Scheduling Drones to Recharge Electric Vehicles

Eleni Karagianni, Emmanouil S. Rigas, Nick Bassiliades
{karageleni,erigas,nbassili}@csd.auth.gr
School of Informatics, Aristotle University of Thessaloniki,
Thessaloniki, Greece

ABSTRACT

Drones and Electric Vehicles (EVs) are two technologies that are growing fast and have the potential to revolutionize the transportation sector. In this paper, we take inspiration from a recent patent submitted by Amazon, and we study the problem of scheduling drones that carry a large battery and can partially recharge EVs when a ground charging station is not available. The drones have a limited range, as they are also using an electric motor powered by batteries which need regular recharging. Thus, given a fixed set of drones, the problem that arises is to calculate a schedule for the flying and the recharging of the drones aiming to maximize the number of EVs that are actually serviced. In this vein, we develop a greedy algorithm that uses a heuristic search mechanism. We evaluate our algorithm in a realistic setting and in a plethora of scenarios to verify its effectiveness.

CCS CONCEPTS

• **Computing methodologies → Planning and scheduling**; **Search methodologies**.

KEYWORDS

scheduling, heuristic, drones, electric vehicles

ACM Reference Format:
Eleni Karagianni, Emmanouil S. Rigas, Nick Bassiliades. 2022. Scheduling Drones to Recharge Electric Vehicles. In *12th Hellenic Conference on Artificial Intelligence (SETN 2022), September 7–9, 2022, Corfu, Greece.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3549737.3549763

1 INTRODUCTION

In recent years, Electric Vehicles (EVs) have returned to the spotlight due to their technological developments [12] and the increasing interest in renewable energy sources [7]. It is a fact that the adoption of EVs in parallel with the development of the smart grids [13] is considered a keystone for reducing carbon emissions which cause the greenhouse effect and the consequent climate change.

Unmanned Aerial Vehicles (UAVs) or simply drones are vehicles and can fly either autonomously based on a pre-programmed flight plan, or they can be controlled by a remote operator [5]. The rapid development and the direct influence of this technology on the quality of human life is remarkable. Drones are becoming increasingly popular due to their small size, easy deployment, low maintenance costs and high flexibility. Currently drones are used in a multitude of applications such as transportation of goods [4], search and rescue [8], traffic monitoring [2] and healthcare [14] amongst others.

Despite their many advantages EVs still suffer from relatively low range, long charging times and unavailability of charging stations [11]. To tackle these issues intensive research is taking place in terms of utilizing the existing charging infrastructure efficiently [9, 16], or in terms of optimally placing charging stations [15]. Despite such advances, the possibility of having an EV running out of battery remains. This causes the so-called range anxiety [3], which is a central barrier towards the further adoption of EVs. Recently, Amazon filed a patent [6] that describes the use of UAVs that carry large batteries and can recharge EVs on the go. Despite the fact that such services may take years until they become a reality, using drones to recharge EVs raises several algorithmic challenges that can be tackled using powerful AI algorithms and tools. Specifically, given that drones are in principal electric vehicles that are using their own battery to fly, their use as a carriage of a battery to recharge another electric vehicle on the ground creates a highly complicated scheduling problem where several decisions need to be made: 1) Schedule the drones to maximize the number of EV charging requests that are completed. 2) Schedule the recharging of the drones. 3) Optimize the utilization of the (limited) available batteries that the drones are using to recharge the EVs, as well as the charging of these batteries. 4) Optimally select the stations where the drones are located.

In this paper, we study the problem of scheduling drones that carry a large battery to recharge EVs offline, where the charging requests are assumed to be known in advance. In so doing, we develop a greedy heuristic search algorithm. To the best of our knowledge, this is the first paper that studies this challenging problem. Two papers that share some similarities are presented next.

In [17] the authors consider an Internet of Things (IoT) scenario where devices located on the edge are recharged wirelessly by drones. In so doing, the authors aim to optimize the charging procedure, given a set of constraints, as well as the positions of the drones to maximize service. They propose both an optimal Integer Linear Programming (ILP) approach, as well as a suboptimal greedy one. In contrast to our work, in [17] the authors do not consider the recharging of the drones themselves, although they do consider the drones' limited range as a constraint.

In [10] the authors extend the well-known multiple traveling salesman problem and study the problem of scheduling drones to perform monitoring missions. They schedule the travel path of a set of drones across a graph, where nodes need to be visited multiple times at pre-defined points in time. They propose an optimal

SETN 2022, September 7–9, 2022, Corfu, Greece

ACM ISBN 978-1-4503-9597-7/22/09…$15.00
https://doi.org/10.1145/3549737.3549763

ILP solution, as well as a greedy algorithm that uses a one-step look-ahead heuristic search mechanism, and an algorithm that is based on ant colony optimization (ACO). In this work the authors optimize both the placement of the drones and the recharging of them, however their problem does not involve the recharging of another device.

2 PROBLEM DEFINITION

In this paper, we study the problem of scheduling drones that carry a large battery to recharge EVs. We model the area where both drones and EVs are moving in, as a fully connected undirected graph $G(N, E)$ and we assume a discrete time horizon with $t \in T \subseteq \mathbb{N}$ points in time to exist. The nodes $n \in N$ represent either the points where electric vehicles can stop and request charging n_{EV}, or the stations where drones n_{drn} can land. The edges $e \in E \subseteq N \times N$ of the graph represent the paths connecting all pairs of nodes. Each edge has a time t_e that any drone needs to fly over it, assuming a fixed average speed. Moreover, we consider a set of drones $i \in I \subseteq \mathbb{N}$. In this context we assume that all drones have an initial location n_{drn}^{init} and that all vehicles' charging requests $r \in R \subseteq \mathbb{N}$ are collected in advance. A charging request consists of a tuple $p = \{n_{EV,r}, b_r, t_r^{arr}, t_r^{dep}\}$. Specifically, $n_{EV,r}$ is the location where the EV is asking to charge an amount of energy b_r within a given time window $t_r^{dep} - t_r^{arr}$. At each time point, a drone that is not at its station, i.e., either flying or charging an EV, consumes c% of its battery, assuming a fixed average speed. At each time point, a drone can charge an EV with $c \in \mathbb{N}$ units of energy. In other words, if the EV demands x units of energy, it will take $\lceil x/c \rceil$ times for a drone to charge this energy. Now, each drone is assumed at $t = 0$ to have a fully charged battery that allows it to operate for τ_i points in time, and a battery containing b units of energy for EV charging. Once a drone returns to a station, both batteries are assumed to be replaced with fully charged ones (i.e., battery swapping [1] is used). Moreover, we assume that if a drone is en-route to charge an EV, this route cannot change, even if another request has been made. An example representation of the problem is depicted in Figure 1.

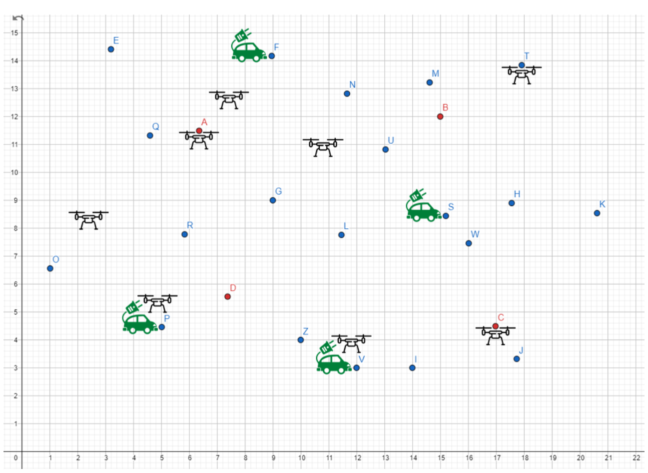

Figure 1: Example representation

3 SCHEDULING ALGORITHM

The algorithm for scheduling the flight of drones in order to maximize the EVs that are serviced is detailed below.

We initially read the data regarding the graph and we store them in a two-dimensional table, and we also read the charging requests and store them in a queue. Then, at each point in time, we check if there is a request from an electric vehicle (see line 2 of Algorithm 1). If such a request exists, we follow a procedure to find the most suitable drone (see line 4 of Algorithm 1) that can serve the request.

Firstly, we find which of the drones, that are neither flying, committed, nor selected in a previous request can serve the request, i.e., which drones have enough battery power both for the electric vehicle and to travel to the EV's location, charge the EV, and return to their station (see lines 3-5 of Algorithm 2). We group these drones into available and busy. The availableDrones are those that are at their station, while the busyDrones are those that are currently at another node and are charging another EV. We find the drone with the shortest distance from the node where the request was made, both from the availableDrones and from the busyDrones (see lines 1-6 of Algorithm 3). If the distance calculated from the availableDrones is less than the distance from the busyDrones, then the available drone is selected (see lines 7-8 of Algorithm 3). In the opposite case, we consider how many points in time the busy drone needs until it finishes charging the previous electric vehicle. Finally, after comparing the two values, the drone that is going to reach sooner the node, where the electric vehicle is located, is selected (see lines 9-11 of Algorithm 3).

In case more than one drones ha the shortest distance, either from the availableDrones or from the busyDrones, we divide them into minDistAvailable and minDistBusy. If a drone is to be selected from minDistAvailable, then we send the one with the higher battery level (see line 8 of Algorithm 3). If a drone from minDistBusy is to be selected, then we find which one finishes charging the previous EV earlier and bind it for this request (see line 10 of Algorithm 3). If again more than one drones exist, (i.e. if two or more drones end at the same time), then the one with the highest battery level is chosen.

After finding the most suitable drone, we make some necessary changes (see lines 5-9 of Algorithm 1). We define the position in which the drone is located at each time point. In addition to the N nodes of the graph, the position of the drone can be set equal to -1 if it is flying , or -3 if it is at another node and is committed for the current request. We reduce the battery level accordingly from the drone and we subtract from the battery for electric vehicles the units requested by the EV.

Unless a drone that can serve the request is found, then we inform the EV that the request cannot be served and continue with the next one. At the end of each time point, we perform the necessary checks and coordination of the drones (see lines 10-11 of Algorithm 1). More specifically, for the drones located at one of the nodes and are not selected at the given time, if it is the last instant of time they are charging the electric vehicle, then we send them back to their station. We update their position to -1 for each time instant until they reach the station and set their batteries to full.

Algorithm 1 Main scheduling algorithm

for each time t **do**
  if there is a demand at time t **then**
    Remove the demand from the Queue
    $FindSuitableDrone(node, demand, time)$, Alg. 2
    if chosen drone is at its station **then**
      Set its position equal to -1 for the times it flies
    else if chosen drone is at another node **then**
      Set its position equal to -3 for the times until it finishes charging the previous EV and -1 for the times it flies
    Update the value of two batteries
    Coordination ▷ The drones needed are going back to the station
  Coordination

Algorithm 2 Finding suitable drone - preprocessing

Create two ArrayLists ▷ availableDrones and busyDrones
for each Drone d **do**
  if $pos! = -1$ and $pos! = -3$ **then**
    Calculate the battery it needs to spend
    if $droneBat >= batNeeded$ and $droneBatForEV >= demand$ **then**
      Add drone at the appropriate ArrayList
Call Alg. 3 providing $droneWithMinDist(availableDrones, busyDrones, node, time)$

Algorithm 3 Finding closest available drone

Create ArrayList minDronesAvailable ▷ Availble drones with min distance
if $availableDrones.size > 0$ **then**
  From available drones find those with the min distance from the node
Create ArrayList minDronesBusy ▷ Busy drones with min distance
if $busyDrones.size > 0$ **then**
  From busy drones find those the min distance from the node
if $minDistAvailable < minDistBusy$ **then**
  From minDronesAvailable find and return the drone with max battery
else if $minDistAvailable >= minDistBusy$ **then**
  From minDronesBusy find the drone that will finish sooner charging the previous EV
  Return the drone that will reach sooner to the node

4 EVALUATION

After developing the algorithm as this was presented in the previous chapter, we proceed in its evaluation by testing its performance in different scenarios. In all cases we used a graph with 24 nodes, four of which representing the stations for the drones, where each of which initially has three drones, and the remaining 20 nodes represent the points where the electric vehicles can stop and request charging.

First Experiment

The purpose of this experiment is to evaluate how the algorithm behaves and how many vehicles are served in each case. In this experiment, we create 3 scenarios by changing the number of drones and requests each time.

More specifically, the first scenario consists of 40 requests, while in each of the following scenarios we add 10 requests. In all cases the total number of drones belongs to the set 4, 8, 12, 16, 20, 24 and the algorithm runs for 100 time points.

The graphs below show how the percentage of requests served changes as the number of drones increases in each of the three cases. On the horizontal axis we have the number of drones and on the vertical axis we have the percentage of requests served. Observing Figure 2, although at the beginning the percentage of vehicles served is not quite satisfactory, we see that with 16 drones and more we serve all our vehicles. This means that it would be redundant if we used more than 16 drones for this scenario. Similar conclusions are drawn for the other two scenarios we tested. Observing the three curves, it is easy to see that as the number of drones increases, so does the number of vehicles served.

Interestingly, we observe that while there is a gradual increase in the number of vehicles served, as expected, it seems that at a certain point the upward curves stop rising. Thus, we can easily decide how many drones we should ideally use in order to service the highest percentage of requests without a surplus of drones.

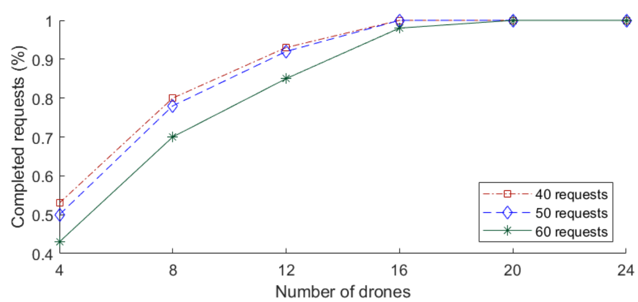

Figure 2: Efficiency with variable number of drones and fixed requests

In another experiment, we evaluate how the increase in charging requests affects the EVs that are actually charged. By observing Figure 3, we see that the percentage of vehicles serviced differs for different numbers of requests, with the same number of drones. More specifically, while at 20 requests we serve all vehicles, at 80 requests the percentage decreases to 61%.

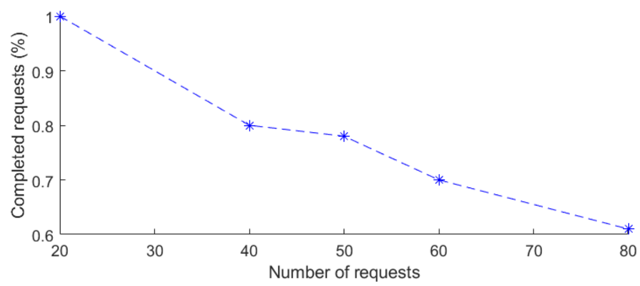

Figure 3: Efficiency with variable number of requests and fixed drones

Second Experiment

The purpose of this experiment is to observe how many vehicles are served as the distribution of requests in time becomes increasingly sparse. To achieve this, we consider a setting where we increase the number of points in time from 100 to 400 with a step of 100, keeping the number of requests fixed to 50 and the number of drones fixed to 4.

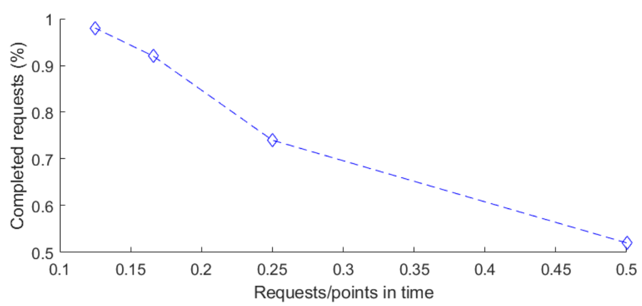

Figure 4: Efficiency with variable sparsity of requests

As we can observe from the results of this experiment, as these are depicted in Figure 4, when the sparsity of requests decrease, the percentage of vehicles served also decreases. More specifically, when we have 100 time points, almost half of the drones are serviced and this is because for many incoming requests there is no availability of drones, as they have not had time to release from previous requests. In contrast, when we have 400 time points, almost all requests are serviced since the availability of drones is much higher. So, we conclude that when the demand is more sparse in the time horizon, the percentage of requests that are completed increases, as more drones are available to service vehicles.

5 CONCLUSIONS AND FUTURE WORK

In this paper we dealt with a problem that arises due to the lack of charging stations for electric vehicles. The idea is to fly a drone to the point where an electric vehicle has stopped and charge it. We therefore needed to design an algorithm to coordinate and schedule the flight of drones in order to maximize the completion of charging requests. We concluded that in order to make the algorithm as efficient as possible, we need to pay special attention to the parameters we set on it. These parameters are the number of drones, the number of requests and the range of time instants. The combination of these three elements plays a crucial role in the results the algorithm will give.

An idea for future work is to collect data from real events, apply the algorithm on them and then build some demand prediction model. Moreover, based on the algorithm we built, we could develop an optimal algorithm, in addition to the heuristic one which can operate as a benchmark for other non-optimal algorithms. Additionally, an algorithm that will be able to solve the problem in an online manner would be very useful since charging requests are more possible to arrive on the fly. Finally, a tool to simulate this scenario using graphics could be developed in the future.

REFERENCES

[1] Furkan Ahmad, Mohammad Saad Alam, Ibrahim Saad Alsaidan, and Samir M Shariff. 2020. Battery swapping station for electric vehicles: opportunities and challenges. *IET Smart Grid* 3, 3 (2020), 280–286.

[2] Emmanouil Barmpounakis and Nikolas Geroliminis. 2020. On the new era of urban traffic monitoring with massive drone data: The pNEUMA large-scale field experiment. *Transportation research part C: emerging technologies* 111 (2020), 50–71.

[3] Fang Guo, Jun Yang, and Jianyi Lu. 2018. The battery charging station location problem: Impact of users' range anxiety and distance convenience. *Transportation Research Part E: Logistics and Transportation Review* 114 (2018), 1–18.

[4] Robin Kellermann, Tobias Biehle, and Liliann Fischer. 2020. Drones for parcel and passenger transportation: A literature review. *Transportation Research Interdisciplinary Perspectives* 4 (2020), 100088.

[5] Christos Kyrkou, Stelios Timotheou, Panayiotis Kolios, Theocharis Theocharides, and Christos Panayiotou. 2019. Drones: Augmenting Our Quality of Life. *IEEE Potentials* 38, 1 (2019), 30–36. https://doi.org/10.1109/MPOT.2018.2850386

[6] Jon Arron McClintock, Daniel Buchmueller, Varadarajan Gopalakrishnan, Fabian Hensel, Jesper Mikael Johansson, Brandon William Porter, and Andrew Jay Roths. 2017. Systems, devices and methods delivering energy using an uncrewed autonomous vehicle. US Patent 9,778,653.

[7] NL Panwar, SC Kaushik, and Surendra Kothari. 2011. Role of renewable energy sources in environmental protection: A review. *Renewable and sustainable energy reviews* 15, 3 (2011), 1513–1524.

[8] Sarvapali D Ramchurn, Trung Dong Huynh, Feng Wu, Yukki Ikuno, Jack Flann, Luc Moreau, Joel E Fischer, Wenchao Jiang, Tom Rodden, Edwin Simpson, et al. 2016. A disaster response system based on human-agent collectives. *Journal of Artificial Intelligence Research* 57 (2016), 661–708.

[9] Emmanouil S Rigas, Enrico H Gerding, Sebastian Stein, Sarvapali D Ramchurn, and Nick Bassiliades. 2022. Mechanism Design for Efficient Offline and Online Allocation of Electric Vehicles to Charging Stations. *Energies* 15, 5 (2022), 1660.

[10] Emmanouil S. Rigas, Panayiotis Kolios, Michalis Mavrovouniotis, and Georgios Ellinas. 2021. Scheduling a Fleet of Drones for Monitoring Missions With Spatial, Temporal, and Energy Constraints. *IEEE Transactions on Intelligent Transportation Systems* (2021), 1–13. https://doi.org/10.1109/TITS.2021.3137359

[11] Simone Steinhilber, Peter Wells, and Samarthia Thankappan. 2013. Socio-technical inertia: Understanding the barriers to electric vehicles. *Energy policy* 60 (2013), 531–539.

[12] Xiaoli Sun, Zhengguo Li, Xiaolin Wang, and Chengjiang Li. 2020. Technology Development of Electric Vehicles: A Review. *Energies* (01 2020), 90. https://doi.org/10.3390/en13010090

[13] Maria Lorena Tuballa and Michael Lochinvar Abundo. 2016. A review of the development of Smart Grid technologies. *Renewable and Sustainable Energy Reviews* 59 (2016), 710–725.

[14] Sharon Wulfovich, Homero Rivas, and Pedro Matabuena. 2018. Drones in healthcare. In *Digital Health*. Springer, 159–168.

[15] Ying Zhang, Yanhao Wang, Fanyu Li, Bin Wu, Yao-Yi Chiang, and Xin Zhang. 2020. Efficient deployment of electric vehicle charging infrastructure: Simultaneous optimization of charging station placement and charging pile assignment. *IEEE Transactions on Intelligent Transportation Systems* 22, 10 (2020), 6654–6659.

[16] Yongmin Zhang, Pengcheng You, and Lin Cai. 2018. Optimal charging scheduling by pricing for EV charging station with dual charging modes. *IEEE Transactions on Intelligent Transportation Systems* 20, 9 (2018), 3386–3396.

[17] Dimitrios Zorbas and Christos Douligeris. 2018. Computing optimal drone positions to wirelessly recharge IoT devices. In *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. 628–633. https://doi.org/10.1109/INFCOMW.2018.8406897