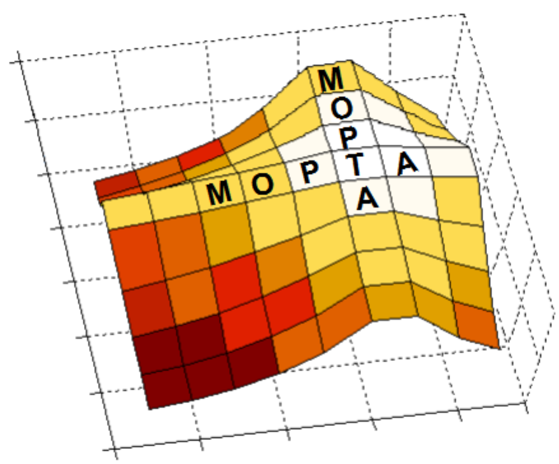
MOPTA Competition 2023

Team ChargED

Optimal placement of electric vehicle charging stations in PA

Author:
Heidi Wolles L.
Kilian Wolff
Scott Jenkins

Advisor:
Dr. Kit Daniel Searle



May 31, 2023

Abstract

Problem, model setup, solution approach, algorithms, sampling, assumptions, results, recommendations.

We first present a continuous maximal covering location problem to obtain x-y-coordinates for optimal facility locations. This mixed integer problem (MIP) has a large number of variables, and incorporating samples would significantly challenge any state-of-the-art solver (AIMMS, Gurobi, CPLEX, Xpress, etc.).

Instead, a heuristic solution is sufficient to approximate good potential charging locations (PCLs). Given the steady state nature of the problem, we consider an location-allocation model expressed as an integer optimization problem with sampling. This is convex.

Contents

update th

Nomenclature

a	Constant building price in pounds	5000
$b_{y,i}$	Binary variable denoting whether at location i is built in year y	dec.variable
$c_{y,i}$	The total charging capacity at location i in year y	dec.variable
$e_{i,t}$	Denotes whether at location i a charger of type t exists	$\in \{0,1\}^{I\times T}$
$f_{i,j}$	Capacity flow from charging location i to demand point j	dec.variable
$h_{i,j}$	the walking time from charging point i to demand point j	
J	Demand matrix at j in year y	
k^{min}	Low capacity estimates of the respective charger type	$(2000, 4000, 30000)$
l_i	Straight distance to the city centre for charging point i	
M	Allowed number of chargers at every charging location	5
N	Allowed chargers in the city centre	100
p	Percentage of demand that needs to be fulfilled at every demand point	0.85
r	Radius around city centre, for which chargers are considered to be in the city	$2.5km$
$s_{y,t}$	Installation costs for charger of type t in year y in pounds	$(8300, 12500, 18750)$
u^{max}	Maximum fraction of type t chargers of the total chargers	$(0.5, 0.5, 0.5)$
u^{min}	Minimum fraction of type t chargers of the total chargers	$\in [0,1]^3$
v_y	Allowed budget in year y in thousands	
w	Allowed walking time in minutes to reach a charging point from a demand point	20
$x_{y,i,t}$	Number of chargers of type t built at location i in year y	dec.variable
$z_{y,i,t}$	Number of chargers at location i of type t in year y	dec.variable
$\mathcal{I}$	Index set of electric vehicle locations	$\{1, 2, \dots, 10\,790\}$
$\mathcal{J}$	Index set of potential charging locations	$\{1, 2, \dots\}$
$\mathcal{J}_A$	Index set of built charging stations, called active	
$\mathcal{K}$	Index set of improved active charging station locations	

1 Some comments

- There are many abbreviations which make things difficult to read. Can we maybe stick to only very standard abbreviations such as EV and write out the rest? Check this at the end.
- Is the nonlinear problem a MILP and the linear problem an ILP? Only the nonlinear has both continuous and integer variables (hence mixed).
- Seems we are swapping between 'scenarios' and 'samples'. Do we want to pick a side?! ▷ How about this? Sample for the 10790. Scenario for the sums $s \in S$. Replication for the output analysis when doing the allocation problem multiple times for confidence interval. I am happy to stick with this convention. Can someone please just make sure of it?
- Do they specify a margin size for the report? ▷ No but 15 pages is including references !!

2 Introduction

Electric vehicles (EVs) are becoming a key incentive to reduce emissions in the transportation industry, and hence contribute to the green transition on a national scale. The Biden administration's efforts to cut emissions and tackle climate change include a shift towards greener transportation options. The Federal Highway Administration Designated Alternative Fuel Corridor is currently the largest U.S. investment into the EV charging sector with a national network of half a million charging stations (CSs) [1]. Encouraging more people to switch to electric vehicles will, in the long run, reduce pollution and minimize the impact of fluctuating fuel prices. Although most EVs are charged at home, having a reliable network of public CSs will diversify the use of EVs and minimize car owners' chances of being out of range of a charger, hence relieving first-time buyers' anxiety over choosing an EV [7, 18].

The Pennsylvania Department of Transportation and the Department of Environmental protection are also partaking in the systemic change through initiatives like the Alternative Fuels Incentive Grant and Driving PA Forward [7, 20]. The latter specifically prioritizes projects which (i) consider strategic locations, (ii) contribute to existing or planned fueling networks, (iii) provide expected usage levels, and (iv) are cost-effective. Along with uncertain driving ranges, user behaviours and a large number of potential locations, these conditions highlight the necessity of a mathematical framework for EV CS system design and operations.

The problem proposed for the 15th AIMMS-MOPTA Optimization Modeling Competition is to identify CS locations in Pennsylvania and determine the number of CSs at each location to minimize total cost. The EV charging demand is stochastic in nature, and is given as 1 079 locations, called demand points, each with 10 EVs. These demand points are stationary and are not assumed to have additional descriptive data regarding typical length of journeys nor to have a time-related charging pattern. The challenge is divided into two phases; firstly, the location of 600 CSs must be determined to obtain the minimum cost charging infrastructure. Secondly, the *number* of CSs and corresponding locations which minimise the cost of establishing the charging infrastructure are to be determined.

The remainder of this report is structured as follows: Relevant models for EV charging infrastructure and mathematical frameworks to tackle such facility location problems are briefly reviewed in §3. The optimization modelling approach adopted by Team ChargED together with important modelling assumptions is presented in §4. Thereafter, the tailored solution approach for solving the aforementioned model is discussed in §5 which, in §6, is followed by an ensuing analysis of the performance measures obtained from the solution. A sensitivity analysis of model results are discussed in §7, illustrating how certain model parameters impact solution performance. Lastly, we tie this back into our model assumptions and discuss possible further extensions to the model in §8.

3 Literature review

The design of EV CS infrastructure falls into an increasingly important category of research into the green transition and sustainable operations research. Many optimization models with similar motivations in terms of design and optimization of EV charging networks have been presented in the literature. Typical approaches range from standard facility location models; p-median problems and maximal covering problems [9]; to simulation-optimization models [13, 23], including agent-based modelling [24], and data-driven multi-criteria methods [10]. Three main considerations are the choice of objective (and distance measure), the standard trade-off between a discrete vs. continuous formulation, and the more difficult case of time-/space-varying, potentially stochastic, demand [17].

Public EV service region design (districting) is closely related to optimization of EV sharing systems and battery swapping. He et al. [11, 12] incorporated user decisions based on EV energy levels and formulated a nonlinear program for a queuing-location model for sharing systems which can be approximated as a mixed-integer second order cone program. Mak et al. [19] proposed a robust optimization model for the case of EV CS infrastructure accommodating EV battery swapping. The transportational aspect of planning EV CSs also highlights a close connection to routing problems which are becoming increasingly popular as a primary part of the optimization model [14] or as a hybrid [15]. Stakeholders are often incentivized to construct CS ensembles within existing infrastructure, e.g. along major national highways with programs like the Alternative Fuel Corridors [1].

We refer the interested reader to [2, 6, 8, 17] for the more general case of continuous facility location problems, and [21, 22] for facility location problems under uncertainty. Sections §4-5 further elaborate on the relevancy of the modelling approach adopted by TeamED with regards to the previously established literature.

4 Optimization model

The problem of determining the coordinates of facilities in the two-dimensional Euclidean plane which minimize the distance (or some other cost-metric) from another set of known locations (or demand points) is widely known in the literature as the Weber problem [3, 4, 5]. In this section we formulate a suitable continuous location-allocation model in an attempt to solve the problem described in §2. This takes the form of a generalised Weber problem which is non-convex and notoriously difficult to solve. Therefore, in §5 we propose a suitable discretization procedure to find high quality solutions in a suitable time at the sacrifice of a proof of optimality.

Let $\mathcal{I}$ denote the set of EVs and let $\mathcal{J}$ denote the set of potential charging locations (PCLs) for CSs in the 290×150 mile area in Pennsylvania. For now we assume that $\mathcal{J}$ is sufficiently large. Given the stochastic nature of the problem, we employ a robust optimisation framework and consider a finite set S of scenarios, such that $s \in \mathcal{S}$ represents one scenario.

4.1 Decision variables

Let the tuple of decision variables $\boldsymbol{x} = (x, y) \in \mathbb{R}^{2\times\mathcal{J}}$ denote the geographical position of CSs, such that $\boldsymbol{x}_j$ is the spatial coordinates of a CS $j \in \mathcal{J}$. Hence let auxiliary variable $\tilde{d}_{i,j} = ||\boldsymbol{x}_j - \boldsymbol{a}_i||$ denote the distance from a EV $i \in \mathcal{I}$ to a PCL $j \in \mathcal{J}$, where the parameter $\boldsymbol{a}_i \in \mathbb{R}^2$ is a tuple of spatial coordinates of EV $i \in \mathcal{I}$, and $||\cdot||$ is the standard 2-norm[1] in $\mathbb{R}^2$ (Euclidean distance).

Moreover, EVs are allocated to CSs through decision variables, $u \in \{0,1\}^{\mathcal{I}\times\mathcal{J}\times\mathcal{S}}$, such that

$$u^s_{i,j} = \begin{cases} 1 & \text{if EV } i \in \mathcal{I} \text{ is assigned to CS } j \in \mathcal{J} \text{ in } s \in \mathcal{S}, \\ 0 & \text{otherwise.} \end{cases}$$

Allocation in this case corresponds to the assumption that each EV $i \in \mathcal{I}$ charges at its assigned station, *i.e.* that there is some "authority" assigning EVs to CSs. Hence EV $i \in \mathcal{I}$ can be "assigned" to station $j \in \mathcal{J}$, which will be in the setup with overall lowest cost for all stakeholders.

Since we will not build CSs at every PCL, we introduce the decision variables $v \in \{0,1\}^{\mathcal{J}}$ such that

$$v_j = \begin{cases} 1 & \text{if a CS is built at PCL } j \in \mathcal{J}, \\ 0 & \text{otherwise.} \end{cases}$$

Finally, the integer decision variables $w \in \mathbb{Z}^{\mathcal{J}}_+$ to model the number of chargers to install at each location, such that w_j is the number of chargers to install at CS $j \in \mathcal{J}$.

4.2 Constraints

The number of chargers that can be built at each CS is bounded from above by some $m \in \mathbb{Z}_+$. It is given in the problem description that $m = 8$. Therefore, if a station is built, it must contain between 1 and 8 chargers, which

[1] The 2-norm in in $\mathbb{R}^2$ is $\tilde{d}_{i,j} = \sqrt{(x_j - a_i)^2 + (y_j - b_i)^2}$, where $\mathbf{a}_i = (a_i, b_i)$.

we enforce with the constraints:

$$v_j \leq w_j \leq m v_j \quad \forall j \in \mathcal{J}. \tag{1}$$

Also from the problem description, it is given that at most one EV can queue for each charger, so that the number of EVs which are allocated to each station in each scenario may not exceed two times the number of chargers at that station. We enforce this with the constraints,

$$\sum_{i \in \mathcal{I}} u_{i,j}^s \leq 2 w_j \quad \forall j \in \mathcal{J},\ s \in \mathcal{S}. \tag{2}$$

However, we would rarely expect all EV owners to visit the same allocated CS at the same time. This constraint assumes a worst-case scenario where all assigned EVs need to use the CS at any given time. Realistically, more EVs can be assigned to a CS than we expect to see in queue for that CS simultaneously. Note that this constraint set also ensures that for any scenario $s \in \mathcal{S}$ an EVs cannot be assigned to stations which are not built, i.e. if there exists an $j^* \in \mathcal{J}$ such that $v_{j^*} = w_{j^*} = 0$ then $u_{i,j^*}^s = 0\ \forall i \in \mathcal{I}, s \in \mathcal{S}$. Similarly, each EV can be assigned to at most one station in any scenario by the constraints,

$$\sum_{j \in \mathcal{J}} u_{i,j}^s \leq 1 \quad \forall i \in \mathcal{I},\ s \in \mathcal{S}. \tag{3}$$

We introduce the notion of service level from queuing theory to signify a level of charging demand which must be fulfilled. This allows stakeholders the option to choose how important it is that all EVs which need to charge are granted service, and e.g. only plan for the chargers which fulfill 50% of the demand. The concept is implemented by setting a relative lower bound, $\alpha \in [0,1] \subseteq \mathbb{R}$, on the total number of allocated EVs,

$$\sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} u_{i,j}^s \geq \alpha L \quad \forall s \in \mathcal{S}, \tag{4}$$

where $L \in \mathbb{R}$ denotes the total number of EVs which are able to reach any station. By default, we choose $\alpha = 0.95$. Note that this constraint heavily affects the "feasibility" of the model.

Let r_i^s denote the range of EV $i \in \mathcal{I}$ in scenario $s \in \mathcal{S}$. If, in any scenario $s \in \mathcal{S}$, EV $i \in \mathcal{I}$ is out of reach of a PCL, $j \in \mathcal{J}$, that is, there exist $i \in \mathcal{I}, j \in \mathcal{J}, s \in \mathcal{S}$ such that $d_{i,j} > r_i^s$, then EV $i \in \mathcal{I}$ cannot be allocated to that CS $j \in \mathcal{J}$ in that scenario $s \in \mathcal{S}$. This is enforced with the constraint,

$$\tilde{d}_{i,j} u_{i,j}^s \leq r_i^s \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, s \in \mathcal{S}. \tag{5}$$

Finally, we have the domain constraint on the CS coordinates,

$$\boldsymbol{x}_j \in [0, 290] \times [0, 150] \subseteq \mathbb{R}^2 \quad \forall j \in \mathcal{J} \tag{6}$$

to ensure that each CS is only built in the region under consideration.

4.3 Objective function

Costs associated with the problem description are given by $c_b, c_m, c_d, c_{\tilde{c}} \in \mathbb{R}$, where c_b is the annualized construction investment of a CS (build cost \$/station); c_m is the maintenance cost of a charger (\$/charger); $c_{\tilde{c}}$ is the cost of charging (\$/mile) which is constant up to the full range of 250 miles; c_d is the cost of driving to the station (\$/mile). In particular, we are given in the problem description that $c_b = 5000, c_m = 500, c_d = 0.041, c_{\tilde{c}} = 0.0388$.

The objective function we want to minimize is the overall annualized cost of setting up CSs in Pennsylvania,

$$\begin{aligned} f_{\text{NL}} = \frac{365}{|\mathcal{S}|} \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} \sum_{s \in \mathcal{S}} &\left[\underbrace{c_d \tilde{d}_{i,j} u_{i,j}^s}_{1.} + \underbrace{c_{\tilde{c}}(250 + \tilde{d}_{i,j} - r_i^s) u_{i,j}^s}_{2.} + \underbrace{c_{\tilde{c}}(250 - r_i^s)(1 - u_{i,j}^s)}_{3.} \right] \\ &+ \sum_{j \in \mathcal{J}} (\underbrace{c_b v_j}_{4.} + \underbrace{c_m w_j}_{5.}) \end{aligned} \tag{7}$$

$$= \sum_{j \in \mathcal{J}} \left[c_b v_j + c_m w_j + \frac{365}{|\mathcal{S}|}(c_{\tilde{c}} + c_d) \sum_{i \in \mathcal{I}} \sum_{s \in \mathcal{S}} \tilde{d}_{i,j} u_{i,j}^s \right] + \frac{365}{|\mathcal{S}|} |\mathcal{J}| c_{\tilde{c}} \sum_{i \in \mathcal{I}} \sum_{s \in \mathcal{S}} (250 - r_i^s), \tag{8}$$

where $|\mathcal{S}|$ is the number of scenarios. The first term in (7) models the driving costs of EVs to their assigned CS. The second term is the cost of charging to full capacity, that is, the range subtracted from the total of 250 miles and the driving distance. However, cars which are not assigned to a CS still need to charge to full capacity by some other means. Hence the third term adds the cost of charging to full capacity whenever an EV is not assigned, i.e. if there exist $i^* \in \mathcal{I}, j^* \in \mathcal{J}$ such that $u^s_{i^*,j^*} = 0$ in any scenario $s \in \mathcal{S}$. In this case, we assume the EV owner cannot make use of the CSs in our model and will need to use private chargers or have the EV towed at an extra cost.

The fourth term in (7) accounts for the construction cost of each station, and the fifth for the maintenance cost of each charger at a station. Since the charger maintenance cost is lower than the station build cost, these terms incentivize utilising existing stations over opening new ones. These costs incur on an annual basis, while driving and charging costs incur discretely for each scenario. Hence the first three terms (which sum over the scenarios) are annualized by multiplying by 365 days and dividing by the number of scenarios.

The terms have been simplified in (8), where the constant does not impact the optimization but is included for an accurate total cost. The objective function (8) is nonlinear and nonconvex. Whilst the continuous-allocation problem can be solved to optimality in a reasonable time for a small number of vehicles, the problem for 10790 vehicles poses a significant challenge to any state-of-the-art solver (AIMMS, Gurobi, Xpress, etc.). In section 5, we discuss our heuristic solution approach.

4.4 Data and simulation

Given the file `MOPTA2023_car_locations.csv`, we initially duplicate each row entry 10 times to obtain a list of 10 790 locations of all EVs in `full_car_locations.csv`, as set out in the problem description. In the user interface for our model, we allow the user to upload their own dataset of vehicle locations. This data representation is generally applicable, and does not necessitate that 10 EVs are at each EV location.

Suppose that the range r^s_i of EV $i \in \mathcal{I}$ in scenario $s \in \mathcal{S}$ is normally distributed such that r^s_i is a realization of $R \sim \mathrm{N}(100, 50^2)$, truncated with bounds between 20 and 250 miles. Then the probability that an EV $i \in \mathcal{I}$ requires charging in scenario $s \in \mathcal{S}$ is determined by the map, $p^s_i : [20, 250] \subseteq \mathbb{R} \mapsto [0, 1] \subseteq \mathbb{R}$, defined by $p^s_i(r^s_i) = \exp(-\lambda^2(r^s_i - 20)^2)$, where λ is a scaling parameter. In this case, we let $\lambda = 0.012$.

A Monte Carlo type simulation is employed to determine which EVs $i \in \mathcal{I}$ require charging at a CS $j \in \mathcal{J}$ for each scenario $s \in \mathcal{S}$. To this end, for every scenario $s \in \mathcal{S}$, an EV $i \in \mathcal{I}$ requires charging, modelled by the Boolean parameter $\delta^s_i \in \{\mathit{True}, \mathit{False}\}$, if the probability of the EV needing to charge, p^s_i, is greater than or equal to a random uniform number, $\psi^s_i \sim \mathrm{U}(0, 1)$. The method is constructed as follows: Firstly, generate $\psi^s_i \sim \mathrm{U}(0, 1)$ for each $i \in \mathcal{I}$ and $s \in \mathcal{S}$. If $\psi^s_i \leq p^s_i$, then set the Boolean value δ^s_i to *True*, otherwise set it to *False*. Hence, each scenario $s \in \mathcal{S}$ constitutes 10 790 EV ranges and Boolean values describing whether EVs needs to charge or not. An example output for this simulation procedure is shown in Table 1. This modelling approach allows us to reduce the size of the problem, by removing the decision variables and constraints corresponding EVs which do not need to charge in a scenario $s \in \mathcal{S}$.

Table 1: Table of generated values in one outcome $s \in \mathcal{S}$ of a Monte Carlo simulation.

EV i	Range r^s_i	Probability of charge p^s_i	Uniform random number ψ^s_i	Charging required δ^s_i
1	98	0.42	0.50	*False*
2	73	0.67	0.13	*True*
⋮	⋮	⋮	⋮	⋮
10 790	101	0.39	0.48	*False*

With 10 790 samples and multiple replications, the mean probability that an EV is required to charge is observed to be approximately 0.42. Hence the expected proportion of EVs which are required to charge in each replication is 0.42. We argue that this is the weighted sum of the density function of the truncated normal distribution, denoted $\mathbb{P}(r)$, and the given probability-of-charge function, $p(\delta; r) = \exp(-\lambda^2(r - 20)^2)$, which is not normalized and hence not a density function. In the large-system limit (i.e. after a large number of replications), the sum becomes a standard Riemann integral over the continuous domain of possible ranges,

$$\mathbb{E}[p(\delta;r)] = \lim_{i,s\to\infty} \sum_{i\in\mathcal{I}} \sum_{s\in\mathcal{S}} p(\delta; R = r_i^s)\mathbb{P}(R = r_i^s) \tag{9}$$

$$= \int_{20}^{250} p(\delta;r)\mathbb{P}(r)dr \tag{10}$$

$$= \int_{20}^{250} e^{-(0.012)^2(r-20)^2} \frac{e^{-\frac{1}{2}\left(\frac{r-100}{50}\right)}}{50\sqrt{2\pi}(\Phi(\frac{250-100}{50}) - (\Phi(\frac{20-100}{50}))} dr \tag{11}$$

$$\approx 0.42, \tag{12}$$

where Φ is the cumulative distribution function of the standard normal distribution.

5 Solution approach

The nonlinear continuous model (1)–(8) proved difficult to solve for the full vehicle set in reasonable time, so we propose a heuristic approach, accepting that we can not guarantee a global optimum is reached.

The model can be transformed into a two-stage location-allocation problem similar to [4]. In this approach we initialize $\mathcal{J}$ with a discrete set of candidate charging locations such that auxiliary variable $\tilde{d}_{i,j}$ becomes a parameter which can be replaced by parameters $d_{i,j}$ for all $i \in \mathcal{I}, j \in \mathcal{J}$. The variable $\boldsymbol{x}$ in the nonlinear model becomes a property of set $\mathcal{J}$ Clearly, in this case, the objective value is linear and the problem becomes a mixed-integer programming problem.(MILP) model which can be solved with commercial software.

We can iteratively improve the global objective function by including additional PCL which we know will result in an improved solution. The pseudo code for this procedure is outlined in Algorithm 1.

Algorithm 1 Location-allocation algorithm

Input: (P) : mixed integer linear program (MILP), n : number of initial locations, '*type*' : random, k-means or k-means constrained, T : time limit in seconds, ϵ : objective value improvement tolerance, $d_{\min}$: smallest distance allowed between CSs, κ : radius from CS

Output: $f, (u, v, w)$: solution to P

$\mathcal{J} \leftarrow$ INITIAL_LOCATIONS(n, '*type*')
$(u, v, w)_{\mathrm{WS}} \leftarrow \emptyset$
Improvement $\leftarrow$ *True*
while Improvement = *True* **do**
    f, $(u, v, w) \leftarrow$ SOLVE_MILP($P, T, \epsilon, (u, v, w)_{\mathrm{WS}}$)
    $\mathcal{K} \leftarrow$ FIND_IMPROVED_LOCATIONS(u, v)
    $\mathcal{K}, \mathcal{J}_A, \{\mathcal{I}_{A_j}\}_{j\in\mathcal{J}_A} \leftarrow$ FILTER_LOCATIONS($\mathcal{J}, \mathcal{K}, d_{\min}, \kappa$)
    if $|\mathcal{K}| = 0$ **then**
        Improvement $\leftarrow$ *False*
    else
        $\mathcal{J} \leftarrow \mathcal{J} \cup \mathcal{K}$
    end if
    Construct distances i to k
    f_{WS}, $(u, v, w)_{\mathrm{WS}} \leftarrow$ CONSTRUCT_WARMSTART($\mathcal{J}_A, \{\mathcal{I}_{A_j}\}_{j\in\mathcal{J}_A}, \mathcal{K}, f, (u, v, w)$)
    if $|f_{\mathrm{WS}} - f| \leq \epsilon$ **then**
        Improvement $\leftarrow$ *False*
    end if
end while

The algorithm takes as input the linearization of the location-allocation problem, (P); a positive integer, n, which corresponds to the initial size of $\mathcal{J}$; real numbers, T, and ϵ, which are the maximum allowable time and minimum objective function improvement between successive solutions in the branch-and-cut tree of the MILP solver; a real number, $d_{\min}$, denoting a soft threshold distance between CSs (greater than which new PCLs are always added, and otherwise with a certain probability); and finally a real number κ which is a reasonable radius from a CS to other CSs and EVs (within which alternative allocations are considered). The algorithm returns a solution $(f,\ (u, v, w))$ which is an upper bound on the problem (1)–(8).

The algorithm begins by initiating the set of PCLs $\mathcal{J}$ by calling the function INITIAL_LOCATIONS, described in detail in §5.1, and initiating the Boolean variable Improvement as *True*. The following process is then repeated until Boolean variable Improvement becomes *False*. The function SOLVE_MILP is called, in which the linearized location-allocation problem (P) is solved, described in detail in §5.2. It returns a solution, in the form of a set of locations which are used as CSs, the allocation of EVs to CSs, the number of chargers to install at each CS, and the objective function value. The function FIND_IMPROVED_LOCATIONS takes as input the locations where CSs are built ~~in any scenario~~ and the corresponding allocations of EVs to CSs for every scenario, and returns a set of ~~locations~~ additional PCLs, denoted by $\mathcal{K}$, which ~~result in an improved objective function~~ minimises the total distance between each CS and the EVs which are allocated to it across all scenarios. This procedure is described in detail in §5.3. The algorithm then calls a helper function called FILTER_LOCATIONS to remove some of the improved locations. This function is employed in an attempt to reduce the size of $\mathcal{J}$ by only adding new locations to $\mathcal{J}$ if the ratio of PCLs to EVs does not exceed a random uniform number., also described in more detail in §5.3. If there are no improved locations then the Boolean variable Improvement is switched *False* and the algorithm terminates, otherwise the set of PCLs $\mathcal{J}$ is updated to also include the filtered subset of $\mathcal{K}$. The function CONSTRUCT_WARMSTART is then called to build a warm start to the MILP (P), described in detail in §5.4, and if the objective function value corresponding to the warm start is not sufficient, then the Boolean variable Improvement is switched to *False* and the algorithm terminates; otherwise the iteration is over, and we repeat the process. Algorithm 1 is illustrated schematically by means of a flow chart in Figure 1.

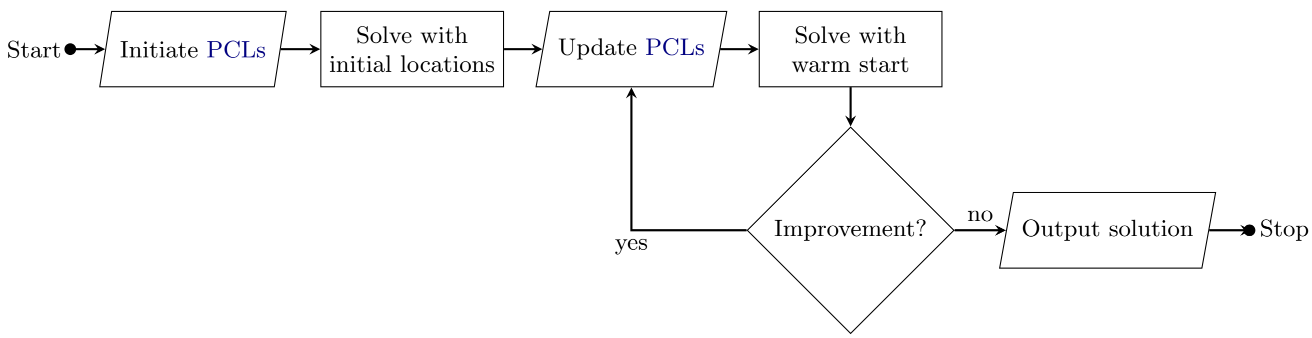
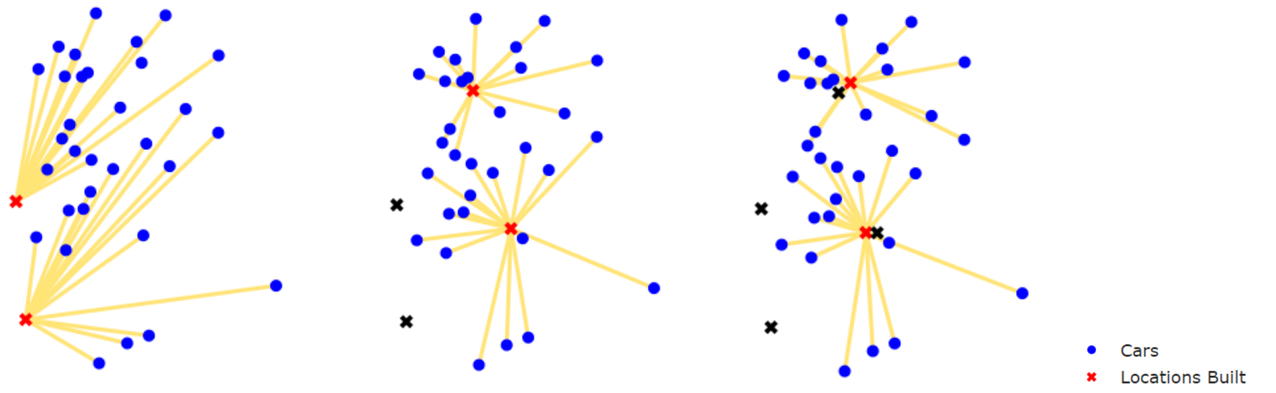


Figure 1: Flow chart of the two-stage solution to the location-allocation problem.

Next, we demonstrate the mechanics of Algorithm 1 by means of a simple example. In this example we have $|\mathcal{I}| = 32$ EVs and choose $n = 2$ initial PCLs chosen randomly in the [... × ...] plane. A graphical representation of the solution returned from the function SOLVE_MILP is provided in Figure 2(a). Thereafter the functions FIND_IMPROVED_LOCATIONS and FILTER_LOCATIONS are called and two improved locations are found an a warm start for (P) is constructed and, another iteration of the algorithm is ~~completed, and another~~ initiated. The graphical representation of the solution returned from the function SOLVE_MILP is provided in Figure 2(b). In this figure, we can see two new PCLs which where added to $\mathcal{J}$ ~~by the functions FIND_IMPROVED_LOCATIONS and FILTER_LOCATIONS~~, and consequently all of the EVs where allocated to these CSs. Thereafter, another iteration of the algorithm is completed, and graphical representation of the solution returned from the function SOLVE_MILP is provided in Figure 2(c). In this case another two PCLs where added to $\mathcal{J}$, however, this time only one of the new CSs is allocated to EVs. At this point, there are no more improved locations and therefore the algorithm is terminated.

5.1 Initial locations

In this section we provide more details on the first function in Algorithm 1, INITIAL_LOCATIONS This function takes as input a positive integer $\tilde{k}$ which is the number of initial PCLs and a string '*type*' which specifies which procedure must be employed to determine the initial PCLs. By choosing a set of good initial PCLs, we can potentially reduce the number of iterations required for Algorithm 1 to converge to a good solution. By default, we provide



(a) Initial feasible solution (b) Location improvement (c) New MILP solution

Figure 2: Two-stage location-allocation solution approach.

three options for the user to choose from (i) PCLs generated at random, or obtained with (ii) standard k-means clustering or (iii) k-means constrained clustering, as outlined in Algorithm 2. The first option generally prompts an exploratory search with the aim of avoiding premature convergence to local minima. Locations are generated from from a smaller subset the spatial domain, with corner points defined by the extremal points of the given EVs coordinates, $\boldsymbol{x}$, that is, the rectangle $[x_0, x_N] \times [y_0, y_M] \subseteq [0, 290] \times [0, 150] \subseteq \mathbb{R}^2$, with

$$x_0 = \min_{i \in \mathcal{I}} \{x_i\}, \quad x_N = \max_{i \in \mathcal{I}} \{x_i\},$$
$$y_0 = \min_{i \in \mathcal{I}} \{y_i\}, \quad y_M = \max_{i \in \mathcal{I}} \{y_i\}.$$

The second method for obtaining initial PCLs is k-means clustering, which partitions the set of EVs into $\tilde{k}$ clusters such that each PCL is placed at the geographical centre of each cluster. In constrained clustering, an additional constraint is set on the minimal/maximal number of vehicles assigned to each cluster.

Algorithm 2 Initial locations

Input: $\tilde{k}$: number of initial locations, '*type*' : random, k-means or k-means constrained
Output: $\mathcal{J}$: initial locations

procedure INITIAL_LOCATIONS($\tilde{k}$, '*type*')
  if '*type*' = random **then**
    $\boldsymbol{\pi}_1, \ldots, \boldsymbol{\pi}_n \leftarrow \boldsymbol{\Pi} \sim$ (U(0, 1), U(0, 1)) ▷ Generate 2-dim random uniform tuples
    $\mathcal{J} \leftarrow \{j = 1, \ldots, \tilde{k} : \boldsymbol{x}_j = \boldsymbol{\pi}_j(x_N - x_0, y_M - y_0) + (x_0, y_0)\}$
  else if '*type*' = k-means **then**
    $\mathcal{J} \leftarrow \{j = 1, \ldots, \tilde{k} : \boldsymbol{x}_j$ are centre points of k-means clusters$\}$
  else
    $\mathcal{J} \leftarrow \{j = 1, \ldots, \tilde{k} : \boldsymbol{x}_j$ are centre points of k-means constrained clusters$\}$
  end if
end procedure

5.2 Solve MILP

In this section we explain the function SOLVE_MILP. This function takes as input the linearization of the location-allocation problem, (P), real numbers, T, and ϵ, which are the maximum allowable time and minimum objective function improvement between successive solutions in the branch-and-cut tree of the MILP solver. We begin by explaining the linearization of (P). For each scenario, if an EV needs to charge, we compute the Euclidean distance from its location to each of the PCLs in the set $\mathcal{J}$, denoted by $d_{i,j}$, from EV each EV $i \in \mathcal{I}$ to each CS $j \in \mathcal{J}$. At this point, we note that many of the allocation decision variables are actually not required to solve the problem.

Suppose there exist $i^* \in \mathcal{I}, j^* \in \mathcal{J}, s^* \in \mathcal{S}$ such that $d_{i^*,j^*} > r_{i^*}^{s^*}$, i.e. the EV $i^* \in \mathcal{I}$ cannot reach CS $j^* \in \mathcal{J}$ in that scenario $s^* \in \mathcal{S}$, then we can exclude variable $u_{i^*,j^*}^{s^*}$ from the model. Moreover, if EV $i' \in \mathcal{I}$, in scenario $s' \in \mathcal{S}$, does not require charging, i.e. $\delta_{i'}^{s'} = \mathit{False}$, then the allocation decision variables $u_{i',j}^{s'}$ can be excluded for all $j \in \mathcal{J}$. For the sake of convenience, we introduce the set of all EVs $i \in \mathcal{I}, whicharesettochargeinscenarios \in \mathcal{S}$, that is, $\mathcal{I}_s = \{i \in \mathcal{I} \ : \ \delta_i^s = True, s \in \mathcal{S}\}$. In other words, the allocation variables are only defined as the following set $\{u_{i,j}^s \in \{0,1\} \ \forall \ s \in \mathcal{S}, i \in \mathcal{I}_s, \ j \in \mathcal{J} \ : \ d_{i,j} \leq r_i^s\}$. In this case, the nonlinear objective function (8) is simplified to the linear objective function

$$f_{\mathrm{L}} = \sum_{j \in \mathcal{J}} \left[c_b v_j + c_m w_j + \frac{365}{|S|}(c_{\tilde{c}} + c_d) \sum_{s \in \mathcal{S}} \sum_{i \in \mathcal{I}_s : d_{i,j} < r_i^s} d_{i,j} u_{i,j}^s \right] + C, \tag{13}$$

where C is the constant from (8) and the constraint set (5) is implicitly enforced. This then results in the MILP ~~for all $j \in \mathcal{J}, s \in \mathcal{S}$ in the constraints when passing the model to the CPLEX solver. As we are minimizing, it will be set to zero, and as argued in §4.4, this allows the exclusion of 42% of all EVs in the model.~~

~~By pre-computing the distance to any PCL $j \in \mathcal{J}$, we can exclude the spatial coordinates as variables and transform the model into a linear one as follows, where C is the constant from (8). In summary, we have modelled the problem posed for the 15th AIMMS-MOPTA competition as the integer linear programming (ILP) problem,~~

$$\left.\begin{array}{lll}
\min & f_{\mathrm{L}} & \\
\text{subject to} & v_j - w_j \leq 0 & \forall j \in \mathcal{J} \\
& w_j - m v_j \leq 0 & \forall j \in \mathcal{J} \\
& \displaystyle\sum_{i \in \mathcal{I}_s : d_{i,j} \geq r_i^s} u_{i,j}^s - 2w_j \leq 0 & \forall j \in \mathcal{J}, \ s \in \mathcal{S} \\
& \displaystyle\sum_{j \in \mathcal{J} : d_{i,j} \geq r_i^s} u_{i,j}^s \leq 1 & \forall s \in \mathcal{S}, \ i \in \mathcal{I}_s \\
& \displaystyle\sum_{i \in \mathcal{I}_s} \sum_{j \in \mathcal{J} : d_{i,j} \geq r_i^s} u_{i,j}^s \geq \alpha L & \forall s \in \mathcal{S} \\
& u_{i,j}^s \in \{0,1\}, \quad v_j \in \{0,1\}, \quad w_j \in \mathbb{Z} & \forall s \in \mathcal{S}, \ i \in \mathcal{I}_s : d_{i,j} \leq r_i^s, \ j \in \mathcal{J}.
\end{array}\right\} \tag{P}$$

~~The program~~ This model can be solved to near-optimality for reasonable choices of $\mathcal{J}$ using any commercial solver for mixed integer programs. In our case, however, we may be required to solve a potentially difficult problem many times and therefore in each iteration of Algorithm 1 we will not always solve (P) to optimality. Instead we employ an incumbent call back for two reasons. The first is to check if there is an improvement in the objective function of at least ϵ and the second is to reset the time that is allowed between each incumbent solution. ~~An implementation of the solution method~~ The pseudo code for the function SOLVE_MILP is given in Algorithm 3~~, constituting the function SOLVE_MILP in Algorithm 1, where a time limit t is set on the search, and we require a set improvement, ϵ on the objective value each time the model is solved.~~ In ~~particular~~ this algorithm, the function SOLVE denotes the call of the commercial ~~built-in~~ branch-and-cut solver we utilize, and ADD_WARMSTART is a built-in function which ~~prevents losing the last found solution if the algorithm breaks~~ adds a warm start to the solver, and consequently a upper bound on (P).

5.3 ~~Location improvement heuristic~~ Find improved locations

Having solved the linear problem (P) for a given set $\mathcal{J}$, we can further improve the locations of all the CSs in continuous space ~~using heuristic~~ a local search heuristic outlined in Algorithm 4. The algorithim takes as input components the solution, namely (u, v), returned from SOLVE_MILP and returns a set of improved PCLs, denoted by $\mathcal{K}$. The algorithm begins by populating the set $\mathcal{J}_A$ which is the set of PCLs at which CSs are built. Thereafter,~~For each active CS, $j \in \mathcal{J}_A \subseteq \mathcal{J}$~~, the set of EVs allocated to ~~that~~ each CS in $\mathcal{J}_A$ in every sample $s \in \mathcal{S}$ which is ~~precisely a subset~~ $\mathcal{I}_{A_j} \subseteq \mathcal{I}_s$ satisfying $u_{i,j}^s = 1$, for some $s \in \mathcal{S}$. The location of the active CS $j \in \mathcal{J}_A$ can then be improved by one of two ways. If the distance from $i \in \mathcal{I}_{A_j}$ is less than or equal to the geometric median of the coordinate tuples in $\mathcal{I}_{A_j}$, then we simply add the geometric median as a new coordinate PCL and avoid further computations. Note that the standard geometric median of a set of points, $\mathcal{A} = \{\boldsymbol{a}_i : i \in \mathcal{I}_{A_j}\}$ is defined as $\text{GEOMETRIC\_MEDIAN}(\mathcal{A}) = \arg\min_{\boldsymbol{g} \in \mathbb{R}^2} \sum_{i \in \mathcal{I}_{A_j}} ||\boldsymbol{a}_i - \boldsymbol{g}||$, where $|| \cdot ||$ is the standard 2-norm in $\mathbb{R}^2$. Otherwise, the

Algorithm 3 Solve MILP

Input: (P) : MILP, T : time limit in seconds, ϵ : objective value improvement tolerance
Output: $f, (u, v, w)$

procedure SOLVE_MILP($(P), T, \epsilon$)
  $f_{\text{old}} \leftarrow 0$
  if $(u, v, w)_{\text{WS}} \neq \emptyset$ **then** ▷ This is the case in the first iteration
    $f_{\text{WS}},\ (u, v, w) \leftarrow$ ADD_WARMSTART($(u, v, w)_{\text{WS}}$)
    $f_{\text{old}} \leftarrow f_{\text{WS}}$
  end if
  while $t \leq T$ **do**
    $f,\ (u, v, w) \leftarrow$ SOLVE(P)
    if incumbent solution found **then**
      if $|f_{\text{old}} - f| \leq \epsilon$ **then**
      else $f_{\text{old}} \leftarrow f$
        $t \leftarrow 0$
      end if
    end if
  end while
  return $f,\ (u, v, w)$
end procedure

minsum problem,

$$\begin{aligned} \min \quad & \sum_{i \in \mathcal{I}_{A_j}} \tilde{d}_{i,j} && (D_j) \\ \text{subject to} \quad & 0 \leq \tilde{d}_{i,j} \leq r_i^s \quad i \in \mathcal{I}_{A_j}, s \in \mathcal{S}, \end{aligned}$$

can be solved using a trust region method to obtain the optimal placement of a CS when considering the sum of distances to its allocated EVs. By repeating this process for every active CS, $j \in \mathcal{J}_A$, we obtain a full set of improved locations, $\mathcal{K}$.

5.3.1 Filtering PCLs

Since some CSs may already be in their best possible location for a given allocation, adding another location may not be necessary or desirable. Algorithm 5 avoids this issue. By filtering through existing (old) locations, $\mathcal{J}$, and newly identified PCLs, $\mathcal{K}$, we check if the Euclidean distance between each location, $\boldsymbol{x}_j$, and $\boldsymbol{x}_k$, is less than a set distance $d_{\min}$. If it is, then a Monte Carlo method is adopted to randomly remove some, but not all, new locations. The allocation may be different in the next solution, and hence we may not wish to remove all such instances, but still maintain a diverse population of PCLs. If there are plenty of chargers within a radius κ of station $j \in \mathcal{J}$, and few EVs within the same radius expecting to use station $j \in \mathcal{J}$, then the PCL $k \in \mathcal{K}$ is most likely not needed. Let C_{EV}, C_{CS} denote the number of EVs and PCLs within radius κ, respectively. Hence the ratio,

$$\rho := \frac{\mathbb{E}(p)\, C_{\text{EV}}}{2m\, C_{\text{CS}}},$$

measures the "probability" that another PCL, $k \in \mathcal{K}$, will be needed nearby. Note that $\mathbb{E}(p)$, as previously defined, is the expected proportion of EVs needing to charge, and m is the number of chargers at a CS. Next, a random uniform number ω is generated from $\Omega \sim$ U(0,1). If it is less than ρ, PCL k is excluded from set $\mathcal{K}$.

5.4 Warm start

A new solution (warm start) can be constructed manually from the improved locations identified in §5.3. The function CONSTRUCT_WARMSTART, the pseudo code for which is given in Algorithm 6, takes as input a feasible solution to (P), sets, and returns $(u, v, w)_{\text{WS}}$, a solution constructed from the old solution and the new improved

Algorithm 4 Location improvement heuristic

Input: u, v : solution to (P) (locations and allocations)
Output: $\mathcal{K}$: set of improved locations, $\mathcal{J}_A$: old active locations, $\{\mathcal{I}_{A_j}\}_{j\in\mathcal{J}_A}$: old set of allocated EVs

procedure FIND_IMPROVED_LOCATIONS(u, v)
  $\mathcal{K} \leftarrow \emptyset$
  $\mathcal{J}_A \leftarrow \{j \in \mathcal{J} : v_j = 1\}$ ▷ Find active CSs
  for $j \in \mathcal{J}_A$ **do**
    $\mathcal{I}_{A_j} \leftarrow \emptyset$
    for $s \in \mathcal{S}$ **do**
      $\mathcal{I}_{A_j} \leftarrow \mathcal{I}_{A_j} \cup \{i \in \mathcal{I} : u^s_{i,j} = 1\}$ ▷ Find EVs allocated to CS j, and their ranges
    end for
    $\mathcal{A} \leftarrow \{\boldsymbol{a}_i : i \in \mathcal{I}_{A_j}\}$
    $\boldsymbol{g} \leftarrow$ GEOMETRIC_MEDIAN$(\mathcal{A})$
    if $||\boldsymbol{a}_i - \boldsymbol{g}|| \leq r^s_i$ for all $i \in \mathcal{I}_{A_j}$ **then**
      $\mathcal{K} \leftarrow \mathcal{K} \cup \boldsymbol{g}$ ▷ Add new location
    else
      $k \leftarrow$ SOLVE_DIST(D_j)
      $\mathcal{K} \leftarrow \mathcal{K} \cup k$
    end if
  end for
end procedure

locations. For each of the new locations, the variable v is set to 1 to indicate that now it is active, and w is set to the same number as in the location it is replacing. For each CS which was active in the previous solution, we set the variables v, w to zero to revert the decision on the built locations. Next, the old allocations are removed, and the new allocations are made, where the distance is feasible.

Warm start should be before editing the set J so that we don't have to keep track of old indices.

6 Output analysis

After obtaining a good solution, (u^*, v^*, w^*), across multiple scenarios using Algorithm 1, it may be the case that the solution and associated cost is exceptionally optimistic, e.g. if the randomly generated samples are all very similar, or too few replications were performed. When performing the Monte Carlo-type simulation described in §4.4, we obtain a collection of Boolean values, $\{\delta^s\}^{s\in\mathcal{S}}$, describing which EVs need to charge in each scenario $s \in \mathcal{S}$. These are correlated with the EV driving ranges, $\{r^s\}^{s\in\mathcal{S}}$, generated from a given truncated normal distribution. Adopting a statistical mechanics interpretation, the pair, $\{(\delta^s, r^s)\}^{s\in\mathcal{S}}$, makes up an ensemble of all possible configurations of the system of EVs [16]. There are only finitely many configurations of EVs needing to charge or not, but there are nevertheless infinitely many virtual copies of the ensemble since the ranges are drawn from a continuous distribution.

Similarly, the spatial search space of the problem is continuous so we allow infinitely many possible placements for CSs. Hence there is no guarantee of optimality in the two-stage location-allocation problem which chooses from only a finite set of PCLs. However, to evaluate the statistical validity of the solution, a re-allocation is performed over another (larger) set of replications, to establish the total cost within a 95% confidence interval of its true value from the previous section.

6.1 Confidence interval for the total cost

Having established where to place CSs, and how many to install at each, we test the system on new replications to obtain allocations which depend on the random state of the ensemble $\{(\delta^s, r^s)\}^{s\in\mathcal{S}}$. Setting $v = v^*$, $w = w^*$, in

Algorithm 5 Improved location filtering

Input: $\mathcal{J}$: old locations, $\mathcal{K}$: new locations, $d_{\min}$: smallest distance allowed between CSs, κ : radius from CS
Output: $\mathcal{J}$: old (filtered) locations, $\mathcal{K}$ new (filtered) locations

procedure FILTER_LOCATIONS($\mathcal{J}, \mathcal{K}, d_{\min}, \kappa$)
    for $j \in \mathcal{J}, k \in \mathcal{K}$ **do**
        if $||\boldsymbol{x}_j - \boldsymbol{x}_k|| \leq d_{\min}$ **then**
            $C_{\text{EV}} \leftarrow |\{i \in \mathcal{I} : d_{i,j} \leq \kappa, j \in \mathcal{J}\}|$ ▷ Number of EVs within given radius
            $C_{\text{CS}} \leftarrow |\{j \in \mathcal{J} : d_{i,j} \leq \kappa, i \in \mathcal{I}\}|$ ▷ Number of CSs within given radius
            $\rho \leftarrow \frac{\mathbb{E}(p) C_{\text{EV}}}{2m C_{\text{CS}}}$ ▷ Ratio of EVs needing charge and available chargers within radius
            Generate $\Omega \sim \text{U}(0{,}1)$
            if $\Omega = \omega < \rho$ **then**
                $\mathcal{K} \leftarrow \mathcal{K} \setminus k$ ▷ Controlled random removal of PCLs
            end if
        end if
    end for
end procedure

Algorithm 6 Warm start construction

Input: (u, v, w) : solution to (P), $\mathcal{J}_A$: old active locations, $\{\mathcal{I}_{A_j}\}_{j \in \mathcal{J}_A}$: old set of allocated EVs, $\mathcal{K}$: new locations
Output: $(u, v, w)_{\text{WS}}$: updated warm start solution

procedure CONSTRUCT_WARMSTART($(u, v, w), \mathcal{J}_A, \mathcal{K}$)
    for $j \in \mathcal{J}_A, k \in \mathcal{K}$ **do**
        $v_k \leftarrow 1$ ▷ New location made active
        $w_k \leftarrow w_j$ ▷ Assume the same amount of chargers
        $v_j \leftarrow 0$ ▷ Old locations/chargers are removed
        $w_j \leftarrow 0$
        for $s \in \mathcal{S}, i \in \mathcal{I}_{A_j}$ **do**
            $u^s_{i,j} \leftarrow 0$
            if $d_{i,k} \leq r^s_i$ **then**
                $u^s_{i,k} \leftarrow 1$ ▷ if $d_{i,k} > r^s_i$ raise error, not reachable
            end if
        end for
    end for
end procedure

MILP (P) only the variable $u_{i,j}$ remains to be evaluated. The resulting allocation problem is

$$\begin{aligned}
\min \quad & \frac{365}{|S|}(c_{\tilde{c}} + c_d) \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} d_{i,j} u_{i,j} + K && && (Q)\\
\text{subject to} \quad & \sum_{i \in \mathcal{I}} u_{i,j} \leq 2w^*_j && j \in \mathcal{J}, d_{i,j} \leq r_i \\
& \sum_{j \in \mathcal{J}} u_{i,j} \leq 1 && i \in \mathcal{I}, d_{i,j} \leq r_i \\
& \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} u_{i,j} \geq \alpha L && d_{i,j} \leq r_i \\
& u_{i,j} \in \{0, 1\} && i \in \mathcal{I}, j \in \mathcal{J},
\end{aligned}$$

where

$$K = \sum_{j \in \mathcal{J}} (c_b v^*_j + c_m w^*_j) + \frac{365}{|\mathcal{S}|} |\mathcal{J}| c_{\tilde{c}} \sum_{i \in \mathcal{I}} \sum_{s \in \mathcal{S}} (250 - r^s_i)$$

is a constant from the location-allocation problem (P).

Let n denote the number of replications, i.e. the number of times (Q) is re-solved. In each replication, we obtain the optimal values $Z_1, Z_2, ..., Z_n$. These are independent and identically distributed in terms of the ensemble of demand configurations $\{(\delta^s, r^s)\}^{s\in\mathcal{S}}$, with mean, $\mathbb{E}(Z_i)$, and variance, σ^2. By the central limit theorem, the sample mean, $\bar{Z} = \sum_{i=1}^{n} Z_i/n$, should follow a standard normal distribution $\bar{Z} \sim \mathcal{N}(\bar{Z}, \sigma^2/n)$. A confidence interval for the mean cost can then be calculated as

$$\left(\bar{Z} - \frac{t_{n-1,1-\alpha/2}\,\sigma}{\sqrt{n}}, \bar{Z} + \frac{t_{n-1,1-\alpha/2}\,\sigma}{\sqrt{n}}\right),$$

where t is the test statistic on a $(1-\alpha)100\%$ confidence interval with $n-1$ degrees of freedom, and σ is the standard deviation of each Z_i. For the normal distribution, the cumulative distribution gives that $P(-1.96 < \bar{Z} < 1.96) = 0.95$, so we can set $t = 1.96$. The number of replications needed to obtain an interval of relative width $\epsilon\bar{Z}$, is

$$n = \left(\frac{t_{n-1,1-\alpha/2}\,\sigma}{\epsilon\bar{Z}}\right)^2 = ...$$

This can be shown by plugging this expression in for n in the confidence interval.

6.2 Sensitivity analysis

To further analyse how sensitive the model is to small perturbations in some of its parameters, we perform a sensitivity analysis on a set of potentially interesting factors. Firstly, since the heuristic approach is more dependent on initial locations, we test how the number of initial locations, $\tilde{k}$, affects the objective value obtained at the end of the optimization procedure 1. Secondly, the service level.

Thirdly, the maximum number of chargers at each CS, m.

6.2.1 Number of initial stations

Figure 3 shows that when the solver is fed too few initial locations, the problem is infeasible. This is because we enforce the 95% service level constraint from the very first iteration. For between 300 and 550 initial locations, the final objective value is stable. All trials were performed with a 300 second time limit. We see that initialising the model with a large number of locations results in a worse final objective value: the model would require more time to search this larger solution space.

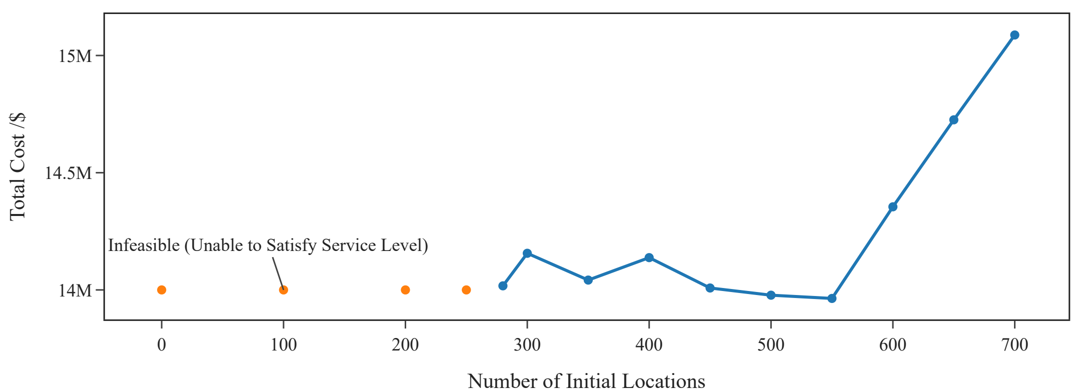


Figure 3: Objective value for different numbers of initial locations using k-means clustering.

6.2.2 Service level

what we c
is to find
highest at
tainable s
vice level
a maximu
matching
on a bipar
tite graph
(charging
cars to 2
n nodes).

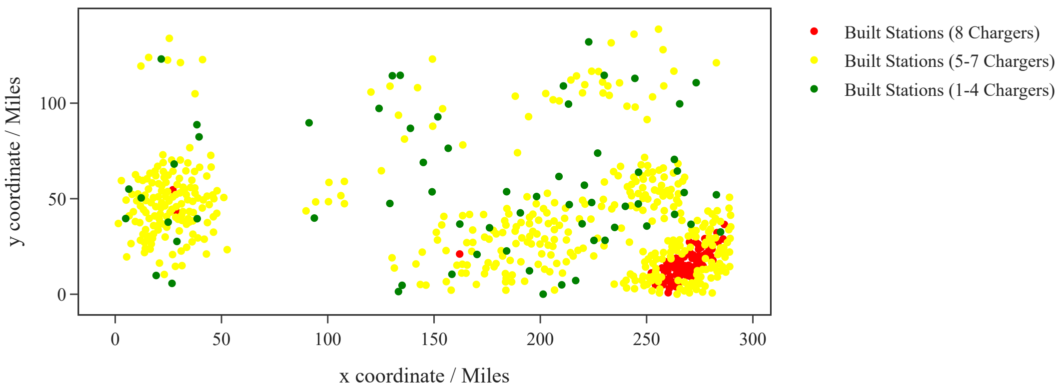


Figure 4: Solution: 600 Station Locations

an example of the bipartite graph with maximum matching would be nice

Compute the attainable service level as a maximum bipartite matching between the EVs that require charging and are within reach of a CS, and all active CSs.

Also, percentage of EVs not in reach of a charger (breaking down).

6.2.3 Number of chargers at each station

Utilization of charger.

7 Final recommendations

Figure X shows a solution, f, (v, w), we obtain in section 5 in terms of built CSs and how many chargers are installed at each, with 3 scenarios considered. The total number of chargers installed is X, and the total cost is \$. By solving (Q) times (performing n replications), we obtain the mean, and variance, respectively, n

$$\bar{Z} = \sum_{i=1}^{n} \frac{Z_i}{n} = ..., \qquad \sigma^2 = \sum_{i=1}^{n} \frac{(Z_i - \bar{Z})^2}{n-1} = .$$

Note that the original solution, $f =$, deviates from the mean $\bar{Z}$ by %. The 95% confidence interval for the total cost is hence

$$\left(\bar{Z} - \frac{t_{n-1,1-\alpha/2}\,\sigma}{\sqrt{n}}, \bar{Z} + \frac{t_{n-1,1-\alpha/2}\,\sigma}{\sqrt{n}} \right).$$

We provide a construction plan for 600 charging stations. The Total Cost is \$15418221 (\$ 15.42M), and y chargers are built. The total solve time for this solution was 1 hour, 8 minutes (4153 seconds), and the solver took the following parameters: timelimit=300, epsilon=10000, number of initial locations = 600.

Figures 4 and 5 show strong alignment between the vehicle locations and the infrastructure plan. Since each charging station can meet the demand of 16 vehicles, each vehicle location can be fully satisfied by a single station, and at most 1079 stations would ever be built.

When we relax the condition to build a fixed number of charging stations, a solution with lower total cost can be obtained. Our construction plan is shown in ... X stations are built, with a total of y chargers.

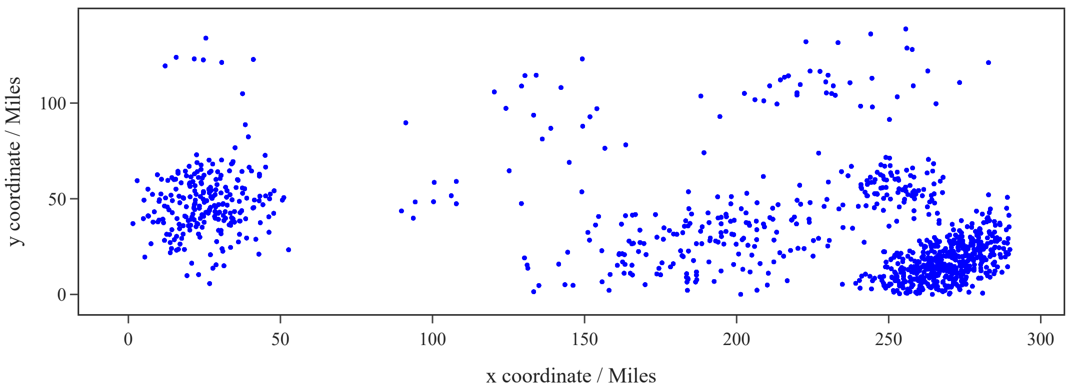


Figure 5: Vehicle Locations provided in Problem Set-up

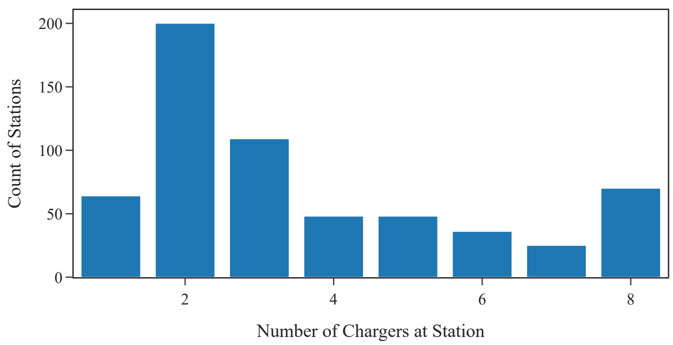


Figure 6: Distribution of Chargers in the 600 Station Solution

7.1 Graphical User Interface

We have built an interactive front end to our solver with the python library Streamlit. On opening the application, the user is presented with a map of the vehicle locations. The 'Optimise' button calls our solver, and after each iteration, the map, along with metric boxes, and additional plots are updated. In the left-hand sidebar, the user can select different solver parameters, fix the number of stations to be built, or provide a new dataset of vehicle locations. Once the optimisation run is complete, the 'Validate' button tests the solution for robustness on a specified number of new samples, and plots the results.

Screenshots: - Interface Initially. - Maps of finished solutions with KPIs. - Validation Example.

7.1.1 Error messages

Explain possible error messages, e.g. CPLEX built-in warnings.

8 Possible Extensions

The fact that some cars decision variables u are excluded from the model. Realistically, we may assume drivers rarely let the driving range fall below a reasonable mileage as most systems provide multiple low-charge warnings, or enter battery-saving mode below a given threshold [18]. An EV which is critically low on charge would arguably benefit more from a nearby battery-swapping station rather than a standard CS [19].

We should go over possible extensions and improvements of the model to make it more realistic. EV owners have agency, and can see which chargers are available, so they will go to the nearest available, which is something we have not incorporated. Discuss more accurate model of agency, time periods, > 10.000 replications.

In addition, CSs may be interconnected on an energy grid for example, which further complicates the nonlinear model and has been shown to be NP-hard [2].

Brief justification of "greater good/authority" model rather than game theory model.

Instead of summing over scenarios, we could consider expected cost.

remove lin

References

[1] *Alternative Fuel Corridors - Environment - FHWA*. URL: https://www.fhwa.dot.gov/environment/alternative_fuel_corridors/ (visited on 05/29/2023).

[2] V. Blanco and R. Gázquez. “Continuous maximal covering location problems with interconnected facilities”. en. In: *Computers & Operations Research* 132 (Aug. 2021), p. 105310. ISSN: 03050548. DOI: 10.1016/j.cor.2021.105310. URL: https://linkinghub.elsevier.com/retrieve/pii/S0305054821000988 (visited on 02/06/2023).

[3] L. Cooper. “An Extension of the Generalized Weber Problem†”. en. In: *Journal of Regional Science* 8.2 (1968). _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-9787.1968.tb01323.x, pp. 181–197. ISSN: 1467-9787. DOI: 10.1111/j.1467-9787.1968.tb01323.x. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9787.1968.tb01323.x (visited on 05/16/2023).

[4] L. Cooper. “Heuristic Methods for Location-Allocation Problems”. In: *SIAM Review* 6.1 (1964). Publisher: Society for Industrial and Applied Mathematics, pp. 37–53. ISSN: 0036-1445. URL: https://www.jstor.org/stable/2027512 (visited on 05/16/2023).

[5] L. Cooper. “Location-Allocation Problems”. In: *Operations Research* 11.3 (1963). Publisher: INFORMS, pp. 331–343. ISSN: 0030-364X. URL: https://www.jstor.org/stable/168022 (visited on 05/16/2023).

[6] M. Daskin. *Network and Discrete Location: Models, Algorithms and Applications*. Second Edition. John Wiley & Sons, Ltd, June 2013. URL: https://onlinelibrary.wiley.com/doi/book/10.1002/9781118537015.

[7] *Electric Vehicles in PA*. en-US. URL: https://www.dep.pa.gov:443/Business/Energy/OfficeofPollutionPrevention/ElectricVehicles/Pages/default.aspx (visited on 05/07/2023).

[8] R. Z. Farahani et al. “Covering problems in facility location: A review”. en. In: *Computers & Industrial Engineering* 62.1 (Feb. 2012), pp. 368–407. ISSN: 0360-8352. DOI: 10.1016/j.cie.2011.08.020. URL: https://www.sciencedirect.com/science/article/pii/S036083521100249X (visited on 02/06/2023).

[9] I. Frade et al. “Optimal Location of Charging Stations for Electric Vehicles in a Neighborhood in Lisbon, Portugal”. In: *Transportation Research Record* 2252.1 (Jan. 2011). Publisher: SAGE Publications Inc, pp. 91–98. ISSN: 0361-1981. DOI: 10.3141/2252-12. URL: https://doi.org/10.3141/2252-12 (visited on 02/06/2023).

[10] D. Guler and T. Yomralioglu. “Suitable location selection for the electric vehicle fast charging station with AHP and fuzzy AHP methods using GIS”. In: *Annals of GIS* 26.2 (Apr. 2020). Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/19475683.2020.1737226, pp. 169–189. ISSN: 1947-5683. DOI: 10.1080/19475683.2020.1737226. URL: https://doi.org/10.1080/19475683.2020.1737226 (visited on 05/29/2023).

[11] L. He et al. *Charging an Electric Vehicle Sharing Fleet*. en. SSRN Scholarly Paper. Rochester, NY, Sept. 2019. DOI: 10.2139/ssrn.3223735. URL: https://papers.ssrn.com/abstract=3223735 (visited on 02/24/2023).

[12] L. He et al. “Service Region Design for Urban Electric Vehicle Sharing Systems”. en. In: *Manufacturing & Service Operations Management* (Apr. 2017). Publisher: INFORMS. DOI: 10.1287/msom.2016.0611. URL: https://pubsonline.informs.org/doi/abs/10.1287/msom.2016.0611 (visited on 02/24/2023).

[13] K. Huang et al. “Analysis of Optimal Operation of Charging Stations Based on Dynamic Target Tracking of Electric Vehicles”. en. In: *Electronics* 11.19 (Jan. 2022). Number: 19 Publisher: Multidisciplinary Digital Publishing Institute, p. 3175. ISSN: 2079-9292. DOI: 10.3390/electronics11193175. URL: https://www.mdpi.com/2079-9292/11/19/3175 (visited on 02/06/2023).

[14] Y.-C. Hung, H. PakHai Lok, and G. Michailidis. “Optimal routing for electric vehicle charging systems with stochastic demand: A heavy traffic approximation approach”. en. In: *European Journal of Operational Research* 299.2 (June 2022), pp. 526–541. ISSN: 0377-2217. DOI: 10.1016/j.ejor.2021.06.058. URL: https://www.sciencedirect.com/science/article/pii/S0377221721005890 (visited on 02/06/2023).

[15] Ö. B. Kınay, F. Gzara, and S. A. Alumur. "Full cover charging station location problem with routing". en. In: *Transportation Research Part B: Methodological* 144 (Feb. 2021), pp. 1–22. ISSN: 0191-2615. DOI: 10.1016/j.trb.2020.12.001. URL: https://www.sciencedirect.com/science/article/pii/S0191261520304434 (visited on 05/07/2023).

[16] L. D. Landau and E. M. Lifshitz. "CHAPTER I - THE FUNDAMENTAL PRINCIPLES OF STATISTICAL PHYSICS". en. In: *Statistical Physics (Third Edition)*. Ed. by L. D. Landau and E. M. Lifshitz. Oxford: Butterworth-Heinemann, Jan. 1980, pp. 1–33. ISBN: 978-0-08-057046-4. DOI: 10.1016/B978-0-08-057046-4.50008-7. URL: https://www.sciencedirect.com/science/article/pii/B9780080570464500087 (visited on 05/30/2023).

[17] G. Laporte, S. Nickel, and F. Saldanha da Gama, eds. *Location Science*. en. Cham: Springer International Publishing, 2015. ISBN: 978-3-319-13110-8 978-3-319-13111-5. DOI: 10.1007/978-3-319-13111-5. URL: https://link.springer.com/10.1007/978-3-319-13111-5 (visited on 02/07/2023).

[18] M. K. Lim, H.-Y. Mak, and Y. Rong. "Toward Mass Adoption of Electric Vehicles: Impact of the Range and Resale Anxieties". en. In: *Manufacturing & Service Operations Management* (Dec. 2014). Publisher: INFORMS. DOI: 10.1287/msom.2014.0504. URL: https://pubsonline.informs.org/doi/abs/10.1287/msom.2014.0504 (visited on 02/24/2023).

[19] H.-Y. Mak, Y. Rong, and Z.-J. M. Shen. "Infrastructure Planning for Electric Vehicles with Battery Swapping". en. In: *Management Science* (Apr. 2013). Publisher: INFORMS. DOI: 10.1287/mnsc.1120.1672. URL: https://pubsonline.informs.org/doi/abs/10.1287/mnsc.1120.1672 (visited on 05/07/2023).

[20] P. D. o. E. Protection. *Driving PA Forward*. da. Mar. 2023. URL: https://storymaps.arcgis.com/stories/6f5db16b8399488a8ef2567e1affa1e2 (visited on 05/07/2023).

[21] L. V. Snyder. "Facility location under uncertainty: a review". In: *IIE Transactions* 38.7 (June 2006). Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/07408170500216480, pp. 547–564. ISSN: 0740-817X. DOI: 10.1080/07408170500216480. URL: https://doi.org/10.1080/07408170500216480 (visited on 02/24/2023).

[22] L. V. Snyder and M. S. Daskin. "Stochastic p-robust location problems". In: *IIE Transactions* 38.11 (Nov. 2006). Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/07408170500469113, pp. 971–985. ISSN: 0740-817X. DOI: 10.1080/07408170500469113. URL: https://doi.org/10.1080/07408170500469113 (visited on 02/24/2023).

[23] X. Xi, R. Sioshansi, and V. Marano. "Simulation–optimization model for location of a public electric vehicle charging infrastructure". en. In: *Transportation Research Part D: Transport and Environment* 22 (July 2013), pp. 60–69. ISSN: 1361-9209. DOI: 10.1016/j.trd.2013.02.014. URL: https://www.sciencedirect.com/science/article/pii/S1361920913000345 (visited on 02/06/2023).

[24] Z. Yi et al. "An agent-based modeling approach for public charging demand estimation and charging station location optimization at urban scale". en. In: *Computers, Environment and Urban Systems* 101 (Apr. 2023), p. 101949. ISSN: 0198-9715. DOI: 10.1016/j.compenvurbsys.2023.101949. URL: https://www.sciencedirect.com/science/article/pii/S0198971523000121 (visited on 05/07/2023).

A Optimization model