Intention Aware Routing System with variable station pricing

Roderick de Britto Heemskerk[1] , **Valentin Robu**[1]
[1]TU Delft

Abstract

Intention Aware Routing System is a route-planning algorithm for electric vehicles that minimizes overall travel time by taking into consideration congestion at charging stations. This paper extends this algorithm to allow choices to be made based on prices at charging stations. The goal of this paper is to find a way to minimize maximum congestion while maximizing overall profit across the stations.

1 Introduction

There seems to be a common consensus in the scientific world and the vehicle industry: The future of vehicles is going to be electric. This is great, as Electric vehicles (EV) play a significant role in tackling climate change, which is one of the biggest problems of our time. EVs do not emit any greenhouse gases, as opposed to petrol vehicles which do.
While the time has not yet arrived at which driving electric is the standard, companies like Tesla, Volkswagen and Renault are pushing electric vehicles to their limits, and are steadily advancing this market. But while advances are being made in charging EVs, they are not universal. Many vehicles can not make use of superchargers. And even superchargers, which are supposed to be the fastest way to charge EVs, still cannot compare to petrol vehicles. The quickest an EV can get the charge equivalent to 600km is about 30 minutes, compared to a petrol vehicle where this takes about 5 minutes. If EVs want to charge en route, any queue at the charging stations increases waiting times linearly with the size of the queue. Considering the case where each vehicle charges for 30 minutes, on average this will increase waiting times with 15 minutes per vehicle[3]. As such, the waiting times at charging stations have to be taken into account while scheduling. This is not required for petrol vehicles, as the time to refill a tank is significantly quicker.
De Weerdt et al. [3] has already attempted to tackle this issue. Their paper suggested an Intention Aware Routing System (IARS). The basic idea is that individual vehicles share their intentions with a central system. The system then updates the traffic information. And as a result the vehicles can then formulate a better route. One thing that is missing in this model is the pricing at the charging stations as different stations might have different prices, and this might affect if people may want to charge or not.Various authors suggest different ways to deal with pricing of electricity for electric vehicles, like [2] [4], and [5]. [2] discusses a way to balance EVs over charging stations by setting certain prices. [4] suggests having a bidding system for charging spots. And finally, [5] describes the game-theoretic nature of buying and selling electricity for EVs. In this project, aspects from [2] and [5] will be used to extend the model developed by de Weerdt et al.
Adding pricing to the model then begs the question:

How can prices be set across the stations to minimize maximum congestion while maximizing overall profit across the stations?

To aid in answering this question, this paper will answer these subquestions:

1. How can pricing be included in the model of de Weerdt et al.[3]?
2. How can decision policies of a distribution of EVs across the system be modelled?
3. What strategies can stations use to affect the flow of vehicles?
4. What is the optimal strategy to reduce maximum congestion?

This Paper is structured as follows. Section 2 introduces the model used to solve the problem. Subsection 2.1 focuses on explaining the model given in [3], and subsection 2.2 focuses on extending the model to allow for station prices. Next Section 3 deals with the setup of the experiments performed in this paper. Section 4 demonstrates the found results, based on the experiments of Section 4. In Section 5 the ethical aspects of this paper will be discussed. Section 6 will contain a conclusion and discuss possible extensions of the model described in this paper.

2 Model

To effectively give an answer to the questions asked in this paper, a good model is needed. As such, this section discusses the model which is used for the simulations. Subsection 2.1 explains the model introduced in [3], which our model uses as a base. Next, Subsection 2.2 introduces the extension of this model which allows for stations to include prices, and the vehicles to react to these prices.

2.1 Base Intention Aware Routing System

The model used for this problem was introduced in [3]. This model works on a domain given by $\langle V, E, T, P, S, C\rangle$. Where $e = (v_i, v_j) \in E$ are edges, with $v_i \in V$ vertices. Both roads and charging stations are represented by these edges, but charging stations are represented by loops, so edges where $v_i = v_j$. We use the notation $E_{station} \subset E$ and $E_{road} \subset E$ for roads and stations respectively. For each edge there is a probabilistic distribution P, which models possible waiting times. This P is time-dependent for a finite set of time points represented by $T = \{1, 2, ..., t_{max}\}$. Since Vehicles have a finite amount of charge, the model includes S which represents the state of charge of a vehicle, which has a finite domain $\{0, 1, 2, ..., s_{max}\}$. Where 0 and s_{max} represent respectively an empty battery, and a fully charged battery. Finally each edge depletes a certain amount of charge based on cost function $C(e)$, which gives the cost of charge for each edge in the graph. For charging stations however, $C(e)$ is negative to represent charging the vehicle. Charging stations always fully charge EVs in this model. P represents a probability mass function, which for edge $e = (v_a, v_b)$ gives that $P(\Delta t = t_b - t_a|e, t_a)$ gives the probability that if you arrive at v_a at time t_a the chance that you take Δt time units on that edge.
To plan a route, this model creates a policy as opposed to a simple route. This policy is a function $\pi : V \times T \times S \to V$, which for state, consisting of a vertex v_c, current time at the vertex t_c, and state of charge at that vertex s_c gives the next vertex w. Using this, the next edge for a current state (v_c, t_c, s_c), and a policy π is given by $e = (v_c, \pi(v_c, t_c, s_c) = w)$. The final policy is calculated by finding a policy which maximizes the expected utility function given by

$$\begin{aligned}&\mathrm{EU}(e_c = (v_c, w), t_c, s_c|\pi)\\ &= \begin{cases} -\infty, & \text{if } s_c < 0\\ \sum_{\Delta t\in T} P(\Delta t|e_c, t_c)\cdot U(t_c+\Delta t, s') & \text{if} w = v_{dest}\\ \sum_{\Delta t\in T} P(\Delta t|e_c, t_c) & \\ \quad\cdot \mathrm{EU}((w, \pi(w, t_c+\Delta t, s')), t_c+\Delta t, s'|\pi) & \text{otherwise}\end{cases}\end{aligned}$$

Where s' is the new state of charge after taking an edge.

2.2 Pricing Extension

To influence the traffic in the model based on pricing, first the model needs to be equipped to deal with pricing. To this end, this paper extends the IARS model to include pricing. This leads to a model which works on a domain defined by $\langle V, E, T, P, S, C, M\rangle$. This introduces money to the model, where the amount of money spent is represented by a value in a finite set $M = \{0, 1, ..., m_{max}\}$, where m_{max} is the maximum price charged in the system. This then also affects the states of the individual EVs, which changes from $(v_c, t_c, s_c) \in (V \times T \times S)$ to $(v_c, t_c, s_c, m_c) \in (V \times T \times S \times M)$. The cost of charging at a charging station is defined by

$$\Pi(e) = \begin{cases} 0, & \forall e \in E_{roads}\\ p_e & \forall e \in E_{stations}\end{cases}$$

where p_e is a fixed station price. It is possible to make p_e time dependent, but in this paper we consider the impact on the period with highest possible congestion, i.e. rush hour. As such, making the price time dependent is not necessary.
The original model worked with a utility function dependent on charge and arrival time, where the utility function U is given by

$$U(t_c, s_c) = \begin{cases} -\infty, & \text{if } s_c < 0\\ -t_c, & \text{otherwise.}\end{cases}$$

To extend IARS to handle pricing multiple different methods can be used. One possible example is as given in [4], where U is given by:

$$U(t_c, s_c, m_c) = \begin{cases} -\infty, & \text{if } s_c < 0\\ -t_c - \gamma\cdot m_c, & \text{otherwise.}\end{cases} \tag{1}$$

with $\gamma > 0$ representing a time/ money trade-off. Here $\gamma = 10$ could represent that 10 minutes of detour is worth 1 euro of discount. In concept, the final utility function is similar, but it uses normalizing factors both in terms of decision parameters, and based on the values in the domain.
The final utility function decided upon was:

$$U(t_c, s_c, m_c) = \begin{cases} -\infty, & \text{if } s_c < 0\\ \gamma\left(\frac{T_{max}-t_c}{T_{max}-T_{min}}\right) & \\ +(1-\gamma)\left(\frac{M_{max}-m_c}{M_{max}-M_{min}}\right) & \text{otherwise}\end{cases}$$

In this formula M_{max} represents the highest price available at a charging station in the system. M_{min} was set to 0 as it was decided that the highest utility should be received from not charging at all. T_{max} is the maximum time the vehicle is willing to arrive at the destination, and T_{min} is the minimum possible time to reach the destination excluding charging time. In the end, T_{max} was decided to be $3 * T_{min}$ as this seemed to be a reasonable upper bound on the max willingness to make a detour. $\gamma \in [0, 1]$ is a normalized decision parameter, used to represent the time/ money trade-off. As such, $\gamma = 1$ represents a pure focus on arriving early, and $\gamma = 0$ represents only caring about getting the cheapest price. The choice to add normalizing factors based on $T_{min}, T_{max}, M_{max}$, and M_{min} was made because decisions regarding prices are usually based on relative discount.

3 Experimental Setup

To find meaningful results, this paper discusses a number of different graph layouts and corresponding distributions of Vehicles. This subsection presents scenarios in increasing order of complexity. These graphs will be used in the corresponding subsections of Section 4. The first three graphs will be bottleneck graphs, which are defined by having only one starting node, and one destination node, with a row of stations in the middle only connected to the starting node and destination node.

3.1 General Two Stations

For the first scenario we will be considering a graph with one starting node, and one destination node. Between these two, there are two stations, Station 1 and Station 2.

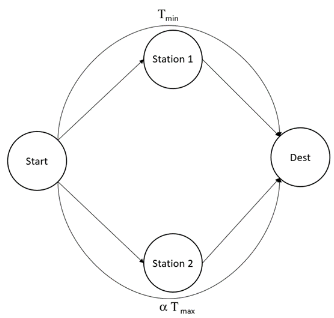

Figure 1: Graph used for the first scenario. Edge lengths represent travel time.

The route via Station 1 will be shorter than the route via Station 2, and will be denoted by $T_1 = T_{min}$. The time it takes to travel to the destination node via Station 2 will be considered to be a proportion of $T_{max} = 3T_{min}$. This time will be denoted by $T_2 = \alpha T_{max}$, where $\alpha \in [\frac{1}{3}, 1]$. This domain is to ensure that $T_{min} \le T_2 \le T_{max}$. We consider one class of vehicle, with $\gamma \in [0, 1]$. Since $T_1 \le T_2$, the price at Station 2 needs to be lower than the price at Station 1. As such we set the price at Station 1 $p_1 = M_{max}$. The price at Station 2 will be a proportion of M_{max} and will be given by $p_2 = \beta M_{max}$, where $\beta \in [0, 1]$. This setup allows for the formulation of a direct formula to calculate what β should be, this formula will be worked out in 4.1.

3.2 Two Stations

The second scenario that will be dealt with is an arbitrary case of the first one. This allows for verifying whether the model works as expected.

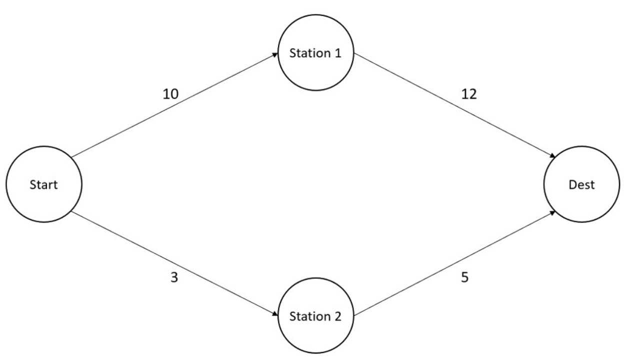

Figure 2: Graph used for the first scenario. Edge lengths represent travel time.

This layout uses one starting node and one destination node. As such all vehicles travel from Start to Dest. To go from the starting node to the destination node, only two possible paths are possible. The path via Station 1 takes 22 minutes in total, while the path via Station 2 takes 8 minutes. Station 1 and Station 2 each have their own prices, and these can be changed in the parameter file. We analyze this scenario in 2 ways. In the first case there is just 1 vehicle type, for which first $\gamma = 0.6$ will be considered, and then $\gamma = 0.4$. In the second case, we will consider 2 types of vehicles, V_1 and V_2, where both use IARS, but V_1 has $\gamma = 0.6$ and V_2 has $\gamma = 0.4$. This means that V_1 focuses more on getting to the destination quickly compared to C_2 which has a higher focus on getting a good deal. In this scenario, both V_1 and V_2 will have 5 vehicles in the system. What is interesting about this scenario is that if all vehicles in the system were using base IARS (equivalent to $\gamma = 1.0$ in the current system), all vehicles would choose station 2, as even with the extra congestion it would be faster than using Station 1. To affect this, Station 1 will be given a lower price than Station 2.

3.3 Three Stations

The third scenario is similar to the first two, but has an extra route to the destination.

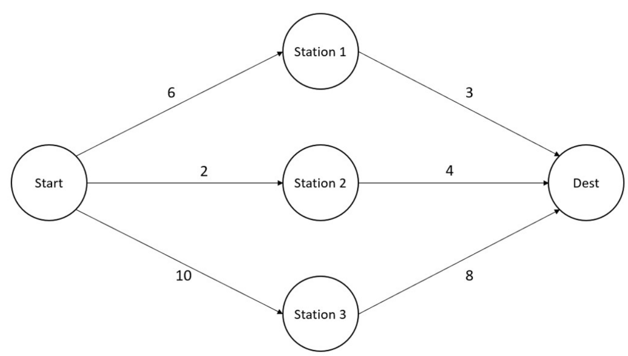

Figure 3: Graph used for the second scenario. Edge lengths represent travel time.

Once again this layout uses one starting node, and one destination node. To go from the starting node to the destination node, three paths are possible. The path via Station 1 takes 9 minutes, the path via Station 2 takes 6 minutes, and the path via Station 3 takes 18 minutes. Like scenario 2, we analyze 2 types of vehicles, separately the case where there is only one class of car, and then consider 2 classes of cars.

3.4 Two Station Grid

The final scenario, while having two stations like the first two, is differentiated by having multiple start nodes and destination nodes.

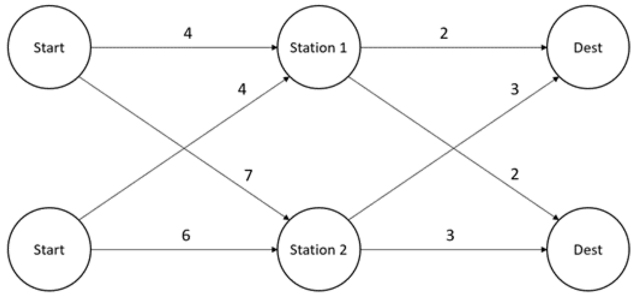

Figure 4: Graph used for the final scenario. Edge lengths represent travel time.

This layout uses two starting nodes, and two destination nodes. Vehicles are randomly assigned a starting node and destination node. This means that there are 4 unique (starting node, destination node) pairs. Each starting node is connected to both Station 1 and Station 2, and so are the destination nodes. As such, for every (starting node, destination node) node pair, a vehicle can either choose to take the route via Station 1 or the route via Station 2. The fact that each unique pair of starting/destination nodes has different routes available, means that it also has its own unique T_{min}.

4 Results

In this section, various results from the scenarios introduced in Section 3 will be shown. In 4.1, a formula will be derived to directly calculate the required prices to create an even split. The following subsections will use this formula to show various results.

4.1 General two stations bottleneck

For there to be an even split across the two stations, it is required for the utility function of the route via Station 1 and Station 2 to be equal, considering an equal queue. Using that we get that the Utility function for the path past Station 1 is

$$\begin{aligned}U_1 &= \gamma\frac{T_{max}-T_1}{T_{max}-T_{min}} + (1-\gamma)\frac{M_{max}-p_1}{M_{max}-M_{min}}\\ &= \gamma\frac{T_{max}-T_{min}}{T_{max}-T_{min}} + (1-\gamma)\frac{M_{max}-M_{max}}{M_{max}-M_{min}}\\ &= \gamma\end{aligned}$$

Then the Utility function for the path past Station 2 is

$$\begin{aligned}U_1 &= \gamma\frac{T_{max}-T_2}{T_{max}-T_{min}} + (1-\gamma)\frac{M_{max}-p_2}{M_{max}-M_{min}}\\ &= \gamma\frac{T_{max}-\alpha T_{max}}{T_{max}-\frac{1}{3}T_{min}} + (1-\gamma)\frac{M_{max}-\beta M_{max}}{M_{max}}\\ &= \gamma\frac{T_{max}-\alpha T_{max}}{\frac{2}{3}T_{max}} + (1-\gamma)(1-\beta)\\ &= \gamma\frac{3(1-\alpha)}{2} + (1-\gamma)(1-\beta)\end{aligned}$$

Transforming the domain of α from $[\frac{1}{3}, 1]$ to $[0, 1]$. Using

$$\alpha = (\frac{2}{3}\alpha_{new} + \frac{1}{3})$$

gives

$$\begin{aligned}U_1 &= \gamma\frac{3-3((\frac{2}{3}\alpha_{new}+\frac{1}{3}))}{2} + (1-\gamma)(1-\beta)\\ &= \gamma\frac{2-2\alpha_{new}}{2} + (1-\gamma)(1-\beta)\\ &= \gamma(1-\alpha_{new}) + (1-\gamma)(1-\beta)\end{aligned}$$

This α_{new} is defined in such a way, that $\alpha = 0$ corresponds to T_{min} and $\alpha = 1$ to T_{max}. So, if the time the route takes is T_1, we get

$$\alpha_{new} = \frac{T_1 - T_{min}}{T_{max} - T_{min}}$$

α_{new} will be just referred to as α from now on. Equating U_1 and U_2 gives the following equation:

$$\gamma(1-\alpha) + (1-\gamma)(1-\beta) = \gamma \tag{2}$$

Solving for β gives

$$\beta = 1 - \frac{\alpha\gamma}{1-\gamma} \tag{3}$$

Plotting this gives the following results:

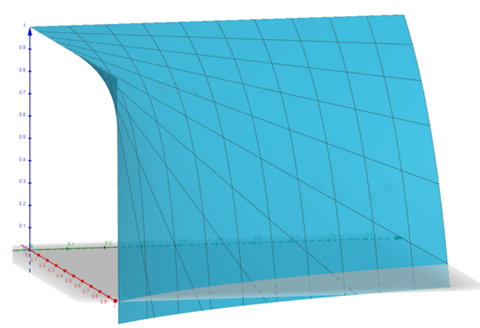

Figure 5: Graph showing the equilibrium β(blue axis) for each γ (red axis) and α (green axis)

Looking at Figure 5, there are some interesting results. When the two routes have the same length, so $\alpha = 0$, the function gives $\beta = 1$, meaning that the price should be the same at both stations. This makes sense, since the distances are the same, the algorithm would evenly split vehicles across the station based on the time aspect. Making one of the two stations cheaper would increase the utility function for that route, skewing the distribution towards that station. For $\gamma = 0$, once again we have $\beta = 1$. This is due to the fact that for $\gamma = 0$, the algorithm only looks at the price to determine the route. If there was even a small difference in price, all vehicles would go to one station. Next to that, this formula can be used to calculate what price would create an even split, it also gives another interesting result. To some extent, it is a disappointing result, but it was to be expected. What it shows is that, for some values of α and γ, there is no possible β, since $\beta \in [0, 1]$, for which the vehicles would split evenly across the stations. From the perspective of drivers, it does make sense, as there is a maximal detour which drivers would find acceptable, though this can depend on the individual driver.

4.2 Two stations bottleneck

This scenario is an arbitrary instance of the first scenario, it is possible to directly use the formula to calculate the price p_1 at Station 1. First, it is necessary to calculate the α from the values in the graph. This gives $T_{min} = 8$ and $T_{max} = 24$, and the route via Station 1 takes a total of 22 minutes. This gives $\alpha = \frac{22-8}{24-8} = \frac{14}{16} = 0.875$. Filling this in in the formula for $\gamma = 0.6$ gives

$$\beta = 1 - \frac{0.875 * 0.6}{1 - 0.6} = -0.3125$$

The formula gives $\beta < 0$, which means it is impossible to get an even split for this graph, if all vehicles have $\gamma = 0.6$. Doing the same however for $\gamma = 0.4$ gives

$$\beta = 1 - \frac{0.875 * 0.4}{1 - 0.4} \approx 0.4167$$

With $M_{max} = 10$, this gives $p_1 = 4$ as the model uses integer values for prices. Figure 6 shows a bar graph of how the vehicles, with $\gamma = 0.6$ distribute themselves for different prices at Station 1.

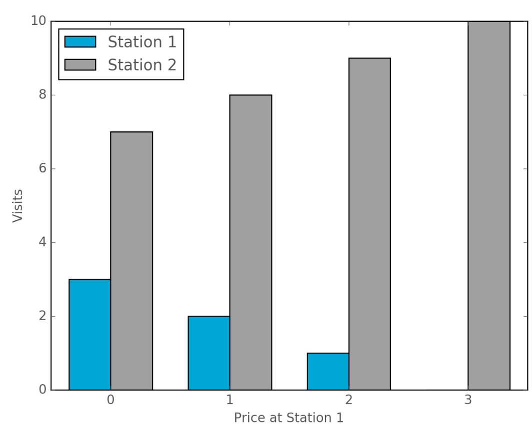

Figure 6: Bar Graph showing different amounts of visits across the stations for various p_1 with $\gamma = 0.6$

The distribution for $p_1 = 0$ is also shown, even though the price is realistic. This was done to show that even with the lowest price possible, vehicles still prefer Station 2 because of the shorter travel time. The distributions for $p_1 \geq 4$ is not shown as all vehicles are already choosing Station 2. Figure 7 shows a bar graph of how the vehicles distribute themselves for different prices at Station 1, this time the cars have $\gamma = 0.4$.

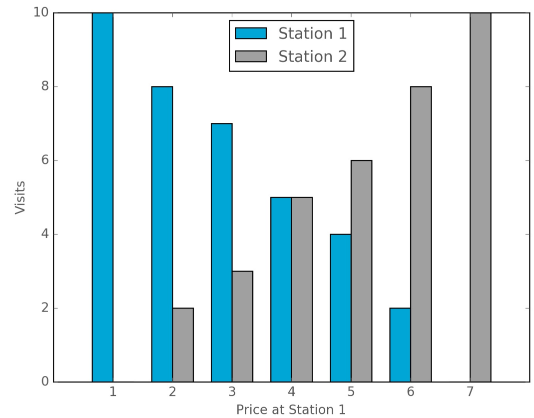

Figure 7: Bar Graph showing different amounts of visits across the stations for various p_1 with $\gamma = 0.4$

As can be seen, if $p_1 = 1$, all vehicles choose for Station 1. As p_1 increases, more vehicles start choosing for Station 2, until for $p_1 \geq 7$ all vehicles choose Station 2. The even split occurs when $p_1 = 4$, just as calculated.
Extending this scenario to two classes of Vehicles adds another dimension to this problem. If we generalize the problem saying we have two classes of vehicles V_1 and V_2 with γ_1 and γ_2 respectively, where $\gamma_1 \geq \gamma_2$. Assuming there are the same amount of vehicles of both in the system, it is possible to do the following. As we are trying to reach an even split, the easiest way to solve the problem would be to have all vehicles of one class to go to one station, and all of the other class to the other station. For this, define $U_{i,j}$ as the utility function for class i, for the route via Station j.
To get all vehicles of one class to one station, and all of the other class to the other station, we want $U_{1,1} \leq U_{1,2}$ and $U_{2,1} \geq U_{2,2}$. This was done because by assumption V_1 has a preference for the shorter route, so it is easiest to let them keep choosing that route. These inequalities are basically equivalent to Equation 2, save for the equals sign being replaced by the lesser or equal symbol and greater or equal symbol respectively. As such:

$$U_{1,1} = \gamma_1(1-\alpha) + (1-\gamma_1)(1-\beta) \leq \gamma_1 = U_{1,2}$$

which gives

$$\beta \leq 1 - \frac{\alpha\gamma_1}{1-\gamma_1}$$

and

$$U_{2,1} = \gamma_2(1-\alpha) + (1-\gamma_2)(1-\beta) \geq \gamma_2 = U_{2,2}$$

which gives

$$1 - \frac{\alpha\gamma_2}{1-\gamma_2} \leq \beta$$

Which when combined gives:

$$1 - \frac{\alpha\gamma_2}{1-\gamma_2} \leq \beta \leq 1 - \frac{\alpha\gamma_1}{1-\gamma_1}$$

Since $\gamma_1 \geq \gamma_2$,

$$1 - \frac{\alpha\gamma_2}{1-\gamma_2} \leq 1 - \frac{\alpha\gamma_1}{1-\gamma_1}$$

As such there will be values of β to make it valid. The issue comes from the fact that $\beta \in [0,1]$, which might not always be between $1 - \frac{\alpha\gamma_2}{1-\gamma_2}$ and $1 - \frac{\alpha\gamma_1}{1-\gamma_1}$. These values were calculated in the first case, so we get $-0.3125 \leq \beta \leq 0.4167$. Any value of β between those two values should give an even split. Seeing as profit should be maximized simultaneously, $\beta = 0.4167$ is chosen. This leads to $p_1 = 4$ once again. In Figure 8, once again distributions for different prices at Station 1 can be seen.

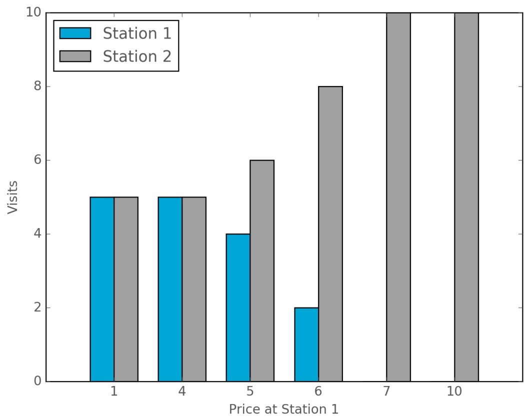

Figure 8: Bar Graph showing different amounts of visits across the stations

The prices 2 and 3 have been left out as they are the same as 1 and 4. This seems to confirm that the inequality gives valid values for an equal split.
It is also possible to directly solve the problem removing the assumption that both classes of vehicles have the same amount in the system. To do this we solve the problem for the class with the most vehicles in it, for which the problem is still solvable. $\gamma = 0.6$ for example is not solvable in this scenario, as it gives a $\beta < 0$. By solving for the largest possible group, the vehicles of that class will head to the path which has fewer vehicles in the queue, because that path will have the higher utility. This leads to them first equalizing both queues, and after that equally dividing over the stations.

4.3 Three stations bottleneck

The third scenario is a bit more complex, and serves as a step towards an N stations bottleneck problem. Assuming there is only one class of vehicle, it is possible to directly solve the problem, assuming the edge lengths allow for this. This can be done by making the utility function of all routes equal once again. This is equivalent to solving equation 3, for both non-shortest routes. To solve this example, set α_i to be the α of the route via Station i. Then $\alpha_1 = \frac{9-6}{18-6} = 0.25$. Assuming $\gamma = 0.4$, we get $\beta_1 = 1 - \frac{0.25 \cdot 0.4}{0.6} = 0.8333$. Doing the same for $\alpha_3 = \frac{18-6}{18-6} = 1$ gives $\beta_3 = 1 - \frac{1 \cdot 0.4}{0.6} = 0.3333$. Which since $M_{max} = 10$ means, $p_1 = 8$ and $p_3 = 3$. Filling this into the model gave an exact even split. Of course, if for any of the α_i, equation 3 gives a negative value, the problem is infeasible. When dealing with multiple classes of vehicles, the problem becomes a bit more complex. Assuming there are two classes of vehicles, no general way was found to solve the problem. If one of the classes for which the problem is solvable is large enough, it is possible to solve the problem for that class, leading to an even split. But if both classes are of equal size no direct solution was found. Figure 9 is a bar chart which shows how the vehicles distributed themselves across the stations, with increasing amounts of vehicles with $\gamma = 0.4$. For this scenario there are 6 vehicles with $\gamma = 0.6$.

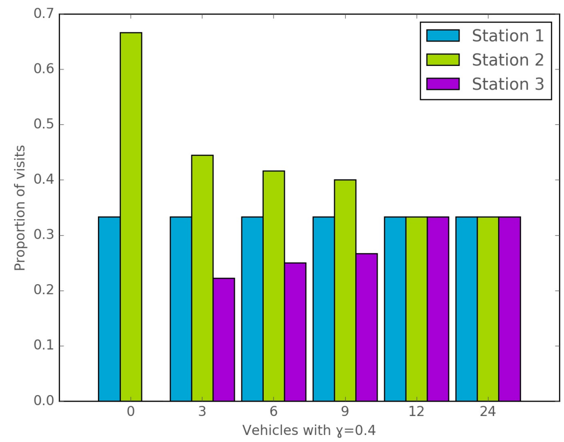

Figure 9: Bar graph showing different proportions of visits across the stations for various amounts of vehicles with $\gamma = 0.4$

The station prices in this case have been solved for $\gamma = 0.4$ so $p_1 = 8$ and $p_3 = 3$. What this graph shows, is that the vehicles are disproportionately splitting across the stations to in the end make it an even split. In this case, since we keep adding 3 cars to the system, one car always splits to Station 1. This ensures that Station 1 keeps the same proportion of visits, since it already has $\frac{1}{3}$ of the visits with the first distribution.
This form of solution will work similarly for any such N-station bottleneck scenario

4.4 Two Stations Grid

This final scenario, while seeming very similar to the two stations problem, immediately becomes more complex. Firstly, the utility functions for vehicles with different (start, destination) pairs is different, even for vehicles within the same class. This is due to the fact that T_{min} differs for unique (start, destination) pairs. And secondly, there are 2 stations that have to even out 8 total potential routes. The following figure shows the distribution of cars for different prices at Station 1. p_2 in this case was set to $M_{max} = 10$, and $\gamma = 0.4$ for all vehicles.

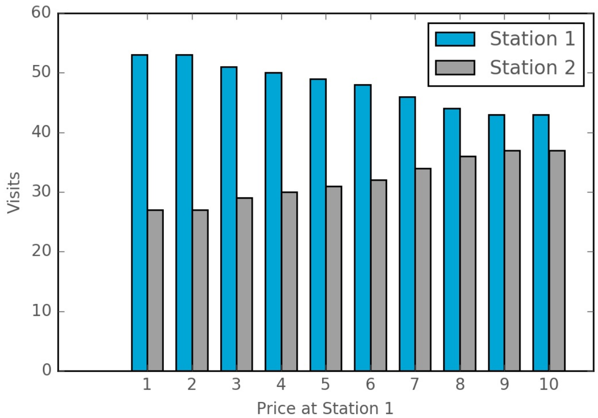

Figure 10: Bar Graph showing different amounts of visits across the stations for various p_1 with $\gamma = 0.4$

In this case p_1 was changed, because it was the only station price which had influence on the distribution. Setting $p_1 = M_{max} = 10$ and varying p_2 always led to a 43:37 distribution. With most edge lengths there seemed to be no major changes when influencing the price of the less popular station. Overall, it seems to be difficult to find an even split in this scenario. There are two likely reasons for this behavior. The first, is that unlike the previous scenarios, the vehicles might have a different travel time to a station, based on where they start. This makes it so vehicles which have a longer route to a station, will be guaranteed to have to queue behind cars that have a shorter route. In the previous scenarios, there was a chance that they would be further ahead in the queue. As such vehicles are less likely to take a station that another starting point can reach quicker. The second is, this problem can be seen as 4 separate two station problems, one for each (start, destination) pair. To solve the original 2 station problem, there was one degree of freedom, being the β. This made it possible to equate the utility functions for both routes. However, in this case we would require 4 degrees of freedom, where we only have 2. This makes solving it directly, nearly impossible.

4.5 Discussion

Having seen these examples, a few conclusions can be made. For bottleneck scenarios equation 3 (see Section 4.1) can be used to calculate even splits. However more complicated graph structures like the grid structure, can't be solved in this manner. The problem also becomes significantly more restrictive, and many might not be solvable at all. This idea is strengthened by the fact that even the simplest scenarios, like the bottleneck scenario, are impossible to solve.
Even though no direct method was found to solve the grid scenario, the solution could technically be found through brute force methods. However, these are too time consuming, considering that for each combination of prices the entire simulation needs to run again. According to [3] the IARS algorithm is bounded in time complexity by $\mathcal{O}(|T|^2 \cdot |V| \cdot |S| \cdot |E|)$. Adding the money aspect increased the complexity to $\mathcal{O}(|T|^2 \cdot |V| \cdot |S| \cdot |E| \cdot |M|)$. All possible combinations of prices comes down to $|M|^{|Stations|}$ since for each station, all values in M have to be considered. This ends up making the final complexity of a brute force algorithm $\mathcal{O}(|T|^2 \cdot |V| \cdot |S| \cdot |E| \cdot |M|^{|Stations|+1})$. Considering that the amount of stations will usually be bounded, this complexity is technically not truly exponential. However, it was considered to still be excessive.

5 Responsible Research

According to [1] many researchers have tried and failed to reproduce experiments of other researchers. This is problematic since it then becomes questionable if the right conclusions have been drawn. To circumvent this, we have taken care to ensure that it is possible to replicate the results found in this paper. This was done by explicitly stating which graphs were used. The formula used most often in the paper, also was carefully derived step by step. This was done to ensure that the formula could be replicated. Next to that, the code is open source, and can be accessed from GitLab. This is done not only to allow other scientists to try to reproduce the results, but also to improve the model.

6 Conclusions and Future Work

In this paper the IARS model introduced in [3] was extended to adjust to station prices. This was done to find a way to calculate prices to minimize maximum queue size across the stations, while maximising profit.
To this end, formula 3 was derived, and subsequently applied to different graphs. This led to the conclusion that the easiest way to split vehicles across a bottleneck graph, is to solve the problem only considering the largest class of vehicles. For more complicated graphs the problem quickly becomes too restricted, due to too few degrees of freedom to influence the traffic. Finally the use of a brute force algorithm was discarded, as the complexity was too high.
For future work, more research could be done regarding the grid scenario. Even though no direct way to solve it was found, maybe changes in the model might make it possible to solve. Using equation 1 as the utility function, might be more interesting and lead to more varied solutions. Next to that, the current model makes assumptions which might need to be relaxed to make it more realistic. The two main ones are that both charge time and money spent at a charging station are not charge dependent. In reality, you have to pay for the amount of charge, and charging an empty tank takes longer than charging a tank that is half full.

References

[1] Monya Baker. 1, 500 scientists lift the lid on reproducibility. *Nature*, 533(7604):452–454, May 2016.

[2] Daehyun Ban, George Michailidis, and Michael Devetsikiotis. Demand response control for phev charging stations by dynamic price adjustments. In *2012 IEEE PES Innovative Smart Grid Technologies (ISGT)*, pages 1–8, 2012.

[3] Mathijs M. de Weerdt, Sebastian Stein, Enrico H. Gerding, Valentin Robu, and Nicholas R. Jennings. Intention-aware routing of electric vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 17(5):1472–1482, May 2016.

[4] E. Gerding, S. Stein, V. Robu, D. Zhao, and N. R. Jennings. Two-sided online markets for electric vehicle charging. *AAMAS Conference on Autonomous Agents and Multi-Agent Systems*, pages 989–996, May 2013.

[5] Francesco Malandrino, Claudio Casetti, and Carla-Fabiana Chiasserini. The role of its in charging opportunities for evs. In *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pages 1953–1958, 2013.