Vehicle to grid capacity in The Netherlands

An agent-based modelling study regarding VTG and charging behaviour

B. Boereboom 4556232
F. Boendermaker 4655605
B. Hermans 4690435

Report for SEN9120 Advanced Agent Based Modelling

Cover image by Andreas (2019)

TU Delft

Summary

The pressure on a more sustainable use and generation of energy is rising. With it come many challenges, one of which is the balance or imbalance that might occur on the electricity grid. The use of Vehicle to Grid (VTG), in which plugged in electric vehicles (EVs) are used to balance the strain on the electricity grid, is being raised as a potential solution. However, dynamic information on VTG capacity in the Netherlands is lacking. This report documents the construction and the use of a calibrated and validated agent-based model of the entire EVs population of the Netherlands to approximate VTG capacity and power demand over time. Additionally, we added a spatial component to the capacity aspect, in order to also estimate the where aspect of VTG capacity over time. Results indicated sequacious capacity behaviour over time, with significantly high peaks during nighttime. This indicates the simultaneously charging behaviour of smart charging EVs during low energy price intervals. Furthermore, we found an interesting spatial distribution of capacity, with an indication of relatively high capacity in some regions. Our research could potentially be extended with more complex charging behaviour, including feedback mechanisms between energy usage and price.

B. Boereboom 4556232
F. Boendermaker 4655605
B. Hermans 4690435
Delft, January 2022

Contents

1 Introduction & conceptualisation **1**
1.1 The role of EVs in the energy transition . . . 1
1.2 Chosen modelling approach . . . 2
1.3 The energy transition model . . . 2
1.4 Conceptualization of the vehicle to grid system . . . 2
1.4.1 The main agent: an EV. . . . 2
1.4.2 Configuration of the environment . . . 3
1.4.3 Variables and processes . . . 4

2 Model construction **7**
2.1 timestep . . . 7
2.2 Encoding . . . 8
2.2.1 locations and driving . . . 8
2.2.2 charging . . . 8
2.3 KPI implementation . . . 9
2.3.1 power demand . . . 9
2.3.2 VTG capacity . . . 9
2.3.3 Postponing charge . . . 10
2.3.4 Feeding back electricity . . . 10
2.3.5 Example cases . . . 11
2.4 Verification . . . 14
2.4.1 Initial verification of output . . . 14
2.4.2 Structured verification . . . 15
2.4.3 Seeds . . . 15
2.5 runtime optimization . . . 16

3 Validation & experimentation **17**
3.1 Calibration of the model . . . 17
3.2 SOBOL analysis . . . 18
3.3 Map visualization of model run. . . . 18
3.4 Validation against etm model . . . 19

4 Model application **20**
4.1 Generating profiles . . . 20
4.2 Integration profiles into etm . . . 21
4.3 VTG capacity of a profile . . . 22

5 Conclusion & Discussion **23**
5.1 VTG capacity over time . . . 23
5.2 Model limitations . . . 24

A Appendix **26**
A.1 Example cases reference graphs . . . 26
A.2 assumptions . . . 27
A.3 Profiles exploration . . . 28

1

Introduction & conceptualisation

1.1. The role of EVs in the energy transition

The ever present effects of climate change press policymakers to change to a more sustainable use and generation of energy. Electrification has a high priority in the sustainability agenda. This is in part because the use of electricity in itself does not produce greenhouse gasses, and it is possible to generate electricity carbon-neutrally on a large scale using solar and wind. Aside from increasing the supply of electricity, policymakers strive towards replacing fossil fuel-based energy demand with electric devices. Replacing fossil fuel-based cars with electric vehicles (EVs) is an example of such a replacement.

Electrification is not without its downsides, however. Unlike gas or oil systems, electricity cannot be stored within the power grid. For the proper functioning of the grid, supply, and demand of electricity always needs to be in balance. This is problematic, as solar panels, as well as windmills, produce energy whenever the sun or wind is there. Their production can therefore not be fully controlled or predicted by humans, as was the case with power stations based on fossil fuel. In addition, demand for electricity can also not be fully predicted. Especially with increasing loads on the electricity grid due to electrification, it becomes even more important that the electricity grid functions properly. There is a need for balancing methods.

The increased load on the electricity grid due to EVs can be seen as an addition to the problem of grid balancing, due to the high loads needed for charging. A subset of EVs are able to charge smart, meaning that instead of charging whenever plugged in they try to reach their desired charge by charging only when the electricity is cheapest, if possible. If many EVs are charging during similar (cheap) times, this could cause large demand spikes in the network. The size and frequency of these spikes remains uncertain. However, when viewed from a different angle, EVs present a great opportunity for grid balancing. The accumulated capacity of EV-batteries attached to the system creates a significant disaggregated source of potential grid balancing storage capacity.

Vehicle to Grid (VTG) is a relatively new term, which entails using plugged in EVs to reduce the load imbalance strain on the electricity grid. EVs are able to relieve load strain by feeding power back from their batteries into the grid, hence the name of VTG. Load strain could also be relieved by means of a car removing its own demand, meaning that the EV stops charging until a later time when there is less strain on the network. It is important to note that EVs belong to an owner who has to decide on whether they would like to offer their battery, or a percentage of it, for use in VTG. This is referred to as the ”VTG percentage” in the report. How potential VTG capacity is spread across time and on which scale this capacity could contribute to relieving load strain on the electricity grid remains uncertain.

In order to be able to use EVs as a type of electricity buffer, it is important to know when and where this capacity is available. Unlike any other electricity production or storage method, EVs are mobile and can move, charge and discharge throughout the day. For example, EV users could prefer to charge at home, resulting in more plugged-in EVs for potential VTG at night, or during the day, resulting in more potential for VTG during the day.

This report aims to shed light on the uncertainties regarding EV load demand and VTG capacity over time. Our primary research question is:

What are the total VTG capacity and the power demand of all plugin electric vehicles in the Netherlands over time, given the preference to charge at home or at work, the allowed VTG percentage, and the percentage of cars that are smart charging?

1.2. Chosen modelling approach

To answer the research question, an agent based model (ABM) will be utilized. An ABM is favoured over system dynamics models, because it supports the modelling of low level EVs. Furthermore, it enables us to model a heterogeneous population of EVs in an intuitive way. Compared to other modelling approaches, ABM is the most favoured here. System dynamics models depart from a more aggregated perspective, and therefore lose essential information. Furthermore, the main advantage of an ABM, compared to a discrete event model, is that interaction between model components is easier to implement.

1.3. The energy transition model

This project is part of a larger modelling project: The energy transition model (ETM). The ETM is a model developed by a company name Quintel, and is primarily used to inform Dutch decision makers about the consequences of energy related policies. The energy transition model has a broad scope, is open source, and can be used through an interactive webpage.

The scope of this project is to integrate the estimation of the Dutch VTG capacity into the ETM. This poses two key challenges. The first challenge, is that the runtimes of an ABM are generally rather long. This is a problem, as the ETM is designed to give users interactive feedback in a web browser. The second problem, is the fact that the ETM is based upon a set of mathematical equations, and does not support feedback. This means that the output of the ABM can be used to perform further computation, but the output cannot be generated by means of interaction between the ETM and the ABM.

We aim to solve these challenges, by developing an ABM that generates a set of profiles, based upon a set of input variables. Some relevant input variables are the amount of smart charging vehicles, the location that users prefer to charge (work or home) and the amount of EV owners that allow their EV to be used for a larger VTG capacity. The variation of these input variables will change the output profiles, which can be integrated into the ETM, and can be used to perform further computations. This way, interaction between the ETM and the ABM is not needed during runtime.

1.4. Conceptualization of the vehicle to grid system

To estimate the VTG capacity using an ABM model, first a conceptualization is required. The first section of this chapter will guide the reader through a story line of the most relevant agent type in the model, the electric vehicle. Afterwards, useful details of the environment will be discussed. Finally, this chapter will cover a more detailed conceptualization, which includes all variables and processes which we plan to use in the final computational model.

1.4.1. The main agent: an EV.

To estimate the VTG capacity in the Netherlands, we choose to model EVs as the main type of agent in our model. EVs have a battery, and can either stay at home or travel to work. Figure 1.1 and figure 1.2 contain an illustration of a regular day that an EV goes through. Section 1.4.1 contains a storyline of an EV.

The agent's story

Early in the morning, a person wakes up and is ready to travel to work. This person's EV is charged and ready to leave. If it's time to leave, the person rides his EV to work and empties its battery. When the EV has arrived at work, this person decides whether to plug in his EV at work. The EV will now start charging. Depending on whether his car uses smart charging or not, the battery will be charged immediately, or later on. When the user is finished working, he plugs off his EV and starts driving home. Once arrived home, the person decides again whether to plug in the car or not. This cycle repeats the next day.

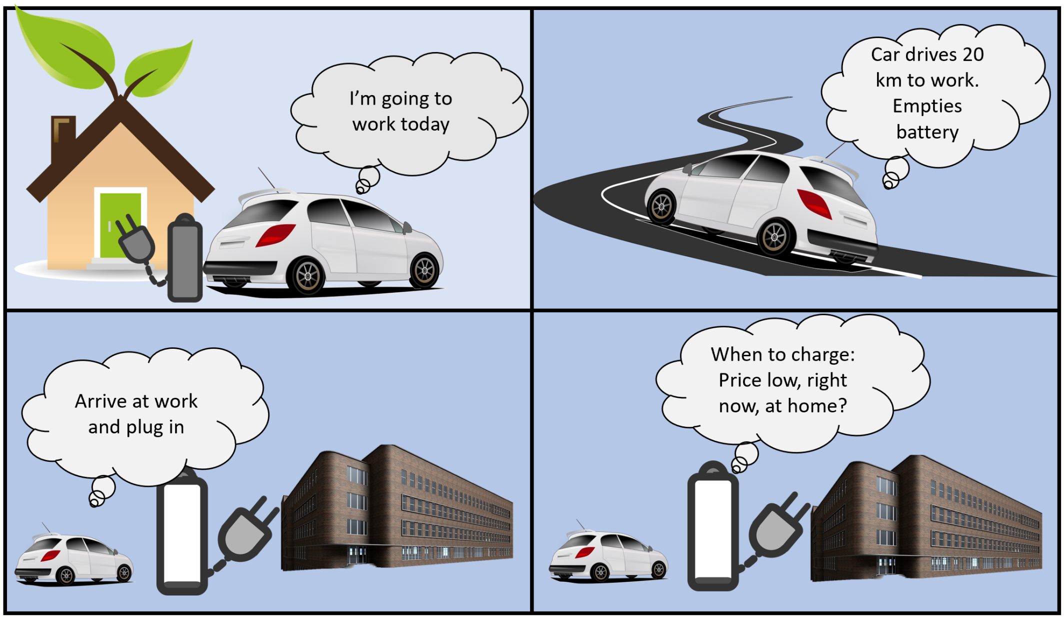

Figure 1.1: Story of Electric Vehicle Agent (1/2)

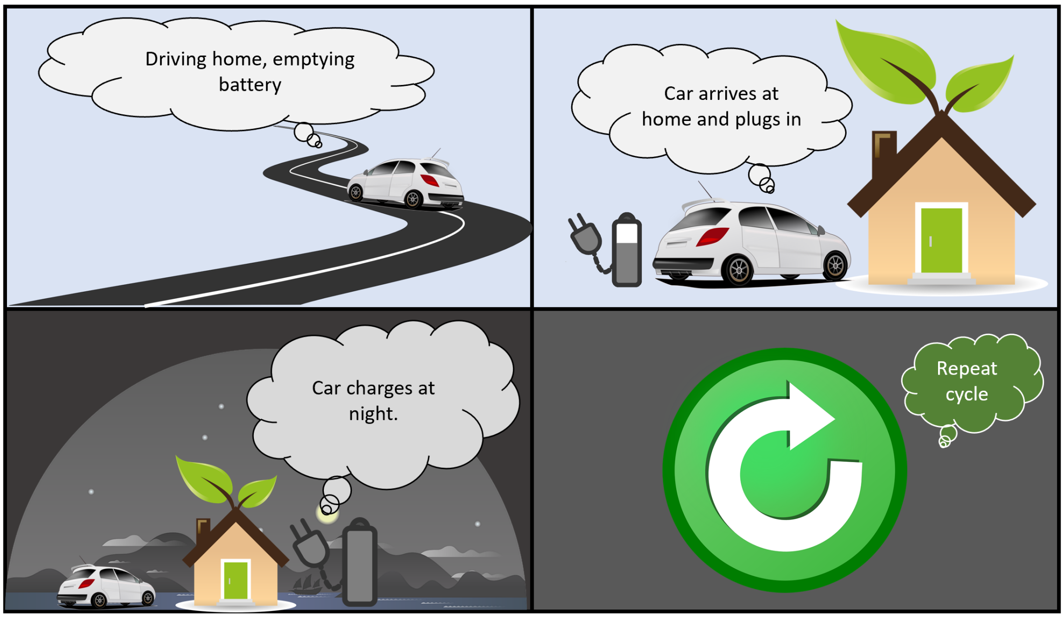

Figure 1.2: Story of electric vehicle agent (2/2)

1.4.2. Configuration of the environment

All EVs in the model have a working location, as well as a home location. Each consecutive day, the agent travels between its house and its work location. To make this work, the environment of the model consists of multiple municipalities. The municipalities function both as work locations and home locations at the same time. Municipalities must have some basic properties. For example, municipalities need a location and the amount of inhabitants.

Travel demand between municipalities

To determine how many agents travel between the municipalities, each of the EV's is assigned a home and work location. We aim to do this based on the distances between the municipalities, as well as the size in terms of inhabitants of the municipality. To calculate the amount of agents travelling between two municipalities, we aim to use a gravity model. This model utilizes both the distance between both municipalities and the amount of inhabitants in such a municipality.

1.4.3. Variables and processes

This section covers the relevant logic to give an EV realistic behaviour. First, the logic relating to the agent's travel behaviour will be discussed. Secondly, the logic related to the agent's charging behaviour will be discussed. Finally, a UML diagram is presented that contains the variables that we aim to us in the final model.

Each time step, an EV will first 'flow' to the logic drawn in figure 1.3. The logic in this figure is used to change the location of the EV. The rectangular blocks in the figure are used to indicate a decision. In these blocks, the model checks whether the listed condition is true for each EV. The model starts by checking whether the current time step is equal to its departure time and whether the agent is at home. If these conditions are true, the EV departs to work. The departure to work is a method, and is denoted by an oval shape in the diagram. The rest of the logic can be seen in figure 1.3.

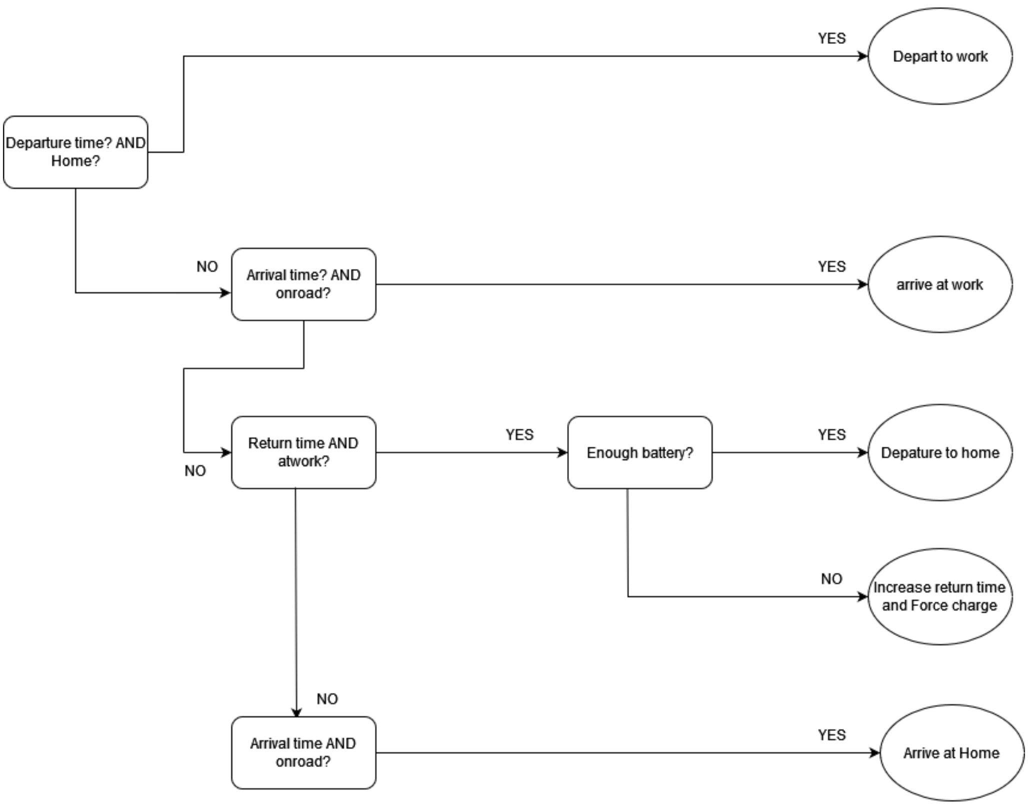

Figure 1.3: Step diagram of electric vehicle agent, decision scheme to depart or not.

After determining the location of the EV, the EV 'flows' through the logic shown in figure 1.4. This logic is used to decide whether to charge the EV or not. The logic starts by checks whether the EV is at the road or not. If the agent is on the road, the energy level is reduced. If the agent is not on the road, the model checks if the agent has enough energy to depart in the future. The rest of this process is visualized in figure 1.4

Figure 1.5 shows the relevant classes in the model: the overarching model, the EV and the Municipality. Attached to the classes are relevant attributes. This diagram was made before the actual model construction, therefore some variables have been added during development. However, we sticked to this UML as much as possible.

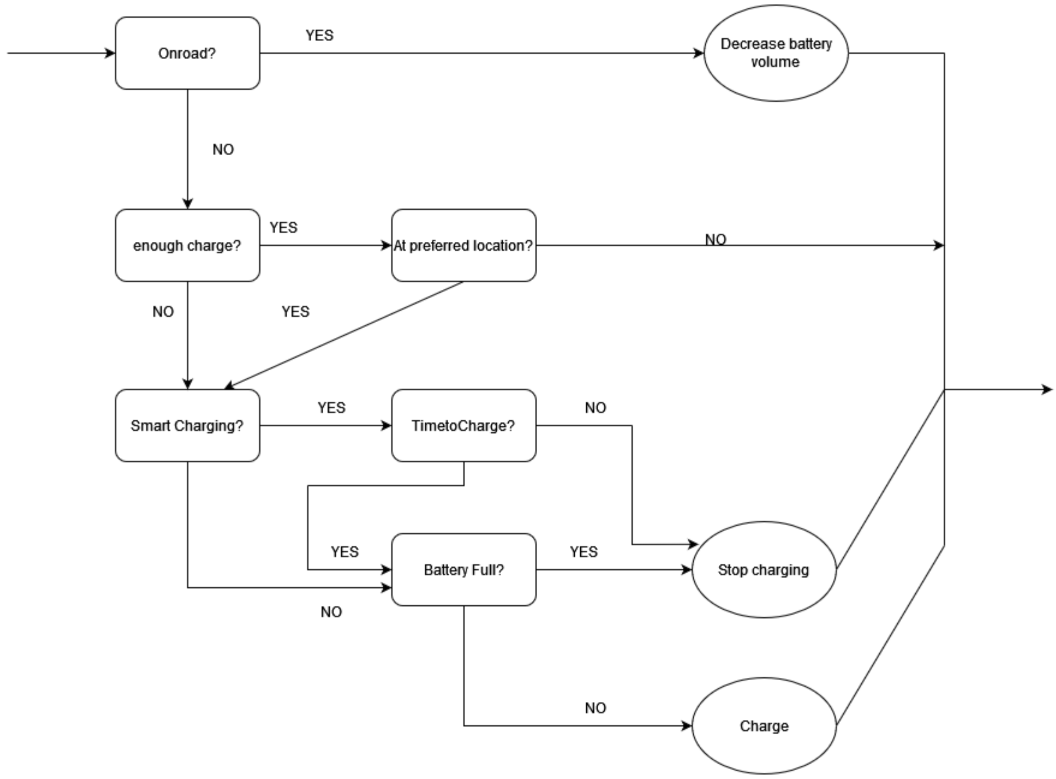

Figure 1.4: Step diagram of electric vehicle agent, decision scheme to charge or not.

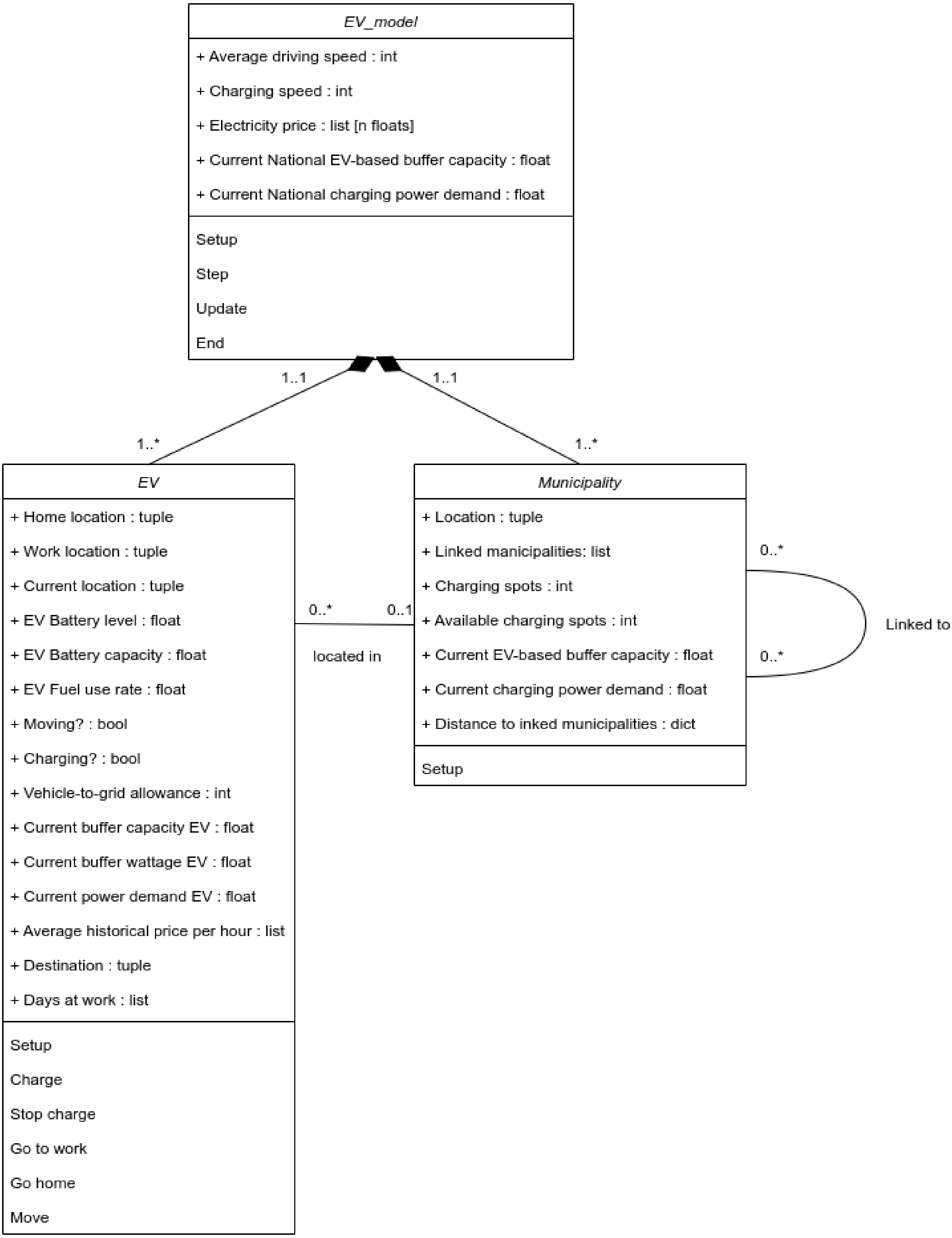

Figure 1.5: Initial UML diagram of the model

2 Model construction

An agent-based model is build in this research using the programming language Python and the modelling framework Agentpy (Foramitti, 2021). This section covers the choice of timestep and the implementation of the model conceptualization, as covered in 1.4, in Python code. Furthermore, a discussion regarding model verification and runtime optimization is included afterwards. Full model code is made publicly available [1].

2.1. timestep

A crucial decision in any Agent-based model is the timestep. The timestep for the EVs model is based on several facets:

1. System characteristics.
   The raison d'être for this model lies in the potential of VTG for alleviating the larger problem of electricity grid imbalance. TenneT is the actor responsible for managing this imbalance (TenneT, 2022b). TenneT regulates imbalance on the grid in 15-minute intervals, during which all kinds of buffer capacity can be used (**acm_transparantie_2004**).
2. Smallest modelled action
   The model, in short, simulates vehicles performing the actions "driving" and "charging". Both of these actions are continuous, meaning they do not adhere to a discretized time to complete. For now, we assume that the smallest time step in which either of these actions could meaningfully change the EVs state in the natural system is 1 minute.
3. Run time The modelling language used for this model is Python. The computational resources available to run the model are the researchers' home computers and the computational power available through educational channels. Nevertheless, given a set simulated time span, say one year, the model would run faster given a larger timestep (e.g. 1 week) than a smaller timestep (e.g. 1 minute). A larger timestep could enable more runs or faster runs, while a smaller timestep could increase the resolution of the model (Natrajan et al., 1997).

Given the facets presented above, the timestep for the model was set at **15 minutes**. A timestep of 15 minutes is in line with the time interval of the imbalance market, which makes the model more fit for use. This timestep size creates a lower resolution in which actions can be done, which especially impacts EV charging and driving behaviour. However, 15 minutes was still kept as the chosen timestep. Firstly, the timestep was not lowered because of computational restraints. Secondly, it was not lowered because the assumption was made that a maximum error in driving or charging of 7.5 minutes compared to the natural system would not cause significantly different outputs, given a large number of agents.

[1] https://github.com/floristevito/VTG-Dynamics

2.2. Encoding

The central agent in this model is called the EV-agent (EV). This agent represents of an EV and the charging preferences of its owner combined. The Netherlands counts around 174,000 EVs (CBS, 2021a). EV agents can vary between a few core states, namely being on the road or being stationary, either at work or at home. Stationary EVs are able to charge, which is based on the need for electricity (to get from point A to B) and the charging preferences. When and where EVs charge eventually creates output in the forms of two KPI's: the power demand in KW and the vehicle to grid (VTG) capacity in KW. Each subsection below will walk through how these states and outputs were encoded.

2.2.1. locations and driving

EVs drive from their home to their work. The locations of an EVs 'home' and 'work' are assigned at the start of the model and do not change. The amount of EVs that travel between certain location is based on a gravity model, similar to the model used in the work of Jung et al. (2008). Using the gravity model, the flow between two municipalities centroids was calculated. The gravity model takes the distance between the municipalities and the amount of inhabitants as inputs. The data about the amount of inhabitants was collected from CBS. Furthermore, spatial data was used to compute the distances. The spatial data can be seen on a map 2.1. Within this map, the centroids of all municipalities are indicated by green dots. The highways are indicated by red lines. Software by Raffler (2021), named QNEAT3, was used to compute a distance matrix. The distances between the centroids were quantified, by measuring the shortest distance between two municipalities over the road network. This was possible, as the coordinates of the roads, and centroids were transformed to the 'rijksdriehoekscoordinaten stelsel' (RDS) (Kadaster, n.d.). RDS is a local coordinate reference system, which allows measurement of distances.

At the start of a model run, a user provides how many EVs the model should have. Based on the relative population size of a municipality, each municipality is given a number of EVs. This is done cautiously, to make sure that the percentage of inhabitants that own an EV is constant in each municipality.

The destination of each EV is determined by the gravity model. For each municipality, the relative importance of each connected flow is computed. A probability for each flow was computed, based on the relative importance of each link. Each EV samples from this set of probabilities, to get their own destination.

2.2.2. charging

Charging is the central function of the model. Through examining the various ways, locations and levels on which EVs charge, lessons can be learned on how the grid is loaded throughout the day in the Netherlands. In the model, an EV is able to charge when stationary and plugged into a charger. The charging speed per charger is determined during model setup and does not change during the run. There are three rules that limit battery charging in the model:

1. Battery percentage must stay between 0 % and 100 %.
2. An EV is not allowed to end a charging with less charge than it started with, for any reason. This creates a 'Lower Bound (lb)', which will be used to calculate available VTG capacity.
3. EV's are free to choose which timesteps to charge on (e.g. price-dependent or not), as long as they reach their desired charge on the scheduled time of leave. Figure 2.2 shows that at for a given EV at t2, a certain charge must be in the battery. Then, for any certain set charging speed of the EV, a linear function can be calculated that shows when the EV can no longer postpone charging if it were to meet the desired charge on time. This bound on charging is called the 'Latest charging bound (Lcb)' and will also be used for VTG capacity calculation.
4. If there are insufficient timesteps available to reach this desired charge on time, the EV will postpone the scheduled time of leave and forcibly charge every timestep non-stop until the desired charge is reached. In this forced state of charging, no room for leniency to accommodate VTG capacity exists.

In the model, cars are able to either charge 'smart' or not. This entails basing charging times on the lowest price if possible, or not. Example graphs of charging behaviour can be found in 2.3.5.

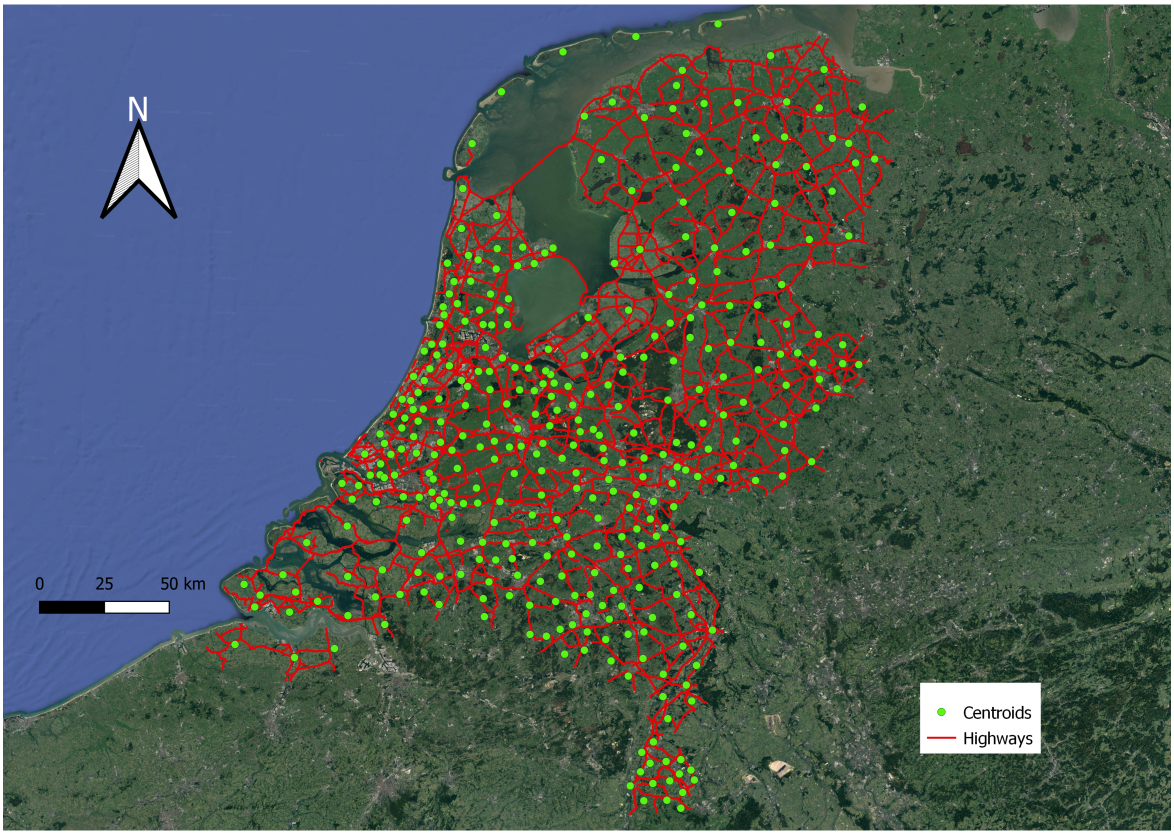

Figure 2.1: Data used to compute the distances. (satellite image from Google)

2.3. KPI implementation

Now that EV agents are able to move, charge and discharge, the KPI's from the research question can be implemented. Section 2.3.1 highlights how the KPI ”power demand” was implemented. Section 2.3.2 highlights how the KPI ”VTG capacity” was implemented.

2.3.1. power demand

Power (electricity) can be requested by EVs when charging. When and where EVs charge is based on their travelling behaviour. EVs are able to charge at home or at work. In the model each of these locations is assigned an unlimited amount of charging locations that EVs can use. For a given charging location a charging speed is assigned, for example 20 KW. A single charging EV would create a graph that created bumps of charging over time, as illustrated in figure 2.4 and 2.7.

2.3.2. VTG capacity

The VTG capacity is based on two factors:

1. Postponing a current demand for charge (see 2.3.3)
2. Feeding back electricity into the grid (see 2.3.4

This creates equation 2.1 for VTG capacity, which is calculated per EV every timestep during charging. In order to collect the total VTG capacity in the model, the capacity from 2.1 is summed as seen in equation 2.2.

$$VTG_{EV} = VTG_{postponing\ charge} + VTG_{feeding\ back\ electricity} \tag{2.1}$$

$$VTG_{model} = \sum_{i=1}^{n} VTG_{EV}\ for\ n\ in\ model.EVs \tag{2.2}$$

2.3.3. Postponing charge

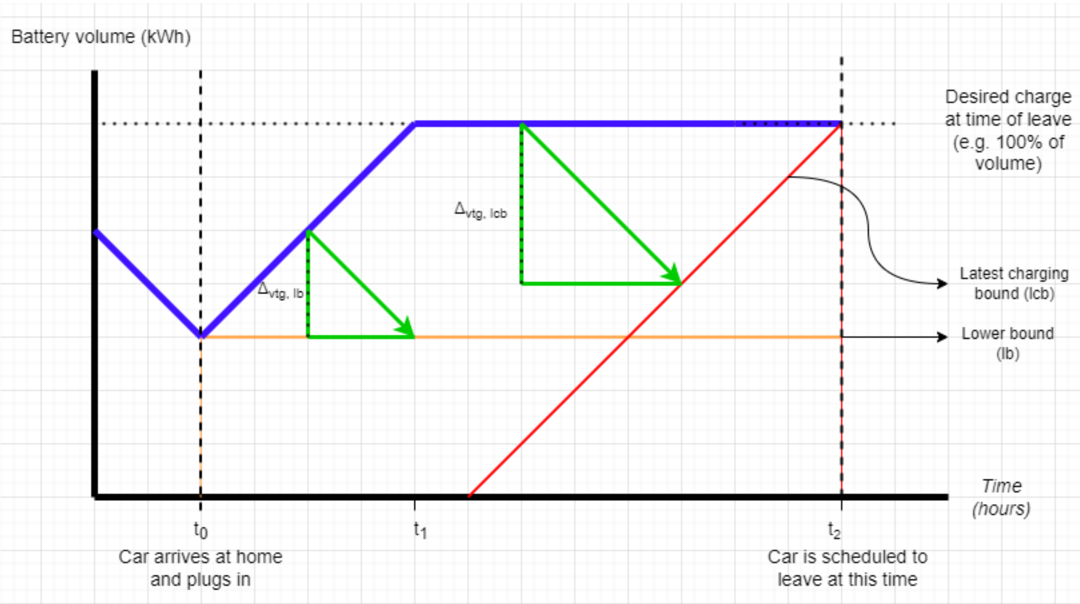

Figure 2.2: example EV with charging and VTG calculation lines

The gained VTG capacity of the first factor, postponing a charge, simply equals the wattage that an EV would have charged at. Even though the EV is not directly providing electricity to the grid, removing a demand for power has an equivalent balancing effect. For an EV charging at 20KW, the VTG capacity from postponing that charge would be 20KW. However, there are limits regarding when charges can be postponed. Charging cannot be postponed if postponing would lead to the desired charge being unable to be reached. Therefore, charge postponing is allowed as a postponement would not lead to the charge going under the Lcb bound. This was done using equation 2.3, implemented in python as 2.4, using values from figure 2.2:

$$battery\ volume\ at\ t_{t+1} > Lcb\ charge\ level\ at\ t_{t+1} \tag{2.3}$$

$$\begin{aligned} self.current\_battery\_volume - self.charging\_speed * 0.25 > \\ self.needed\_battery\_level\_at\_charging\_end - \\ (self.charging\_speed * 0.25 * (self.time\_charging\_must\_finish - self.model.t)) \end{aligned} \tag{2.4}$$

2.3.4. Feeding back electricity

The gained VTG capacity of the second factor, feeding back electricity, is based on how much capacity the EV is able to provide, whilst remaining within the allowed boundaries. Aside from the Lcb and LB, the model also includes a user-changeable parameter regarding the maximal percentage of EVs batteries that can be used for feeding electricity back. The resulting VTGfeedback for EVs on a given timestep is then calculated by examining equation 2.5, calculating which boundary is most limiting.

$$VTG_{feedback} = min(VTG_{lb\ limited}, VTG_{lcb\ limited}, VTGuser\ set\ percentage\ limited) \tag{2.5}$$

As seen in figure 2.2, the VTG_{lb} and VTG_{lcb} are determined by first calculating a linear intersection between a discharge line and the Lb or Lcb. The VTG is then calculated by subtracting the Y coordinate of the intersection from the Y coordinate of the starting point. The Y axis here refers to the battery volume in kWh, as seen in figure 2.2. Equation 2.6, implemented in python as 2.7, is used to acquire the VTG_{lb} value for any given EV. Equation 2.8, implemented in python as 2.9 and 2.10, was used to acquire the VTG_{lcb} value for any given EV.

$$VTG_{lblimited} = Y_{initialcharge} - Y_{lbintersect} \tag{2.6}$$

$$VTG_{lblimited} = self.current\_battery\_volume - self.battery\_level\_at\_charging\_start \tag{2.7}$$

$$VTG_{lb\ limited} = Y_{initial\ charge} - Y_{lcb\ intersect} \tag{2.8}$$

$$\begin{aligned} Y_{lcb\ intersect} = 0.25 * self.charging\_speed* \\ (-Intersection\_Xcor\_lcb + self.model.t) + self.current\_battery\_volume \end{aligned} \tag{2.9}$$

$$\begin{aligned} Intersection\_Xcor\_lcb = 0.5 * (self.time\_charging\_must\_finish + self.model.t) + \\ ((2/self.charging\_speed) * (self.current\_battery\_volume - \\ self.needed\_battery\_level\_at\_charging\_end)) \end{aligned} \tag{2.10}$$

2.3.5. Example cases

In order to illustrate the expected charging behaviour of EVs and the resulting power demand and VTG KPI's, this section will walk through two example cases. Case one will follow a non-smart EV, meaning it does not charge based on electricity prices. Case two does follow an EV with smart charging, resulting in different charging behaviour. The graphs in this section display the expected behaviour of the model. After construction was completed, the model behaviour matched the expected behaviour. Reference graphs from the model can be seen in A.1.

Case 1: non price-based charging

This case will walk through an example of the behaviour of a non-smart-charging EV. The charging behaviour, see 2.3, of the EV will be discussed, along with the corresponding KPI graphs for power demand, see 2.4, and VTG, see 2.5.

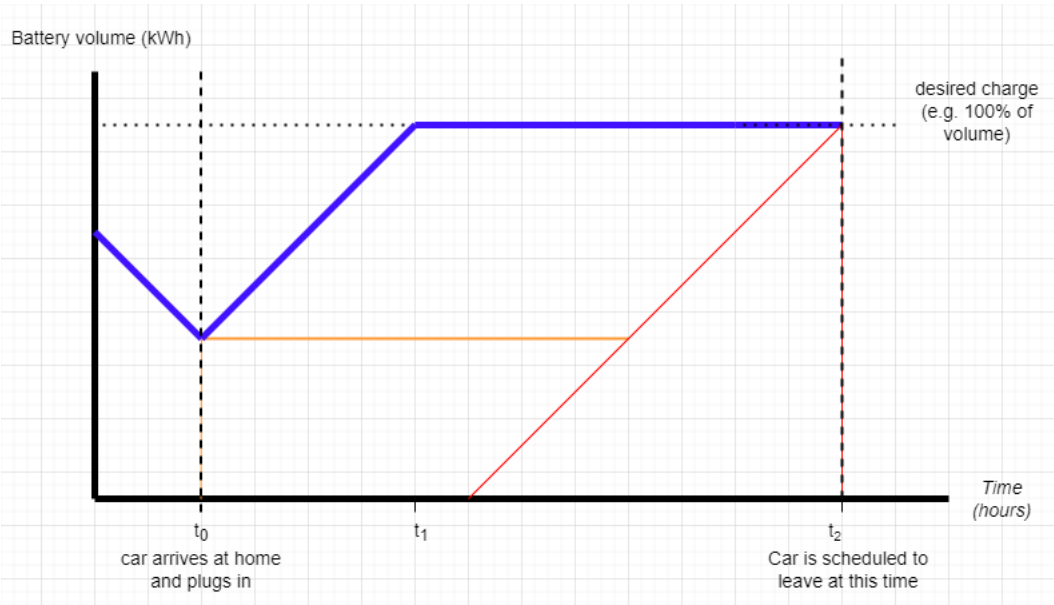

Figure 2.3: non price-based charging graph of an EV

Figure 2.3 shows the charging of an EV battery that plugs in and starts charging as soon as it arrives at home (t_0). At t_1 the battery has reached its desired level and goes idle until t_2. The demand for power in this graph is fairly straight-forward and represented in figure 2.4.

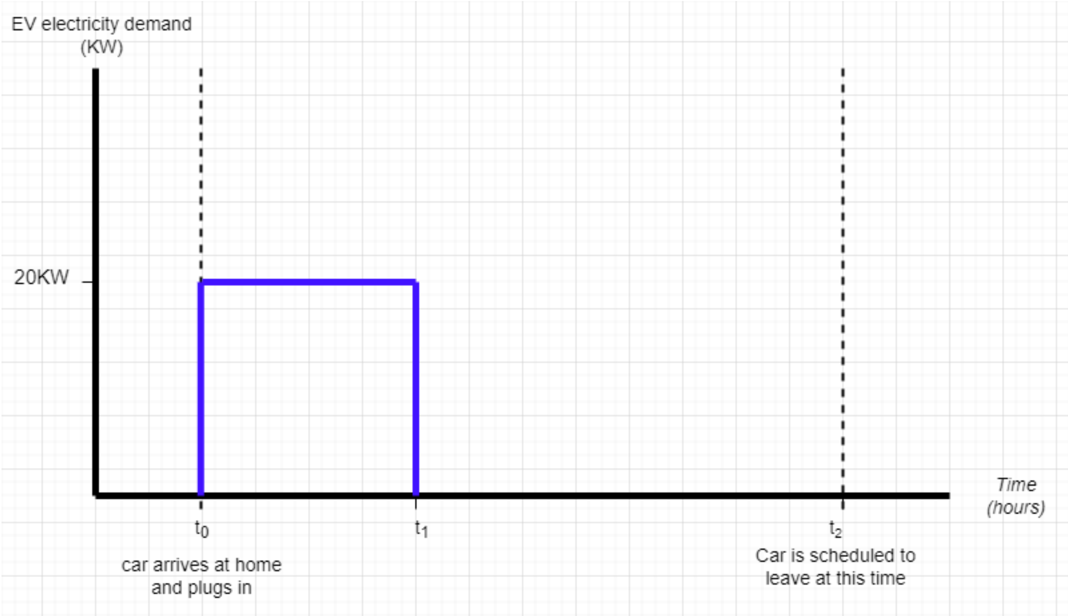

Figure 2.4: EV electricity demand resulting from figure 2.3

The EV is plugged into a 20KW charger and produces a single block of power demand. Figure 2.5 shows the VTG availability for this EV.

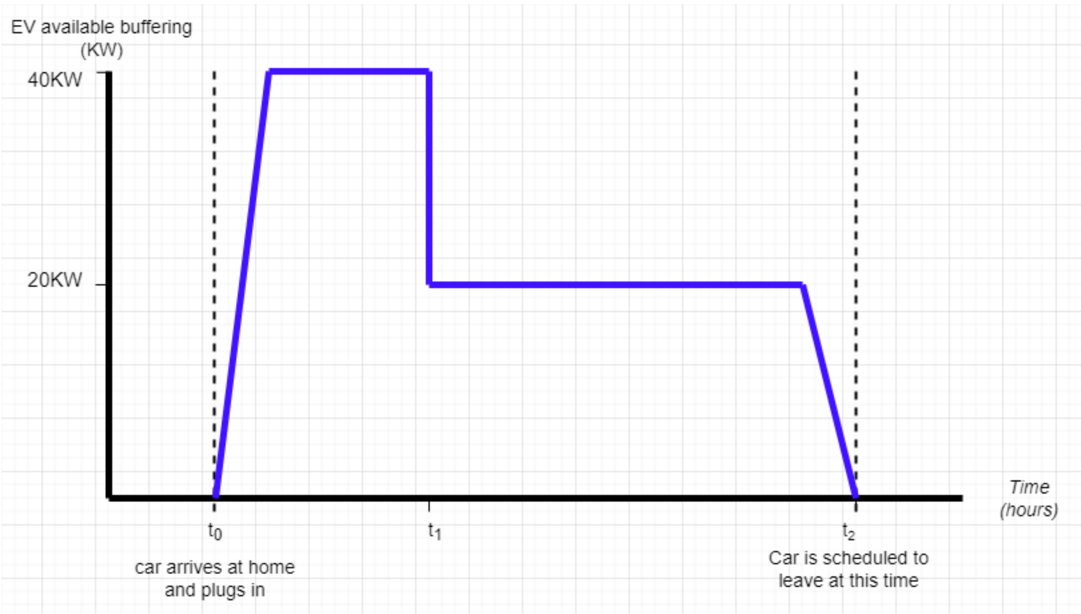

Figure 2.5: EV VTG availability resulting from figure 2.3

Figure 2.5 shows that this EV is most capable of providing VTG capacity between t_0 and t_1. In this time the battery is actively charging and has enough charge to feed back into the grid without hitting a lower bound. This results in a $20 + 20 = 40KW$ VTG capacity from $t_0 - t_1$. T_0 starts diagonally because at t_0 it cannot feed back power yet without hitting the lower bound.

Case 2: price-based charging
This case will walk through an example of the behaviour of a smart-charging EV. The charging behaviour, see 2.6, of the EV will be discussed, along with the corresponding KPI graphs for power demand, see 2.7, and VTG, see 2.8.

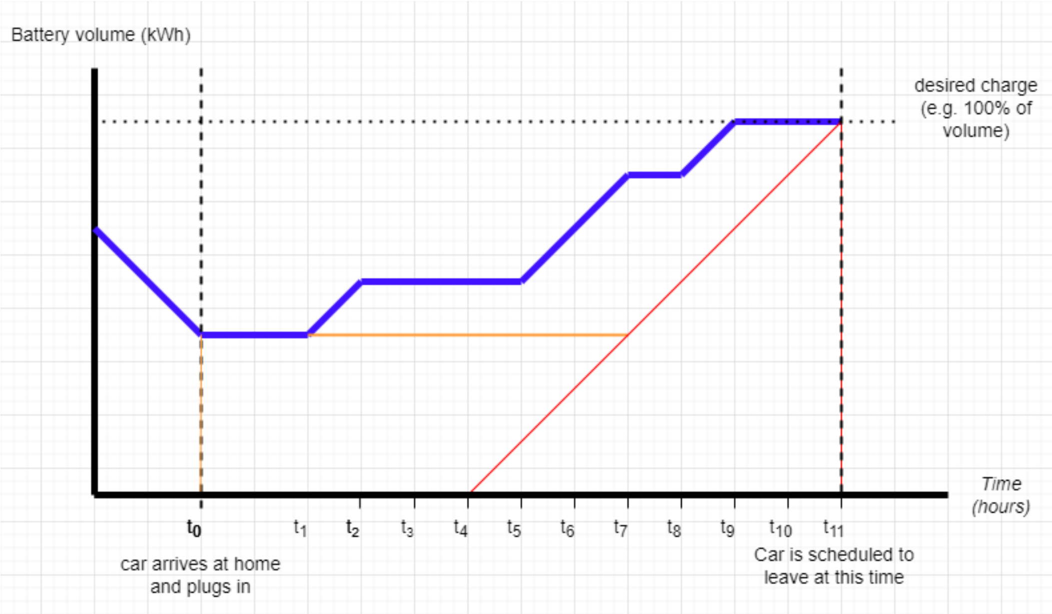

Figure 2.6: price-based charging graph of an EV

Figure 2.6 shows an example of how an EV would “smart charge” based on electricity prices. In this example, electricity was cheapest between $t_1 - t_2$, $t_5 - t_7$ and $t_8 - t_9$ to fill the battery to the desired level in time. This EV does not start charging immediately at t_0, but instead remains on the lower bound until the electricity becomes cheaper. Figure 2.7 and 2.8 show the charging demand over time and the VTG availability.

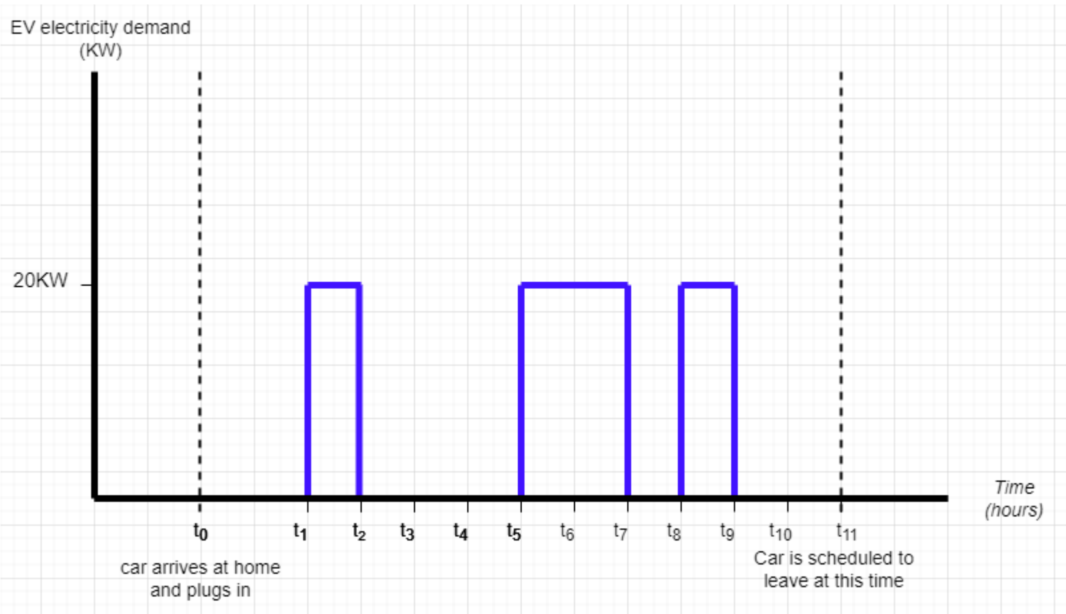

Figure 2.7: EV electricity demand resulting from figure 2.6

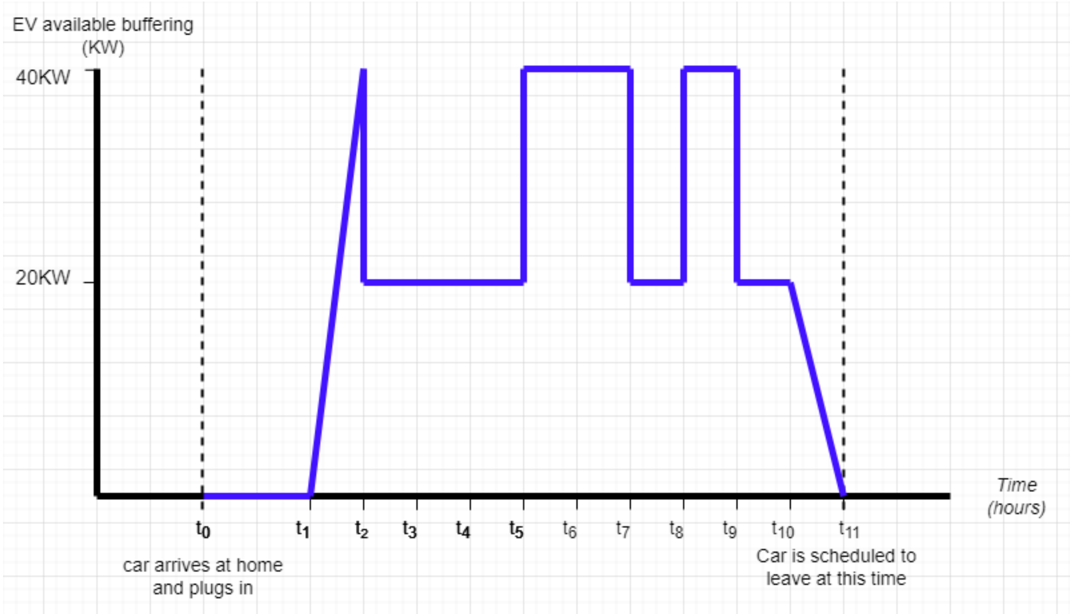

Figure 2.8: EV VTG availability resulting from figure 2.6

Figure 2.8 shows some interesting behaviour. From t_0 to t_1, the EV is unable to provide any VTG capacity, due to the battery not being charged and remaining on the lower bound for battery level. Postponing charging is not possible in this interval, because the EV is not charging in the first place. Feeding back electricity to the grid would bring the EV's battery under the initial charge level, which is not allowed by the model. After charging from $t_1 - t_2$, there is more charge in the EV than the initial level. This means that during $t_2 - t_5$, while the battery sits idle, it could feed back some electricity to the grid. Keep in mind that electricity can only be fed back once, and that therefore **one cannot simply integrate this graph to calculate the total VTG availability**. A calculation would have to be done in combination with the remaining charge in the EV in figure 4. It is also important to note that any use of VTG will affect the remaining charging behaviour. During $t_5 - t_7$ and $t_8 - t_9$ the EV charges further and is able to deliver 40KW of VTG buffer capacity. At t_9 the EV is at full charge and remains idle. It is however still able to feed back 20KW to the grid. This eventually lowers down to zero from $t_1 0$ to $t_1 1$, because of the red bound in figure 2.6.

2.4. Verification

To make sure that our model is encoded correctly, and behaves as intended, we have performed multiple verification methods. First of all, a large portion of this verification is performed in an informal manner. Hereby meaning that it was done ad-hoc, during model building and in an unstructured way. This enabled us to quickly recognized poorly implemented model functionalities. Secondly, a more formal verification approach was also performed using structured logs and python tests. Additionally, a consistent run with various seeds was performed to get an insight into the effect that random variable have on the outcomes of the model.

2.4.1. Initial verification of output

We verified model implementation in an ad-hoc style using various methods. Often, performing quick runs with a small runtime and fewer EVs, provided a lot of useful insights in model working. Using model reporters and dynamic records, overall behaviour could be verified. For this, we used an automatic model run file, that took parameters from a specified JSON file and outputted model behaviour plots. An example of such as plot of a quick model run with 17000 agents simulating one week can be seen in 2.9. Note that these plots only served as a quick method for verification.

Furthermore, the included Python debugger in Visual Studio Code often came in handy after weird or wrong behaviour was observed using the quick model runs. Debugging using breakpoints helped us to establish what went wrong. One good example of this was finding mistakes made in the VTG

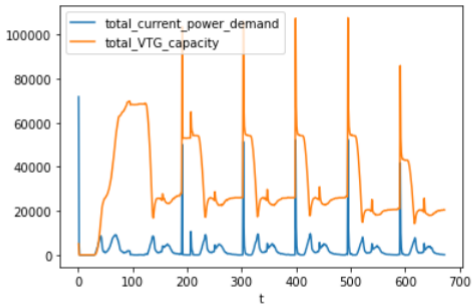

Figure 2.9: test run output, $N_{EVs} = 17000$, 672 steps

calculation. This calculation is one of the more complex elements in the model, as discussed in section 2.3.2. We often found small mistakes here in the beginning, resulting in $None$ values of variables that should not be $None$, breaking other model functionalities. The debugger enabled us here to clearly trace back the problem to its origin.

2.4.2. Structured verification

Two methods for verification were more structured: verification by model logs and verification using python tests.

For model logging, a mechanism was implemented that logged all relevant model events to a separate .log file. Here, we made the distinction between info logs, debug logs and warnings. Upon model initialization, the logging level can be changed to only show the logs that we wanted to see. An example of this log file can be seen in figure 2.10. Here, the logging level is set to info, meaning that only info logs and warnings are displayed, leaving out debug logs as they would clutter the file significantly. This model log provided us with a structural way to verify model behaviour. Furthermore, this log also served as a history file regarding what happened during runtime.

Python tests were constructed to test model implementation in an even further structured and automatic way. These tests, made with the Pytest library, are all tests specific, isolated, parts of our code. We divided these tests in two categories: model tests and component tests. The main difference between both is the level that the evaluation takes place. The components test are specifically written to test component logic, such as the EVs and municipalities in the model, and their behaviour. The model tests are instead focused on model level logic, such as the number of all agents in the model. All python tests are included in separate test files, that can be found in the same directory as the model files.

```
model.log
1   root - INFO - Model init completed in 2.645051527999385 seconds
2   root - INFO - MODEL CONFIGURATION
3   root - INFO - EVs in model: 1740
4   root - INFO - Municipalities in model: 352
5   root - INFO - average battery volume of EVs (kWh): 58.90170194567298
6   root - INFO - average energy rate of EVs (kWh/km): 0.1872941050659451
7   root - INFO - 1 it's no weekend.
```

Figure 2.10: Model log file

2.4.3. Seeds

The constructed model makes use of various randomly determined variables in order to mimic real-world behaviour as good as possible. Ideally, for the results in terms of profiles, we would use multiple iterations in order to compensate for the effect of the random single-run occurrences on the outcomes. However, given the large number of profiles (484 profiles) and the long runtime of a full year (35040 steps), also see section 2.5, the use of multiple iterations was not doable given the available computational power. Therefore, we decided to test the order of the effect different seeds have on the model outcomes on a single profile. This one profile ran for two weeks (1344 steps) and was repeated twenty times, only varying the seed. Results of this analysis can be seen in table 2.1.

Table 2.1 shows that varying the seeds has little effect on model outputs. The standard deviations of all outputs fall within 1% of the mean value, meaning that the majority of model runs are quite close to the mean. Looking at the min and max outliers, these runs also do not produce substantial deviations from the mean. From this we conclude that creating profiles without repeating them all will not cause any seed-related randomness issues.

Table 2.1: statistics about the mean value of reporters with 20 different seeds for a profile run with all EVs

	average_battery _percentage	total_current_ power_demand	total_VTG _capacity	mean_charging
count	20	20	20	20
mean	73.97	25362.06	284911.08	0.015192
std	0.09	47.04	862.49	0.000035
min	73.74	25284.85	282898.90	0.015132
25%	73.91	25339.06	284590.09	0.015169
50%	74.00	25360.44	285015.30	0.015190
75%	74.06	25386.88	285272.77	0.015222
max	74.07	25472.81	286356.75	0.015269

2.5. runtime optimization

One issue we faced frequently during the model building and verification were significantly long runtimes. Due to the large amount of heterogeneous EVs, generated precisely according to input data files calibrated on the real world, initialization and running all model steps took a very long time. We performed countless iterations to improve the model for faster runtimes. In the end, we succeeded in bringing the model to a reasonable runtime by performing a series of steps. This total runtime will depend on the exact model settings and hardware. For a full profile run, 174000 EVs and two weeks (1344 time steps), it takes about 50 minutes on a desktop machine equipped with an AMD Ryzen 5000 7 series (Ryzen 7 5700g).

Firstly, we optimized the model initialization according to the data file by minimizing specific row based references to the input data files. These files include specific EV attributes, like destinations and energy usage. Given numerous agents, this initialization was very slow. Optimizing this initialization process brought runtimes back up to a factor of 30 in situations with a large number of agents.

Secondly, we limited the model runtime for experimentation and the profile calculation to less than our desired total length of one year. We observed repeated weekly behaviour, as expected, due to no programmed interaction between EVs. Therefore, we limited the total runtime to two weeks, or 1344 steps of 15 minutes for the calculation of profiles. The first week is hereby seen as warm-up time, since all EVs started at a set battery volume. The resulting final week could then be repeated for a yearly pattern.

Lastly, we limited the number of EVs for some analysis, lowering the overall runtime. However, note that we did not do this for the profiles. The calculation of the profiles for the energy transition model was specifically according to the exact number of EVs in The Netherlands. This was done so that we acquired the exact behaviour and VTG capacity for a correct output that can be used in the ETM model.

3 Validation & experimentation

This section validates the model build by comparing its behaviour to real world numbers. Although a lot of assumptions were made to make this model, we believe that our model corresponds to the real system. Likewise, we believe that it captures the essence of the physical system. To strengthen this argument, a SOBOL global sensitivity analysis was conducted. On top of that, crucial parameters were calibrated to empirical values. These are summarized in section 3.1. Furthermore, a map of all the EVs was made to check for inconsistencies.

3.1. Calibration of the model

All key model parameters, summarized in table 3.1, were based on real world data. Other than the alignment of key parameters to empirical values, various actions were taken to make the model more realistic.

To make the model realistic, we computed the minimum driving range and maximum driving range of the EVs in the model to see if these were realistic. Afterwards, these were compared to real world values.

Other than that, the trip distances of the EVs were based on actual GIS data from CBS and open street map. These datasets included locations of highways, and the centroid locations of municipalities. These locations were used to compute distances between the municipalities, which resulted in a distance matrix with distances between all Dutch municipalities. This was done to make the driving distances as realistic as possible. A more detailed explanation can be found in section 2.2.1

Table 3.1: Used queries and databases

Name of model parameter	Source	Value
Number of EVS	CBS, 2021b	174000 EVs
Charging speeds	Van Barlingen and Simpson, 2021 RVO, 2018	min 20 max 60 kWh
Battery volumes	EV-Database.nl, 2021a	min 16.7 median 59.6 max 107.8 kWh
Energy usage	EV-Database.nl, 2021b	min 0.104 median 0.192 max 0.281 kWh
Inhabitants per municipality	CBS, 2021a	Dataset
Distance between home and work	CBS, 2018	22.7 km*

Note. * The model output was calibrated to this value, it is not a model parameter.

3.2. SOBOL analysis

We used the EMA Workbench (Kwakkel, 2017) to perform a SOBOL, a global sensitivity analysis. This analysis was used to see what level of direct (S1) and indirect (ST) impact varying the input variables has on the variation in output. A more detailed explanation of the input variables can be found in 4.1. Unfortunately, the SOBOL function in the EMA workbench did not allow for the output of dynamic variables such as graphs or time series. Instead, the performed SOBOL shows the effect of input variations on the mean of the two KPI's, power demand and VTG capacity, as shown in figure 3.1.

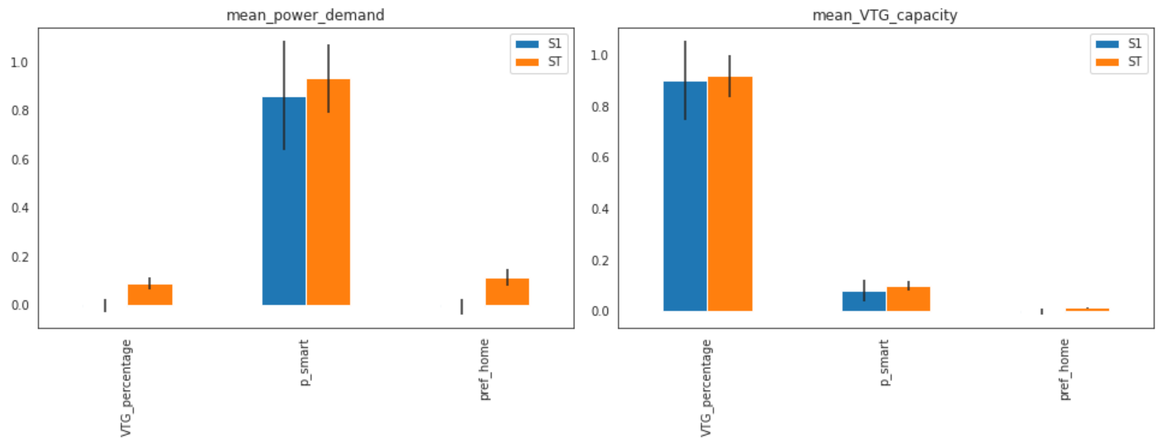

Figure 3.1: SOBOL analysis graphs

The left graph in figure 3.1 shows that the amount of EVs that charge smart has a dominant direct effect on the resulting mean power demand. One explanation for this is that, because all smart EVs examine the same electricity prices, a higher percentage of smart EVs will result in a higher peak outlier as seen in figure 2.9. This outlier then could affect the mean quite significantly. VTG_percentage and pref_home do seem to have an indirect ST effect on the output in the order of 10%, but no real direct S1 effect compared to p_smart. From this we can conclude that a large amount of EVs smart charging based on the same prices will result in high peak power demands. These short high peaks in power demand are undesirable from a grid balancing perspective, as it could disrupt the grid balance. Unfortunately, these peaks are desirable from an EV-users' perspective, as charging during the lowest-priced times is the cheapest. It is important to note however that the gap between these two perspectives is amplified, because charging behaviour has no effect on the electricity price in this model. The mean load on the Dutch power grid is approximately 12.4 GW (WorldData, 2018), with daily peaks around 17 GW and yearly peaks around 28 GW (TenneT, 2022a). All EVs in the model provide an average load in the order 100-300MW and a daily peak in the order 2GW. The average EV load is small compared to the average grid load and might not be significant. Although, if a feedback was present, it could potentially reduce the peak load problem.

Figure 2.9 also shows a significant peak in VTG capacity, caused by a peak in (smart) EVs charging. Even though this is the case, the right figure in 3.1 shows that p_smart is not dominant. The p_smart-created outlier still results in some direct and indirect effect of the VTG capacity, but the parameter VTG_percentage on its own almost fully explains the variation in VTG_capacity output. This observation reveals that, from the three bounds set on VTG_capacity in figure 2.2 and equation 2.5, the user-set VTG percentage is most limiting. This observation is a blessing in disguise, because this boundary is entirely man-made. From the SOBOL can be concluded that putting effort into convincing EV users to allow for more VTG feedback can result in significantly increased VTG capacity.

3.3. Map visualization of model run

To validate whether a typical run has a logical spread of EVs throughout the country, the average number of EVs was visualized in a map 3.2. What can be seen, is that EV's tend to concentrate in big cities. Rotterdam, The Hague, Utrecht and Amsterdam have the highest number of EVs during a model run.

This is in line to our expectation, because the model was configured to distribute EVs by the municipalities' population. Rotterdam, The Hague, Utrecht and Amsterdam are the largest municipalities in our model.

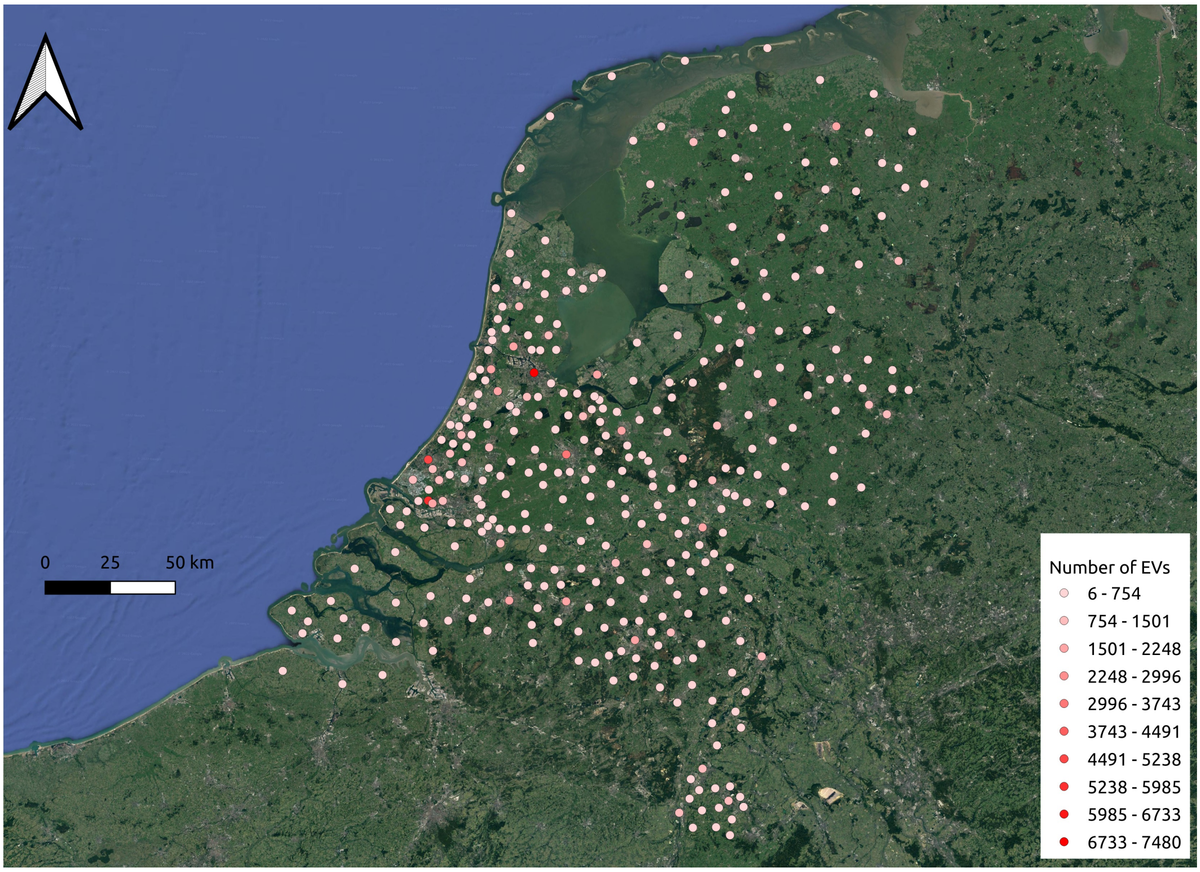

Figure 3.2: Average number of EVs per municipality during a typical run (satellite image from Google)

3.4. Validation against etm model

The profiles that resulted from the ABM model, were compared towards the intitial estimations of the VTG capacity in the ETM model. [1] Our model estimated about the same order of magnitude of VTG capacity.

[1] https://pro.energytransitionmodel.com/scenario/flexibility/flexibility_storage/batteries-in-electric-vehicles

4

Model application

The goal of this research was to build an ABM model that would be able to generate a set of profiles to be used in the energy transition model, as discussed in section 1.3. This section illustrates how the model was successfully used to generate these profiles and how they can be integrated into the energy transition model.

4.1. Generating profiles

From the stakeholder, three input variables were given that users should be able to variate in the original energy transition model. These variables are shown in table 4.1. For each combination of these variables, a full model run of our agent-based model reporting the VTG capacity per 15 minutes is needed to serve as input for the ETM model. This resulted in a total need of 484 full runs.

As discussed in section 2.5, the total runtime of the model is rather large. A full profile run, given all EVs in The Netherlands and a timespan of two weeks (where one week serves as a warm-up period), takes around 50 minutes. Hence, we decided to rent multiple virtual machines on the Microsoft Azure cloud environment. With free student credit, we were limited to 4vcp's per machine. However, we were able to split the work over three different machines, resulting in a doable runtime.

Table 4.1: Overview of input parameters for profile runs

Input variable	Description	Range	Resolution (smallest increment)	Total options
P_smart	The percentage of EVs in the model that charge smart	0-100%	10%	11
VTG_percentage	The maximal percentage of an EVs' battery that the power grid would be allowed to drain for VTG purposes	0-100%	10%	11
Pref_home	The level of preference of cars to charge either fully at home (1), fully at work (0) or a mix	0-1	4 discrete options: 0: preferably only charge at work 0.25: preferably charge at work 0.75: preferably charge at home 1: preferably only charge at home	4

After the full profile runs, we merged the output data to one full profile csv file. The warm-up period was removed, and the resulting week repeated so that one year worth of VTG capacity data per time step was acquired. Figure 4.1 shows the resulting VTG capacity pattern for all profiles from Monday to Friday. Note the lower peak on Monday morning (left peak) and the lower on Sanday morning (right one). This is due to a reduced travel behaviour on weekend days.

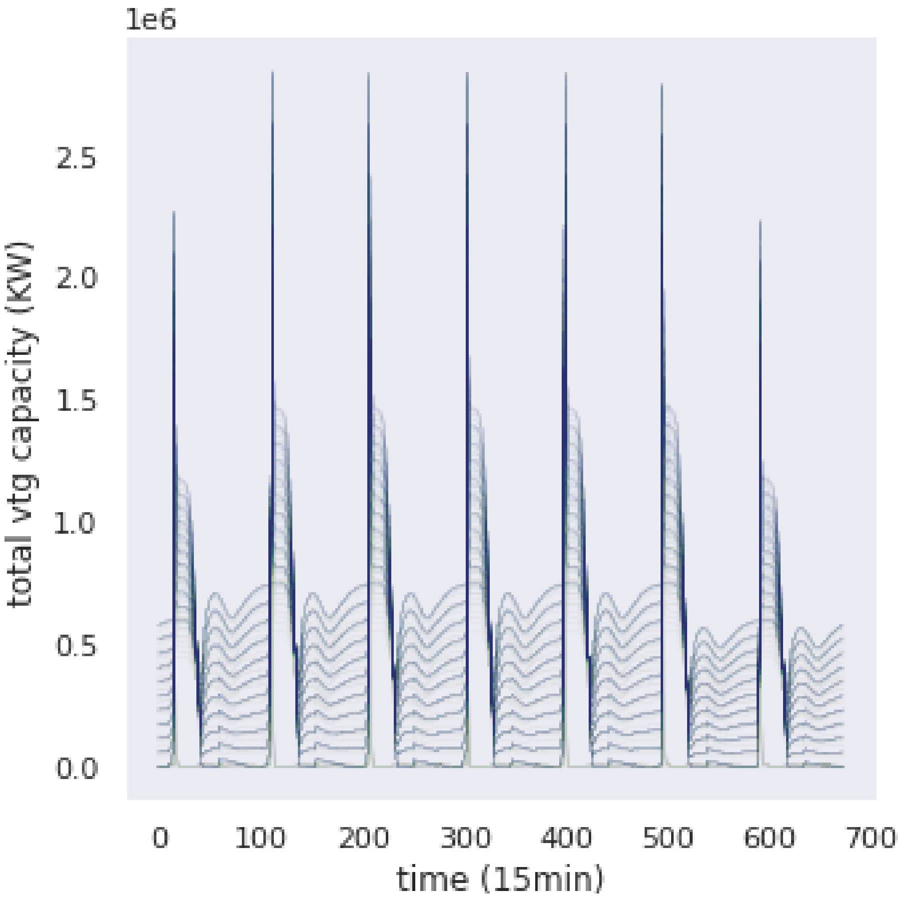

Figure 4.1: One week VTG capacity for 484 profiles

4.2. Integration profiles into etm

As stated, the resulting profiles were merged to one full file, giving the VTG capacity per 15 minute for one year for all possible input combinations. The intention hereby is that this file could be directly loaded into the etm model. However, we acknowledge that assumptions made in this research could be reconsidered in the future by the stakeholder. For this reason, we've constructed the model in such a way that a rerun with new assumptions or a slightly different implementation regarding EV behaviour is easily possible. Furthermore, all values in the model, such as the battery capacities of the EVs, are programmed as model input variables, making it easy for the user to change them.

Furthermore, a couple of observations regarding resulting model behaviour as implemented now regarding the profiles (see 4.1) can be made:

Some input combinations resulted in very large peaks in the total VTG capacity. This could indicate the presence of a large amount of smart charging cars. Since the electricity prices are global in our model, meaning that all EVs acts regarding the same price information, smart EVs would consequently charge largely at the same (cheapest) time steps.

We can also notice much larger peaks in VTG capacity in the early mornings (nights) compared to the peaks later on the day. It is important to note hereby that we made the assumption that smart EVs would charge to 100 percent only and always at home. At work, they would charge only to the needed capacity to arrive back home. This was done because we assumed that a EV parked at work would need a charge sufficient merely for the task to return to home. On the other hand, we assumed that a car at home would likely charge full in order to accommodate work trips and other activities. See section A.2 for a complete overview of made assumptions.

4.3. VTG capacity of a profile

In map 4.2, the VTG capacity per square km per municipality is shown. The map shows that the VTG capacity concentrates in the centre of the country. It is normalized by the municipalities' area to prevent bias. This map shows that the output of our model can easily be used to compute the VTG capacity per municipality in the Netherlands. However, since the plot is based is on a single run, hard conclusions should not be drawn from this map.

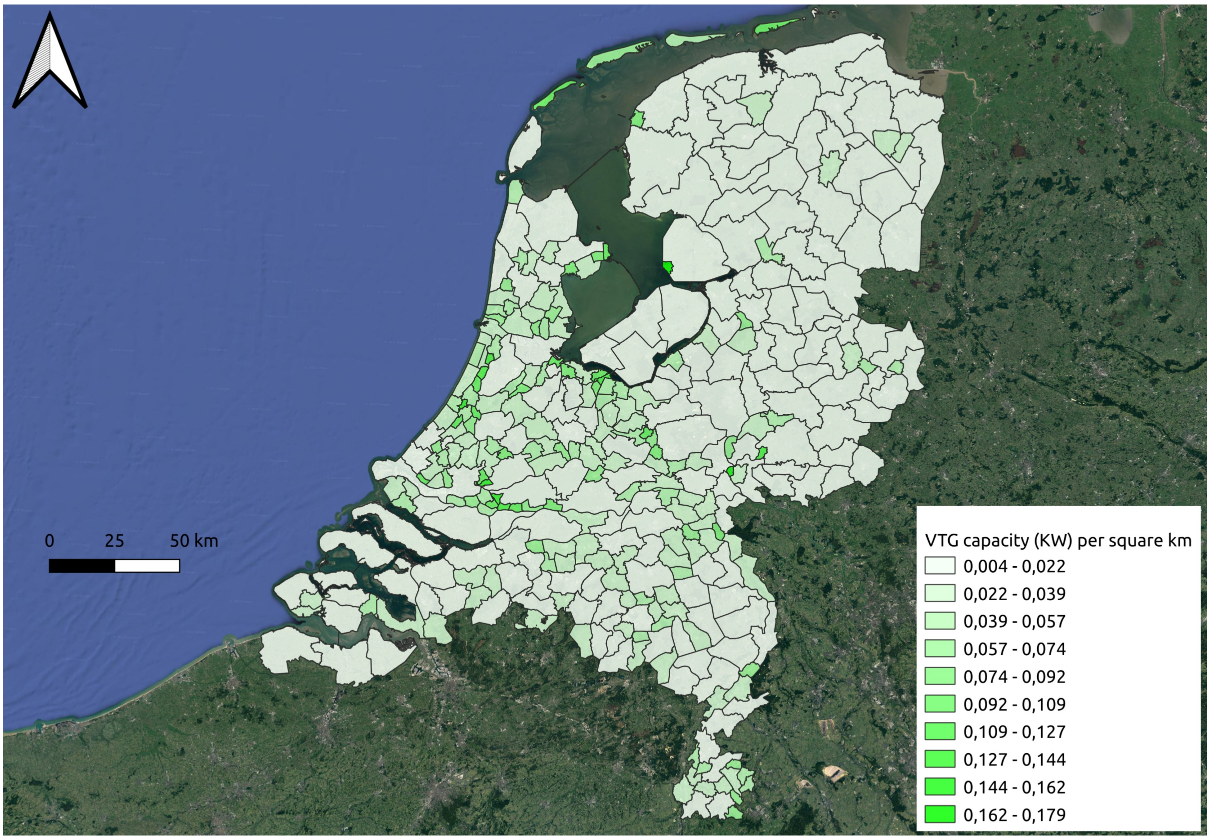

Figure 4.2: VTG capacity per square km per municipality of a profile. (Satellite image from Google)

5 Conclusion & Discussion

This research aimed to investigate the VTG capacity of all plugin electric vehicles in The Netherlands over time, given input variables from the original ETM model. These variables included: the preference to charge at home, the allowed VTG capacity and the percentage of cars that are smart charging enabled. With the use of an agent-based model, including the full set of EVs in the Netherlands, we have determined this total VTG capacity in time steps of 15 minutes for a full year. Full model code is made publicly available [1].

5.1. VTG capacity over time

A large set of different profiles regarding the total VTG capacity of all EVs in The Netherlands were calculated using the agent-based model constructed in this research. As illustrated in 5.1, we found a sequacious VTG capacity pattern over time, with peaks validated against current peaks within the ETM. From the profiles, it can be concluded that the VTG capacity varies between 100 MW and 300 MW, with peaks ranging between 1 GW and 3 GW.

VTG capacity peaks are spread out over the week, with seven large peaks during nighttimes. In these high peaks, two lower ones are visible on Monday night (left) and Sunday night (right), due to lower travel activity during weekend days. Furthermore, lower peaks can be seen during daytime, when people are assumed to charge at work.

Resulting profiles can potentially be used directly within the current ETM model. Furthermore, the current model setup makes it easy for the change of assumed parameters and thereby a recalculation of the profiles in the future. Finally, while the ETM model does not account for spatial diversity, our model does. This can lead to useful insights on spatial distribution of VTG capacity, in which we found indications that some regions have a relatively high VTG capacity.

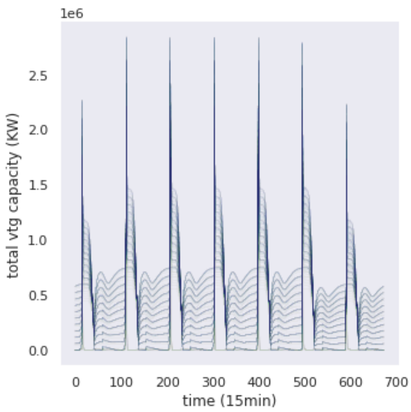

Figure 5.1: One week VTG capacity for 484 profiles. For more information, see 4.1

[1] https://github.com/floristevito/VTG-Dynamics

5.2. Model limitations

It is important to note that the constructed agent-based model in this research is by no means a full representation of reality. Some important choices were made regarding the implementation of the model that directly reflect in the outcomes. We therefore want to highlight the following current limitations in our research, together with recommendations for the extensions of this work.

The gravity model that was used in this model was rather limited, and could be extended. For example, it does not accurately model cities that have a low amount of jobs. Few people have a job in these types of cities, even though lots of people live in these types of cities. On top of that, the current gravity model does not generate internal traffic. Currently, every trip is from a municipality towards another, instead of within a municipality.

Another limiting factor of the model is that currently there exists no feedback between the electricity price and the usage of electricity by EVs. As mentioned before, in 3.2, a feedback mechanism between usage and pricing could influence peak load demands. However, this impact is probably limited only to those peaks, because normal daily EV load patterns are relatively small compared to the complete Dutch energy sector.

An additional limitation, is the way the amount of EVs per municipality is determined. In the current implementation, the amount of EVs per municipality is purely based on the total amount of EVs and the amount of inhabitants. In further research, this could be extended by other empirical variables. For example: local income, car ownership and driving licence ownership.

Trips in the model are all trips from home to work, or in the reversed direction. Activities, such as sports and leisure, are not properly represented. Although the model does change the amount of trips in the weekend, it might be that the VTG capacity changes when the destinations in the weekend change as well.

A final limitation of the model, is that each EV charges to 100% when plugged in. Although we think that this is a reasonable assumption, it might be better to assume that the EVs do not charge towards 100%.

References

Andreas. (2019, August 3). *Free image on pixabay - electric car, automobile*. Retrieved January 13, 2022, from https://pixabay.com/photos/electric-car-automobile-carsharing-4381728/

CBS. (2018, March 15). *Woon-werkafstanden*. Retrieved November 15, 2021, from https://www.cbs.nl/nl-nl/achtergrond/2018/11/woon-werkafstanden-2016

CBS. (2021a). *Inwoners per gemeente*. Retrieved January 26, 2022, from https://www.cbs.nl/nl-nl/visualisaties/dashboard-bevolking/regionaal/inwoners

CBS. (2021b, November 10). *Verkeersprestaties personenauto's, brandstof uitgebreid, leeftijd*. Retrieved January 26, 2022, from https://opendata.cbs.nl/#/CBS/nl/dataset/83703NED/table?ts=1642416879857

EV-Database.nl. (2021a). *Accu capaciteit van electrische auto's*. https://ev-database.nl/cheatsheet/accu-capaciteit-elektrische-auto

EV-Database.nl. (2021b). *Verbruik van elektrische auto's*. Retrieved January 27, 2022, from https://ev-database.nl/cheatsheet/verbruik-elektrische-auto

Foramitti, J. (2021). AgentPy: A package for agent-based modeling in python. *Journal of Open Source Software*, *6*(62), 3065. https://doi.org/10.21105/joss.03065

Jung, W.-S., Wang, F., & Stanley, H. E. (2008). Gravity model in the korean highway - IOPscience. https://doi.org/10.1209/0295-5075/81/48005

Kadaster. (n.d.). *Rijksdriehoeksstelsel*. https://www.kadaster.nl/zakelijk/registraties/basisregistraties/rijksdriehoeksmeting/rijksdriehoeksstelsel

Kwakkel, J. H. (2017). The exploratory modeling workbench: An open source toolkit for exploratory modeling, scenario discovery, and (multi-objective) robust decision making. *Environmental Modelling & Software*, *96*, 239–250. https://doi.org/10.1016/j.envsoft.2017.06.054

Natrajan, A., Jr, P., & Srinivasan, S. (1997). MRE: A flexible approach to multi-resolution modeling., 156–163. https://doi.org/10.1109/PADS.1997.594601

Raffler, C. (2021, February 21). *QNEAT3 - qgis network analysis toolbox*. Retrieved January 26, 2022, from https://github.com/root676/QNEAT3

RVO. (2018, October). *Electric vehicle charging*. https://nederlandelektrisch.nl/u/files/2018-11-ev-charging.pdf

TenneT. (2022a). *Annual peak load and load curve*. Retrieved January 28, 2022, from https://www.tennet.eu/electricity-market/transparency-pages/transparency-germany/network-figures/annual-peak-load-and-load-curve/

TenneT. (2022b). *Systeembalansinformatie*. Retrieved January 28, 2022, from https://www.tennet.org/bedrijfsvoering/Systeemgegevens_uitvoering/Systeembalans_informatie/index.aspx

Van Barlingen, W., & Simpson, J. D. (2021, August 31). *EV charging levels explained*. Retrieved November 17, 2021, from https://blog.evbox.com/ev-charging-levels

WorldData. (2018). *Energy consumption in the netherlands*. Retrieved January 28, 2022, from https://www.worlddata.info/europe/netherlands/energy-consumption.php

A

Appendix

A.1. Example cases reference graphs

This section shows the charging, power demand and VTG capacity figures for a given EV in the model. These graphs show that the behaviour from 2.3.5 occurs as expected in the model.

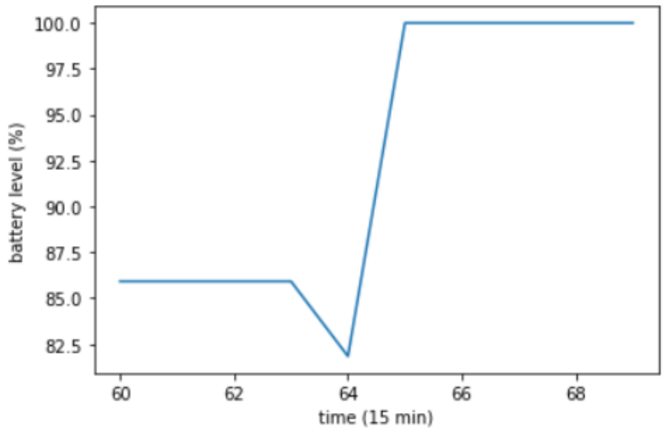

Figure A.1: model EV charging behaviour, as expected from figure 2.3 and 2.6

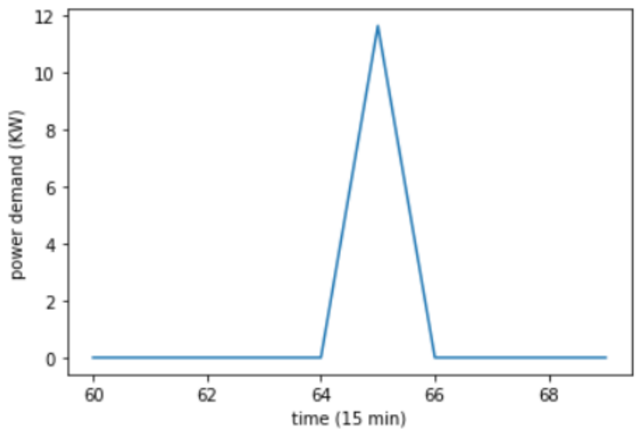

Figure A.2: model EV power demand, as expected from figure 2.4 and 2.7

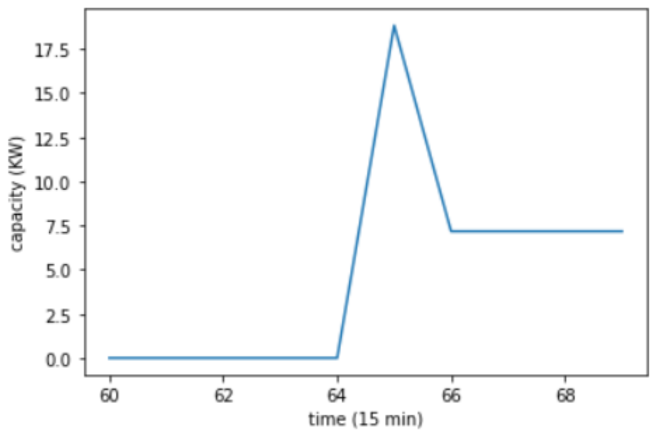

Figure A.3: model EV VTG availability, as expected from figure 2.5 and 2.8

A.2. assumptions

Every model will have to make assumptions about the system at one point. The list below shows an accumulation of assumptions that were made.

1. We start this project without any explicit assumptions
2. The number of electric cars remains constant throughout time
3. All home and work locations will be aggregated to be in a centroid on municipality level
4. Travel distance will be calculated from centroid to centroid via the road network
5. Fuel use per km is the same across all roads and cars
6. All electric cars are created equal
7. Speed on all roads is equal
8. The time tick is 15 minutes, based on the tick of the electricity market
9. Travel times are rounded to the nearest 15-minute interval
10. Overshoot and undershoot in travel time will average out across all cars
11. There is no feedback between battery storage and electricity prices to consumers (capacity is assumed too small compared to system)
12. EV's can only charge when stationary
13. EV's can only charge in a municipality, NOT on the highway for example
14. When at work or at home, EVs always have a charger available.
15. EV-owners always work outside their own municipality. There's no internal traffic.
16. Agents' home and work locations do not change
17. All EV agents travel to work every workday except for the weekend.
18. No EV commutes within district
19. EV's plug in as soon as they reach a destination.
20. EV's always plug in when arriving at a destination.
21. EV's will be initialized with full batteries in the first time step.

22. EV battery usage is only dependent on distance travelled, not on time (disruption).
23. The small variation in model outputs when varying seeds with a static set of input parameters infers a similar small variation in model outputs, should the seed be varied with a different set of model inputs.
24. There is an absolute offset (parameter) in time spent on work and on the time of departure to work. A relative offset in the form of disruption is present on the travel time.
25. The commute distance of EVs are set and do not change throughout weekdays. However, weekends do have less traffic compared to week days, as can be specified by a model parameter.
26. there is a limitless amount of charging locations in each municipality
27. When charging at home, EVs desire to charge to full (100%)
28. When charging at work, EVs desire to charge until the charge in the EV is enough to return home

A.3. Profiles exploration

This section contains more detailed plots of how the ETM profiles look. In each figure, one parameter is varied per graph. It should be noted that for each combination of parameters, a single run was done. This is due to the fact that each run had a rather large amount of EVs (174000). Therefore, the computation time was rather long. In the foreseeable future, when these profiles are finalized, they should be based on more replications. Therefore, no hard conclusion should be drawn from these profiles.

Despite the limited number of replications, figure A.4 shows that the amount of peaks in VTG capacity increases when the percentage of smart EVs is increased. Other than that, figure A.5 shows that the VTG capacity of the EVs increases when the allowed VTG percentage is increased. Finally, A.6 shows that there isn't much difference in the VTG capacity, when the preferred charging location is varied.

The uncertainty margins in all plots (figures A.4, A.5, A.6)

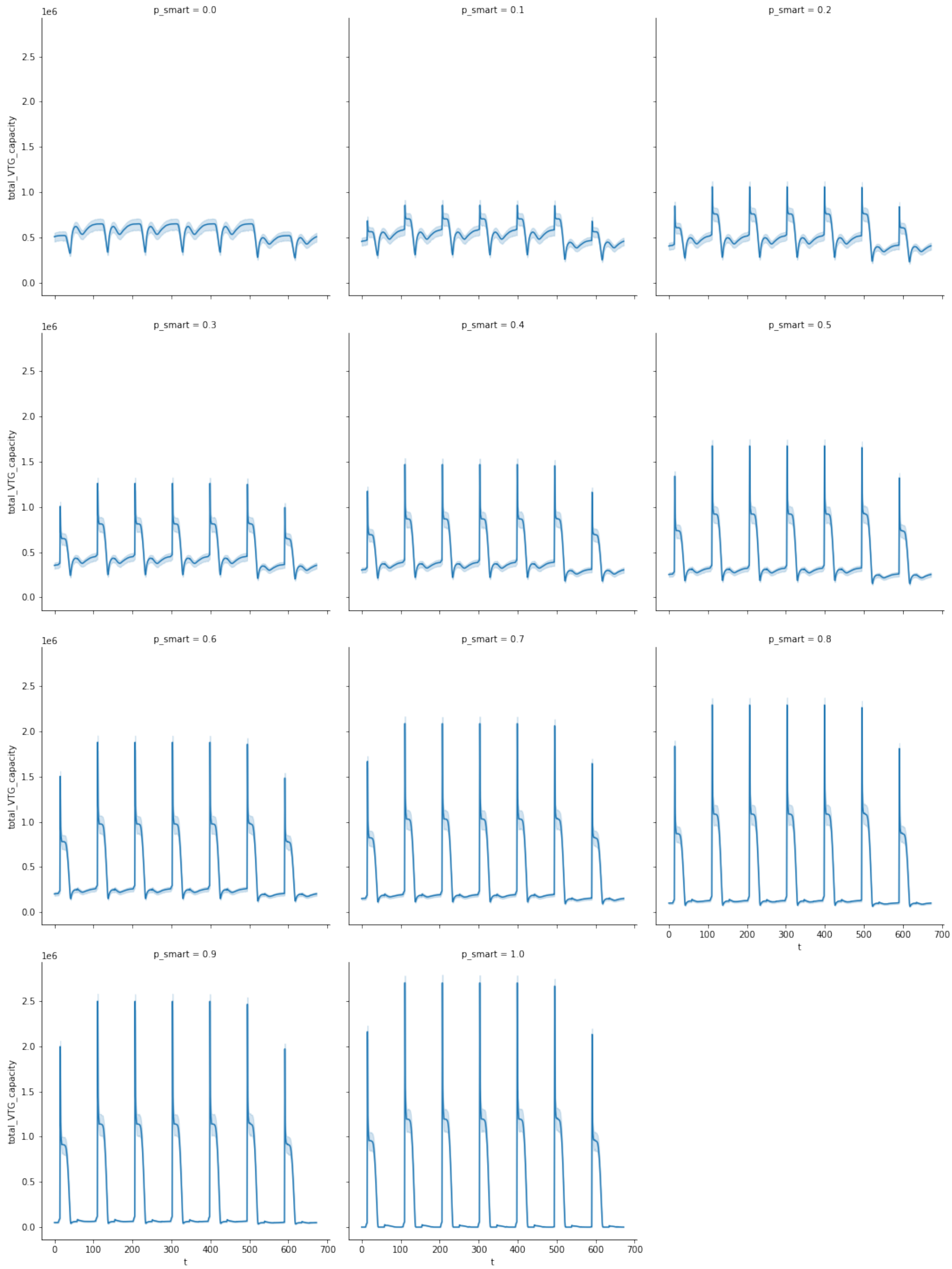

Figure A.4: VTG capacity with a varying percentage of smart charging cars.

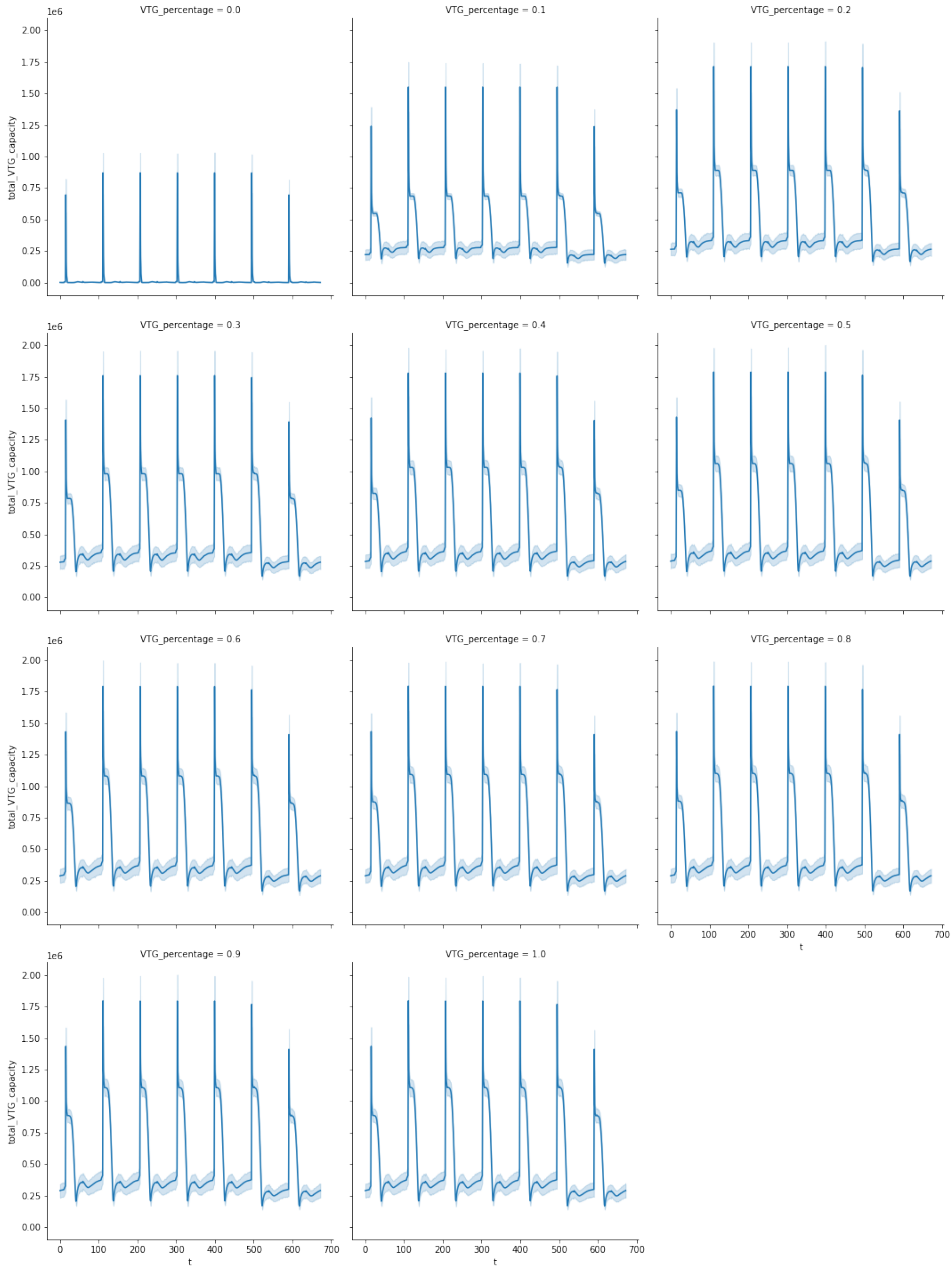

Figure A.5: VTG capacity with a varying amount of VTG percentage.

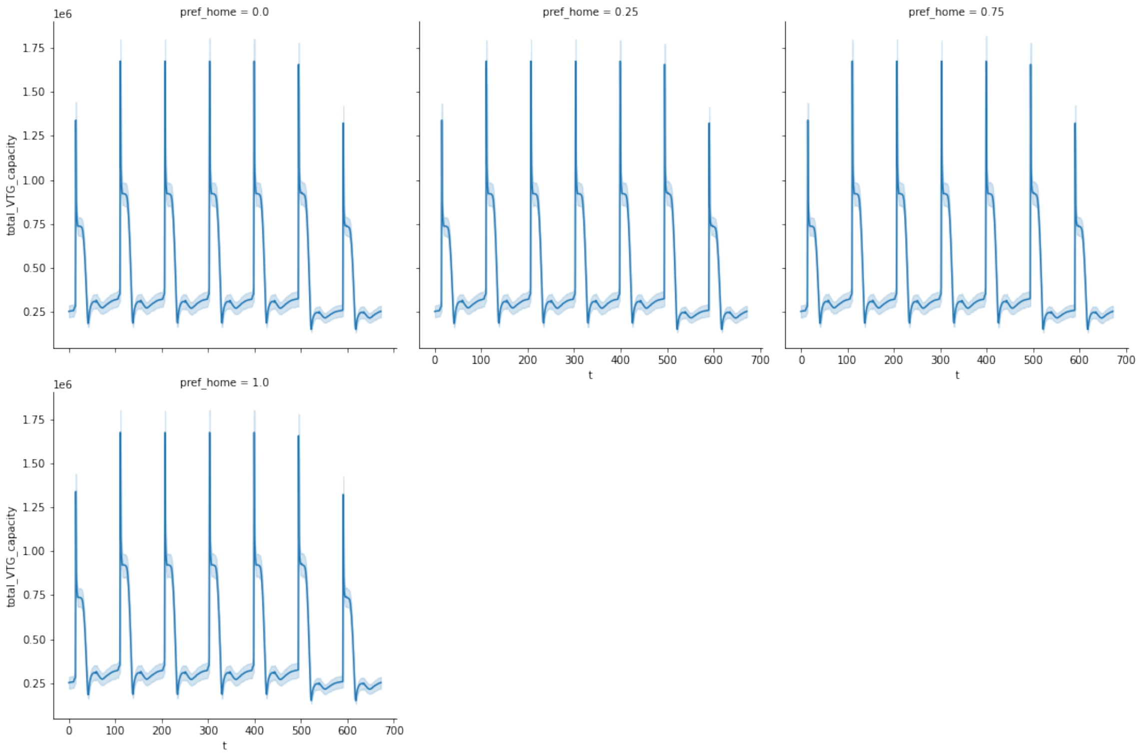

Figure A.6: VTG capacity with a varying preference of charging location