Multi-objective Evolutionary Algorithms for Solving the Electric Vehicle Charging Station Infrastructure Problem

Harith Elamin

May 2025

Contents

List of Figures

Abstract

The growing awareness among people, including governments and individuals, of the risks posed by the pollution of gasoline and diesel vehicle emissions has contributed significantly to the shift toward clean energy for both public and private transport. Efficient and accessible transportation is a critical challenge that affects urban planners and corporate decision makers. Significant investments in technology and infrastructure are necessary to ensure that sufficient charging stations are available in strategic locations. This is crucial to ensure a successful transition to sustainable electric transport.

This thesis explores the use of the multi-objective evolutionary algorithm to optimize the placement and configuration of electric vehicle charging stations (EVCS). The goal is to find a solution that maximizes the geographic area covered by EV charging stations, minimizes the cost of setting up the infrastructure, and maximizes the power level of the stations to improve charging efficiency and reduce waiting times. By applying MOOP, such as the NSGA-II algorithm, we demonstrate the potential of evolutionary algorithms to address the complexities of EV charging station layout in an urban environment. The results show that NSGA-II provides an efficient and versatile solution to the challenges of the electric vehicle charging station infrastructure.

Chapter 1

Introduction

The rapid expansion of electric vehicle adoption presents both an opportunity and a challenge for urban transport systems. Electric vehicles have the potential to reduce air pollution and the use of conventional fuels such as diesel and gasoline, but their widespread use requires the establishment of a reliable and accessible charging infrastructure.

Selecting the location of electric vehicle charging stations is a complex problem that involves balancing several conflicting objectives. As electric vehicle adoption continues to increase worldwide, establishing reliable, efficient, and accessible infrastructure is becoming increasingly important. A key challenge in this process is to identify the optimal locations for charging stations to meet the needs of electric vehicle users while minimizing associated costs and covering as much area as possible to ensure connectivity.

Minimizing overall infrastructure costs an important goal when selecting the location of charging stations. This includes both direct costs, such as installing and maintaining charging stations, and the costs of users accessing those stations. Reducing costs is a critical aspect of urban planning, so it is essential to ensure that charging stations are strategically distributed to keep costs low while still meeting demand. Efficient distribution plays a key role in balancing affordability and user accessibility.

In addition to cost considerations, another important factor in determining the location of EV charging stations is minimizing the travel distance for EV users. The availability of charging stations directly affects the possibility of using EVs. Stations should be strategically placed, whether in densely populated urban areas or rural areas where access to charging infrastructure may be limited. Reducing the distance that drivers have to travel to find a charging point is crucial as it enhances the desire of drivers to use EVs. Conversely, long waits at charging stations or difficulty finding a nearby station can be a reason for drivers to avoid using EVs altogether.

Furthermore, the power level of the chargers should be considered to minimize the charging time. By optimizing the charging power of each station, the overall efficiency of the network can be improved, resulting in faster charging times and a better user experience. This consideration is crucial in ensuring that the charging infrastructure meets the growing demand for electric vehicles while minimizing user waiting times.

Moreover, traditional approaches to solving the EVCS location problem typically rely on optimization techniques such as mathematical programming. This approach often focuses on single-objective optimization, where only one aspect of the problem is prioritized, such as minimizing cost. Although these methods can provide effective solutions in certain scenarios, they often fail when dealing with trade-offs between multiple objectives, such as cost, coverage, and load balancing. In addition, traditional optimization techniques can be computationally expensive and time-consuming, especially when the problem involves large urban areas with many potential charging station locations. Multi-objective optimization techniques, such as multi-objective evolutionary algorithms (MOEAs), have emerged as a promising alternative to address the complexity of the EVCS problem. These algorithms are designed to find a variety of solutions that represent different trade-offs between competing objectives. By incorporating multiple objectives into the optimization process, multi-objective evolutionary algorithms can provide decision makers with a broader set of possible solutions, allowing them to choose the solution that best meets the needs of society while balancing the different costs and benefits of charging station placement. These methods are particularly useful when dealing with real-world problems that require the simultaneous optimization of multiple, often conflicting, objectives. In short, locating electric vehicle charging stations is a complex task that involves addressing different objectives, such as maximizing coverage, and maximizing the power of the charger to reduce waiting time. Traditional methods may not be sufficient to handle competing objectives, making multi-objective evolutionary algorithms an attractive solution for optimizing EV charging infrastructure placement.

1.1 Motivation

The transition to electric vehicles is a key component of sustainable transportation systems, addressing growing concerns about air pollution, climate change, and reliance on conventional fuels such as gasoline and diesel. As EV adoption continues to rise, one of the most pressing challenges is developing an efficient, widely available, and accessible electric vehicle charging station (EVCS) infrastructure. The success of this transition depends largely on optimizing the placement, configuration, and capacity of charging stations to meet the needs of both private and public transportation. However, this task is complicated by the multiple conflicting objectives that must be balanced, such as minimizing costs, maximizing coverage, and minimizing charging time.

Traditional optimization methods are often inadequate to solve such multidimensional problems. This is where multi-objective evolutionary algorithms (MOEAs), such as NSGA-II, come in. These algorithms are well-suited to handling the complexities of EVCS infrastructure optimization due to their ability to search for solutions that simultaneously meet multiple objectives.

The motivation behind this thesis stems from the need to explore advanced optimization techniques such as MOEAs to address the pressing challenges faced by urban planners and decision makers in the field of electric vehicle charging infrastructure. As the number of electric vehicles grows, the need for a strong charging network increases. The placement and configuration of electric vehicle charging stations must ensure adequate coverage across urban areas, while minimizing infrastructure costs.

Furthermore, this research aims to demonstrate how evolutionary algorithms, especially NSGA-II, can provide efficient and scalable solutions to these challenges, contributing to the creation of a sustainable and efficient electric vehicle charging infrastructure.

1.2 Problem Definition

The rapid trend of governments to push for electric vehicles has created an urgent need for a strong and widespread electric vehicle charging infrastructure. Ensuring that electric vehicle charging stations are optimally distributed across geographies with high capacity and cost-effectiveness to meet the needs of electric vehicle users is a complex challenge. The problem is complex, involving a number of objectives, such as increasing coverage area, minimizing costs, and reducing waiting time. This research focuses on leveraging multi-objective evolutionary algorithms to address these objectives and find optimal solutions for the deployment of electric vehicle charging infrastructure. The problem can be formalized as follows:

- Coverage: The goal is to increase the geographical area covered by electric vehicle charging stations, as the good distribution of electric vehicle charging stations allows electric vehicle owners to easily access these stations, which reduces time and effort.

- Chargers Power Level: Charger power level: The efficiency of charging stations plays an important role in charging electric vehicles, as it affects the speed of the charging process. Charging stations with high capacity reduce charging time, which leads to less waiting time for users. In addition, reducing charging time enhances the user experience and increases the efficiency of the network in general. It is necessary to provide charging stations capable of handling high power requirements to meet the growing needs of electric vehicles.

- Reducing the number of stations: Reducing the number of charging stations without compromising network coverage is critical to reducing costs and improving operational efficiency. Balancing station location with network coverage ensures adequate coverage of the service area while minimizing infrastructure costs.

- Reducing the number of chargers: In addition to reducing the number of charging stations, it is also necessary to reduce the number of individual chargers at each station. This helps reduce initial investment and operating costs while maintaining service levels and user convenience.

- Cost: Reducing the cost of building electric charging stations is of great importance to decision makers. The balance between comprehensive coverage and affordability is a key element of the optimization problem, but deploying a large number of charging stations across a large area can be very expensive.

1.3 Objective of the Thesis

This thesis aims to:

- Investigating multi-objective evolution algorithms (MOEAs) to solve the EVCS infrastructure problem.
- Optimizing multiple objectives at the same time, including infrastructure cost, coverage, and power levels per charging stand.
- Comparing the results obtained from the application of the genetic algorithm with those found in the literature review.
- Presenting the results from the research thesis experiments and analyzing their performance.

Chapter 2

Background

This chapter provides an overview of the basic concepts needed to understand the problem of optimizing electric vehicle charging station infrastructure. It discusses electric vehicles and their charging needs, multi-objective optimization (MOO), evolutionary algorithms (EAs), and multi-objective evolutionary algorithms (MOEAs), and specifically the NSGA-II algorithm, which is a central method in this study.

2.1 Electric Vehicles and Charging Infrastructure

The use of electric vehicles has increased in recent years, driven by growing concerns about climate change. Government policies promoting sustainable transportation have significantly fueled this growth.

In addition, advances in battery technology have played a key role in supporting this shift, as they have become more affordable and accessible. All of these factors have significantly contributed to accelerating the adoption of electric vehicles.

Moreover, gasoline and diesel vehicles can be developed to handle large amounts of fuel at a reasonable cost and can be refueled quickly and easily. However, despite significant advances in battery technology, electric vehicles face the challenge of increasing their battery capacity due to their high cost, making a reliable and accessible charging network important.

There are generally three levels of EV chargers:

- **AC Level 1 Charging:** Level 1 charging is the most basic method to charge an electric vehicle (EV). It uses a standard 120-volt household outlet. This charger level is often used when no other higher voltage charging options are available. Although this type of charger is very slow, it is still practical for many drivers, especially those who drive short distances every day. Level 1 charging is simple and requires no installation. Most electric cars come with a Level 1 charging cable that can be plugged directly into a household electrical outlet. Moreover, Level 1 charging meets simple daily needs. For example, charging an electric car for 8 hours increases its driving range to approximately 64 kilometers (40 miles). This is sufficient for normal daily driving. Overall, Level 1 charging may not be ideal for long-distance travel, but for many EV owners, it offers a cost effective and easy way to charge at home without needing extra equipment**?**.

- **AC Level 2 Charging:** Level 2 charging provides electric vehicle (EV) charging through a 240 volt electrical supply in residential settings, or 208 volts in commercial environments**?**. Unlike Level 1 charging, which uses a standard 120V outlet and charges slowly, Level 2 significantly reduces charging time, making it a popular choice for both home and public use. Level 2 charging is much faster than Level 1, using a 220V outlet instead of the standard 120V**?**. This significantly reduces charging time, making it a popular choice for both home and public use. One of the main benefits of Level 2 charging is its ability to fully charge a typical EV battery overnight. This makes it ideal for daily use.

  However, Level 2 equipment is also widely used at public stations, workplaces, and shopping centers to support EV users. Overall, Level 2 charging offers a good balance between speed and efficiency, making it the preferred choice for many EV users.

- **DC Fast Charging:** Level 3 charging operates at around 400 volts, allowing electric vehicles (EVs) to charge significantly faster than with standard Level 1 or Level 2 charging**?**.

  This technology provide up to 500 kilowatts (kW) of power, making it ideal for busy highways and transportation routes where fast charging is crucial**?**. In addition, DC fast charging stations are especially useful for long-distance travel and commercial applications, as they can recharge an EV battery to 80% in as little as 20–30 minutes, depending on the vehicle and charger capacity**?**.

The growing adoption of medium and heavy duty electric vehicles (EVs) such as electric buses, delivery vans, and heavy trucks increased demand for DC fast charging infrastructure. These vehicles have larger batteries, which requiring higher charging capacities.

Overall, DC fast charging plays a critical role in supporting the widespread adoption of EVs across personal, commercial, and public transportation sectors.

Figure 2.1 shows the distribution of charger levels in use in the United States**?**.

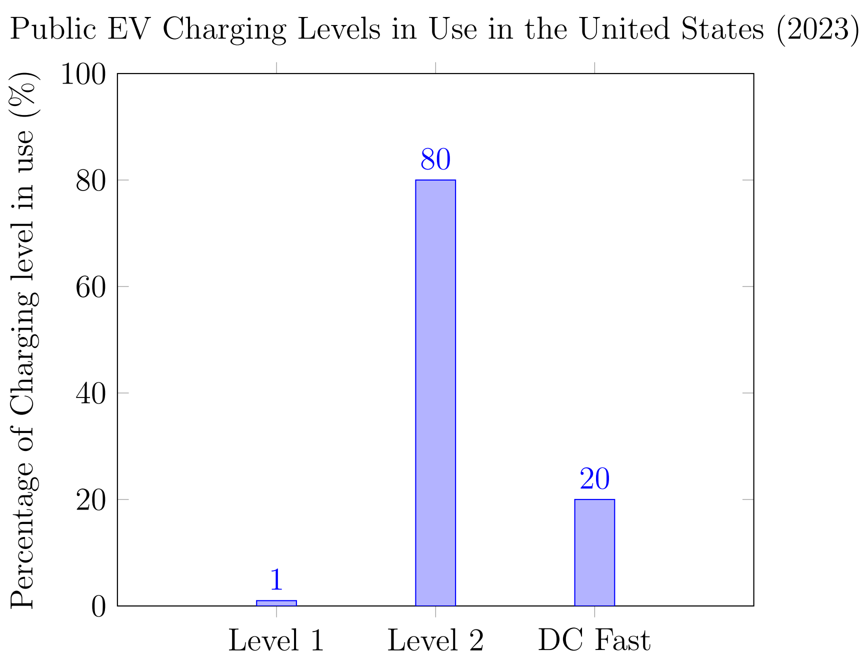

Figure 2.1: Distribution of EV Charging Port Types in the United States (2023)

Connectors for electric vehicle (EV) charging vary depending on the region and charging level. Generally, the standard SAE J1772 connector is widely used for both Level 1 and Level 2 charging. For Level 3, or DC fast charging, the most commonly used connectors are the Combined Charging System (CCS) and CHAdeMO**?**.

Moreover, Tesla, as a major player in the EV industry, uses a proprietary connector for its vehicles. However, it also provides compatibility with the CCS connector through the use of adapters**?**.

Figure 2.2 illustrates the types of connectors used for Levels 1, 2, and 3.

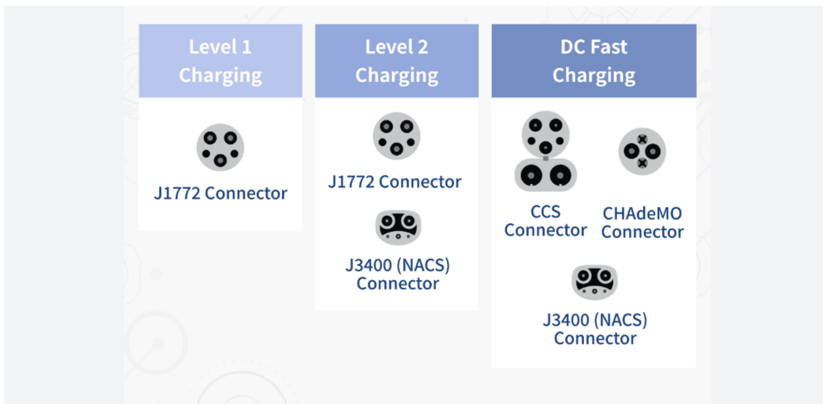

Figure 2.2: Types of EV connectors used for Levels 1, 2, and 3 **?**

Figure 2.3 illustrates the relationship between average charging speed and charging time for the three levels of electric vehicle (EV) charging based on data from the U.S. Department of Transportation **?**. Level 1 charging has the longest average charging time of approximately 65 hours, and delivers the lowest charging speed of around 5 miles per hour. In contrast, Level 2 charging reduces the charging time to an average of 5.5 hours and increases the average charging speed to 46 miles per hour. Level 3 charging, or DC fast charging, offers the highest efficiency with an average charging time of just 0.75 hours and a substantial charging speed of approximately 638 miles per hour.

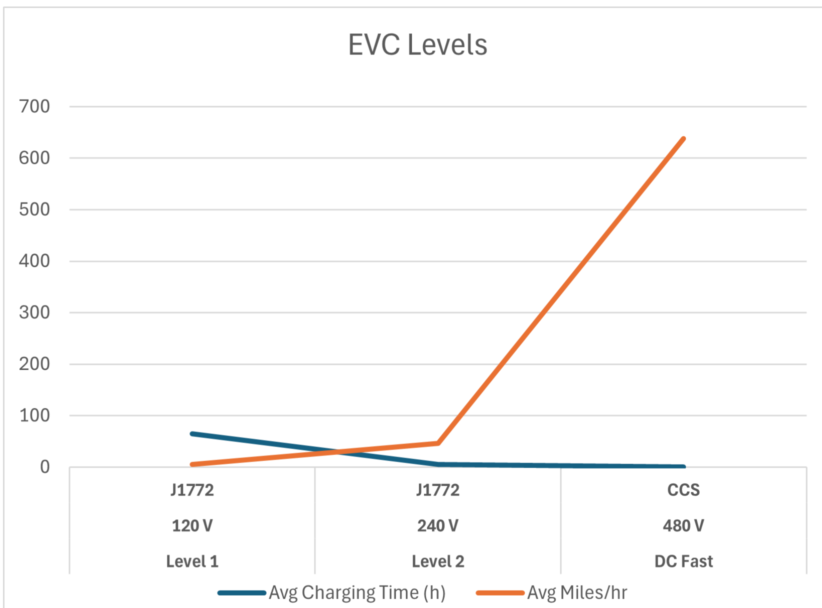

Figure 2.3: Average charging speed versus charging time for different EV charging levels

This clear trend highlights the performance differences between charging levels, with faster charging options offering significantly higher power output and reduced wait times. These differences are crucial for consumers and EV charging station operators when choosing the right charging infrastructure to meet their specific operational needs. This clearly contributes to supporting the spread and convenience of EV.

2.2 Multi-Objective Optimization (MOO)

Multi Objective Optimization (MOO) seeks to optimize multiple conflicting objectives simultaneously. Unlike single-objective optimization, which produces one best solution, MOO generates a set of Pareto optimal solutions**?**.

In addition, these solutions represent trade-offs, where improving one objective results in worsening another. The set of all Pareto optimal solutions forms the Pareto front, which shows the best possible trade-offs between the objectives. Each point on this front represents a solution where no objective can be improved without degrading another. MOO is commonly used in complex decision-making problems where multiple criteria must be considered. It provides valuable insights into the balance between competing objectives.**?**.

Regarding **?** The general expression of a multi objective optimization (MOO) problem:

$$\text{Minimize } \mathbf{F}(x) = [f_1(x), f_2(x), \ldots, f_k(x)], \quad \text{subject to } x \in \Omega$$

where

- $\mathbf{F}(x)$: Represents the vector of k objective functions. Each $f_i(x)$ corresponds to a distinct goal or objective, and the decision vector x influences all of them.

- Ω: Refers to the feasible set of decision variables, which defines the constraints the decision variables must satisfy.

- **Minimization**: The aim is typically to minimize each objective function, although maximization is also common. In many cases, improving one objective could negatively affect another, creating trade-offs that are a key challenge in multi-objective optimization.

MOO is used across many domains:

- **Engineering**: Balancing trade-offs among cost, weight, and performance. - **Energy Systems**: Optimizing efficiency, cost, and environmental impact **?**. - **Finance**: Portfolio optimization involving return and risk. - **Machine Learning**: Hyperparameter tuning involving accuracy, complexity, and fairness **?**.

2.2.1 Trade-offs in Multi-Objective Optimization: The Role of Pareto Optimality

In multi-objective optimization (MOO), it is often impossible to identify a single solution that optimally satisfies all objectives simultaneously, especially when the objectives conflict **?**. Instead, the goal is to find a set of solutions that represent acceptable trade-offs. A solution is considered *Pareto optimal* if there is no other feasible solution that can improve at least one objective without worsening another. According the principle of MOO, Pareto optimality plays a crucial role in evaluating the quality of a solution amidst conflicting objectives **?**.

The collection of all Pareto optimal solutions is known as the *Pareto front*. Each point on this front represents a non-dominated solution, meaning that no other solution in the feasible space is strictly better in all objectives. As such, the Pareto front provides decision-makers with a spectrum of choices, each embodying different trade-offs among the objectives. This allows for flexibility in selecting solutions that align with specific priorities or operational constraints **?**.

In addition, identifying the Pareto front supports more informed decision-making, particularly in complex systems where multiple performance metrics must be balanced. For example, in engineering design, a solution may involve trade-offs between cost, efficiency, and robustness. In such scenarios, a decision-maker might prioritize cost-effectiveness while ensuring that other performance criteria remain within acceptable limits **?**.

However, selecting a single solution from the Pareto front is not trivial. This process typically involves incorporating decision-maker preferences, which may be explicit (e.g., assigning weights to objectives) or implicit (e.g., using interactive methods). Furthermore, these preferences may change over time or remain uncertain, adding another layer of complexity to the optimization process **?**.

Understanding trade-offs among objectives and visualizing the Pareto front play a vital role in revealing interactions between decision variables, defining the limits of achievable performance, and supporting strategic decision-making **?**.

The shape of the Pareto front can impact the performance of optimization methods. Classical methods, like the weighted sum approach. In contrast, evolutionary algorithms are better at exploring different areas of the Pareto front because they use a population-based method and can handle complex, nonlinear problems **?**.

In summary, Pareto optimality provides a rigorous framework for analyzing and interpreting trade-offs in multi-objective problems. It serves as a decision support mechanism, helping stakeholders to navigate complex trade-offs and make choices that best align with their objectives and constraints.

2.2.2 Solution Methods

Several methods have been developed to solve Multi-Objective Optimization (MOO) problems, categorized into classical and evolutionary approaches. Both approaches provide valuable tools depending on the nature of the problem. Classical methods tend to be more straightforward, but may face limitations with complex, high-dimensional problems. Evolutionary algorithms, on the other hand, are well suited for handling such complexity and are the primary focus of this research.

Classical Methods: In classical Multi-Objective Optimization (MOO), scalarization techniques are commonly employed to simplify the problem by converting multiple objectives into a single objective function. One such technique is the weighted sum method, where each objective is assigned a weight and then combined into a single function. This method is computationally efficient and performs well when the Pareto front is smooth and regular. However, it may fail in cases where the Pareto front is non-smooth or discontinuous, as it might not capture all potential solutions. Another widely used classical method is the ϵ constraint method. In this approach, one objective is optimized, while the other objectives are constrained within specific bounds. Although this method ensures that all objectives are considered, it requires multiple runs with different constraint settings to fully explore the Pareto front. While classical methods are efficient for low-dimensional problems, they tend to struggle with high-dimensional, nonlinear, or complex objective landscapes **?**, **?**.

Evolutionary Algorithms (EAs):

Evolutionary Algorithms (EAs) are population-based, stochastic optimization techniques inspired by the principles of natural selection and genetics (**?**),**?**. They employ biologically inspired operations such as selection, crossover, and mutation to iteratively evolve a set of candidate solutions toward optimal or near-optimal trade-offs across multiple objectives.

The general structure of an EA consists of the following steps **?**:

1. **Initialization:** Generate an initial population of solutions randomly or heuristically.

2. **Evaluation:** Assess the fitness of each individual using the defined objective functions.

3. **Selection:** Choose high quality solutions based on fitness to serve as parents for the next generation.

4. **Variation:** Apply genetic operators such as crossover and mutation to generate new offspring.

5. **Replacement:** Form a new generation by selecting individuals from the current population and the offspring.

EAs are particularly effective for tackling complex, nonlinear, and high dimensional optimization problems where classical methods may struggle, especially when the Pareto front is irregular **?**. Its strength lies in its ability to maintain a diverse set of solutions, which allows it to explore different regions of the solution space simultaneously. This makes it particularly suitable for multi objective optimization problems.

Given these advantages, this research focuses on applying evolutionary algorithms as a fundamental framework for optimization, leveraging their flexibility and robustness in dealing with complex trade-offs in multi-objective problems.

2.3 Multi-Objective Evolutionary Algorithms (MOEAs)

Multi-objective evolutionary algorithms (MOEAs) are extensions of traditional evolutionary algorithms designed to address problems with multiple, often conflicting, objectives. Unlike single-objective evolutionary algorithms, MOEAs maintain a set of solutions, allowing multiple solutions to be explored simultaneously (**?**).

According to (**?**), the key features of MOEAs are as follows:

- **Diversity Preservation:** Mechanisms such as crowding distance and niching ensure a well-distributed set of solutions across the Pareto front.

- **Pareto Dominance:** Selection is based on dominance relations, where one solution dominates another if it is no worse in all objectives and better in at least one.
- **Archiving:** High-quality non-dominated solutions are stored in an external archive to preserve progress across generations.

MOEA models have been widely applied in various fields, including engineering design, logistics, supply chain optimization, telecommunications, and power systems planning, due to their ability to handle complex, nonlinear, and high-dimensional objective spaces **?**.

2.4 NSGA-II Algorithm

The Non-dominated Sorting Genetic Algorithm II (NSGA-II), proposed by Deb et al. **?**, is one of the most popular and efficient Multi-Objective Evolutionary Algorithms (MOEAs).

According **?** , NSGA-II introduces several key innovations as follows:

- **Fast Non-dominated Sorting:** This method efficiently classifies the population into different levels of non-domination, ensuring the diversity of solutions.
- **Crowding Distance:** Measures the density of solutions surrounding a particular solution, promoting diversity by maintaining a spread of solutions across the objective space.
- **Elitism:** Retains the best solutions of individuals across generations, ensuring that the quality of the population improves over time.

The algorithm can be summarized in the following steps **?**:

1. Combine the parent and offspring populations into one single population.
2. Rank the combined population by non-dominance, categorizing solutions into different levels.
3. Select the best individuals based on crowding distance, ensuring diversity is preserved, until the next generation is complete.

4. Apply variation operators (crossover and mutation) to generate offspring for the subsequent generation.

In addition, NSGA-II is computationally efficient, with a time complexity of $O(MN^2)$, where M is the number of objectives and N is the population size **?**. Its ability to balance convergence and diversity makes it highly suitable for solving complex multi objective problems, such as Electric Vehicle Charging Station (EVCS) optimization. This matches well with the objectives of this research, which are to maximize coverage, maximize charger speed to reduce waiting times, minimize the number of stations, minimize the number of chargers, and minimize the average distance to stations. NSGA-II explores a variety of possible charging solutions and converges toward Pareto-optimal solutions, effectively addressing the trade-offs between these conflicting objectives. By efficiently handling these goals, NSGA-II proves to be a powerful tool for optimizing the design of EVCS, Ensuring efficiency and coverage while reducing wait times and lowering overall costs.

Chapter 3

Literature Review

3.1 Multi-Objective Optimization of Electric Vehicle Charging Station Deployment Using Genetic Algorithms

Developing an efficient charging infrastructure is a critical research area, especially with the increasing use of electric vehicles. One of the most prominent studies in this area is a paper titled ”Multi-objective Optimization of Electric Vehicle Charging Station Deployment Using Genetic Algorithms.” This study focuses on optimizing the location and capacity of electric vehicle charging stations by treating the problem as a multi-objective optimization task **?**.

This study focuses on several key, often conflicting objectives: minimizing installation and operating costs, minimizing user inconvenience (such as travel distance to charging stations), maximizing service coverage, and minimizing the impact on the electricity grid. Balancing these objectives is essential to create a cost-effective and user-friendly charging network**?**.

To solve this complex problem, the authors use a genetic algorithm (GA), which is a method inspired by nature that helps to find good solutions by searching through many possibilities. In the proposed method, potential charging station configurations are encoded as chromosomes. The genetic algorithm then evolves these configurations over a series of generations using genetic processes such as selection and mutation. This process produces a Pareto front of optimal solutions, each representing a different trade-off between competing objectives**?**.

The results indicate that a genetic algorithm-based method can identify deployment strategies that perform better than traditional single-objective or heuristic approaches. Using multiple evaluation criteria, such as cost and accessibility, provides a comprehensive view of the solution and allows stakeholders to select the most appropriate plan according to their priorities**?**.

In general, this study makes a significant contribution to the field of electric vehicle infrastructure planning by providing a flexible, data-driven framework. It demonstrates that multi-objective optimization techniques, particularly those based on genetic algorithms, can play a vital role in supporting the sustainable development of electric vehicle charging networks. This approach serves as a foundation for further research and practical applications in smart city planning and energy management**?**.

Chapter 4

Methodology

4.1 Problem Formulation

The Electric Vehicle Charging Station(EVCS) infrastructure problem can be formulated as a multi-objective optimization problem with the following decision variables:

- A set of candidate charging station locations, indicated as $S = \{s_1, s_2, ..., s_n\}$, where each location s_i is defined by its geographic coordinates (x_i, y_i) within the urban area.
- The power level assigned to chargers at each station, denoted by $P_i \in \{11.5, 14.2, 19.2, 25, 60, 62, 80, 120, 150, 180, 200, 240, 250, 300, 325, 350, 400\}$ kW, represents the charging speed selected for station i to minimize user waiting time.
- The installation cost associated with each charger, denoted by C_i, represents the total economic expense of deploying a charger at station i, including infrastructure and equipment costs.
- The number of charging stations deployed, denoted by N_s, which impacts both installation cost, and the resource usage.
- The total number of chargers installed across all stations, denoted by N_c, which also contributes to overall cost and infrastructure requirements.

The constraints considered in the optimization model are as follows:

- The total number of charging stations (N_s) must not exceed a predefined upper limit or budget constraint, ensuring feasibility in terms of infrastructure cost and urban space availability.

- The total number of chargers installed at each station must respect the station's physical capacity and grid connection limits, and the aggregate charging demand must not exceed this capacity based on both the number of chargers and their power levels (P_i).

- Each electric vehicle (EV) must be within an acceptable distance from at least one charging station to ensure adequate spatial coverage and accessibility for all users.

- The total installation cost, calculated based on the number and type of chargers deployed (C_i), must remain within the available budget or funding constraints.

- Charger power levels must be selected from the discrete set $\{11.5, 14.2, 19.2, 25, 60, 62, 80, 120, 150, 180, 200, 240, 250, 300, 325, 350, 400\}$ kW, corresponding to standardized fast-charging options.

4.2 Multi-objective Evolutionary Algorithm (MOEA) Framework

We employ the **NSGA-II** algorithm, which is is a multi-objective evolutionary algorithm that applies non-dominated sorting to rank solutions into various fronts **?**.

Standard genetic operators such as selection, crossover, and mutation are employed within the algorithm. The fitness of each solution is assessed according to the four objectives previously defined: coverage, charger speed, stations number, and chargers number.

4.2.1 NSGA0-II Algorithm

Here we define NSGA-II xxxxxxxx xxx

Chapter 5

Experiment

As mentioned in Chapter 1, this research focuses on the strategic deployment of electric vehicle charging stations (EVCS) in urban environments. This chapter presents the experimental design, implementation, and analysis of a multi-objective optimization problem (MOOP) to solve the Electric Vehicle Charging Stations problem using the non-dominated genetic sorting algorithm II (NSGA-II). The goal is to optimize the locations and configuration of EVCS.

The problem is modeled with four conflicting objectives: (1) maximizing coverage , (2) maximizing charger speed, (3) minimizing the number of stations, and (4) minimizing the total number of chargers per station. These objectives often conflict with one another; for example, increasing coverage and charger speed usually requires deploying more chargers and stations, which conflicts with the goal of minimizing infrastructure.

NSGA-II was chosen for its proven effectiveness in tackling complex, multi-objective, nonlinear optimization problems with discrete decision variables. This algorithm is well suited for problems that require a variety of balanced solutions for decision-making.

5.1 Experimental Environment

The algorithm was implemented using the DEAP framework in Python. Experiments were run on a system with the following specifications:

- OS: Windows 10

- Processor: Intel(R) Core(TM) i7-10610U CPU @ 1.80GHz 2.30 GHz
- RAM: 32 GB
- Python Version: 3.10
- Libraries: NumPy, DEAP, Matplotlib, Pandas

The demand data used is extracted from publicly available electric vehicle charging station data models, ensuring realistic modeling of public coverage needs.

5.2 Dataset

5.2.1 Stations dataset

In this research on electric vehicle charging stations (EVCS), we selected a California data set due to its leading role in the adoption of electric vehicles and the development of its charging infrastructure. In addition, the installation costs for the chargers were sourced from standard prices used across the United States.

California is among the regions that have implemented EV policies to reduce pollution. Furthermore, California boasts a high population density and a large number of electric vehicles and EV charging stations, providing a rich source of data on usage, location, and other key factors influencing the development of charging stations. This data enhances the potential for EV charging stations to be deployed in other regions.

In this experiment, we evaluate the performance of the NSGA-II algorithm. The test case is based on one of the California dataset, with the following details:

- 100 potential locations for charging stations, each location has number of charger, with 1000 EVs in the area. The dataset includes the following attributes for each location:
  - **Id**: charging station number.
  - **Coordinates**: The geographical coordinates of each charging station location.

- **Charger speed (kW)**: The chargers speed power for each charger, (11.5, 14.2, 19.2, 25, 60, 62, 80, 120, 150, 180, 200, 240, 250, 300, 325, 350, 400).
- **Chargers per station**: The number of chargers installed at each station.

5.2.2 station cost dataset

Power Range	**Type of Charger**	**Hardware Cost**	**Installation Cost**	**Total Estimated Cost**
3 kW - 7 kW	Level 1 or Level 2 Charger	$500 - $1,500	$300 - $500	$800 - $2,000
7 kW - 22 kW	Level 2 Charger	$1,500 - $5,000	$1,000 - $2,500	$2,500 - $7,500
50 kW - 100 kW	DC Fast Charger	$30,000 - $50,000	$50,000 - $100,000	$80,000 - $150,000
100 kW - 150 kW	DC Fast Charger	$50,000 - $70,000	$50,000 - $100,000	$100,000 - $170,000
150 kW - 200 kW	DC Fast Charger	$70,000 - $100,000	$100,000 - $150,000	$170,000 - $250,000
200 kW - 350 kW	Ultra-Fast Charger	$100,000 - $150,000	$150,000 - $250,000	$250,000 - $400,000
350 kW - 400 kW	Ultra-Fast Charger	$120,000 - $150,000	$150,000 - $250,000	$270,000 - $400,000

Table 5.1: Cost Estimates for Different Types of EV Chargers

5.2.3 Decision Variables

The optimization problem considers a set of locations (stations), where each station can use a number of electric vehicle chargers (EVCs) with a specified charging speed. The decision variables include:

- $x_i \in \{0, 1\}$: Binary variable indicating whether a station is deployed at location i.
- $c_i \in \{0, 1, 2, ..., C_{max}\}$: Integer variable indicating the number of chargers at location i.

- $c_i \in \{0, 1, 2, ..., C_{max}\}$: Integer variable indicating the number of station i.
- $s_i \in \{s_1, s_2, ..., s_k\}$: Discrete variable indicating the speed of chargers at Station i.

5.2.4 Objective Functions

1. **Maximize Coverage (f_1)**: Coverage is defined as the percentage of demand points (e.g., traffic or residential clusters) that fall within the effective range of any installed station.

$$f_1 = \frac{\sum_{j=1}^{m} \delta_j}{m}, \quad \delta_j = \begin{cases} 1 & \text{if demand point } j \text{ is covered by any station} \\ 0 & \text{otherwise} \end{cases}$$

2. **Maximize Charger Speed (f_2)**: Charger speed is value of the installed charger speed, where higher-speed chargers receive a higher weight.

$$f_2 = \frac{\sum_{i=1}^{n} x_i \cdot c_i \cdot s_i}{\sum_{i=1}^{n} x_i \cdot c_i}$$

3. **Minimize Number of Stations (f_3)**:

$$f_3 = \sum_{i=1}^{n} x_i$$

4. **Minimize Number of Chargers (f_4)**:

$$f_4 = \sum_{i=1}^{n} c_i$$

5.2.5 Constraints

The optimization is subject to the following constraints:

- **Budget Constraint**:

$$\sum_{i=1}^{n} (x_i \cdot F + c_i \cdot V(s_i)) \leq B$$

  where F is the cost per station, $V(s_i)$ is the cost of a charger with speed s_i, and B is the total budget.

- **Location Feasibility**: Only pre-approved location can be selected, based on land availability and zoning regulations.
- **Station Capacity** : Each station has a maximum number of chargers it can accommodate.
- **Redundancy Constraint**: Ensure no stations are placed without covering demand.

5.3 NSGA-II Algorithm Setup

5.3.1 Overview

As discussed in Chapter 3, NSGA-II is an evolutionary algorithm that maintains a population of candidate solutions. In each generation, new offspring are generated through crossover and mutation. The next generation is then selected based on Pareto dominance and a crowding distance metric, which together promote both optimality and diversity within the population.

5.3.2 Parameters

- Population Size: 100
- Number of Generations: 50
- μ (Number of Individuals Selected for the Next Generation): 50
- λ (Number of Offspring Generated Each Generation): 100
- Crossover Probability (*cxpb*): 0.7
- Mutation Probability (*mutpb*): 0.2
- Selection Method: NSGA-II (Non-dominated Sorting Genetic Algorithm II)
- Crossover Operator: Two-Point Crossover
- Mutation Operator: Shuffle Index Mutation

- Possible power speed : [11.5, 14.2, 19.2, 25, 60, 62, 80, 120, 150, 180, 200, 240, 250, 300, 325, 350, 400]
- The dataset is described in the Dataset section(Stations dataset).

5.3.3 Encoding Strategy

Each individual in the population is encoded as a tuple:

$$\textbf{Chromosome} = [(x_1, c_1, s_1), (x_2, c_2, s_2), ..., (x_n, c_n, s_n)]$$

This representation allows flexible adjustment of whether a station is deployed, how many chargers it has, and their speed types.

5.3.4 Steps of the Applied NSGA-II Algorithm

Step 1: Preprocessing the Data

In this experiment, The stations data was imported using the OpenChargeMap API **?** through the `opencchargemap` fetcher. The data was filtered to include only stations from the United States ("country code": "US") and from California ("state": "California")As mentioned in the dataset section. After filtering, we selected the important columns for this expierenment, which were: "station_id", "latitude", "longitude", "number_of_points", and "power_kw". We then converted the "number_of_points" and "power_kw" columns into numeric values so that they could be processed correctly during optimization process. Some column names have been updated to make the data more easier.

In addition, since one of our objectives relates to coverage, it was necessary to calculate the coverage for each selected station. To achieve this, we implemented the `calculate_coverage` method. This method computes the total coverage based on the distances between stations. We then used the `add_coverage` method to assign these calculated coverage values to the stations dataset.

In the end, As shown in Figure **??**the data was cleaned, formatted,to use as input of the NSGA-II algorithm used in this study.

```
station_id,latitude,longitude,num_chargers,charger_speed,coverage
311709,43.01966188077989,-83.32169477891497,2.0,125.0,153388.78472706157
311613,41.41619830139183,-84.10499089014816,4.0,350.0,146779.17610872354
311587,42.18464893920972,-83.74172601109308,2.0,60.0,149458.6022384466
311586,42.5400152934009,-83.78877648936871,12.0,250.0,150542.73753766733
311585,42.42733150000629,-82.90891833438627,6.0,350.0,152103.99701625452
311581,30.18393867398157,-81.62541156086436,8.0,350.0,169327.78998358388
311579,33.47922772277374,-111.98561491929566,1.0,350.0,187809.91803861147
311578,39.05586437918734,-104.85310571754088,1.0,350.0,160573.63742879446
311577,35.4808394197364,-108.433538293779,1.0,350.0,172423.2556040314
311576,38.41137461248874,-96.2366737725612,1.0,350.0,144640.16190935837
311575,35.49790189661056,-80.56150457086042,4.0,350.0,153312.8141417414З
311574,37.18401357570761,-77.3180918831336,1.0,350.0,166785.22356610224
311573,41.41679625863407,-84.10496337251186,1.0,350.0,146780.55381444635
```

Figure 5.1: Stations Dataset

Step 2: Define the Multi-Objective Optimization Problem

In this step, the electric vehicle charging station (EVCS) problem was defined as a multi-objective optimization problem with four conflicting goals. These objectives are: (1) maximizing geographical coverage, ensuring that as many areas as possible are served by at least one station; (2) maximizing charger speed, which helps reduce the waiting time for electric vehicle (EV) users; (3) minimizing the number of installed charging stations, which contributes to reducing the overall infrastructure cost; and (4) minimizing the total number of chargers across all selected stations to further manage costs and operational complexity.

To implement this formulation, we used the DEAP framework**?**. The problem was modeled using a custom fitness class where the objectives were assigned specific weights: positive weights (1.0) for maximizing coverage, positive weights (1.0) for maximizing charger speed, negative weights (-1.0) for minimizing the number of stations and negative weights (-1.0) for minimizing the number of chargers. This formulation allows the algorithm to simultaneously balance service quality and cost efficiency. Each individual in the population represents a unique combination of selected charging stations, and the fitness evaluation returns values to each of the four objectives.

Step 2: Define, and initialize DEAP Components

In this step, the DEAP framework is utilized to define the requirement components for the evolutionary algorithm, such as the fitness function, individual structure, and algorithm settings. These settings include population size, selection strategy, and the methods for crossover and mutation. To initialize these components, we first create a toolbox using DEAP's base.Toolbox. The toolbox is then populated with several registered functions that define the operations for the evolutionary process.

In additon, the individual function is registered to generate an individual from a specified structure using the tools.initIterate method. The population function initializes the population by creating multiple individuals through tools.initRepeat. The mate function, which controls the crossover operation, is defined with the cxTwoPointCheck method. Similarly, the mutation operation is defined using mutShuffleIndexesCheck with a mutation probability. For selection, the selNSGA2 method is registered, which ensures that the selection process follows the NSGA-II strategy for multi-objective optimization.

Howerver, the evaluate function is linked to the toolbox to evaluate individuals based on their fitness values.

By registering these functions, the necessary components of the evolutionary process are prepared, allowing the algorithm to run efficiently.

Step 3: Define Objective Functions

In this step, we define the four objective functions, as mentioned in the 'Objective Functions' subsection above, which will guide the evolutionary algorithm towards finding the best solution:

- **Objective 1: Maximize Coverage**
  The `calculate the coverage` function. This function calculates the total coverage between stations by calculating the distance between them.

$$\text{Coverage} = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} d(s_i, s_j) \tag{5.1}$$

  Where:

- n is the total number of selected stations,
- s_i and s_j are the coordinates (latitude and longitude) of the selected station,
- $d(s_i, s_j)$ is the geographical distance between stations,

- **Objective 2: Maximize Charger Speed**
  Calculate charger speed function which identifies the maximum charger speed from the selected stations. This objective ensures the improvement of the overall efficiency of the electric vehicle charging stations.

  $$\text{Charger Speed} = \max\left(\text{charger\_speed}(s_i) \mid s_i \in \text{stations}\right) \tag{5.2}$$

  where:

  - s_i represents the i-th selected station,
  - $\text{charger\_speed}(s_i)$ is the charging speed of station s_i,
  - The function returns the maximum charging speed from the selected stations.

- **Objective 3: Minimize the Number of Stations**
  The `calculate the number of stations` function counts the number of selected stations. Minimizing the stations number helps reduce infrastructure costs while ensuring good service coverage.

  $$\text{Number of Stations} = |S| \tag{5.3}$$

  Where:

  - S represents the set of selected stations,
  - $|S|$ denotes the cardinality (or size) of the set S, which is the total number of stations in the selection.

- **Objective 4: Minimize the Number of Chargers** This function calculates the number of chargers sums the total number of chargers across the selected stations. This helps minimize the cost by deploying only the necessary infrastructure.

These algorithmic objectives help achieve a good balance between coverage and efficiency by taking into account the distance between stations, charging speed, the number of stations and chargers. This ultimately leads to optimized solutions that meet the charging network's needs with reducing the final cost.

Step 4: Define the evaluation function that combines all objectives

The evaluation function plays a critical role in guiding the evolutionary algorithm by assessing the quality of each solution based on multiple objectives**?**. In this step, we define a function that combines all four previously established objectives to evaluate each individual in the stations dataset in the population. The individual represents a selection of electric vehicle charging stations.

The evaluation function calls four separate methods, as discussed in Subsection 3 in this chapter:

1. **Coverage**: Calculated using the `calculate_coverage` method, it sums the distances between all unique pairs of selected stations to check station placement across the area.

2. **Charger Speed**: Determined using the `calculate_charger_speed` method, this objective selects the highest available charging speed among the chosen stations to enhance charging efficiency.

3. **Number of Stations**: Computed with `calculate_num_stations`, this counts the selected stations, encouraging minimal infrastructure to reduce overall costs.

4. **Number of Chargers**: Using `calculate_num_chargers`, this sums all chargers at selected stations to further minimize overall costs.

The function returns a tuple containing these four values. These outputs are then used by the NSGA-II algorithm within DEAP to rank and evolve the population toward optimal solutions that balance all objectives effectively**?**.

Step 5: Create a random individual with fewer stations selected

The create_individual function is responsible for generating a random initial solution, or "individual," for the evolutionary algorithm. Each individual represents a subset of electric vehicle charging stations selected from the dataset.

In this function, a random number of stations is selected, in range from one to half of the total available stations. This approach creates diverse solutions some with fewer stations and others with more. This allowing the algorithm to explore a wide range of possible solution during optimization.

To create the individual, the function uses Python's random.sample() to select a unique set of station indices without replacement. This avoids selecting the same station more than once and keeps each solution valid. The result is a list of station numbers that can be checked using the objective functions.

Generating a diverse population of individuals is essential in evolutionary algorithms such as NSGA-II**??**. This approach helps the algorithm explore more options, and avoid getting stuck in poor solutions too early. The function keeps the population diverse by randomly changing how many stations are picked and which ones are included. These randomly created station groups form the starting point for the algorithm to begin finding the best charging station solution.

Step 6: Apply Genetic Operators and Evolve Population

In this step, the algorithm improves the population by applying genetic operators: selection, crossover, and mutation. NSGA-II is used for selection, favoring individuals that balance multiple objectives while maintaining diversity**?**.

The crossover process is handled by a two point method (cxTwoPointCheck), which swaps segments between two parent individuals to generate offspring**?**. A check ensures both parents have at least two elements before applying the crossover, preserving solution validity**?**. In addition, the mutation is handled by mutShuffleIndexesCheck, which introduces small random changes to individuals by occasionally removing a station. This reduces the number of stations, promoting simpler, more efficient solutions. It also helps prevent duplicate values and ensures that each individual remains a valid configuration**?**. These operations are repeated over generations, allowing the population to explore new combinations, avoid premature convergence, and evolve toward optimal**?** electric vehicle charging station solution.

5.4 Performance Metrics

- **1. Pareto Front Coverage**: Shows how well the results match the ideal set of solutions.
- **2. Convergence and Diversity**: Checks how close the solutions are to the ideal front and how spread out they are.
- **3. Area Coverage**: Assesses how well the selected stations cover the region to reduce travel distance for EV users.
- **5. Charger Cost**: Estimates the total cost of installing and running the chargers.

5.4.1 Pareto Front Visualization

Figure **??** shows the final Pareto front obtained after 100 generations. The front highlights the trade-offs between infrastructure cost and performance metrics. However, solutions that prioritize higher coverage and charger speed generally require more stations and chargers. This is consistent with the objectives of maximizing coverage and charger speed, while minimizing the number of stations and chargers. As expected, the solutions provide high coverage and charger speed, while minimizing the number of stations and chargers.

5.4.2 Trade-off Analysis

Table **??** presents four representative solutions from the Pareto front, each highlighting distinct trade-off configurations. These solutions illustrate how different balances between the objectives (coverage, charger speed, number of stations, and number of chargers) affect the overall performance.

In addition, by analyzing these solutions, we can better understand how adjusting one objective impacts the others, helping to identify optimal configurations for the charging station network. This provides valuable into the relationship between the competing objectives and their influence on the final solution.

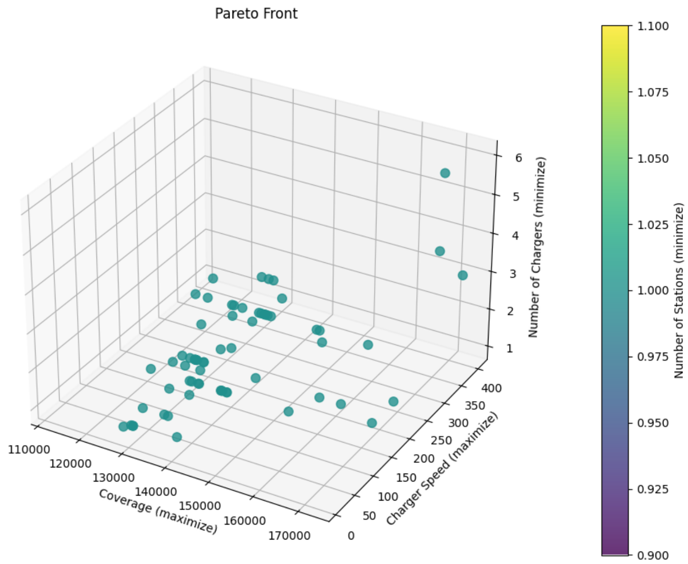

Figure 5.2: Pareto Front from NSGA-II Optimization

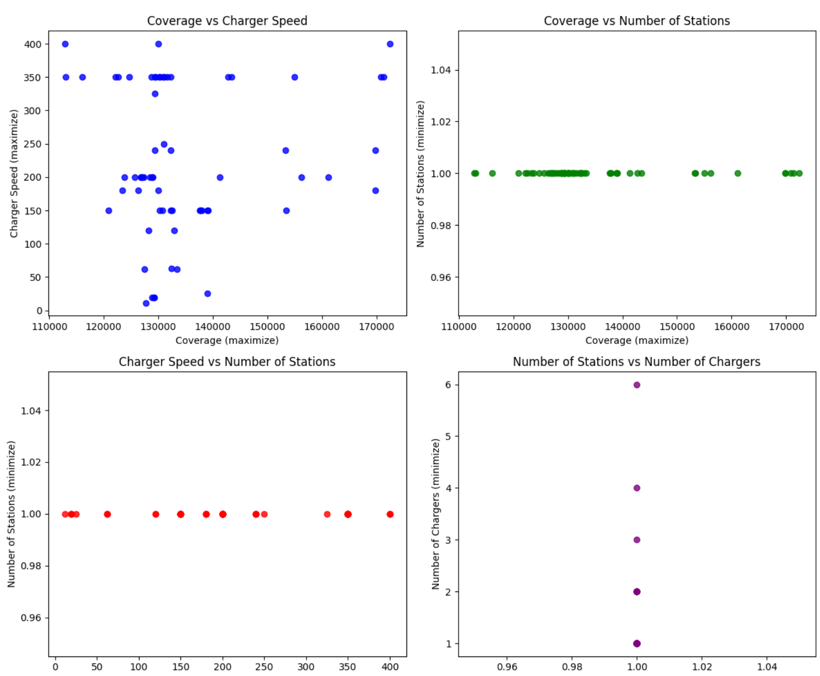

Figure 5.3: Trade-Off from NSGA-II Optimization

Chapter 6

Results and Discussion

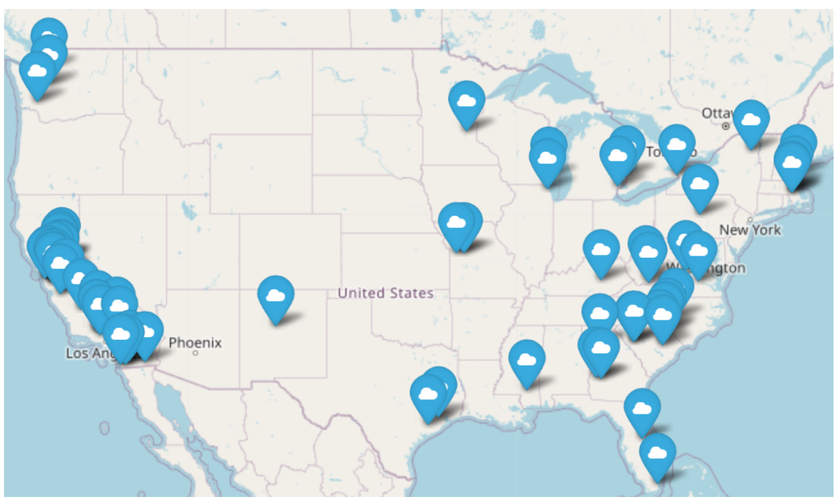

Figure 6.1: Optimized stations map.

Chapter 7

Conclusion

In the original dataset, there were 92 stations and 91 chargers, each with different power speeds $[14.2, 11.5, 19.2, 25, 60, 62, 80, 120, 150, 180, 200, 240, 250, 300, 325, 350, 400]$. After applying the NSGA-II algorithm, the following objectives were addressed:

1. Maximize coverage.
2. Maximize the speed of the charger to reduce the waiting time.
3. Minimize the number of stations.
4. Minimize the number of chargers.

The aim was to develop a solution that finds a balance between these objectives.

As a result, the optimization process led to a reduction in the number of stations to 66 and a decrease in the number of chargers to 81. In general, optimizing station locations ensured the best possible coverage, while reducing the number of stations and chargers per station contributed to a decrease in associated costs.

Appendix: Github Resource

Source code is available on GitHub.

Chapter 8

References

Bibliography

Alternative Fuels Data Center. (n.d.). *Electric vehicle charging station locations*. U.S. Department of Energy. Retrieved April 30, 2025, from https://afdc.energy.gov/fuels/electricity-stations

Power Sonic. (n.d.). DC Fast Chargers for Electric Vehicles. Retrieved from https://www.power-sonic.com/dc-fast-chargers/

U.S. Department of Transportation. (2023). Charging speeds. Urban E-Mobility Toolkit. https://www.transportation.gov/urban-e-mobility-toolkit/e-mobility-basics/charging-speeds

Deb, K. (2001). *Multi-Objective Optimization using Evolutionary Algorithms.* John Wiley & Sons.

Bäck, T. (1996). *Evolutionary algorithms in theory and practice: Evolution strategies, evolutionary programming, genetic algorithms.* Oxford University Press.

Miettinen, K. (1999). Nonlinear multiobjective optimization. Kluwer Academic Publishers.

Coello, C. A. C. (2006). Evolutionary algorithms for solving multi-objective problems (2nd ed.). Springer.

Jin, H., Song, Q., & Hu, X. (2019). Auto-Keras: An efficient neural architecture search system. Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining

Fang, W., Zhang, L., & Yang, Z. (2018). Multi-objective evolutionary algorithms: A survey of the state of the art. Swarm and Evolutionary Computation

Marler, R. T., & Arora, J. S. (2004). Survey of multi-objective optimization methods for engineering. Structural and Multidisciplinary Optimization

Zhang, Y., Yang, H., & Gao, Z. (2021). A multi-objective optimization model for electric vehicle charging station location planning. *Journal of Advanced Transportation*, 2021, Article ID 6697971.

Wang, X., Li, Y., & Zhang, Z. (2020). A hybrid multi-objective optimization approach for the location of electric vehicle charging stations. *Energy Reports*, 6, 2345-2357. https://doi.org/10.1016/j.egyr.2020.07.010

Huang, Y., & Wang, L. (2019). A genetic algorithm-based optimization for the location of electric vehicle charging stations. *Applied Energy*, 243, 39-50. https://doi.org/10.1016/j.apenergy.2019.03.063

Khodaparastan, M., & Abbas, A. (2020). A multi-objective optimization model for the location of electric vehicle charging stations. *Energy*, 207, 118295. https://doi.org/10.1016/j.energy.2020.118295

Asensio, E., & García, A. (2021). A multi-objective optimization model for the deployment of electric vehicle charging stations. *Transportation Research Part C: Emerging Technologies*, 126, 103039. https://doi.org/10.1016/j.trc.2021.103039

Asensio, E., & García, A. (2021). A multi-objective optimization model for the deployment of electric vehicle charging stations. *Transportation Research Part C: Emerging Technologies*, 126, 103039. https://doi.org/10.1016/j.trc.2021.103039

Miralinaghi, M. R., & Fadaei, M. (2022). Location and capacity optimization of electric vehicle charging stations using a multi-objective approach. *Journal of Cleaner Production*, 335, 130235. https://doi.org/10.1016/j.jclepro.2021.130235

Sadeghi, J., & Moghaddam, H. (2014). Location optimization of electric vehicle charging stations using a genetic algorithm. *Transportation Research Part C: Emerging Technologies*, 47, 120-134. https://doi.org/10.1016/j.trc.2014.07.011

Gao, Z., & Yang, H. (2019). Optimization of the electric vehicle charging station location considering both the facility location and the traffic flow.

Transportation Research Part E: Logistics and Transportation Review, 128, 138-152. https://doi.org/10.1016/j.tre.2019.06.007

Deb, K. (2001). *Multi-Objective Optimization using Evolutionary Algorithms.* John Wiley & Sons.

Eiben, A. E., & Smith, J. E. (2003). Introduction to evolutionary computing. Springer.

K. Deb, A. Pratap, S. Agarwal, & T. Meyarivan, *A Fast and Elitist Multi-objective Genetic Algorithm: NSGA-II*, IEEE Transactions on Evolutionary Computation, vol. 6, no. 2, pp. 182-197, 2002. DOI: 10.1109/4235.996017.
$EVInfrastructureCostSource/$

U.S. Department of Energy - Electric Vehicle Charging Infrastructure. This source provides a general overview of costs for Level 1, Level 2, and DC Fast Chargers. The estimates are based on information from various industry reports. Link: U.S. Department of Energy - Electric Vehicle Charging Infrastructure

PV Magazine - What Does an EV Fast Charger Cost? This article offers insights into the cost breakdown of installing fast chargers, including hardware and installation costs for DC Fast Chargers. Link: What Does an EV Fast Charger Cost?

DEAP Documentation, "DEAP Documentation," `https://deap.readthedocs.io/en/master/`.

DEAP Documentation, "NSGA-II (NSGA2) in DEAP," `https://deap.readthedocs.io/en/master/tutorials/faq.html#how-can-i-use-nsga2`.

Fortin, F., De Rainville, F.-M., Gardner, M.-A., Parizeau, M., & Gagné, C. (2012). DEAP: Evolutionary algorithms made easy. *Journal of Machine Learning Research*, *13*, 2171–2175.

K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

Geopy Documentation, "Geopy Documentation," `https://geopy.readthedocs.io/en/stable/`.

C. A. C. Coello, "Evolutionary Multi-objective Optimization: A Historical View of the Field," in *Handbook of Evolutionary Computation*, Oxford University Press, 2006.

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). NSGA-II: A fast and elitist multiobjective genetic algorithm. *IEEE Transactions on Evolutionary Computation*

S. Nagar, *Hands-On Genetic Algorithms with Python: Solving Optimization Problems with Python*, Packt Publishing, 2019.

Requests Documentation, "Requests Documentation," `https://requests.readthedocs.io/en/latest/api/#requests.get`.

Pandas Documentation, "Pandas DataFrame Reference," `https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.html`.

Open Charge Map, "Open Charge Map API," `https://openchargemap.org/site/develop/api`.

Pandas Documentation, "Pandas Documentation," `https://pandas.pydata.org/pandas-docs/stable/`.

Matplotlib Documentation, "Matplotlib Documentation," `https://matplotlib.org/stable/contents.html`.