A Rental System of Electrical Cars in Amsterdam

Leon Rothkrantz

Delft University of Technology

Mekelweg 4

2628 CD Delft

The Netherlands Defence Academy

Nieuwe Diep 8

1781 AC Den Helder

The Netherlands

+31 15 2787504 (tel) +3115 2787141 (fax)

l.j.m.rothkrantz@tudelft.nl

Joost Boehle

Delft University of Technology

Mekelweg 4

2628 CD Delft

JLBoehle@tele2.nl

Michiel van Wezel

Econometric Institute

School of Economics

Erasmus University Rotterdam

P.O.Box 1738

3000 DR Rotterdam

The Netherlands

MCwezel@gmail.com

Abstract

This paper is about the introduction of an electric car rental system in the city of Amsterdam. These cars can be used as an alternative to public transport or taxis by travelers or tourists arriving at train stations or park houses who want to reach places in the inner city. After the trip, cars will be parked in one of the park houses distributed over the city close to the traveler's destination. Electric cars will be used to reduce the pollution in the inner part of the city. To facilitate the drivers a dedicated navigation system has been designed including lanes not accessible by private cars. This dynamic routing system is based on the Ant Based Routing algorithm. It has been implemented on the power line infrastructure of streetlights. Cars are wirelessly connected to lampposts using their smart phone. The cameras attached to the lampposts will be used to localize, identify and track the cars and enable updates of the routing tables in the navigation system. The network of distributed streetlights/cameras is the backbone of the Park and Routing-system. A software prototype of the system has been implemented. We will report about the simulated system and the test results.

Keywords

Electrical cars, car rental system, dynamic routing, Ant Based Control algorithm, video surveillance system.

1. Introduction

The city of Amsterdam was the first city with a bike sharing system. A number of bicycles are made available for shared use to travel from one point in the city to another. To prevent that the bikes were stolen or damaged users can release a bike from a locked terminal by a small cash deposit. By parking the bike at the arrival place the deposit can be retrieved. Unfortunately the weather conditions in the Netherlands don't support biking all the time. The idea is to introduce an electrical car rental system (P&R). The cars will be parked at Park houses distributed over the city. Travelers arriving by their own car or by train switch to electrical cars instead of public transport. The advantage of the electrical cars is that they have access to many areas of the inner city which are closed for private cars. And it will enable tourist to cross the city using the electrical cars. This plan fits in the grant schema of the city council's goal of having 200,000 battery-driven vehicles on Amsterdam's roads by 2040.

The P&R is a distributed system that comprises two complementary services: parking and routing. The parking service allows P&R participants to rent an electrical car. Next the driver plans his trip. The parking service then selects parking places based on the planned tour, destination and stops in between, the preferences of the driver and the set of other drivers currently requesting a parking place. For efficient use of the cars, the P&R-system will also provide a routing system to guide cars from the starting position to the stops/end position.

Because the electrical cars are allowed to follow special tracks not available for regular traffic, a dedicated routing system had to be developed, because in current commercial navigation systems these special tracks are blocked. There are three different types of vehicles. The first group is the private cars which have limited access to parts of the inner city. Next group consists of taxis, public transport and trams which have access to special lanes. The last group is the group of electrical cars having access to all the roads in the inner city.

The requested routing system has to be a dynamic system using up-to-date information about traffic jams and incidents. One possible approach is a centralized static system based on Dijkstra's algorithm, see "Dijkstra (1959)". But a centralized system is vulnerable, if the ICT system breaks down, or no GPS signals are received then the routing service is not available. It can be expected that during the rush hours there are a lot of traffic jams. In those situations the shortest route in distance is usually not the shortest route in time. In the inner city of Amsterdam traffic routing has to be very dynamic. There are many narrow streets, many incidents and limited parking places, many stopping cars, blocking the road for a short or longer time. Most of the delays can be solved locally instead of globally. For that reason we developed a decentralized system taking care of different traffic loads on the roads. The system is based on the Ant Based Control algorithm (ABC). Since the introduction of the ABC algorithm, see "Dorigo et al. (2006)", "Di Cario et al. (1998)", we are involved in Ants based research. We compared the ABC algorithm with other routing algorithms and it proves that ABC has a similar performance, see "Schoonderwoerd et. al. (1996)". Since 2000 we applied ABC to traffic problems, see "Tatomir et al., (2004), (2005), (2006)". Some of the research problems we had to solve in our application are:

- *When does the dynamic decentralized ABC routing algorithm outperform the static, centralized Dijkstra routing system?*
- *What is the minimal percentage of users of electrical cars compared to other car drivers needed to update the routing tables in a reliable way?*

The ABC routing system is a decentralized system which requires local data. It is common use to update such a system using a telephone network, see "Radu et al. (2012)", "Tatomir et al. (2004)". Car drivers send at regular time their current position using a communication service via their (smart) phones. This data enables the system to update the routing tables dynamically. The problem posed by this alternative is that the service of a telephone provider is needed. Such a service is not free of charge. The favored solution was to use existing networks under control of the City Council. In the first place the power line infrastructure of streetlights will be used. Secondly, the dense surveillance network of video cameras will be involved. The surveillance system of video cameras will be used in the first place for security reasons and to assist drivers of the electrical cars for example in case of mechanical problems.

One of the research challenges of this project was to use the surveillance network to route the cars. In our first developed prototype local data including time and position was provided by the cameras attached to the lampposts. An alternative is to use the wireless network around the lamppost which communicates messages from the passing cars including identity, position and timestamp. In the current system data from the video surveillance system has been used. Using dedicated software installed at the cameras, a camera system is able to localize, identify, and track the cars. The routing system constantly gathers time and delay data from vehicles. The combined information from all vehicles is used to determine time optimal routes from the vehicles current position to the destination parking place and to route electrical cars. The information exchange between vehicles and the routing system is facilitated by means of intelligent lampposts located at each intersection in a city and near conglomerations of parking places under system control. The intelligent lampposts form the backbone of the routing system, as they support the communication structure through which data is collected, transformed into information and distributed to the various components of the routing system. Intelligent lampposts are modified normal lampposts. Through their common connection on the cities power-line infrastructure they form a distributed interconnected network which is able to exchange information via power-line communication technology. Each intelligent lamppost also contains a wireless communication device that allows it to communicate with vehicles of users and a small processing unit to process routing information and maintain routing tables (usually a smart phone). By collecting and exchanging data about current traffic conditions on the road of the city and the usage of the ant-based routing algorithm the intelligent lampposts are able to discover time optimal routes to every destination within the city. This information is then shared with the users of the routing system via wireless communication.

Figure 1. Example of a control room with surveillance employee.

The city of Amsterdam is covered by a dense network of video cameras. These cameras are used for video surveillance to detect special incidents. These cameras are attached to monitor screens, which are monitored by security personnel in order to detect unusual behavior or incidents. To monitor all the electric vehicles 24 hours a day and 7 days a week requires too many resources. There is a need for automatic surveillance. Tracking many objects with many sensors has been researched by “Pasula et al. (1999)”. They describe the possibility of tracking highway traffic using multiple cameras. Unfortunately the software proved to be unstable and missed too many cars. For the tracking task we used a tool called Predator, see “Kalal et al. (2010)” also known as OpenTLD, to localize cars and track these cars simultaneously. From the recorded tracks we can compute data needed for the routing systems. To identify the cars, image processing software has been used to recognize the unique numbers on the doors of the cars. We used an adapted version of the Neocognitron Neural Network which was developed to recognize license plates, see “Cornet et al. (2003)”. This data is needed to update the probability tables in the ABC algorithm as explained in one of the next sections. From the computed tracks of the cars we can draw conclusions of the driving behavior and special incidents. If a car remains at the same position for a longer time, an alert can be sent to the surveillance operator to investigate if the car stop is caused by mechanical problems, a car crash, or whatever. If it can be expected that the road is blocked for a longer period of time the corresponding link in the routing network can be disabled. This results in a fast update of the system.

In our simulation experiments we used the specific locations of the surveillance cameras, locations which we further employed to update the routing tables of the routing system. We researched the following questions:

- *How far is it possible to use the existing network of security cameras to update the routing tables of the ABC algorithm?*

- *How far is it possible to use the existing system of static security cameras located in the city area to assist the security personal in observing electrical cars?*

The outline of the paper is as follows. In the next section we will introduce related work of routing systems. Then we will discuss the ABC routing system in more details. In section 3 we will present our simulation environment, while in Section 4 the results of our experiments. are provided. We conclude this paper in Section 5 and list our references in Section 6.

2. Related work

2.1 Dynamic versus static routing

Traffic assignment is defined as the problem of finding traffic flows given an origin-destination trip matrix and a set of costs associated to the links. One solution for this problem is either that the driver drives on the optimum path according to his preferences, known as the User Equilibrium (UE) assignment or alternatively the path that minimizes the overall network's traveling time, known as the System Optimum (SO) assignment.

“Wardrop (1952)” was the first to differentiate the two methods. A spectacular example that shows that the UE assignment is in general different from the SO solution is the Braess network. “Braes et al. (2005)” obtained the paradoxical result that the addition of an arc to the network can result in increased origin to destination and overall travel cost. “Fisk (1979)” studied the Braess paradox in more detail. She presented the sensitivity of travel costs to changes in the input flows while they are in Wardropian equilibrium. Examples showing that an increased capacity of the input flow can decrease the traveling time are presented.

Non-equilibrium methods assign traffic to a single minimum path between two zones. The minimum path infers the minimum travel time. Minimum path algorithms include for example the models developed by “Dantzig (1957)” and ”Dijkstra (1959)” Other non-equilibrium methods include diversion models, multipath assignments and eventually combined methods.

When a time dimension is added to the models previously described then DynamicTraffic Assignment (DTA) is obtained. By including temporal dimensions we can represent traffic situation and compute traveling times more realistically. Literature surveys in this field generally mention two main approaches for DTA: the analytical-based models and the simulations.

The analytical-based model considers two time indices: the time at which the path flow leaves its origin and the time at which it is observed on a link. In other words, this approach assumes that the whole time is divided in intervals. Static mathematical analytical control models are applied to each interval, on the assumption that one interval is long enough so that drivers can complete the trip within that certain time interval.

The simulation-based model simulates the behavior of the drivers in different traffic settings. Due to their capability of better representing the real world they have increased in popularity. Simulations usually try to replicate the complex dynamics of the traffic. Although this is considered a different approach, the mathematical abstraction of the problem is a typical analytical formulation.

Analytical-based approaches

Literature within this area of research is extensive. DTA has evolved a lot since the work of “Merchant et al. (1978)” who considered a discrete time model for dynamic traffic assignment with a single destination. The model they assumed was nonlinear and non-convex. “Ziliaskopoulos (2000)” split the analytical models in four broad methodological groups where the first ones are the mathematical programming formulations. Within this approach flow equations are deducted and a nonlinear mathematical programming problem has to be solved. “Merchant et al. (1978)” and “Ho (1980)” studied such models. Due to the complexity of a nonlinear problem, a linear version of the model with additional constraints can be created and solved for a global optimum using a simplex algorithm. The linear program has a staircase structure and can be solved by decomposition techniques.

In optimal control theory the routes are assumed to be known functions of time and the link flows are considered continuous functions of time. The constraints are similar to the ones of the mathematical programming formulation, but defined in a continuous-time setting. This results in a continuous control formulation and not in a discrete-time mathematical program. “Friesz et al. (1989)” discuss two continuous link-based time formulations of the DTA for both the SO and UE objectives considering the single destination case. The model assumes that the adjustments of the system from one state to another may occur while the network conditions are changing. The routing is done based on the current condition of the network but it is continuously modeled as conditions change. The SO model is a temporal extension of the static SO model and proves that for the optimal solution the costs for the O-D used paths are identical to the ones of the unused paths. They established as well a dynamic generalization of Beckmann's equivalent optimization problem.

Simulation-based approaches

Simulation environments address key issues of the traffic assignment, such as the flow's propagation in time and the spatio-temporal interactions. Contemporary DTA models were developed using different simulators, such as CONTRAM (CONtinous TRaffic Assignment Model), DYNASMART or SATURN). SATURN “Hall et al. (1980)” is an early DTA simulation tool that uses an equilibrium technique. The CONTRAM, see “Taylor (1980)” simulation environment is more dynamic than the previous ones as it allows the re-routing of cars if traffic conditions worsen. However, it does not consider a maximum storage capacity for roads and it assigns cars only based on the Wardropian principle. DYNASMART is a contemporary DTA model which uses the basic CONTRAM concept. “Abdelfatah et al. (2001)” showed an example of a DTA model developed by the DYNASMART approach. “Lum at al. (1998)” showed that the average speed depends on the road's geometry, on the traffic flow characteristics and on the traffic signal coordination. A new travel time-density model was formulated by incorporating the minimum-delay per intersection and the frequency of intersections as parameters.

The traveling time and the traffic volume are two main field items that have to be considered for the speed flow study along arterial roads. Most influencing factors that have been cited in literature are the special incidents and holidays, signal delays, weather conditions and the level of congestion. The prediction error might be also directly proportional with the length of the forecasting period, see ”Kisgy (2002)”. “Hobeika et al. (1994)” constructed three models for short-term traffic prediction by combining the current traffic, the average historical data and the upstream traffic. “Li et al. (2002)” use GPS equipped probe vehicles and determine mean speed values in order to develop a fuzzy mathematical travel time estimation model. Time series analysis is as well a popular method to infer the travel time prediction due to their strong potential for on-line implementation. “Ishak et al. (2002)” described a short-term prediction model for speed that follows a nonlinear time series approach and uses a single variable. In literature, researchers have used parametric models in order to forecast the travel time, such as regression models or time series and non-parametric models that include ANN models, see “Lint (2005)”, “Yu et al. (2008)”. Studies have shown that ANN’s (including modular neural network model and state space neural network model) are a powerful tool to predict travel time on freeways, see “Lint (2005)”. “Yu et al. (2008)” proposed a travel time prediction model which comprised two parts: a base travel time and a travel time variation.

2.2 Ant Based Control routing

The ant-based routing algorithms “Di Caro et al. (1998)”, “Dorigo et al. (2002, 2007)”, “Schoonderwoerd et al (1996)”, “Tatomir et al. (2004, 2005, 2006)” upon which our routing system is based mimics the food searching behavior that Argentinean ants exhibit. Ants travelling back from a food source to their colony deposit a natural pheromone – a chemical substance that is used to exchange messages between ants – that marks the trail they have taken. At each obstacle encountered on the route back to the colony the ants make a decision about which path to follow next. Since ants do not rely on visual information for path finding every choice the ant makes when diverting an obstacle is randomly chosen. Therefore approximately fifty percent of ants will choose to divert by going left around the obstacle and the other fifty percent by going right. If for example the right route is shorter than the left the ants following that route create a denser pheromone trail than the ants traveling via the left route. The increased density of the left trail will entice other ants heading towards the food source in following the right route. This increase in traffic via the right route will eventually lead to evaporation of the pheromone on the left route since fewer ants are enticed to follow it. This process allows ants to establish the most optimal route between food source and nest over a period of time depending on the number of obstacles on the route.

In traffic networks obstacles on the route between point A and B usually exhibit themselves in the form of congestions, road maintenance works or accidents. Avoidance of these obstacles is paramount when attempting to reach point B in the shortest time. However, when comparing the behavior of ants with the one of human drivers, we notice that where ants cooperate to determine the shortest route human drivers take an individual approach. This individual approach will in most cases lead to inferior obstacle avoidance solutions since the human driver only possesses local obstacle knowledge and is unaware of obstacles further along the route or developments within the network that will create obstacles in time.

The ant-based routing algorithm enables human drivers to cooperate in a manner equal to ants in order to form time optimal shortest paths throughout the city. Via the routing infrastructure of interconnected intelligent lampposts and wireless communication devices, vehicles are able to provide the system with a constant stream of up-to-date data concerning obstacles. This data known to the algorithm as traffic condition data enables intelligent agents called ants to approximate the shortest route through a city. This information is then relayed to vehicles upon request to ensure the most optimal route over the entire duration of the trip.

2.3 Road Network Hierarchy

Routing vehicles via time optimal path towards their destination involves the determination of vehicle traffic flows and densities on the roads within the network. Low vehicle traffic flows combined with high vehicle densities often indicate the presence of an obstacle and should therefore be avoided in favor of roads with a high vehicle traffic flow and low vehicle density. While this approach functions in theory, in practice certain negative side effects can occur that are unacceptable in modern cities. The most common example of such a negative side effect occurs when a main road leading into a city is congested leading the ant-based algorithm to favor roads that lead through the densely populated residential area next to the main road. While in terms of optimal time routing this path is valid, residents will soon start to complain (an undesirable effect from the city councils point of view). In order to prevent occurring the negative side effects,we have opted to impose a hierarchy on the road network within the city.

A second reason for imposing a hierarchy on the road network is the relation between nodes and ants within the ant-based algorithm. The ant-based algorithm requires that each node periodically sends an ant to all other nodes within the network. Thus, when the number of nodes increases the number of ants required for the proper functioning of the algorithm grows proportionally. This inevitably leads to an increase in ant processing time which can cause the formation of suboptimal routes within the network and overall degenerated performance of the ant-based algorithm, see “Kassabalidis et al. (2002)”.

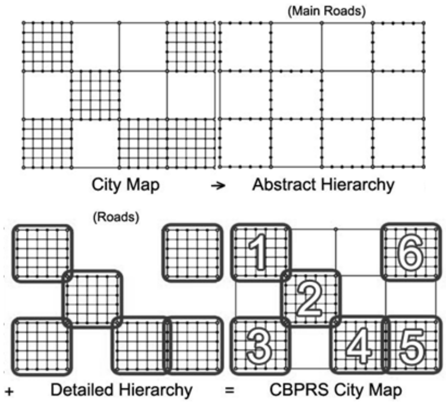

Figure 2. Transformation from city map to P&R network hierarchy.

Road networks within cities display a 'natural hierarchy' when dealing with them in terms of throughput capacity. The high capacity roads (main roads) serve to quickly transport vehicles from the outskirts of the city to the city centre, while roads with a lower capacity enable drivers to access the main roads within the city. The hierarchy we have designed consists of two hierarchical layers that exploit the 'natural hierarchy' within a city as shown in Figure 2. The abstract hierarchical layer consists of the main roads within a city and forms an interconnected network for transporting large amounts of vehicles between the sectors of the detailed hierarchical layer. The detailed hierarchical layer consists of the lower capacity roads within the city that are divided into sectors. We created these sectors by subtracting the abstract hierarchical layer from the normal city map creating independent groups of roads that form the basis of the sectors. To couple the abstract and detailed hierarchical layers we linked each sector to the nearest abstract layer intersections to enable the travelling between sectors. However, vehicles that travel on the abstract hierarchical level to a destination within a sector on the detailed hierarchical level require a way to indicate that destination since destinations on the detailed hierarchical level are not known on the abstract level. The solution to this problem was found by introducing virtual nodes. A virtual node combines all destinations within a sector on the detailed hierarchical level into a single destination. Vehicles travelling between sectors use the virtual node associated with their destination to travel between sectors via the abstract hierarchical level.

2.4 Routing tables

The ant-based algorithm presented in this paper employs two types of routing tables, identical in configuration, to serve both hierarchical levels described in the previous section. The global routing table serves to route vehicles between sectors by using the virtual node identifiers, while the local routing table serves to route vehicles within a sector. The routing table configuration, as shown in Table 1, contains an entry for each destination to which the node is allowed to route vehicles. This entry consists of the address of the destination, a probability value for each neighbor that represents the pheromone trail strength for the current destination via the neighbor and an average delay representing the average time in seconds required to travel from the current intelligent lamppost towards the destination.

Table 1 Basic layout of routing table of a node.

Destinations	Neighbor 1	……………..	Neighbor n	Average Delay
D_i	$P_{n1\rightarrow i}$		$P_{nn\rightarrow i}$	μ_i

Valid destinations for the local routing table of a given node in a certain sector are those nodes that reside in the same sector. The neighbors listed in the local routing table are those nodes that reside in the same sector and can be reached by following an outbound lane from the current node. A node has one local routing table for each sector it is part of. Valid destinations for the global routing table are the virtual nodes that represent the different sectors in the detailed hierarchical level. The determination of the valid neighbors however differs between nodes that are only present on the detailed hierarchical level and nodes that are present on the abstract hierarchical level or on both. Nodes only present on the detailed hierarchical level use the global routing to route vehicles towards the nearest of the abstract hierarchical level that is also a member of the current sector. The valid neighbors for this type of nodes are therefore identical to those within the local routing table. Nodes that are present on the abstract hierarchical level or on both determine the set of neighbors, as those nodes that reside on the abstract level can be reached by following an outbound lane from the current node.

The Ant Based Routing algorithm requires a constant stream of up-to-date travel time information concerning all roads within the network to generate time optimal routes. The video cameras attached to the intelligent lamppost track cars at crossings. These devices monitor the time required to travel between intelligent lampposts and communicate this perceived delay to the first intelligent lamppost it encounters.

$$M_r = M_r + \omega\ (D_r - M_r),\ \omega = 0.3\ \in [0,1] \qquad (1)$$

The intelligent lamppost adds this newly received information to an average delay for the road r the vehicle has just traveled on using Equation 1. M_r represents the weighted mean of the delays reported to the lamppost. The weight ω determines the influence that a newly reported delay D_r has on the measurement and how long such a reported delay influences this measurement. The number of weights that really influence the measurement can be approximated via $\approx 5(1/\omega)$. Thus, the value of $\omega = 0.3$ indicates that the latest ± 17 delays reported by vehicles influence this measurement. This relatively small amount of delays that influence the average delay allows the routing algorithm to react quickly to ever-changing traffic conditions. The intelligent lamppost stores the delay information in this manner for each road with incoming lanes on the intersection it is monitoring. The average delay information is then used by the route finding system, described in the next section, to determine the optimal time routes within the network.

2.5 Routing Table Probability Updating

The Ant Based Routing algorithm requires a constant stream of information to ensure up-to-date and time efficient routes within the network. The ants supply this constant stream of information through collection of delay data by the forward agents and the translation of that data into routing information by the backward agents. This section discusses how the backward agent transforms the information provided by the forward agent into routing information.

$$T_{id} = \sum T_{jk},\ where\ j,k \in I \rightarrow d \qquad (2)$$

Upon arrival at an intelligent lamppost, the backward ant calculates the average delay T_{id} between the current intelligent lamppost i and the destination intelligent lamppost d by summing up the individual average delays T_{jk} for the roads between the intelligent lampposts on the path between i and d – using equation 2. The average delay T_{id} between the current lamppost and the destination lamppost as determined by the ant is used to alter the average delay μ_d for the route from the current intelligent lamppost towards the destination as stored by the intelligent lamppost. The average delay μ_d is determined via equation 3. The variable η is used as a weight to limit the influence of each delay on the average delay.

$$\mu_d = \mu_d + \eta(T_{id} - \mu_d),\ \eta = 0.1 \in [0,1] \qquad (3)$$

As shown in equation 4, the virtual delay T_{id} for the current route is divided by the average delay μ_d multiplied with a scaling factor c to determine the strength of the reinforcement λ given to the probability P_{dn} where n represents the next intelligent lamppost on the path towards the destination intelligent lamppost. If the virtual delay for the route between intelligent lampposts i and d is less than the average delay, the probability eventually receives a positive increment, otherwise the probability is left unaltered.

$$\lambda = \begin{cases} \frac{T_{id}}{c \cdot \mu_d}, \frac{T_{id}}{c \cdot \mu_d} < 1, c = 1.1 \in [1,2], \\ 1, \frac{T_{id}}{c \cdot \mu_d} \geq 1 \end{cases} \qquad (4)$$

Once the probability reinforcement strength is known, the backward ant can adjust the probability values for each valid neighbor V_n from the set of neighbors N_n of the current intelligent lamppost. The set of valid neighbors is a subset of the set of neighbors for a certain intelligent lamppost; the members of this set are those neighboring intersections to which traffic rules at the intersection allow a vehicle to move. The probability value P_{dn} for the neighbor n on the shortest path towards the destination intelligent lamppost d receives a positive stimulus using equation 5 where α is used as a scaling factor to dampen oscillation caused by the probability updating process, see “Schoonderwoerd et al. (1996)”.

$$P_{dn} = \alpha\,(1 - \lambda)(1 - P_{dn}),\ \alpha = 0.1 \in [0,1] \qquad (5)$$

The probabilities P_{dr} of the other intelligent lamppost r out of the set of valid neighbors V_n are decreased using equation 6. This decrease in probability values for these intelligent lampposts reflects their current status as sub-optimal routing solutions for the P&R routing service.

$P_{dr} = -(1 - \lambda) P_{dr}$, where $dr \neq dn$, $dr \in V_n$, $V_n \subset N_n$ (6)

After alteration of the probabilities, the values are normalized and clipped between 0.05 and 1. The lower value of 0.05 is set to prevent ants from ignoring a possible route towards the destination via an alternative intelligent lamppost. Intelligent lampposts between which the route should be disabled can set their probability to zero for that specific neighbor; this prevents ants from inspecting that route. The backward ant repeats this process on every intermediate node between *s* and *d*. Once the backward ant arrives in *s* – the source of the forward-ant - the backward ant is destroyed and the path between the source and destination nodes is updated according to current information available to the algorithm.

3. Simulation Environment

We designed and implemented our own simulation environment while taking into account the lessons learned during the study of other simulation environments, see "Tatomir, (2004)", "Rothkrantz, (2001)". The simulation environment was implemented in the C# programming language and based on design patterns described by "Gamma, (1995)". To ensure flexibility and extendibility required for future research we divided the functionality of the simulation into four distinct groups of classes names. The simulation group provides functionality and objects for conducting and regulating the flow and speed of the simulation as well as basic data logging services. The infrastructure group contains the basic classes that define a road network such as intersections, roads, lanes and a default implementation of a vehicle. The routing group provides classes for experimenting with different routing algorithms other than the Ant-Based Control algorithm and Dijkstra's routing algorithm that are provided by default. We were able to develop a full implementation of the simulation environment, including extended data logging services that monitor vehicles. The GUI contains classes and functionality for providing the graphical display of the road network and vehicles. The GUI group is the most coupled of all groups since it combines functionality of all other groups by providing a uniform overview to the user.

The simulation of vehicle movements in a city environment is a complex and potentially computational heavy task, certainly when modeling in a microscopic simulation environment with a large set of parameters that influence the behavior of vehicles and drivers. Certain appraisals have to be made between the natural behavior of vehicles and the running time of the simulation and graphical representation thereof. The cellular automata principle, see "Taub, (1961)" enabled us to create 'realistic' vehicle movement and behavior without sacrificing simulation execution speed.

When applying the concept of cellular automata to road networks the driving lanes of the roads and intersections are divided into blocks of equal length. Each block can be occupied by one vehicle at a time and is 7.5 meters long – the space required by a vehicle standing still in a traffic jam – although it is possible to shorten the block length in order to simulate lower speeds. At every time step of one simulated second, all vehicles are moved to their new positions based on the rule-set in use. In our simulation environment, we have opted for the vehicle movement rule-set defined by "Nagel et al (1992)", extended with ideas presented in

“Wagner, (1996)” in order to produce ‘realistic’ accident free driving and lane changing behaviors in vehicles. Parameters within the rule-set enable us to influence the traffic flow, the tendency to change lanes, and the chance that a vehicle will not properly assess the current traffic situation by braking in an excessive manner. We extended the model with two additional parameters that influence the lane-changing tendency before intersections to ensure that vehicles arrive at an intersection in a valid driving lane for their current route. The incorporation of visual features such as turn lights and breaking lights provided new means for simulating the reactions of specific types of behavioral driving models within the simulation environment.

The infrastructure within the simulation environment consists of a relatively small number of classes that when linked together within the simulation environment enable the user to create complex road networks. Via the simulation environment users are able to manually position intersections and roads as well as determine the number of lanes on each road. Extended configuration parameters on each object further allows the user to increase the realism of the simulation (although this process can become quite time consuming). Table 2 contains an overview of the infrastructural class taken from all simulation environment groups and their capabilities. The precedence and traffic light intersections in the simulation environment have a great impact on the patterns of vehicle movement during a simulation. Intersections that allow only a single vehicle to pass at each time step unnecessarily delay the progress of other vehicles causing traffic jams and delays to appear on roads while they would not appear in reality, see “Schadschneider, (2000)”, “Giridhar, (2006). Therefore, certain precautions have to be taken in order to insure that multiple vehicles can cross the intersection at the same time without leading to collisions and disobedience of the precedence rules applicable to the intersection.

Each intersection consists of a number of inbound lanes and outbound lanes. A vehicle traveling on an inbound lane has the possibility to cross the intersection onto an outbound lane with the exception of the outbound lane that belongs to the same road as the inbound lane the vehicle is currently driving.

Table 2 : Simulation environment class descriptions.

Class	Capabilities	Description
Lane	- Vehicle lane changing. - Complex vehicle movement through cellular automata structure. - Turning restrictions for vehicles at intersections.	One-directional part of a road divided into blocks of 7.5 meters that allow a vehicle to move according to the global traffic rules. Turning restrictions {right, left, ahead, etc} at intersection can be used to stimulate complex presorting behavior of vehicles.
Road	- Supports multiple lanes.	Connection between two intersections that supports multiple interconnected lanes in

	- Maximum vehicle speed. -Supports placement of parking places / garages.	either direction. Facilitates lane changing for vehicles and supports the placement of parking places and intelligent lampposts.
Intersection	- Enables collision free crossing of multiple vehicles simultaneously. - Supports traffic lights. - Supports precedence rules. -Contains an intelligent lamppost, with cameras.	Intersections are the basis of a messaging system. The vehicle on the intersection requests a crossing, the intersection then reports when the vehicle can. Intersections support the crossing of multiple vehicles from different lanes at the same time. Contains a intelligent lamppost that can be queried by vehicles for routing directions and the reservation of a parking place.
Generator	- Creates different types of vehicles based on user defined periodic intervals. - Enables vehicles to leave the simulation environment.	The generator enables natural flow of vehicles within city by creating entry/exit points that generate different types of vehicles based on user preferences. Generators connect to the city infrastructure via roads and can be used to represent highway connections to other cities, dense residential areas that serve traffic focal points during peak traffic hours.
Vehicle	- Different driver behaviors. -Collision free traffic movement. -Uniform congestion behavior based on global rule-set. - Pre-sorting at intersection incorrect drive lanes. - Maximum vehicle speed. - Vehicle length. -Communication with intelligent lampposts. -Different routing algorithms. - Parking and routing behavior. -Different types of cars, electrical cars, taxis, public	The vehicle enables the realistic creation and resolving of congestions in combination with different vehicle behaviors as found in human drivers. The vehicle can be specialized by adjusting parameters and overwriting different behavioral functions. Specialization of this vehicle present in the simulation environment are the ant vehicle, ant parking vehicle, Dijkstra Vehicle, Dijkstra parking vehicle, bus and truck. Vehicles prefixed with ant are users of the P&R where Dijkstra vehicles route themselves using Dijkstra's algorithm. The bus is used to provide additional information to the P&R concerning a fixed route through the city.

	transport, regular cars.	
Parking place	- Parking for a single vehicle - Hourly tariff specification - Tracks and reports parking place status through incorporated parking place sensor. - Surveillance cameras	Parking place located at the side of a road and accessible via the outermost lane of a road. Parking places can be queried for their status {free, occupied, reserved} and track their own occupancy rate and revenue.

Therefore, depending on the number of inbound and outbound lanes and the allowed turn directions a directed graph can be generated in which the edges represent a path from an incoming lane to an outgoing lane. The edges in this directed graph represent all valid manners in which vehicles can cross an intersection. Using the directed graph of the intersection the conflicting edges are determined. Conflicting edges are those edges that would cause collisions between two vehicles when being travelled on at the same time. In order to eliminate the problem of conflicting edges a conflict matrix is constructed on the basis of the directed graph of the intersection. The conflict matrix consists of a series of entries that specify if two edges can be active at the same time without leading to vehicular collisions.

Each vehicle that approaches an intersection notifies the intersection of its arrival and the edge it intends to follow when crossing the intersection. The intersection then verifies if the edge that the vehicle wants to follow is available – meaning that there are currently no conflicting routes in use – and applies precedence and traffic light rules. The intersection should then formulate a response to the vehicle stating whether it is allowed to proceed into the intersection. In the next time step, the vehicle determines if it is allowed to continue and enter the intersection or to break to a halt before entering the intersection. This type of messaging system ensures collision free traffic movement over intersections and reduces the computational overhead.

4. Experiments

Participating vehicles in our simulation experiments can be characterized along two dimensions. The first is the routing algorithm employed, which can either be Dijkstra's algorithm or the Ant Based Control algorithm. (We do not simulate vehicles that do not employ routing.) Since Dijkstra's algorithm is non-adaptive we expect it to perform inferior when many traffic jams occur. The second dimension concerns parking behavior: vehicles can either park or not, and if they do, electrical cars can use the P&R parking service or they can search for a place randomly. This leads to six possible vehicle types.

Our assumption is that drivers of electrical cars start and end in park houses. There are limited places to charge the electrical cars. Parking along the streets or at parking lots has its limitations and requires a random search for free places unless the facilities of P&R are used. Most car drivers want to park as close as possible to their destination. Because they have no idea where the empty places are they circle around the place of destination hoping to find an empty place. This takes a lot of time and generates additional pollution. The P&R system has access to a database of free parking places in park houses or parking lots. One or more free places are selected by the system according to the requests of the driver and the driver is routed to that empty place. We experimented with various relative frequencies for these vehicle types, as shown in Table 3.

Table 3: Vehicular distributions per experiment.

	Vehicles	Dijkstra Vehicle	Dijkstra parking vehicle	Ant Vehicle	Ant Parking vehicle
Experiment one	10000	90%	0%	0%	0%
Experiment two	10000	90%	0%	0%	10%
Experiment three	10000	0%	0%	90%	10%
Experiment four	10000	60%	0%	30%	10%
Experiment five	10000	40%	0%	50%	10%
Experiment six	10000	90%	10%	0%	0%

Experiment one served as our benchmark situation where all vehicles were routed via Dijkstra's algorithm and no electrical cars are used. Experiment two measured the effectiveness of the P&R when 10 percent of all vehicles are electrical cars, the minimal participation limit set in this study, which used the P&R services. Experiment three shows the maximum possible level of effectiveness of the P&R when all vehicles are electrical cars. Other experiments illustrate upper and lower limits of effectiveness found in this simulation environment.

The experimental results including traveling time for the six experiments conducted using the environment of the city of Amsterdam are displayed in Figures 3and 4 and Tables 4 and 5. Comparing experiment one with experiment six, we found that when Dijkstra parking vehicles in experiment six are allowed to reserve a parking place via the P&R parking service the average travel time for parking the normal vehicles decreases. Again, we found that an increase in traffic condition information leads to a decrease in average travel times. However, we found that the relation between an increase in traffic condition information and a decrease in average travel times started to deteriorate when more than forty percent of the vehicles supplied traffic condition information. Comparing the results in Table 3 from experiment four and five, we found that the increase in traffic condition information supply from forty to sixty percent has a small effect on the average travel times. However, the relation between an increase in traffic condition information and the reliability of the routing solutions offered by the P&R keeps improving significantly.

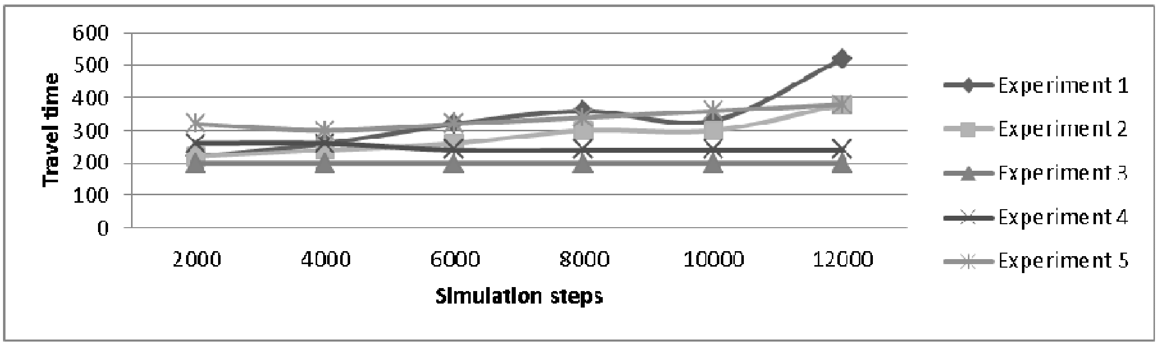

Figure 3. Average travel time for parking vehicles during the experiments 1-6.

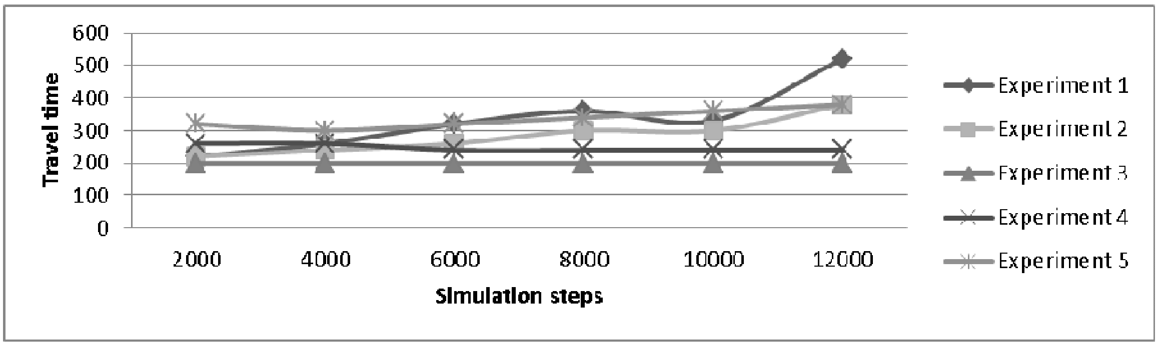

Figure 4 Average travel time for non-parking vehicles during experiments 1-6.

Table 4: Experimental results of experiments one, two and three.

	Experiment one		**Experiment two**		**Experiment three**	
	Parking	*Normal*	*Parking*	*Normal*	*Parking*	*Normal*
Samples	3565	16452	3264	17671	2949	16954
Average travel time	573	648	236	261	215	219
Std. Deviation	885	1088	121	202	92	112
Avg. Parking	5096	-	5097	-	5036	

Table 5: Experimental results of experiments four, five and six.

	Experiment four		**Experiment five**		**Experiment six**	
	Parking	*Normal*	*Parking*	*Normal*	*Parking*	*Normal*
Samples	3319	17758	3276	17443	3030	17098
Average traveling time	223	251	222.98	229	371	469
Std. Deviation	107	268	87	115	665	745
Avg. parking	5132	-	5082	-	5048	

We conclude that the superior ability of the ant based algorithm incorporated in the P&R to distribute vehicles over a large city environment using the main road network of such an environment to its fullest extend leads to significant decreases of overall travel times for all vehicles present in the environment. We found that when using Ant Based Routing parking vehicles the average travel times decrease dramatically for both parking and non-parking vehicles. This means that random search for a free parking place increases the traveling time significantly. This trend of reduction of traveling time continues during each experiment constantly having the biggest impact on the average travel times for non-parking vehicles. Reviewing the benefits associated with the P&R we found that during each experiment the travel time for participants has decreased and that the effects of the P&R on non-parking participants are even greater due to a better overall distribution of vehicles. We found that overall traffic flows improve once the number of ant based vehicles increases and traffic jams are reduced. Therefore, it is concluded that using the P&R within a metropolis environment has significant benefits for all vehicles within the environment and justifies any investment made in a system such as the P&R. Taking into account the results obtained for the previous simulation using the city environment we found that the trend visible in that environment and the conclusions drawn are in line with results presented in this section. This indicates that results found during both these experiments were caused by the influence of the P&R on traffic rather than environmental circumstances that could affect the outcomes.

5. Conclusions

In this paper, we have presented a car rental system for electrical cars for the city of Amsterdam. We introduced a city based parking and routing system (P&R) that guides participants through a city environment using a distributed hierarchical algorithm based on the Ant colony meta-heuristic. Through a series of computer-simulated experiments, we have been able to show that finding a parking place at or near the driver’s point of destination in combination with a dynamically determined optimal route towards the parking place can lead to significant benefits for participants and non-participants in the P&R.

As to be expected, the Dijkstra guided vehicles suffer from the static solution provided to them when the number of vehicles in an environment increases. The level of vehicular distribution over the environment is low and traffic is concentrated in a small portion of the main roads within the environment. The lack of equal vehicle distribution inevitability leads to the formation of traffic jams that have a negative effect on individual traffic times and traffic flows. The P&R needs traffic condition information to ensure the proper calculation of optimal routes. Through the conducted experiments we found that an increase in participating vehicles has a positive effect on individual travel times, traffic flows and traffic jams. While an exact relation between an increase in dynamic routing information provided by the participants and the reduction of overall travel times could not be determined, all experiments conducted showed either improvement for parking vehicles, non-parking vehicles or both. The results of the experiments clearly show that the number of participants influences the performance and effectiveness of the Ant Based Routing algorithm incorporated into the P&R.

The effects of the parking service provided by the P&R helped to reduce travel time for ant-based parking vehicles. The benefit of a parking place at the destination point helped to reduce disturbances in traffic flows. This trend was also visible when Dijkstra oriented parking vehicles were allowed access to the P&R parking service. While the parking service as currently implemented in the P&R only provides simple scheduling of parking places to participants, benefits were clearly visible once the number of occupied parking places increased. Further enhancement of the performance of the P&R as a whole could be gained from scheduling individual users and their preferences in a global optimal manner, but this has not been researched during this project. Dijkstra oriented parking vehicles found themselves significantly disadvantaged once the number of occupied parking places increased. On average, the hinder to Dijkstra parking vehicles from an inferior routing solution was more significant than the time lost searching for a parking place.

In our simulation environment we implemented a realistic environment of the city of Amsterdam. Some roads are blocked for private cars and only accessible by public transport or electrical cars. The existing surveillance system of video cameras was also implemented in our simulation environment. It proves that the system of video cameras covered all the possible tracks of electrical cars. Based on the track from one camera to the next one we computed the traveling time and used that data to update the probability tables in our routing system. To identify cars, we used a special neural network, called Neocognitron. We tested the system on the recognition of car templates. The results of recognition are published elsewhere, see “Cornet, B., et al (2003)”.

We therefore conclude that the P&R as tested in the simulation environment is able to decrease travel times for participants, increase the overall flow of traffic and decrease the number of traffic jams and the amount of time spent in such a traffic jam by vehicles once the simulation environment reaches the size of a city. Furthermore, we found that the P&R is able to match the performance in terms of travel times for statically routed vehicles in small environments while the P&R always significantly outperforms the static routing solution in larger environments.

References

Abdelfatah, A. and Mahmassani, H. (2001). “A simulation-based signal optimization algorithm within a dynamic traffic assignment framework”, in Intelligent Transportation Systems, Proceedings. 2001 IEEE, 2001, Oakland , CA. pp.428-433.

Braess, D., Nagurney, A. and Wakolbinger, T. (2005), "On a paradox of traffic planning," Journal Transportation Science, volume 39, 2005, 446–450.

Cario, G.D. and Dorigo, M. (1998). “AntNet: Distributed Stigmergetic Control for Communication Networks”, IEEE, Journal of Artificial Intelligence Research, 9, 317-365.

Cornet, B. and Rothkrantz, L.J.M. (2003). “Recognition of car license plates using a Neocognitron neural network”. *Neural Network World*., 2, 115-132.

Dantzig, G.B. (1957). “Discrete-variable extremum problems”, Operations Res., 5, 266-277.

Dijkstra, E.W. (1959). “A note on two problems in connexion with graphs”, Numer. Math., 1, 269-271.

Dorigo, M. and Stützle, T. (2002).“The Ant Colony Optimization Metaheuristic: Algorithms, Applications, and Advances”, Handbook of Metaheuristics. Vol 57, Kluwer Academic Publishers, Norwell, MA., 251-285.

Dorigo, M. and Socha, K (2007). “An introduction to ant colony optimization,” in Handbook of Approximation Algorithms and Metaheuristics, T. F. Gonzalez, Ed., pp. 26.1–26.14, Chapman & Hall/CRC, Boca Raton, Fla, USA, 2007.

Dorigo, M. and Stützle, T. (2004). Ant Colony Optimization, The MIT Press, Cambridge, Mass, USA, 2004.

Fisk, C. (1979). “More paradoxes in the equilibrium assignment problem”, Transportation Research Part B: Methodological, 13(4), 1979, 305-309.

Friesz, T.L., Luque, J., Tobin, R.L. and Wie, B.W. (1989). “Dynamic network traffic assignment considered as a continuous time optimal control problem”, Operations Research, 37(6), 1989, 893-901.

Gamma, E., Helm, R., Johnson, R. and Vlissides, J. (1995). “Design Patterns , Elements of Reuseable Object-Oriented Software”. Reading, MA: Addison- Wesley.

Giridhar, A. and Kumar, P.R. (2006). “Scheduling Automated Traffic on a Network of Road”, IEEE Transactions on Vehicular Technology, 2006, 55, 1467-1474.

Hall, M.D., Van Vliet, D. and Willumsen, L.G., Saturn, (1980). “A simulation assignment model for the evaluation of traffic management schemes”, Traffic Engineering and Control, 21(4), 1980, 168-176.

Ho, J.K. (1980). “A successive linear optimization approach to the dynamic traffic assignment problem”, Transportation Science, 14(4), 1980, 295-305.

Hobeika, A. and Kim, C.K. (1994). “Traffic flow-prediction systems based on upstream traffic”, European Journal of Operational Research, 253-261.

Ishak S. and Al-Deek, H. (2002). “Performance evaluation of short-term time-series traffic prediction model”, Journal of Transportation Engineering, 128(6), 2002, 490-498.

Kalal, Z., Matas, J. and Mikolajczyk, K.. (2010). “”P-N Learning: Bootstrapping Binary Classifiers by Structural Constraints” . IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, USA, June 2010.

Kassabalidis, I., El-Sharkawi, M.A., Marks, R.J. II, P. Arabshahi, and Gray, A.A. (2002). “Adaptive-SDR: adaptive Swarm-based Distributed Routing”, IEEE Proceedings of the 2002 International Joint Conference on Neural Networks, Honolulu, Hawaii May 2002.

Kerner, B.S. (2004). “The Physics of Traffic”. Springer, 2004.

Kisgy, K. and Rilett, L.R. (2002). “Travel time prediction by advanced neural network”, Periodica Polytechnica Ser. Civ. Eng, 46, 2002.

Li, Y and McDonald, M. (2002). “Link travel time estimation using single GPS-equipped probe vehicle”, IEEE 5th International Conference on Intelligent Transportation Systems, 2002, 932-937.

Lint van, J.W.C., Hoogendoorn, S.P and van Zuylen, H. (2005). “Accurate freeway travel time prediction with state-space neural networks under missing data”, Transportation Research Part C: Emerging Technologies, 13(5-6), 2005, 347-369.

Lum, K.M., Fan, H.S.L., Lam, S.H. and Olszewski, P. (1998). “Speed-flow modeling of arterial roads in Singapore”, Journal of Transportation Engineering, 124(3), 1998, 213-222.

Merchant, D.K. and Nemhauser, G.L. (1978). “A model and an algorithm for the dynamic traffic assignment problems”, Transportation Science, 12(3),1978, 183-199.

Nagel, K. and Schrekenberg, M. (1992). “A Cellular Automaton model for Freeway Traffic”, Journal Physics France, 1992, 2221-2229.

Pasula, H., Russell, S., Ostland, M. and Ritov, Y. (1999). “Tracking many objects with many sensors.” Proceedings of the International Joint Conferences on Artificial Intelligence. Stockholm, 1999. 26.

Radu, A.A., Rothkrantz, L.J.M. and Novak, M. (2012). “Digital Traveler Assistant. Informatics in Control, Automation and Robotics” Springer Verlag, 174:101-114.

Rothkrantz, L.J.M. and Ehlert, P.A.M. (2001). “Microscopic traffic simulation with reactive driving agents”, IEEE Intelligent Transportation Systems Conference Proceedings, 2001, 860-865.

Schadschneider, A. (2000). “Statistical Physics of Traffic Flow”, Physica A, 2000, vol.285(1-2), 101-120.

Schoonderwoerd, R., Holland, O.E., Bruten, J.L. and Rothkrantz, L.J.M. (1996). “Ant based Load Balancing in Telecommunication networks”, Adaptive Behavior, 1996, 169-207.

Tatomir, B., Dibowsky,H. and Rothkrantz, L.J.M.(2004). “Hierarchical routing in traffic networks”, 16th Belgian-Dutch Conference on Artificial Intelligence, Groningen, 2004.

Tatomir, B., Kroon, R. and Rothkrantz, L.J.M.(2004). “Dynamic Routing in Traffic Networks Using AntNet”, Ant Colony Optimization and Swarm Intelligence ANTS 2004, Lecture Notes in Computer Science 3172, 2004, 424-425.

Tatomir, B., Boehlé, J.L. and Rothkrantz, L.J.M. (2005). “Dynamic routing in traffic networks and MANNETs using Ant Based Algorithms”, 7th International Conference on Artificial Evolution, 2005.

Tatomir, B. and Rothkrantz, L.J.M., (2006). “Hierarchical routing in traffic using swarm-intelligence”, The 9th International IEEE Conference on Intelligent Transportation Systems, I, 2006, 230-235.

Taub, A.H. (ed). (1961-63). John von Neumann: Collected Works, 1903-1957, 6 Vols. Pergamon Press, Oxford (UK)

Taylor, N.B. (1980). “Contram dynamic traffic assignment model”, Networks and Spatial Economics, 1980, 297-322.

Wagner, P., Nagel, K. and Wolf, D.E. (1996).“Realistic Multi Lane Traffic Rules for Cellular Automata”, Physica A, 1996, 687-698.

Wardrop, J. (1952). “Some theoretical aspects of road traffic research”, Proceedings of the Institution of Civil Engineers, Part II, Vol. 1. no. 36, 1952, 352-362.

Yu, J., Chang, G.L., Ho, H. and Liu, Y. (2008). “Variation based online travel time prediction using clustered neural networks”, Transportation Research Record: Journal of the Transportation Research Board, 2008.

Ziliaskopoulos, A.K. (2000). “A linear programming model for the single destination system optimum dynamic traffic assignment problem”, Transportation Science, 34(1), 2000, 37-49.