents switch to the motorway, which also takes 1 hour, but has an initially low toll value. This, in turn, increases the toll on the motorway, after which all agents switch back to the provincial road. As a result, all agents switch between the two alternatives, on both of which they experience a travel time of 1 hour. In this case, tolling does not have the desired effect (Sharon, Hanna, et al., 2017). This effect is referred to as *overreaction* (Ben-Akiva, De Palma, & Isam, 1991).

To counter these "spikes" in tolling and oscillation between routes, Δ-tolling uses a second parameter which gives weight to the toll in the previous time step. The calculation of the current toll then takes the height of previous toll into account, and thus toll values over time are levelled. The complete algorithm for Δ–tolling provided by Sharon, Hanna, et al. (2017) is then the following:

$$\Delta = t_e - T_e \tag{2.1}$$

$$\tau_e = R\beta(\Delta) + (1 - R)\tau_{e(t-1)} \tag{2.2}$$

In which t_e denotes the current travel time on link e, T_e denotes the free flow travel time and τ_e denotes the current toll value. $\tau_{e(t-1)}$ is the toll value calculated in the previous time step. β is the parameter which sets the proportionality between Δ and τ_e, and R is the responsiveness parameter to give weight to the previously set toll. It can be seen that a higher value of R leads to a higher "responsiveness": the toll will respond stronger to current travel time, and weaker to the toll in the previous time step (Sharon, Hanna, et al., 2017). R and β need to be set to fit the network being modelled (Sharon, Levin, et al., 2017).

Sharon, Hanna, et al. (2017) note that this "responsiveness parameter" R is required due to the lack of true knowledge on the future: if future network states and agent decisions were known, agents could respond to congestion in advance (Ben-Akiva et al., 1991). The potential advantage of knowledge on the future on the performance of Δ-tolling is acknowledged by Sharon, Levin, et al. (2017). Such knowledge allows a system to arrive in the most optimal equilibrium faster.

Sharon, Levin, et al. (2017) specifically note the possibility to predict recurrent congestion, but leave research on the effect of applying such predictions in a Δ-tolling tolling scheme for further research. Other authors, however, have examined the effect of predictions in other tolling schemes. Dong et al. (2011) recognize the potential for predictive data to improve the effect of a tolling scheme, and propose an "anticipatory" tolling scheme as opposed to a reactive tolling scheme. In a toll-setting method similar to Δ-tolling, their anticipatory tolling scheme not only takes current link occupancy to determine toll values, but also previous network states and earlier predicted toll values to predict future network states and toll values. It is shown that the anticipatory tolling scheme outperforms the reactive tolling scheme when congestion increases or decreases, leading to a shorter travel time (Dong et al., 2011).

2.3.5 Intention-based prediction

Besides using historical data as input for predictions, as done by Dong et al. (2011) (and to a lesser extent by Sharon, Hanna, et al. (2017) with introducing the R parameter, and by Kaddoura and Nagel (2019) with taking the previous toll as a starting point), predictions can also be made using route intentions of individual agents (Claes, Holvoet, & Weyns, 2011). First proposed by (Weyns, Holvoet, & Helleboogh, 2007), such predictive methods work in the following manner. It is assumed that each agent traversing the network has a travelling intention: at any point in time, an agent has a certain route it plans to follow. In intention-based predictive models, agents make "reservations" (Weyns et al., 2007): they share their intentions with a system, which in turn uses these shared intentions to predict future states of the network. This information can then be used by agents to recalculate their route, and change it if the predicted network state motivates them to do so. This process is then repeated so that the network prediction is continuously updated (Weyns et al., 2007). Claes (2015) describes the advantage of his method using intentions:

"[it] *does not extrapolate from the current traffic conditions in order to compute a traffic forecast. The information contained in the road users' intentions is not an extrapolation. It is actual information about the future state of the traffic network*" (2015, p. 45)

This concept can be implemented in an agent based model using a delegate multi-agent system (DMAS). In a DMAS, each agent sends out delegate agents, which traverse the network and travel an agent's intended route before the agent itself does that. As such, these delegates simulate the expected behaviour of the agent, and inform network sections or intersections along their way that "their" agent is coming past these points. Thus, network links are capable of predicting their occupancy, and inform agents about the time needed to traverse each link. This way, each agent can travel in such a way that it minimised its own travel time (Weyns et al., 2007). Following the initial work by Weyns et al., other authors have built on and advanced this method, applying it to a real-world simulation on the road network on the city of Leuven, Belgium, (Claes et al., 2011), defining design aspects required for implementation (Mahajan et al., 2019) and theoretically proving and exploring the workings of the method (Varga, 2014).

A similar approach is applied by Hashemi and Abdelghany (2016) and De Juncker et al. (2018). Rather than sending agent-specific delegates, these authors run a simulation of traffic on the network before assigning definitive routes to agents traversing the network. The travel times and occupancy found in this simulation can then be used to assign more efficient routes to agents. Even though intention-based prediction does not universally prevent inefficient routing in all cases (Varga, 2014), multiple authors demonstrate that applying intention-based prediction in agent based traffic models can lower average driving times (Weyns et al., 2007; Claes et al., 2011; De Juncker et al., 2018), Varga notes that: "*The technique of intention-propagation-based traffic forecast is* [...] an *important improvement to the simple naive online strategy* [i.e., a reactive strategy as described above]" (2014, p. 230). None of the researches mentioned above, however, is specifically aimed at nearing or achieving system optimum as opposed to user equilibrium. Indeed, it can lead to user equilibrium (De Juncker et al., 2018). Note that this does not prevent the method from being an improvement to a situation worse than user equilibrium.

2.4 Implications for this research

Several conclusions can be drawn from the theory above. A first conclusion is that in a traffic system, striving for user equilibrium or striving for system optimisation can have different implications. Reconsider the example in paragraph 2.1.1. In the user equilibrium, all drivers incurred a travel time of 1 hour, while in the system optimum, half of driver was better off with 45 minutes, lowering the average travel time to 52,5 minutes. It is clear that in the second situation, general welfare is higher: the system as a whole is better off. Individual welfare, however, is more unequal than in the user equilibrium: half of the drivers travels 33% longer than the other half. This implies that the choice between user equilibrium and system optimum is not only about general welfare, but also about equality. When considering a solution to reach system optimum, it is thus highly relevant to asses not only the impact of such a measure on performance of the traffic system, but also the impact on the distribution of travel times: is the distribution of shorter and longer travel times within an acceptable range (Levy & Ben-Elia, 2016)? Closely related to equality is the notion of fairness. While equality refers to the distribution of travel times, fairness refers to the impact of a system optimising measure on the travel time of an individual. Does the measure affect the travel time of all users in a similar way, or does it induce a much higher benefit or cost to some user than to others (Roughgarden, 2002)? Given these questions, solving for system optimum is not only a performance issue, but also a political issue—as is allowing the user equilibrium to emerge.

Some authors argue that user equilibrium is preferable because it optimises routes for individuals, while also providing an equal distribution of travel times: in user equilibrium, any two drivers travelling between the same origin and destination at the same time have the same travel time (Mahajan et al., 2019). Other authors argue the contrary: realising system optimum lowers average travel time and thus increase average welfare, while allowing for some inequality among drivers (Levy & Ben-Elia, 2016). This research goes with the second group, although using a slightly different argument. While the trade-off between equality and general welfare is a political debate, the contradiction between user equilibrium and system optimum is has an important implication. If, in given circumstances, the average travel time of a traffic system can be lowered (SO) when compared to a previous situation (UE), this means that in this previous situation, the traffic system was used inefficiently (Helbing et al., 2005). When considering growing urban areas worldwide and consequent congested urban networks, one can argue that inefficient use is undesirable (Adler & Blue, 2002; Klein et al., 2018).

A second conclusion which can be made is that after making this choice to strive for system optimum, one is left with multiple options to reach this goal. Even though the first identification of the problem by Pigou (1920) also included a proposed solution (tolling), the real-world examples of Boston and Singapore show that the problem has not yet been solved (Zhang et al., 2016b; Monnot et al., 2017). Indeed, the current rise of "smart cars" and communication devices available to the public have increased the relevance of research into new solutions (Sharon, Hanna, et al., 2017). Since inter-agent cooperation is unlikely to emerge in real-world systems or complex models (Helbing et al., 2005; Klein et al., 2018), tolling seems to be a more suitable approach to apply in actual networks. Especially methods which are easily scalable to large networks are promising to adopt in real life (Sharon, Hanna, et al., 2017). This research therefore applies the Δ-tolling approach as proposed by Sharon, Hanna, et al. (2017). As demonstrated above, tolling enables a traffic system to move from the user equilibrium towards system optimum. Theoretically, however, the workings of a tolling scheme could be improved when tolls are set based on predictive information rather than on current information (Ben-Akiva et al., 1991; Sharon, Hanna, et al., 2017; Mahajan et al., 2019). Even though earlier work has conducted research into the use of predictive information for individual routing, these researches were not aimed at achieving system optimum. It is therefore valuable to explore the effect of using intention-based predictions in a system-optimising tolling method like Δ-tolling (as proposed by, e.g., De Juncker et al. (2018)). Table 2.2 provides a schematic overview.

Table 2.2: *Traits of Δ-tolling and intention-based prediction. Source: author*

Concept	System optimising	Considering future
Δ-tolling	yes	no
Intention-based prediction	no	yes
Δ-tolling using intention-based prediction	yes	yes

A third conclusion which can be drawn is that to investigate the working of proposed solutions, multiple approaches can be taken. While some authors take a mathematical and theoretical approach to prove the workings of solutions (e.g., Button, 2004), others use models or simulations (e.g., Sharon, Hanna, et al., 2017). A theoretical approach allows for mathematically proving the effect of a measure. To study the implication of a measure on a dynamic traffic network with many drivers, however, modelling and simulation are more suitable approaches. Traffic models are mainly categorised in two groups: macroscopic models and microscopic models. Some authors define a third group: mesoscopic models (Chiu et al., 2011). Macroscopic models are also referred to as fluid-dynamic (Helbing, 2001) or hydrodynamic because similar to fluid-dynamic research, they model traffic like a flow of vehicles between an origin and a destination. As such, a macroscopic model does not discriminate between individual drivers, and monitors traffic in terms of average velocity, density and flow (Catalin, Dauphin-Tanguy, & Popescu, 2012). Although this method does not allow to model individual behaviour, it can be efficient in calculating traffic distribution on large networks (Sharon, Hanna, et al., 2017). A microscopic model (also referred to as a molecular model (Helbing, 2001)) models traffic as set of individual entities (drivers) with specific characteristics and behaviour. Although the modelling of each individual driver requires more computational power, the individual oriented approach allows for more detailed modelling and monitoring, because the position, velocity, route and other variables can be measured at the individual (micro-) level. More detailed behaviour such as reaction time, keeping distance to other vehicles and lane-changing can also be modelled in a microscopic model (Sharon, Hanna, et al., 2017). Mesoscopic models are similar to microscopic models in the sense that these model traffic as a set of individuals. However, they do not model behaviour in such a detailed way: concepts like right-of way at intersections, distance to other vehicles and lane-changing are not modelled (Chiu et al., 2011).

For this research, mesoscopic modelling seems to be the best suitable approach. Although Δ-tolling can be applied in a macroscopic model (as shown by (Sharon, Hanna, et al., 2017)), intention-based prediction requires individual-level knowledge on the location and route intentions of each driver. Since small-scale vehicle behaviour like mentioned above is outside the scope of this thesis, a true microscopic model is not required.

Individual level modelling is often executed using an agent-based model (ABM), or multi-agent system (MAS). Such a model consists of individual agents who follow a predefined set of rules. All agents act on these rules while responding to their environment and to other agents, and thus from this individual behaviour, a macro-scale pattern can be modelled (Ehlert & Rothkrantz, 2001). As such, an ABM is suitable to study complex, system wide patterns which result from relatively simple individual rules. The possibility to study patterns emerging on system-level and to control and monitor behaviour on agent-level makes ABM a well suited modelling approach for this research.

Following the conclusions made above, this study uses an agent-based-model, in which tolling and intention-based prediction are used to improve system performance. The following chapter explains how this is done, and how this study aims to achieve the general research objective introduced in paragraph 1.2.

Chapter 3

Methods

Building on the insights of the previous chapter, the following chapter provides an overview of the methods applied in this study. It first provides a general outline of the steps taken in this study. It then describes each of these steps in detail by providing a conceptual model, an implemented model and experiment settings. The chapter closes with describing the verification process, sensitivity analyses and scenario analyses.

3.1 General approach

The objective of this research is *to develop and assess an agent-based traffic model (ABM) in which agents are supplied with multi-temporal network data to improve insight in the effects of predictive information on system performance, specifically when compared to user equilibrium.*

The study aims to reach this objective by answering the following research questions:

- **RQ 1** What conceptual model using multi-temporal data can be used to improve the performance of a traffic system?
- **RQ 2** How can this conceptual model be implemented in an agent based traffic model?
- **RQ 3** To what extent is the performance of the traffic system sensitive to different scenarios?
- **RQ 4** What is the impact of the different scenarios on the equality and fairness of a traffic system when compared to user equilibrium?

Figure 3.1 provides an overview of the general approach of this study. After finishing the first step of conducting a theoretical research, this research proceeds with creating a conceptual model (RQ1), which explains the general workings of the ABM. Then, this conceptual model is implemented in a functioning agent-based model (RQ2). As a fourth step, the experimental setup of this study is determined after which the model is verified. Subsequently, the sensitivity of model output to parameter settings is tested and the fairness and equality of different scenarios is explored (RQ 3 and 4). Finally, chapter 5 draws conclusions from the results and answers the research questions, after which chapter 6 discusses the study.

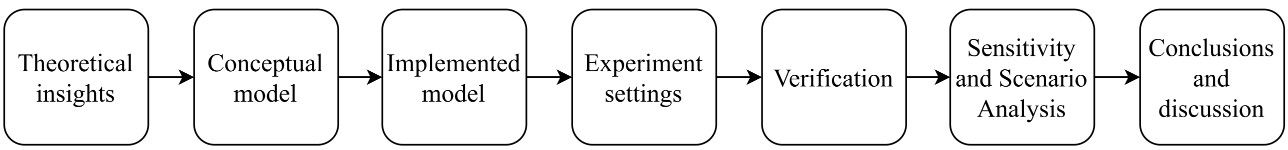


Figure 3.1: The general workflow of this study

In order to describe the conceptual and implemented models, the ODD-protocol is used as a guideline (see Grimm et al., 2006, 2010). By separating model overview, description and details, this protocol separates the verbal description of a model from its mathematical implementation, and thus serves as a guide for authors and readers to present and explain ABM's in a clear and consequent manner (Grimm et al., 2006). Note that this study does not strictly adhere to the protocol, or mentions its components explicitly. Rather, it implicitly follows the structure of the protocol to provide a structured overview of the model of this study. For a thorough explanation of the protocol, see (Grimm et al., 2010).

After describing the model, this chapter provides the experiment settings, and then explains how the model is verified and how model sensitivity and performance are analysed. The outcome of these analyses are provided in chapter 4.

One might have noticed that model validation and calibration are not mentioned in the trajectory sketched above. This is because the scope of this research is aimed at understanding processes, impact and side-effects of multi-temporal network data in a traffic system, rather than accurately modelling the impact of such measures on an actual traffic network. Therefore, the absolute system performance is of smaller relevance. As such, calibration and validation in the sense of assessing the model's fitness to reality (De Smith, Goodchild, & Longley, 2018) is not relevant in this study.

3.2 Conceptual model

The conceptual model takes in consideration insights from the theory presented in chapter 2, together with assumptions made and limitations set in this study. The conceptual model explains the inputs and outputs of the eventual model, and the basic mechanisms. It does so by explaining the model's purpose, agents, variables, processes and assumptions. Before these elements are explained, however, the following paragraph first defines the model background.

3.2.1 Model background

The model in this study is largely based on two main concepts as dealt with in chapter 2. The first concept is the concept of Δ-tolling as introduced by Sharon, Hanna, et al. (2017). In this study, is used as a tolling method to increase system performance. The second relevant concept is the concept of intention-based prediction, as first introduced by Weyns et al. (2007). Agents in the model use thus share their intentions, which are then used to predict network occupancy. This prediction is then used to adapt individual routing decisions.

The following model characteristic is important to note before addressing the model design. Note that the model in this study is mesoscopic. This means that it is similar to a microscopic model in the sense that it handles traffic as individual entities (agents). Contrary to macroscopic models, microscopic models also allow to model traffic situation and the behaviour of drivers on a detailed level. For example, right-of way at intersections, distance to other vehicles and lane-changing could all be incorporated in a microscopic model. As this research merely aims to evaluate the effect of a varying supply of multi-temporal data, however, such details are not strictly necessary, and a mesoscopic model is applied. Although more details would make the model more realistic, this is left for further research.

3.2.2 Model Purpose

The purpose of the model is to analyse the effect of multi-temporal information on the performance of a traffic system, in particular the effect of system-optimising tolling and intention-based predictions on the average travel time of agents when compared to the user equilibrium. In line with (Sharon, Levin, et al., 2017), system performance is measured in (average) travel time, and does not consider toll expenses or revenue: it is assumed that all toll payments are effectively reinvested in society, and thus do not harm general welfare.

3.2.3 Model agents and variables

The model has three entities: drivers, network links and nodes. Drivers travel from an origin to a destination while moving on network links. Links set their current cost value based on their length, speed limit, current occupancy and a toll value determined in the previous time step. The link uses an occupancy matrix to register predicted occupancy. This occupancy matrix has 1-minute timeslots. Nodes are the points where network links meet or intersect, and have the following state variables: node ID, link IDs. Drivers always spawn at and drive to nodes. The agents and their variables are shown in table 3.1.

The model is executed on three networks: two hypothetical networks displaying the binary road example (network A) and Braess example (network B) mentioned in 2.1, and on the more Sioux Falls road network. Details on these networks are provided in section 3.3.2. When using the hypothetical networks, one time step equals 5 in-model seconds, and the model is ran for 10.000 steps. In the Sioux Falls network, one time step represents 10 seconds, and the model is ran for 4.000 steps, approximately representing 11 in-model hours.

Table 3.1: *Model entities and variables*

Entity	Variables
Driver	• Origin node • Destination node • Speed • Direction • Planned route • Travel time
Link	• Length • Free flow travel time • Current occupancy • Latency Function • Current speed • Current travel time • Current toll • Current cost • Occupancy matrix
Node	• Node ID

3.2.4 Model processes

Figure 3.2 displays a conceptual model of the model in this study. At the start of each day, all drivers are located at their origin. The drivers have the goal to drive from their origin to their destination on the given network. The network links have a cost.

Drivers choose a route with the lowest possible aggregate cost, and inform the network links along their path about their time of arrival. The drive along the selected route. Every time a driver reaches a node, it recalculates its route. If a driver is "smart", it uses the information each link has on their occupancy at any given time, reassesses the network and selects the shortest route. It then cancels the notification on the links of its previous intended route, and notifies the links on the new intended route about its time of arrival (ToA) and time of departure on each link. This process repeats until the driver has reached its destination. If a driver is not "smart", it reassesses its route at every node, but does not take predictive information into account.

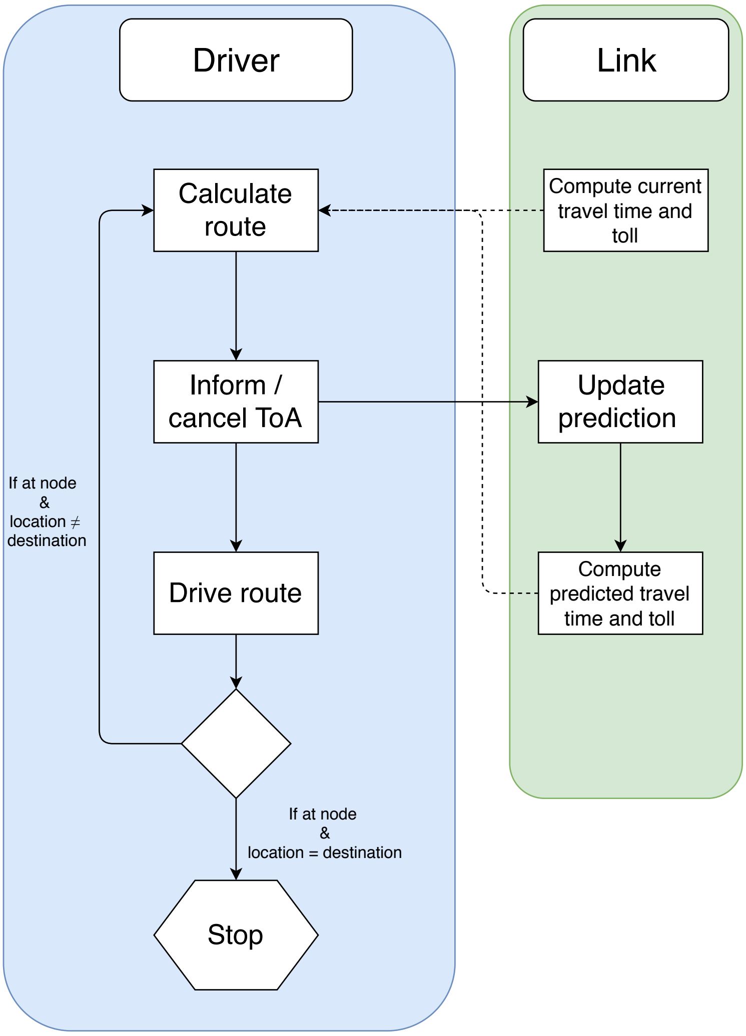


Figure 3.2: Conceptual model

Every time step, each network **link** computes a current toll value and a current travel time. The toll value is based on current occupancy and the toll in the previous time step. The current travel time is determined by the free flow travel time, the link's latency function and the current number of drivers on the link. The link uses the toll value and the current travel time to calculate the current cost for traversing it. Every time the link is informed on the arrival of a driver, it updates its occupancy matrix at the informed time of arrival. Drivers can use both the current travel time and toll as the predicted occupancy to calculate their shortest route. This entire process repeats until all drivers have arrived at their destination.

3.2.5 Assumptions in the model

As explained in paragraph 1.2.1, this research does not aim to create a model which represents an actual traffic situation as accurately as possible. It rather aims to test the workings of a proposed solution. This allows us to make a set of limiting assumptions. Although these assumptions render the model less realistic, they also allow for a simpler and easier to understand model. As such, the effect of changes in information available to agents or in behavioural rules can be understood better.

With this in mind, we make the following assumptions in the model:

- The traffic system consists only of a road network and drivers. No other traffic is included.
- Each link has a known, constant "free flow travel time" and "free flow speed" when there is no traffic on it.
- Each link knows, at any time, how many drivers are on it.
- Each link has a "current travel time" and a "current speed". The links are load-dependent. This means that the possible speed on a link and the time required to traverse a link is related the number of drivers on it. The load dependency is characterised by link-dependent latency functions.
- The load-dependency and free flow travel time varies between links.
- Each link keeps track of its occupancy at any time, based on the communicated travelling intentions of drivers.
- The links may charge an additional toll value, besides the time required to traverse a link.
- The cost of a link is determined by the time required to traverse a link and any additional toll value.
- The drivers are agents traversing the network. Each driver has a given set of origin and destination, and a given time of departure.

- The drivers aim to minimise their costs while doing travelling: they try to find the shortest route in terms of travel time and cost.
- All drivers value time equally
- A driver can be “smart”: equipped with a fully autonomous car with navigation and communication devices capable of instant communication with network links and prediction of future network occupancy.
- The drivers have complete knowledge of the current travel time and toll on each link.
- The drivers’ speed and travel time on a link is determined by the link they are on, according to the link’s current speed and current travel time.
- The drivers are fully rational and will always act in the best possible way given the information and aims they have.
- The drivers are entirely compliant and always trust the information given to them through their navigation and communication devices.
- The drivers can pass through one another.
- The model is aimed at decreasing (average) travel time, and does not consider toll expenses or revenue: it is assumed that all toll

3.3 Implemented model

After describing the conceptual model, this section provides insight into how these concepts are implemented in an agent based model. After introducing the software that this study uses, this section describes this study’s input data. Finally, it describes the detailed processes, rules and algorithms which define the model.

3.3.1 Software choice

The model is built using the GAMA platform. The GAMA (GIS Agent-Based Modelling Architecture) platform is an agent-based modelling platform which allows the user to develop complex, large scale models. GAMA offers the possibility to provide geographical information in commonly used dataformats, like Shapefile, CSV and others. This makes GAMA suitable to apply in modelling with a geospatial component. GAMA uses its own agent-oriented modelling language (GAML). This implies that all components of a model can be agents which follow certain rules. Even though this study does not directly create a large-scale model, the GAMA platform can be used to expand the model, using data on real-world networks (Taillandier et al., 2019).

A complete overview of the workings of GAMA can be found in (Taillandier et al., 2019), but the following foundations of the software are important to note. A model in GAMA consists of three parts: a global section, definition of species (agents) and an experiment plan. In the global section, the global components of the model are defined, like the number of agents, or the input data for the environment. In the definition of the species, all agents and their behaviour are defined. This does not only include moving agents, but also non-moving agents, like a network or buildings. In the experiment section, the settings for the simulation are defined. It defines the format of the output and the values of any parameters (Taillandier et al., 2019).

3.3.2 Model input data: networks and traffic flows

Hypothetical networks The two hypothetical networks display the binary road example (network A) and Braess example (network B). These networks are mainly used to test and display the workings of the method. On these one-directional networks, behaviour leading to user equilibrium and system optimum is known, which offers the possibility to examine the effectiveness of the model. The networks and the link lengths are displayed in figure 3.3 and figure 3.4. The networks assume a free-flow speed of 120 km/h, or 33,33 m/s. The drivers traversing the networks are spawned from node A at a one-minute interval. Each driver has the same destination (node F in network A, node E in network B). Networks A and B are provided as a Shapefile, created manually by the author using ArcGIS Pro 2.4.0.

Sioux Falls network The Sioux Falls network (figure 3.5) is more complex. This simplified road network of the city of Sioux Falls (USA) was introduced to traffic modelling by LeBlanc, Morlok, and Pierskalla (1975) and has since been used in many other traffic modelling studies (see, e.g., Sharon, Hanna, et al., 2017; Bar-Gera, Hellman, & Patriksson, 2013). Although its 76 links and 24 nodes make the Sioux Falls network a more complex one than the two hypothetical networks, it is still simplified. However, it does allow us to test the workings of the proposed methods on a network with a larger diversity of routes and origin-destination combinations.

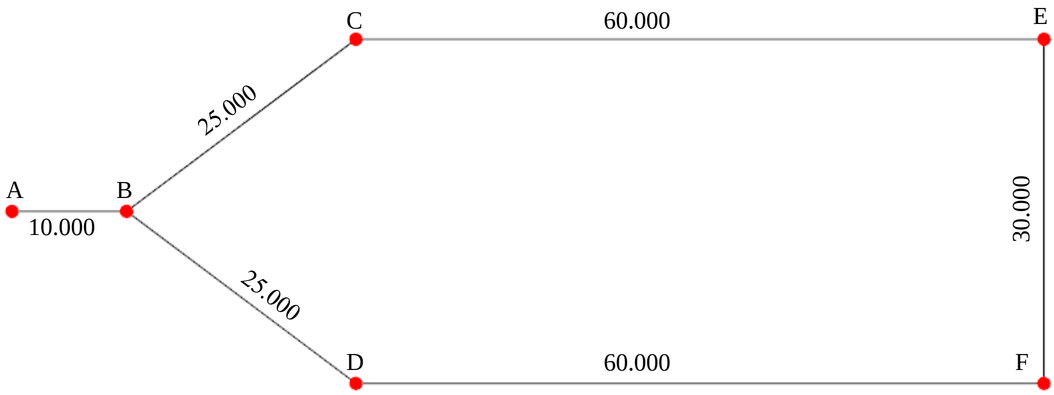


Figure 3.3: Network A with link lengths in meters

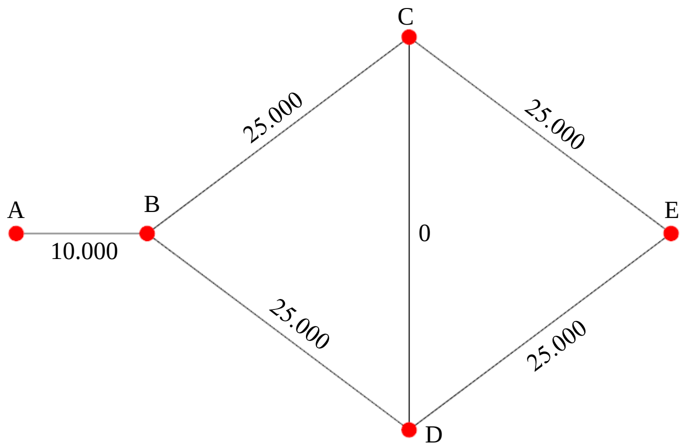


Figure 3.4: Network B with link lengths in meters

The network data used in this study, created by the *Transportation Networks for Research Core Team* (2020), is the same data as used in Sharon, Hanna, et al. (2017). The dataset provides the links and nodes in the network, accompanied by additional link-specific data such as X and Y coordinates, link travel time, length and capacity. The dataset does not include speed limits. This data has been converted to a Shapefile by the author using ArcGIS 2.4.0. Because the original dataset is not geographically correct– i.e., the ratio link length/travel time does not correspond between links–and GAMA uses geographical information as input, the coordinates were modified and speed limits were computed in order to fit the link lengths and travel times as provided in the dataset. The complete original dataset, the modified dataset and the method of modification can be found in appendix A.

The traffic flow on the Sioux Falls network is also provided by Transportation Networks for Research Core Team (2020). The dataset provides an overview of travel demand between all possible OD pairson one particular day. This overview represents a daily travel demand. Since this study does not run a model on a daily basis, but on a 10-second basis, this demand is converted to demand per second. This demand, however, leads to many agents to be modelled at the same time, hence drastically slowing down the model. This per-second demand is therefore divided by 400. Since this still leads to a slow running model, a selection of the 50 OD-pairs with the highest demand has been made. The original OD-travel demand data and the travel demand data used in this study can both be found in appendix B. The dataset also provides latency functions, which are dealt with in the following paragraph.

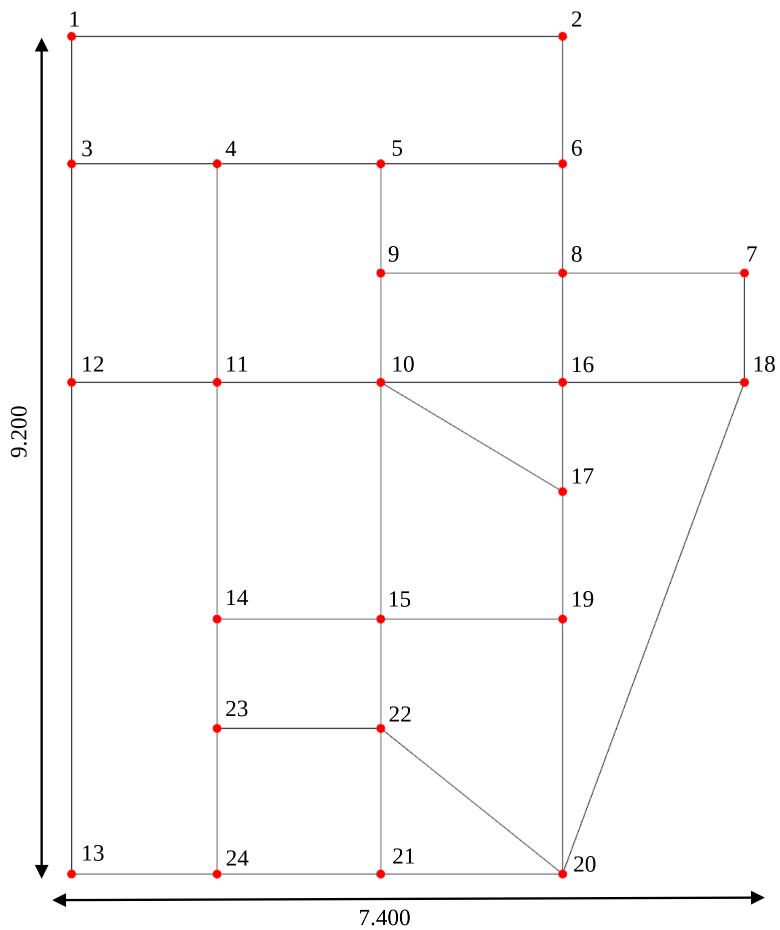


Figure 3.5: The Sioux Falls network with node IDs and network size in meters

3.3.3 Computation of tolls, travel times, speed and cost

As mentioned before, this model makes use of Δ-tolling. The formula for Δ-tolling is the following, after (Sharon, Hanna, et al., 2017):

$$\Delta = t_e - T_e \tag{3.1}$$

$$\tau_e = R\beta(\Delta) + (1 - R)\tau_{e(t-1)} \tag{3.2}$$

In which t_e denotes the current travel time on link e and T_e denotes the free flow travel time. τ_e denotes the new toll value to be computed. $\tau_{e(t-1)}$ is the toll value calculated in the previous time step. β is the parameter which sets the proportionality between Δ and τ_e, and R is the responsiveness parameter to give weight to the currently set toll (Sharon, Hanna, et al., 2017). Both β and R need to be adjusted to fit the network at hand (Sharon, Levin, et al., 2017).

This formula requires knowledge on current travel time and the free flow travel time. In this study, the free flow travel time is computed by dividing the link length by the links speed limit. The current travel time can be computed using the current occupancy on a link, according to a link's latency function. On network A and B, the latency functions are set as linear latency functions. They have the following form, after Roughgarden (2003):

$$current\, travel\, time = a * x + b \tag{3.3}$$

In which t is the link's current travel time, x denotes the occupancy of the link, and a is a parameter which sets the impact of one unit of traffic on the travel time of a link. So how high should a be?

Both network A and B are configured in such a way that with this traffic flow, they recreate the examples as given in section 2.1. For network A, this means that the bottom route (DF) is load-dependent, whereas all other routes are not. In Pigou's example, with no interventions, all traffic takes the bottom route. This route then becomes so congested that it slows down, resulting in equal travel times on both routes. What does this mean for model settings? Given is the speed limit on al links in network A of 33,33 m/s. Pigou's example plays out when the travel time on section CEF is equal to the travel time on section DF. Since the top route (CE+EF) is 1.5 times as long as the bottom route (DF), this implies a maximum increase in travel time on DF of 50%. This requires a reduced speed of 1/1.5 * 33,33 = 22,22 m/s. This increased travel time and speed should occur when the road is fully congested. So when is it fully congested? At a 1-minute interval, drivers driving at a speed of 22,22 m/s are 60*22,22 =

1.333 meters apart. Given the length of link DF (60.000 meters), this results in 45 drivers on the link at the most congested moment. Because these 45 drivers cause a total delay of 50%, the delay caused by one driver is 1.1 per cent. Thus, in network A, a is 1.1% or 0,011, for link DF. Other links do not delay.

On network B, a slightly different logic applies. Since this network is an example of the Braess paradox, links BC and DE are load-dependent, and can delay up to 100%, doubling the travel time on the links. With no interventions, all traffic should follow the route ABCDE. Since the lengths of sections BC, BD, CE and DE are equal, this implies that the free-flow travel time om BD and CE should be twice as long as the free-flow travel time on BC and DE. It is therefore assumed that the speed limit is 33,33 m/s on BD and CE, and 66,66 m/s on BC and DE. When fully congested, BC and DE delay up to 100%, returning a speed of 33,33 m/s. Given the lengths of the links (25.000m) and a 1-minute interval between drivers driving at 33,33 m/s, a fully congested link has at least 12 drivers on it. As such, the delay caused by one driver is equal to 8.3%. Thus, $a = 8.3\%$ or 0,083 for links BC and DE on network B. Other links do not delay.

The latency functions in the Sioux Falls network are non-linear. Rather, these latency functions have been derived from the Sioux Falls dataset (Transportation Networks for Research Core Team, 2020). Conforming LeBlanc et al. (1975), the dataset provides the latency functions determining a link's current travel time in the following format (Transportation Networks for Research Core Team, 2020):

$$current\,travel\,time = free\,flow\,travel\,time + b * flow^4 \tag{3.4}$$

In which parameter b can be used to tune the function to fit the link at hand. Parameter b is defined in the following way:

$$b = \frac{free\,flow\,travel\,time}{B * (capacity^{power})} \tag{3.5}$$

In which B is a parameter to fit the function to specific traffic situations. This parameter stems from the road travel time function as created by the American Bureau of Public Roads (therefore commonly known as the BPR function) (Maerivoet & De Moor, 2005), and is commonly assumed to be 0,15 (Sharon, Hanna, et al., 2017). Capacity is a link-specific characteristic provided in the dataset and power is assumed to be equal to 4 (Transportation Networks for Research Core Team, 2020).

Since the travel time in function 3.4 is defined for the original demand data, the flow in the function has to be adapted in order to compensate for the modified, lower flow used in this study. In the original dataset, the set capacity leads to a total flow on the network of 877.603 drivers. In this model, the flow will be adapted proportionally to the number of drivers on the network. The link latency functions then compute a link's current travel time in take the following way:

$$current\,travel\,time = free\,flow\,travel\,time + b * ((877.603/drivers\,on\,network) * flow)^4 \quad (3.6)$$

In both the hypothetical networks as in the Sioux Falls network, the computed travel time can be used to determine the current speed on a link:

$$current\ speed = \frac{link\ length}{current\ travel\ time} \quad (3.7)$$

Δ-tolling derives the toll value directly from time values, and weighs these with parameter β. As such, the outcome of function 2 can directly by added to the current travel time of a link to compute the total cost:

$$current\ cost = current\ travel\ time + current\ toll \quad (3.8)$$

Because the computations of the tolls, travel times and speed can be computed using only a link's free flow travel time and its occupancy, this method is easily applicable in predicting values as well. In that case, the flow is replaced with the predicted occupancy at the relevant time. The final considered costs of a link are then computed as a combination of current cost predicted costs, weighted by weighing parameter α as follows:

$$final\ link\ cost = (1 - \alpha) * current\ costs + \alpha * predicted\ costs \quad (3.9)$$

By varying α between 0 and 1, the final link cost gives a higher weight to either current or predicted costs.

3.3.4 Rules and algorithms

Driver algorithms

The agents each have their own set of rules—or algorithm—to follow. Note that both the driver-algorithm and the link-algorithm do not display actual implemented code, but only the rules that the agents follow. The algorithms for the driver are presented below.

The algorithms apply the following logic. If a driver is at a node which is not its destination, it (re)calculates its route (algorithm 1). If it is not at a node, it continues along its route. If it is at is destination, the driver stops (algorithm 4).

Algorithm 1 (figure 3.6) computes the path of the agent. For explanatory purposes, the algorithm has been divided in three levels: the network level, the path level, and the link level. When calculating the route, the driver first stores its current path as old path (line 1). In order to prevent the agents from assessing all possible paths in the network, it then selects the k-shortest paths based on current costs in the network (2). In this study, k = 10. It examines the current and predicted costs of each of these k paths (3), and selects the path with the lowest costs (4).

When examining each path (path level), the driver first notes the current model time (in minutes) as its time of departure from its current location (1). It also creates a variable which stores the final cost of the currently examined path (2). It then examines all links in the path to update the final path cost (3), and it then returns the final path cost (4).

At the link level, the driver first sets its time of arrival at the link equal to its last known time of departure (1). It then requests the predicted link occupancy at the time of arrival (2). It uses the link occupancy to compute the travel time and toll (3). It updates the final path cost with the current and predicted travel time and toll, using α as a weighing factor (4), and lastly updates the time of departure by adding the computed travel time to the time of arrival (5).

After this algorithm has ran, all links in all k-shortest paths have been evaluated, and the least cost path has been selected.

The agent then proceeds by cancelling and updating its presence at links (algorithms 2 and 3). Note that the driver first cancels its reservations. This is because when the driver makes reservations on a path, it creates a new arrival schedule, which is a list that contains the time of arrival and time of departure at every link along the path. To cancel reservations, the old arrival schedule is required, so cancellation has to take place before a new arrival schedule is created by algorithm 3.

Algorithm 2 (figure 3.7) cancels reservations. The driver first sets an index at 0 (1), and then updates all links in the old path (2). When it is finished, it clears the arrival schedule so it can be updated by algorithm 3.

When updating the first link (link level), the agent first looks up the stored time range at

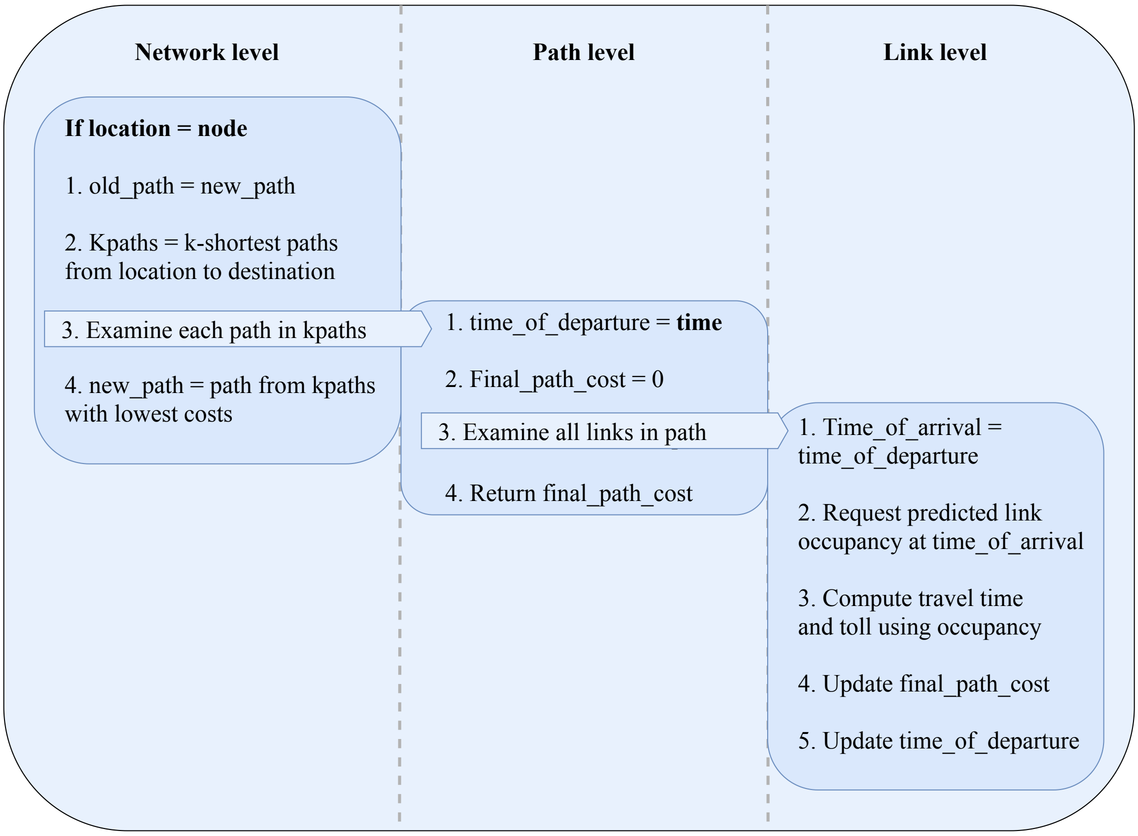


Figure 3.6: Algorithm 1: route planning

which it will arrive and leave at the link (1). This can be found at the first index (i.e., 0). Over this time range, it decreases the occupancy in the links occupancy matrix with 1 (2). It then adds 1 to the index (3), so it can assess the second link in the path.

When making reservations (algorithm 3, shown in figure 3.8), the agent first sets its time of departure equal to the current model time in minutes (1). It then updates all links in the new path (2).

When updating the links along the path (link level), the driver first sets the time of arrival at the link equal to the last known time of departure (1). It then updates the time of departure by adding the link travel time to the time of arrival (2). It saves the range from the time of arrival through the time of departure as a variable (3), and it adds this variable to its arrival schedule (4). Finally, it increases the occupancy of the link at the time range with 1 (5). After these steps, the links in the network have been updated on the drivers' intentions.

Finally, in algorithm 4, the driver either proceeds or start to drive along its planned route (1), or it stops and cancels its reservations if it has arrived at its destination (2 and 3).

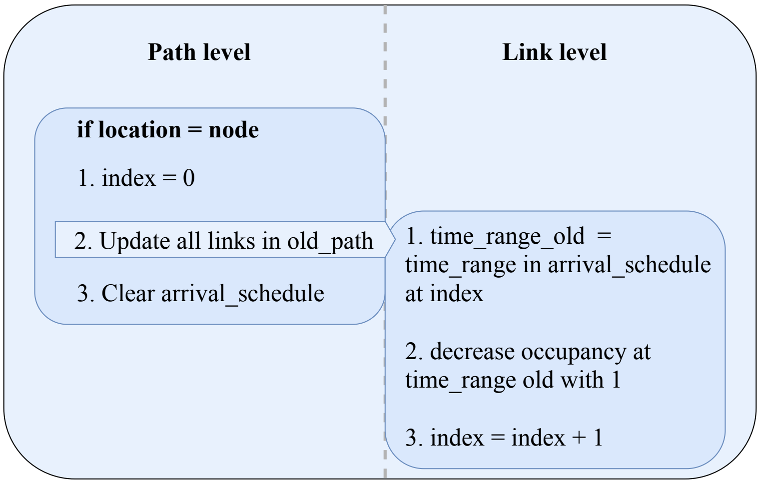


Figure 3.7: Algorithm 2: cancellation of reservations

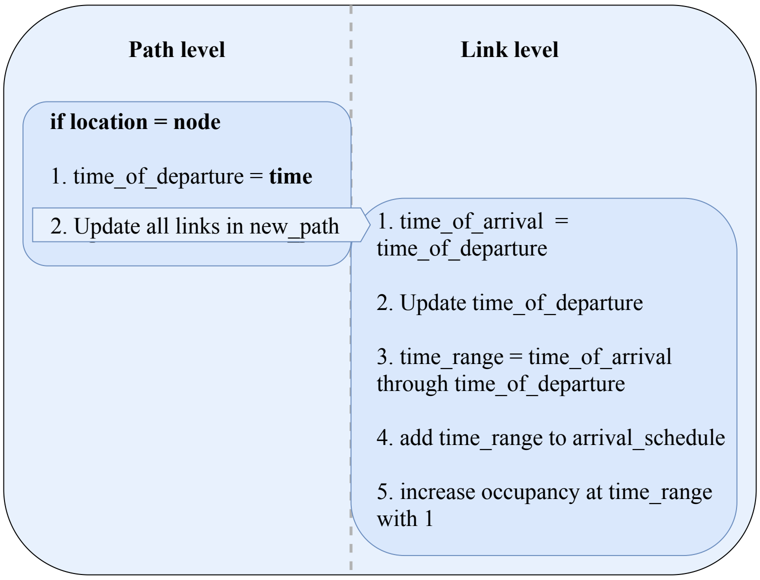


Figure 3.8: Algorithm 3: making reservations

Figure 3.9: Algorithm 4: driving and stopping

Link Algorithm

The algorithm for the links is presented below. The links executes this algorithm every time step. It follows these steps. The link first retrieves its current occupancy (1). With this occupancy, the current travel time and speed are computed (2 and 3). The toll is computed using the current and free flow travel time and the previous toll (4). Then the cost of the link is computed by adding the toll and travel time (5). Further actions by the link (updating it occupancy matrix) have already been dealt with by algorithms 2 and 3.

1. get current_occupancy
2. compute current_travel_time
3. compute current_speed
4. compute current_toll
5. compute current_cost

Figure 3.10: Algorithm 5: link algorithm

3.4 Experiment settings

This research aims to gain insight in the effects of predictive information on system performance, specifically when compared to user equilibrium. This implies two things. First, to assess the effect on system performance, knowledge on the user equilibrium is required. This can then be compared to a method aimed at system optimisation. Second, to assess the effect of predictive information, knowledge on a non-predictive, system optimising model is required. This can then be compared to a method applying predictive information. This study therefore examines four scenarios. All scenarios are tested on the hypothetical network, and performed on the Sioux Falls network. Besides the settings of the parameters, the scenarios do not differ: the networks are the same over different runs, as is the traffic flow (section 3.3.2). The scenarios are:

1. Not aimed at system optimum, not predictive
2. Not aimed at system optimum, predictive
3. Aimed at system optimum, not predictive
4. Aimed at system optimum, predictive

Scenario 1: not aimed at system optimum, not predictive This is the scenario "as-is": there are no intervening measures imposed on either driver behaviour on the network. To prevent the model from aiming at system optimum, parameters β and R must be set equal to 0. This results in a Δ-toll of 0, and thus only individually incurred travel time is considered by drivers choosing their shortest path. Also, to prevent the drivers from using predictions, α is set to 0.

Scenario 2: not aimed at system optimum, predictive To test the effects of prediction, parameter α is increased from 0 to 1 with intervals of 0,1. β and R are set to 0. For the value of α where the average travel time is minimum, the model also tests the influence of varying shares of "smart" agents which use predictive information. The share of smart agents is varied from 0 to 1, with intervals of 0,2. When the share of smart agents is lower than 1, it is stochastically determined which agents are smart, and which are not, according to a probability which is equal to the share of smart agents. Therefore, when the model tests the influence of varying shares of smart agents, it takes the average outcome of three runs per parameter setting.

Since this model enables drivers to optimise their individual route with complete predictive knowledge on the network, this study assumes that the outcome of scenario 2 with the lowest average travel time is the user equilibrium. Note that this is not a proven user equilibrium. However, the ability to accurately predict future states of the network does enable a faster emergence of user equilibrium (2017). Moreover, since this study aims to explore the effects of tolling and prediction, this scenario suffices to provide a "user equilibrium-benchmark" to use in scenario 3 and 4.

Scenario 3: aimed at system optimum, not predictive This scenario explores the effects of Δ-tolling. In their study on the effect of Δ-tolling on system performance, Sharon, Hanna, et al. (2017) find that combinations of β and R define the effect of Δ-tolling rather than specific β or R values. Therefore, this study tests different β and R combinations rather than testing the parameters in isolation. In theory, parameter β has no limits. However, Sharon, Hanna, et al. (2017) do not find very different results when using β values twice as large or small than the β value (β =4) with which they found their optimal result. Therefore, this study examines a limited range of β values (β=0 to β=10 with intervals of 0,5). Sharon, Hanna, et al. (2017) test R with values varying between R=10^0 to R=10^{-10} and find their most optimal results with R values around R = 10^{-4}. This study therefore tests R as R= 10^{-x}, with x ranging from 0 to

5 with an interval of 1,0, and an interval of 0,5 between 2 and 5. In this scenario too, α is set to 0.

Scenario 4: aimed at system optimum, predictive To explore the workings of prediction on networks with Δ-tolling, this scenario uses the same settings for parameter β and R as scenario 2. To limit the amount of runs required, parameter α is increased from 0 to 1 with intervals of 0,2. These implications result in the experiment settings as shown in table 3.2.

Table 3.2: *Parameter settings for each scenario. Numbers in parentheses indicate step size*

Sc.	Aimed at SO	Predictive	β	R	α	Smart share
1.	no	no	0	0	0	0
2.	no	yes	0	0	0-1 (0,1)	1, 0-1 (0,2) for best α
3.	yes	no	0-10 (0,5)	0, 1 & 2-5 (0,5)	0	0
4.	yes	yes	0-10 (0,5)	0, 1 & 2-5 (0,5)	0-1 (0,2)	1

3.5 Model verification

In this stage, it will be assured that the implemented works as intended (De Smith et al., 2018). Especially the two hypothetical networks are well suited for this task, since the expected behaviour of the agents on these networks can be predicted relatively simple. During the verification process, the following aspects of the model will be verified:

1. The links. It is checked whether a link adequately counts drivers present on the link, and whether its variables are computed correctly. It is also assessed whether a link updates its occupancy matrix correctly.

2. The driver. It is checked whether the driver indeed takes the shortest route to its destination when no interventions are implemented, whether it drives with the correct speed, and whether the implemented interventions have the expected effect. Does a driver indeed assess multiple paths, and communicate its plans with the links?

3. Model behaviour. It is checked whether drivers' behaviour creates a pattern which is expected. Consider network A. In this network, it is known that with no interventions, all traffic takes the bottom road. It also known that when Δ-tolling is applied, half of the traffic should take the bottom route, and half of the traffic should take the top route. When prediction is applied, agents in the model should head towards the top road, even before the bottom road is congested – but is inevitably becoming so. In network B, a similar reasoning applies. With no interventions, all agents would first travel to C, then to D, and then to E. With Δ-tolling applied, half of traffic should take the top route, and half of traffic should take the bottom route. When prediction is applied, agents in the model should head towards the section BD, even before section CE is congested – but is inevitably becoming so.

These tests serve to demonstrate and check the workings of the model. If the model proves to work, it ca then be implemented on the Sioux Falls network.

3.6 Sensitivity and scenario analyses

After defining the model specifications and experimental environments, this section now explains how the output is analysed. As pronounced in research questions 3 and 4, this study aims to find the effects of the proposed method on *system performance*, *equality* and *fairness*.

System performance

In this study, system performance is defined as average travel time. To answer research question 3, the average travel times of each scenario are analysed. Using the different scenarios with the same travel demands, the response of average travel time to differing parameter settings is examined. Per scenario, this analysis results in a different output. In **scenario 1**, the output is straightforward: there is only one scenario setting, which returns one average time. This average time is used for comparison in scenario 2. In scenario 2 and 3, the output is slightly more complex. **Scenario 2** returns a graph of the percentage decrease in average travel time for each setting of parameter α. For the α which returns the lowest average travel time, it also returns a graph of the average travel time for different shares of "smart agents". As explained in section 3.4, the lowest average time that follows from this scenario is used as a "user equilibrium benchmark" to compare the outcome of other scenarios 3 and 4 with. In **scenario 3**, the average travel time per parameter setting is analysed. This returns a heat map, showing the percentage average travel time decrease per β-R combination.

Since **scenario 4** has three varying variables, it does not allow for a direct graphic representation of the results. Therefore, for each β-R combination, the run with the value of α which returns the largest decrease in average travel time after the use of prediction is presented in a heat map (in percentage, compared to the user equilibrium). This scenario also returns a scatterplot of decrease in average travel time due to tolling set out against decrease in average travel time due to prediction.

Equality

In this study, equality refers to the distribution of travel times over the population of drivers. Based on literature, it is expected that the distribution of individual becomes less equal when the model strives for system optimum. In the hypothetical networks, this can easily be assessed, since in those models, with no intervention, all drivers take the same route and thus have the same travel time. Any alteration from the initial situation is thus likely to create a less equal distribution of travel times. In the Sioux Falls network, this analysis is less straightforward. Because drivers on the network have different origins and destinations, individual travel times among the drivers are inherently unequal. To assess the equality of a measure, this study therefore compares the distribution of travel times of each scenario to the distribution of travel times in user equilibrium. This distribution is displayed using a Lorentz curve and Gini-coefficient, similar to Levy and Ben-Elia (2016). The Gini-coefficient is computed using an online tool (Shlegeris, 2020).

Fairness

Fairness relates to impact that interventions have on individual travel times. For this analysis too, the different scenarios are compared to user equilibrium. Since in both the hypothetical networks and the Sioux Falls networks traffic is set on regular intervals or by a predefined schedule (which means that, e.g., the tenth spawned driver in one run, is also the tenth spawned driver in another), the travel times of the same agents over different runs can be directly compared. This way, for each agent, the impact of a measure on its individual travel time can be determined. Using this data, the impact of different scenarios can be compared and analysed. In order to do this, a histogram is created, which shows the impact of a measure on individual travel time, and the share of drivers affected by this impact.

Chapter 4

Results

This chapter presents the results as found after constructing the model and running different scenarios. First, section 4.1 addresses model verification by inspecting different components of the model. Then, the other sections address the sensitivity and scenario analyses (4.2), results on equality (4.3) and results on fairness (4.4).

4.1 Verification

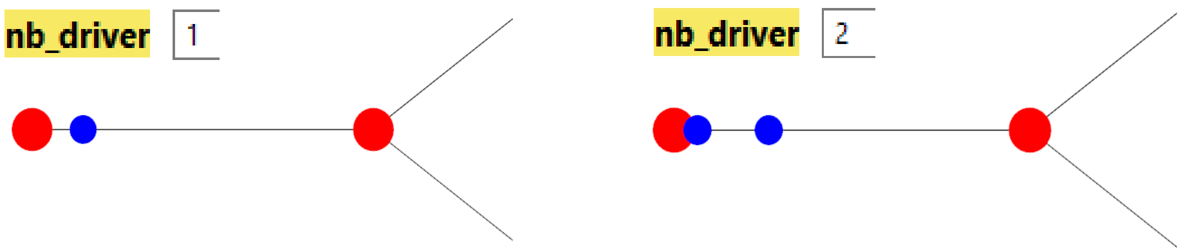
The model is verified on three aspects: the links, the driver, and model behaviour. The aspects driver and link are verified using network A only, since these aspects are independent from the network used. Model behaviour is verified on both network A and network B.

4.1.1 The link

The links needs to accurately count the number of drivers on it, and compute its variables accordingly. It also needs to update its occupancy matrix correctly.

Counting drivers

When the model is ran, it can be seen that the model accurately counts the number of agents on it. A close-up of the first link, accompanied with the driver counter, is shown in figure 4.1.



(a) With one driver

(b) With two drivers

Figure 4.1: Part of network A with the driver counter

4.1.2 Variable computation

With the link occupancy and the network details given in the previous chapter, the link variables can be easily computed. The table below provides an overview of a set of link variables, their computed value and the value found in the model for link DF, with an occupancy of 3. It is concluded that the variables are computed correctly.

Table 4.1: *Link variables, computed manually and found in the model, for link DF with an occupancy of 3. The discrepancy in the link toll is caused by rounding.*

Variable	Value (computed)	Value (found in model)
Link travel time	1.859,42 s	1.859,42 s
Link speed	32,27 ms	32,27 ms
Link toll ($\beta = 1$, R=0)	59,42 s	59,40 s
Link cost	1.918,84 s	1.918,82 s

Occupancy matrix

If the model is ran, it can be seen that the first agent has entered link AB at t = 10, and left the link at t = 305 (in seconds). This means that the agent is on the link from minute 0 to minute 5. Figure 4.2 shows a part of network A after one executed cycle, accompanied with the occupancy matrix of link AB. It can be seen that the agent has indeed correctly communicated its intentions, and that the link has registered this in the column at the right.

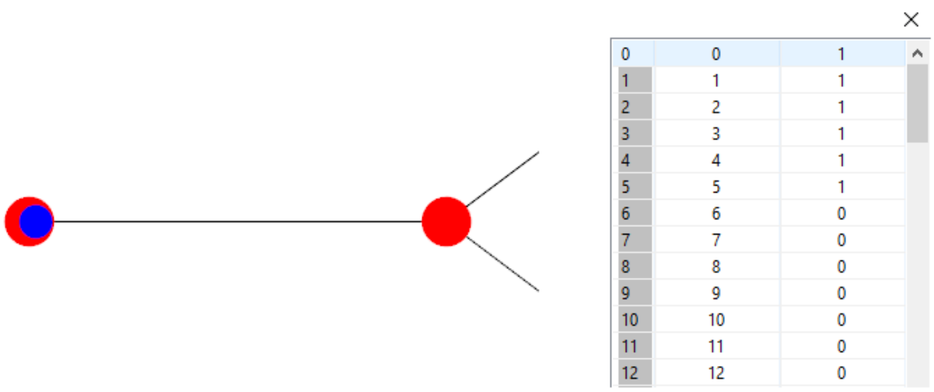
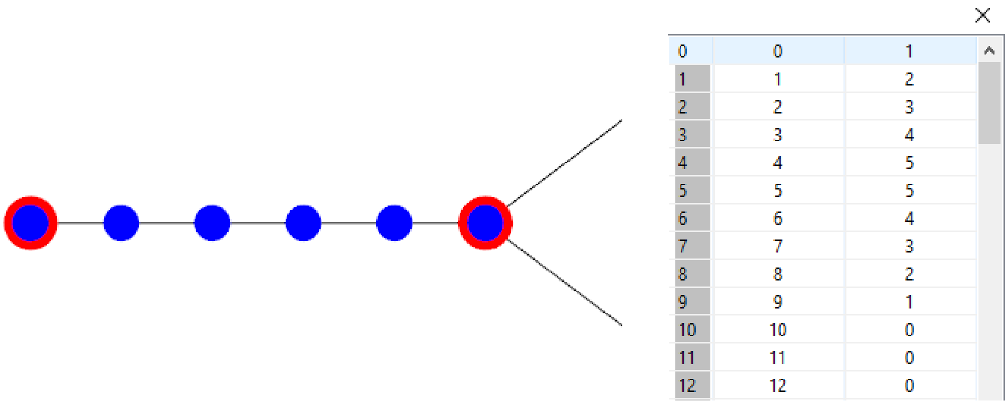


Figure 4.2: Part of network A and the occupancy matrix of link AB at t=10 s

Figure 4.3 shows the same link and occupancy matrix at t = 305 seconds. Here it can be seen that the first agent indeed leaves after 5 about minutes. Moreover, it can also be seen that the agents following the first drivers have also announced their presence at the link. At 5 minutes, the occupancy matrix indicates that there are 5 agents on it. This corresponds with the figure, since the driver at the left hand side has not entered the link yet: it is still on the starting node.



0	0	1
1	1	2
2	2	3
3	3	4
4	4	5
5	5	5
6	6	4
7	7	3
8	8	2
9	9	1
10	10	0
11	11	0
12	12	0

Figure 4.3: Part of network A and the occupancy matrix of link AB at t=305 s

Since the link registers agents on it, computes variables correctly and updates its occupancy matrix adequately, it can be concluded that the links function as designed.

4.1.3 The driver

With no intervention, the drivers need to take path to their destination. They also need to drive at the correct speed, and when interventions are implemented, they need to consider tolls. That the agents correctly communicate their intentions has already been shown above.

Shortest path

If the model is initiated, it can be seen that the driver indeed computes a correct shortest path. With no interventions, all drivers take the bottom route. Also with other OD combinations on an undirected network A, drivers return the correct shortest path.

Speed

In the given settings for network A, route DF delays with 1,1% per driver on it. This implies that for every driver on the link, the travel time required to traverse the link rises with 1,1%. This, in turn, means that the speed in this new situation is 33,33 / 1,011 = 32,97 m/s. According to this method, table 4.2 gives an overview of numbers of drivers and the associated speed. Running the model shows that agents indeed adapt to the speed on the link.

Table 4.2: *Computed and in-model speed of agents, for different numbers of agents on link DF*

Number of agents on the link	Speed (computed, m/s)	Speed (found in model, m/s)
1	32,97	32,97
2	32,62	32,62
5	31,60	31,60
10	30,03	30,03
20	27,32	27,32

Toll

When in network A tolling is applied, agents should, when the toll is high enough, choose another path. In network A, this implies that when the total cost of link DF is higher than the cost of CEF, drivers at point B should head for route CEF. Given the settings as described in section 3.3.2, this should occur at a delay of 25% (since that causes a toll of 25% as well, and thus makes DF 50% more costly than CEF). Since each agent causes a 1,1% delay, route CEF becomes favourable over route DF at 25/1,1 = 23 drivers. Figure shows that when 23 drivers are on route DF, the driver at B has indeed decided to opt for route CEF. This shows that agents indeed respond to tolls on the network.

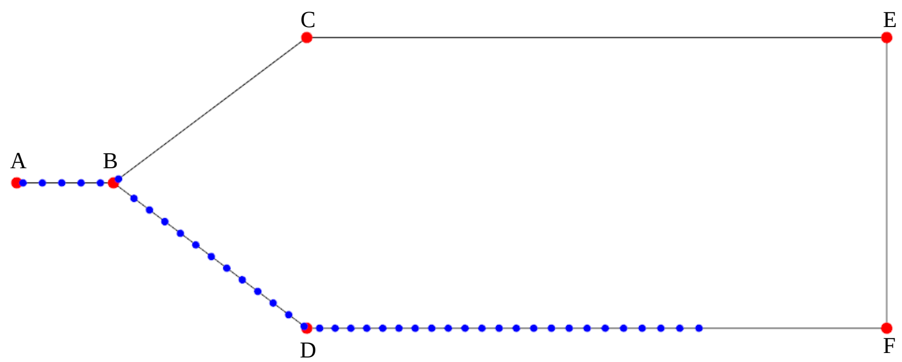


Figure 4.4: Part of network A, with one agent responding to tolls

4.1.4 Model behaviour

This paragraph addresses both network A and network B. On both networks, the traffic patterns are examined in order to verify the models workings. On each network, the model will be ran using multiple parameter settings, to determine whether the model's response corresponds to our expectations.

Patterns in network A

Network A is a modified implementation from Pigou's example mentioned in paragraph 2.1.1. Therefore, one route is load-dependent (bottom), and one route isn't (top). In the bottom route, network link DF can become congested, increasing the travel time on the link up to 50%. Since the combined length of CEF is 50% longer than the length of DF, this means that with no intervention, all drivers will choose the bottom route. As shown in figure 4.5, this is also what happens in the model.

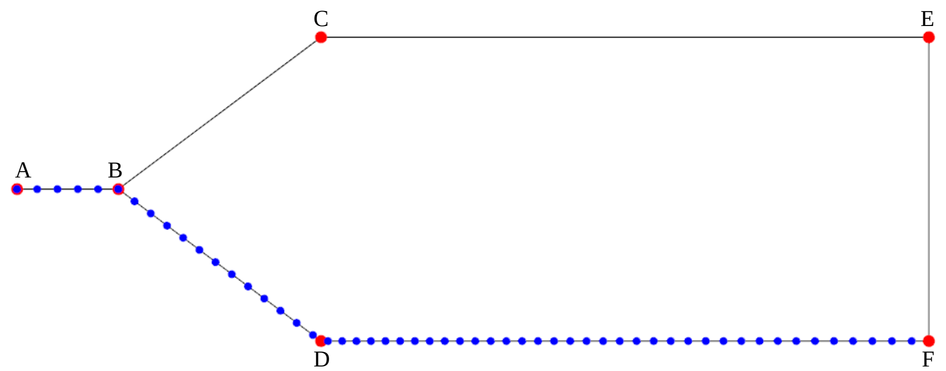


Figure 4.5: Network A with no interventions implemented

As demonstrated above, when tolls are implemented, agents opt for route CEF if there are 23 drivers on link DF. As soon as there are fewer than 23 drivers on DF, drivers at node B head for node D. Figure 4.6 demonstrates this. The figure shows that when there are only 22 drivers left on link DF, the first agent at B decides to take route DF. However, this is not optimal: the moment that the latter agent arrives at link DF, other agents will have already left link DF. This implies that other agents before the agent currently at B could have decided to head for link DF, and have arrived on a link with less than 23 drivers on it. As can be seen in figure 4.4, this late response works is also apparent when link DF has not yet reached 23 drivers. At the time of figure 4.4, DF has 23 drivers on it. However, the agents on link BD are already inevitably heading for link DF. If they had known link DF was about to be congested, they could have opted for route CEF.

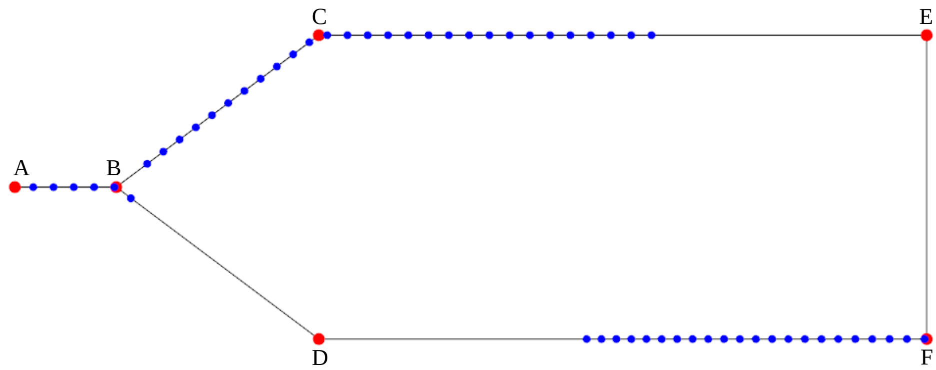


Figure 4.6: Network A with tolls

This is where intention-based prediction can assist. If all agents are given “smart” properties, they could anticipate on future occupancy of the links. This is displayed in figure 4.7, where it can be seen that while there are only 23 agents on link DF, and that other agents are already heading towards DF, in anticipation of the departure of the other agents on the link. This shows that the drivers respond to predictive information.

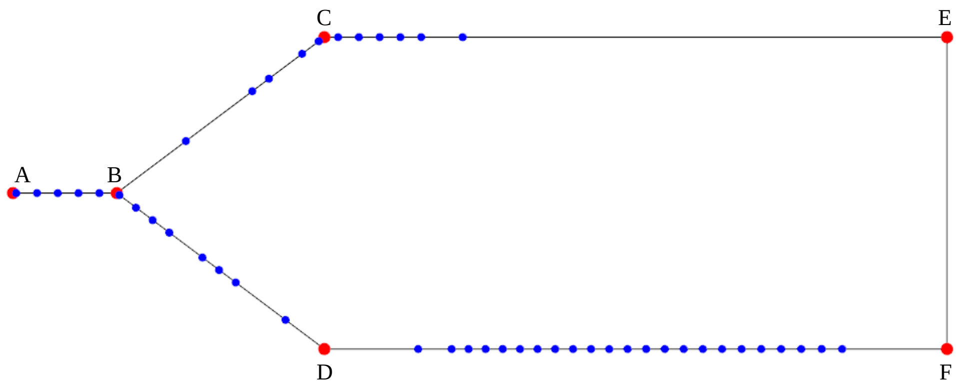


Figure 4.7: Network A with tolls and intention-based prediction

Patterns in network B

Network B is a modified example of the Braess paradox as introduced in paragraph 2.1.2. In this network, with no interventions, all traffic should follow the route ABCDE, of which link CD has no cost. Figure 4.8 shows that the Braess paradox also occurs in this network.

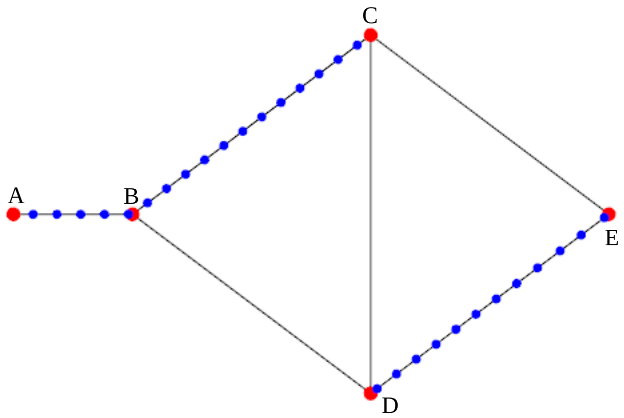


Figure 4.8: Network B with no interventions

When tolls are applied, the costs of links BC and DE should increase, so that links BD and CE become the favourable options. This way, traffic should disperse itself over the two routes. Since each driver causes a delay of 8,3%, and BD and CE are twice as slow as BC and DE, drivers should opt for BD and CE when 7 drivers are on link BC or DE, because 7 drivers cause a delay of 50,81% and an additional toll of 50,81%, rendering BC and DE slower than BD and CE. Figure 4.9 illustrates this.

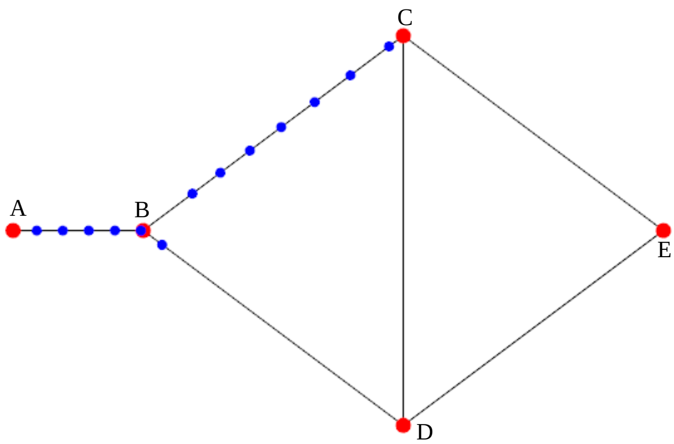
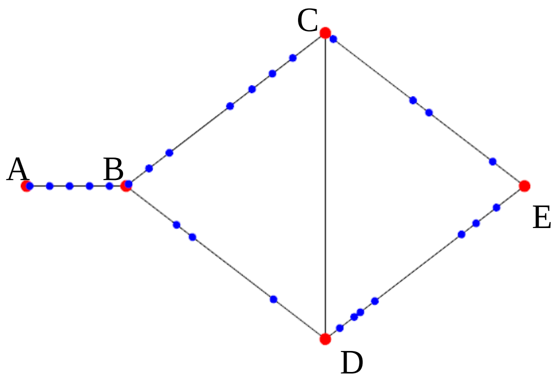
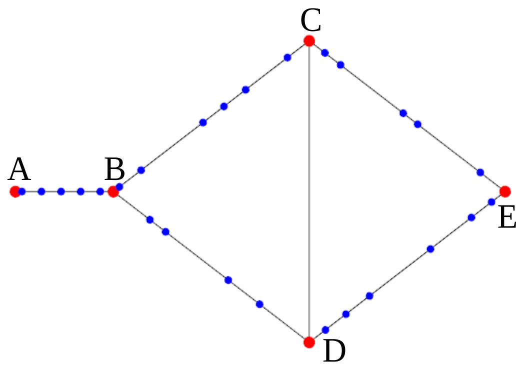


Figure 4.9: Network B, with one agent responding to tolls.

The case for the use of prediction is harder to find in network B. However, prediction does offer benefits for network B, too. Consider figure 4.10a of network B with only tolls applied. At this moment, link DE has 7 drivers on it, while CE only has 4. DE is thus more congested than necessary. This crowdedness on link DE is caused by a lack of knowledge on the future. Namely, lack of knowledge on the future makes that agents at node C might travel to link DE, even though drivers on BC are already about to enter DE. Thus, DE is overcrowded. This shows in the figure due to the cluttered drivers on link DE. If the agents at node C had already known that more agents were about to enter link DE, they would have reconsidered, and chosen for link CE, which would cost less.



(a) Network B with tolls



(b) Network B with tolls and prediction

Figure 4.10: Network B with tolls or tolls and prediction

This effect is visible in figure 4.10b. Here, intention-based prediction is applied. On link CE there are 5 agents and there are 6 on link DE, none of which are cluttered. This implies that none of the agents at point C transferred to link DE. Thus, link DE is not overcrowded. This demonstrates that the drivers adequately respond to predictive information.

The above examples mainly served to test the model, and display its workings. Now the functioning of the model has been verified, the following section addresses the effect that different model parameter settings have on model output. In that section, the model is also implemented on the Sioux Falls network.

4.2 Sensitivity analysis and system performance

This section addresses the sensitivity of model outputs to different parameter settings. In that sense, this section functions as both a sensitivity analysis as an analyses of model performance in different scenarios. Once again, this study first deals with networks A and B, in order to demonstrate the potential effects of the measures. After that, the workings of the model on the Sioux Falls models are examined. For network, the different scenarios are examined.

4.2.1 Results on network A

Scenario 1 In this scenario, no tolling and no prediction is applied. As seen in the previous section, this leads to all traffic travelling over the bottom route. After 10.000 cycles, the average travel time (ATT) is 3.744,61 seconds.

Scenario 2 In this scenario, the influence of intention-based prediction is measured. It is found that in no parameter setting for α and for no share of smart agents, this scenario changes the system outcome with respect to scenario 1. This can easily be accounted for: without tolling, the bottom road never becomes more than 50% more costly. The bottom road is thus never more expensive than the top road. As a result, drivers will never opt for the top road, even if they can predict the crowdedness on link DF. This means that the user equilibrium benchmark in this scenario is 3.744,61 seconds.

Scenario 3 In this scenario, parameters β and R are tested over the ranges specified in section 3.4. The travel times found in this scenario, are then compared to the user equilibrium benchmark. The heat map in figure 4.11 displays the decrease in travel time for each β-R combination. It is found that the combination β=1,5 and R=10^0 yields the highest decrease in travel time, leading to an ATT of 3.541,72 seconds (a decrease of 6%). For R values between R = 100 and R = 10^{-3}, β values close to $\beta = 1$ yield the greatest reduction in travel time, although overall differences in effect are small. For most β values, the effect of tolling initially decreases as R decreases, but rises again when R nears R = $10^{-4.5}$.

Scenario 4 If tolling and prediction are combined, prediction shows to have potential value in some cases. In paragraph 4.1.4 we saw that prediction can help drivers to adequately respond to tolls. In scenario 4, it is found that for the examined β-R combinations, with the use of

		β																				
		0	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6	6.5	7	7.5	8	8.5	9	9.5	10
R	10^0	0	6	6	6	6	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4	4
	10^-1	0	6	6	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
	10^-2	0	5	5	5	5	5	5	5	5	5	5	5	4	4	4	4	4	4	4	4	4
	10^-2.5	0	5	5	5	5	4	4	4	4	4	4	4	4	4	4	4	4	3	3	3	3
	10^-3	0	5	5	5	5	5	5	4	4	4	4	3	3	3	3	3	3	3	3	3	2
	10^-3.5	0	5	6	6	5	5	5	5	4	4	4	4	4	3	3	3	3	3	3	2	2
	10^-4	0	4	5	5	5	5	5	5	5	5	5	5	5	4	4	4	4	3	3	3	3
	10^-4.5	0	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
	10^-5	0	3	4	4	4	4	4	4	4	5	5	5	5	5	5	5	5	5	5	5	5

Figure 4.11: Decrease in travel time (%) due to tolling for each β-R combination on network A

prediction, the minimum ATT decreases to 3.495,45 seconds, for β = 2,5, R = $10^{-4.5}$ and α = 0,4. Compared to the same β-R settings in scenario 3, this is an additional decrease of 2%. The heat map in figre 4.12 displays, for each β-R combination, the maximum additional decrease in travel time that prediction brought in comparison to the same β-R settings in scenario 3. The highest benefit of prediction was found for β = 1 and R = 10^{-5}, which returned an ATT of 3.616,48 seconds (3% decrease).

		β																				
		0	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6	6.5	7	7.5	8	8.5	9	9.5	10
R	10^0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	10^-1	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	10^-2	0	1	2	2	2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	10^-2.5	0	1	2	2	2	2	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
	10^-3	0	1	2	2	2	1	1	1	1	1	1	1	1	1	1	0	1	1	1	0	0
	10^-3.5	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	10^-4	0	2	2	2	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	10^-4.5	0	2	3	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1	1	1
	10^-5	0	3	3	3	3	3	3	3	2	2	2	2	2	2	2	2	2	2	1	1	1

Figure 4.12: Additional decrease in travel time (in %) due to prediction for each β-R combination on network A

When comparing figure 4.11 and fig 4.12, there seems to be some correlation. The top right quadrant in 4.11 shows rather high impact of tolling, whereas the same quadrant in 4.12 shows a low impact of prediction. For the bottom left quadrant, this observation seems to be the other way around: whereas 4.11 show relatively low results there, 4.12 show relatively high results. To examine this suspected relation, figure 4.13 shows a scatterplot on which the value of 4.11 are set out against the values of 4.12. On this figure, each dot on the plot represents one β-R setting. It's location along the x-axis displays the percentage decrease of ATT after tolling (scenario 3). The location on the y-axis indicates the maximum additional decrease of ATT caused by prediction for the β-R combination. Here, the relation between tolling and prediction seems to be less apparent. Only for β-R combinations in which prediction is relatively successful, there appears to be a negative relation between prediction impact and tolling impact. For β-R settings leading to less impact from prediction, the effect of tolling is varied, ranging between a 2% and 6% decrease in ATT. Note that for some β-R combinations prediction has a slight negative effect.

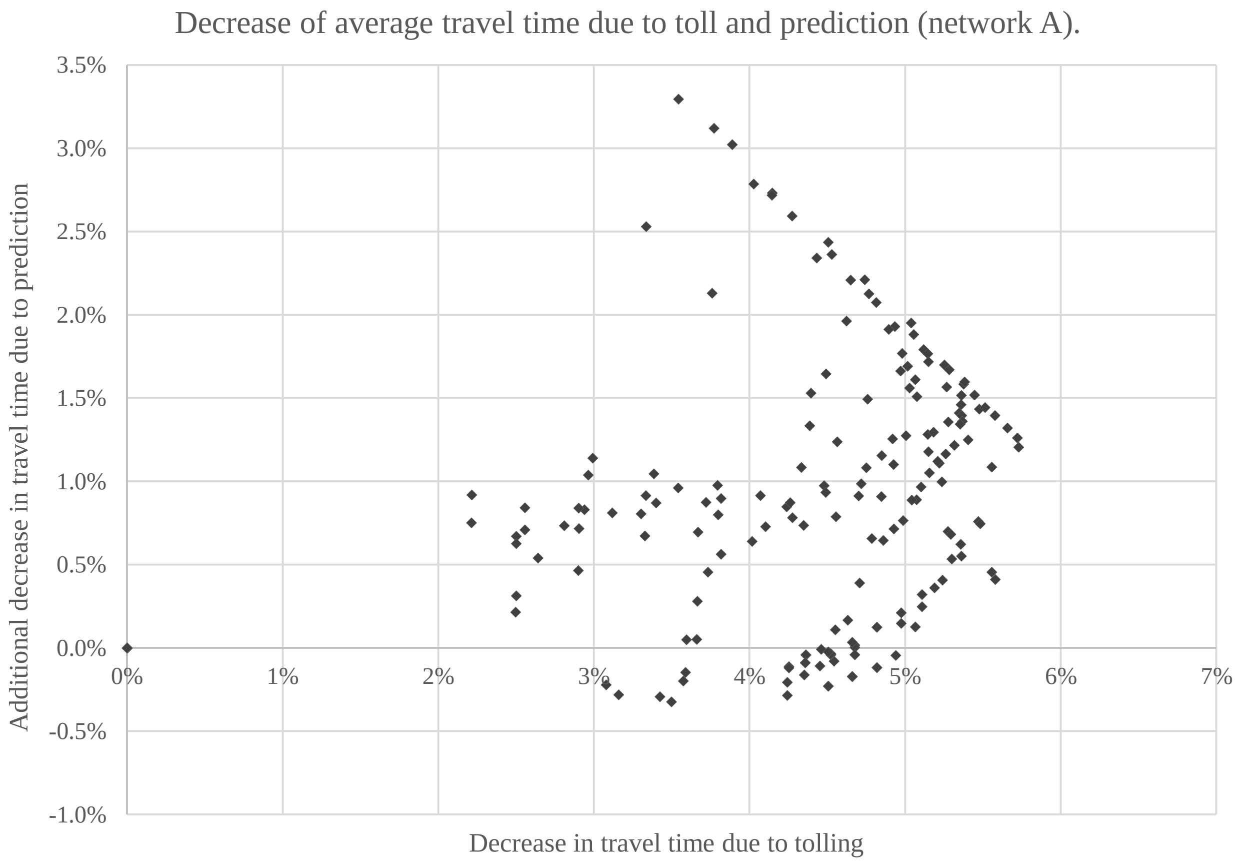


Figure 4.13: β-R combinations and the decrease of travel time, due to toll and predictions, network A

4.2.2 Results on network B

Scenario 1 With no interventions, the classic Braess paradox plays out on network B. All traffic chooses the travel over the load-dependent roads, resulting in an average travel time of 1.800,12 seconds after 10.000 cycles.

Scenario 2 Similar to network A, in this network, prediction without tolling does not have effect. The explanation is the same: given the model's settings, the links BC and DE are never costlier than BD and CE. Drivers will not change behaviour when using predictive information. This implies a user equilibrium benchmark of 1.800,12 seconds.

Scenario 3 The heat map in figure 4.14 shows the effect of tolling for different β and R combinations in a similar fashion to 4.11. It shows that tolling can lead to a decrease in travel time of up to 16% for β=1,5 and R=10^0 (ATT 1.556,18 seconds). The optimal settings found are thus similar to the settings found in network A, but their impact is larger on network B. The heat map shows that in general, the effect of toll decreases as R decreases, with an exemption seen for values of $\beta >= 5$. There, a value of R smaller than 10^{-1} initially lessens the effect of toll, but toll becomes more effective again if R is decreased to 10^{-3} and smaller. For all values of β, R=10^{-5} returns a relatively small effect of the toll.

		β																				
R		0	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6	6.5	7	7.5	8	8.5	9	9.5	10
	10^0	0	13	15	16	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15
	10^-1	0	12	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15
	10^-2	0	12	15	14	13	13	15	15	14	14	13	8	8	9	8	8	8	8	8	8	8
	10^-2.5	0	12	15	15	15	14	15	11	14	14	9	14	10	14	8	8	8	8	8	8	7
	10^-3	0	11	14	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15	14	15	15
	10^-3.5	0	9	13	14	14	14	14	14	15	14	15	15	15	14	15	15	14	15	15	15	15
	10^-4	0	6	10	12	13	13	13	14	14	14	14	14	14	14	14	14	14	15	14	14	14
	10^-4.5	0	3	5	7	8	9	10	11	11	12	12	13	13	13	13	13	13	13	13	13	14
	10^-5	0	2	3	3	4	5	5	6	6	7	7	8	8	9	9	9	10	10	10	11	11

Figure 4.14: Decrease in travel time (%) due to tolling for each β-R combination on network B

Scenario 4 Similar to network A, a combination of tolls and prediction can make that drivers respond timely to congestion. On network B, however, the use of prediction does not lower the minimum found ATT. With the use of prediction, the lowest reached ATT is slightly higher than the minimum ATT in scenario 3: 1.559,68 seconds ($\beta = 1{,}5$, $\mathrm{R} = 10^{-2}$ and $\alpha = 0{,}4$). The heat map in figure 4.15 displays, for each β-R combination, the maximum additional decrease in travel time that prediction brought in comparison to the same β-R settings in scenario 3. The highest ATT decrease due to prediction is found at $\beta = 1{,}0$, $\mathrm{R} = 10^{-5}$. Here, prediction decreases the ATT with 11% for $\alpha = 0{,}4$.

		β																				
R		0	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6	6.5	7	7.5	8	8.5	9	9.5	10
	10^0	0	1	1	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1
	10^-1	0	1	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1
	10^-2	0	1	1	1	2	2	1	0	1	1	2	6	6	5	6	6	6	6	6	7	7
	10^-2.5	0	2	1	0	1	1	1	3	1	1	6	1	4	1	6	6	6	6	6	7	7
	10^-3	0	2	1	1	1	1	0	0	0	0	0	1	0	1	0	1	1	1	1	0	1
	10^-3.5	0	4	2	1	1	1	1	1	0	1	1	1	1	1	1	1	1	0	0	1	0
	10^-4	0	7	5	3	2	2	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	10^-4.5	0	9	9	8	6	5	4	4	3	3	3	2	2	2	2	2	2	2	2	2	1
	10^-5	0	10	11	10	10	9	9	8	8	7	7	6	6	6	5	5	5	4	4	4	4

Figure 4.15: Additional decrease in travel time (in %) due to prediction for each β-R combination on network B

When comparing the two figures above, the two heat maps look remarkably opposite: β-R combinations which lead to a high decrease in ATT in figure 4.14 are associated with low effects of prediction in figure 4.15, and vice versa. The scatterplot in figure 4.16 explores the relation between the effect of tolling and the additional effect of prediction. Each dot on the plot represents one β-R setting, with the percentage decrease of ATT after tolling (scenario 3) on the x-axis, and the maximum additional decrease of ATT caused by prediction for the β-R combination on the y-axis. It clearly shows a negative relation between the effect of the two measures: β-R combinations leading to effective tolls, experience a lower effect of prediction. Indeed, it is specifically noteworthy that the for β=1,5 and R=10^0, which returned the lowest ATT in scenario 3, the effect of prediction is negative: applying predictive information in that case thus increased the average travel time. A cause of this relation might be that with effective tolling there is little possibility for prediction to further improve the system's efficiency. When traffic is already navigated in a rather efficient way (due to tolling), there is less potential for prediction to improve traffic flows than when traffic is not routed efficiently. This effect might be better explained on an individual level: a driver which already drives efficiently due to tolling, is not probable to drive *more* efficiently if it is using prediction as well.

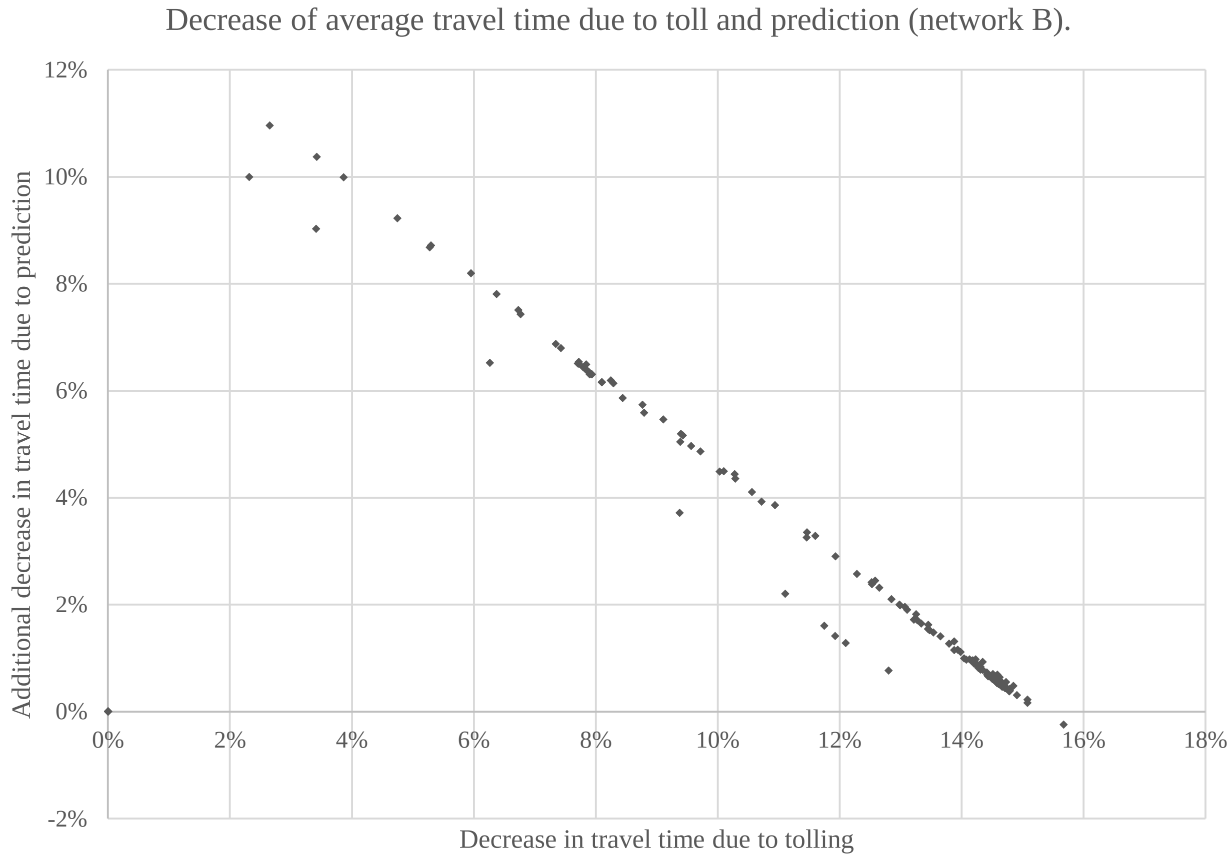


Figure 4.16: β-R combinations and the decrease of travel time, due to toll and predictions, network B

4.2.3 Results in Sioux Falls

Before the result on the Sioux Falls network can be tested, one loose end from paragraph 3.3.3 needs solving. Because this study modifies the travel demand as provided in the original dataset, the link latency functions still need to be corrected for the number of drivers on this network. When using the modified demand data in the model, there are approximately 90 to 130 drivers on the network at one time. Therefore, the flow is multiplied by 8.000 (877.603 divided by 110, the average of 90 and 130, is approximately 8.000), to compensate for a lower overall presence of drivers on the network. The link latency functions in this network thus get the following format:

$$current\,travel\,time = free\,flow\,travel\,time + b * (8.000 * flow)^4 \tag{4.1}$$

In similar fashion to the paragraph above, this paragraph now explores the effects of the different scenarios on model output.

Scenario 1 With no tolls or prediction in place, the Sioux Falls network returns an average travel time of 3.067,22 seconds after 4.000 cycles.

Scenario 2 Figure 4.17 shows the effect of intention-based prediction when not applying tolls. Different values of α and the accompanying percentage decrease in travel time (compared to scenario 1) are displayed. It can be seen that at a value of $\alpha = 0{,}5$, prediction has the highest effect, returning a travel time of 2.686,87 seconds (12% decrease). Most other values return a positive effect as well. However, when predictive information is weighed at 100%, prediction has a slight negative effect on average travel time when no tolls are implemented. This shows that without tolling, taking both current and predictive information into account leads to better system performance than merely considering current or predictive data.

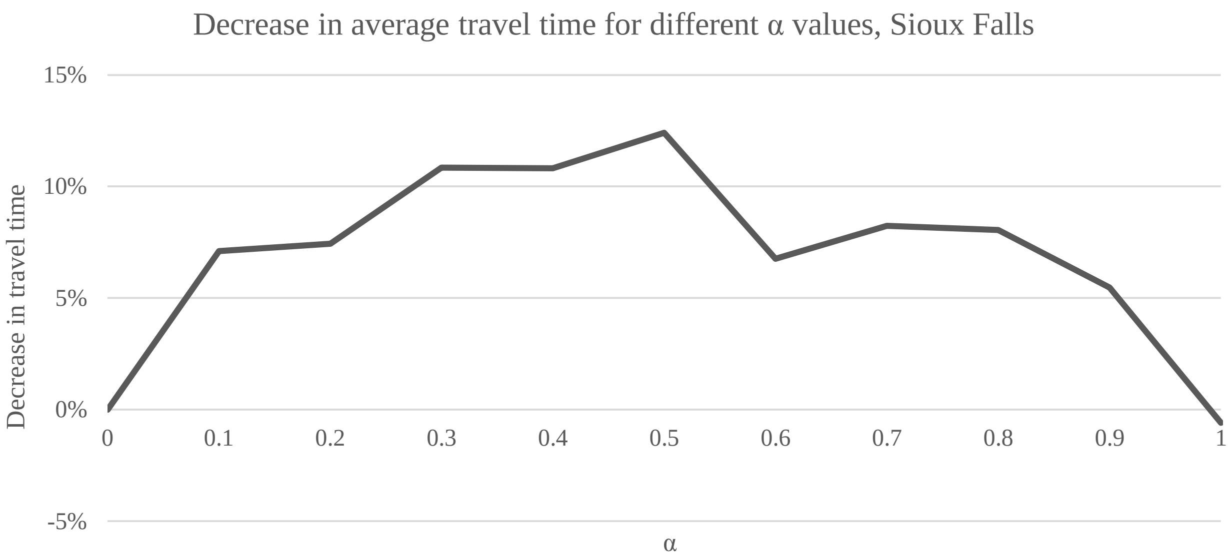


Figure 4.17: Decrease in average travel time due to prediction for different values of α, Sioux Falls (%)

Figure 4.18 shows for $\alpha = 0{,}5$ the impact of different shares of smart agents on average travel time. For each share of smart agents, the graph shows the average of three model runs. It shows that, in general, higher shares of smart agents lead to a higher impact of prediction. Note that for a small share of smart agents (10%), prediction leads to an increase in the average travel time compared to scenario 1. This demonstrates that the effect of predictive information on system performance is highest when there are more smart agents. This might be explained by the reasoning that if not all agents participate, the predictions of smart agents are less reliable. After all, since only smart agents share their intentions, only the presence of smart agents is considerations by smart agents making travel plans, and the presence of regular agents might be overlooked. This can then result in non-optimal routing decisions. The lowest average travel time found in this scenario occurs at $\alpha = 0{,}5$ with only smart agents. Thus, 2.686,87 seconds is considered to be the user equilibrium benchmark.

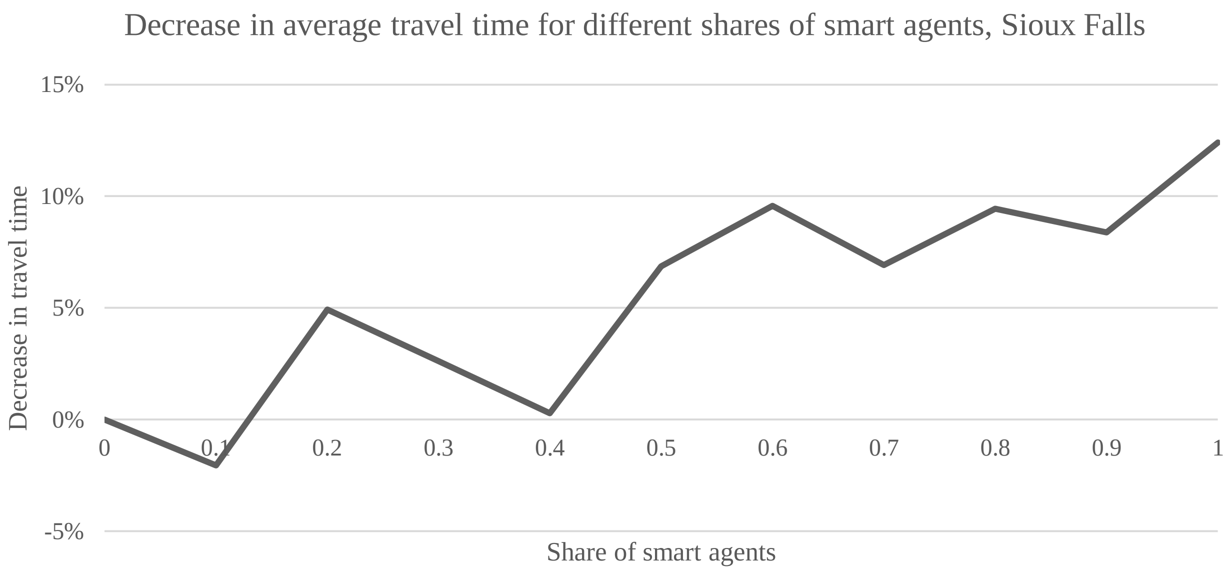


Figure 4.18: Decrease in average travel time due to prediction for different shares of smart agents, Sioux Falls (%)

Scenario 3 The heat map in figure 4.19 shows the results of scenario 3, with β-R combinations set as described in paragraph 3.4. Note that these figures relate to a percentage decrease in ATT when compared to the user equilibrium benchmark of 2.686,87 seconds, found in scenario 2. The largest decrease in travel time (18%) is found with a setting of $\beta = 10$ and $\mathrm{R} = 10^{-3.5}$. This leads to an ATT of 2.195,24 seconds. This is an decrease of almost 28% with respect to scenario 1. When comparing scenario 2 and 3, it appears that tolling (scenario 3) has a larger effect on the average travel time than prediction (scenario 2). Considering β, it is found that for almost any value, tolling has the potential to decrease the average travel time when compared to the user equilibrium. However, a right value of R is of high importance for the effectivity of the toll. It can be seen that low settings of R often lead to a less effective toll, not being able to improve the network when compared to the user equilibrium. R-values in the range of $10^{-2.5}$ to 10^{-4} render more effective tolls. The influence of of R-values on the effect of tolls is shown in figure 4.20. This graph displays the average decrease in ATT per value of R.

		β																				
		0	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6	6.5	7	7.5	8	8.5	9	9.5	10
R	10^0	-14	-14	-23	-22	-23	-23	-30	-27	-30	-27	-26	-29	-25	-26	-30	-31	-26	-30	-34	-27	-30
	10^-1	-14	-13	-18	-20	-24	-24	-27	-24	-29	-26	-29	-24	-24	-28	-28	-28	-28	-27	-22	-22	-34
	10^-2	-14	-3	-7	-3	-4	-9	-7	-5	-9	-10	-7	-11	-11	-10	-11	-13	-12	-9	-15	-10	-14
	10^-2.5	-14	1	3	7	9	9	8	6	6	8	10	7	7	5	8	7	9	6	7	6	9
	10^-3	-14	2	6	10	10	12	15	16	15	17	15	15	17	17	16	16	16	16	17	16	14
	10^-3.5	-14	-5	-1	4	7	9	12	11	11	13	15	13	13	12	16	16	18	15	18	18	18
	10^-4	-14	-10	-5	-6	-3	0	-1	7	5	2	8	6	6	8	10	12	12	13	12	14	11
	10^-4.5	-14	-10	-7	-9	-10	-8	-7	-4	-3	-4	-1	-3	-6	-5	-2	-4	1	-2	4	3	3
	10^-5	-14	-19	-14	-14	-11	-9	-7	-13	-15	-12	-11	-9	-10	-9	-10	-10	-7	-7	-6	-5	-6

Figure 4.19: Decrease in travel time (%) due to tolling for each β-R combination on Sioux Falls network

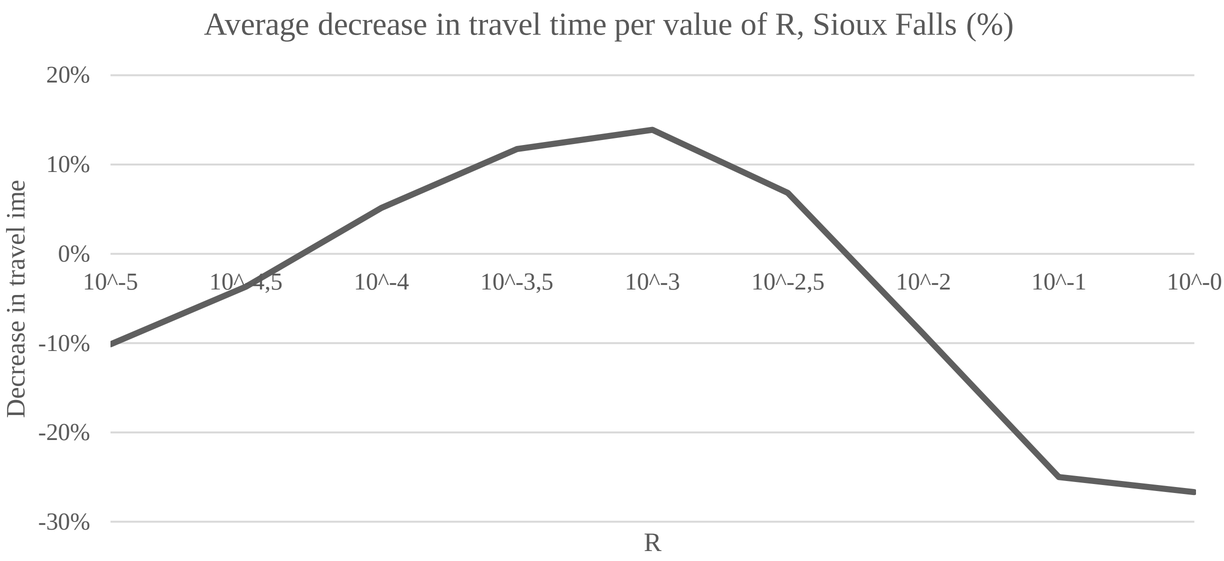


Figure 4.20: Average decrease in travel time per value of R, Sioux Falls network (%) (%)

Scenario 4 Similar to network B, the ATT on the Sioux Falls network in scenario 4 does not improve vis-á-vis scenario 3. The lowest ATT found when applying tolls and prediction is 2.198,08 seconds (β = 3,5, R = 10^{-3} and α = 0,2), slightly higher than the ATT of 2.195,24 seconds found in scenario 3. The heat map in figure 4.21 displays combinations of β and R and the maximum decrease in travel time found in scenario 4. Note that this these figures do not relate to the user equilibrium benchmark found in scenario 2, but to the travel times found in scenario 3, and thus display the additional effect that prediction has on the travel time found after tolling with the relevant β-R setting. It is visible that the highest positive impact of prediction occurs at low values of β and R. In this network, too, these values seem to coincide with β-R combinations which yielded relatively less effective tolls.

Figure 4.22 displays a scatterplot which projects each β-R combination and the effect of tolling on the ATT when compared to user equilibrium (x-axis) and the effect of prediction on the ATT (y-axis). Again, there appears to be a negative relation between the effectiveness of toll, and the effectiveness of prediction. This relation might be explained following the same logic as presented in paragraph 4.2.2: a network in which tolling leads to effective routing of drivers, prediction is less likely to improve the navigation of drivers. The relation here, however, seems less strong than the relation shown in figure 4.16. This can be explained by the fact that the Sioux Falls network is more complex than network B. As a result, Sioux Falls offers more diverse routing possibilities than network B. This, in turn, results in a larger variation in model outcomes, and a weaker relation between the two interventions.

R		β																				
		0	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6	6.5	7	7.5	8	8.5	9	9.5	10
	$10^{\wedge 0}$	11	7	10	10	7	7	10	8	8	8	5	7	3	5	8	12	2	6	11	8	9
	$10^{\wedge -1}$	11	6	9	7	11	13	13	11	13	9	8	7	7	9	11	11	9	9	2	3	10
	$10^{\wedge -2}$	11	5	7	6	9	13	8	7	8	9	7	9	10	6	6	12	11	7	12	9	8
	$10^{\wedge -2.5}$	11	4	4	5	5	7	3	4	4	4	2	2	6	6	8	5	6	6	4	4	4
	$10^{\wedge -3}$	11	5	1	-1	2	3	0	3	1	-5	0	0	-2	0	-3	0	0	2	0	1	1
	$10^{\wedge -3.5}$	11	10	7	0	0	-3	-3	-2	-5	2	-3	-1	-1	0	-4	-3	-9	1	-10	-6	-6
	$10^{\wedge -4}$	11	10	6	7	4	1	4	-6	-4	0	-2	1	-4	-5	-11	-14	-12	-12	-14	-18	-9
	$10^{\wedge -4.5}$	11	6	8	4	2	2	1	-5	-5	3	-7	-5	1	-3	-3	-6	-9	-3	-8	-15	-16
	$10^{\wedge -5}$	11	14	11	8	3	1	1	3	5	3	4	-5	-2	-6	1	-1	-7	-6	-10	-12	-9

Figure 4.21: Additional decrease in travel time (in %) due to prediction for each β-R combination on Sioux Falls network

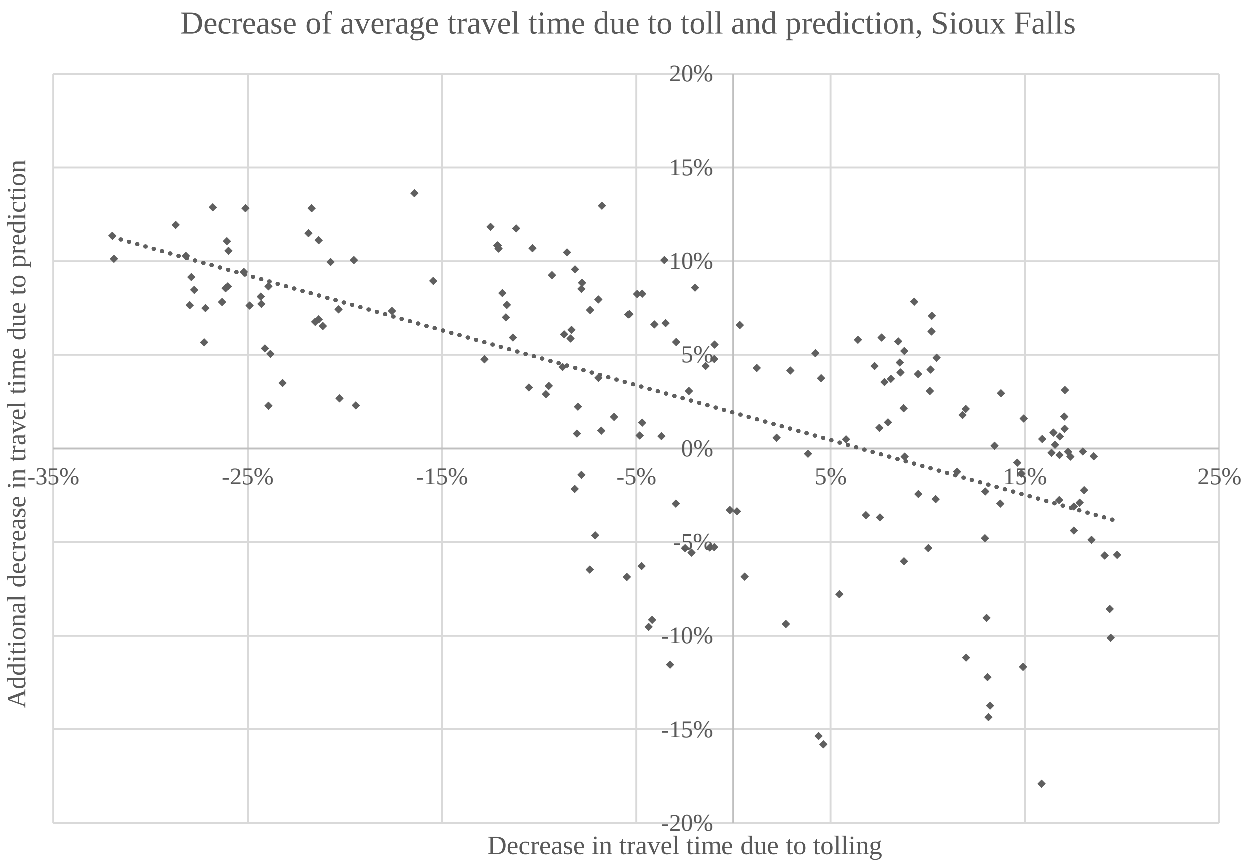


Figure 4.22: β-R combinations and the decrease of travel time, due to toll and prediction, Sioux Falls

4.3 Equality results

4.3.1 Network A

To compare the equality of travel times over different scenarios, the Lorentz curves and Gini-coefficients of different scenarios are computed. Figure 4.23 shows the Lorentz curve and table 4.3 displays Gini-coefficient of three scenarios on network A. Note that since scenario 2 did not return different results than scenario 1, scenario 2 is not included.

It can be seen that the different scenarios do not have a large influence on equality. This can be explained by the fact that, as demonstrated in paragraph 4.2.1, the different scenarios had only limited impact on model outcome. Scenario 1 results in an almost perfectly equal distribution of travel times. This is due to the fact that all drivers choose the same route in this scenario. Scenarios 3 and 4 return a slightly more unequal outcome, as can be expected from implementing a tolling scheme. As explained in section 2.4, tolling diverts drivers from their individual optimal route to a slightly longer route, which benefits others. This way, inequality can rise.

4.3.2 Network B

Also on network B, scenario 2 did not return different results than scenario 1, and is thus excluded from this analysis. Figure 4.24 and table 4.4 show remarkable similarity between the Lorentz curves and Gini-coefficients of different scenarios, even though scenario 3 and 4 lead to a considerably lower ATT. This can be explained by the fact that in a Braess paradox, all drivers can benefit from an intervention in the network. If drivers spread out evenly over the two routes, travel times over these two routes are equal (see paragraph 2.1.2). On this network too, however, there is a small increase in inequality when tolling is applied (table 4.4). This can be explained following the same logic as above.

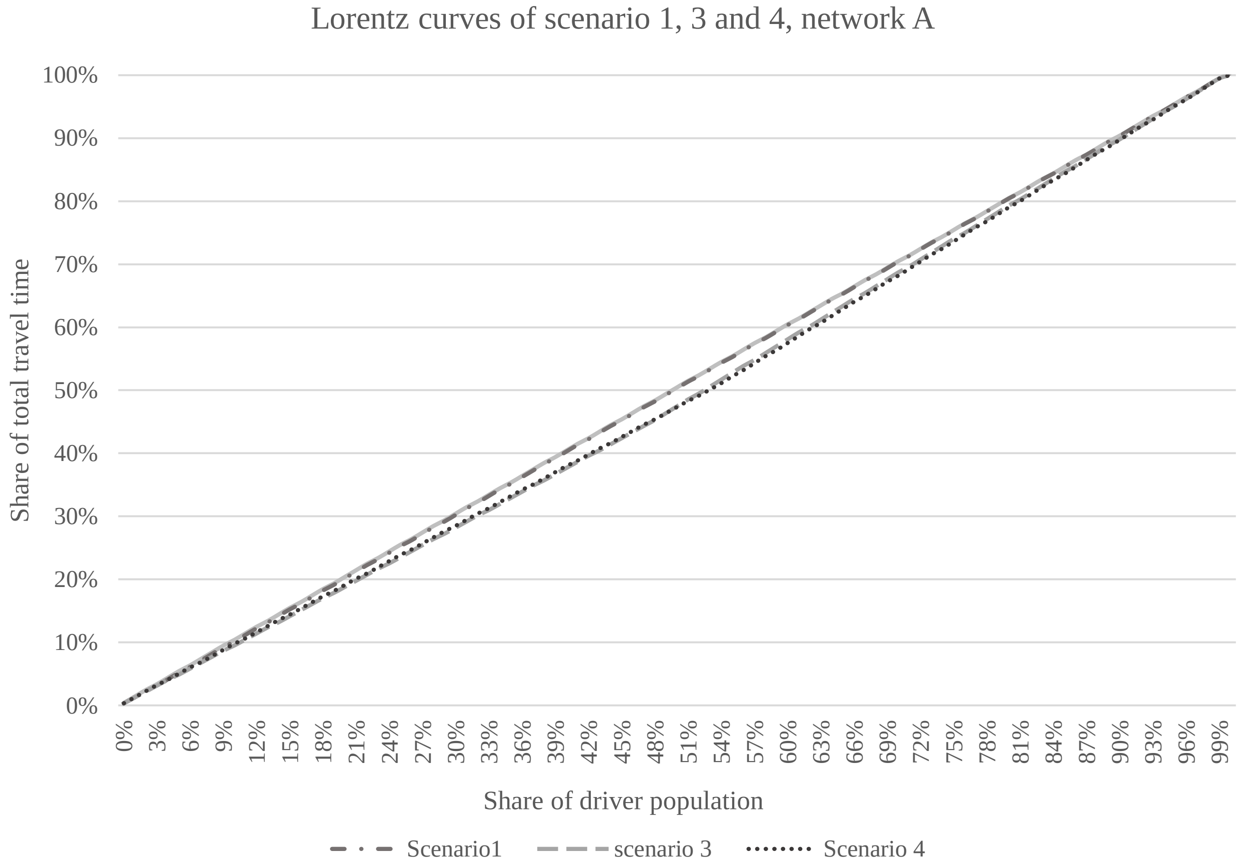


Figure 4.23: Lorentz curves of travel time distribution in scenario 1, 3, 4 on network A

Table 4.3: *Scenarios and Gini-coefficients on network A*

Scenario	Gini-coefficient
1	0,003
3	0,030
4	0,034

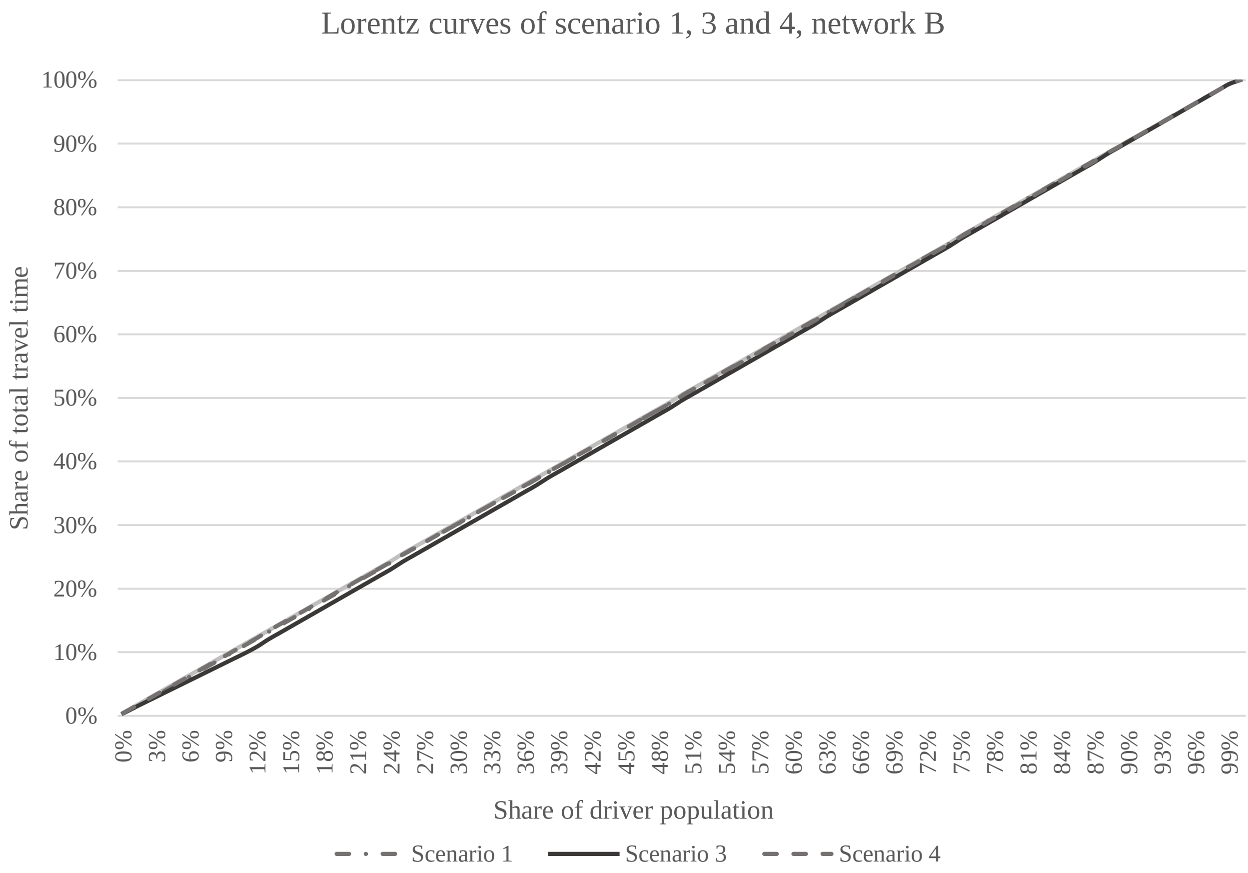


Figure 4.24: Lorentz curves of travel time distribution in scenario 1, 3, 4 on network B

Table 4.4: *Scenarios and Gini-coefficients on network B*

Scenario	Gini-coefficient
1	0,003
3	0,016
4	0,003

4.3.3 Sioux Falls

This paragraph examines the impact on travel time distribution of four scenarios. For scenario 1 and 2, the outcomes of the scenarios as described in paragraph 4.2.3 are used. For scenario 3 and 4, the outcome of the model is more diverse. Therefore, this section examines the equality of two different settings for scenario 3 and 4. Scenario 3a and 4a display the Lorentz curves of the β-R combination which returned the lowest ATT: $\beta = 10$, R $= 10^{-3.5}$ and $\alpha = 0{,}2$ for scenario 4a. Scenario 3b and 4b display the Lorentz curves of the β-R combination which returned the lowest ATT after using prediction: $\beta = 3.5$ and R $= 10^{-3}$, and $\alpha = 0{,}2$ for scenario 4b. This comparison is especially interesting since scenario 3b and scenario 4b returned near equal ATTs (2195,24 seconds and 2198,08 seconds, respectively).

Figure 4.25 displays the Lorentz curves for scenario 1, 2, 3a and 4a. For reference, a line of equal distribution has been added. It shows that between the different scenario's, there is little difference in equality. Table 4.5 shows that all measures lead to a slightly more equal distribution of travel times when compared to scenario 1. Of these measures, prediction leads to the most equal distribution, whereas the addition of toll causes a slightly less equal distribution in scenarios 3a and 4a. This corresponds to the reasoning in paragraph 4.3.1.

Figure 4.26 displays a similar image. For these setting of β and R, the distribution of travel times does not vary largely among scenario 3b and 4b. This shows that the application of tolls and prediction on this network does not have a large effect on equality. Also the Gini-coefficients (table 4.5) of these two scenarios are similar to the coefficients found above. The addition of prediction in scenario 4b leads to only a slightly more unequal distribution of travel times. This might indicate that in this scenario, due to prediction, some experience larger benefits than others.

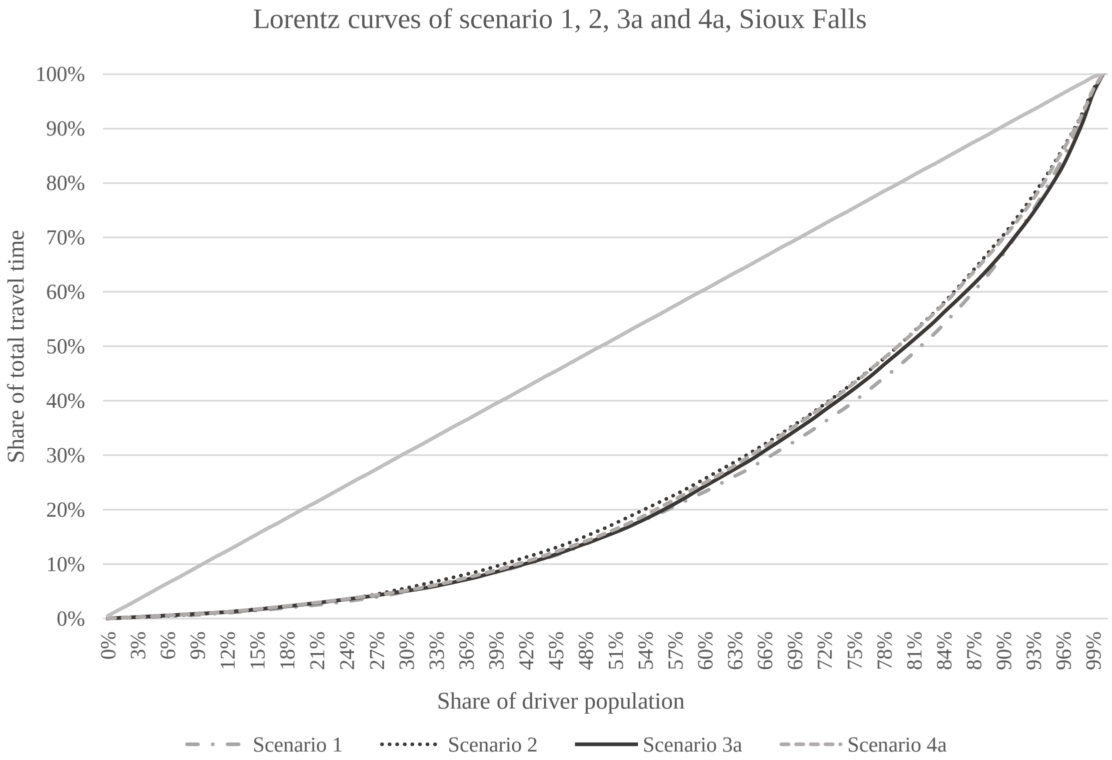


Figure 4.25: Lorentz curves of scenario 1, 2, 3a and 4a, Sioux Falls. The linear line from (0,0) to (100,100) is the line of equal distribution

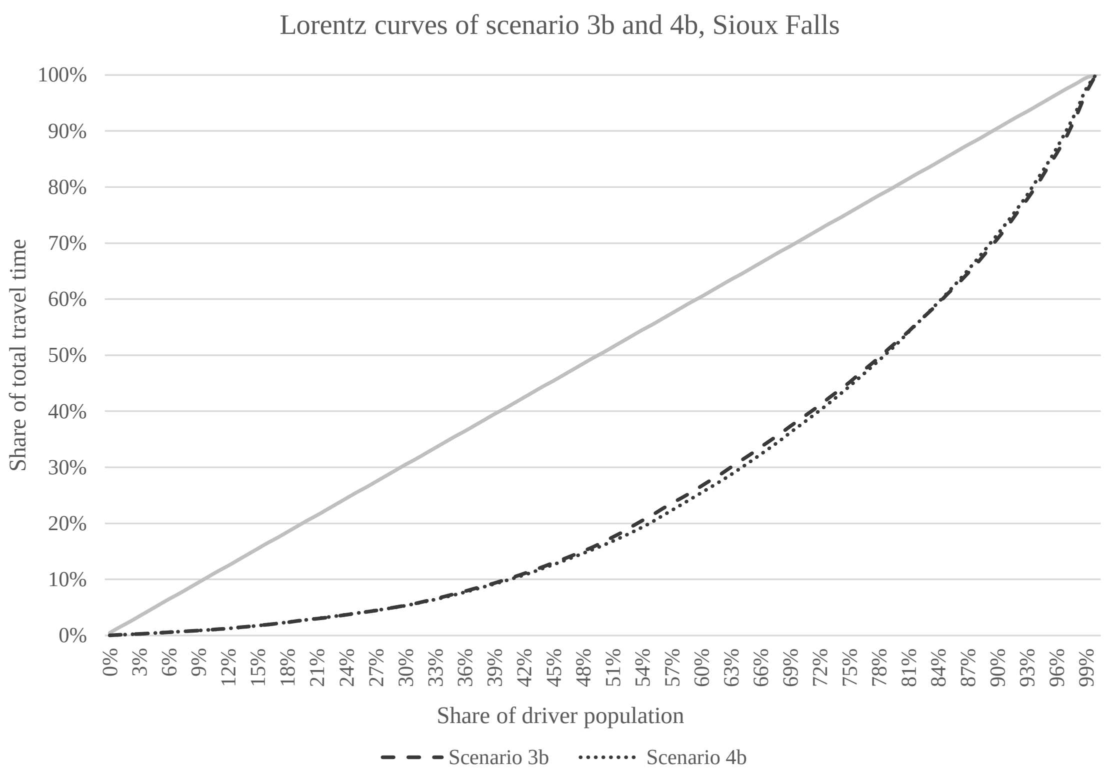


Figure 4.26: Lorentz curves of scenario 3b and 4b, Sioux Falls, with a line of equal distribution

Table 4.5: *Scenarios and Gini-coefficients on Sioux Falls network*

Scenario	Gini-coefficient
1	0,509
2	0,475
3a	0,497
3b	0,467
4a	0,482
4b	0,471

4.4 Fairness results

4.4.1 Network A

Figure 4.27 displays a histogram of the impact of scenarios 3 and 4 on the individual travel time of drivers. Note that scenario 2 has been left out of analysis for similar reasons as in the previous section. The histogram shows, for both scenarios, how high the impact of a measure was, and what share of all drivers was experienced this impact. The total amount of drivers in this analysis is 771. The lines display the cumulative share of drivers which have been affected by the scenario. For scenario 3, it shows that on almost half of all drivers, the measure had no impact. For the other half of the drivers, impact was mostly positive: it decreased the individual travel time up to 14%. Only a small share of the drivers has been negatively affected by the measure.

For scenario 4, the histogram shows a similar distribution of affected and non-affected drivers. For more than half of the drivers, the measure led to a 12% decrease in travel time. Only few drivers experienced an increase in travel time due to the measure. This pattern can be explained by the fact that on network A, there are only two options: the top road or the bottom road. In any scenario, the top road takes as much time as the bottom road in scenario 1. Thus, drivers opting for the top road experience no change in travel time, whereas drivers on the bottom road benefit from a less congested road.

4.4.2 Network B

On network B, almost all of the 803 drivers experience a decrease in travel time in both scenarios. In figure 4.28 both scenarios show a similar pattern, in which for the large majority of the drivers, individual travel time decreases with 12%. Outside the range of this histogram, only a few individual drivers experience a negative effect of tolling and/or prediction on their travel time. This corresponds with the Braess paradox, in which it is possible to improve the travel time of all traffic.

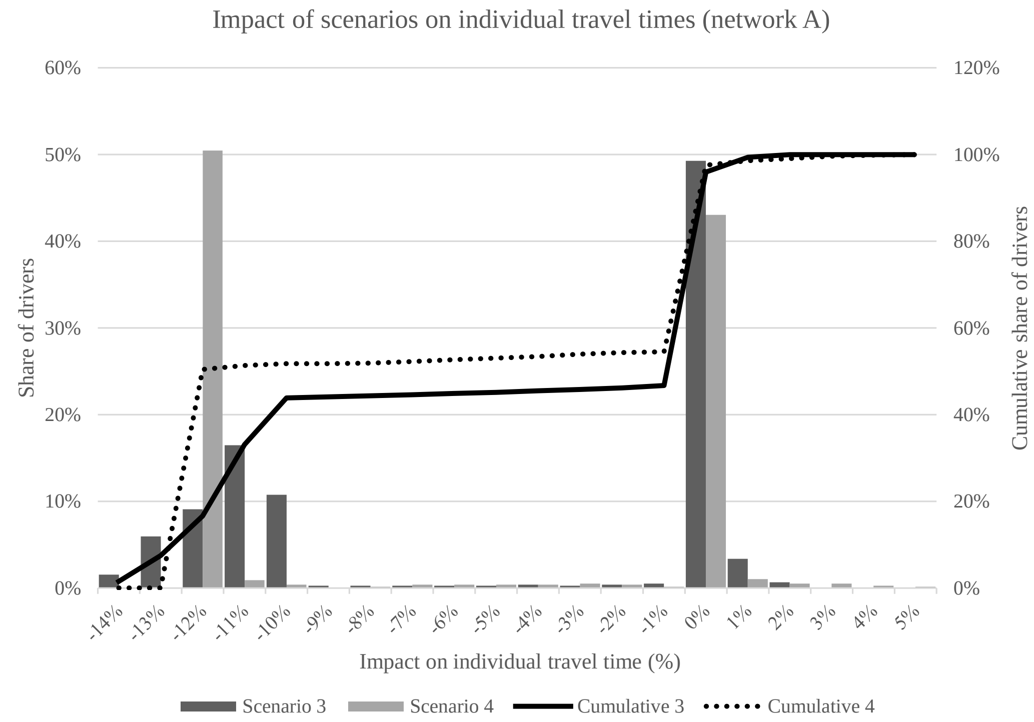


Figure 4.27: Impact of scenarios 3 and 4 on travel time (%) among the driver population, compared to scenario 1 on network A

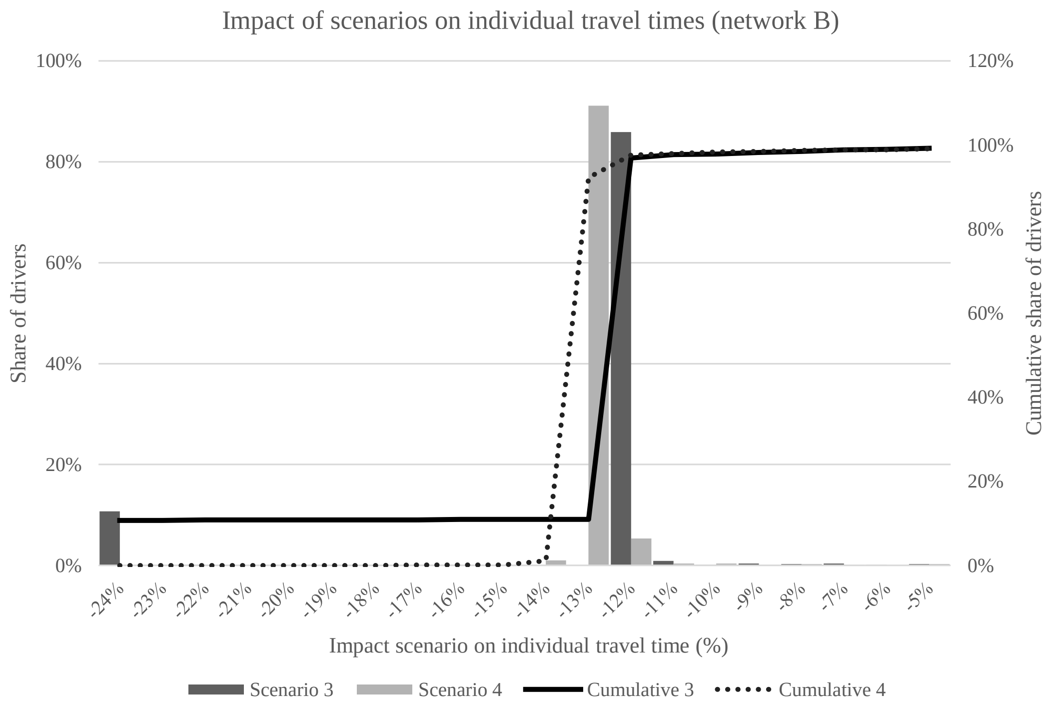


Figure 4.28: Impact of scenarios 3 and 4 on travel time (%) among the driver population, compared to scenario 1 on network B

4.4.3 Sioux Falls

Since the drivers on the Sioux Falls network have varying origins and destinations, it is no surprise that the impact of scenarios shows more variation than on networks A and B as well. Figure 4.28 displays for each scenario the impact of the scenarios on the individual travel times of drivers. The three scenarios show similar patterns (indeed, since scenarios 3b and 4b show equal patterns, these have been left out of this analysis). For all three scenarios, it can be seen that for more than half of the drivers, the measures decrease their individual travel time, up to about 90% for individual drivers.

Although most drivers benefit from the measures in these scenarios, the disadvantage for other drivers can be high. In scenario 2, for example, for about 20% of the drivers, the individual travel time shows an increase of more than 100%. In all scenarios, there are individual drivers who experience a travel time of more than ten times their original travel time. This shows that while the measures work at the benefit of most drivers, other drivers might be heavily disadvantaged. The reason for this might be that due to the interventions, normally little used roads become more congested. This, in turn, slows down users who heavily rely on such roads.

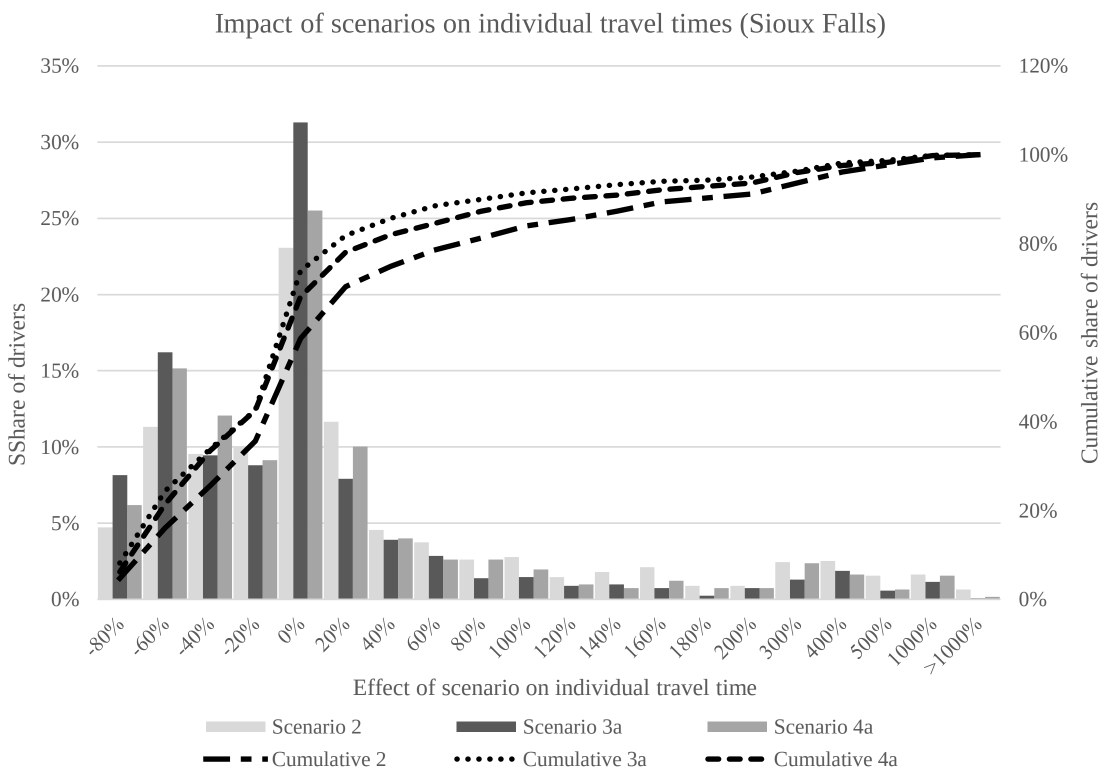


Figure 4.29: Impact of scenarios 3 and 4 on travel time (%) among the driver population, compared to scenario 1 on the Sioux Falls network

Chapter 5

Conclusion

After presenting the results, this chapter now continues with drawing conclusions from these results and this study as a whole. It does so by first providing a short overview of the study, after which the findings throughout the study are used to answer the research questions. The chapter finishes by revisiting the main objective of this study.

5.1 Study overview

A traffic system in which all drivers optimise their individual efficiency does not always lead to the most efficient use of a road network. Rather, ”selfish-routing” can lead to user equilibrium, which is not necessarily equal to the system optimum. This study refers to this phenomenon as the *inefficiency problem.* Modern communication and navigation devices arguably increase the tendency of a network to reach user equilibrium. However, new technologies also offer possibilities to route traffic in a more efficient way. This study explores some of these possibilities and has examined the effects of tolling and predictive methods on the performance of a traffic system. It did so by creating and assessing an agent-based traffic model, in which Δ-tolling and intention-based prediction were combined with the aim to lower the average travel time and thus improve the efficiency of the network. Consequently, the model outcome was tested using different scenarios on three networks. The study not only assessed average travel time, but also the effect that different scenarios had on the equality of the distribution of travel times and the impact that each scenario had on individual drivers. By doing so, this study aimed to find answers to the four research questions, which are addressed in the remainder of this chapter.

5.2 Answers to research questions

RQ 1. *What conceptual model using multi-temporal data can be used to improve the performance of a traffic system?*

In 2.3, it is found that when viewed from a game-theoretical perspective, the efficiency problem can be regarded as a routing game of which the result can be improved through two methods: cooperation among agents, or a change in payoffs. Even though *cooperative methods* can offer a solution for the efficiency problem in iterative situations with few agents, it is concluded that a *change of payoffs* is required to move a more complex traffic system towards system optimum. Tolling is a method to change these payoffs. Specifically, Δ-tolling is a promising tolling method to decrease average travel time and requires little knowledge on network characteristics. Current work on Δ-tolling only makes use of actual traffic information. However, it is acknowledged that the use of future traffic information in a Δ-tolling scheme can potentially improve the effect of the tolling scheme. Such future information can be predicted based on historical data, but it can also be predicted by making use of travelling intentions shared by drivers: intention-based prediction. Although current literature related to intention-based predictive methods is aimed at establishing *user equilibrium*, it is also recognized that prediction allows for application in *system-optimizing* methods. The conceptual model of this research therefore combines both Δ-tolling, current information and future information to improve system performance. As a method to acquire future information, this study uses intention-based prediction, in which drivers communicate their travel intentions with the network. By using intention-based predictions to set tolls according to the Δ-tolling method, multi-temporal data can thus be used to improve the performance of a traffic system.

RQ 2. *How can this conceptual model be implemented in an agent based traffic model?*

This study implements this conceptual model in an agent-based model using the GAMA platform. In this platform, a model is created which is populated by drivers, links and nodes. The model used in this study is a strongly simplified, mesoscopic traffic model, with a restricting set of assumptions. The model entails drivers, links and nodes. The drivers travel between nodes, and communicate their intended paths with the links, which keep track of the expected occupancy at any time. The drivers can then use this information to recompute the shortest path to their destination. In order to increase the performance of the traffic network, the links levy tolls which are computed according to the Δ-tolling method, using current and predicted occupancy. The model was run on three networks: two hypothetical networks which represented theoretical examples of the efficiency problem, and the simplified Sioux Falls network. Since this model is implemented in the GAMA platform - which allows for GIS data to serve as input data - the implemented model allows for the use of real-world network and traffic data as well.

RQ 3. *To what extent is the performance of the traffic system sensitive to different scenarios?*

The sensitivity of the model was tested in four scenarios using a range of settings for four parameters. Parameter α gave weight to predictive information over current information, and a share of smart agents was set to define the share of agents capable of making and using predictions. Parameters β and R define the tolling scheme. β sets the proportionality between the current delay on a link and the toll value, and R gives weight to the toll value computed in the previous time step. The effects of each scenario differed per network. For network A and B it was found that prediction on itself did not benefit the system performance when compared to scenario 1. On the Sioux Falls network, applying prediction led to a decrease in travel time. It was found that when the cost of a link depends for 50% on predictive information ($\alpha = 0.5$), the impact of prediction alone was at its largest. For this parameter setting, prediction lowered the average travel time (ATT) with 12%. In general, this effect increased when the share of smart agents was larger, with a population of only smart agents resulting in the lowest average travel time. This implies that intention-based prediction on itself can lower the average travel time of a traffic system, and that the benefit is highest when all drivers are smart. The outcomes of the predictive scenario 2 were assumed to be a user equilibrium benchmark for further comparisons.

Considering tolling, it was found that the effectiveness of the measure was dependent on the β-R combination applied. In networks A and B, this method tolling decreased the average travel time for all β-R combinations, in some cases up to 6% (network A) and 16% (network B) when compared to user equilibrium. In network A and B positive effects of tolling where found for all values of β and R, but on the Sioux Falls network, the value of R seems to be of higher importance than the value of β. There, the most positive results of tolling were found in a range of R=$10^{-2.5}$ to $R = 10^{-4}$. This means that when a tolling scheme is implemented, its effectiveness is defined by assigning a "correct" *weight* to previous tolls, rather than assigning a correct *height* to current tolls. These settings returned an ATT 18% lower than the user equilibrium benchmark found in scenario 2, and 28% lower than the ATT found in scenario 1. When comparing scenario 2 and 3, it appears that tolling (scenario 3) has a larger effect on the average travel time than prediction (scenario 2).

When Δ-tolling and prediction are combined, the ATT can decrease when compared to applying only tolling. For network A and B, this happens in all settings for β and R, additionally decreasing the average travel time up to 3% (network A) and 11% (network B). On network A, the minimum ATT found when using predictions, was the lowest ATT found for the entire network. For network B, the minimum ATT was found when using tolling alone. For the Sioux Falls network, for most – but not all – of the β-R settings, the ATT further decreases when applying predictive information, although on this network as well, the lowest ATT was found in scenario 3.

When the effect of toll was set out against the additional effect of prediction, network B showed a clear negative correlation between the two: in β-R combinations where toll led to a larger decrease in travel time, prediction led to a smaller decrease in travel time. This correlation also seemed to occur in the Sioux Falls network. An explanation for this finding is that when traffic is already navigated over a network in a rather efficient way (due to tolling), there is less potential for prediction to improve traffic flows; less than on a network where traffic is *not* routed efficiently. On driver-specific level, this means that a driver which already follows an efficient route due to tolling, is not probable to drive more efficiently if it is using prediction as well.

RQ 4. *What is the impact of the different scenarios on the equality and fairness of a traffic system when compared to user equilibrium?*

Considering the distribution of travel times, it can be concluded that for all three networks, the different scenarios did not have a large impact on equality when compared to user equilibrium. Both the Lorentz curves and GINI-coefficients of different scenarios on the same network lie close together. For network A, this can be explained by the fact that the change in average travel time–and thus agent behaviour–was relatively small (a maximum ATT decrease of 7%). For B this can be explained by the fact that all agents can benefit from an intervention in the Braess paradox. For the Sioux Falls network, the fact that equality does not vary largely among scenarios can also partially be explained by the fact that the majority of the drivers benefit from the measures. Even though it can be seen in figure 4.29 that some people do not benefit from the measure, this is a relatively small group. As such, the differences might not be sufficient to create a shift in equality.

For the changes in equality that *were* found, on network A and B it is found that tolling and prediction cause a slightly less equal distribution of travel times when compared to scenario 1 with no measures. On the Sioux Falls network, the distribution of travel times becomes slightly more equal when measures are implemented. Scenario 2, which was assumed to be the user equilibrium benchmark, leads to a more equal distribution than applying tolls only or applying tolls and prediction. This is to be expected from literature, since prediction enables individual agents to optimise their own route quicker, and thus establish a user equilibrium–where all individual travel times are minimalised–more easily.

Considering fairness, networks A and B show different patterns than Sioux Falls. For network A, tolling and prediction seem to have either no effect on individual driver time, or to decrease the individual driving time with about 12%. This is a logical result, since network A only offers two route options. Since the top route has the same travel time as the bottom route when it is completely congested, the traffic that takes the top route experiences no benefit when tolling or prediction is implemented. Traffic that takes the bottom roads, benefits from the less congested road. As a result, traffic is separated in a group which benefits, and a group on which the measure has no effect.

On network B, the interventions lead to a similar decrease in individual travel time for almost all drivers. This too is to be expected, since, following the Braess paradox, both load-dependent roads become less congested, and traffic spreads out over the top and bottom road. As a result, all traffic experiences similar travel time traveling from origin to destination.

On the Sioux Falls network, the pattern is more diverse. It can be seen here that in any scenario, more than half of the drivers experiences a decrease in travel time due to interventions. However, some drivers experience a high increase in travel time. Since this study did not analyse the discrepancy of impact among drivers into depth, it is an opportunity for further research to explore whether and how the impact of interventions relates to driver characteristics, like origin, destination and distance travelled. Further knowledge on the individual impact on drivers can also aid in explaining the effects of tolling and prediction on equality. Although the reasoning above might partially explain the rather small change in equality, the current study does not provide a full answer as to why the interventions have only limited impact on equality. In order to completely answer this question, further research on the impact of interventions on individual drivers is required.

By answering these four research questions, this study has created an agent-based model and examined the effect of multi-temporal network data on system performance. As such, this study has carried out the objective stated in the introductory chapter. However, the study does have limitations, and offers opportunities for further research. These will be discussed in the following chapter.

Chapter 6

Discussion

This section provides a discussion and reflection on the conclusions presented in the previous section. First, this study and its conclusions are related to existing literature. Afterwards, limitations of this study and opportunities for future research are presented. Besides a discussion of the current study, this section also provides a discussion on the interventions proposed in this study: tolling and prediction. It first addresses the potential added value of implementation of such interventions, after which it also considers limitations and challenges of tolling and (intention-based) prediction.

6.1 Link to existing literature

This study mainly builds on the earlier proposed methods of Δ-tolling and intention-based prediction. As such, it contributes to existing literature in three ways: it not only adds to research on both of these methods, but also provides new insights in the combination of tolling and prediction. Relating to existing literature on Δ-tolling, this study supports the results found in Sharon, Hanna, et al. (2017) and Sharon, Levin, et al. (2017). Both studies show the potential effect of Δ-tolling for certain β-R settings. Specifically the patterns this study found in the effect of tolling for different settings of β and R (figure 4.19 and figure 4.20) are comparable to the patterns presented in Sharon, Hanna, et al. (2017). This shows that Δ-tolling can indeed be an effective method to improve system performance.

Relating to current work on intention-based prediction, this study has shown the effectivity of applying predictive information to decrease the average travel time. As such, even though earlier works focussed mainly on establishing a user equilibrium (e.g., Mahajan et al., 2019), the method is also capable of improving the system efficiency besides individual efficiency. This is also found by De Juncker et al. (2018). Moreover, this study has shown that the optimal effect of prediction is not always found when relying fully on predictive information: rather, combining current and future information might lead to a better performance.

By combining the two methods this study has found that for given settings of β-R, applying prediction can improve the performance of a traffic system. Not only does this contribute to an existing gap in literature on Δ-tolling (Sharon, Levin, et al., 2017), it also demonstrates that intention-based prediction is applicable in an explicit system-optimising design.

6.2 Limitations and opportunities for future network.

Besides these additions to existing work, this study also has a set of limitations. These limitations mainly follow from the scope of this study. Since this research did not aim to model a real-world network, the workings of the proposed methods on such networks remain to be explored. This study defined a set of limiting assumptions. This enabled a specified analysis of the model using a set of parameters, but it also restricted the possibility to "translate" the results of this study to actual traffic situations. These limitations relate to assumptions about the network and assumptions about the drivers.

Firstly, concerning the network, this study only explored the workings of the method on three networks, two of which were hypothetical, and one of which was a simplified representation of a real world network. The traffic demand on these networks was assumed as set. Moreover, it was assumed that the congestion on the links was defined by static latency functions. This is a rather unrealistic assumption, since dynamic factors such as weather conditions and road decay can have an influence on the capacity of roads. Future work might therefore test the workings of the model on actual networks with differing traffic demands and dynamic latency functions, taking changing conditions into account.

Secondly, this study also assumed values of β and R to be static and the same across the network. Earlier work has shown that this might not be an optimal strategy. Mirzaei et al. (2018) have shown that setting an individual R-value for each link can lead to a better system performance than assuming a generalized value of R. A reinforcement learning algorithm can be used to set these values of R (Mirzaei et al., 2018). Reinforcement learning might also be applied to optimise predictions on future network occupancy.

Concerning the drivers, this study assumed that all drivers are equal, driving in equal vehicles, having equal motivations. This is, obviously, a highly restrictive assumption. Actual traffic consists of a variety of users, “smart” or not “smart”, in a variety of vehicles, with a variety of individual behaviour. To properly explore the effects of implementing intervening measures, it is imperative that future research addresses these issues, and that the impact of interventions is tested for diverse populations of traffic users. This can provide more insight in why some drivers experience positive effects of an intervention, while others might be badly affected.

A specific driver-related limitation of this study is the concept of value of time. As shortly mentioned in the paragraph 3.2.5, the value of time was assumed to be equal for all drivers. This is not only rather unrealistic, it also has effect on model outcome. In reality, different drivers in different circumstances might value time differently. An example: for someone travelling to the airport, a delay might be much costlier, whereas someone travelling home is slightly annoyed, but otherwise not very affected by a delay. Future research should thus not only consider traffic users with different characteristics, but also with different values of time.

6.3 Added value and challenges of implementing tolling and prediction

As stated in the introduction, the motivation of this study was found in the increasing and expected usage of autonomous and other connected vehicles. Supporting earlier work on Δ-tolling and intention-based prediction, this study has shown that indeed, these methods have the potential to improve the efficiency of a traffic system with connected vehicles. In addition, this study has shown that the combination of these to concepts can benefit system performance, too.

This study found that prediction can decrease the average travel time in a traffic system if used in combination with certain β-R settings. This has two important implications, which are demonstrated by scenario 3 and 4 on the Sioux Falls network. In scenario 3, an ATT of 2195 seconds was found, for β= 10 and R = $10^{-3.5}$. When combining tolling and prediction, scenario 4 returned a minimum ATT of 2.198 seconds for β = 3,5 and R = 10^{-3}. First, this implies that when tolling is implemented with sub-optimal settings, prediction can be applied as an additional measure to improve system performance. This might be of added value when optimal tolling settings of a specific network are unknown.

Secondly, these two scenarios show that β can be lowered, and that the resulting sub-optimality can largely be corrected for by applying prediction. This means that when prediction is applied, in order to install an effective tolling mechanism, the toll values can be lower than in a situation in which no prediction is possible. As such, a less-intrusive tolling mechanism which yields a similar result as an optimal tolling mechanism can be implemented.

Despite the effect of tolling and prediction demonstrated in this study, there are a few caveats when implementing these interventions on a real traffic network. As pointed out in section 2.4, solving the efficiency problem is not only a matter of finding an effective intervention. For example, an intervention can cause the traffic system to become less equal. Then, intervening in a traffic system also arises questions about how fair an intervention is. The analyses in section 4.3 and 4.4 show that both Δ-tolling and intention-based prediction have only little impact on equality in the traffic system and benefit most traffic users. Nevertheless, some users are badly disadvantaged by both measures. Thus, no clear answer on the fairness-question is given. Besides finding how the impact is experienced by different types of traffic participants, any intervention in traffic should also consider the question to what extent an unfair distribution of impact or travel times weighs against an improvement in system efficiency.

Another challenge, specifically of intention-based prediction, relates to privacy. A traffic system might be better off by using travel plans with links, but are people willing to share their personal plans to improve the efficiency of the system? This might sound like a daunting prospect of applying intention-based prediction. However, the methods proposed in this and earlier study do allow for a privacy-considerate intervention. After all, if intention-based prediction was to be implemented in real life, there would be no need for any entity–besides a driver's navigation system–to know the entire travel plan of an individual. A navigation system might only need to update the links along its path to update their occupancy at the expected time of presence. As such, no link knows either the origin or destination of an agent: it only knows, at any time, how occupied it will be.

This study has demonstrated the workings of measures to improve the efficiency of a traffic system, and adds to existing literature on other of such measures. Besides that, by showing the effects of the measures, the study can also be of help for policy makes in the field of transportation. It may be clear, however, that before intervening in traffic in any way, there are policy questions left to be answered. Yet, these questions are also to be answered when there is no intervention implemented explicitly. It follows from theory that a system with no intervention eventually leads to user equilibrium, and thus to inefficient network use. In lack of realistic and effective traffic intervention, these questions might not have been relevant in the past. Yet, as the wide-spread use of smart vehicles seems near, there may soon be a demand for an explicit choice in the way traffic is managed.

References

Adler, J. L., & Blue, V. J. (2002). A cooperative multi-agent transportation management and route guidance system. *Transportation Research Part C: Emerging Technologies*, *10*(5-6), 433–454. doi: 10.1016/S0968-090X(02)00030-X

Bagloee, S. A., Tavana, M., Asadi, M., & Oliver, T. (2016). Autonomous vehicles: challenges, opportunities, and future implications for transportation policies. *Journal of modern transportation*, *24*(4), 284–303.

Bar-Gera, H., Hellman, F., & Patriksson, M. (2013). Computational precision of traffic equilibria sensitivities in automatic network design and road pricing. *Procedia-Social and Behavioral Sciences*, *80*, 41–60.

Ben-Akiva, M., De Palma, A., & Isam, K. (1991). Dynamic network models and driver information systems. *Transportation Research Part A: General*, *25*(5), 251–266.

Braess, D. (1968). Über ein paradoxon aus der verkehrsplanung. *Unternehmensforschung*, *12*(1), 258–268.

Braess, D., Nagurney, A., & Wakolbinger, T. (2005). On a paradox of traffic planning. *Transportation science*, *39*(4), 446–450.

Button, K. (2004). The rationale for road pricing: standard theory and latest advances. *Road pricing: theory and evidence*, *9*, 3–25.

Catalin, D., Dauphin-Tanguy, G., & Popescu, D. (2012). Macroscopic modeling of road traffic by using hydrodynamic flow models. In *2012 20th mediterranean conference on control & automation (med)* (pp. 42–47).

Chiu, Y.-C., Bottom, J., Mahut, M., Paz, A., Balakrishna, R., Waller, T., & Hicks, J. (2011). Dynamic traffic assignment: A primer. *Dynamic Traffic Assignment: A Primer*.

Claes, R. (2015). *Anticipatory vehicle routing.* [PhD thesis]. Belgium: Katholieke Universiteit Leuven.

Claes, R., Holvoet, T., & Weyns, D. (2011). A decentralized approach for anticipatory vehicle routing using delegate multiagent systems. *IEEE Transactions on Intelligent Transportation Systems*, *12*(2), 364–373.

De Smith, M. J., Goodchild, M. F., & Longley, P. A. (2018). *Geospatial Analysis, A comprehensive Guide to Principles, Techniques and Software Tools* (6th ed.). Spatialanalysisonline.com.

De Juncker, M. A. M., Phillipson, F., Bruijns, L. A. M., & Sangers, A. (2018). Optimising routing in an agent-centric synchromodal network with shared information. In R. Cerulli, A. Raiconi, & S. Voß (Eds.), *Computational logistics. ICCL 2018. lecture notes in computer science, vol 11184* (pp. 316–330). Cham: Springer.

Dong, J., Mahmassani, H. S., Erdoğan, S., & Lu, C.-C. (2011). State-dependent pricing for real-time freeway management: Anticipatory versus reactive strategies. *Transportation Research Part C: Emerging Technologies*, *19*(4), 644–657.

Dresner, K., & Stone, P. (2008). A multiagent approach to autonomous intersection management. *Journal of artificial intelligence research*, *31*, 591–656.

Ehlert, P. A., & Rothkrantz, L. J. (2001). Microscopic traffic simulation with reactive driving agents. In *2001 ieee intelligent transportation systems proceedings* (pp. 860–865).

Grimm, V., Berger, U., Bastiansen, F., Eliassen, S., Ginot, V., Giske, J., ... DeAngelis, D. L. (2006). A standard protocol for describing individual-based and agent-based models. *Ecological Modelling*, *198*(1), 115 - 126. doi: https://doi.org/10.1016/j.ecolmodel.2006.04.023

Grimm, V., Berger, U., DeAngelis, D. L., Polhill, J. G., Giske, J., & Railsback, S. F. (2010). The odd protocol: A review and first update. *Ecological Modelling*, *221*(23), 2760 - 2768. doi: https://doi.org/10.1016/j.ecolmodel.2010.08.019

Hashemi, H., & Abdelghany, K. F. (2016). Real-time traffic network state estimation and prediction with decision support capabilities: Application to integrated corridor management. *Transportation Research Part C: Emerging Technologies*, *73*, 128–146.

Helbing, D. (2001). Traffic and related self-driven many-particle systems. *Reviews of modern physics*, *73*(4), 1067.

Helbing, D., Schönhof, M., Stark, H. U., & Holyst, J. A. (2005). How individuals learn to take turns: Emergence of alternating cooperation in a congestion game and the prisoner's dilemma. *Advances in Complex Systems*, *8*(1), 87–116. doi: 10.1142/S0219525905000361

IT University of Copenhagen. (2018, September 17). *Tim Roughgarden: the Price of Anarchy* *[video file].* Retrieved from `https://www.youtube.com/watch?v=-w7ddIRFfqM` (Accessed on 24-10-2019)

Kaddoura, I., & Nagel, K. (2019). Congestion pricing in a real-world oriented agent-based simulation context. *Research in Transportation Economics*, *74*, 40–51. doi: 10.1016/J.RETREC.2019.01.002

Klein, I., & Ben-Elia, E. (2018). Emergence of cooperative route-choice: A model and experiment of compliance with system-optimal ATIS. *Transportation Research Part F: Traffic Psychology and Behaviour*, *59*, 348–364. doi: 10.1016/j.trf.2018.09.007

Klein, I., Levy, N., & Ben-Elia, E. (2018). An agent-based model of the emergence of cooperation and a fair and stable system optimum using ATIS on a simple road network. *Transportation Research Part C: Emerging Technologies*, *86*, 183–201. doi: 10.1016/j.trc.2017.11.007

LeBlanc, L. J., Morlok, E. K., & Pierskalla, W. P. (1975). An efficient approach to solving the road network equilibrium traffic assignment problem. *Transportation research*, *9*(5), 309–318.

Levy, N., & Ben-Elia, E. (2016). Emergence of System Optimum: A Fair and Altruistic Agent-based Route-choice Model. In *Procedia computer science* (Vol. 83, pp. 928–933). Elsevier. doi: 10.1016/j.procs.2016.04.187

Levy, N., Klein, I., & Ben-Elia, E. (2018). Emergence of cooperation and a fair system optimum in road networks: A game-theoretic and agent-based modelling approach. *Research in Transportation Economics*, *68*, 46–55. doi: 10.1016/j.retrec.2017.09.010

Liao, H., Dai, G., & Chen, R. (2012). Research and simulation on the upper bound of the price of network anarchy. In *2012 9th international conference on fuzzy systems and knowledge discovery* (pp. 2759–2763). IEEE. doi: 10.1109/FSKD.2012.6233759

Maerivoet, S., & De Moor, B. (2005). Transportation planning and traffic flow models. *arXiv preprint physics/0507127*.

Mahajan, N., Hegyi, A., Hoogendoorn, S. P., & van Arem, B. (2019). Design analysis of a decentralized equilibrium-routing strategy for intelligent vehicles. *Transportation Research Part C: Emerging Technologies*, *103*, 308–327. doi: 10.1016/j.trc.2019.03.028

Mirzaei, H., Sharon, G., Boyles, S., Givargis, T., & Stone, P. (2018). Link-based parameterized micro-tolling scheme for optimal traffic management (Extended Abstract). In *17th international conference on autonomous agents and multiagent systems* (pp. 2013–2015).

Monnot, B., Benita, F., & Piliouras, G. (2017). How bad is selfish routing in practice? *Computing Research Repository*, *1703*.

Papadimitriou, C. (2001). Algorithms, games, and the internet. In *Proceedings of the thirty-third annual acm symposium on theory of computing* (pp. 749–753). New York, NY, USA: ACM. doi: 10.1145/380752.380883

Pigou, A. (1920). *The economics of welfare* (1st ed.). MacMillan.

Roughgarden, T. (2002). How unfair is optimal routing? In *Proceedings of the thirteenth annual acm-siam symposium on discrete algorithms* (pp. 203–204).

Roughgarden, T. (2003). The price of anarchy is independent of the network topology. *Journal of Computer and System Sciences*, *67*(2), 341–364.

Roughgarden, T., & Tardos, É. (2002). How bad is selfish routing? *Journal of the ACM*, *49*(2), 236–259. doi: 10.1145/506147.506153

Sharon, G., Hanna, J. P., Rambha, T., Levin, M. W., Albert, M., Boyles, S. D., & Stone, P. (2017). Real-time adaptive tolling scheme for optimized social welfare in traffic networks. In *Proceedings of the 16th conference on autonomous agents and multiagent systems* (pp. 828–836).

Sharon, G., Levin, M. W., Hanna, J. P., Rambha, T., Boyles, S. D., & Stone, P. (2017). Network-wide adaptive tolling for connected and automated vehicles. *Transportation Research Part C: Emerging Technologies*, *84*, 142–157.

Shlegeris, B. (2020). *Gini coefficient calculator.* Retrieved from `http://shlegeris.com/gini` (Accessed on 17-02-2020)

Taillandier, P., Gaudou, B., Grignard, A., Huynh, Q.-N., Marilleau, N., Caillou, P., … Drogoul, A. (2019). Building, composing and experimenting complex spatial models with the gama platform. *GeoInformatica*, *23*(2), 299–322.

Transportation Networks for Research Core Team. (2020). *Transportation networks for research.* Retrieved from `https://github.com/bstabler/TransportationNetworks` (Accessed on 20-01-2020)

Tsekeris, T., & Voß, S. (2009). Design and evaluation of road pricing: state-of-the-art and methodological advances. *NETNOMICS: Economic Research and Electronic Networking*, *10*(1), 5–52.

Varga, L. Z. (2014). On intention-propagation-based prediction in autonomously self-adapting navigation. In *2014 ieee eighth international conference on self-adaptive and self-*

organizing systems workshops (pp. 38–43).

Wardrop, J. G. (1952). Road paper. some theoretical aspects of road traffic research. *Proceedings of the institution of civil engineers*, *1*(3), 325–362.

Weyns, D., Holvoet, T., & Helleboogh, A. (2007). Anticipatory vehicle routing using delegate multi-agent systems. In *Ieee conference on intelligent transportation systems, proceedings, itsc* (pp. 87–93). doi: 10.1109/ITSC.2007.4357809

Zhang, J., Pourazarm, S., Cassandras, C. G., & Paschalidis, I. C. (2016a). Data-driven estimation of origin-destination demand and user cost functions for the optimization of transportation networks. *Computing Research Repository*, *1610*.

Zhang, J., Pourazarm, S., Cassandras, C. G., & Paschalidis, I. C. (2016b). The price of anarchy in transportation networks by estimating user cost functions from actual traffic data. In *2016 ieee 55th conference on decision and control (cdc)* (pp. 789–794).

Appendix

A. Sioux Falls network data

1. First, in order to scale the network to a more realistic scale in, all coordinates are divided by 50.
2. Then, the length of each link is computed in ArcGIS Pro 2.4.0
3. Then, this length is divided by the free flow travel time in seconds, to arrive at the speed limit in meters per second

Figure A.1 displays the network data.

Link_ID	Init_node	Term_node	Capacity	Length	Free_Flow_Time	FFT_Seconds	B	Power	Speed_limit	Toll	Type	Init_X	Init_Y	Term_X	Term_Y	I_X_ADAP	I_Y_ADAP	T_X_ADAP	T_Y_ADAP
1	1	2	25900.20064	6	6	360	0.15	4	15	0	1	50000	510000	320000	510000	1000	10200	6400	10200
3	2	1	25900.20064	6	6	360	0.15	4	15	0	1	320000	510000	50000	510000	6400	10200	1000	10200
37	12	13	25900.20064	3	3	180	0.15	4	30	0	1	50000	320000	50000	50000	1000	6400	1000	1000
38	13	12	25900.20064	3	3	180	0.15	4	30	0	1	50000	50000	50000	320000	1000	1000	1000	6400
2	1	3	23403.47319	4	4	240	0.15	4	5.83	0	1	50000	510000	50000	440000	1000	10200	1000	8800
5	3	1	23403.47319	4	4	240	0.15	4	5.83	0	1	50000	440000	50000	510000	1000	8800	1000	10200
7	3	12	23403.47319	4	4	240	0.15	4	10	0	1	50000	440000	50000	320000	1000	8800	1000	6400
18	7	18	23403.47319	2	2	120	0.15	4	10	0	1	420000	380000	420000	320000	8400	7600	8400	6400
35	12	3	23403.47319	4	4	240	0.15	4	10	0	1	50000	320000	50000	440000	1000	6400	1000	8800
54	18	7	23403.47319	2	2	120	0.15	4	10	0	1	420000	320000	420000	380000	8400	6400	8400	7600
56	18	20	23403.47319	4	4	240	0.15	4	23.99	0	1	420000	320000	320000	50000	8400	6400	6400	1000
60	20	18	23403.47319	4	4	240	0.15	4	23.99	0	1	320000	50000	420000	320000	6400	1000	8400	6400
50	16	18	19679.89671	3	3	180	0.15	4	11.11	0	1	320000	320000	420000	320000	6400	6400	8400	6400
55	18	16	19679.89671	3	3	180	0.15	4	11.11	0	1	420000	320000	320000	320000	8400	6400	6400	6400
9	4	5	17782.7941	2	2	120	0.15	4	15	0	1	130000	440000	220000	440000	2600	8800	4400	8800
11	5	4	17782.7941	2	2	120	0.15	4	15	0	1	220000	440000	130000	440000	4400	8800	2600	8800
6	3	4	17110.52372	4	4	240	0.15	4	6.67	0	1	50000	440000	130000	440000	1000	8800	2600	8800
8	4	3	17110.52372	4	4	240	0.15	4	6.67	0	1	130000	440000	50000	440000	2600	8800	1000	8800
45	15	19	14564.75315	3	3	180	0.15	4	11.11	0	1	220000	190000	320000	190000	4400	3800	6400	3800
57	19	15	14564.75315	3	3	180	0.15	4	11.11	0	1	320000	190000	220000	190000	6400	3800	4400	3800
25	9	10	13915.78842	3	3	180	0.15	4	6.67	0	1	220000	380000	220000	320000	4400	7600	4400	6400
26	10	9	13915.78842	3	3	180	0.15	4	6.67	0	1	220000	320000	220000	380000	4400	6400	4400	7600
28	10	15	13512.00155	6	6	360	0.15	4	7.22	0	1	220000	320000	220000	190000	4400	6400	4400	3800
43	15	10	13512.00155	6	6	360	0.15	4	7.22	0	1	220000	190000	220000	320000	4400	3800	4400	6400
13	5	9	10000	5	5	300	0.15	4	4	0	1	220000	440000	220000	380000	4400	8800	4400	7600
23	9	5	10000	5	5	300	0.15	4	4	0	1	220000	380000	220000	440000	4400	7600	4400	8800
27	10	11	10000	5	5	300	0.15	4	6	0	1	220000	320000	130000	320000	4400	6400	2600	6400
32	11	10	10000	5	5	300	0.15	4	6	0	1	130000	320000	220000	320000	2600	6400	4400	6400
46	15	22	9599.180565	3	3	180	0.15	4	6.67	0	1	220000	190000	220000	130000	4400	3800	4400	2600
67	22	15	9599.180565	3	3	180	0.15	4	6.67	0	1	220000	130000	220000	190000	4400	2600	4400	3800
17	7	8	7841.81131	3	3	180	0.15	4	11.11	0	1	420000	380000	320000	380000	8400	7600	6400	7600
20	8	7	7841.81131	3	3	180	0.15	4	11.11	0	1	320000	380000	420000	380000	6400	7600	8400	7600
49	16	17	5229.910063	2	2	120	0.15	4	10	0	1	320000	320000	320000	260000	6400	6400	6400	5200
52	17	16	5229.910063	2	2	120	0.15	4	10	0	1	320000	260000	320000	320000	6400	5200	6400	6400
65	21	22	5229.910063	2	2	120	0.15	4	13.33	0	1	220000	50000	220000	130000	4400	1000	4400	2600
69	22	21	5229.910063	2	2	120	0.15	4	13.33	0	1	220000	130000	220000	50000	4400	2600	4400	1000
41	14	15	5127.526119	5	5	300	0.15	4	6	0	1	130000	190000	220000	190000	2600	3800	4400	3800
44	15	14	5127.526119	5	5	300	0.15	4	6	0	1	220000	190000	130000	190000	4400	3800	2600	3800
39	13	24	5091.256152	4	4	240	0.15	4	6.67	0	1	50000	50000	130000	50000	1000	1000	2600	1000
74	24	13	5091.256152	4	4	240	0.15	4	6.67	0	1	130000	50000	50000	50000	2600	1000	1000	1000
73	23	24	5078.508436	2	2	120	0.15	4	13.33	0	1	130000	130000	130000	50000	2600	2600	2600	1000
76	24	23	5078.508436	2	2	120	0.15	4	13.33	0	1	130000	50000	130000	130000	2600	1000	2600	2600
63	20	22	5075.697193	5	5	300	0.15	4	8.54	0	1	320000	50000	220000	130000	6400	1000	4400	2600
68	22	20	5075.697193	5	5	300	0.15	4	8.54	0	1	220000	130000	320000	50000	4400	2600	6400	1000
62	20	21	5059.91234	6	6	360	0.15	4	5.56	0	1	320000	50000	220000	50000	6400	1000	4400	1000
64	21	20	5059.91234	6	6	360	0.15	4	5.56	0	1	220000	50000	320000	50000	4400	1000	6400	1000
21	8	9	5050.193156	10	10	600	0.15	4	3.33	0	1	320000	380000	220000	380000	6400	7600	4400	7600
24	9	8	5050.193156	10	10	600	0.15	4	3.33	0	1	220000	380000	320000	380000	4400	7600	6400	7600
22	8	16	5045.822583	5	5	300	0.15	4	4	0	1	320000	380000	320000	320000	6400	7600	6400	6400
47	16	8	5045.822583	5	5	300	0.15	4	4	0	1	320000	320000	320000	380000	6400	6400	6400	7600
59	19	20	5002.607563	4	4	240	0.15	4	11.67	0	1	320000	190000	320000	50000	6400	3800	6400	1000
61	20	19	5002.607563	4	4	240	0.15	4	11.67	0	1	320000	50000	320000	190000	6400	1000	6400	3800
70	22	23	5000	4	4	240	0.15	4	7.5	0	1	220000	130000	130000	130000	4400	2600	2600	2600
72	23	22	5000	4	4	240	0.15	4	7.5	0	1	130000	130000	220000	130000	2600	2600	4400	2600
30	10	17	4993.510694	8	8	480	0.15	4	4.86	0	1	220000	320000	320000	260000	4400	6400	6400	5200
51	17	10	4993.510694	8	8	480	0.15	4	4.86	0	1	320000	260000	220000	320000	6400	5200	4400	6400
4	2	6	4958.180928	5	5	300	0.15	4	4.67	0	1	320000	510000	320000	440000	6400	10200	6400	8800
14	6	2	4958.180928	5	5	300	0.15	4	4.67	0	1	320000	440000	320000	510000	6400	8800	6400	10200
12	5	6	4947.995469	4	4	240	0.15	4	8.33	0	1	220000	440000	320000	440000	4400	8800	6400	8800
15	6	5	4947.995469	4	4	240	0.15	4	8.33	0	1	320000	440000	220000	440000	6400	8800	4400	8800
42	14	23	4924.790605	4	4	240	0.15	4	5	0	1	130000	190000	130000	130000	2600	3800	2600	2600
71	23	14	4924.790605	4	4	240	0.15	4	5	0	1	130000	130000	130000	190000	2600	2600	2600	3800
10	4	11	4908.82673	6	6	360	0.15	4	6.67	0	1	130000	440000	130000	320000	2600	8800	2600	6400
31	11	4	4908.82673	6	6	360	0.15	4	6.67	0	1	130000	320000	130000	440000	2600	6400	2600	8800
33	11	12	4908.82673	6	6	360	0.15	4	4.44	0	1	130000	320000	50000	320000	2600	6400	1000	6400
36	12	11	4908.82673	6	6	360	0.15	4	4.44	0	1	50000	320000	130000	320000	1000	6400	2600	6400
16	6	8	4898.587646	2	2	120	0.15	4	10	0	1	320000	440000	320000	380000	6400	8800	6400	7600
19	8	6	4898.587646	2	2	120	0.15	4	10	0	1	320000	380000	320000	440000	6400	7600	6400	8800
66	21	24	4885.357564	3	3	180	0.15	4	10	0	1	220000	50000	130000	50000	4400	1000	2600	1000
75	24	21	4885.357564	3	3	180	0.15	4	10	0	1	130000	50000	220000	50000	2600	1000	4400	1000
34	11	14	4876.508287	4	4	240	0.15	4	10.83	0	1	130000	320000	130000	190000	2600	6400	2600	3800
40	14	11	4876.508287	4	4	240	0.15	4	10.83	0	1	130000	190000	130000	320000	2600	3800	2600	6400
29	10	16	4854.917717	4	4	240	0.15	4	8.33	0	1	220000	320000	320000	320000	4400	6400	6400	6400
48	16	10	4854.917717	4	4	240	0.15	4	8.33	0	1	320000	320000	220000	320000	6400	6400	4400	6400
53	17	19	4823.950831	2	2	120	0.15	4	11.67	0	1	320000	260000	320000	190000	6400	5200	6400	3800
58	19	17	4823.950831	2	2	120	0.15	4	11.67	0	1	320000	190000	320000	260000	6400	3800	6400	5200

Figure A.1: Sioux Falls network data. Adapted from (Transportation Networks for Research Core Team, 2020)

B. Sioux Falls travel demand data

1. The dataset (A.2) was an adaptation of the data of (LeBlanc et al., 1975). To restore the data to the original LeBlanc-data, the figures traffic flows were multiplied by ten.

2. Then, this flow data was converted to hourly flows by dividing the total flow by 24.

3. Then, this data was converted to minute flows by dividing the hourly flows by 60

4. To compute the time interval in seconds at which cars are spawned, 60 was divided by the minute flows.

5. To decrease the number of cars, the intervals were multiplied by 400.

6. Then, the figures were rounded, and the 50 highest (non-zero) OD combinations were selected (A.3).

The numbers in figure A.2 represent traffic units per minute. The numbers in figure A.3 represent second intervals between drivers spawning at a node.

Node	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	0	100	100	500	200	300	500	800	500	1300	500	200	500	300	500	500	400	100	300	300	100	400	300	100
2	100	0	100	200	100	400	200	400	200	600	200	100	300	100	100	400	200	0	100	100	0	100	0	0
3	100	100	0	200	100	300	100	200	100	300	300	200	100	100	100	200	100	0	0	0	0	100	100	0
4	500	200	200	0	500	400	400	700	700	1200	1500	600	600	500	500	800	500	100	200	300	200	400	500	200
5	200	100	100	500	0	200	200	500	800	1000	500	200	200	100	200	500	200	0	100	100	100	200	100	0
6	300	400	300	400	200	0	400	800	400	800	400	200	200	100	200	900	500	100	200	300	100	200	100	100
7	500	200	100	400	200	400	0	1000	600	1900	500	700	400	200	500	1400	1000	200	400	500	200	500	200	100
8	800	400	200	700	500	800	1000	0	800	1600	800	600	600	400	600	2200	1400	300	700	900	400	500	300	200
9	500	200	100	700	800	400	600	800	0	2800	1400	600	600	600	1000	1400	900	200	400	600	300	700	500	200
10	1300	600	300	1200	1000	800	1900	1600	2800	0	3900	2000	1900	2100	4000	4400	3900	700	1800	2500	1200	2600	1800	800
11	500	200	300	1400	500	400	500	800	1400	4000	0	1400	1000	1600	1400	1400	1000	200	400	600	400	1100	1300	600
12	200	100	200	600	200	200	700	600	600	2000	1400	0	1300	700	700	700	600	200	300	500	300	700	700	500
13	500	300	100	600	200	200	400	600	600	1900	1000	1300	0	600	700	600	500	100	300	600	600	1300	800	700
14	300	100	100	500	100	100	200	400	600	2100	1600	700	600	0	1300	700	700	100	300	500	400	1200	1100	400
15	500	100	100	500	200	200	500	600	900	4000	1400	700	700	1300	0	1200	1500	200	800	1100	800	2600	1000	400
16	500	400	200	800	500	900	1400	2200	1400	4400	1400	700	600	700	1200	0	2800	500	1300	1600	600	1200	500	300
17	400	200	100	500	200	500	1000	1400	900	3900	1000	600	500	700	1500	2800	0	600	1700	1700	600	1700	600	300
18	100	0	0	100	0	100	200	300	200	700	100	200	100	100	200	500	600	0	300	400	100	300	100	0
19	300	100	0	200	100	200	400	700	400	1800	400	300	300	300	800	1300	1700	300	0	1200	400	1200	300	100
20	300	100	0	300	100	300	500	900	600	2500	600	400	600	500	1100	1600	1700	400	1200	0	1200	2400	700	400
21	100	0	0	200	100	100	200	400	300	1200	400	300	600	400	800	600	600	100	400	1200	0	1800	700	500
22	400	100	100	400	200	200	500	500	700	2600	1100	700	1300	1200	2600	1200	1700	300	1200	2400	1800	0	2100	1100
23	300	0	100	500	100	100	200	300	500	1800	1300	700	800	1100	1000	500	600	100	300	700	700	2100	0	700
24	100	0	0	200	0	100	100	200	200	800	600	500	800	400	400	300	300	0	100	400	500	1100	700	0

Figure A.2: Original travel demand on Sioux Falls (Transportation Networks for Research Core Team, 2020)

Node	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	0	34400	34400	6800	17200	11600	6800	4400	6800	2800	6800	17200	6800	11600	6800	6800	8800	34400	11600	11600	34400	8800	11600	34400
2	34400	0	34400	17200	34400	8800	17200	8800	17200	5600	17200	34400	11600	34400	34400	8800	17200	0	34400	34400	0	34400	0	0
3	34400	34400	0	17200	34400	11600	34400	17200	34400	11600	11600	17200	34400	34400	34400	17200	34400	0	0	0	0	34400	34400	0
4	6800	17200	17200	0	6800	8800	8800	4800	4800	2800	2400	5600	5600	6800	6800	4400	6800	34400	17200	11600	17200	8800	6800	17200
5	17200	34400	34400	6800	0	17200	17200	6800	4400	3600	6800	17200	17200	34400	17200	6800	17200	0	34400	34400	34400	17200	34400	0
6	11600	8800	11600	8800	17200	0	8800	4400	8800	4400	8800	17200	17200	34400	17200	4000	6800	34400	17200	11600	34400	17200	34400	34400
7	6800	17200	34400	8800	17200	8800	0	3600	5600	2000	6800	4800	8800	17200	6800	2400	3600	17200	8800	6800	17200	6800	17200	34400
8	4400	8800	17200	4800	6800	4400	3600	0	4400	2000	4400	5600	5600	8800	5600	1600	2400	11600	4800	4000	8800	6800	11600	17200
9	6800	17200	34400	4800	4400	8800	5600	4400	0	1200	2400	5600	5600	5600	3600	2400	4000	17200	8800	5600	11600	4800	6800	17200
10	2800	5600	11600	2800	3600	4400	2000	2000	1200	0	800	1600	2000	1600	800	800	800	4800	2000	1200	2800	1200	2000	4400
11	6800	17200	11600	2400	6800	8800	6800	4400	2400	800	0	2400	3600	2000	2400	2400	3600	17200	8800	5600	8800	3200	2800	5600
12	17200	34400	17200	5600	17200	17200	4800	5600	5600	1600	2400	0	2800	4800	4800	4800	5600	17200	11600	6800	11600	4800	4800	6800
13	6800	11600	34400	5600	17200	17200	8800	5600	5600	2000	3600	2800	0	5600	4800	5600	6800	34400	11600	5600	5600	2800	4400	4800
14	11600	34400	34400	6800	34400	34400	17200	8800	5600	1600	2000	4800	5600	0	2800	4800	4800	34400	11600	6800	8800	2800	3200	8800
15	6800	34400	34400	6800	17200	17200	6800	5600	4000	800	2400	4800	4800	2800	0	2800	2400	17200	4400	3200	4400	1200	3600	8800
16	6800	8800	17200	4400	6800	4000	2400	1600	2400	800	2400	4800	5600	4800	2800	0	1200	6800	2800	2000	5600	2800	6800	11600
17	8800	17200	34400	6800	17200	6800	3600	2400	4000	800	3600	5600	6800	4800	2400	1200	0	5600	2000	2000	5600	2000	5600	11600
18	34400	0	0	34400	0	34400	17200	11600	17200	4800	34400	17200	34400	34400	17200	6800	5600	0	11600	8800	34400	11600	34400	0
19	11600	34400	0	17200	34400	17200	8800	4800	8800	2000	8800	11600	11600	11600	4400	2800	2000	11600	0	2800	8800	2800	11600	34400
20	11600	34400	0	11600	34400	11600	6800	4000	5600	1200	5600	8800	5600	6800	3200	2000	2000	8800	2800	0	2800	1600	4800	8800
21	34400	0	0	17200	34400	34400	17200	8800	11600	2800	8800	11600	5600	8800	4400	5600	5600	34400	8800	2800	0	2000	4800	6800
22	8800	34400	34400	8800	17200	17200	6800	6800	4800	1200	3200	4800	2800	2800	1200	2800	2000	11600	2800	1600	2000	0	1600	3200
23	11600	0	34400	6800	34400	34400	17200	11600	6800	2000	2800	4800	4400	3200	3600	6800	5600	34400	11600	4800	4800	1600	0	4800
24	34400	0	0	17200	0	34400	34400	17200	17200	4400	5600	6800	4400	8800	8800	11600	11600	0	34400	8800	6800	3200	4800	0

Figure A.3: Adapted travel demand on Sioux Falls. The green cells are among the 50 most frequent, non-zero cells