Computational Tools for Battery-Electric Bus Systems: From Infrastructure Planning to Daily Operations

Dan McCabe

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2024

Reading Committee:
Xuegang Ban, Chair
Balázs Kulcsár
Chaoyue Zhao

Program Authorized to Offer Degree:
Civil and Environmental Engineering

University of Washington

Abstract

Computational Tools for Battery-Electric Bus Systems:
From Infrastructure Planning to Daily Operations

Dan McCabe

Chair of the Supervisory Committee:
Xuegang Ban
Department of Civil and Environmental Engineering

Public transit operators that phase out vehicles powered by fossil fuels in favor of battery-electric buses must confront the major challenges of electric vehicles: limited driving range and significant recharging times. Buses can extend their effective range with supplemental "opportunity" charging in between passenger trips, but these operations must be managed carefully to avoid increasing delays and degrading passenger experience. Accordingly, this dissertation develops computational tools to anticipate and manage recharging needs for the complete transit system. The first tool is a public web app that processes General Transit Feed Specification data to calculate key metrics for fleet electrification planning. This processed data then serves as an input to two optimization models to design opportunity charging infrastructure and schedule its daily utilization, with an emphasis on maintaining the quality of passenger service across a broad range of operating conditions. The models' performance under real-world variations in schedule adherence and energy consumption is evaluated with a customized simulation platform based on operations data from King County Metro. The results suggest that if moderately conservative estimates of energy consumption and trip durations are used in the planning models, buses can maintain good passenger service under varying conditions.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

This dissertation is the product of much more than just my own work and would not have been possible without the help of so many others. Thank you first and foremost to my advisor, Dr. Jeff Ban, who has been a dedicated supporter of and advocate for me throughout graduate school. I am lucky to have worked with him. I am also deeply grateful to my other reading committee members, Drs. Balázs Kulcsár and Chaoyue Zhao, for providing feedback on my work, guiding my research directions, and teaching much of the core technical material that this dissertation is built on.

Thank you to my family for supplying endless love and support. My parents, Ellen and Brian, served as my first ever teachers and could not possibly have done more to support my education. I am so grateful for the sacrifices they've made and the love they've provided, all the way from preschool through Ph.D., while always letting me choose my own path. My sister Ally has long set an example of devoting a life and career to making the world better, and I hope to continue following those footsteps in my own way. My partner Kellie has provided many years of support, patience, advice, and math tutoring. Having someone alongside me who fully understands the difficulties of completing a Ph.D. but also knows and cares about me so deeply was a blessing I can't hope to summarize in a few sentences. It is hard to imagine how I could have accomplished this without her.

I am fortunate to have worked with such a great group of students in the iUTS Lab for the last five years. The early years of grad school were also made much more enjoyable thanks to the friendship of Zack Aemmer, Borna Arabkhedri, and Sai Pavuluri. Outside of CEE, special mentions go to Peter Mumford, Mike Pun, and

the entire Husky Cycling Club for making my complete graduate school experience more fulfilling, rewarding, and fun. I also want to recognize the many great people I worked with at Chalmers University of Technology in 2023 for welcoming me into their program. In addition to Balázs, thank you especially to Rémi Lacombe and Sten Elling Tingstad Jacobsen for all the helpful conversations about research and fun times around Gothenburg.

Several current and former King County Metro staff members, in particular Danny Ilioiu and Natalie Westberg, helped enhance the quality of this dissertation with their feedback on modeling approaches, data sharing, and candid conversations about their agency's experience. Speaking of data sharing, Zack Aemmer's archive of GTFS Realtime data for King County was critical for the final analysis done in this work, and I'm so thankful that Zack found time to help me use that data while in the throes of his own dissertation writing.

This work was made possible by several sources of financial support that allowed me to invest fully in my work. These include the support of the National Science Foundation through the Graduate Research Fellowship Program (Grant Number DGE-1762114), the U.S. Department of Transportation via the Pacific Northwest Transportation Center (PacTrans), and the UW Valle Scholarship & Scandinavian Exchange.

Finally, I want to thank all of the many teachers—at every level of school I've been through—who have inspired, supported, and guided me during my long academic journey. This dissertation is dedicated to them.

Chapter 1

INTRODUCTION

Battery-electric buses (BEBs) have become increasingly popular among public transit agencies in recent years due to their potential to achieve zero emissions, quiet operations, low operating costs, and minimal maintenance. Major transit agencies in the United States like New York City's MTA [80], Los Angeles Metro [73], the Chicago Transit Authority [18], and Seattle's King County Metro [63] have committed to operating fully zero-emission bus fleets by 2040 or sooner and plan to rely primarily on BEBs to reach those goals. Making such large-scale fleet conversions in the next one to two decades poses many novel challenges for agencies. With conventional diesel, hybrid, or compressed natural gas buses, vehicles can generally carry all the fuel that is needed to complete a day of service and if not, full refueling takes only a few minutes. While BEBs offer major environmental benefits and may reduce life-cycle costs, their limited range and substantial recharging times introduce new complications for agencies.

To accommodate the limitations of BEBs, agencies must introduce changes at various time scales, from their long-term planning processes to daily operations. To operate BEBs in a reliable and cost-effective manner, agencies must select the right vehicles for their needs, develop adequate charging infrastructure consistent with their anticipated charging strategy, ensure vehicle blocks (the sequences of trips served by each bus) are appropriate for the range of their buses, and plan recharging activities for each day of service. Poor decision-making can lead to reduced passenger service quality due to delayed or canceled trips, decreased reliability, and unnecessarily high electricity costs.

Inspired by these challenges that many transit agencies are currently confronting, this dissertation presents a suite of decision-making tools intended to help agencies plan for and

operate BEBs. These tools include a data processing pipeline and publicly available web app that helps agencies assess electrification challenges (Chapter 2), an optimization model for determining the sizes and locations of opportunity charging stations (Chapter 3), and an optimization model to schedule daily opportunity charging under any operational conditions (Chapter 4). The real-world effectiveness of these tools is tested with a simulation model based on BEB operations data from King County, WA, which helps identify strategies to achieve good performance under uncertainty (Chapter 5).

The remainder of this chapter provides additional context for this dissertation by describing the current state of BEB usage among transit agencies in the US and abroad (Section 1.1), BEB vehicles and chargers (Section 1.2), and the ways in which BEBs disrupt the traditional transit planning process (Section 1.3). Section 1.5 concludes this chapter with an overview of the structure of the complete dissertation.

1.1 BEB Adoption in the US and Worldwide

China is well established as the world leader in both the manufacturing and operations of electric buses and other electric heavy-duty vehicles (HDVs). From 2019–2021, 92% of the world's zero-emission HDV sales were in China [12]. Zero-emission bus sales in China have been consistently high compared to the rest of the world for many years; sales varied from about 50,000 to 100,000 vehicles per year over the period from 2017–2022 [100].

Although China is the worldwide epicenter of BEB deployment, many other countries have begun to catch up. The United Kingdom had nearly 2,800 BEBs on the road as of 2024, about half of those in the London area [121]. Approximately 1,600 electric buses operated in Colombia by 2022, nearly all of them in the capital city of Bogotá [94]. In the European Union, zero-emission bus sales totaled 2,500 in 2021, representing 10% of all new buses [85]. Electric bus adoption varied dramatically by country; on the high end, 78% of new bus sales in Finland were BEBs, while both the Netherlands and Denmark had BEB sales shares in excess of 40%. On the other hand, Greece, Ireland, and Portugal reported BEB sales shares under 1%.

Until recently, BEBs were a relatively niche technology in North America and limited to a small number of pilot projects testing the technology. In the last few years, their use has grown substantially to the point that BEBs now comprise a small but significant share of the overall American transit fleet. According to Hynes et al. [46], the total number of BEBs adopted by transit agencies in the U.S. reached 5,775 by September 2023, a 10% increase over the previous year. Here, "adopted" refers to buses that have been allocated funding to purchase as well as those that have been ordered, delivered, or deployed into service. Given that the Federal Transit Database [29] reports a total inventory of 58,688 standard and articulated buses, this number of BEBs represents nearly 10% of US transit buses. BEBs are currently the dominant technology among zero-emission buses in the US, with a total of only 327 fuel-cell electric buses (FCEBs) adopted [46].

BEB numbers can be expected to grow as many major American transit agencies pursue aggressive goals to reduce or eliminate their emissions. New York City's Metropolitan Transportation Authority (MTA), which operates the country's largest bus fleet with 5,900 total buses, committed to reaching a 100% zero-emissions fleet by 2040 [80]. MTA expects the vast majority of their zero-emissions vehicles to be BEBs. Most other large agencies have established similar goals and plan to meet them with a heavy reliance on BEBs. Los Angeles Metro has a particularly ambitious goal of full zero-emissions by 2030, 10 years ahead of California's state-mandated 2040 deadline [73]. Their fleet comprises over 2200 buses, and like MTA, they expect to use BEBs rather than FCEBs or other zero-emission technologies to meet this goal. Chicago plans to electrify its fleet of 1800 transit buses by 2040 [18], while Seattle's target year is 2035 for a 1400-vehicle fleet expected to include about 1200 BEBs and 200 trolleybuses [63]. As of now, BEBs are the technology of choice for American agencies aiming for zero-emissions status, though agencies state they plan to monitor FCEB developments and consider them as a part of future fleets. However, agencies express concerns over the higher purchase and fuel costs of FCEBs, fire risks, and general uncertainty about the ultimate costs [18, 73, 80].

1.2 Overview of BEB Technology

The BEB sector has grown to encompass a wide range of vehicle manufacturers, bus types, battery options, and charging infrastructure styles. This section provides an overview of the technologies currently available and the decisions that influence agencies' decisions of which to utilize.

BEB vehicle availability has grown over time to the point that many models are now available in various world markets. These options are compiled in the Zero-Emission Technology Inventory (ZETI) [14]. For 2022, the most recent year available as of writing, ZETI reports a total of 275 battery-electric bus models available worldwide, 23 of which could be purchased in the U.S. & Canada. Some major manufacturers in the North American market include BYD, New Flyer, and Gillig. In comparison, 175 models were available in China, 48 in Europe, 19 in India, 11 in Mexico, and 3 in South America. Among those buses available in the US, advertised range varied from 75 mi. (121 km) to 329 mi. (529 km). A variety of bus types are sold, from smaller buses 30 ft. long or shorter to 60-ft. articulated buses.

In addition to the variety of vehicle models available, there are also various types of charging equipment available with different interfaces, power outputs, and manufacturers. This charging equipment may be intended for a *depot charging* application in which buses are slowly charged overnight at a bus base, or high-power *opportunity charging* at bus stops or layover locations during the service day, which is also sometimes referred to as *on-route* or *layover* charging.

There are three main categories of charging systems for BEBs: plug-in, overhead conductive, and wireless (or *inductive*) [111, 112]. Plug-in chargers, as the name suggests, must be manually connected to a vehicle. These are the least expensive variety of charger and power output ranges from 40-120 kW [111]. Overhead conductive chargers connect the bus to an overhead power source using a pantograph, a mechanical arm that connects the bus and charger automatically. These systems offer the maximum charging power currently available [111], reaching up to 600 kW [87]. Wireless charging systems are typically embedded into

the roadway and are used for on-route charging at a power output up to about 250 kW. Despite their lower power output compared to conductive chargers, wireless systems have some advantages because they do not require a physical connection between bus and charger. This decreases the time and effort required to charge and makes frequent on-route charging (for example, at all bus stops) a possibility [112]. All three charging approaches can be utilized in either depot/overnight or on-route/opportunity charging; however, plug-in charging is most commonly used for low-power depot chargers, while overhead and wireless charging are typically used for opportunity charging [112] while buses are in service.

Given the wide range of available battery sizes and charger power levels, agencies that use BEBs often choose one of two schemes: equipping buses with relatively large batteries and relying solely on overnight depot charging, or utilizing buses with smaller batteries and recharging them frequently throughout the day [111]. A hybrid strategy is also common, in which buses are fully charged at depots overnight but also use opportunity charging to extend their range during the service day for buses with particularly long blocks. This hybrid strategy is the primary focus of this dissertation. The optimization models developed in Chapters 3 and 4 focus on the design and operations of opportunity chargers that are used to extend buses' daily range when necessary, while assuming that agencies also use overnight depot charging as much as possible.

1.3 Impact of BEBs on Transit Planning and Operations

Traditional transit planning consists of several interrelated steps in which routes, timetables, and vehicle and driver schedules are established in order to provide good service at low cost. Typically, these problems are addressed sequentially rather than simultaneously, as each step is already complex to optimize [15]. Replacing diesel buses with BEBs further complicates the traditional transit planning process by introducing new challenges related to capital investments in buses and chargers, location of chargers, possible schedule revisions, and charging scheduling [93].

Figure 1.1, reproduced from Perumal et al. [93], summarizes the impact that BEBs have

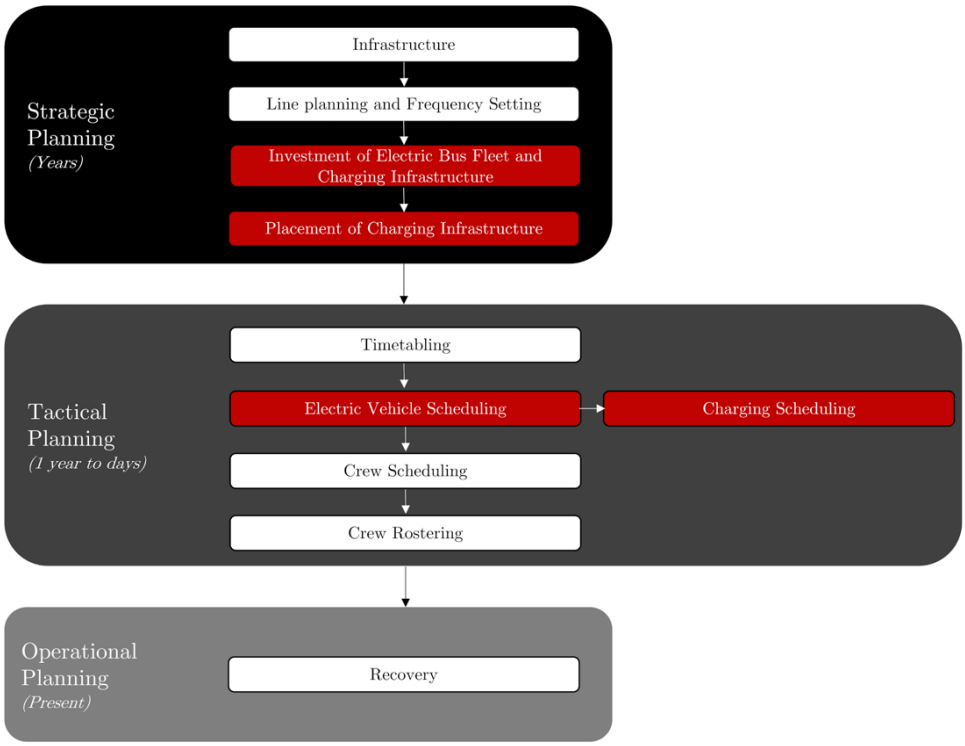


Fig. 4. Electric bus planning process.

Figure 1.1: Impact of electrification on transit planning. Reproduced from Perumal et al. [93] under CC BY-NC-ND 4.0.

on the transit planning process. BEBs introduce long-term "strategic planning" decisions about bus and charging infrastructure purchases as well as charger locations. In the medium-term "tactical planning" phase, electrification primarily impacts vehicle scheduling, the process by which vehicle blocks are designed. BEBs also introduce charging scheduling decisions to the planning process at a variety of stages: potentially when establishing charger locations (such as in Chapter 3), when building vehicle schedules (a particularly active area of research), and when planning for each day of operations (as in Chapter 4).

1.4 Contribution

This dissertation advances the state of research on battery-electric bus planning and operations and makes that progress accessible to practitioners with publicly available software. The main contributions of this work can be summarized as follows:

1. A publicly available web app called the Zero-Emission Bus Range & Recharging Assessment (ZEBRA) that processes General Transit Feed Specification (GTFS) data to provide summary information and interactive visualizations about bus range needs. The data processing core of ZEBRA, called GTFS-BEB, is also essential to the real-world applications of the other work in the dissertation.

2. An optimization model to decide the locations and number of chargers for opportunity charging stations at layover sites. The model is flexible, allowing buses to travel some deadhead miles to visit chargers if it is advantageous. It incorporates GTFS data on current schedules to make sure sufficient chargers are built to serve all blocks while respecting the capacity of charging sites.

3. An optimization model that schedules the use of opportunity chargers within a service day. Compared to prior approaches in the literature that generally restrict charging to take place during scheduled layover time, this model is more flexible, exactly tracking any delays that result from charging but ensuring they are minimized. The model is accompanied by two tailored solution methods, most notably a heuristic method that generates results in a fraction of a second on real-world networks.

4. A discrete event simulation built on real-world BEB operations data to evaluate system performance under underctainty when charging schedules are designed by the aforementioned deterministic optimization algorithms. Testing shows how the results from the charging scheduling model, which can be obtained in only a fraction of a second,

perform well under real-world conditions, fitting in necessary charging activities with minimal impact to passenger trip delays.

1.5 Structure of the Dissertation

This remainder of this dissertation is structured as follows. Chapter 2 describes the ZEBRA web app and GTFS-BEB data pipeline built to analyze transit agencies' BEB fleet conversion potential using GTFS data. Chapter 3 describes an optimization model to determine sizes and locations of opportunity charging stations, originally published as McCabe and Ban [76]. Chapter 4 describes an optimization approach to scheduling charging for a single day of service, recently submitted as a journal paper and published as a preprint at McCabe et al. [77]. Chapter 5 first analyzes GTFS Realtime and BEB energy consumption data from King County Metro to create statistical models of these key parameters for the two optimization models, then employs them within a discrete event simulation model to evaluate system performance and recommend strategies for creating charging schedules that perform well under uncertain conditions. Chapter 6 concludes the dissertation by summarizing the main contributions and limitations as well as discussing promising directions for future research.

Chapter 2

ASSESSING BEB FLEET CHALLENGES WITH GTFS DATA: THE ZEBRA WEB APP

2.1 Introduction

A first step in designing the right recharging strategy for a BEB system is to understand the anticipated driving needs and recharging opportunities. Current or planned vehicle schedules can be analyzed to determine the distribution of vehicle range required to serve all blocks and identify where buses are located at trip and route terminals, which helps inform decisions about opportunity charging locations. This type of static schedule information is documented in the General Transit Feed Specification (GTFS) tables published by most North American transit agencies, but significant data processing must be done in order to piece together information from separate tables and build up a picture of each bus's service day along with the key metrics relevant to BEB operations. This information may then be used directly for decision-making or serve as inputs to optimization and simulation models including those discussed in Chapters 3–5.

This chapter first presents a data processing pipeline referred to as GTFS-BEB to quickly summarize vehicle schedule, trip, and shape information from GTFS and provide insights relevant to electrification. GTFS-BEB forms the backbone of the accompanying Zero-Emission Bus Range & Recharging Assessment (ZEBRA) app, a public screening tool to help understand how suitable a given transit agency's bus network is for BEB service, that is accessible online at the time of writing at `https://bit.ly/zebra-app`. Section 2.1.1 begins by providing an overview of the GTFS standard. Section 2.1.2 describes the small number of other public tools available for BEB planning system planning. Section 2.2 describes the GTFS-BEB data processing pipeline, Section 2.3 discusses the ZEBRA app's design and

implementation, and Section 2.4 shows their combined application to a case study of King County Metro.

2.1.1 The General Transit Feed Specification

Transit agencies throughout the world publish data on routes, schedules, and more in the GTFS format. GTFS arose from the need to have a standardized data exchange between transit agencies and external software developers who created tools such as transit trip planners. As planning software for car trips such as Mapquest and Google Maps became widespread, transit initially lagged behind [98]. In response to this challenge, the GTFS standard arose in 2005 from a collaboration between Google and TriMet in Portland, OR as a way to incorporate TriMet's transit service information into Google Maps and was quickly published by other agencies and consumed by other software providers. In 2011 [82], GTFS was expanded with the complementary GTFS Realtime standard so that agencies could share live information on vehicle positions and service alerts in addition to the static schedule data from standard GTFS. The GTFS standard has been adopted widely, with over 10,000 transit agencies in more than 100 countries publishing GTFS data [81]. In 2023, the U.S. Federal Transit Administration began requiring all American transit agencies to supply static GTFS data along with their annual National Transit Database reports [32].

The static GTFS files consist of several individual tables distributed as text files, typically published on an agency's website and easily accessible to developers. The standard defines a variety of tables that can be either required or optional, to allow agencies to publish only the bare minimum schedule information used by travelers or provide complete details about trip schedules, blocks, fares, stops, and more [81]. The fields utilized by GTFS-BEB and ZEBRA are discussed in more detail in Section 2.2.1.

2.1.2 Other Tools for Transit Electrification

A limited number of public tools are available to aid transit agencies with their electrification programs. These tools tend to be specialized and focused on one particular part of

the electrification process, such as estimating emissions impacts from fleet conversion or predicting energy consumption needs. The U.S. Federal Transit Administration developed the FTA Transit Bus Electrification Tool [31], a spreadsheet tool that helps agencies estimate the partial life-cycle emissions impacts of replacing a standard bus fleet with low- or zero-emisson buses such as BEBs. This tool does not use detailed fleet or schedule data; the only agency-specific inputs are the number of vehicles of each fuel type estimates of yearly vehicle miles traveled. As such, the tool can estimate emissions impacts of fleet changes, but does not provide insights into how agencies can go about replacing some of their existing buses with BEBs.

A number of other tools focus specifically on energy consumption modeling at the route or trip model. The open-source Drivecycle Python package [83] uses GTFS shape and stop information to create vehicle drive cycles and estimate energy consumption. Similarly, the Route_Dynamics project offers a Python package to simulate vehicle driving patterns based on GTFS route data with a focus on impacts to battery performance [24]. The National Renewable Energy Laboratory offers the FASTSim [11] and RouteE [43] software packages to estimate vehicle energy consumption by simulating drive cycles using Python or Excel. While these can be valuable resources for predicting vehicle energy needs, they are not intended to analyze complete transit networks, focusing instead on standardized or simulated drive cycles for individual trips.

In summary, while a handful of free public tools are available to help transit agencies with analysis of their systems' potential to be electrified, there are no well-known tools that can analyze fleet GTFS data holistically to summarize electrification challenges and recharging needs. Transit agencies typically need to analyze expected battery size and recharging needs on their own or hire consulting firms to do this work for them. A standardized approach to processing GTFS data into metrics and visualizations as provided by GTFS-BEB and ZEBRA can help agencies understand key electrification challenges with a free and easy-to-use online tool.

2.2 GTFS-BEB Data Processing Pipeline

Both the ZEBRA app and the BEB planning and operations models in Chapters 3–5 rely on a common backbone: the GTFS-BEB data processing pipeline, presented in the current section. Section 2.2 first describes the GTFS data tables and fields required for the analysis, followed by Section 2.2.2 that describes how the GTFS files are used together to gather necessary information about BEB trips and blocks.

2.2.1 GTFS Data Requirements

Static GTFS tables contain a large portion of the underlying data needed to analyze electrification potential and ultimately run ZEBRA. Table 2.1 lists all the data fields that are needed to run GTFS-BEB and ZEBRA, as well as their requirement status according to the GTFS standard. Most of these files and fields are mandatory for static GTFS feeds. The key exceptions are the `block_id` column of the `trips.txt` table and the `shapes.txt` file. Most large agencies include this information in their GTFS feeds, but relying on them in our analysis does limit applicability a bit. However, in order to assess the difficulty of electrifying a fleet, we must have information about which trips are completed by the same bus (the `block_id` field) and the shape of each trip (from `shapes.txt`), which is used to determine trip distances and ultimately vehicle range requirements.

2.2.2 Data Processing

Figure 2.1 provides an overview of the GTFS-BEB pipeline. The inputs consist of the six necessary GTFS tables defined in Table 2.1 along with the user-supplied parameters of an analysis date and a list of routes to be served by BEBs (which could be the full set of bus routes). The date is used along with the `calendar.txt` and `calendar_dates.txt` files to identify the full set of GTFS service IDs that are active on the given day. The service IDs are then used to filter down the full set of trips in `trips.txt` to only those active on the given date. The supplied route names are matched with their route IDs from `routes.txt`

Table 2.1: GTFS data files and fields required by GTFS-BEB and ZEBRA.

File	File Required?	Column	Column Required?
`calendar.txt`	Conditionally	all columns	Yes
`calendar_dates.txt`	Conditionally	all columns	Yes
`trips.txt`	Yes	trip_id	Yes
		route_id	Yes
		service_id	Yes
		block_id	No
		shape_id	Conditionally
`routes.txt`	Yes	route_id	Yes
		route_type	Yes
		route_short_name	Conditionally
`shapes.txt`	No	shape_id	Yes
		shape_pt_sequence	Yes
		shape_lat	Yes
		shape_lon	Yes
`stop_times.txt`	Yes	trip_id	Yes
		arrival_time	Conditionally

and used to further filter down the trips—all trips with a particular block ID are included if at least one of those trips serves one of the input routes. After this filtering step, the full set of trips served by BEBs has been identified, and the remainder of the data process consists of adding key parameters from other GTFS tables to each of those trips.

The data in `shapes.txt` is processed to calculate the total length of each shape as well as extract the latitude and longitude coordinates of the trip start and end locations. Shape distances are calculated by estimating the consecutive distance between all points in each shape using the well-known haversine formula, which calculates the distance between two points on a sphere. The distances may be incorrect if consecutive points are a large distance apart, the bus does not take a direct path between them, or there is a significant elevation difference. However, using the haversine approximation rather than relying on a detailed surface model makes it possible to calculate the distance of hundreds of shapes (each containing thousands of points) almost instantly and should be sufficiently accurate.

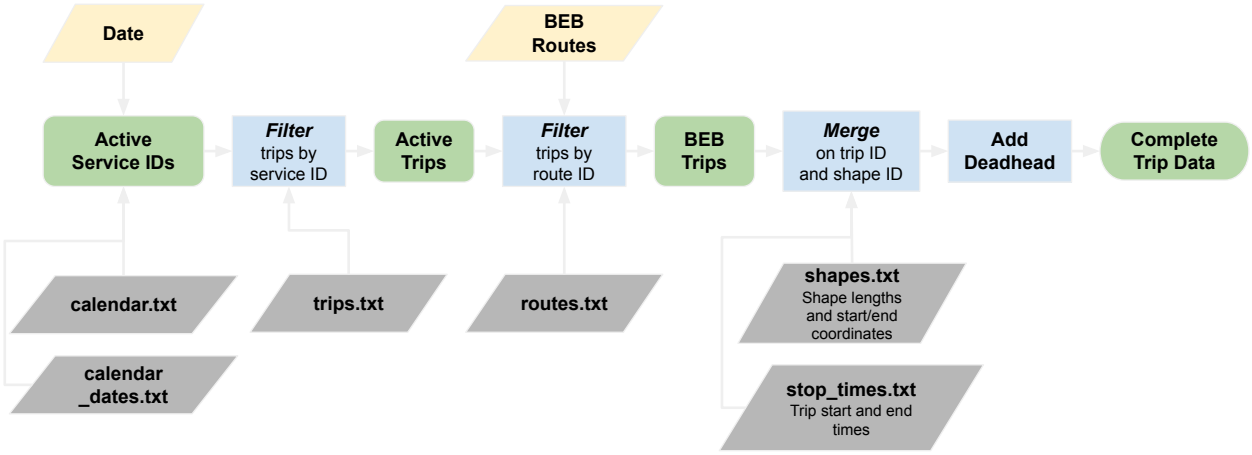


Figure 2.1: Flowchart showing GTFS-BEB data processing pipeline.

The `stop_times.txt` file contains arrival and departure times at all stops serviced by each trip and is used in GTFS-BEB only to identify the start and end times of each trip based on the arrival times at the first and last stops, respectively. These trip times are critical to BEB service because they define the amount of layover time available for charging in between trips without delaying passenger service.

The final step in processing all BEB trip data is to estimate deadhead distances in between consecutive trips. While successive trips usually start and end in the same place, sometimes a vehicle must drive a significant distance from the end of one trip to the start of another. GTFS-BEB uses the end and start coordinates identified from `shapes.txt` as origins and destinations between consecutive trips and estimates the distance between them using the "Manhattan" distance (mathematically, the L1 norm) between these points. As with the shape distance approximation, using the Manhattan distance for this purpose introduces some approximation error in exchange for expediting calculations, which is especially important when used as part of the ZEBRA web app. Approximating the Manhattan distances also avoids the need to use a street distance calculation service such as the Openrouteservice [89] or Google Maps [38] APIs in the public app, which subject their users to usage limits (in the case of free programs) or charge usage-based costs.

One notable omission from the deadhead distance calculations in GTFS-BEB is pull-out and pull-in trips for each bus at the start and end of the day, respectively. The GTFS standard does not include any information about bus depot locations, which would be required to estimate the distance of these trips. In the detailed case studies of the forthcoming chapters, we specify a depot location in order to incorporate pull-out and pull-in trips, but this is not part of the standardized GTFS-BEB methodology or ZEBRA app because the necessary information is not included in GTFS.

The resulting trip-level data includes the columns: block ID, trip ID, trip number within the block, route served, start time, end time, start latitude and longitude, end latitude and longitude, service trip distance, and deadhead distance to the start of the next trip. This data can then be used for further analysis by the ZEBRA app or the optimization and simulation models discussed throughout this dissertation.

GTFS-BEB is implemented as a Python module with a customized `GTFSData` class that relies heavily on the popular `pandas` library. Each of the GTFS tables are stored as Pandas `DataFrame` objects for efficient manipulation.

2.3 ZEBRA Web App

Because it is based on standardized and widely available data, GTFS-BEB can quickly generate a thorough overview of BEB fleet conversion prospects for any agency publishing a complete set of GTFS files. To extend the impact of this capability, the Zero-Emission Bus Range & Recharging Assessment (ZEBRA) app was built upon GTFS-BEB to provide an interactive, web-based interface for analyzing transit systems' electrification opportunities. Agencies that are in the early stages of planning for BEBs are likely to share many basic questions, including whether the range of current electric buses is sufficient to meet their needs—and, if not, which blocks or routes are the most or least ready for electrification. This information can help agencies both identify the blocks and routes that could immediately be served by BEBs with depot charging alone and which would require opportunity charging, revisions to their block structures, or longer-range vehicles.

To maximize its impact, ZEBRA was developed with the goals of being easy to use, straightforward to understand, and applicable to any transit agency. The app requires only a small number of user inputs beyond the GTFS files so that the tool is easy to run, and free and open-source dependencies are used to make sure the app can be distributed at no cost. Rather than making specific suggestions about exactly how an agency might electrify its fleet, ZEBRA helps users assess both the capability and limitations of a transit system served with BEBs, regardless of their comfort level manipulating the data themselves. From there, agencies can use their own expertise to make further decisions about electrification, whether that involves block/schedule revisions, opportunity charging, fleet expansions, or alternative zero-emission bus types.

2.3.1 App Structure

Figure 2.2: Structure of ZEBRA user experience.

Figure 2.2 provides a high-level overview of the ZEBRA workflow. Users first specify GTFS source files by either uploading their own or selecting from some default options that are packaged with the app. Users are then prompted to select one of the service dates within the range covered by these files (as defined by the contents of `calendar.txt` and `calendar_dates.txt`), which defines a set of active trips as referenced in Figure 2.1. The backend code identifies all of the route names active on that day and presents them as options for the user to select on the next screen. Specifying the routes lets users analyze partial electrification scenarios involving a limited set of routes, or to exclude routes that will be served by a different vehicle type. Whenever the user changes the set of trips selected, GTFS-BEB filters down the list of all active trips to only those on the given routes and calculates all trip parameters as discussed in Section 2.2.

The final input users are asked for is anticipated vehicle range, which is input as a total driving distance in miles, rather than a battery size, for simplicity. The range setting allows ZEBRA to report the percentage of blocks that can be completed by vehicles with the specified range and also enables further analysis of the out-of-range blocks, including plotting potential charging locations and summarizing which routes are most difficult to serve with BEBs. These summarized results are shown along with the app usage example in Section 2.4.

2.3.2 Implementation

The ZEBRA app is written entirely in Python. It relies heavily on the Streamlit library to create a user interface, which made it possible to develop a useful and reliable app without extensive coding in multiple languages. Data processing is done primarily with the Pandas library; the interactive plots and maps are produced using Plotly.

ZEBRA is hosted on a University of Washington server. The deployment process involves building and testing a local Docker image that contains the Python code and all dependencies, pushing that image to Docker Hub, then pulling and building that image on the server via SSH.

2.4 Case Study: King County Metro RapidRide Routes

As an illustrative example, in this section we analyze the potential of electrifying all of King County Metro's RapidRide high-frequency routes in the Seattle metro area. In the GTFS Data Specification page (Figure 2.3), we choose the default option of using King County Metro's GTFS files. Likewise, for the analysis date, use the default (busiest) day of May 17, 2024.

On Select Bus Routes (Figure 2.4), we start by clearing the list of routes (which includes all by default). Then, we manually add all current RapidRide lines: A, B, C, D, E, F, and H. Now we progress to Range Requirements, where we're presented with the first meaningful results of ZEBRA's analysis. On this page, users can set the range of the buses that will

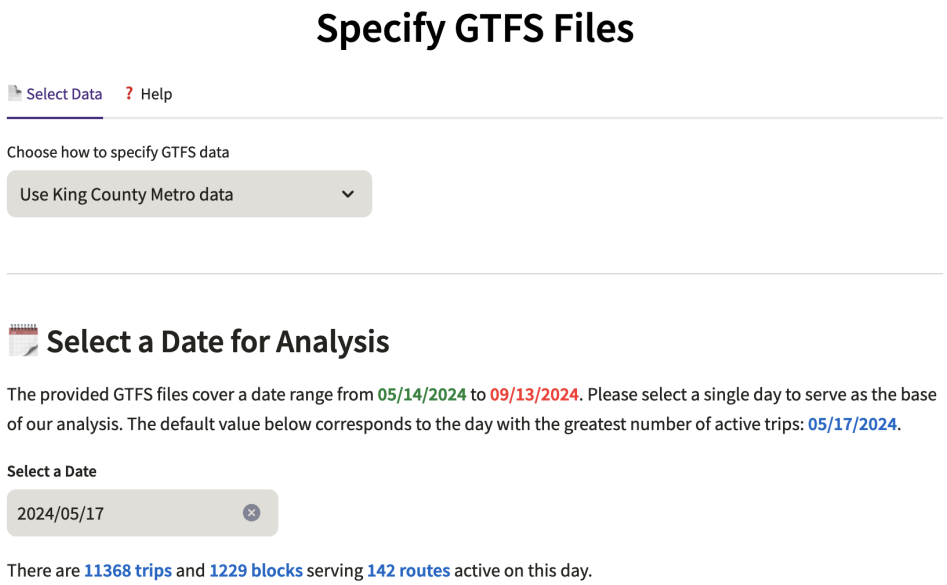


Figure 2.3: Specifying default GTFS files to start the analysis.

serve the routes in their analysis. We ask for range in miles rather than battery size in kWh to keep the technical knowledge barrier to a minimum. In this example, we keep the default range of 150 miles.

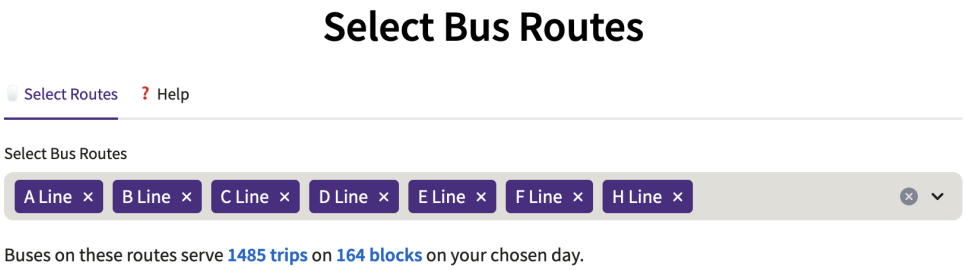


Figure 2.4: Selecting RapidRide bus routes for analysis.

On the Block Distances tab (Figure 2.5), we show two interactive figures: a histogram of the distance of each block from GTFS as well as an empirical cumulative distribution function (CDF) of that same data. The histogram gives a familiar, easy to interpret visual

of how far each bus needs to drive each day. The empirical CDF, especially in interactive format, makes it easy to examine the impact of bus range. By hovering over the figure, we can check exactly what percentage of blocks are below a certain distance threshold. For example, in the RapidRide case, 95% of blocks could be completed with depot charging alone if buses had a 200-mile range (compared to 76% with a 150-mile range).

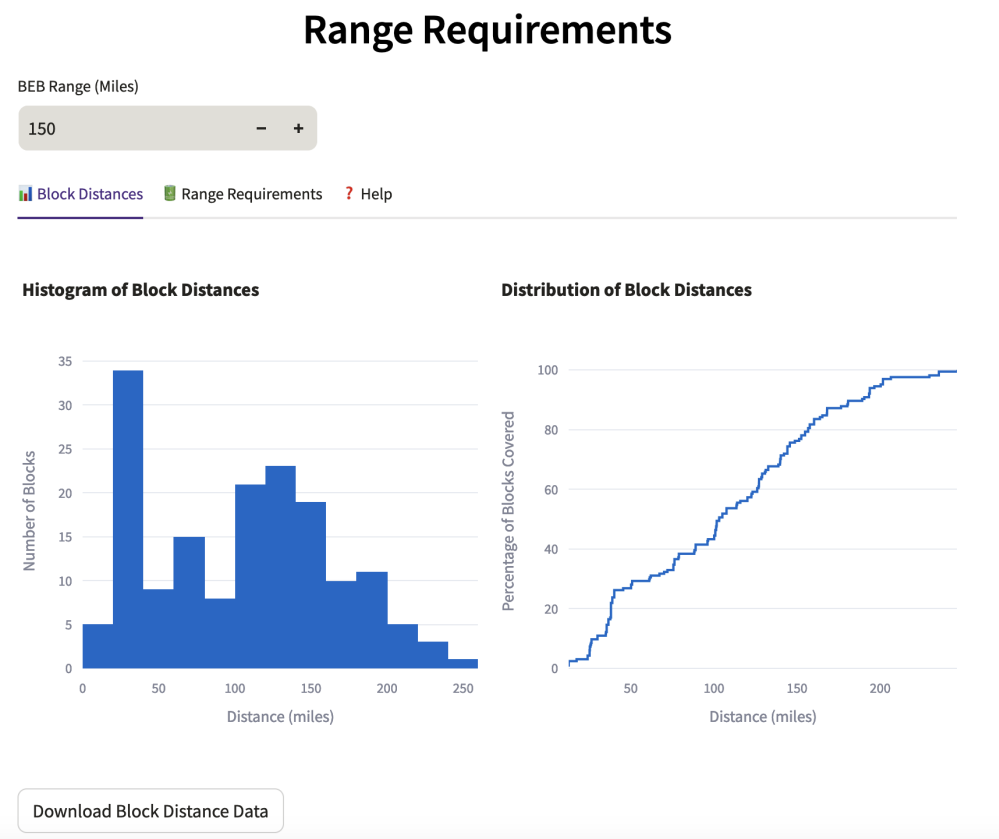


Figure 2.5: Block Distances tab showing RapidRide results.

The Range Requirements tab of the same page (Figure 2.6) provides more details on which blocks would be covered with depot charging alone and how much additional range is necessary. In our example, we're informed that 76% of RapidRide buses (59% of total RapidRide vehicle miles) can be served with depot charging alone, and the average bus that can't rely on depot charging alone would need an additional 31 miles of range. The chart

provides a detailed breakdown of how much more range each bus requires.

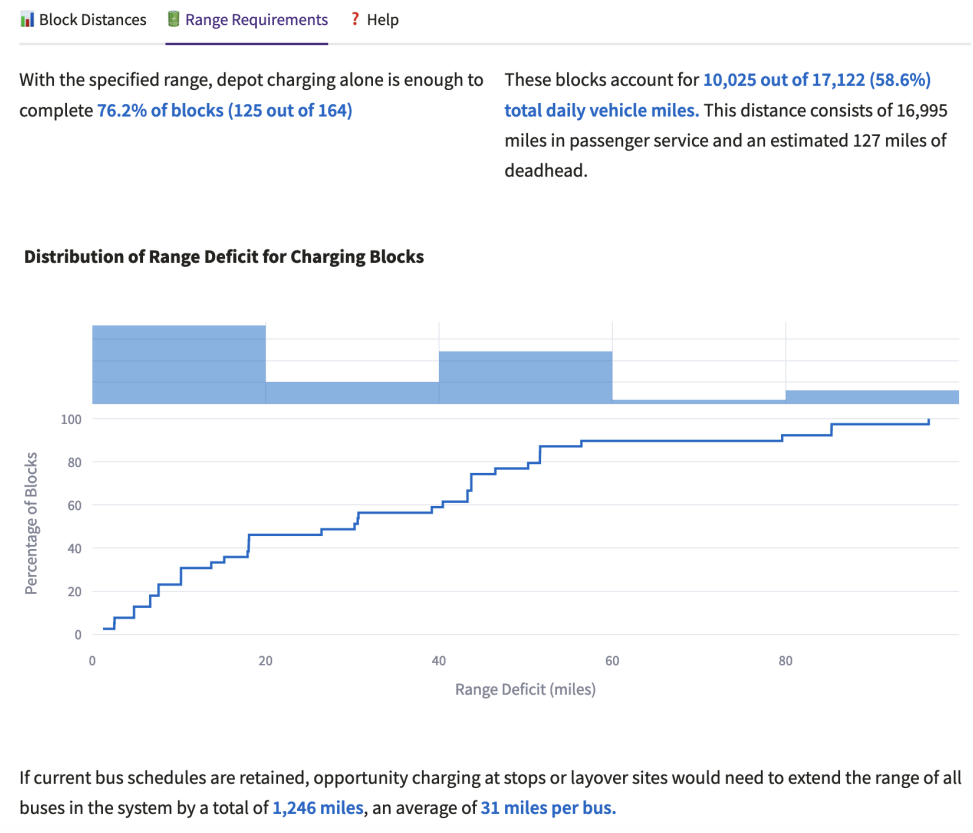


Figure 2.6: Range Requirements tab showing RapidRide results.

We can examine these out-of-range blocks more thoroughly on the Charging Needs page. ZEBRA provides nicely formatted tables summarizing which routes most often feature on these blocks as well as the details of these long blocks themselves. Longest Blocks lets us select a specific block ID and either view all its relevant data processed from GTFS or see a map of all trips it includes, zeroing in on exactly what makes the block difficult for a BEB. This page also hosts the Recharging Map tab (Figure 2.7) where we can see a heatmap of all trip terminals on blocks that must use opportunity charging. This map identifies the common locations where buses are when they're free to charge between trips, helping agencies get a preliminary idea of where opportunity charging might be feasible.

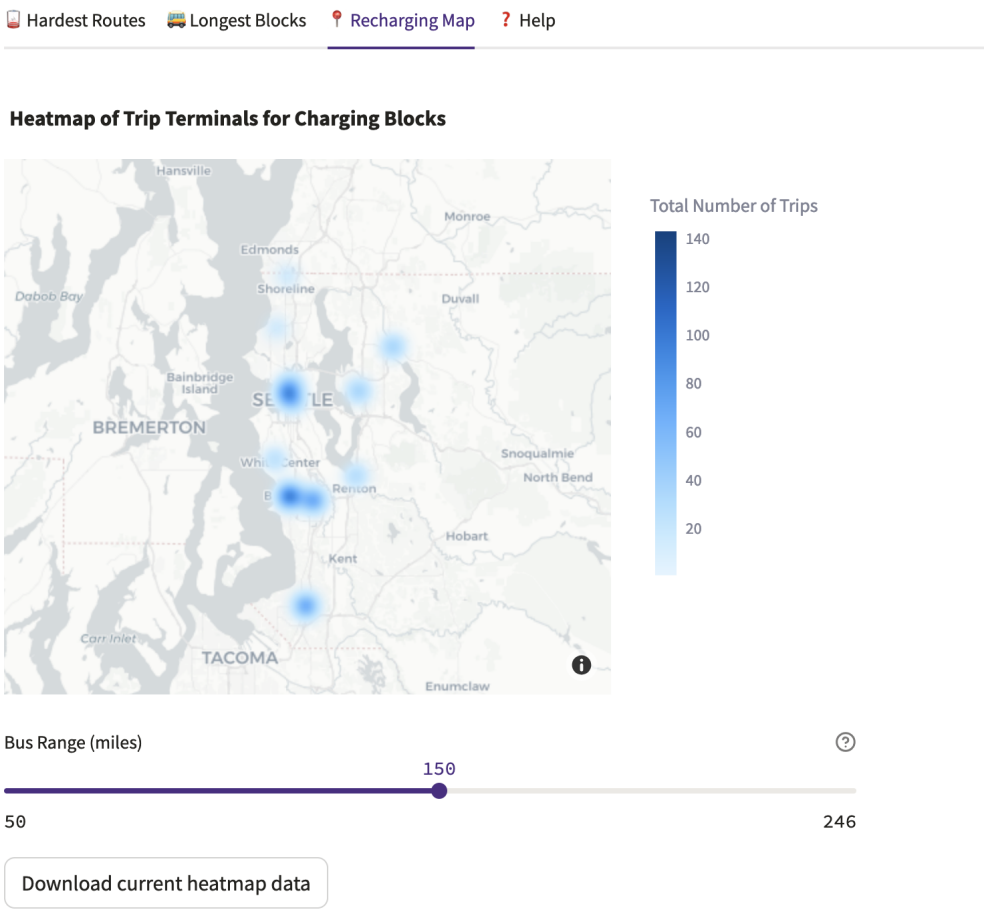


Figure 2.7: Heatmap from Recharging Map tab on Charging Needs page with RapidRide results.

Finally, ZEBRA provides some estimates of environmental benefits on the Emissions Impact page. Based on average emissions rates from TCRP Research Report 226 [79], we estimate emissions reductions compared to diesel buses by calculating total bus miles per day. ZEBRA breaks down these figures by block type (within or beyond range) to help agencies assess the potential value of opportunity charging, schedule revisions, or other measures necessary to deploy electric buses on routes outside their stated range. We find that converting to electric buses would reduce emissions on the day of analysis by 35 metric tons of CO2 equivalent: about 21 tons from buses that can rely on depot charging only and

about 15 tons more from buses serving blocks longer than the input range of 150 miles.

2.5 Discussion

This chapter presented the GTFS-BEB data processing library and ZEBRA app used to extract key information from transit agency GTFS files and present the results interactively. GTFS-BEB combines information about trip attributess and vehicle schedules to estimate the distance of all passenger and deadhead trips, ultimately giving the total distance of each block to be served by a BEB. These details are further compiled and visualized in ZEBRA to help users understand how far buses need to drive each day, which routes and blocks are most challenging to serve with BEBs, and where opportunity chargers might be located.

While the results shared by ZEBRA provide a useful high-level screening of BEB fleet conversion challenges, many detailed questions remain for agencies, including decisions about opportunity charging infrastructure design and operations. To help create more detailed plans for recharging BEBs, the remainder of this dissertation focuses on optimization models for designing charging infrastructure (Chapter 3) and scheduling daily charger usage (Chapter 4), both of which rely on GTFS-BEB to perform real-world case studies based on GTFS data.

Chapter 3

OPTIMAL LOCATIONS AND SIZES OF LAYOVER CHARGING STATIONS FOR ELECTRIC BUSES

3.1 Introduction

Transit agencies throughout the world have commenced a large-scale transition to buses powered by electricity rather than diesel or natural gas. As of 2021, an estimated 600,000 battery-electric buses (BEBs) were on the road across the world [8]. Although BEBs make up only a small portion of the current global transit fleet, nearly 40% of new bus sales in 2020 were BEBs and this market share is expected to grow rapidly, given the current urgent need to combat climate change in nearly all countries. While the vast majority of the world's existing BEBs are in China, many other countries are also seeing rapid growth in BEB adoption. The United Kingdom's electric bus fleet is projected to nearly triple by 2024 [5], for example, and several South American cities have growing BEB numbers; Santiago, Chile, and Bogotá, Colombia, each have hundreds of BEBs in service already [109]. In the United States, at least 70 transit agencies were operating BEBs as of 2018 [111], and the recent Bipartisan Infrastructure and Jobs Act allocated over $5 billion to support agencies' purchases of low- and zero-emission transit vehicles [30]. In the Seattle area, King County Metro Transit (hereafter referred to as Metro) purchased over 100 BEBs in 2020 [55] to start the transition to a fully zero-emissions bus fleet by 2035 [56].

BEBs are seeing widespread adoption because they offer several key advantages over other bus types. They are quieter and smoother in operation than their diesel and hybrid counterparts, have fewer moving parts, and eliminate tailpipe emissions that degrade urban air quality [111]. Research has consistently found that BEBs have lower life-cycle greenhouse gas emissions than conventional buses [25, 88, 122], especially if electricity is generated from

renewable sources. Electric buses can also have much lower lifetime energy costs when compared to buses propelled by internal combustion engines, especially in areas with low electricity costs, and lower maintenance costs due to their comparatively simpler powertrains [51]. On the other hand, the main disadvantages of current BEBs are high capital costs (for buses, batteries, and chargers), limited driving range, and relatively slow recharging times. Agencies seeking to gain the benefits of BEBs therefore need to plan carefully to ensure that their fleet conversion is financially achievable and that charging does not significantly interfere with passenger service.

During this planning process, agencies adopting BEBs need to determine not only where charging stations are located and how many chargers are installed at each site, but also when and where buses will be charged. A common strategy is to charge buses overnight as much as possible, using relatively low-power plug-in chargers with an output of 40-120 kW [111]. Most or all of an agency's buses are likely to be out of service overnight, allowing plenty of time to recharge batteries, which typically takes anywhere from 1-8 hours depending on battery size and charger power output [111]. Electricity prices are also usually at their lowest overnight because demand is low [112]. However, this strategy alone may not be enough to serve all of an agency's routes each day. In the Seattle case, Metro anticipates that the range of BEBs with current battery technology is sufficient to serve about 70% of their bus assignments without requiring charging during the day [61]. The remaining 30% of buses could not be converted to electric without making significant changes to their daily operations.

One way to make such a change would be to reduce the number of trips served by each bus to decrease the total daily energy demand. Doing so would require increasing the bus fleet size and would increase spending on bus purchases, maintenance, and driver pay. Alternatively, buses could rely on supplemental charging in between trips, which is referred to as *layover charging* and is the focus of this paper. This approach relies on high-power chargers with a maximum output of 150-600 kW [87], which allow batteries to gain meaningful amounts of energy in minutes rather than hours. These fast chargers are typically located at the terminal stops of routes or other locations where buses park for several minutes or more in

between trips, so that buses do not need to drive too far to access them. Despite the high upfront cost of fast chargers, a single charger can enable all-day operation for multiple buses without reducing the number of trips they serve, as shown in our results in Section 3.4.1. Accordingly, Metro plans to use layover charging where necessary [63], as do transit agencies in other major American cities including Los Angeles [73] and Boston [75].

To clarify the meaning of layover charging and related transit terminology in this work, we introduce some definitions here. A *trip* refers to a single one-way completion of a specified route, starting and ending at stops referred to as *terminals*. On a given day of service, each BEB is assigned to a *block*, which is a series of trips to be completed sequentially by the same vehicle. A block often consists of several trips serving the same route in both directions, but may include trips on any number of different routes. Serving multiple routes with the same vehicle is known as *interlining* and can help to reduce the total fleet size needed [15]. In either case, buses may need to travel some empty *deadhead* miles in between two successive trips.

Most buses run on a predefined schedule with target arrival and departure times at all stops along each trip. Some down time between trips, called *layover time* (or *recovery time*), is built into each block to provide required rest time to drivers as well as to allow a bus that has been delayed to catch back up to schedule. We assume buses operate on such a schedule, which is a typical approach throughout the world [15]. Data on blocks of this form is widely available via the General Transit Feed Specification (GTFS) [39] in the United States and many other countries [110], making it easy to obtain relevant data for our models.

In layover charging, we assume buses may deadhead from the last stop (terminal) of a trip to a charger located elsewhere, then deadhead from there to the first stop of the next trip. This differs from the alternative strategy of *on-route charging* in which buses charge for shorter durations (less than 5 minutes) at stops while passengers are onboard, as modeled in works such as Xylia et al. [119]. While each approach may be appropriate for a particular transit system, layover charging ensures that bus trips do not take longer than necessary due to charging at stops, and also allows charging sites to be shared by buses serving different

routes with different terminals, which may reduce the total of chargers needed. Both layover and on-route charging strategies are sometimes referred to as *opportunity charging*, but in this work we use the term "layover charging" to emphasize the planned nature of this charging and the fact that it is likely to take place at existing layover sites where drivers take scheduled breaks in between trips. Figure 3.1 illustrates the structure of a block together with the layover charging decision, which is consistent with the charging strategy Metro plans to implement beginning in 2026 [63].

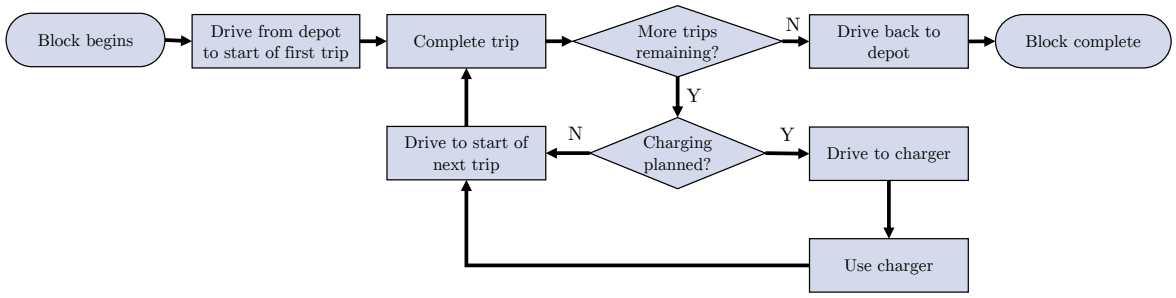


Figure 3.1: Flowchart showing the scope of a single bus block with layover charging.

The choice of layover charging locations has significant cost and performance implications; the chosen sites should provide convenient and reliable charging (i.e., ensuring that chargers are available and queuing is not required), but agencies need to balance these operational concerns with the large upfront costs of high-power charging infrastructure. Indeed, in the current early stage of the conversion to electric transit buses, agencies cite both infrastructure costs and the added complexity due to charging operations as barriers to BEB expansion [9]. There is a clear need for models that help transit agencies develop charging infrastructure systems that limit costs with minimal disruption to transit services.

Inspired by the charging infrastructure challenge faced by transit agencies as well as the relatively limited existing literature on this problem, this paper proposes a BEB charging optimization approach that considers practical bus operation characteristics (such as the block structure), while at the same time balancing model complexity and serving the charging

decision needs of transit agencies. Our unique approach stands out from existing work in a few key ways. First, throughout our work we focus on adhering to existing trip timetables and preserving existing bus blocks as much as possible. Ensuring all trips run as previously scheduled for conventionally fueled buses helps preserve the passenger experience, ensuring that switching to BEBs doesn't degrade quality of service. Retaining bus blocks is also practical from both an operational and computational standpoint. In practice, bus scheduling is part of a complex series of decisions including route design, timetabling, and crew scheduling [15]. Agencies each have their own approaches for making these decisions and may be subject to important constraints such as driver labor policies that are not readily available to the public. As a result, existing bus blocks reflect agencies' comprehensive decisions after considering a full set of their resources and constraints, which should be respected/retained as much as possible to ensure the modified operations after converting to BEBs are feasible from an agency perspective. Our approach only alters bus schedules (and deploys backup buses) when necessary to ensure that all buses stay charged throughout their daily assignments, which reduces the operational changes that agencies must make to electrify their fleets. This is markedly different from approaches that aim to jointly optimize BEB charging and vehicle scheduling [41]. In practice, deploying backup buses implies extra cost (buses and drivers) to the agency, which may or may not be possible. In this case, the information provided by our model, such as infeasible blocks and infeasible trips, can be helpful: an agency can decide to continue to use diesel buses to serve these infeasible bus blocks, or even to revise the bus schedule for these infeasible blocks and infeasible trips. See Section 3.3.3 for more discussions on this.

A second key feature of this work is its emphasis on determining not only the locations but also the sizes of charging stations, along with a novel strategy of "queue prevention" which ensures that the number of buses assigned to charge simultaneously at any site never exceeds the number of chargers installed there. Current BEB charging infrastructure is expensive; in 2018, American transit agencies reported per-charger costs of between $380,000 and $1,000,000 per charger for high-power chargers [111]. Accordingly, limiting the number

of chargers built can save agencies millions of dollars. Simultaneously optimizing locations and sizes of charging sites promotes more efficient operations compared to other models that select charger sites without regard to the number of chargers built at each.

A third advantage of our approach is the promotion of more efficient charger usage by allowing deadhead driving to chargers, which allows a given charging site to serve a greater number of buses. Deadheading is penalized in our charger location model in order to encourage charging at the most convenient sites whenever possible. Fourth, we allow buses to charge for any duration within the scheduled layover time, an improvement over more restrictive models that assume predefined charging durations [116], discrete charging blocks of predefined length [4], or charging to full battery capacity any time a charger is visited [118]. Flexible partial recharging allows buses to charge for a long time when there is sufficient layover time scheduled, or to make multiple short charges during the day if necessary, appropriately fitting the charging time within the given block.

To simplify the models and solution methods, we decompose the charging decisions into three main steps. The first step is to identify bus blocks that need layover charging to finish their trips and "infeasible blocks", those blocks that cannot be sufficiently charged to serve all trips of their blocks even if charging is available at all candidate charging sites. We then introduce the BEB Block Revision Problem (BEB-BRP) to ensure buses in all infeasible blocks in the transit network can complete their blocks by using "backup" buses to serve some trips in the blocks if needed. Third, the BEB Optimal Charger Location (BEB-OCL) model can be used by transit agencies seeking to electrify their fleets to identify ideal locations for layover charging infrastructure. The mixed-integer linear programming (MILP) model identifies the locations at which to install charging infrastructure as well as the number of individual chargers to install at each site. Relying on widely available data and reflecting the structure of real-world bus schedules, BEB-OCL also determines the sequencing of charges for each bus in the system (i.e., where, when, and for how long each bus charges) to ensure buses have sufficient battery while keeping trips on schedule. The models are applied to a real-world case study of electrifying the 15 busiest bus routes (that includes hundreds of

buses serving thousands of trips each day) operated by King County Metro Transit as well as a sensitivity analysis of key parameters.

The remainder of this paper is structured as follows. Section 3.2 provides a detailed review of the existing literature on BEB charging facility location and related topics. Section 3.3 describes our methodology for the problem, including the BEB-BRP (Section 3.3.3) and BEB-OCL (Section 3.3.4) models. Section 3.4 applies the models to the actual bus network of King County, WA, where layover charging of BEBs is planned beginning in 2026. Section 3.5 presents a sensitivity analysis that explores the impact of varying key input parameters on model outputs including charger locations, cost components, and bus fleet size. Section 3.6 summarizes the findings and concludes the paper.

3.2 Literature Review

Charger location for BEBs is part of a larger transit planning process that involves many closely related decisions including route design, trip timetabling, and vehicle and driver scheduling. In traditional transit planning with diesel buses, these problems are addressed sequentially rather than simultaneously, as each step is already complex to optimize [15]. Replacing diesel buses with BEBs further complicates the traditional transit planning process by introducing new challenges related to capital investments in buses and chargers, location of chargers, possible schedule revisions, and charging scheduling [93]. Although many models for location of recharging (and, more generally, refueling) infrastructure for vehicles have been proposed in the literature [101], many of these efforts have focused on privately owned passenger vehicles rather than public transit applications. This difference in applications is important because BEBs run on fixed schedules along established routes and must adhere to published timetables to maintain good quality of service. Because of these key differences, some of the widely used modeling approaches for EV charger siting for private vehicles—which typically utilize estimated origin-destination demand and make assumptions about how traffic is distributed through a network [42, 65, 113, 117]—are not appropriate for the transit case.

This section provides an overview of past research on models for locating BEB charging infrastructure (Section 3.2.1), related work on scheduling problems for BEBs (Section 3.2.2), relevant commercial software available to transit agencies (Section 3.2.3), and approaches to incorporating charger capacity restrictions into EV infrastructure models (Section 3.2.4).

3.2.1 Locating BEB Chargers

A number of published works have used simulation to evaluate the feasibility of replacing existing diesel buses with BEBs and assess charging needs and system performance. De Filippo et al. [21] built a simulation model to explore the feasibility of converting bus routes at the Ohio State University to BEB service. They used a physics-based energy consumption model for each bus and a simple queuing model to examine the bus network's performance with different numbers of chargers and different charger power outputs. Rogge et al. [96] took a similar approach to study the potential electrification of a city bus network in Muenster, Germany. They assumed charging stations were located at the terminal stops of all bus routes and simulated bus operations with different combinations of charging power output and bus battery size in order to determine the technology requirements for all buses to continue operating on their existing schedules. Gao et al. [35] used a detailed vehicle simulation model to examine the feasibility and performance of electrifying bus systems. They explored the impact of different drive cycles, battery sizes, recharging power levels, and recharging strategies on service reliability, concluding that ultrafast chargers with appropriately selected (i.e., as small as possible) battery capacities offer the most cost-effective way to provide high-quality service with BEBs.

Rather than relying primarily on simulation, other works have applied optimization techniques (sometimes combined with simulation models) to establish BEB charging locations. The first category of such works assumes that buses are permitted to charge at intermediate stops on a trip while passengers are onboard, which may not be realistic from a passenger quality of service perspective. Wang et al. [116] proposed a relatively simple model to determine where to locate chargers at a subset of bus stops along various routes, assuming buses

gain a fixed amount of energy at any charger. Sebastiani et al. [99] developed a discrete-event simulation to estimate BEB energy consumption and recharging behavior. They used simulation-optimization to then minimize the total number of chargers and the additional time spent at stops devoted to charging. Xylia et al. [119] used optimization methods to identify locations for on-route fast charging stations in Stockholm's bus network, assuming that buses could charge for up to five minutes at any stop. These decisions were made alongside choices of the type of fuel used on a route; each could be served by battery-electric, biodiesel, or biogas buses. Kunith et al. [66] performed a similar study with a case study applied to Berlin. They used an optimization model to choose fast charging station locations as well as bus battery capacities. This model allowed charging at intermediate stops based on assumed dwell times of up to 40 seconds, as well as layover charging at terminals based on observed layover times of up to 21 minutes. Liu and Song [72] considered a different type of on-route charging in the form of in-roadway dynamic wireless charging infrastructure. Their model determined which road segments include wireless charging while simultaneously optimizing battery size and used robust optimization to account for uncertainty in energy consumption and travel time. Teichert et al. [104] used a mixed-integer nonlinear program to decide the battery size, charger power, and number of chargers for a single bus route. Their approach employed a simulation model to estimate some key parameters such as annual energy consumption and bus/battery lifetime.

Other works have approached BEB recharging needs from a similar perspective to this work, restricting charging operations to times when no passengers are onboard. Wei et al. [118] developed a mixed-integer linear programming model to replace a given number of buses with BEBs at minimum cost by deciding route assignments and the locations and sizes of both on-route layover charging stations and overnight charging sites at depots. They assumed buses could charge to full capacity during any layover period of 10 minutes or more, which is only appropriate for buses with small batteries and high-power chargers. Lin et al. [69] developed a multistage planning model to select the optimal locations and sizes of fast-charging stations based on bus operation and power grid characteristics. Demand was

aggregated at bus depots for charging when buses were out of service, and the model selected the number of individual chargers to build at each site based on how much of this demand was allocated to each site. The number of chargers should be seen as an upper bound as the model does not attempt to coordinate charging schedules and merely sets the number of chargers equal to the number of buses that charge at a particular site, independent of the time they charge at. An [4] developed a stochastic model for BEB charger location. The model assumes that vehicles may only charge between trips and that the existing schedule for diesel bus operations is maintained. Optimal charging station locations and BEB fleet size are identified based on random energy consumption and time-of-use electricity pricing. The model does not directly schedule charging operations for each bus under study; instead, it aggregates demand at the terminals based on battery level and locates charging stations to meet this demand. The capacity of charging stations is accounted for by establishing discrete 1-hour blocks of charging time that can be booked by buses and ensuring the total number of available blocks for a given time interval is not exceeded.

Table 3.1 summarizes the literature on optimization models for locating BEB charging infrastructure. The scope of this table is limited to works in which the location of conductive charging infrastructure is a key decision and locations are selected using an optimization approach. Each column of Table 3.1 describes one attribute of the modeling approach. *Objective* describes the cost components of the objective function to be minimized. Different works considered charger and related infrastructure (e.g., transformer [66]) cost in various forms, bus purchase costs, battery costs, and operations costs in various forms (e.g. energy cost, driver costs). *Locations* indicates the type of sites considered for charger locations. The options include existing bus stops (possibly restricted to terminals), bus depots/bases, transit centers, or *general* locations that could fall into the preceding categories or be a different type of site. Papers categorized as using general locations assume that a finite set of candidate charger locations is provided as an input to the optimization model, rather than assuming all sites of a particular type (e.g., bus stop) are possible charger locations. The *Sizing* column indicates whether the size of charging stations (in terms of the number of plugs/pantographs)

is optimized by the model, *Deadhead* indicates whether a model allows for buses to deadhead to chargers (rather than requiring them to charge at stops or terminals directly on-route), and *Blocks* indicates whether bus schedules are modeled as blocks. As indicated in Table 3.1, our BEB-OCL model considers general charging locations, optimizes the number of chargers at each site while respecting charger capacity limitations, allows deadheading, and inputs a generic block schedule rather than simply assuming buses all serve the same routes repeatedly. The objective of BEB-OCL is to minimize the cost of chargers and operations (in the form of added deadhead), while the complementary BEB-BRP model aims to minimize the cost of additional BEBs if the fleet size must be increased.

Table 3.1: Summary of literature on optimization models for BEB charger location.

Reference	**Objective**	**Locations**	**Sizing**	**Deadhead**	**Blocks**
An [4]	Ch, Bu, Op	General	No	Yes	Yes
Kunith et al. [66]	Ch, Ba	Stops	No	No	Yes
Lin et al. [69]	Ch, Op	General	Yes	Yes	No
Sebastiani et al. [99]	Ch, Op	Stops	No	No	No
Wang et al. [116]	Ch	Stops	No	No	No
Wei et al. [118]	Ch, Bu	Terminals, Depots	Yes	No	Yes
Xylia et al. [119]	Ch, Bu, Op	Stops, TCs	No	No	No
This paper	Ch, Bu, Op	General	Yes	Yes	Yes

Definitions. Ch: charger cost, Bu: bus cost, Ba: battery cost, Op: operations cost, TCs: transit centers.

3.2.2 Electric Vehicle Scheduling

Many recent works have focused on variants of the Electric Vehicle Scheduling Problem (EVSP). The EVSP extends the classical Vehicle Scheduling Problem (VSP), which aims to design daily vehicle schedules (blocks) to serve all timetabled trips at minimum cost. The primary driver of cost is the fleet size (number of vehicles), while the amount of deadheading between trips may also contribute to cost. Two trips i and j can be served sequentially only if there is sufficient time between the end of i and the start of j to deadhead between them and

allow for any required layover time [15]. The EVSP extends this framework by adding vehicle range constraints and recharging requirements. A thorough overview of work on the EVSP, as well as key sources for the VSP and some related problems in bus electrification, is given in Perumal et al. [93]. The EVSP in general is very difficult to solve and accordingly much of the literature focuses on heuristic methods for its solution. Among the references reviewed in Perumal et al. [93], none report solving instances with over 300 trips to optimality; larger scheduling problems are addressed only by heuristics without performance guarantees. This is especially the case when decisions about acquisition or location of charging infrastructure are included in an EVSP model. For example, Rogge et al. [97] formulated an optimization model that combines vehicle scheduling with optimization of bus fleet composition and the number of depot chargers (layover charging is not considered). They solved the scheduling problem with a genetic algorithm and the largest case study instance considered involves 200 trips along 3 bus routes. Recently, He et al. [41] proposed a joint optimization model to determine BEB charging locations, vehicle scheduling, and charging management, which was applied to a case study with three bus routes and 70 trips. Li et al. [68] considered both vehicle scheduling (based off origin-destination demands rather than trips) and charging infrastructure location, in which deadheading to chargers is allowed. The largest problem instance solved to optimality contains 12 distinct bus terminals and 288 time-varying origin-destination pairs. Liu and Ceder [71] also studied an EVSP variant in which they aimed to determine the number of fast chargers to install at each bus terminal in addition to bus schedules. They proposed an adjusted max-flow method as well as a heuristic based on deficit function theory to solve their bi-objective optimization model and applied these methods to case study instances that included up to 272 trips along three bus lines.

It is clear from the existing literature that the EVSP is in general a very difficult problem to solve and that adding charger location decisions generally yields a model that is too large to be solved exactly for urban bus networks. Our case study of the Seattle metro area in Section 3.4 includes 1,980 trips on 15 routes currently served by 132 diesel hybrid buses, which to our knowledge is larger than any EVSP instance solved with exact methods in the

literature, even if charger locations are set in advance. Hence, because of the complexity of electric vehicle scheduling, in this work we retain existing schedules whenever feasible. Our approach aims to fit charging events within scheduled layover time and to alter bus blocks only if they cannot feasibly be completed by BEBs.

3.2.3 Commercial Software

Several software companies offer products for planning electric bus fleet conversions. Most of these options focus on scheduling and recharging planning rather than infrastructure decisions. The Mobility House [105] advertises fleet design tools that incorporate charging infrastructure planning. Other companies including Giro, Inc. [37] (creator of HASTUS software), ChargePoint [16], IVU Traffic Technologies [48], and Optibus [90] all advertise software for planning BEB operations, including vehicle and recharging scheduling. However, the models these companies use do not appear to be publicly available and are generally focused more on daily operations or periodic bus scheduling rather than long-range infrastructure planning.

3.2.4 Charger Capacity and Queuing

A key element of this paper that some previous works have neglected is the impact of charger capacity on charging infrastructure location. Because a single charger plug (or pantograph) can only service one bus at a time, effective models must ensure that enough chargers are built so that buses can charge without enduring unexpected waits at busy chargers. Otherwise, multiple vehicles may be scheduled to use the same charger at the same time, resulting in delays for buses that must wait for others that arrived earlier. Such delays disrupt bus schedules and add uncertainty to bus operations. One approach to accounting for capacity in the BEB literature is to divide the time period under study into discrete intervals of a specified duration and schedule charges by having buses "book" one or more of these intervals [4]. This discretization makes it easy to formulate constraints that ensure no two vehicles are scheduled to use any charger at the same time, but it has some disadvantages. Requiring charge duration to be an integer multiple of some base duration imposes an artificial

restriction on charging time that may not produce a good solution. This undesirable feature can be mitigated by decreasing the size of the minimum interval, but doing so increases computational complexity.

Another possible approach to addressing capacity is to use results from the established field of queuing theory. Detailed queuing models based on stochastic processes sometimes give rise to closed-form expressions for key performance metrics, such as the average length of the queue or average time spent waiting in the queue [7]. Because queuing models are stochastic by nature, they are best suited for problems in which uncertainty is of key interest or some features of queuing processes are out of the decision-maker's control. For the latter case, e.g., the number of vehicles visiting a public EV charging station each day can be viewed as a random variable, as this outcome depends on the decisions of individual drivers over which the charger owner has very little influence. In the BEB case, on the other hand, transit agencies have the ability to schedule charger usage themselves, so charger utilization is the direct outcome of the agency's decisions and treating the charger queuing process as fully random may not be appropriate.

Perhaps for this reason, there do not appear to be any examples in the currently published literature of queuing models incorporated into optimization models for locating BEB chargers. However, several authors have applied the results of queuing theory to inform models of electric passenger vehicle charging. We next briefly review these works to motivate the approach used in this paper (as described in Section 3.3). Many of these examples come from the literature on the EVRP or electric taxi operations. In Keskin et al. [53], an M/G/1 queuing model was used at each charger when deciding which public charging stations a fleet of EVs should use in order to complete a set of deliveries with time windows at minimum cost. Keskin et al. [54] proposed a similar model for EV routing with time windows that includes a recourse procedure for preserving feasibility when large queue delays would make deliveries late. An M/G/1 queuing model is again used to calculate expected waiting times in a first-stage decision problem, while a second-stage model uses simulation to obtain exact waiting times for each charger visit and construct a new feasible solution using the recourse

procedure.

Besides queuing models and simulation, some works take different approaches to accounting for charger capacity in optimization models. Sweda et al. [103] proposed an adaptive routing problem for EVs in which each potential charging station has a specified probability of being either available or occupied; if the station is occupied, the driver must wait for some amount of time that is itself a random variable. For a type of EVRP, Froger et al. [33] developed a formulation that ensures charging station capacities are not exceeded by adding constraints based on the Resource Constrained Scheduling Problem. Ding et al. [23] added constraints to the EVRP that prohibit any two vehicles from utilizing a charger at the same time. These constraints are nonlinear, but can be linearized by introducing binary variables and additional constraints.

All of the aforementioned charger capacity models focus on EV operations when chargers have already been constructed in specific locations. There are not as many works that incorporate a queuing model into an optimization model that identifies locations for chargers. Jung et al. [52] considered charger queues in an optimization problem for locating charging infrastructure. They developed a bilevel optimization model with an upper-level problem of locating electric taxi chargers modeled as a queuing network and a lower-level simulation model that was used to evaluate queue times. The model was solved iteratively until the queue estimates in the upper model agreed with the simulation output from the lower model. Similarly, Yang et al. [120] used an M/M/x/s queuing model in a charging infrastructure location problem for EV taxis. They used this queuing model to estimate the probability that each charger is available when a taxi intends to use it and added a constraint to their optimization model to ensure the probability of charging at some point throughout the day is sufficiently large. These two models are the only ones identified in the EV literature that combine the consideration of queue wait times with facility location.

3.2.5 Summary

In the past several years, researchers have developed diverse models for BEB charging infrastructure optimization. As BEBs are still a relatively new technology that have not seen widespread large-scale deployment, studies tend to consider different technologies (e.g. plug-in vs. wireless charging) and strategies (e.g. slow overnight charging vs. fast on-route or layover charging) and make different assumptions. The scope of decisions to be made must be clearly defined for any optimization model, and including all decisions related to infrastructure and operations of an electric bus system would not be computationally tractable, so researchers have taken on different subproblems related to bus fleet electrification. For instance, several authors have addressed the problem of developing blocks for BEBs through the EVSP and its variants, but these are fundamentally complex models that are very difficult to solve at the scale of urban transit systems. So, in this work and most existing research on BEB charging infrastructure location, bus schedules and blocks are treated as fixed.

Some limitations are present throughout the published literature on charger location for BEBs. One such limitation is neglecting adherence to vehicle schedules. For example, some works (e.g. Sebastiani et al. [99], Xylia et al. [119]) allow charging at all bus stops along a route while passengers are onboard, which may result in delays. Alternatively, Wei et al. [118] only allows charging to take place in between passenger trips, but makes the restrictive assumption that buses can always fully charge in between trips. This is not realistic unless batteries are small and charger power is high, or excessive layover time is built into the blocks.

A second significant limitation is that the limited capacity of individual charging facilities and the resulting impact on the number of chargers needed is often neglected. Many works assume a charger plug or pantograph is always available when a vehicle needs to charge and do not quantify either the number of chargers needed to avoid charger queues or the impact of departure delays due to queuing. Works that do model charging station capacity still suffer from shortcomings; Lin et al. [69] assigns exactly one charger for each bus that charges

at a particular location, which is not efficient for layover charging, while An [4] and Wei et al. [118] use discretization approaches that may also be overly restrictive as buses are assumed to charge for a full time interval of predefined duration.

A third limitation is that models often assume chargers are located on-route, even if layover charging is taking place. That is, buses can only charge at depots and trip terminals where they will already be located at some point in the day and cannot deadhead to any other chargers. This assumption prevents bus operators from making the most of their infrastructure investments by sharing them among a larger number of buses. While deadheading a long distance to visit a charger may not always be practical, multiple trip terminals are often located close together and it can be more cost-effective to have buses in this area share a common charging site rather than install one at each terminal.

In this paper, we propose a new approach to optimizing BEB charging infrastructure that is unique compared to the currently published literature. First, we focus specifically on layover charging in which deadheading is permitted. Second, we base our model on actual bus block structures and ensure that charging only happens in between trips within the scheduled layover time so charging does not introduce delays. Third, we directly account for the capacity of charging stations and ensure an appropriate number of chargers is installed at each site. We do so with a novel concept and modeling technique of *queue prevention* to avoid queue delays at chargers. Section 3.3 next describes the methodology in detail.

3.3 Methodology

3.3.1 Overview

As discussed in Section 3.1, we propose a modeling approach for BEB layover charging that maintains the original block structure as much as possible, while deploying backup buses (which may be BEBs as well) only if needed due to insufficient layover time. Our approach consists of three related models/algorithms as illustrated in Figure 3.2. The primary focus of this work is the BEB Optimal Charger Location (BEB-OCL) model (Section 3.3.4), but we

also propose the BEB Feasibility Check (BEB-FC, Section 3.3.2) and BEB Block Revision Problem (BEB-BRP, Section 3.3.3) to check layover charging feasibility and optimize the backup buses, while ensuring that any transit agency's schedule and blocks can be analyzed with our approach. The process begins by identifying layover blocks, those blocks with total energy demand that exceeds the usable battery capacity of the buses assigned to them, so that they can only be completed with layover charging. Any other blocks that can be completed with overnight depot charging only are excluded from the analysis. BEB-FC then checks whether each layover block can feasibly be completed by a BEB with specified parameters (e.g., energy consumption per trip, battery size, and distance to layover charging stations). A block is *infeasible* for layover charging if no possible choice of charger locations would allow a bus serving that block to be sufficiently charged throughout the day.

All infeasible blocks are supplied as inputs to BEB-BRP. We assume that the transit agency will modify the blocks that are infeasible by serving some of their trips with additional buses, thereby increasing the total fleet size compared to the pre-electrification scenario. BEB-BRP calculates the minimum number of additional buses that would be required to make all of these blocks feasible, giving a better estimate of total costs. On the other hand, all feasible blocks are included as inputs to BEB-OCL, which determines the optimal layover charger locations and a charging schedule for each bus in order to minimize the total capital and operational costs associated with charging.

Definitions of all sets and parameters used in BEB-FC, BEB-BRP, and BEB-OCL are provided in Table 3.2. We consider a set of BEBs $\mathcal{V}$, where each vehicle $v \in \mathcal{V}$ is assigned to a layover block that consists of K_v trips, represented as a set $\mathcal{T}_v = \{1, \ldots, K_v\}$. We are also given a set of candidate charging sites $\mathcal{S}$, a subset of which will be selected for charging infrastructure. Decision variables unique to each of the two optimization models are defined in the corresponding sections.

Our methodology relies on the following assumptions throughout BEB-FC, BEB-BRP, and BEB-OCL. Additional assumptions relevant to only individual models are documented in Sections 3.3.3-3.3.4.

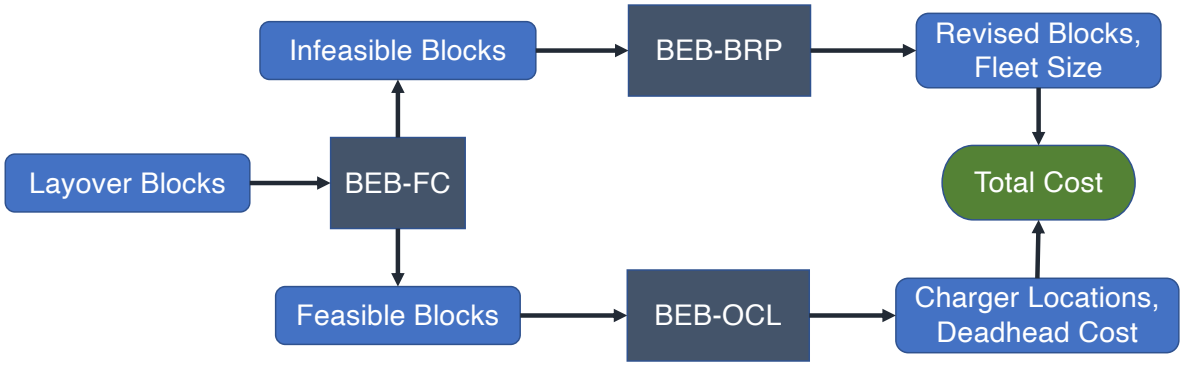


Figure 3.2: Flowchart showing the relationship between the algorithms and optimization models introduced in this study.

1. Bus schedules and blocks, as well as the assignment of specific vehicles to blocks, are given. BEBs may have different characteristics (e.g. battery capacity, energy consumption per mile), but the attributes of the bus serving each block are given.
2. Buses must operate on schedule. Charging can only take place during scheduled layover time.
3. Driving times and energy consumption, both for scheduled bus trips and deadheading to/from chargers, are known deterministically.
4. Buses enter service each day on time and with fully charged batteries.
5. Battery charge is gained as a linear function of time.

Assumption 1 reflects our belief that it is best to maintain existing bus blocks whenever possible, as well as the widespread availability of block data through standard formats such as GTFS. Assumption 2 is included in order to ensure the current passenger level of service is maintained. Assumption 3 helps limit the complexity of the models and make them practical to solve at real-world scale. While energy consumption and driving times will vary somewhat from day to day, using conservative estimates of these values should help the robustness of the models. Assumption 4 reflects the common practice of using slow overnight charging at depots. Assumption 5 should provide a good enough approximation even if the true

behavior is nonlinear, especially if the upper bound on battery level $\overline{u}_v$ is not too high, since the charging rate is typically constant until the state of charge exceeds about 80% [84].

3.3.2 Block Feasibility Check BEB-FC

To ensure that enough chargers are built to serve all BEBs, BEB-OCL requires that the battery level of each BEB stays between its lower and upper limits at all times (as described in Section 3.3.4). A consequence of this modeling approach is that including any infeasible blocks as an input to BEB-OCL would make the complete model infeasible. To avoid this issue, we apply the BEB Feasibility Check (BEB-FC) as described here to separate feasible and infeasible blocks, as well as BEB-BRP (Section 3.3.3) to modify infeasible blocks (using backup buses) so they can be served by BEBs. The use of BEB-FC and BEB-BRP in addition to BEB-OCL enables studying bus systems in which not all blocks can be served with BEBs as-is and testing model sensitivity over a greater range of parameter values.

Algorithm 1 describes the BEB-FC procedure that is applied to each bus $v \in \mathcal{V}$. Algorithm 1 iterates through all trips in bus v's block and aims to maximize charge level at all times, assuming that all candidate charger locations are available. If the bus always behaves in order to maximize its charge and still runs out of battery at some point, then we know this block cannot be served by any bus with the given parameters and available charging station locations. We therefore categorize it as an infeasible block and supply it as an input to BEB-BRP as illustrated in Figure 3.2.

In Algorithm 1, the return value *feas* is a Boolean that takes value `TRUE` if the block is feasible and `FALSE` otherwise. Δu_t^s denotes the battery level increase that results from charging at each potential site s for as long as possible after trip t. The other terms used in Algorithm 1 are as defined in Table 4.1, though we drop the bus index v for brevity since Algorithm 1 is applied to one bus at a time.

Lines 1-3 of Algorithm 1 initialize variables so that analysis begins with the first trip for each bus, the battery starts fully charged, and *feas* is set to `TRUE`. The algorithm then iterates over all trips in a block and ensures that the charge level u_t is always kept as high as

Algorithm 1: BEB-FC: Feasibility Check for a Single Bus

1 $t \leftarrow 1$
2 $u_t \leftarrow \overline{u}$
3 *feas* $\leftarrow$ TRUE
4 **while** $t \leq K$ **and** *feas* == TRUE **do**
5 **for** $s \in \mathcal{S}$ **do**
6 **if** $u_t - \Delta_t - \delta_t^{1s} < \underline{u}$ **then**
7 $\Delta u_t^s \leftarrow -\overline{u}$
8 **else**
9 $\Delta u_t^s \leftarrow \min\{\rho^s \overline{y}_t^s - \delta_t^{1s} - \delta_t^{2s},\ \overline{u} - (u_t - \Delta_t - \delta_t^{1s}) - \delta_t^{2s}\}$
10 **end**
11 **end**
12 $s_{\max} \leftarrow \arg\max_s \Delta u_t^s$
13 **if** $\Delta u_t^{s_{max}} \leq -\delta_t^3$ **then**
14 $u_t^{\min} \leftarrow u_t - \Delta_t - \delta_t^3$
15 $u_{t+1} \leftarrow u_t - \Delta_t - \delta_t^3$
16 **else**
17 $u_t^{\min} \leftarrow u_t - \Delta_t - \delta_t^{1 s_{\max}}$
18 $u_{t+1} \leftarrow u_t - \Delta_t + \Delta u_t^{s_{\max}}$
19 **end**
20 **if** $u_t^{min} < \underline{u}$ **then**
21 *feas* $\leftarrow$ FALSE
22 **end**
23 $t \leftarrow t + 1$
24 **end**
25 **return** *feas*

possible. Lines 5-11 calculate the charge difference Δu_t^s. Lines 6-7 set the charge difference to the smallest possible value if deadheading to site s would cause the bus to run out of battery, so that this site will not be chosen for charging. Lines 8-9 set the charge difference when s is within reach. Δu_t^s is the charge gain $\rho^s \overline{y}_t^s$ minus the deadhead energy $\delta_t^{1s} + \delta_t^{2s}$, unless charging for the full duration $\overline{y}_t^s$ would exceed the battery capacity $\overline{u}$, in which case as much charge as possible without exceeding the capacity is gained.

After identifying the charging site $s_{\max}$ offering the maximum charge gain (which maximizes u_{t+1}) in Line 12, Line 13 of Algorithm 1 checks whether it is preferable to not charge at

all and simply drive the deadhead distance δ_t^3 to the next trip. Lines 14-19 then appropriately set the charge level at the start of trip $t+1$ (u_{t+1}), as well as the minimum charge level $u_t^{\min}$ attained an any point during trip t, depending on whether or not visiting a charger results in a greater battery level. In Lines 20-22, the block is marked as infeasible if the minimum charge for the current trip is less than $\underline{u}$; otherwise, the algorithm proceeds to the next trip (Line 23). Algorithm 1 is applied to all buses $v \in \mathcal{V}$ independently in order to populate the set of infeasible buses $\mathcal{V}^{IF}$. The set of feasible buses is likewise defined as $\mathcal{V}^F = \mathcal{V} \setminus \mathcal{V}^{IF}$.

3.3.3 Block Revision Problem BEB-BRP

Premise, Terminology, and Assumptions

If some blocks are not feasible for layover charging, transit agencies may choose to respond in various ways. They could simply cancel some trips to allow more time for layover charging, but doing so would result in poor service for travelers. Alternatively, they may choose to serve infeasible blocks with conventionally fueled buses rather than BEBs, choosing only to dispatch electric buses on feasible blocks. But if agencies want to serve all layover blocks with BEBs and complete the same trips as were scheduled for conventional buses, they must revise the infeasible blocks in order to make it feasible for BEBs to complete them. Doing so will most likely require increasing the bus fleet size. In this work, we assume that agencies faced with infeasible blocks will revise them so that they can be completed by BEBs, but aim to change the blocks as little as possible and to increase the fleet size as little as possible. Based on this premise, this section presents the BEB-BRP model to handle infeasible blocks and estimate the cost of electrifying them. The key assumptions of BEB-BRP are as follows:

1. A sufficiently large fleet of additional "backup" buses is available to serve some trips from any block that does not include sufficient layover time for all necessary charging to be done.
2. Infeasible blocks are revised so that no layover charging is necessary; enough trips will be served by backup buses so that overnight depot charging alone is sufficient.

3. Any passenger trip can be served either as part of an original block or by a backup bus (i.e., part of a backup block).
4. Backup buses are dispatched from a single depot.

Assumption 1 is included to ensure the feasibility of the model. If the agency is not willing to increase fleet size at all, it is likely not possible to use BEBs. Assumption 2 is made to reduce the complexity of the model and reflects the reasonable likelihood that if agencies are already revising schedules, they will develop new blocks that avoid the necessity of layover charging. Assumption 3 reflects the expectation that the assignment of specific vehicles to specific trips is not important. If certain trips needed to be completed by the same vehicle (for example, to reduce passengers' transfer needs when a vehicle serves two different routes consecutively), it would be straightforward to add corresponding constraints to BEB-BRP, but we expect this is not usually the case. Assumption 4 is appropriate for small to medium BEB deployments where buses are located at a single depot; where necessary, this assumption could be relaxed by generalizing BEB-BRP in a similar fashion as the Single-Depot Vehicle Scheduling Problem is extended to the Multi-Depot VSP [22].

The key inputs to BEB-BRP are a set of infeasible blocks $\mathcal{V}^{IF}$ and a set of backup buses $\mathcal{B}$. In the model formulation, we also refer to the infeasible blocks as "original blocks" for clarity, since they will be modified to be made feasible. Each infeasible block $u \in \mathcal{V}^{IF}$ consists of an ordered set of trips $i = 1, \ldots, K_u$. Any unique trip can therefore be identified by a tuple (u, i) . Each backup bus $b \in \mathcal{B}$ can be assigned to any trip, but two trips (u, i) for $u \in \mathcal{V}^{IF}$, $i \in \mathcal{T}_u$ and (v, j) for $v \in \mathcal{V}^{IF}$, $j \in \mathcal{T}_v$ can only be served successively by the same backup if the arc (u, i, v, j) is contained in a feasible set $\mathcal{F}$. The idea of the feasible set is adopted from a common modeling approach to the VSP [15, 22]. For two trips (u, i) and (v, j), let the deadhead distance between them be δ^*_{uivj} and the deadhead time be τ^*_{uivj}. Trips (u, i) and (v, j) are *compatible* if a backup bus can complete (v, j) immediately after (u, i); that is, the difference between the start time of (v, j) and the end time of (u, i) minus the deadhead time and a minimum layover time L must be nonnegative. Using our terminology,

$$\mathcal{F} = \left\{(u,i,v,j) : u,v \in \mathcal{V}^{IF}, i \in \mathcal{T}_u, j \in \mathcal{T}_v,\ \ \sigma_{vj} - \epsilon_{ui} - \tau^*_{uivj} - L \geq 0\right\} \tag{3.1}$$

We can view BEB-BRP (like the VSP) as being defined over a network where nodes represent trips and arcs in $\mathcal{F}$ represent deadhead connections between compatible trips. Figure 3.3 provides an example of this representation for a simple problem instance in which there are three infeasible blocks, i.e., $\mathcal{V} = \{1,2,3\}$. Figure 3.3(a) shows the start and end times of the 9 trips in this example represented on a timeline. Figure 3.3(b) shows the corresponding network. To make this example as simple as possible, in Figure 3.3 it is assumed that there is no layover requirement ($L = 0$) and that there is zero deadhead time between any pair of trips ($\tau^*_{uivj} = 0$ for all $(u,i,v,j) \in \mathcal{F}$).

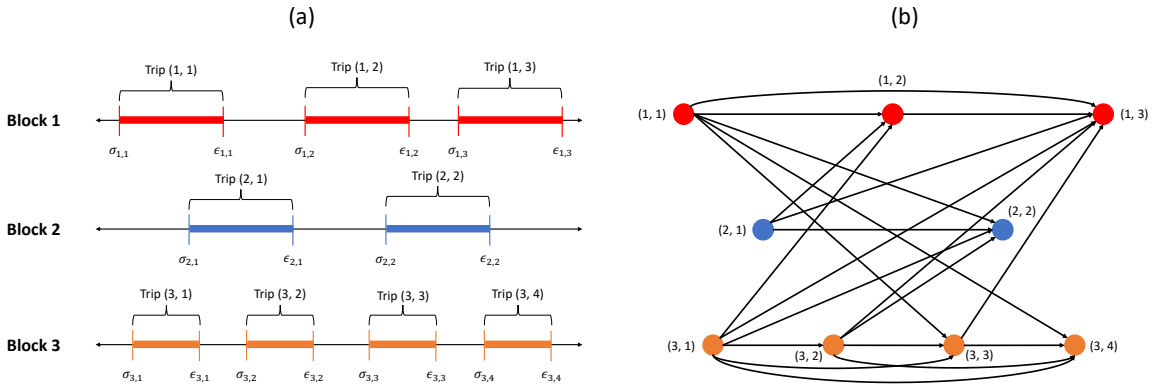


Figure 3.3: Example consisting of 3 infeasible blocks (a) and the corresponding network for BEB-BRP (b).

In addition to the set $\mathcal{F}$ of compatible arcs, we define the complete arc set $\mathcal{A}$ to consist of all arcs in $\mathcal{F}$ as well as arcs that connect each trip node to the depot, which is represented as a dummy vehicle and trip $v = 0, t = 0$. That is, $\mathcal{A} = \mathcal{F} \cup \{\{(0,0)\} \times \mathcal{T}\} \cup \{\mathcal{T} \times \{(0,0)\}\}$. Then, to formulate BEB-BRP as an integer program, we define binary decision variables Z^b_{uivj} that equal 1 if backup bus b serves trip (v,j) immediately after (u,i) and 0 otherwise. We also define binary decision variables X_{uij} that equal 1 if original bus u serves trip j

immediately after trip i and 0 otherwise.

BEB-BRP Formulation

The BEB-BRP integer program formulation is as given in Equations (3.2)-(3.9).

$$\min \quad \sum_{b\in\mathcal{B}}\sum_{u\in\mathcal{V}}\sum_{i=1}^{K_u} Z^b_{00ui} \tag{3.2}$$

$$\text{s.t.} \quad \sum_{i\in T_u}\left[\sum_{j=i+1}^{K_v}\left(\Delta_{ui}+\delta^*_{uiuj}\right)X_{uij}\right] \le \overline{u_u}-\underline{u_u} \qquad \forall u\in\mathcal{V}^{IF} \tag{3.3}$$

$$\sum_{(u,i,v,j)\in\mathcal{A}}\left(\Delta_{ui}+\delta^*_{uivj}\right)Z^b_{uivj} \le \overline{u_b}-\underline{u_b} \qquad \forall b\in\mathcal{B} \tag{3.4}$$

$$\sum_{j=i+1}^{K_u} X_{uij} + \sum_{b\in\mathcal{B}}\sum_{\substack{v,j:\\(u,i,v,j)\\\in\mathcal{A}}} Z^b_{uivj} = 1 \qquad \forall u\in\mathcal{V}^{IF}, i\in\mathcal{T}_u \tag{3.5}$$

$$\sum_{j=i+1}^{K_u} X_{uij} = \sum_{j=1}^{i-1} X_{uji} \qquad \forall u\in\mathcal{V}^{IF}, i\in\mathcal{T}_u \tag{3.6}$$

$$\sum_{\substack{v,j:\\(u,i,v,j)\\\in\mathcal{A}}} Z^b_{uivj} = \sum_{\substack{v,j:\\(v,j,u,i)\\\in\mathcal{A}}} Z^b_{vjui} \qquad \forall b\in\mathcal{B}, u\in\mathcal{V}^{IF}, i\in\mathcal{T}_u \tag{3.7}$$

$$X_{uij}\in\{0,1\} \qquad \forall u\in\mathcal{V},\ \ i=1,\ldots,K_u,\ \ j=i+1,\ldots,K_u \tag{3.8}$$

$$Z^b_{uivj}\in\{0,1\} \qquad \forall b\in\mathcal{B},\ \ (u,i,v,j)\in\mathcal{A} \tag{3.9}$$

The objective function (3.2) aims to minimize the total number of backup buses used (i.e., the number of backup buses that leave the depot). Constraints (3.3) ensure that all original bus blocks are feasible based on trip and deadhead energy requirements and the variables X_{uij} that indicate which trips are still served in the original block. Analogously, Constraints (3.4) ensure all backup blocks are feasible. Constraints (3.5) require that every trip is included in either an original block or a backup block. Constraints (3.6) ensure that all original blocks are fully connected. For every trip i on original block u, the number of arcs exiting trip node (u, i) can be expressed as $\sum_{j=i+1}^{K_u} X_{uij}$. This sum must be equal to the number of arcs entering this same node, $\sum_{j=1}^{i-1} X_{uji}$. Note that each summation is restricted to be either 0 or 1 as a result of Constraints (3.5). Constraints (3.7) enforce equivalent logic for all backup blocks. Constraints (3.8)-(3.9) establish that decision variables are binary.

Solution Method

The BEB-BRP integer program (3.2)-(3.9) is difficult to solve directly for problems with a large number of infeasible blocks because the number of compatible arcs $|\mathcal{A}|$ grows exponentially with the number of trips. In order to solve BEB-BRP at scale, we developed the heuristic Algorithm 2 to find reasonable feasible solutions. The algorithm consists of two main stages. In the first stage, a subset of trips is removed from each infeasible block so that the shortened block is feasible. For each bus, one trip at a time is removed from its block and its energy demand recalculated until it is less than the usable battery capacity of the bus. Each removed trip is added to the set $\mathcal{T}_{BU}$ of trips to be served by backup buses. In the second stage, these trips are allocated to backup buses to construct backup blocks. This portion of the algorithm moves through the backup trips $\mathcal{T}_{BU}$ in increasing order of departure time, adding each trip to an existing block where possible, or creating a new backup block if no existing one is compatible. For this approach, we rely on one additional assumption: the fleet of backup buses is homogeneous, with identical battery capacities and energy consumption parameters.

We introduce some new notation to express the algorithm compactly. For an original

block v, let $v[0]$ and $v[-1]$ be the first and last trip, respectively. Likewise, for a backup block b, let $b[0]$ and $b[-1]$ be the first and last trip, respectively. For some trip t in original block v, let δ^*_{vt} be shorthand for the deadhead distance from the end of trip (v,t) to the start of the next trip in $\mathcal{T}_v$ (which may not be trip $t+1$, if $t+1$ is served by a backup). If t is the last trip in $\mathcal{T}_v$, then $\delta^*_{vt} = 0$. Let ξ_v (or ξ_b) be the total energy demand of an original (or backup) block, including all passenger trips and deadheading. Then the heuristic solution method for BEB-BRP is presented in Algorithm 2.

Algorithm 2: Heuristic Solution Method for BEB-BRP

$\mathcal{T}_{BU} \leftarrow \emptyset$
for $v \in \mathcal{V}^{IF}$ **do**
  $\xi_v \leftarrow \delta^*_{00v[0]} + \sum_{\mathcal{T}_v} (\Delta_{vt} + \delta^*_{vt}) + \delta^*_{v[-1]00}$
  while $\xi_v > \overline{u_v} - \underline{u_v}$ **do**
    $t \leftarrow v[-1]$ **or** $t \leftarrow v[0]$
    $\mathcal{T}_v \leftarrow \mathcal{T}_v \setminus \{t\}$
    $\mathcal{T}_{BU} \leftarrow \mathcal{T}_{BU} \cup \{(v,t)\}$
    $\xi_v \leftarrow \delta^*_{00v[0]} + \sum_{\mathcal{T}_v} (\Delta_{vt} + \delta^*_{vt}) + \delta^*_{v[-1]00}$
  end
end
$\mathcal{B} \leftarrow \emptyset$
$b \leftarrow 0$
for $(v,t) \in \mathcal{T}_{BU}$ **do**
  $\mathcal{B}^F_{vt} \leftarrow \{b \in \mathcal{B} : (b[-1], v, t) \in \mathcal{F}\}$
  $\mathcal{B}^F_{vt} \leftarrow \mathcal{B}^F_{vt} \cap \{b \in \mathcal{B} : \xi_b + \delta^*_{b[-1]vt} + \Delta_{vt} + \delta^*_{vt00} \leq \overline{u_b} - \underline{u_b}\}$
  if $\mathcal{B}^F_{vt} = \emptyset$ **then**
    $b \leftarrow b + 1$
    $\mathcal{B} \leftarrow \mathcal{B} \cup \{b\}$
    $\mathcal{T}_b \leftarrow \{(v,t)\}$
    $\xi_b \leftarrow \delta^*_{00vt} + \Delta_{vt}$
  else
    $b^* \leftarrow b \in \mathcal{B} : \epsilon_{b[-1]} = \max\{\epsilon_{b[-1]} \ \forall b \in \mathcal{B}\}$
    $\mathcal{T}_{b^*} \leftarrow \mathcal{T}_{b^*} \cup (v,t)$
    $\xi_{b^*} \leftarrow \xi_{b^*} + \delta^*_{b[0]vt} + \Delta_{vt}$
  end
end
return i

As mentioned previously, Algorithm 1 consists of two separate stages. First, in lines 1-10, we populate a set $\mathcal{T}_{BU}$ of trips that will be served by backup buses. To do so, we iterate through all blocks and repeatedly remove trips from either the start or end of them to reduce their total energy demand until it no longer exceeds the usable battery capacity. Line 1 initializes the set of backup trips $\mathcal{T}_{BU}$ to be empty. For each infeasible bus, Line 3 initializes the total energy demand as the sum of passenger trip and deadhead energy, including deadheading to and from the depot. For as long as this energy demand exceeds the usable battery capacity, Lines 5-7 remove the last (or first) trip from the original block b and add it to $\mathcal{T}_{BU}$. Line 8 updates the total energy consumption of block v after this trip has been removed by summing the energy requirement of all trips and deadheading completed.

Second, in lines 11-26, Algorithm 2 constructs blocks for backup buses by assigning the trips in $\mathcal{T}_{BU}$ to them. We sort $\mathcal{T}_{BU}$ by trip start times, so that the first element has the earliest start time σ_{vt}. For each trip $(v,t) \in \mathcal{T}_{BU}$, let $\mathcal{B}^F_{vt}$ be the set of all backup blocks that can feasibly serve this trip; that is, backup b can serve (v,t) if the last trip currently served by b is compatible with (v,t) (line 14), and if a bus serving b can complete (v,t) and then return to the depot without exceeding its energy capacity (line 15). If there is no such backup block, we create a new block by increasing the block index (line 17), adding it to the set of backups (line 18), initializing block b to contain just the current trip (line 19), and set its energy consumption to this point (line 20). Otherwise, we choose the block $b^* \in \mathcal{B}^F_{bt}$ that ends latest (line 22), append trip (v,t) to b^* (line 23), and update the total energy for b^* (line 24). The algorithm outputs the final value of b, i.e., the total number of backups used (the BEB-BRP objective). In our work, we run Algorithm 2 twice for each BRP instance, with each run corresponding to the choice of removing trips from either the start or end of every original block in Line 5. Whichever choice gives the smaller number of backup buses is taken as our solution.

In practice, an agency may not have the resources needed to deploy backup buses. In this case, we note here that the BEB-BRP solution, i.e., the trips identified to be served by backup buses, together with the infeasible blocks identified by BEB-FC, can collectively provide

useful information if an agency prefers to have alternative solutions (instead of deploying backup buses). For example, the agency can decide to continue to serve all these infeasible blocks by original diesel buses. The agency can also revise bus schedules by removing the backup trips $\mathcal{T}_{BU}$ identified by BEB-BRP or increasing layover times for these trips so that BEBs can be sufficiently charged. Such schedule revisions are expected to be easier than optimizing the schedules of all buses.

3.3.4 Optimal Charger Location Model BEB-OCL

Premise, Terminology, and Assumptions

The charging infrastructure location model BEB-OCL is designed to identify the locations where layover charging stations should be built as well as make operational decisions about how charging is performed. BEB-OCL assumes that the transit operator has identified a set of candidate locations for chargers at existing bus stops or layover facilities, or in undeveloped lots, none of which are necessarily on a bus's route. In order to maintain a high level of service for passengers, buses may only charge in between trips when no passengers are on board and must adhere to published schedules. Upon completion of any trip, buses may drive a known distance to visit a charging station and charge for any (nonnegative) amount of time before driving to the start location of the subsequent trip, as long as the subsequent trip departs on time. In addition to the shared assumptions of Section 3.3.1, BEB-OCL relies on the following further assumptions:

1. The power level of chargers that would be installed at each site is predetermined and all chargers located at a given site have the same power output.
2. As much power as is needed at any given charging location can be delivered to vehicles charging there. That is, no electric grid constraints are imposed.
3. Each vehicle may use only a single charger in between any pair of trips in its block.

The binary decision variable X^s indicates whether a charging station is ($X^s = 1$) or is not ($X^s = 0$) chosen to be located at site $s \in \mathcal{S}$, while decision variable N^s (restricted to

nonnegative integers) represents the number of individual chargers installed at site s. The binary decision variable Y_{vt}^{s} indicates whether bus v does ($Y_{vt}^{s} = 1$) or does not ($Y_{vt}^{s} = 0$) charge at site s after completing trip $t \in \mathcal{T}_v$; likewise, the continuous decision variable y_{vt}^{s} represents the duration of this charge. We define u_{vt} as the battery charge level of vehicle v at the start of trip t.

To ensure that buses always run on schedule, we pre-calculate the maximum feasible duration of charging at each candidate site after all trips made by each bus. This maximum charge duration depends on the location s; longer deadhead times to and from the charger reduce the time available for charging. We calculate the time available for charging as the difference between the scheduled layover time $\sigma_{v,t+1} - \epsilon_{vt}$ and the deadhead time to and from each candidate charging infrastructure site s, $\tau_{vt}^{1s} + \tau_{vt}^{2s}$. We also assume that the transit operator has established some maximum possible layover charging duration $y_{\max}$, which helps limit the size of the conflict sets introduced later in Section 3.3.4. Combined with the fact that this bound must be nonnegative, the charge duration limits are calculated according to Equation 3.10, which also allows for charging up to $y_{\max}$ after the last trip of the day.

$$\overline{y}_{vt}^{s} = \begin{cases} \min\left\{y_{\max}, \max\left[0, \sigma_{v,t+1} - \epsilon_{vt} - \tau_{vt}^{1s} - \tau_{vt}^{2s}\right]\right\} & t = 1, \ldots, K_v - 1 \\ y_{\max} & t = K_v \end{cases} \tag{3.10}$$

Figure 3.4 presents an illustrative example of calculating $\overline{y}_{vt}^{s}$ for a single trip. In this simple case, we consider how much time is available for bus v to charge in between trips 1 and 2, where the second trip begins at the same location as the first trip ends. There is 30 minutes of scheduled layover time in between trips ($\sigma_{v2} - \epsilon_{v1} = 30$ min.). Deadheading (indicated by a black dashed arrow) to and from charging site A takes 15 minutes, whereas deadheading to and from Site B takes 5 minutes. Thus, the maximum feasible charging durations are 15 minutes at Site A and 25 minutes at Site B. Note that since the number of trips and the number of candidate sites are finite (and not huge even for a large size

network), the above simple calculation process can be done in negligible time.

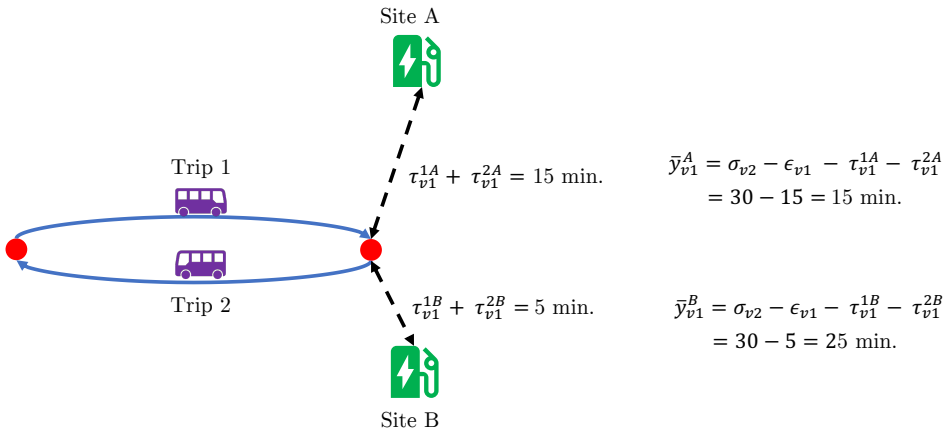


Figure 3.4: Example showing calculation of maximum charge time.

Sections 3.3.4-3.3.4 introduce the full BEB-OCL model for sizing and locating layover charging sites.

Objective Function

$$\min \quad \sum_{s \in \mathcal{S}} \left(f^s X^s + g^s N^s \right) + \alpha \sum_{v \in \mathcal{V}} \sum_{t \in \mathcal{T}_v} \sum_{s \in \mathcal{S}} \left(\tau_{vt}^{1s} + \tau_{vt}^{2s} - \tau_{vt}^{3} \right) Y_{vt}^{s} \tag{3.11}$$

The objective function (3.11) seeks to minimize the cost of purchasing and installing chargers while keeping charging operations efficient (i.e., limiting the deadhead time due to charging). We assume that installing any chargers at site s incurs a fixed setup cost f^s that is independent of the number of chargers installed. This setup cost reflects the expenses required to prepare a site to host charging infrastructure, which could include land purchase/leasing costs, required electric grid upgrades, etc. Additionally, each charger that is installed at site s incurs a per-unit cost g^s, which reflects the sale price and installation

cost of a charger. Accounting for both of these components, the total capital cost of charging infrastructure can be expressed as $\sum_{s\in\mathcal{S}}(f^s X^s + g^s N^s)$.

The second component of the objective function reflects operational performance of the bus system and ensures that charging is done as efficiently as possible. A user-specified parameter α controls the weighting of these two components; it can be conceived of as the monetary value of a single unit of deadhead time each day over the lifetime of the infrastructure. This component seeks to minimize the net increase in deadhead time due to charging, which helps limit driver pay expenses and generally incentivizes less deadheading which may introduce delays due to traffic. The added deadhead time (relative to not charging) is $\tau^{1s}_{vt} + \tau^{2s}_{vt} - \tau^{3}_{vt}$ if charging takes place ($Y^s_{vt} = 1$) and 0 otherwise. Note that the objective function does not include all components of operational costs, such as driver pay and energy costs while buses are in service. These costs should be approximately the same regardless of charger location choice, so they are omitted from the objective.

Charger Management

$$Y^s_{vt} \leq X^s \qquad \forall v \in \mathcal{V},\ t \in \mathcal{T}_v,\ s \in \mathcal{S} \quad (3.12)$$

$$y^s_{vt} \leq \overline{y}^s_{vt} Y^s_{vt} \qquad \forall v \in \mathcal{V},\ t \in \mathcal{T}_v,\ s \in \mathcal{S} \quad (3.13)$$

$$N^s \leq \overline{N}^s X^s \qquad \forall s \in \mathcal{S} \quad (3.14)$$

$$\sum_{s\in\mathcal{S}} Y^s_{vt} \leq 1 \qquad \forall v \in \mathcal{V},\ t \in \mathcal{T}_v \quad (3.15)$$

BEB-OCL must include constraints that ensure its recommended charging behavior is implementable in practice. Specifically, Constraints (3.12) enforce that charging can only take place at a given site (i.e., $Y^s_{vt} = 1$) if a charger is built at that site ($X^s = 1$). Constraints (3.13) define the relationship between the charging indicator variables Y^s_{vt} and the charge time variables y^s_{vt}; charging time is upper bounded by $\overline{y}^s_{vt}$ if charging takes place and zero otherwise. Constraints (3.14) ensure that the number of chargers installed at a site does

not exceed its maximum feasible value $\overline{N^s}$, which is set by the transit agency. Its value depends on the amount of physical space available and potentially other factors such as power grid limitations. Constraints (3.14) also guarantee that $N^s \geq 1$ only if $X^s = 1$. Lastly, Constraints (3.15) limit buses to visiting a maximum of one charging station in between trips, reflecting Assumption 3 of Section 3.3.4.

Battery Charge Level

$$u_{v1} = \overline{u}_v \qquad \forall v \in \mathcal{V} \quad (3.16)$$

$$u_{v,t+1} = u_{vt} - \Delta_{vt} - \delta^3_{vt}\left(1 - \sum_{s\in\mathcal{S}} Y^s_{vt}\right) + \sum_{s\in S}\left[\rho^s y^s_{vt} - (\delta^{1s}_{vt} + \delta^{2s}_{vt})Y^s_{vt}\right] \qquad \forall v \in V,\ t = 1,\ldots,K_v - 1 \quad (3.17)$$

$$u_{vt} - \Delta_{vt} - \delta^3_{vt} - \sum_{s\in\mathcal{S}}(\delta^{1s}_{vt} - \delta^3_{vt})Y^s_{vt} \geq \underline{u}_v \qquad \forall v \in \mathcal{V},\ t \in \mathcal{T}_v \quad (3.18)$$

$$u_{vt} - \Delta_{vt} + \sum_{s\in\mathcal{S}}\left[\rho^s y^s_{vt} - \delta^{1s}_{vt} Y^s_{vt}\right] \leq \overline{u}_v \qquad \forall v \in \mathcal{V},\ t \in \mathcal{T}_v \quad (3.19)$$

To ensure BEB-OCL's output charging behavior is feasible, we track the battery level of each BEB throughout its assigned block. Constraints (3.16) encode Assumption 4 of Section 3.3.1, ensuring that buses enter service fully charged. Constraints (3.17) update buses' battery levels at the start of each successive trip: the charge $u_{v,t+1}$ at the start of trip $t+1$ is equal to the charge at the start of trip t, plus any charging performed after trip t, minus all energy consumed in completing trip t and traveling to/from a charging station if applicable. The total charge gained is the product of the charging rate ρ^s and the charging time y^s_{vt}. Battery energy consumption from completing trip t is given as Δ_{vt}. This parameter is assumed to be given and may vary by trip depending on the distance of the trip, the efficiency of the vehicle, passenger load, and physical characteristics of the trip being completed such as elevation change and stop frequency. Visiting a charging station s consumes $\delta^{1s}_{vt} + \delta^{2s}_{vt}$ units of energy in deadheading; driving directly between trips requires

deadhead energy δ^3_{vt}. Note that Constraints (3.15) ensure that each bus can charge at only one station after a trip, i.e., $Y^s_{vt} = 1$ holds for at most one station for a given bus v and trip t. Equation (3.13) also ensures that if $Y^s_{vt} = 0$, $y^s_{vt} = 0$ must hold.

Constraints (3.18) and (3.19) ensure that the charge level always remains between the lower and upper bounds $\underline{u}_v$ and $\overline{u}_v$, which are based on the battery capacity and the transit operator's policies. It is important to note that the maximum and minimum battery charges do not necessarily occur at the start of a trip, so we cannot simply place bounds on u_{vt}. For a given vehicle and trip, the minimum charge occurs upon arrival to the charger if a charger is visited, or upon arrival at the start of the next trip if no charger is visited. If the vehicle charges at site s, the battery level of the vehicle when arriving at the site is $u_{vt} - \Delta_{vt} - \delta^{1s}_{vt}$. If no charging occurs, the charge is $u_{vt} - \Delta_{vt} - \delta^3_{vt}$ when the next trip begins. Both cases are captured by Constraints (3.18). BEB-OCL also must ensure that vehicles never charge beyond their maximum capacity. This maximum is attained immediately after charging is completed. If vehicle v visits charger s after completing trip t, then its charge at this time is $u_{vt} - \Delta_{vt} - \delta^{1s}_{vt} + \rho y^s_{vt}$. This logic yields Constraints (3.19).

Queue Prevention Constraints

$$Y^s_{vt} + \sum_{(v',t') \in \mathcal{C}^s_{vt}} Y^s_{v't'} \leq N^s \qquad \forall v \in \mathcal{V},\ t \in \mathcal{T}_v,\ s \in \mathcal{S} \quad (3.20)$$

If the number of buses scheduled to simultaneously charge at a site exceeds the number of chargers located there, queues will form and buses will be delayed, which may subsequently disrupt the schedule. To avoid these delays, we propose constraints for "queue prevention" to ensure that at any time, the number of vehicles charging at a given site does not exceed the number of chargers. Because the number of buses charging only increases when a bus arrives at a charging site, it is sufficient to check this condition only when buses arrive at chargers and not at every instant of time.

With some reasonable assumptions, it is straightforward to determine the number of

buses already plugged in at a charging station when a new bus arrives. Consider some bus v that visits charging site s after completing trip t. It will arrive at the site at time $\epsilon_{vt} + \delta^{1s}_{vt}$. Then, bus v' charging at s after trip t' is already charging when v arrives if it satisfies the following conditions:

1. v' is scheduled to charge at s after some trip t'; that is, $Y^s_{v't'} = 1$
2. v' starts charging at s before v starts; that is, $\epsilon_{v't'} + \tau^{1s}_{v't'} \leq \epsilon_{vt} + \tau^{1s}_{vt}$
3. v' will not finish charging at s until after v arrives; that is, $\epsilon_{v't'} + \tau^{1s}_{v't'} + y^s_{v't'} > \epsilon_{vt} + \tau^{1s}_{vt}$

Note that Condition 2 does not depend on any decision variables and can therefore be evaluated in advance of solving the model. Condition 3 depends on the decision variable $y^s_{v't'}$. To enable *a priori* evaluation of Condition 3, we can implement these capacity constraints based on the assumption that buses always charge for the full time available to them; that is, $y^s_{vt} = \overline{y}^s_{vt} Y^s_{vt}$. Modifying this third condition still gives an upper bound on the number of vehicles charging, so it will ensure that no delays are incurred, though it is potentially over-conservative.

With this assumption in place, it is straightforward to formulate constraints that ensure charger capacity is respected and queues do not form. First, for all $v \in \mathcal{V}$, $t \in \mathcal{T}_v$, $s \in \mathcal{S}$, define the *conflict set* $\mathcal{C}^s_{vt}$ as follows:

$$\mathcal{C}^s_{vt} = \{(v', t') : \epsilon_{v't'} + \tau^{1s}_{v't'} \leq \epsilon_{vt} + \tau^{1s}_{vt} \text{ and } \epsilon_{v't'} + \tau^{1s}_{v't'} + \overline{y}^s_{v't'} > \epsilon_{vt} + \tau^{1s}_{vt}\} \tag{3.21}$$

Note that for a large problem instance, it could in theory be computationally demanding to populate $|\mathcal{T}||\mathcal{S}|$ conflict sets which each require comparison against all $|\mathcal{T}|$ trips in the instance (where $\mathcal{T}$ is the set of all unique vehicle-trip tuples). Fortunately, we can populate the sets efficiently in practice. First, we can skip setting conflict sets for all bus-trip-charger combinations v, t, s for which $\overline{y}^s_{vt} = 0$, as charging is never possible for these cases and the model already forces $y^s_{vt} = 0$ with Constraints 3.13. When layover times are relatively short

and charging sites are geographically dispersed, it is likely that $\overline{y}^s_{vt} = 0$ for many sites. Additionally, because the start and end times of charging can simply be computed *a priori*, stored in memory, and sorted, the true number of comparisons needed to fill the sets $\mathcal{C}^s_{vt}$ is much smaller than its theoretical maximum if implemented efficiently. As discussed in Section 3.4.1, in our King County case study most conflict sets are small and populating them takes about two seconds.

With the conflict sets as defined as in Equation (3.21), summing over the binary variables $Y^s_{v't'}$ for all $(v', t') \in \mathcal{C}^s_{vt}$ determines the maximum number of vehicles that could be charging when bus v arrives at s after trip t. Thus, queues will be prevented as long as $Y^s_{vt} + \sum_{(v',t')\in\mathcal{C}^s_{vt}} Y^s_{v't'}$ does not exceed N^s, giving rise to Constraints (3.20). Note that because of our earlier assumption that vehicles charge for the full feasible duration $\overline{y}^s_{vt}$ when creating the conflict sets, $Y^s_{vt} + \sum_{(v',t')\in\mathcal{C}^s_{vt}} Y^s_{v't'}$ is an upper bound on the number of vehicles charging that may not be tight if charge durations are short compared to their upper limits.

Complete Mathematical Programming Formulation

The complete BEB-OCL model is as follows:

$$\begin{aligned}
\min \quad & \text{Total cost function (3.11)} \\
\text{s.t.} \quad & \text{Charger management constraints (3.12)} - \text{(3.15)} \\
& \text{Battery level constraints (3.16)} - \text{(3.19)} \\
& \text{Queue prevention constraints (3.20)} \\
& X^s \in \{0,1\} && \forall s \in \mathcal{S} \quad (3.22) \\
& N^s \geq 0, \ \text{integer} && \forall s \in \mathcal{S} \quad (3.23) \\
& Y^s_{vt} \in \{0,1\} && \forall v \in \mathcal{V}, \ t \in \mathcal{T}_v \quad (3.24) \\
& y^s_{vt} \geq 0 && \forall v \in \mathcal{V}, \ t \in \mathcal{T}_v, \ s \in \mathcal{S} \quad (3.25) \\
& u_{vt} \geq 0 && \forall v \in \mathcal{V}, \ t \in \mathcal{T}_v, \ s \in \mathcal{S} \quad (3.26)
\end{aligned}$$

BEB-OCL is a mixed-integer linear programming problem that consists of $|\mathcal{T}|(|\mathcal{S}|+1)$ continuous variables, $|\mathcal{S}|(|\mathcal{T}|+1)$ binary variables, $|\mathcal{S}|$ integer variables, and $|\mathcal{T}|(3|\mathcal{S}|+4)+|\mathcal{S}|$ constraints (excluding the variable bounds defined in (3.22)-(3.26)). For reference, the complete Metro service area included 1549 total blocks serving 9298 trips, as calculated based on the GTFS data used in the case study of Section 3.4. With reasonable battery sizes, many BEBs should be able to complete their blocks each day without layover charging, so $|\mathcal{T}|$ (which consists only of trips on layover blocks) will be much smaller than the full number of trips.

Solution Method

Our solution approach relies primarily on the commercial solver Gurobi, but we also apply two techniques to improve computational time. For large problem instances, BEB-OCL may include a large number of both variables and constraints. Our first technique is to reduce the number of variables by removing those that are redundant. For example, it is often infeasible for a bus v to visit charger s after trip t because too much deadheading is required, so there

is no way for Y_{vt}^s to equal 1. In a pre-processing stage, we remove all such decision variables. Let $\Psi = \{(v,t,s) : v \in \mathcal{V},\ t \in \mathcal{T}_v,\ s \in \mathcal{S},\ \overline{y}_{vt}^s > 0\}$. Then any variables and constraints indexed by v, t, and s in BEB-OCL are defined only over $(v,t,s) \in \Psi$. $\overline{y}_{vt}^s$ is defined in Equation (3.10).

Second, when solving BEB-OCL, we handle the potentially large number of queue prevention constraints (3.20) using a cutting plane approach as originally proposed in Dantzig et al. [20] for the Traveling Salesman Problem. We start by solving a relaxed version of the model with all queue prevention constraints omitted. We then check the solution to see whether the number of buses charging at a given site at a given time ever exceeds the number of chargers located there. If so, we add queue prevention constraints only for the indices (v,t,s) that were violated. We repeat this procedure until a solution is obtained in which no queue prevention constraints are violated. Typically, only a small fraction (less than 10% in our experiments) of all possible queue prevention constraints are included in the final model. The approach is detailed explicitly in Algorithm 3.

Algorithm 3: Solution Method for BEB-OCL

```
feas ← FALSE
Ψ' ← ∅
while feas == FALSE do
    Let P = min{(3.11) subject to (3.12)-(3.19), (3.22)-(3.26),
      queue prevention constraints (3.20) for only Ψ' ∈ Ψ}
    Find an optimal solution to P. Let Ŷ_vt^s and N̂^s be the optimal values of Y_vt^s and
      N^s, respectively, in this solution.
    feas ← TRUE
    for (v,t,s) ∈ Ψ do
        if Ŷ_vt^s + Σ_{(v',t')∈C_vt^s} Ŷ_v't'^s > N̂^s then
            feas ← FALSE
            Ψ' ← Ψ' + {(v,t,s)}
        end
    end
end
```

The first two lines of Algorithm 3 initialize the variable *feas*, which indicates whether the most recent solution obtained in the procedure is feasible, and the set $\Psi' \subseteq \Psi$, which is the set of all indices of conflict prevention constraints included in the BEB-OCL instance. Then the following procedure repeats until a feasible optimal solution is found. Line 4 defines a new BEB-OCL instance that includes queue prevention constraints only for the indices $(v, t, s) \in \Psi'$. In Line 5, we use an optimization solver to solve this instance P. In our experiments, BEB-OCL is implemented in Python using the Pyomo optimization modeling library [13, 40] and the Gurobi solver. Gurobi's *Method* parameter is set to 1 (meaning dual simplex is used) and the *MIPFocus* parameter is set to 3, which prioritizes improving objective bounds in the solver's branch-and-bound procedure. After an initial solution is obtained with no queue prevention constraints, each subsequent solve is warm-started to accelerate the solution process. Lines 6-12 check for any violated queue prevention constraints. If a constraint is violated (as checked in Line 8), the current model is marked as infeasible (Line 9), and the index of the violated constraint is added to the set Ψ' (Line 10), at which point the procedure will be repeated.

3.4 Case Study: King County, WA

Part of the transit network of King County, WA was selected as a testbed for the models presented in this work. The case study includes all blocks that serve trips on the 15 highest-ridership routes operated by Metro, as documented in the 2020 System Evaluation report [62] which reports ridership from 2019. We excluded all of the routes that are served by electric trolleybuses from this analysis, as they will not be replaced with BEBs. Accordingly, the case study consisted of blocks that serve routes 5, 8, 40, 41, 45, 62, 65, 67, 106, 120, and 372, as well as the RapidRide A Line, C Line, D Line, and E Line. These 15 routes had an average weekday ridership of 135,300, representing about 35% of all Metro ridership in 2019 [62]. In total, 318 blocks consisting of 2458 trips travel along these routes on a typical weekday, including 132 layover blocks that serve 1980 trips.

Ten candidate charging stations were identified based on Metro's list of Park & Rides

and Transit Centers; we considered all transit centers and large Park & Ride locations, in addition to the overnight depot at the South Base. We removed from consideration any such sites that were not located close to any trip terminals on the case study routes, as they would clearly require too much deadheading to be optimal. The list of all charging sites ultimately considered is documented in Appendix 3.A and also shown on the map in Figure 3.5.

Our case study of high-capacity routes examines a case in which all vehicles are 60 ft. (18.3 m) articulated buses with 466 kWh batteries and an average energy consumption rate of 3 kWh/mile (1.86 kWh/km). Trip distances were calculated based on GTFS data, while deadhead driving distances and times were obtained using the Openrouteservice API [89]. We assumed chargers had a maximum power output of 450 kW each and that all sites could accomodate a maximum of 4 such chargers. Complete details of the data collection process for BEB-BRP and BEB-OCL as well as all parameters are provided in Appendix 3.A.

To summarize the results of the data collection process, Figure 3.5 provides some visualizations of the trips in the case study over a map of the region. Because the case study routes primarily run in a north-south direction, Figure 3.5 (a) shows the northern portion of the case study and Figure 3.5 (b) shows the southern portion. The ten candidate charging sites are labeled as well as the overnight charging location at the South Base (which is itself also a candidate site). In Figure 3.5, the red lines trace the trips completed by all blocks in the case study based on the GTFS data. The opacity of each line corresponds to the number of trips traversing that path. The majority of trips under study travel between Downtown Seattle and more suburban parts of Seattle and King County. Figure 3.5 also plots the terminals of each trip, marked with blue circles. These locations are more significant to the model than the full trip shapes because buses only charge in between trips, so the terminals indicate where buses are located when they are available to visit a charger. The size of the marker for each terminal indicates how many individual trips end at that location.

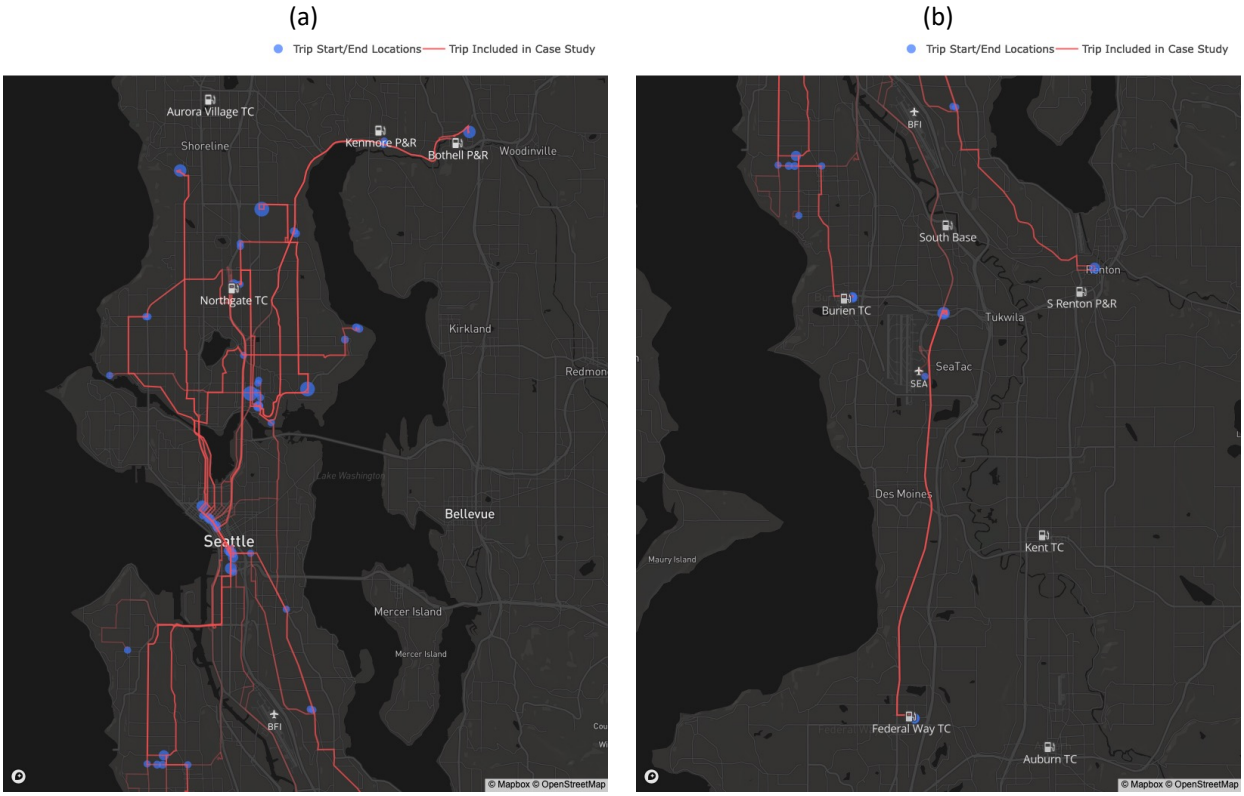


Figure 3.5: Summary of trips included in case study and candidate charging sites.

3.4.1 Results

Filtering down the GTFS data identified a total of 318 blocks serving the case study routes on a typical weekday. Of these 318 blocks, 132 are layover blocks that together serve 1980 unique trips. BEB-FC identifies that among these, 30 blocks are infeasible and subjected to BEB-BRP. These 30 infeasible blocks serve a total of 446 trips. The remaining 102 blocks (which include 1534 trips) are feasible and used as inputs to BEB-OCL. Hence, for the BEB-OCL case study instance, $|\mathcal{V}| = 102$, $|\mathcal{T}| = 1534$, and $|\mathcal{S}| = 10$. Filtering out infeasible charging locations, the set Ψ of feasible (v, t, s) tuples has size $|\Psi| = 4,820$. The complete BEB-OCL case study instance consists of $|\mathcal{T}| + |\Psi| = 6,354$ continuous variables, $|\mathcal{S}| + |\Psi| = 4,830$ binary variables, $|\mathcal{S}| = 10$ integer variables, and $3|\Psi| + 4|\mathcal{S}| + |\mathcal{S}| = 20,606$

constraints. Populating all of the conflict sets $\mathcal{C}_{vt}^{s}$ took 2.05 seconds and the average conflict set size is 9.37.

Key results for the King County case study are summarized in Tables 3.3 and 3.4. The BEB-BRP heuristic obtains a solution in just over 2 seconds, finding that an additional 22 backup BEBs are needed to ensure all 30 infeasible blocks can be completed. At a purchase price of $887,308 [51], this corresponds to $19,520,776 of additional capital costs. For BEB-OCL, Algorithm 3 obtains an optimal solution in 371 seconds. The procedure terminated after 9 iterations and the final model included 156 queue prevention constraints, or only 3.2% of all the possible such constraints (recall that $|\Psi| = 4,820$), demonstrating the effectiveness of this solution approach. BEBs serving the 102 layover blocks charge a total of 161 times; most of these buses charge once or twice during the day to maintain sufficient battery levels. The typical duration per charge is just over 15 minutes. The total cost of chargers and operations as reflected in the BEB-OCL objective is $23.4 million; $6.6 million of this cost comes from charger purchases, while the remaining $16.8 million is due to deadheading. For context, Metro's total capital expenditures in 2019 were $231 million; operating costs totaled $916 million [28].

The layover charging sites selected by BEB-OCL and documented in Table 3.4 are consistent with reasonable expectations based on the terminal locations as shown in Figure 3.5. The five sites identified as optimal are generally located close to a large number of stops where trips begin and end, meaning that vehicles do not have to make large detours to charge. The Northgate and Burien Transit Centers have fewer trip terminals close by, but are used to serve buses in the central Seattle area where no candidate charging sites are located in our case study. Clearly, deadhead time could be reduced and operating expenses would be lower if a charging site closer to Downtown Seattle were considered.

Figure 3.6 displays the allocation of vehicles to chargers as indicated by the optimal values of Y_{vt}^{s}. In this figure, the circular markers indicate the terminals of trips after which charging is performed. The size of each marker corresponds to the number of trips that end at that terminal. For each of these locations, a line also maps the deadhead route that vehicles drive

to the charger, as found with the Google Maps Directions API [38]. Figure 3.6 shows that vehicles generally charge at the station that is geographically closest, and most charger visits happen when buses are in the immediate vicinity of the chosen sites. In a limited number of cases, buses make a long deadhead trip to visit a charger. For example, in Figure 3.6 some buses complete trips in the Downtown Seattle area and then drive to the Burien Transit Center or Northgate Transit Center to charge. This outcome might appear surprising, but in most of these cases there is already a long deadhead trip scheduled (corresponding to a large value of δ^3_{vt}) and the model takes advantage of this to visit a charger that requires minimal deviation from the deadhead trip. Hence, the objective term $\delta^{1s}_{vt} + \delta^{2s}_{vt} - \delta^3_{vt}$ is still small.

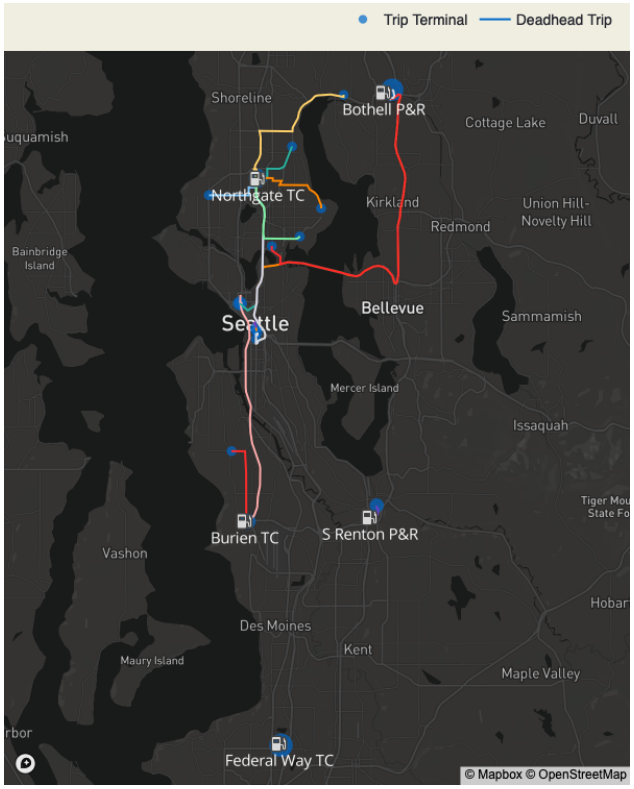


Figure 3.6: Optimal charging deadhead patterns in Metro case study.

To show the charging behavior in greater detail, Figure 3.7 plots the scheduled charger

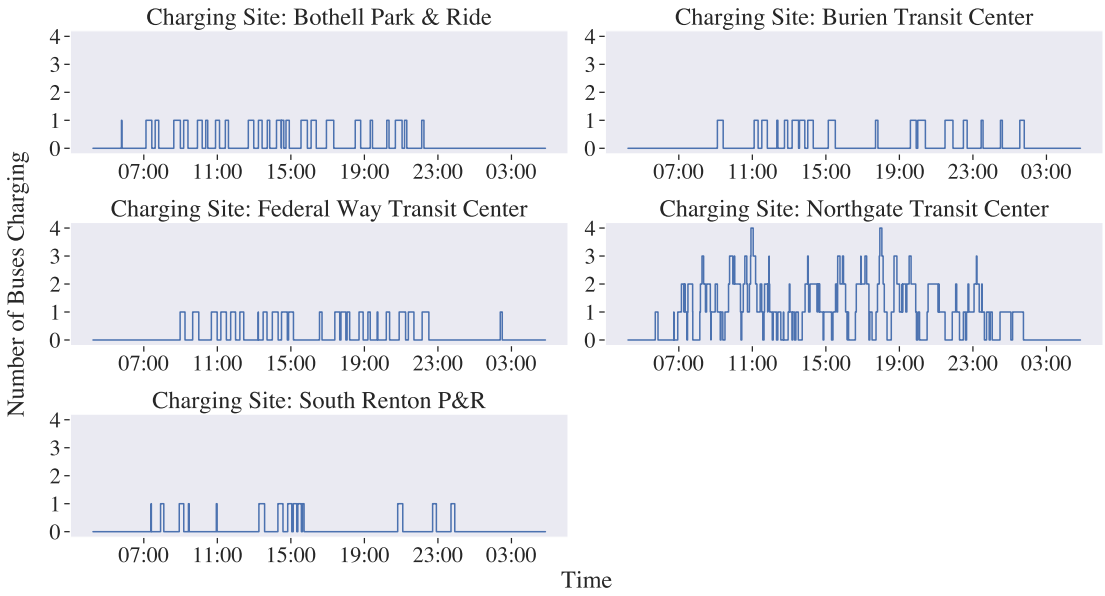


Figure 3.7: Charging station utilization over time in Metro case study.

utilization at each of the five optimal sites over the duration of the service day. The number of vehicles charging at any of the sites at a particular time never exceeds the number of chargers built there, so there will be no queue delays (i.e., queues are prevented). In fact, at all five of the chosen sites, the maximum number of vehicles simultaneously charging is exactly equal to the number of chargers installed, despite the conservative approach used to populate the conflict sets $\mathcal{C}_{vt}^{s}$.

3.5 Sensitivity Analysis

To explore how the outputs of BEB-OCL and BEB-BRP depend on key input parameters, we performed some sensitivity analysis on a simple test network. This notional transit network consists of 16 blocks on 3 routes with different layover times and 3 candidate charging sites. Sites S and W are located at trip terminals and do not require deadheading, whereas site NW requires a short deadhead trip and is less expensive to construct. For a full description of the test network used for sensitivity, see Appendix 3.B. For each parameter tested, all

other parameters retained the baseline values reported in Tables 3.7-3.9. Additionally, in calculating the total system costs plotted in Figures 3.9-3.11, each backup bus was assessed a cost of $887,308, corresponding to the average BEB purchase price reported in Johnson et al. [51]. We assumed these additional buses had a usable capacity of 400 kWh, the same as the original buses as reported in Table 3.9.

The sensitivity analysis examines three key outcomes of BEB-BRP and BEB-OCL: the number of backup buses calculated with Algorithm 2, the total system cost (dependent on both the number of backups and the results of BEB-OCL), and the number of chargers located at each site. In Figures 3.8-3.10, the total cost is broken down into three components: the capital cost $\sum_{s\in\mathcal{S}}\left(f^s X^s + g^s N^s\right)$, the deadhead cost $\alpha \sum_{v\in\mathcal{V}} \sum_{t\in\mathcal{T}_v} \sum_{s\in\mathcal{S}} \left(\tau_{vt}^{1s} + \tau_{vt}^{2s} - \tau_{vt}^{3}\right) Y_{vt}^s$, and the backup bus cost (calculated as $887,308 times the number of backups). Sections 3.5.1-3.5.4 present the results of testing various values of the deadhead cost α, charging power ρ^s, and battery capacity $\overline{u_v} - \underline{u_v}$.

3.5.1 Deadhead Time Cost

Recall that the objective function parameter α, the cost per unit time of deadhead driving, controls the tradeoff between capital costs and deadhead costs of BEB-OCL. Figure 3.8 plots its impact on both capital and deadhead costs as well as the optimal charger configuration. The results of testing 50 values equally spaced between $100 and $8000 are plotted in Figure 3.8; increasing the value of α further did not impact the optimal solution because deadhead was eliminated entirely. Note that backup buses are not necessary for these instances regardless of the value of α.

Figure 3.8 shows that, as expected, the total cost increases linearly with α up to a critical point at which it becomes optimal to build more expensive charging infrastructure (in this case, at $\alpha \approx$ $4400/hr). At this point, the total cost curve flattens out as deadhead to chargers is completely eliminated. This outcome is aligned with Figure 3.8 (b), which shows that this change corresponds to a switch from locating a charger at NW to W, so that vehicles always have convenient charging with no deadhead required. Two chargers are located at S

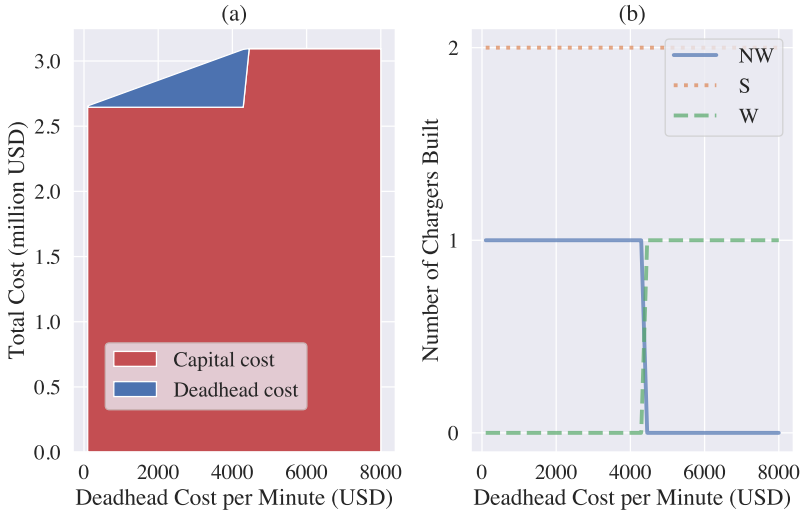


Figure 3.8: Impact of deadhead time cost α on: (a) components of cost and (b) number of chargers built.

regardless of the value of α.

3.5.2 Charger Power Output

A variety of BEB charger types with different power outputs are available commercially. For example, high-power overhead charging options offered by New Flyer, the manufacturer of Metro's BEBs, range from 150 kW to 600 kW [87]. To examine the impact that charger power has on backup bus requirements, charger location decisions, and total costs, we tested the models with charging power ρ^s varied between 50 kW and 600 kW. We tested 50 equally spaced values of ρ^s within this range, and in each instance all sites s were assigned the same power level. The results are summarized in Figure 3.9.

Figure 3.9 (a) shows the number of backup buses dispatched by BEB-BRP as a function of charger power output. The results show that backup buses are only needed when the power output is quite low; no backups are required when the power output is 120 kW or greater. Below 60 kW (a power level generally reserved for slow overnight depot charging), 6

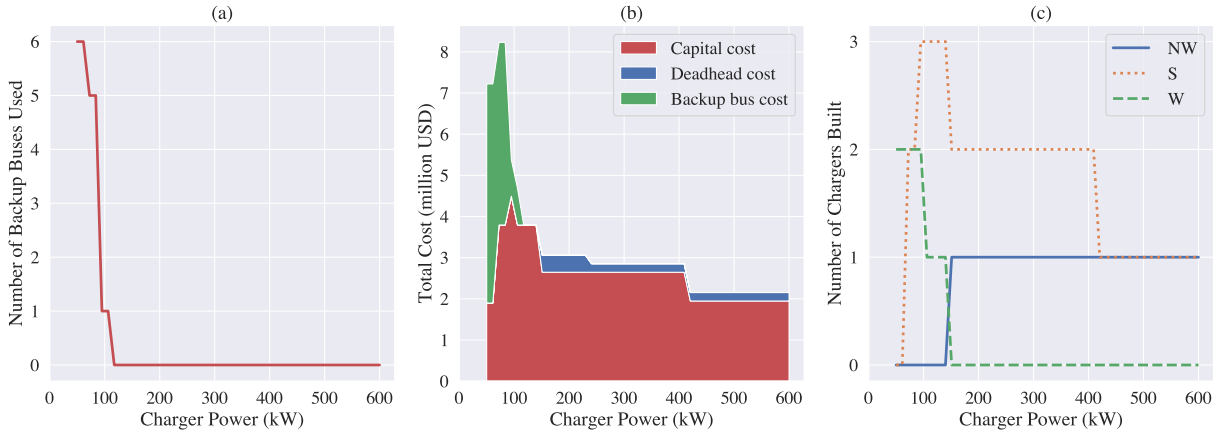


Figure 3.9: Impact of charger power ρ^s on (a) number of backup buses used, (b) components of cost, and (c) number of chargers built.

backup buses are needed for this simple case. Figure 3.9 (b) plots the impact of varying ρ^s for all chargers on costs. Note that this analysis does not attempt to model the impact of power output on charger cost, but the impact of power level on all other costs. Costs generally decrease as power increases, with a notable exception at the lowest power levels, suggesting that without high-power chargers, it is most economical to modify bus blocks using BEB-BRP or a similar approach rather than rely on layover charging. Once ρ^s is 70 kW or above, the total cost is nonincreasing. Figure 3.9(b) shows that increasing charger power output produces the greatest cost savings at relatively low power outputs. At low power levels, even a small increase in power output can eliminate backup buses and substantially reduce the number of chargers needed. At higher power levels, cost reductions occur because the same amount of charge can be gained by a given bus in less time. This may yield capital savings because fewer chargers are needed (for example, at 420 kW or greater, only one rather than 2 chargers are needed at S) or operational savings because buses do not need to take as many deadhead trips to chargers (as is seen in the deadhead cost reduction between 230 and 240 kW in Figure 3.9(b)).

Figure 3.9 (c) illustrates the evolution of charging site selection as ρ^s increases. Several

different charge site combinations are found to be optimal depending on the power level; in fact, there is no site at which it is optimal to build chargers across the whole range of values tested. At the lowest power levels (60 kW and below), it is sufficient to build chargers only at W because of the limited number of feasible blocks and high usage of backup buses. From about 70 to 105 kW, more blocks are feasible, and chargers at both W and S are needed. At higher power levels, all blocks are feasible and backup buses are not used, so the trend is more consistent. The number of chargers needed at S decreases from 3 to 1 as power increases, and at 150 kW and above it becomes preferable to use site NW rather than site W in order to reduce capital costs.

3.5.3 Battery Capacity

As with charger power levels, BEB manufacturers offer a variety of battery sizes for their vehicles. Figure 3.10 shows the sensitivity results for 50 equally spaced values of battery capacity $\overline{u}_v$ from 100 kWh to 500 kWh. Note that for this notional case, the battery capacity $\overline{u}_v$ is equal to the full usable capacity $\overline{u}_v - \underline{u}_v$ since $\underline{u}_v = 0$.

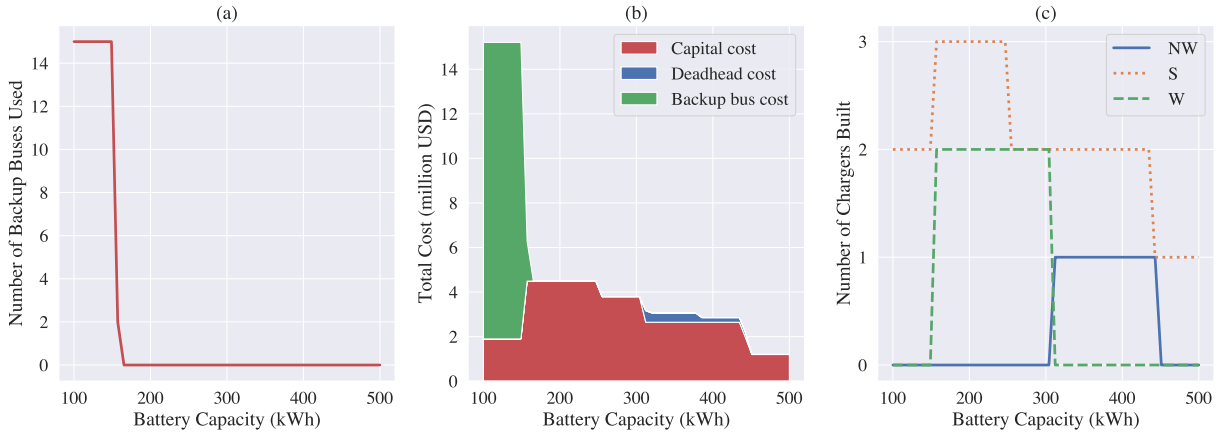


Figure 3.10: Impact of battery capacity $\overline{u}_v$ on number of backup buses used (a), components of cost (b), and number of chargers built (c).

Figure 3.10 (a) shows that backup buses are frequently needed when battery capacity

is low. Some backups are required to make operations feasible when capacity is below 160 kWh; 15 additional BEBs are needed when capacity is less than 150 kWh. Figure 3.10 (b) shows that total cost is nonincreasing as battery size increases, with sharp cost reductions occurring whenever backup buses are eliminated or less expensive charging infrastructure becomes feasible. Figure 3.10 (c) shows that battery size significantly impacts the number of chargers installed and their power output. At low capacities, as many as 5 total chargers are installed, but the number of chargers decreases with the battery size to the point that only a single charger at S is needed when battery capacity is 450 kWh or greater. From 160 kWh to 305 kWh, both sites S and W are used for charging; from 300 kWh to 450 kW, sites S and NW are used.

3.5.4 Combined Charger Power and Battery Size

Figures 3.9 and 3.10 demonstrate that charger power output and battery capacity both have major impacts on system costs; higher-power chargers and larger batteries both generally reduce capital, deadhead, and backup bus costs. These two parameters are also closely related; together, they determine how much time buses must spend charging, since a bus with a smaller battery will need to gain more charge throughout the day and using a lower-power charger will increase the amount of time required to gain a given amount of charge. To demonstrate how these two parameters interact, Figure 3.11 plots the total cost (the sum of backup bus, capital, and deadhead costs) versus the values of battery capacity and charger power. Ten battery sizes from 100 to 500 kWh were considered as well as ten power output levels ranging from 50 kW to 600 kW.

Figure 3.11 shows that, as expected, costs tend to decrease as both charging power and battery size increase. There are sometimes exceptions to this trend for relatively small battery sizes. Similar to the trend shown in Figure 3.9(b), this pattern reflects that in certain cases it is more economical to increase the bus fleet size and modify blocks rather than to use layover charging whenever possible. Once battery capacity is above 250 kWh and charger power is above 250 kW, we see more consistent reductions in cost.

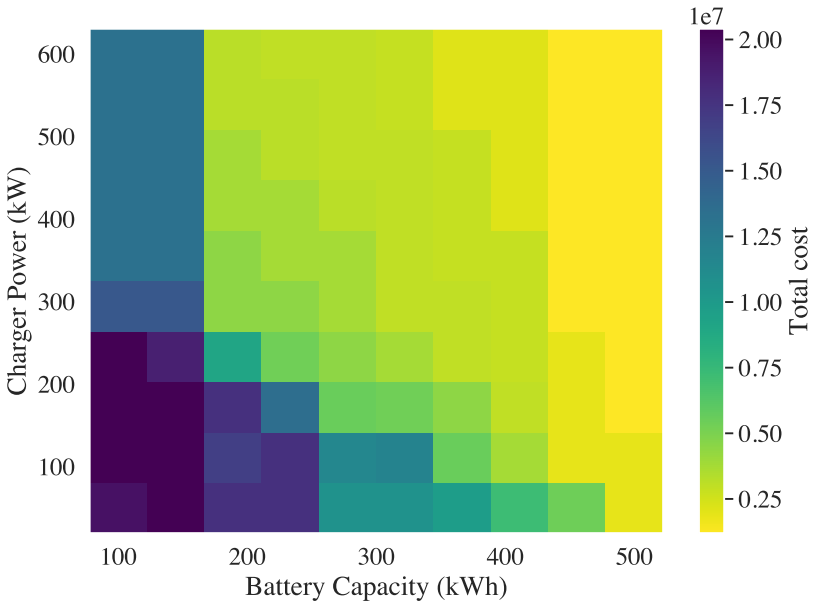


Figure 3.11: Combined impact of varying charger power and battery capacity on costs.

Figure 3.11 also shows that relatively high-capacity batteries are important for achieving the lowest possible costs. For example, the total cost can be low even with comparatively low-power chargers of around 200 kW if 500 kWh batteries are used. On the other hand, total cost is high even with the fastest chargers when battery capacity is small (i.e., below 300 kWh, and especially below 200 kWh). At least in the example transit network used for our sensitivity analysis, larger batteries are a necessary prerequisite to achieve low total costs, whereas high-power chargers provide the greatest benefits when battery capacity is in the middle of the range of values considered. The most cost-effective choice of battery and charger technology will depend on their specific BEB application as well as the relative costs for increasing charger power and battery capacity. If needed, repeatedly running BEB-BRP and BEB-OCL with different values of the power and capacity parameters can help agencies identify the most appropriate solution for their unique transit systems.

3.6 Discussion and Conclusion

This work presented a new approach for locating charging infrastructure for battery-electric buses (BEBs). The approach aims to limit the added complexity of electrifying bus systems by fitting charging within existing bus blocks wherever possible and visiting chargers between trips in a strategy called layover charging. We use layover charging to support bus operations whenever is feasible, as this has minimal impact on existing bus operations planning and does not require increasing the bus fleet size, which incurs more capital costs on buses and labor costs for drivers. We developed two related optimization models, BEB-BRP and BEB-OCL, designed to assist agencies in selecting ideal locations for layover charging infrastructure. After using the BEB-FC procedure to check which bus blocks can feasibly be served by BEBs, BEB-BRP revises any infeasible blocks by serving some trips with backup buses (thereby increasing the fleet size), while BEB-OCL selects the sites for layover charging, the number of chargers located at each location, and the charging schedule for each BEB. Both models are based on data that can be readily obtained from agency GTFS feeds, making them easily transferable to contexts beyond the presented case study. We presented a heuristic algorithm for BEB-BRP and a cutting plane approach to solve BEB-OCL exactly, which allowed both to be solved within a few minutes for our case study of the highest-ridership routes in King County, WA that serve a total of nearly 2000 daily trips.

Our modeling approach, particularly the strategy for locating and sizing charging stations in BEB-OCL, makes some notable improvements over the existing literature reviewed in Section 3.2. Most significantly, BEB-OCL accounts for the limited capacity of chargers and identifies the number of chargers to be built at each site to ensure that buses do not wait in queues to charge. The key to the queue prevention approach is the insight that any time a vehicle visits a charger, there is usually only a small number of vehicles that could potentially be charging at the same location at the same time. As a result, the number of vehicles charging whenever a vehicle arrives at a charger can be calculated quite simply with Equation (3.20). The conflict sets $\mathcal{C}^s_{vt}$ can also be efficiently populated prior to

running the optimization. While the queue prevention approach may be overly conservative if vehicles charge for less than the full layover time that is available to them, our experiments still show good performance. In the King County case study, comparing the number of chargers recommended in Table 3.4 to the charger utilization plots of Figure 3.7 shows that at all sites chosen, the number of chargers built is equal to the maximum number of vehicles that simultaneously charge, as intended. Our flexible model of layover charging that allows deadheading to chargers and partial recharging enables buses to share infrastructure efficiently; the results show that the 102 BEBs that serve feasible blocks in the case study need a total of only 8 chargers to serve them in daily operations.

In addition to validating the queue prevention approach, the King County case study of Section 3.4 shows the insights that BEB-BRP and BEB-OCL provide on a transit system scale. For this selection of 15 major Seattle-area bus routes and 10 candidate charging locations, BEB-BRP finds that 30 of 132 layover blocks are infeasible and 22 additional BEBs are needed to serve them. BEB-OCL finds that the remaining 102 feasible blocks are best served by 5 charging stations with 1 to 4 chargers each, totaling 8 charger ports. Deadhead trips are generally short whenever possible, though some longer deadhead distances are necessary for buses that complete trips in downtown Seattle and do not have chargers available nearby. On a personal computer, the BEB-BRP heuristic runs in just over 2 seconds and BEB-OCL takes just over 6 minutes to solve to optimality. This performance is favorable when compared to some more complex models for BEB operations; for example, as discussed in Section 3.2.2, Vehicle Scheduling Problem variants for BEBs do not appear to handle instances with more than 300 bus trips except with heuristics that lack any performance guarantees.

The sensitivity analysis in Section 3.5 shows that BEB-BRP and BEB-OCL are responsive to changes in key parameters. For instance, the results show that as the charging time cost α increases, BEB-OCL responds by shifting charger locations from sites that are less expensive but less convenient to higher-cost sites that require less deadheading. Likewise, as charger power and battery capacity (both of which impact the amount of time required for charging)

increase, BEB-BRP reduces the number of backup buses dispatched and BEB-OCL reduces capital expenses for chargers while still ensuring buses stay on schedule. These analyses also demonstrate how agencies may use our proposed approach to assess the value provided by improved technology (e.g. high-power chargers), giving a quantitative estimate of the cost reductions they enable.

This work leaves multiple interesting directions for future research. One improvement would be to incorporate uncertainty in some of the input parameters into the optimization models, as other authors have done for related problems [4, 99]. BEB energy consumption can vary widely based factors such as passenger load and weather conditions, and the time available for charging $\overline{y}^s_{vt}$ may be impacted by traffic conditions, passenger load, mechanical issues, and so on. An improved model would therefore recognize that many of the parameters utilized in this work are not known exactly, and seek to provide good system performance over a wide range of real-world scenarios. There are a number of potential ways to extend the models to incorporate uncertainty, but two specific possibilities include simulation-based optimization (similar to Jung et al. [52]) and sample average approximation [4, 64]. Such analysis would also prove more valuable if based on real-world data on energy consumption, schedule adherence, and battery performance. Extending our models to incorporate uncertainty or analyze substantially larger transit networks may require more advanced solution methods, especially for the BEB-OCL mixed-integer program. Possible approaches we expect might improve solution time include using a tailored branch-and-cut algorithm with similar methods to Algorithm 3 or a column generation method.

Future work could also analyze the cost impacts of different BEB recharging approaches in greater detail. For example, more detailed modeling of how charger cost varies with power level would help agencies select the infrastructure that minimizes their cost by purchasing chargers fast enough to reduce some costs as seen in Figure 3.9(b), but without spending more than is necessary. Further generalizing BEB-OCL or a similar model to set the charger power output as a decision variable would also help transit operators make this decision. Additionally, there is a need to compare the ultimate costs of different charging strategies

to help guide such high-level decisions. For example, future work may compare the total costs of a system dependent fully on layover charging as recommended by BEB-BRP and BEB-OCL versus the costs of a system that relies only on overnight depot charging and uses the EVSP to generate new bus blocks from scratch, without layover charging. As transit electrification continues to take hold, such work is increasingly important and can help accelerate a worldwide transition to more environmentally friendly transit vehicles.

Appendix 3.A King County Case Study Data

Several data sources were utilized in developing the case study of future BEB routes in King County. The majority of the data needed to test the model—including bus blocks, trip schedules and distances, and stop locations—is publicly available through the Metro website via the General Transit Feed Specification (GTFS) [58]. Deadhead driving distances and times were calculated based on free-flow conditions using the Openrouteservice distance matrix API. The remaining parameter values needed to run the model were either provided by Metro or assigned reasonable values based on published literature. The remainder of this section provides more details on the collection and processing of this data.

3.A.1 Candidate Charging Sites

Table 3.5 lists the names and coordinates of the 10 candidate charging sites used in the King County case study.

3.A.2 GTFS Data

All data related to bus operations, including blocks, trip start and end times, and the latitude/longitude coordinates of the start and end points of each trip, were obtained from Metro's GTFS feed [58]. The GTFS files used for this project were originally downloaded in February 2020 and contain operating schedules for the period of February 10, 2020 to June 12, 2020. Although more up-to-date information is available, the early 2020 schedules were

chosen for the case study to avoid making charging infrastructure recommendations for the future based on recent timetables that have been revised due to impacts of the COVID-19 pandemic.

The GTFS format separates transit data into several files, each with a table structure [39]. To extract all relevant case study parameters, the GTFS data were filtered down to retain only what was relevant for the case study. First, the dataset was restricted to a typical weekday of service by filtering based on the GTFS `service_id` variable. The `trips.txt` file was used to identify all remaining trips that served the 15 routes within the scope of the case study. All unique `block_id` values across these trips were considered to be within the scope of the study. That is, all trips from any block that included at least one trip on one of the 15 study routes on a typical weekday was included. For each of these blocks v, the corresponding set of trips $\mathcal{T}_v$ was then identified in `trips.txt` based on the `block_id` and sequenced in the order $1, \ldots, K_v$ based on the `trip_sequence` variable in `block_trip.txt`. Trip start and end times were identified based on the `stop_times.txt` file, while the total distance of each trip and start/end coordinates came from `shapes.txt`.

3.A.3 Deadhead Distances and Times

Driving distances and times ($\delta_{vt}^{1s}, \delta_{vt}^{2s}, \delta_{vt}^{3}, \tau_{vt}^{1s}, \tau_{vt}^{2s}, \tau_{vt}^{3}$) were calculated using the Openrouteservice API [89] based on the terminal stop coordinates documented in GTFS.

3.A.4 Trips to and from Bus Depot

Metro's initial BEB deployment will be centered around the existing South Base and adjacent Interim Base that is planned for construction in 2025 [63]. Accordingly, buses entering service will need to drive from the South Campus to the start of their first trip, and buses leaving service will need to return there at the end of the day. To account for the energy needs of these two trips, we added two trips to each bus's block at the start and end of the day. The energy demand of these deadhead trips was estimated by querying driving distance in the Openrouteservice API [89] and using the same energy consumption rate as for all other

driving. We assume that buses never charge between the depot and the start of the first actual trip; that is, $\overline{y}^s_{v1} = 0$ for all sites s and buses v. On the other hand, we assume that buses may charge up to the maximum allowable duration after completing their final trip and prior to returning to the depot; that is, $\overline{y}^s_{v,K_v-1} = y_{\max}$.

3.A.5 Additional Parameters

Table 3.6 documents the values of all model parameters and their sources, where applicable. Based on the BEB models Metro has purchased, the case study assumes that all vehicles are 60 ft. (18.3 m) articulated buses with 466 kWh batteries that consume energy at a rate of 3 kWh/mile (1.86 kWh/km) for all trips served. This energy usage rate was multiplied by the trip distances from GTFS and deadhead distances from Openrouteservice to obtain the trip energy requirements Δ_{vt} and deadhead energy requirements $\delta^{1s}_{vt}, \delta^{2s}_{vt}$, and δ^{3}_{vt}. Furthermore, we assume that operations should keep the battery level between 10% and 85% state of charge at all times to promote battery longevity and provide a margin of error in case energy usage is higher than expected.

All candidate sites were assigned the same capital costs f^s and g^s, charger power ρ^s, and charger limit $\overline{N}^s$. The per-charger cost g^s was selected based on the average value reported in TCRP Synthesis 130 [111]. We used an assumed fixed cost of f^s=\$200,000 for all sites, since this figure is likely to be highly agency- and site-specific. This cost is smaller compared to the charger cost g^s because all sites are already actively used by Metro, so land purchases or extensive site changes are unlikely to be needed. A baseline charging station power of $\rho^s = 450$ kW was chosen as it is in the middle of the range of high-power chargers advertised by New Flyer [87]. The charging cost α was calculated based on Metro's average hourly operations cost (\$190) as reported in the National Transit Database [28]. This figure was scaled by the expected 12-year lifetime of the infrastructure (following Johnson et al. [51]) and assumed daily operations (i.e., 365 days per year), giving the final cost of \$13,870 per minute of deadheading (per day).

Appendix 3.B Simple Network for Sensitivity Analysis

We developed a simple notional transit network for sensitivity testing of BEB-BRP and BEB-OCL in order to ensure that results were easily interpretable. Figure 3.12 and Table 3.7 provide an overview of this simple transit system. There are three routes (A, B, and C) that connect four terminals. Three potential charging sites, named based on their locations (W, NW, and S) are under consideration. The sites W and S are located at route terminals. Because these are actively used bus stops with limited space available, these candidate sites incur a greater setup cost than site NW, which is less expensive but requires some deadhead travel. The full details of the candidate sites, including x- and y-coordinates corresponding to their positions in Figure 3.12 as well as site-specific parameters, are given in Table 3.8.

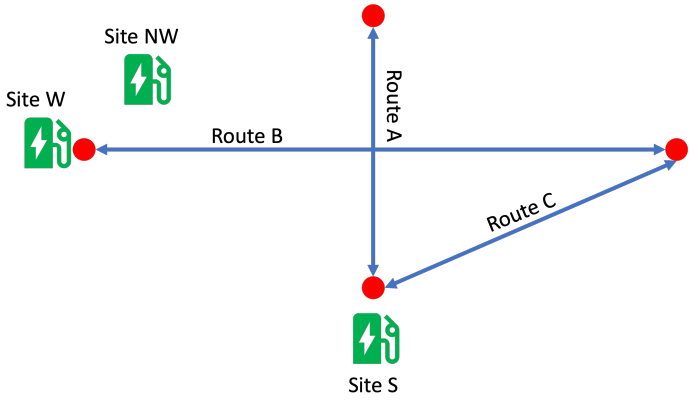


Figure 3.12: Illustration of routes, terminals, and candidate charging sites in simple case study.

To develop a straightforward example, we generated similar blocks for all vehicles. Each bus in the system repeats the same route in alternating directions for the entire service day, with the earliest trips leaving at 7:00 a.m. and the last trips departing at 7:00 p.m. A new block is generated for a set time interval based on the headway reported in Table 3.7. For example, for Route B with 30-minute headways, there are eight blocks generated. Two buses enter service (one at each of the terminals W and E) simultaneously at 7:00, 7:30, 8:00, and

8:30, so that one bus traveling each direction leaves every 30 minutes (note that the bus that originally left W at 7:00 will leave E at 9:00 for its second trip, leave W at 11:00 for its third trip, etc.) These eight buses then serve essentially identical blocks, alternating directions on Route B until 7:00 p.m. Similarly, six buses serve Route A and two serve Route C, giving 16 total blocks.

To run BEB-BRP and BEB-OCL for the simple case study, we also need to identify the deadhead parameters. To do so, we first calculate the distances between each of the four terminals and all the candidate charging sites as the Euclidean distance (in miles) between their coordinates as defined in Table 3.8. We then calculate deadhead energy $\delta_{vt}^{1s}, \delta_{vt}^{2s}, \delta_{vt}^{3}$ and deadhead times $\tau_{vt}^{1s}, \tau_{vt}^{2s}, \tau_{vt}^{3}$ based on an assumed energy consumption rate and average driving speed of 3 kWh/mi (1.86 kWh/km) and 25 mph (40.2 km/hr), respectively. The remaining parameters used to evaluate the simple case study are documented in Table 3.9.

Table 3.2: Notation definitions.

Sets	
$\mathcal{V}$	Buses
$\mathcal{V}^F$	Buses on feasible blocks
$\mathcal{V}^{IF}$	Buses on infeasible blocks
$\mathcal{S}$	Candidate charging sites
$\mathcal{T}_v = \{1, \ldots, K_v\}$	Trips to be completed by each bus $v \in V$
$\mathcal{T}$	All trips completed by all buses
$\mathcal{C}^s_{vt}$	Conflict set for bus v arriving at s after trip t
Parameters	
f^s	Setup cost to locate any number of chargers at site s
g^s	Cost of each charger to be installed at site s
α	Cost of charging time
$u_{v,1}$	Initial charge of bus v
$\overline{u}_v$	Maximum permissible battery charge of bus v
$\underline{u}_v$	Minimum permissible battery charge of bus v
Δ_{vt}	Energy required to drive trip t made by bus v
ρ^s	Power output of chargers at site s
$y_{\max}$	Maximum permissible charging time for any bus at any time (set to 45 minutes)
$\overline{y}^s_{vt}$	Maximum feasible charging duration at site s for bus v after trip t
$\overline{N}^s$	Maximum number of chargers that can be installed at site s
δ^{1s}_{vt}	Energy required to deadhead from end of trip t made by bus v to charging site s
δ^{2s}_{vt}	Energy required to deadhead from charging site s to start of trip $t+1$ made by bus v
δ^{3}_{vt}	Energy required to deadhead from end of trip t made by bus v to start of trip $t+1$
δ^*_{uivj}	Energy required to deadhead from end of trip i of block u to start of trip j of block v
σ_{vt}	Scheduled start time of trip t made by bus v
ϵ_{vt}	Scheduled end time of trip t made by bus v
τ^{1s}_{vt}	Time required to deadhead from end of trip t made by bus v to charging site s
τ^{2s}_{vt}	Time required to deadhead from charging site s to start of trip $t+1$ made by bus v
τ^{3}_{vt}	Time required to deadhead from end of trip t made by bus v to start of trip $t+1$
τ^*_{uivj}	Time required to deadhead from end of trip i of block u to start of trip j of block v
L	Minimum layover time required between trips

Table 3.3: Summary results for case study.

Metric	Result
BEB-BRP Solution Time (s)	2.05
Number of Backup Buses	22
BEB-OCL Solution Time (s)	371
BEB-OCL Objective Value	$23,431,006
Number of Charger Visits	161
Mean Charge Duration	15.1 min.

Table 3.4: Charging sites and sizes for case study.

Charging Site	Number of Chargers
Bothell P&R	1
Burien TC	1
Federal Way TC	1
Northgate TC	4
South Renton P&R	1

Table 3.5: Candidate charging sites in King County case study.

Site Name	Abbreviation	Latitude	Longitude
Auburn Transit Center	Auburn TC	47.3069193	-122.2316235
Aurora Village Transit Center	Aurora Village TC	47.774162	-122.340623
Bothell Park & Ride	Bothell P&R	47.759428	-122.201043
Burien Transit Center	Burien TC	47.4693256	-122.3403857
Federal Way Transit Center	Federal Way TC	47.3179917	-122.3056636
Kenmore Park & Ride	Kenmore P&R	47.757213	-122.241200
Kent Transit Center	Kent TC	47.3836401	-122.2346441
Northgate Transit Center	Northgate TC	47.701365	-122.327236
South Base	South Base	47.495809	-122.286190
South Renton Park & Ride	S Renton P&R	47.4718414	-122.2147887

Table 3.6: Parameter values and sources, where relevant, for South King County case study.

Parameter	**Meaning**	**Value**	**Source**
$\underline{u}_v$	Minimum battery level	46.6 kWh	King County [55]
$\overline{u}_v$	Maximum battery level	396.1 kWh	King County [55]
$\overline{y}^s_{vt}$	Maximum charging duration	Varies	GTFS [58]
Δ_{vt}	Trip distance	Varies	GTFS [58]
δ^{1s}_{vt}, δ^{2s}_{vt}, δ^{3}_{vt}	Deadhead distances	Varies	Openrouteservice [89]
τ^{1s}_{vt}, τ^{2s}_{vt}, τ^{3}_{vt}	Deadhead times	Varies	Openrouteservice [89]
ρ^s	Charger power	450 kW	New Flyer [87]
f^s	Site cost	\$200,000	Assumed
g^s	Charger cost	\$698,447	TRB [111]
$\overline{N}^s$	Maximum number of chargers	4	Assumed
α	Charging time cost	\$13,870	US FTA [28]

Table 3.7: Basic parameters for notional routes used in simple case study.

Route	A	B	C
Distance (mi.)	15	25	15
Time (min.)	40	90	45
Layover Time (min.)	20	30	15
Headway (min.)	20	30	60
Start	(0, -7.5)	(-12.5, 0)	(0, -7.5)
End	(0, 7.5)	(12.5, 0)	(12.5, 0)

Table 3.8: Characteristics of candidate charging sites for simple case study.

Site	**x**	**y**	**f^s**	**g^s**	**N^s**
W	-12.5	0	\$500,000	\$698,447	4
NW	-10	1	\$50,000	\$698,447	4
S	0	-7.5	\$500,000	\$698,447	4

Table 3.9: Additional parameter values for sensitivity analysis.

Parameter	**Value**
$\overline{u}_v$	400 kWh
$\underline{u}_v$	0 kWh
α	\$2,000
δ^3_{vt}	0
$\rho^W, \rho^{NW}, \rho^S$	300 kW

Chapter 4

MINIMUM–DELAY OPPORTUNITY CHARGING SCHEDULING FOR ELECTRIC BUSES

4.1 Introduction

Battery-electric buses (BEBs) make up a significant and growing share of the global transit vehicle fleet. Over 60,000 such vehicles were sold worldwide in 2022, about 5% of global bus sales [47]. While China has been the world leader in electric bus adoption for many years, BEBs are beginning to see greater adoption worldwide. In the United States, the Bipartisan Infrastructure Law of 2021 allocated over $5 billion to help agencies purchase low-emissions transit vehicles and charging infrastructure [30]. Worldwide, BloombergNEF projects that 50% of buses will be battery-powered by 2032, a milestone passenger cars are not expected to reach for a further ten years [102].

As this rapid transition commences, agencies who replace some or all of their conventionally fueled buses with BEBs face several planning challenges across various time scales. Some key long-term decisions are bus fleet composition over time in terms of fuel type and/or battery size [92, 97] as well as charging infrastructure design such as the location, number, and power output of chargers [4, 76]. After these decisions have been made, transit agencies must determine how to efficiently operate BEBs each day while providing good passenger service and maintaining charged batteries on all vehicles. The preferred approach for many agencies is to rely on low-power overnight charging at bus depots in order to mimic traditional bus operating patterns in which refueling is not a significant concern. However, in many cases agencies have to use high-power chargers during the day to effectively extend bus ranges without utilizing a larger battery [63, 73, 75]. This approach is commonly referred to as *opportunity charging* and sometimes *on-route* or *layover charging*. Transit operators

who rely on opportunity charging need to schedule charging for all buses each day to make efficient use of their limited charging resources.

Charging scheduling has often been approached as a joint problem with vehicle scheduling, in which each bus that operates on a given day is assigned to a sequence of trips typically referred to as a *block* [93]. Although combining these phases can help develop vehicle schedules well suited to BEBs, it is important to note that bus schedules are usually determined only once every few months, but agencies need to optimize charging schedules repeatedly at a variety of time scales. Models focused primarily on determining vehicle schedules (such as Liu and Ceder [71]) or charging locations (such as McCabe and Ban [76]) generally design charging schedules based on average or expected conditions over their longer-range planning horizons, but within that time frame, operating conditions can easily change enough to require charging schedule adjustments. For example, weather and ridership patterns significantly impact energy consumption, and predictions of these variables improve as each service day approaches. For optimal daily operations, transit agencies should adjust their planned charging schedules frequently to account for these variations. Agencies also may wish to revise charging schedules within a single day to respond to changes such as vehicle breakdowns and traffic congestion—for example, it may be best to postpone a scheduled charge when a bus is running behind schedule, allowing it to resume service sooner and minimize delays. A revised charging schedule can be used both to improve operations and to provide more accurate estimates of future trip delays to passengers.

Researchers have developed various approaches to charge scheduling over the previous several years, but this work describes a novel modeling approach to the problem. First, our model prioritizes the passenger experience, minimizing departure delays across all trips in a day. We do this by explicitly modeling the queuing process at charging stations to track exactly when each bus is ready to start all of its trips. Our model propagates any delays across trips served by the same bus, ensuring their cascading impacts are captured accurately. Second, our focus on accurate delay tracking makes the model applicable to nearly any operating conditions. Rather than constraining all trips to leave on time like

most other works in the literature, our model handles scenarios where zero delay is not achievable. It produces actionable results under all conditions and extends naturally to robust or stochastic applications where energy consumption and traffic conditions make on-time departures impossible.

Our precise modeling approach produces some computational challenges. In particular, the mixed-integer linear programming model includes many binary variables and "big-M" constraints. To mitigate this issue, we develop both an exact solution method and a polynomial-time heuristic algorithm. Both approaches decompose the complete problem into a series of easier-to-solve problems with easily interpretable structure. The exact method uses Combinatorial Benders (CB) decomposition to split the problem into a master problem that contains all binary variables and a subproblem with all continuous variables, which reduces the burden of the big-M constraints. This restructuring is natural for our problem: the binary variables reflect high-level decisions about the ordering of charging throughout the day, whereas the continuous variables track the exact timing of events, including charging times and trip departures. The Combinatorial Benders approach exploits this structure in an iterative framework.

Our Select—Sequence—Schedule (3S) heuristic algorithm was inspired by the CB approach and is based on a similarly explainable decomposition. As in Combinatorial Benders, we make binary decisions first and continuous decisions second. To expedite this process, 3S initially relaxes the queue tracking constraints that link different buses to each other, so that we can *select* when charging occurs with a separate linear program for each bus. We then establish the *sequence* in which buses visit each charger with a simple sorting operation and finally *schedule* the exact charger plugin times, charging durations, and delays with the same approach as in the CB subproblem. The 3S method has polynomial-time complexity in the worst case, so it can be used to solve real-world instances quickly. Our experiments show that despite its lack of a performance guarantee, 3S reliably identifies good feasible solutions (and often optimal solutions) much faster than exact approaches do.

In summary, the main contributions of this work are as follows:

- A mixed-integer linear programming model for recharging scheduling that exactly tracks queuing behavior at chargers, propagates delays across trips completed by the same bus, and minimizes the total delay.

- An exact solution method based on Combinatorial Benders decomposition to solve this computationally demanding model.

- A polynomial-time heuristic method, motivated by the exact solution method, which generates good solutions quickly for complex real-world transit networks.

The remainder of this paper is organized as follows. Section 4.2.1 provides a review of relevant literature. Section 4.3 describes our methodology and mixed-integer programming formulation. Section 4.4 describes our exact solution method based on Combinatorial Benders decomposition, while Section 4.5 documents the related heuristic. Section 4.6 applies our methods to both a simple notional transit network and the real Seattle-area bus network. Section 4.7 summarizes findings and concludes the paper.

4.2 Literature Review

4.2.1 Recharging Scheduling for BEBs

Recharging scheduling for BEBs connects to a larger transit planning process that designs routes, timetables, and vehicle and driver schedules. In traditional transit planning with diesel buses, these problems are addressed sequentially rather than simultaneously, as each step is already complex to optimize [15]. Replacing diesel buses with BEBs further complicates the traditional transit planning process by introducing new challenges related to capital investments in buses and chargers, location of chargers, possible schedule revisions, and charging scheduling [93]. Because these various stages of the BEB planning process have significant overlap, researchers have studied recharging scheduling on its own as well as its integration with other decisions.

We first summarize the literature on charging scheduling integrated with other decisions, typically related to either charging infrastructure or vehicle schedules. When combining charge scheduling with other decisions, it is common to make significant assumptions to limit computational complexity, and these assumptions limit the models' applicability to day-to-day operations. For example, Esmaeilnejad et al. [26] proposed a stochastic optimization model to determine both charger locations and recharging schedules along a single bus route, where charging may take place at intermediate stops but is penalized based on passenger waiting costs. However, this model assumes buses will always charge to 100% state of charge between a pair of consecutive trips, which is only appropriate for buses with small battery capacities. McCabe and Ban [76] developed a deterministic model for simultaneously locating chargers and scheduling charger usage. Charger capacity is accounted for with predetermined "conflict sets" based on posted timetables, which could result in unexpected queuing at chargers if buses are delayed. Gairola and Nezamuddin [34] proposed a comprehensive robust model to minimize the costs of converting to BEBs by optimizing battery sizes, charging infrastructure, and charging schedules. Their model allowed recharging at terminals during scheduled layover time and handled capacity by discretizing time into unit-length intervals and ensuring the number of buses charging at a terminal did not exceed the number of chargers there, similar to McCabe and Ban [76]. However, neither of these approaches extends well to daily operations where unusually high energy demand or exogenous delays might make it impossible for buses to stay on schedule and charge only within scheduled layover time.

Charging scheduling is also commonly incorporated into vehicle block design, which for diesel buses is classically addressed with the vehicle scheduling problem (VSP). A comprehensive review of BEB VSP publications can be found in Perumal et al. [93], which references 23 such works. Among these, only six considered partial recharging of batteries and only seven accommodated multiple vehicle types, while none include both of these elements simultaneously. Both of these attributes are essential for daily recharging scheduling—many transit agencies operate both standard-length and articulated buses, for example, and par-

tial recharging is a natural strategy for large batteries—but present major computational challenges to incorporate in a model that also designs blocks.

Recognizing the need for a precise charge scheduling model focused on daily operations, some publications treat infrastructure and vehicle schedules as fixed and focus purely on charging scheduling. Abdelwahed et al. [2] developed two related formulations for charge scheduling at terminals and found that an event-based discretization scheme was preferable to using uniform time intervals. Both models sought to minimize the total cost of electricity based on time-of-use (TOU) prices and constrained charging to take place during the scheduled layover time. Liu et al. [70] developed a robust model for scheduling daily recharging, both at the depot and at terminals. Their model allows flexible recharging durations and power levels, respects charging station capacity, and minimizes the total cost of energy based on TOU pricing. They consider uncertainty in buses' energy consumption, but assume that buses always complete all trips as scheduled and require charging to occur during scheduled layover time. Using a different modeling paradigm, Lacombe et al. [67] studied an optimal control problem intended to minimize both energy costs and schedule deviations, where the bus schedule is encoded as a route-specific headway rather than a timetable. They developed a decomposition strategy based on Lagrangian relaxation and local heuristics to apply the method in practice. This approach is much more flexible than restricting charging to scheduled layover time, but is only appropriate for systems where maintaining a target headway is more important than matching an advertised schedule. Finally, we note that there are various works such as Manzolli et al. [74] and Brinkel et al. [10] that focus exclusively on optimizing depot charging. This is a fundamentally different challenge from our setting that is focused on opportunity charging at terminals (possibly including, but not limited to, depots), so we do not review those in detail here.

4.2.2 Combinatorial Benders Decomposition

Most approaches to optimal charge scheduling result in a mixed-integer programming problem, making them difficult to solve. We handle this challenge using a tailored version of

Benders decomposition, a classical algorithm for mixed-integer programming that was first proposed over 60 years ago [6]. The central idea of this approach is to separate the problem into a master problem (MP) containing integer variables and a subproblem (SP) containing continuous variables. In each iteration, the MP is solved to obtain candidate values of all integer variables. These fixed values are an input to the SP, which verifies the feasibility and potential optimality of the candidate solution. If the SP determines that the current candidate solution cannot be an optimal solution, one or more Benders cuts are generated and added to the MP to exclude it (along with, ideally, many other solutions), then the MP is solved again. The procedure repeats until an optimal solution is found [95].

Benders decomposition is often used in stochastic programming applications, where after fixing a limited number of first-stage variables, the second-stage problem decomposes into several independent problems that can be solved quickly. However, in the last two decades researchers have identified additional classes of problems where the approach can outperform standard branch-and-bound or branch-and-cut algorithms. Hooker and Ottosson [45] introduced the idea of logic-based Benders decomposition, wherein cuts are generated not based on the linear programming dual of the subproblem but a so-called "inference dual" that generalizes LP duality. Their approach has been applied successfully to a variety of problems but appears especially effective in cases where specialized constraint programming methods can be applied to the subproblem, including some types of scheduling problems [44].

Codato and Fischetti [19] soon after developed the idea of Combinatorial Benders (CB) cuts for a specific class of mixed-integer linear programs with logical constraints. Their approach largely follows Hooker and Ottosson [45], but they derived a tailored method for finding cuts for their particular problem class and demonstrated its performance benefits on some example problems. A major selling point of this method is that it eliminates the computational problems caused by big-M constraints; once variable values have been set by the MP, the corresponding logical constraints are either included or excluded from the SP. As such, the CB approach can avoid the usual problem of big-M values giving a poor linear

programming relaxation and resulting bounds.

It should be noted that a direct application of the Benders (standard or combinatorial) algorithm can have poor performance for a variety of reasons. These issues and strategies to mitigate them are reviewed thoroughly in Rahmaniani et al. [95]. Among the keys to a successful Benders implementation are initializing the MP with a set of strong cuts to aid in finding feasible solutions, using heuristics to generate good solutions and their correspondingly strong Benders cuts, and embedding Benders cuts within a branch-and-cut algorithm to limit the amount of redundant computations. We found these strategies were critical to make the CB approach competitive with an off-the-shelf solver.

4.2.3 Summary

The worldwide growth of BEBs has produced a significant body of literature on charging scheduling. However, most of these approaches are more suited to long-range planning than day-to-day or real-time scheduling. When charging scheduling is embedded within a model primarily focused on designing a charging infrastructure network or vehicle blocks, it tends to be simplified and restrictive. Even models focused purely on charge scheduling such as Abdelwahed et al. [2] and Liu et al. [70] are not applicable under all conditions, since they assume buses run on schedule and charging can take place within scheduled layover time. There is a need for models that can provide useful output even under challenging conditions when avoiding departure delays is impossible, so that bus operators are provided with actionable instructions rather than being told a problem instance is infeasible. Such models should be accompanied with efficient algorithms so they can be run repeatedly as conditions evolve or forecasts of ridership, traffic, and weather conditions improve.

Based on this research gap, we propose one such model for charging scheduling. Our approach emphasizes quality of service by seeking to minimize departure delays across all trips. We precisely quantify the impact of queuing at chargers and ensure delays are propagated across trips. The resulting model and methods for its solution are presented next in Sections 4.3–4.5.

4.3 Mathematical Programming Formulation

4.3.1 Problem Setting, Assumptions, and Modeling Approach

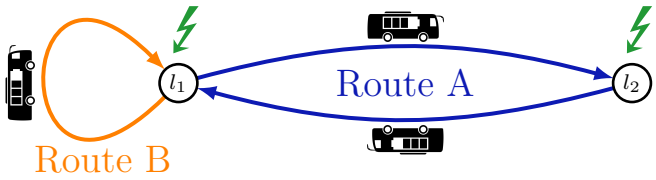


Figure 4.1: Simple example of the type of BEB system considered in this study.

We consider a general setting in which fast chargers with predetermined power outputs have already been installed at some terminals of a BEB system. Figure 4.1 shows a simple illustration of the type of transit network considered in this study. In this basic example, two bus routes (A and B) operate across two terminals and each terminal has a single charger installed. A bus may recharge at either of the two charging sites when out of service in between trips, as long as the bus is at the corresponding terminal and the charger is not already occupied. Note that in our approach, a single bus may serve more than one route (referred to as *interlining*) and/or use multiple different chargers during a day. A single charger also may serve buses on any number of different routes, so that infrastructure is shared as efficiently as possible. Bus trips may take place on a one-way route between distinct terminals (i.e., Route A) or a loop route that starts and ends at the same terminal (i.e., Route B).

Our approach relies on the following assumptions:

1. Buses rely on slow charging overnight at the depot in addition to fast charging during the day. Each bus has a known state of charge when entering service at the start of

the day and a minimum state of charge it must have when returning to the depot at the end of service.

2. Charger locations, trip schedules, and vehicle blocks have been determined in advance and cannot be altered. The agency's primary goal is to adhere to the posted schedule as much as possible, i.e., to minimize delays.

3. Buses may only charge at terminals when passengers are not onboard, never at intermediate stops.

4. Deadheading to chargers does not require significant time or energy. Buses may only uses chargers that are sufficiently close to a trip terminal that the driving distance and time can be neglected.

5. All buses in the network have sufficient battery capacity to complete all trips in between successive charging opportunities.

6. Charging behavior is linear, proportional to the maximum power output of each charger.

Assumption 1 acknowledges the different technologies, time scales, and constraints faced by transit agencies during depot versus opportunity charging. Agencies can be expected to have different priorities for charging scheduling in these different environments. When buses use opportunity chargers during the service day, maintaining schedule adherence is the greatest priority, whereas overnight charging should allow more time to optimize for other objectives such as energy costs and battery health. Bus bases are also less likely to be subject to queuing for chargers due to the much lower cost of lower-power chargers.

Assumption 2 reflects our intended problem setting as described in Section 4.1. Assumption 3 ensures that the current passenger level of service is maintained. Assumption 4 is common in the recharging literature and simplifies the model by making cumulative energy

consumption independent of any charging decisions and their corresponding binary decision variables.

The last two assumptions concern battery range and charging behavior. Assumption 5 is necessary to ensure operations are always feasible with fast chargers; any such infeasible blocks can be discarded from the analysis. Assumption 6 is another common assumption made to limit computational complexity. It can provide a good enough approximation even if the real charging behavior is nonlinear, since charging is typically linear until the state of charge exceeds about 80% when power must be reduced to protect the battery [84].

Following these assumptions, suppose we are given a set of BEBs $\mathcal{V}$, each of which is scheduled to complete a specific set of trips; that is, every vehicle has been assigned to a predetermined block. We represent each trip as a tuple $i = (b_i, n_i)$ where b_i is the bus ID and n_i is the trip number; i.e., bus A completes trips $(A, 1), (A, 2), (A, 3)$, etc. Let $\mathcal{T}$ be the set of all trips completed by any bus. We will use the shorthand indices i and $j \in \mathcal{T}$ throughout this formulation to limit indices, but keep in mind that each of these refer to a specific trip completed by a specific bus as part of a predetermined block. For each trip $i \in \mathcal{T}$, the timetable provides a scheduled departure time σ_i and scheduled duration τ_i (so that the scheduled end time is $\sigma_i + \tau_i$). If a trip i has an immediate predecessor in its block, that predecessor is denoted i^-. Let $\theta(i)$ be the set of trips completed by the same bus prior to i (i.e., all of its predecessor trips).

Figure 4.2 illustrates a simple example of how to interpret these parameters on a timeline for two buses labeled A and B. The colored segments of the timeline represent times when the buses are scheduled to be in service and unavailable to charge. Considering the trips served by bus B in this example, we have $\theta((B, 3)) = \{(B, 1), (B, 2)\}$, $\theta((B, 2)) = \{(B, 1)\}$, and $\theta((B, 1)) = \emptyset$. Likewise, $(B, 3)^- = (B, 2)$ and $(B, 2)^- = (B, 1)$.

Our approach to charging scheduling is aimed at accurately tracking trip departure delays. Tracking and propagating delay within a mixed-integer linear programming paradigm is challenging because of the underlying queuing behavior. To address this challenge, we define multiple continuous variables for each trip that track the time of certain events. This indexing

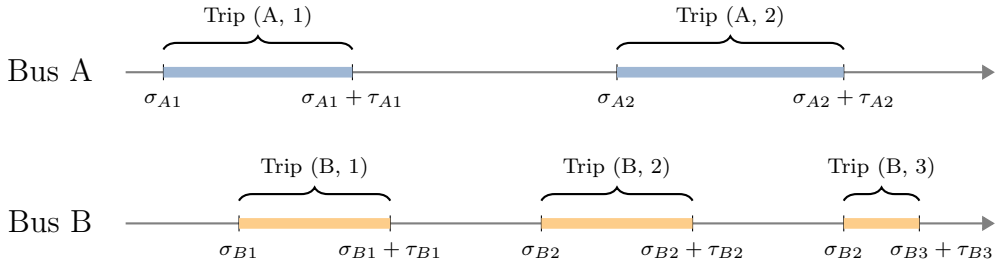


Figure 4.2: Timeline of operations for two example buses.

strategy can be seen as similar to the "discrete event optimization" approach of Abdelwahed et al. [2]. For each trip i, let d_i be a decision variable giving its departure delay and p_i be a decision variable giving the plug-in time after completion of trip i, i.e., the time the bus that completes i connects to a charger. To ensure that delay is tracked properly, we also require that if there is no charging after i, the value of p_i equals the completion time of trip i, i.e., $p_i = d_i + \sigma_i + \tau_i$. Let t_i^l be the amount of time spent charging after trip i at charger l.

To illustrate exactly how these variables are related to one another and the problem parameters, Figure 4.3 shows a simple example based on the schedule of Figure 4.2. Suppose buses A and B both use the same charger l_1 after completing their first trip of the day. Bus B arrives first, immediately plugs in at time $p_{B1} = \sigma_{B1} + \tau_{B1}$ and charges for time $t_{B1}^{l_1}$. It then starts trip $(B, 2)$ on time, since $\sigma_{B1} + \tau_{B1} + t_{B1}^{l_1}$ is earlier than the scheduled start time σ_{B2} of its next trip. When Bus B finishes charging, the charger becomes available and Bus A plugs in at time $p_{A1} = p_{B1} + t_{B1}^{l_1}$. Bus A charges for length $t_{A1}^{l_1}$. Figure 4.3 shows that Bus A departs its second trip late because of the time it spent waiting at the charger; specifically, its departure delay is $d_{A2} = p_{A1} + t_{A1}^{l_1} - \sigma_{A2}$.

Table 4.1 compiles all the set, parameter, and decision variable definitions used in this work. Sections 4.3.2–4.3.6 next describe the formulation of the mixed-integer linear program we developed for recharging scheduling, including the objective function and the constraints that track queuing, delays, and battery levels.

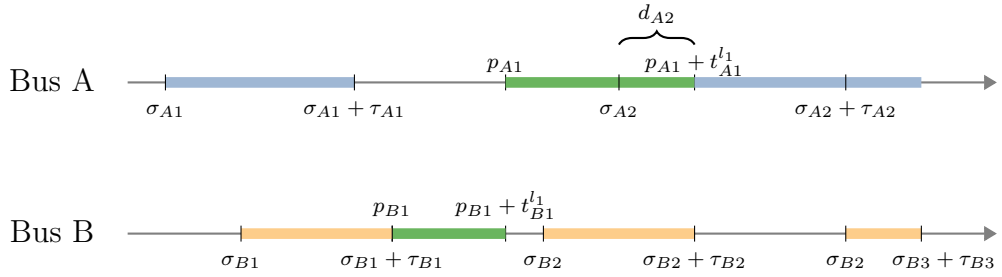


Figure 4.3: Illustration of variable meanings for d_i, t^l_i, and p_i with their relation to schedule parameters σ_i and τ_i.

4.3.2 Objective Function

$$\min_{\substack{d_i, p_i, t^l_i, \\ x^l_i, y^l_{ij}}} \quad \sum_{i \in \mathcal{T}} d_i \tag{4.1}$$

The objective (4.1) is to minimize the total amount of departure delay across all trips $i \in \mathcal{T}$ served by BEBs. While the objective is straightforward, setting the appropriate value of d_i for each trip i requires carefully formulated constraints, as described in the following subsections.

4.3.3 Queue Tracking and Delay Propagation

Our queuing and delay tracking constraints are based on the relationships between plugin time, charging time, and delay depicted in Figure 4.3. We first describe how to correctly set the plugin time. Recall that if bus b_i does not charge after trip i, we define the plugin time p_i to be the end time of trip i. Since charging can only increase the value of p_i, the trip completion time provides a universal lower bound on p_i, encoded by Constraints (4.2).

$$p_i \geq d_i + \sigma_i + \tau_i \qquad \forall\ i \in \mathcal{T} \tag{4.2}$$

If b_i does use some charger l after trip i, then setting its plugin time correctly is more

Table 4.1: Notation definitions.

Sets	
$\mathcal{V}$	Buses
$\mathcal{T}$	Trips
$\mathcal{C}$	Chargers
$\mathcal{A}$	Arcs
Decision Variables	
x_i^l	Binary variable indicating whether the bus on trip i uses charger l afterwards
y_{ij}^l	Binary variable indicating whether charging arc (l, i, j) is used
d_i	Delay at start of trip t
p_i	Plugin time of charging after trip i
t_i^l	Charging duration at l after completion of trip i
Parameters	
b_i	ID of bus that completes trip i
n_i	Trip number of trip i
α_i	Minimum total charge gain by end of trip i
β_i	Maximum total charge gain by end of trip i
ρ^l	Power output of charger l
$\overline{t_i^l}$	Maximum charging duration at site l after trip i
σ_i	Scheduled start time of trip i
τ_i	Scheduled duration of trip i
M	"Big M", arbitrary large number

complex because it may have to queue. We therefore introduce some auxiliary binary variables and related constraints. Let $\mathcal{C}$ be the set of all chargers that are installed. We say that a charger $l \in \mathcal{C}$ *serves* a trip $i \in \mathcal{T}$ if b_i uses l after finishing trip i. Let y_{ij}^l be a binary decision variable that equals 1 if charger l serves trip j immediately after trip i and 0 otherwise. By "immediately," we mean only that no other trips are served in between these two; there may be a large gap between the charging completion time for trip i and the plugin time of trip j. Likewise, let $x_i^l = 1$ if charger l serves trip i and 0 otherwise. Then we account for queuing at the charger with Constraints (4.3):

$$p_j \geq p_i + t_i^l - M(1 - y_{ij}^l) \qquad \forall\ (l,i,j) \in \mathcal{A} \tag{4.3}$$

Constraints (4.3) impose additional lower bounds on plugin time to account for queuing. These "big M" constraints encode the logic that if trip j is served by charger l immediately after trip i, then b_j cannot plug in until b_i has unplugged. Written as an equivalent logical constraint, we would have

$$y_{ij}^l = 1 \implies p_j \geq p_i + t_i \qquad \forall\ (l,i,j) \in \mathcal{A} \tag{4.4}$$

In Constraints (4.3) and (4.4), $\mathcal{A}$ is the set of "arcs" connecting trips i and j that may be served sequentially by charger l. These arcs are discussed in more detail in Section 4.3.4.

With the plugin time values constrained properly, it is straightforward to set the delay for each trip. Note that whether or not bus b_i charges after i, it is ready to re-enter service at time $p_i + \max_{l \in \mathcal{C}} \left\{t_i^l\right\}$. The departure delay of trip i is then $d_i = \max\left(0, p_{i^-} + \max_{l \in \mathcal{C}} \left\{t_{i^-}^l\right\} - \sigma_i\right)$. This maximum operator can be represented exactly with Constraints (4.5) and (4.6).

$$d_i \geq p_{i^-} + t_{i^-}^l - \sigma_i \qquad \forall\ l \in \mathcal{C},\ \ i \in \mathcal{T} : n_i \geq 2 \tag{4.5}$$

$$d_i \geq 0 \qquad \forall\ i \in \mathcal{T} \tag{4.6}$$

4.3.4 Charger Sequencing

The binary variables y_{ij}^l explicitly track the sequence of trips that are served by each charger in the network. Our model therefore includes constraints to ensure that the optimal values of y_{ij}^l encode a valid sequence connecting all trips served by each charger. By a valid sequence, we mean that (1) any time a charger is used (i.e., $x_i^l = 1$), that trip appears somewhere in

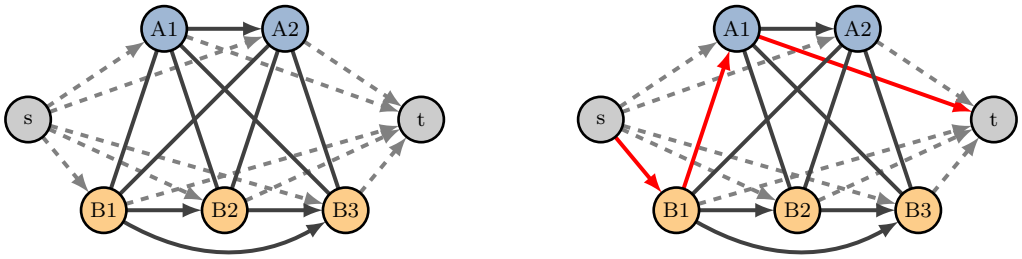
the sequence, and (2) every trip i served by charger l has exactly one trip before and one trip after it in the sequence.

To formulate these constraints, we model the charging sequence for each charger l as a path through a network in which each node corresponds to a trip (a charging opportunity) and arcs represent feasible charging connections. That is, if trip j can be served by the charger immediately after trip i, then arc (l, i, j) is included in the network. In general, a directed arc joins every pair of edges for the same bus (since we can't travel back in time to charge at, say, trip 1 after trip 3). On the other hand, an undirected edge joins any pair of trips completed by different buses, because we don't know *a priori* which order these trips will be optimally served in. For instance, it could be optimal to use arc (l, i, j) in the solution even if trip j is scheduled to end before trip i, because trip j is delayed in the optimal solution.

We also introduce two dummy nodes to model the initial (node s) and final (node t) idle state of each charger. These nodes are necessary because the first and last trips to be served by each charger have no predecessor and successor trip, respectively, so they require slightly different constraints. Moreover, there is no way to know *a priori* which trips will be first and last, so we instead construct dummy nodes to handle these special cases. Now, a feasible sequence for charger l corresponds to a path through this virtual network from s to t. Note that this network model is similar to and inspired by those used in vehicle scheduling approaches (e.g., the maximum flow formulation of the VSP [15]).

Figure 4.4 illustrates our network model for the simple example of two buses and one charger previously depicted in Figures 4.2 and 4.3. Figure 4.4a displays the network structure including the trip and dummy nodes as well as the arc set $\mathcal{A}$ (where arrows indicate directed arcs and lines indicate undirected edges). Figure 4.4b shows in red the path corresponding to the charging decisions from Figure 4.3, where buses charge at l after trips $(B, 1)$ and $(A, 1)$. The corresponding arc variable values are $y^l_{sB1} = y^l_{B1A1} = y^l_{A1t} = 1$, and $y^l_{ij} = 0$ for all other arcs.

Following this network representation, a feasible sequence for each charger can be enforced



(a) Network structure. (b) Example feasible solution from Figure 4.3

Figure 4.4: Example network for charger l_1 serving buses A and B.

using the familiar network flow constraints (4.7)–(4.9).

$$\sum_{i \in \mathcal{T}} y^l_{si} = 1 \qquad \forall\ l \in \mathcal{C} \tag{4.7}$$

$$\sum_{i \in \mathcal{T}} y^l_{it} = 1 \qquad \forall\ l \in \mathcal{C} \tag{4.8}$$

$$\sum_{j:(l,j,i) \in \mathcal{A}} y^l_{ji} - \sum_{j:(l,i,j) \in \mathcal{A}} y^l_{ij} = 0 \qquad \forall\ l \in \mathcal{C}, \forall\ i \in \mathcal{T} \tag{4.9}$$

Constraints (4.7) and (4.8) ensure that exactly one arc leaves the source dummy node and exactly one arc arrives at the sink dummy node, respectively, for each charger. Constraints (4.9) ensure connectivity of the charging sequence; the number of arcs entering and leaving each trip node must be equal. Notably, the network flow constraints (4.7)–(4.9) do not preclude the existence of subtours in paths through the network. It is not necessary to include subtour elimination constraints because any solution containing a subtour cannot possibly be optimal for our problem; in fact, it would result in unbounded delay due to Constraints (4.3) and (4.5).

The model must also establish the relationship between the x^l_i and y^l_{ij} variables. Charging after trip i corresponds to visiting its trip node in Figure 4.4, meaning one arc leaves (or, equivalently, enters) that node, so we have:

$$x_i^l = \sum_{j:(l,i,j)\in\mathcal{A}} y_{ij}^l \qquad \forall\ l \in \mathcal{C},\ \ i \in \mathcal{T} \tag{4.10}$$

If multiple chargers are located close to the end of trip i, then each bus must be restricted to only use one charger at a time. Let $x_i^l = 1$ if charger l serves trip i and 0 otherwise. Then this condition is simple to enforce with Constraints (4.11):

$$\sum_{l\in\mathcal{C}} x_i^l \leq 1 \qquad \forall\ i \in \mathcal{T} \tag{4.11}$$

4.3.5 State of Charge Management

The final subsystem of constraints tracks charging throughout the service day to make sure that each bus's state of charge stays within an acceptable range (e.g., between 20% and 95%). First, we need to relate the continuous charging duration t_i^l to the binary x_i^l. t_i^l must be forced to zero if $x_i^l = 0$; otherwise, it can take any feasible value. Let $\overline{t_i^l}$ be the maximum possible length of charging at charger l after trip i. Let $u_i^{\max}$ be the usable battery capacity of the bus serving trip i and ρ^l be the power output of charger l. We set the value of $\overline{t_i^l}$ with Equation (4.12):

$$\overline{t_i^l} = \begin{cases} u_i^{\max}/\rho^l & \text{if trip } i \text{ ends at } l \\ 0 & \text{otherwise} \end{cases} \qquad \forall\ l \in \mathcal{C},\ \ i \in \mathcal{T} \tag{4.12}$$

Equation (4.12) reflects that bus b_i can only use charger l if trip i ends at l. Additionally, the maximum possible charging duration is the battery capacity of b_i divided by the charging rate of l. With $\overline{t_i^l}$ defined as such, t_i^l is restricted by Constraints (4.13):

$$0 \leq t_i^l \leq \overline{t}_i^l x_i^l \qquad \forall\ l \in \mathcal{C},\ \ i \in \mathcal{T} \tag{4.13}$$

We also need constraints to ensure that each bus charges enough during the day in order to complete its scheduled service. Following Assumption 6, with linear charging it is straightforward to determine lower and upper bounds (α_i, β_i) on the cumulative charge gain through the end of each trip i. The complete state-of-charge constraints are given by Constraints (4.14).

$$\alpha_i \leq \sum_{j \in \theta(i)} \sum_{l \in \mathcal{C}} \rho^l t_j^l \leq \beta_i \qquad \forall\ i \in \mathcal{T} \tag{4.14}$$

To understand how α_i and β_i can be calculated for all trips, consider a simple example. Suppose bus A must keep its battery level between 50 and 200 kWh all day and enters service with 125 kWh of charge. Suppose its first two trips each require 50 kWh of energy. Then, its charge level when starting trip 2 must be no more than its maximum (200 kWh) and no less than enough to complete trip 2 without falling below its minimum ($50 + 50 = 100$ kWh). Since its battery level is $125 - 50 = 75$ kWh after completing trip 1, it must gain between 25 and 125 kWh of charge after that trip. Accordingly, $l_{A1} = 25$ and $u_{A1} = 125$. This simple example for trip (A, 1) easily generalizes across all trips.

4.3.6 Complete MILP Formulation

The complete mixed-integer linear programming formulation for recharging scheduling as developed in Sections 4.3.2–4.3.5 is given by Equations (4.15)–(4.28):

$$\min_{\substack{d_i,p_i,t_i^l,\\ x_i^l,y_{ij}^l}} \quad \sum_{i\in\mathcal{T}} d_i \tag{4.15}$$

$$\text{s.t.} \quad d_i \geq p_{i^-} + t_{i^-}^l - \sigma_i \qquad \forall\ l\in\mathcal{C},\ i\in\mathcal{T}: n_i \geq 2 \tag{4.16}$$

$$p_i \geq \sigma_i + d_i + \tau_i \qquad \forall\ i\in\mathcal{T} \tag{4.17}$$

$$p_j \geq p_i + t_i^l - M\left(1 - y_{ij}^l\right) \qquad \forall\ (l,i,j)\in\mathcal{A} \tag{4.18}$$

$$\sum_{i\in\mathcal{T}} y_{si}^l = 1 \qquad \forall\ l\in\mathcal{C} \tag{4.19}$$

$$\sum_{i\in\mathcal{T}} y_{it}^l = 1 \qquad \forall\ l\in\mathcal{C} \tag{4.20}$$

$$\sum_{j:(l,j,i)\in\mathcal{A}} y_{ji}^l - \sum_{j:(l,i,j)\in\mathcal{A}} y_{ij}^l = 0 \qquad \forall\ l\in\mathcal{C},\ i\in\mathcal{T} \tag{4.21}$$

$$x_i^l = \sum_{j:(l,i,j)\in\mathcal{A}} y_{ij}^l \qquad \forall\ l\in\mathcal{C},\ i\in\mathcal{T} \tag{4.22}$$

$$\sum_{l\in\mathcal{C}} x_i^l \leq 1 \qquad \forall\ i\in\mathcal{T} \tag{4.23}$$

$$0 \leq t_i^l \leq \overline{t_i^l} x_i^l \qquad \forall\ l\in\mathcal{C},\ i\in\mathcal{T} \tag{4.24}$$

$$\alpha_i \leq \sum_{j\in\theta(i)} \sum_{l\in\mathcal{C}} \rho^l t_j^l \leq \beta_i \qquad \forall\ i\in\mathcal{T} \tag{4.25}$$

$$d_i \geq 0 \qquad \forall\ i\in\mathcal{T} \tag{4.26}$$

$$x_i^l \in \{0,1\} \qquad \forall\ l\in\mathcal{C},\ i\in\mathcal{T} \tag{4.27}$$

$$y_{ij}^l \in \{0,1\} \qquad \forall\ (l,i,j)\in\mathcal{A} \tag{4.28}$$

4.4 Exact Solution Method: Combinatorial Benders Decomposition

The formulation (4.15)–(4.28) is difficult to solve for large instances, largely due to the binary variables y_{ij}^l and the corresponding big-M constraints (4.18). Because the number of y variables and big-M constraints scales with the square of the number of trips, solving the problem with an off-the-shelf MIP solver is not possible for many large instances. We develop

two strategies to deal with this computational challenge. In this section, we describe an exact solution approach based on Combinatorial Benders (CB) decomposition, which circumvents the typical issues of weak linear programming relaxations caused by big-M constraints. Section 4.5 later describes a polynomial-time heuristic with randomization that can generate a large number of good solutions quickly. The heuristic helps accelerate the convergence of the CB algorithm on smaller problems and also shows good performance as a standalone method for difficult problems on real networks where exact algorithms are unacceptably slow.

4.4.1 Overview of Combinatorial Benders Decomposition

Figure 4.5 shows a high-level overview of the CB algorithm we implemented. We begin by using a randomized heuristic (described in detail in Section 4.5) to generate a set of feasible charging schedules. The best solutions are used to create initial CB cuts to restrict the MP's feasible region and we further strengthen the MP by adding cuts derived from the state-of-charge Constraints (4.14). Following this initialization step, the main CB loop begins. In each iteration, we first solve the MP to get a candidate solution in terms of the binary x and y variables only. Assuming we obtain one, we solve the subproblem, which aims to find a solution that outperforms the current incumbent. If the SP is infeasible, indicating that this solution cannot outperform the incumbent, we use the CB cut generation procedure to exclude it. If it is feasible, our current MP solution becomes the new incumbent. We update it as such, then re-solve the SP and add CB cuts.

Sections 4.4.2-4.4.4 next describe the individual steps of our Combinatorial Benders implementation in detail, including the MP and SP formulations as well as cut generation.

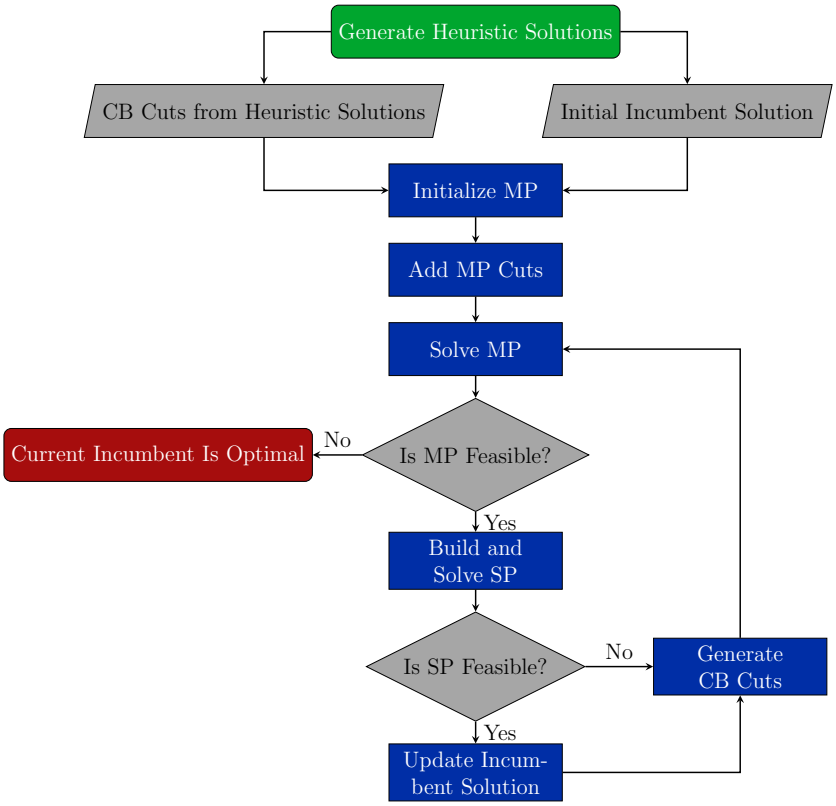


Figure 4.5: Overview of Combinatorial Benders solution process.

4.4.2 Master Problem

$$\min_{x_i^l, y_{ij}^l} \quad \sum_{(l,i,j)\in\mathcal{A}} c_{ij}^l y_{ij}^l \tag{4.29}$$

$$\text{s.t.} \quad \sum_{i\in\mathcal{T}} y_{si}^l = 1 \qquad \forall\ l \in \mathcal{C} \tag{4.30}$$

$$\sum_{i\in\mathcal{T}} y_{it}^l = 1 \qquad \forall\ l \in \mathcal{C} \tag{4.31}$$

$$\sum_{j:(l,j,i)\in\mathcal{A}} y_{ji}^l - \sum_{j:(l,i,j)\in\mathcal{A}} y_{ij}^l = 0 \qquad \forall\ l \in \mathcal{C},\ i \in \mathcal{T} \tag{4.32}$$

$$x_i^l = \sum_{j:(l,i,j)\in\mathcal{A}} y_{ij}^l \qquad \forall\ l \in \mathcal{C},\ i \in \mathcal{T} \tag{4.33}$$

$$\sum_{(i,j)\in\mathcal{A}(\mathcal{S})} y_{ij}^l \le |\mathcal{S}| - 1 \qquad \forall\ \mathcal{S} \subset \mathcal{T} : |\mathcal{S}| \ge 2 \tag{4.34}$$

$$x_i^l \in \{0,1\} \qquad \forall\ l \in \mathcal{C},\ i \in \mathcal{T} \tag{4.35}$$

$$y_{ij}^l \in \{0,1\} \qquad \forall\ (l,i,j) \in \mathcal{A} \tag{4.36}$$

The CB master problem (4.29)–(4.36) includes all constraints on the binary variables x and y from the complete formulation, including the charger sequencing constraints (4.30)–(4.32) and x/y relation (4.33). The MP also introduces new constraints (4.34), which ensure the MP solution does not contain any subtours. In Constraints (4.34), $\mathcal{A}(\mathcal{S})$ denotes the set of arcs contained in the trip subset $\mathcal{S}$, i.e.,

$$\mathcal{A}(\mathcal{S}) = \{(l,i,j) \in \mathcal{A} : i \in S, j \in S\} \tag{4.37}$$

Subtour elimination constraints were not necessary in the complete problem formulation (4.15)–(4.28) because any subtour would produce unbounded delay. But in the CB framework, that delay calculation is delegated to the subproblem, so the master problem will frequently generate solutions with subtours if Constraints (4.34) are omitted. We generate these constraints "lazily" within a branch-and-cut framework by checking for subtours and

adding constraints only when subtours are detected, but having to handle these constraints directly in the MP is still a disadvantage of the CB method for our problem.

Because the full problem objective function (4.15) is not dependent on the binary variables, the MP essentially has a constant objective. In practice, this means that we may select any objective we wish, so we use a heuristic objective function to encourage the MP to generate integer solutions that are more likely to result in optimal delays as calculated by the subproblem. To do so, we define cost coefficients c_{ij}^l for all arcs $(l, i, j) \in \mathcal{A}$ as the lower bound on delay that would result from using that arc.

This lower bound is derived from the queuing and delay constraints. Let j^+ be the successor trip of j on its same block, analogous to the previously defined i^-. If arc (l, i, j) is used, then trip j^+ might be delayed due to bus b_j queuing while b_i charges. Specifically, $p_j \geq p_i$ by (4.18) and because $t_i^l \geq 0$. Accounting for (4.17), we have $p_j \geq \sigma_i + \tau_i$, and then $d_{j^+} \geq \sigma_i + \tau_i - \sigma_{j^+}$. So, we set the MP cost for most arcs according to Equation (4.38):

$$c_{ij}^l = \max\{0, \sigma_i + \tau_i - \sigma_{j^+}\} \tag{4.38}$$

All dummy arcs (see Figure 4.4) and any arcs (l, i, j) for which j has no successor trip are assigned a cost of 0. Setting the costs with Equation (4.38) disincentivizes the solver from setting $y_{ij}^l = 1$ when doing so is guaranteed to delay trip j^+. In our experiments, this objective function gave better performance than a constant objective that essentially chose feasible solutions at random.

Master Problem Cuts

When initializing the MP, we also add some cuts to strengthen its formulation. Adding cuts helps to generate MP solutions that are more likely to be feasible for the SP, which decreases the total number of iterations that must be performed.

We generate MP cuts based on the state-of-charge constraints (4.25), using them to derive lower bounds on the total number of times each bus must charge. Essentially, we

convert Constraints (4.25) to restrict the MP decision variables x_i^l rather than the continuous variables t_i^l, which are now delegated to the subproblem. To derive the MP cuts, we first note that whenever a bus visits any charger after trip i, the maximum amount of charge it can gain is $u_i^{\max}$, the usable battery capacity of bus b_i. Now, for any trip i, let l_i denote the charger located at the final stop of i, i.e., $l_i = l \in \mathcal{C} : \overline{t_i^l} > 0$. Then we obtain valid inequalities (4.39) based on Equations (4.12) and (4.24)–(4.25):

$$\sum_{j \in \theta(i)} u_i^{\max} x_i^{l_i} \geq \alpha_i \quad \forall \ i \in \mathcal{T} \tag{4.39}$$

Dividing the inequalities (4.39) by $u_i^{\max}$ and adding the ceiling operator to the right-hand side since the left-hand side is always integral, we have:

$$\sum_{j \in \theta(i)} x_i^{l_i} \geq \left\lceil \frac{\alpha_i}{u_i^{\max}} \right\rceil \quad \forall \ i \in \mathcal{T} \tag{4.40}$$

Equations (4.40) yield one MP cut for every trip in the problem instance. In practice, we do not need to add all of these to the MP. Rather, we create these cuts by proceeding through each block in order of trip sequence and calculating the right-hand side $\left\lceil \frac{\alpha_i}{u_i^{\max}} \right\rceil$. Each trip we progress to will add one variable $x_i^{l_i}$ to the left-hand side and may or may not increase the right-hand side, since α_i is nondecreasing as we progress through a block. We therefore add a new cut to the MP each time the right-hand side increases to a new integer value; if it does not increase, then the cut for trip i is dominated by the one for i^- and does not improve the MP formulation.

4.4.3 Subproblem

Solving the CB master problem provides a candidate solution in terms of the x and y variable values. The subproblem's role is then to check whether it could be an optimal solution to the complete problem by verifying both that it is a feasible choice of integer variable values and that it can minimize delay. This optimality criterion is evaluated by tracking an incumbent

solution and its total delay value, denoted z^*, as described in Codato and Fischetti [19].

For a given MP iteration k, let $\mathcal{A}_1^{(k)}$ be the set of arcs used in the solution, i.e., $(l,i,j) \in \mathcal{A}_1^{(k)}$ if $y_{ij}^{l(k)} = 1$. Likewise, let $\mathcal{T}_0^{(k)}$ be the set of trip-charger pairs for which charging is skipped, i.e., $(l,i) \in \mathcal{T}^{(k)}$ if $x_i^{l(k)} = 0$. Then the SP for iteration k is:

$$\min_{d_i, p_i, t_i^l} \quad \sum_{i \in \mathcal{T}} d_i \tag{4.41}$$

$$\text{s.t.} \quad \sum_{i \in \mathcal{T}} d_i \leq z^* - \epsilon \tag{4.42}$$

$$d_i \geq p_{i^-} + t_{i^-}^l - \sigma_i \qquad \forall\ l \in \mathcal{C},\ i \in \mathcal{T} :\ n_i >= 2 \tag{4.43}$$

$$p_i \geq \sigma_i + d_i + \tau_i \qquad \forall\ i \in \mathcal{T} \tag{4.44}$$

$$p_j \geq p_i + t_i^l \qquad \forall\ (l,i,j) \in \mathcal{A}_1^{(k)} \tag{4.45}$$

$$\alpha_i \leq \sum_{j \in \theta(i)} \sum_{l \in \mathcal{C}} \rho^l t_j^l \leq \beta_i \qquad \forall\ i \in \mathcal{T} \tag{4.46}$$

$$t_i^l = 0 \qquad \forall\ (l,i) \in \mathcal{T}_0^{(k)} \tag{4.47}$$

$$0 \leq t_i^l \leq \overline{t_i^l} x_i^l \qquad \forall\ l \in \mathcal{C},\ i \in \mathcal{T} \tag{4.48}$$

$$d_i \geq 0 \qquad \forall\ i \in \mathcal{T} \tag{4.49}$$

The CB subproblem (4.41)–(4.49) aims to minimize the delay based on the binary decisions from the MP. Constraint (4.42) requires that the subproblem solution must improve the current incumbent objective value z^* by some small tolerance ϵ, which ensures that when the MP is found to be infeasible, the current incumbent is optimal. Constraints (4.45) and (4.47) replace the corresponding big-M constraints from the full problem, enforcing bounds on plugin time p_i and charging time t_i^l only as needed based on the current MP solution. The remaining SP constraints are identical to those from the complete formulation in Section 4.3.6.

4.4.4 Combinatorial Benders Cuts

As shown in Figure 4.5, when the SP is infeasible, we add Combinatorial Benders cuts to exclude the incumbent MP solution. The key to creating these cuts is to identify a minimal infeasible subsystem (MIS), also called an irreducible infeasible subsystem (IIS) of the infeasible SP instance. An MIS is an inclusion-minimal set of rows of the SP constraint matrix, where "inclusion-minimal" means that if any single constraint is removed, the resulting subsystem of constraints admits a feasible solution. Codato and Fischetti [19] describe an algorithm to generate multiple such MISs for any MP solution that yields an infeasible SP.

An MIS of the subproblem identifies a set of MP variables—corresponding to the instances of the conditional constraints (4.45) and (4.47) included in the MIS—that have forced the problem to be infeasible. To make sure that the MP does not produce a solution with this same set of variables again, we add a CB cut to exclude it. For this purpose, we can represent the MIS as two sets of indices: one set $\mathcal{M}_x$ corresponding to Constraints (4.45) and another $\mathcal{M}_y$ corresponding to Constraints (4.47). That is, $\mathcal{M}_x = \big\{(l,i) \in \mathcal{M} : x_i^l = 0\big\}$ and $\mathcal{M}_y = \big\{(l,i,j) \in \mathcal{M} : y_{ij}^l = 1\big\}$. Given such an MIS, a CB cut is formed by the simple inequality (4.50):

$$\sum_{(l,i)\in\mathcal{M}_x} x_i^l + \sum_{(l,i,j)\in M_y} (1 - y_{ij}^l) \geq 0 \tag{4.50}$$

Equation (4.50) enforces that at least one of the binary variables x_i^l and y_{ij}^l included in the MIS must change its value in order to obtain a feasible solution.

4.4.5 Implementation Details

We implemented the CB algorithm using Python and the Gurobi solver via the `gurobipy` package. Following Figure 4.5, we begin by initializing the master problem and adding CB cuts for heuristic solutions. In our experiments, we generate cuts for any heuristic solution with an objective value within 50% of the best identified objective. Once the MP

is initialized, we run the CB algorithm within a branch-and-cut framework using Gurobi's callback capabilities. When a new optimal solution to the MP is detected, we first check if it contains any subtours. If it does, we cut off any such subtours by adding Constraints (4.34) as lazy constraints. If it does not, we progress to solving the subproblem and generating CB cuts. To generate an MIS each time that the SP is infeasible, we use Gurobi's `computeIIS()` function, which identifies a single MIS out of many possibilities. Identifying multiple MISs (and consequently multiple cuts) can help the algorithm converge faster, so each time an MIS is found, we remove one constraint from that MIS at random, verify that the relaxed SP model is still infeasible, and run `computeIIS()` again to find a new unique MIS. We repeat this procedure until the relaxed SP model is feasible, which typically results in finding several MISs per CB iteration.

4.5 Heuristic Solution Method: Select–Sequence–Schedule

Deriving the CB algorithm for our charge scheduling formulation (4.15)–(4.28) inspired a heuristic algorithm that follows a similar pattern, but with much faster convergence. The heuristic design is based on a few key insights presented by the decomposition approach. First, most of the problem's complexity comes from the two interconnected decisions of selecting which trips include charging (the values of x_i^l) and the sequence in which these trips are served by each charger (the values of y_{ij}^l). Once these values are set, it is easy to determine the optimal charging durations and complete schedule of plugin/departure/delay times with the subproblem (4.41)–(4.49), which is just an LP. Additionally, the problem of selecting the order in which to serve trips may not be too difficult in practice once the charging trips have been selected. We can expect that the optimal charging order is unlikely to differ too much from simple first-in, first-out priority.

Based on this logic, we devised a heuristic algorithm based on relaxing and separating the MIP formulation (4.15)–(4.28). We relax the complicating queue constraints (4.18) and restructure the problem into three phases we call *selection*, *sequencing*, and *scheduling*, together forming what we call the 3S algorithm.

Figure 4.6 gives an overview of the 3S heuristic algorithm. The first step is to initialize some random cost parameters θ_i^l, which helps the algorithm explore a wider range of feasible solutions. Then, in the selection phase, we solve a separate LP for each bus to select a set of trips when it will charge. This is equivalent to setting the values of x or populating the set $\mathcal{T}_0^{(k)}$ in the CB approach. In the sequencing phase, we perform a simple sorting operation for each charger to set the order in which it serves trips. This is analogous to setting the y variable values or the set $\mathcal{A}_1^{(k)}$ from the CB master problem. In the final *scheduling* phase, we solve one more LP to optimize charging durations and the resulting delay given the selection and sequence decisions. This final LP is identical to the CB subproblem with the incumbent bound constraint (4.42) removed. The final output of any run of the 3S heuristic is a feasible solution to (4.15)–(4.28) and its corresponding objective value. Sections 4.5.1–4.5.3 next describe each of the three phases in detail.

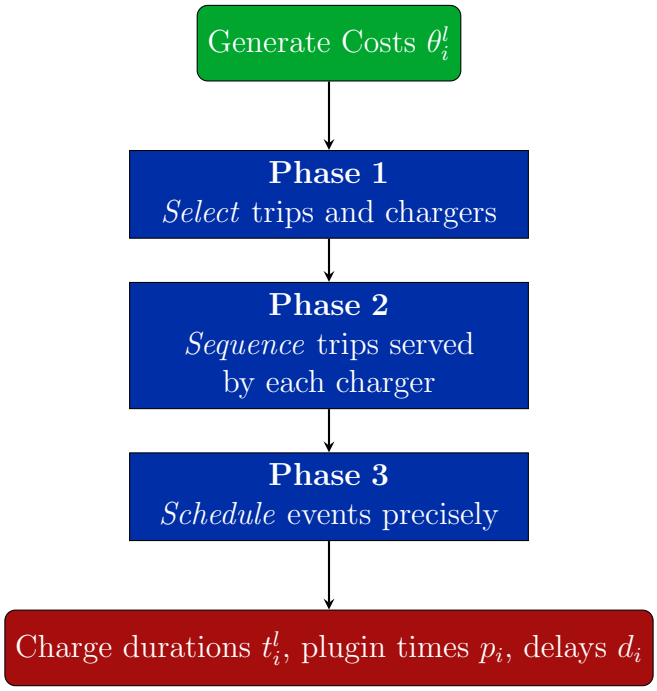


Figure 4.6: Flowchart of 3S heuristic solution procedure.

4.5.1 Phase 1: Trip Selection

The purpose of Phase 1 is to set feasible values of the binary charging decisions x_i^l for all trips i and chargers l. To do this, we relax Constraints (4.18), which are used to accurately calculate the delays d_i but do not affect feasibility with respect to state of charge. As a result, the plugin time variables p_i are controlled only by Constraints (4.17), so that $p_i = d_i + \sigma_i + \tau_i$ for all trips i, and p_i can be removed from the formulation. The binary variables y_{ij}^l are also no longer needed when Constraints (4.18) are removed from the MIP formulation, since they have no impact on the objective function.

One more simplification is needed to eliminate the binary variables x_i^l and make the Phase 1 problem a linear program. Rather than including Constraints (4.11) that rely on x_i^l and ensure each bus uses only one charger after each trip, we select the charger to be used at random from the set $\mathcal{C}(i)$ of accessible chargers for each trip i. Let $\widehat{l_i}$ be the randomly chosen charger for trip i. Then we ensure only one charger is used by defining a new upper bound on the charging time with Equation (4.51):

$$\widehat{t_i^l} = \begin{cases} \overline{t_i^l} & \text{if } l == \widehat{l_i} \\ 0 & \text{otherwise} \end{cases} \tag{4.51}$$

With the single-charger limit handled as such, the binary variables x_i^l can be excluded, yielding the Phase 1 LP (4.52)–(4.56):

$$\min_{d_i, t_i^l} \quad \sum_{i \in \mathcal{T}} \left[d_i + \sum_{l \in \mathcal{C}} \theta_i^l t_i^l \right] \tag{4.52}$$

$$\text{s.t.} \quad d_i \geq \sigma_{i^-} + d_{i^-} + \tau_{i^-} + t_{i^-} - \sigma_i \qquad \forall\ i \in \mathcal{T} :\ \ n_i \geq 2 \tag{4.53}$$

$$\alpha_i \leq \sum_{j \in \theta(i)} \sum_{l \in \mathcal{C}} \rho^l t_j^l \leq \beta_i \qquad \forall\ i \in \mathcal{T} \tag{4.54}$$

$$0 \leq t_i^l \leq \overline{t_i^l} \qquad \forall\ i \in \mathcal{T},\ \ l \in \mathcal{C} \tag{4.55}$$

$$d_i \geq 0 \qquad \forall\ i \in \mathcal{T} \tag{4.56}$$

The Phase 1 objective function (4.52) includes a randomization term $\sum_{i \in \mathcal{T}, l \in \mathcal{C}} \theta_i^l t_i^l$, where θ_i^l is randomly sampled from a suitable distribution. This term helps to generate a variety of feasible solutions through repeated runs of the 3S algorithm. Since there will often be many different charging patterns that result in the same optimal delay in Phase 1 (especially if the minimum total delay is zero), adding this random coefficient to the charging times helps 3S to consider a greater variety of potentially good first-stage solutions. A distribution should be chosen so that θ_i^l takes on a variety of both positive and negative values. Negative values of θ_i^l are useful to produce results in which a bus distributes its charging over a greater number of trips, which may help to reduce queuing. It is also important that the values of θ_i^l are not so large that the randomization term dominates the delay term in the objective function, which could result in poor performance. In our experiments, we sample the values of θ_i^l from a normal distribution with mean 0 and standard deviation 0.5.

The Phase 1 LP (4.52)–(4.56) is separable across all buses in the system, since the conditional plugin time constraints (4.18) that linked trips on different blocks have been removed. Phase 1 therefore consists of N_b independent linear programs, where N_b is the number of buses serving the trips $\mathcal{T}$. Note that since the number of variables only scales with the number of trips served by a single vehicle, each LP is quite small.

Solving the Phase 1 LP (4.52)–(4.56) yields first-stage charging durations t_i^l and delays

d_i for all trips i and chargers l. We convert the charging durations to their equivalent binary values x_i^l, i.e. $x_i^l = 1$ if $t_i^l > 0$ and $x_i^l = 0$ otherwise. These binary decisions also correspond to the set of non-charging trips $\mathcal{T}_0^{(k)}$ from the CB master problem. Each trip i for which $\sum_{l \in \mathcal{C}} x_i^l = 1$ is also assigned a first-stage plugin time $p_i = d_i + \sigma_i + \tau_i$ when it is expected to start charging. These form the inputs to Phase 2.

4.5.2 Phase 2: Charger Sequencing

In Phase 2, we process the charging decisions from Phase 1 into a sequence for every charger that is used. For each charger l, we simply sort all trips i for which $x_i^l = 1$ by their first-stage plugin times p_i. Sorting these trips gives us a simple mapping to a feasible set of charging arcs $\mathcal{A}_1^{(k)}$: for any consecutive pair of trips i and j, we add arc (l, i, j) to $\mathcal{A}_1^{(k)}$. For each charger we also add arcs connecting to the initial and final dummy nodes, respectively. That is, if i is the first and j is the last trip served by charger l, we add the arcs (l, s, i) and (l, j, t).

4.5.3 Phase 3: Event Scheduling

Phase 3 takes the results obtained from Phases 1 and 2—essentially, all the binary decisions of the complete MIP formulation—and uses them to define an exact charging schedule to minimize delays. Since the first two phases populated the sets $\mathcal{T}_0^{(k)}$ and $\mathcal{A}_1^{(k)}$, we have all the inputs necessary to construct an instance of the subproblem (4.41)–(4.49), though we need to either set the incumbent delay value z^* to an arbitrarily high value or remove Constraint (4.42) entirely. Since the subproblem is a linear program whose number of variables and constraints scales roughly with the number of trips in an instance, it can generally be solved quite quickly. Solving the subproblem yields final values of charging duration and delay for each trip that account for queuing. Note that in Phase 3 we allow for changing the charging durations originally output by Phase 1; the purpose of Phase 1 is to make the binary decisions of when and where to charge, but we relax the problem to an LP for efficiency. Also note that including the charging time constraints (4.55) in the Phase 1 LP for each bus ensures that Phase 3 is always feasible; the purpose of Phase 3 is to optimize charging durations

to exactly determine the minimum possible delay for the given set of charging trips and sequence for each charger.

4.6 Case Studies

We constructed a variety of instances in order to analyze the performance of the CB method and 3S heuristic. These instances were based on two different transportation networks. The first, described in Section 4.6.1, is a small notional network consisting of two bus routes served by a single shared charger. The small network allows us to evaluate the performance (in terms of solution time and optimality gap) of both our methods in comparison to directly solving the model with Gurobi. The second network, described in Section 4.6.2, is based on the actual transit system operated by King County Metro in the greater Seattle area. This example shows how the 3S heuristic can support transit operations at a real-world scale.

4.6.1 Simple Notional Network

To test the CB method and assess the performance of the 3S heuristic, we used a small test network originally presented in Appendix B of McCabe and Ban [76]. That original network was further simplified to include only two of the routes (A and C) and a single charger at their shared terminal, as sketched in Figure 4.7. Route A's headway was decreased from 20 to 30 minutes to limit the number of vehicles in the case study, but all other parameter values remained the same as reported in McCabe and Ban [76]. All buses have 400 kWh of usable battery capacity and consume energy at a rate of 3 kWh/mi (1.86 kWh/km); we test a variety of charger power levels from 300 to 500 kW. Table 4.2 documents the schedule and distance parameters of each route. With such a schedule, the network consists of 8 buses that complete 84 total trips.

4.6.2 King County Metro Network

The second test network is based on the transit system of South King County, WA, USA. We use this network to study the performance of the 3S heuristic, as it is too large to be

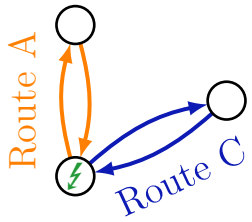


Figure 4.7: Simple case study network.

Table 4.2: Basic parameters for notional routes used in simple case study.

Route	A	C
Distance (mi.)	15	15
Time (min.)	40	45
Layover Time (min.)	20	15
Headway (min.)	30	60
One-Way Distance	15 mi. (24 km)	15 mi. (24 km)

solved to optimality by either Gurobi or our CB method. The case study includes some of the busiest routes planned for electrification in the near future: the RapidRide lines F and H as well as routes 131, 132, 150, 153, and 161. We collected relevant data including trip schedules (corresponding to σ_i and τ_i values) and distances as well as block sequences from the Metro GTFS feed [58] that was posted November 27, 2023. We used Wednesday, December 6, 2023 as a test date.

Data Collection

To filter down the GTFS data to the scope of our case study, we first identified all `service_id` values active on the case study date and all blocks and trips active for these service IDs. To set the cumulative charging bounds α_i and β_i for each trip i, we first determined the distance of each trip by matching it to its corresponding `shape_id` and the maximum `shape_dist_traveled`. We assumed each bus had a usable battery capacity of 300 kWh and an average energy consumption rate of 3 kWh/mi (1.86 kWh/km) for all trips. The

first and last trips of each block also had increased energy demand due to pull-out and pull-in trips from/to the depot. We assumed all buses were kept overnight at Metro's South Base and calculated pull-out/pull-in distances based on driving directions between the base and all relevant terminal locations, as calculated by the Openrouteservice API [89]. To set the upper bound on charging time $\overline{t_i^l}$ for all trips and chargers, we identified the final stop coordinates of each trip and calculated the driving distance to each charger (again using Openrouteservice) and assumed buses were able to use a charger if the distance to it was less than 0.25 miles (0.4 km).

The case study includes all blocks that serve trips on the RapidRide F Line and H Line as well as routes 131, 132, 150, 153, 161, and 165. Buses on these routes are served by one charger each at the Burien Transit Center (which serves the F Line, H Line, 131, 132, and 165), Renton Landing (the eastern terminal of the F Line), and Kent Station (the southern terminal of routes 150, 153, and 161). We assumed all chargers had an identical power output of 450 kW in each test case. Figure 4.8 maps all trips included in this scope as well as the three chargers considered.

In total, the King County network we analyze includes 42 buses that need to use fast chargers at some point during the day. An additional 17 blocks that serve these routes are excluded because they can be completed with depot charging alone. The 42 fast-charging buses complete a total of 493 trips on the test date.

Exogenous Delay Scenarios

We introduce two scenarios to model exogenous delays, which highlight the flexibility of our modeling approach and the 3S algorithm. In Scenario A, all trips run exactly on schedule. In Scenario B, the trip duration τ_i is increased by 40% for all trips with a scheduled departure between 7:00 a.m. and 9:00 a.m., which is meant to roughly simulate heavy traffic during these commute hours. In this case, buses are delayed enough that charging within scheduled layover time is impossible, and we search for a solution that yields the minimum possible total delay. The results in Section 4.6.3 show how our model adapts by shifting charging

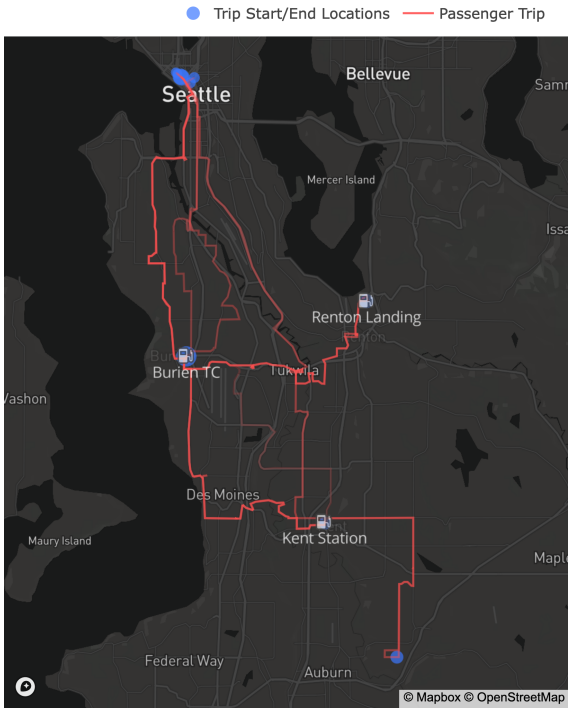


Figure 4.8: Map showing trips, terminals, and charger locations for the King County case.

later in the day, once buses have resumed operating on schedule.

4.6.3 Results

Simple Network

We solved the model for the simple network of Section 4.6.1 at a variety of different charger power levels from 300 to 500 kW. The two exact methods were subject to a time limit of one hour for each instance; the 3S heuristic was run 500 times for each case, though it was terminated as soon as a solution with zero delay was found, if applicable. The results are summarized in Table 4.3. For each method, we report the objective value of the best solution obtained, the time to find the first solution with that best objective value, and the total solution time. A dash is used to represent that the algorithm timed out.

In Table 4.3, we see that optimal delay decreases and eventually reaches zero as the

Table 4.3: Summary of results on simple case study instances. **BO**: best objective value (minutes of total delay). **T-BO**: time to find solution with best objective (s), **T-T**: total solution time (s). A dash indicates that the algorithm did not terminate.

	Direct Solve			***3S Heuristic***			***Benders***		
Power	**BO**	**T-BO**	**T-T**	**BO**	**T-BO**	**T-T**	**BO**	**T-BO**	**T-T**
300 kW	704	133	-	759	0.7	12	759	N/A	-
350 kW	71.4	15	57	73.7	0.05	9	71.4	1463	-
400 kW	5	56	66	5	0.7	10	5	0.7	47
450 kW	0	55	55	0	0.1	0.1	N/A	N/A	N/A
500 kW	0	138	138	0	0.4	0.4	N/A	N/A	N/A

power level increases. For these small instances involving only 8 buses, Gurobi solved the problem to optimality in 4 out of 5 cases. The comparative performance of the heuristic and CB methods varied depending on the instance. At the two lower power levels which represent more difficult instances, the Benders algorithm failed to prove optimality within the time limit, though in the 350 kW case it did successfully identify an optimal solution with 71.4 minutes of delay. With 400 kW chargers, the CB method outperforms the direct solution via Gurobi. As intended, the 3S algorithm quickly identifies an optimal solution with 5 minutes of total delay and the CB procedure proves its optimality in 47 seconds, whereas directly solving the problem took 66 seconds. For the 450 and 500 kW cases, the direct solution approach struggles to find a good solution, whereas 3S identifies a zero-delay solution very quickly and the entire CB procedure can be skipped because total delay can never be negative.

To illustrate the results in more detail, Figure 4.9 shows the complete timeline of activities for all buses in the 400 kW instance. Gray blocks on the timeline indicate that a bus is completing a passenger service trip that departed on time; delayed trips are shown in orange. A blue block in a bus's timeline indicates that it is plugged in at the charger. Looking at block CF1 for example, which completes 7 trips on Route C, we can see that it chargers after trips 1, 3, and 5, and all trips are on time. Blocks AR1 and AR2, with both complete more trips and require more charging, each have one trip that departs a few minutes late. We can

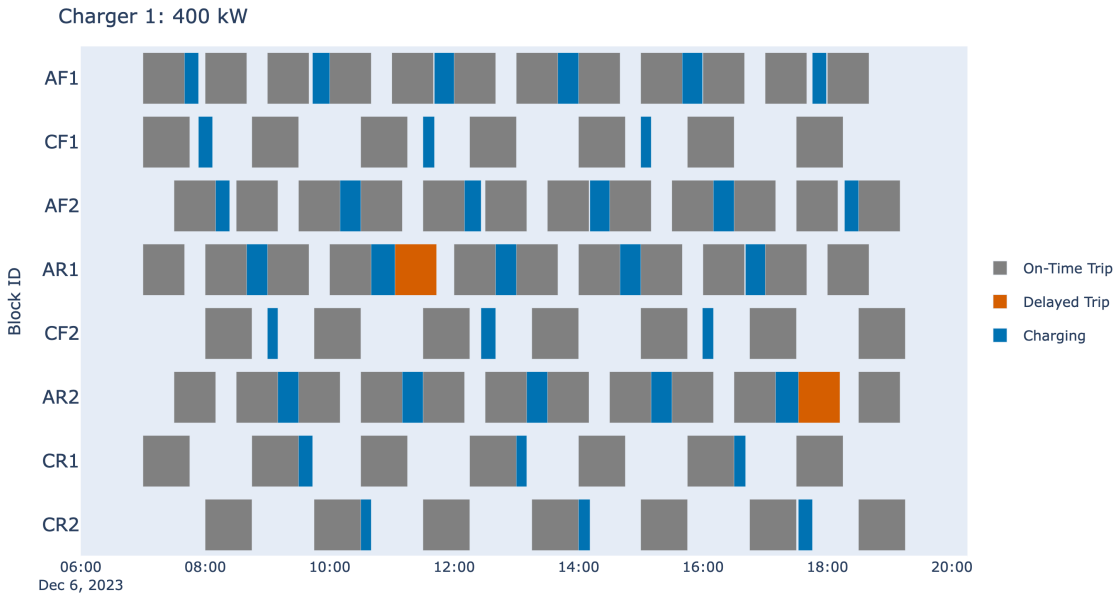


Figure 4.9: Optimized timeline of operations for the simple network with 400 kW chargers.

see that this optimal solution involves frequent short charges—in fact, all eight buses buses plug in every time they are able to (every other trip).

These results highlight both the benefits and limitations of our methods. First, we note that the 3S heuristic identifies a near-optimal (often optimal) solution very quickly in all cases. Its worst performance was for the 300 kW case, when the best solution it identified had only 7.8% more delay than the best one found by Gurobi. One key advantage of the 3S method is that its solution time does not depend significantly on the difficulty of the instance being solved; while neither of the exact methods converged within the time limit for the 300 kW instance, the heuristic produced good feasible solutions just as quickly as it did for other instances.

CB produced mixed results on our instances. We see that for the 400 kW instance, the CB method converges more quickly than solving directly, but it is much slower than Gurobi for the two harder instances. In the 350 kW case, it is notable that CB is able to improve the incumbent solution supplied by the heuristic, but it still does not terminate before the

time limit. With a 450 or 500 kW charger, integrating Benders with the 3S approach means that we can avoid running the Benders process entirely since 0 is the best possible feasible delay.

There are a few likely reasons for the limited benefits of the CB approach for our charge scheduling model. First, delegating all scheduling and delay calculation to the subproblem limits the information that is available to the master problem, which limits Gurobi's ability to speed up the solution process via presolve methods and cuts. Based on our results, it seems that the cuts generated by the CB approach are not particularly strong compared to the cuts Gurobi generates on its own when provided with the complete formulation. Second, the CB method is primarily used for problems in which the objective depends only on the integer variables. While we followed the recommendations from Codato and Fischetti [19] on adapting the method to a problem where the objective depends instead on continuous variables, we found that the approach is not as well suited to this type of problem. For one, convergence is inevitably slower compared to the typical CB application because it usually requires a large number of CB cuts to determine that the MP is infeasible and terminate the algorithm. Additionally, because the MP objective has no real meaning for the complete problem, we do not get any information on the optimality gap. Whereas in the typical CB application the Benders cuts are embedded within a branch-and-cut procedure and the search process continually improves upper and lower bounds on the optimal objective value, in our case we have an upper bound from the incumbent solution, but do not obtain any lower bound until the MP is infeasible and optimality is proven.

King County Metro Network

Table 4.4 documents the key results from applying the 3S heuristic to the King County network for the two delay scenarios. In Scenario A, the algorithm identifies an optimal solution with zero delay in a fraction of a second. In Scenario B where some delay is inevitable, it takes only 0.47 seconds to identify the best solution and just over a minute to run 500 iterations of 3S. 36 out of the 446 total trips are delayed in the best solution, with

Table 4.4: Summary of results on King County Metro network. **BO**: best objective value (minutes of total delay). **T-BO**: time to find solution with best objective (s), **T-T**: total solution time (s), **ND**: number of delayed trips.

Instance	BO	T-BO	T-T	ND
A	0	0.32	0.32	0
B	186.75	0.47	71	36

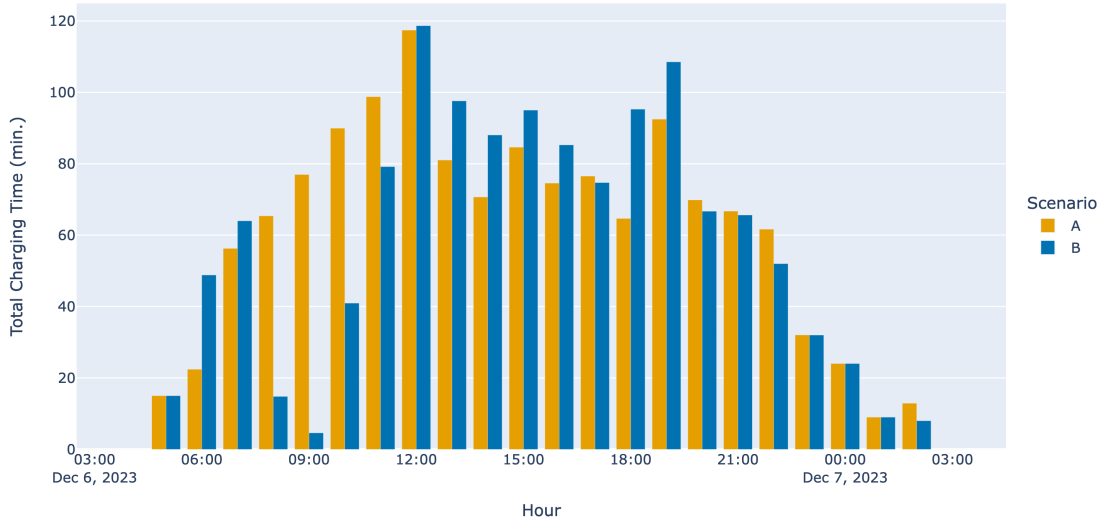


Figure 4.10: Total charging time by hour across all sites for the two scenarios.

a maximum delay of 16.5 minutes for a single trip.

To show how the 3S algorithm adapts to our simulated delays in Scenario B, Figure 4.10 plots the total amount of charging per hour for both scenarios. The charging duration on the y-axis is the sum across all three chargers considered, so the maximum possible value would be 180 minutes. We can see that in Scenario A, charging is fairly evenly distributed throughout the day, with decreased charging at the earliest and latest parts of the day when fewer buses are in service. When we introduce delays by increasing trip durations between 7:00 and 9:00 in Scenario B, our method shifts charging activities away from this period—

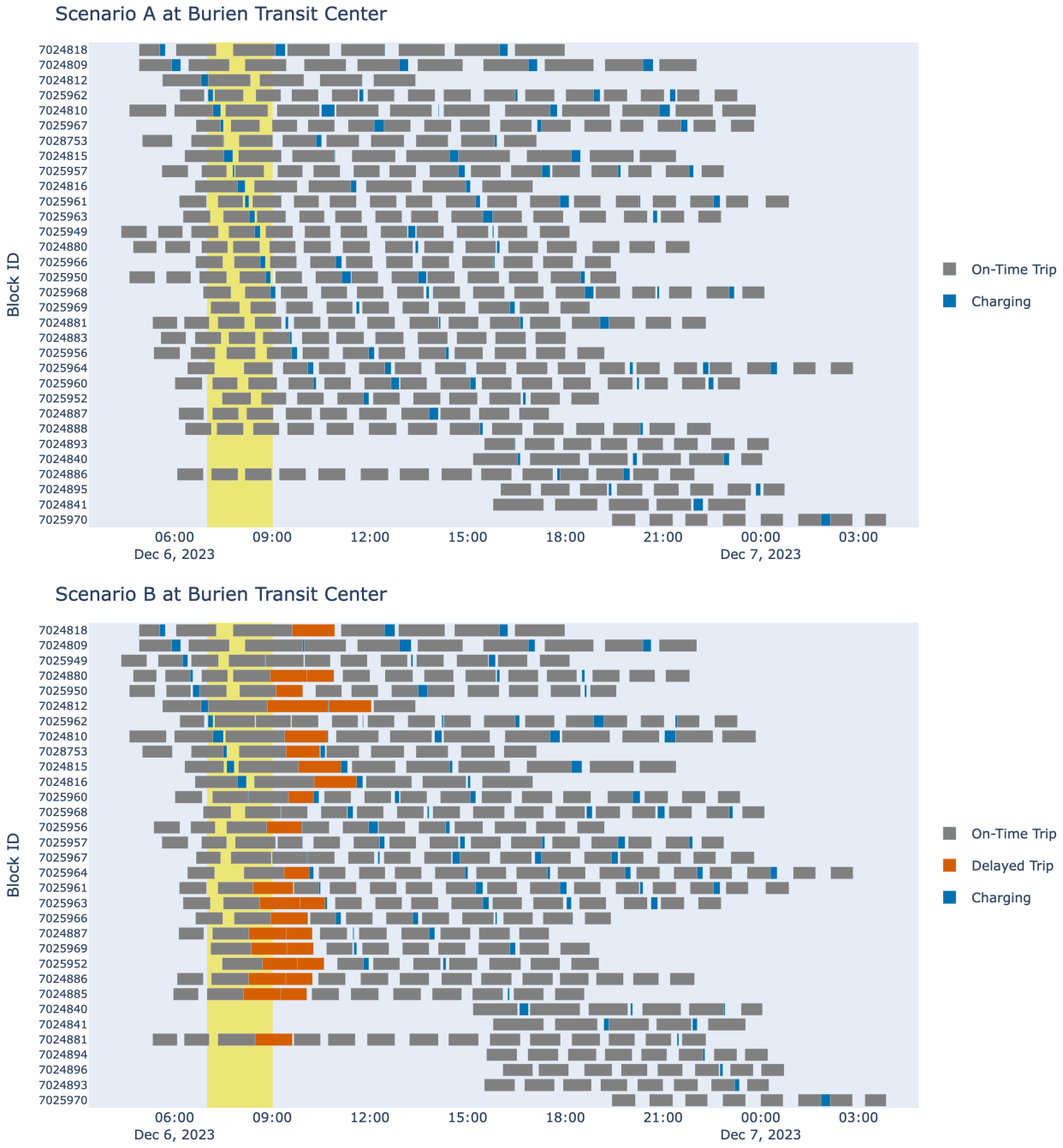


Figure 4.11: Timeline of scheduled charger utilization for Scenario A.

particularly between 8:00 and 11:00, when buses were likely to be running behind schedule due to a previous delay not caused by charging. Accordingly, there is more charging activity in the early morning from 6:00 to 7:00 as well as for most of the afternoon; starting around 8:00 p.m., the charging pattern looks nearly identical to that of Scenario A.

Figure 4.11 plots the timelines of charger utilization in Scenarios A and B in the same style as Figure 4.9. Figure 4.11 only shows buses that charge at the Burien Transit Center, which is the busiest of the three chargers. As before, gray blocks on the timeline represent on-time service trips, orange blocks represent delayed trips, and blue blocks indicate charging in Burien. The period from 7:00 to 9:00 a.m., when trips take longer than expected and force delays in Scenario B, is highlighted with a yellow background in the timelines for both scenarios.

Comparing both scenarios shown in Figure 4.11, we can see that many buses that charged in the morning commute hours in Scenario A instead charge earlier or later in the day under Scenario B, in agreement with Figure 4.10. In particular, we see for Scenario B that buses generally avoid charging prior to a trip that is delayed, in order to avoid prolonging those delays. The charger is used scarcely between about 8:00 and 11:00 a.m. to allow buses to catch back up to schedule, after which frequent charging resumes. Figure 4.11 also shows that most buses charge multiple times throughout the day; in fact, in both scenarios, the median number of times for a bus to charge is 3. Also note that in both scenarios, buses rarely charge for the full amount of layover time available to them. This result suggests in our test cases, individual blocks have plenty of layover time scheduled to allow for recharging, and queuing is the main potential source of delays.

4.7 Discussion and Conclusion

This study developed a novel model for scheduling within-day electric bus fleet recharging as well as exact and heuristic methods for its solution. Our method is unique in that it exactly tracks charger utilization and queuing with a focus on schedule adherence, but with a particularly flexible approach. Our model precisely sets delays and propagates them across

trips with the objective of minimizing total delays. Because of this precise yet flexible approach, our model still produces actionable results when operating conditions such as traffic delays or high energy consumption make it impossible to meet charging needs without delaying any trips. This is unique in the literature, as existing methods tend to constrain charging to take place during scheduled layover time and are not naturally extendable to scenarios where insufficient time is available.

Our exact solution method based on Combinatorial Benders decomposition outperforms a state-of-the-art commercial solver on some instances. However, our results show that this approach is still not efficient enough to be suitable for large real-world transit networks under demanding conditions. Although this method can provide a proof of optimality for some instances that provides theoretical value, it has considerable limitations. In particular, our usage of Combinatorial Benders for a problem in which the objective value is set solely by the subproblem is not ideal.

However, our Select–Sequence–Schedule heuristic algorithm that was inspired by the CB approach shows excellent results on our test instances. On our smaller test network instances in Section 4.6.1, the 3S method found good feasible solutions orders of magnitude faster than exact methods and its optimality gap was always below 8%. The two test scenarios on the King County Metro network from Section 4.6.2 demonstrated how the heuristic could quickly find a zero-delay solution under favorable conditions, as well as its ability to respond to exogenous delays by adjusting the recharging plan. Its strong performance in finding good feasible solutions suggests that 3S could also be run repeatedly for the purpose of real-time scheduling, taking into account up-to-the-minute information on delays, traffic conditions, and energy consumption predictions. This approach could help agencies respond to a variety of disruptions such as charger breakdowns while minimizing the ultimate service impacts.

Our method is also straightforward to adapt to support other objectives besides delay minimization. For instance, the objective function (4.15) can be adapted to incorporate other concerns such as time-dependent energy costs rather than focusing purely on delays. Agencies may also wish to prioritize delays on some trips more than others based on expected

passenger volumes or equity concerns. Such features can be added to our model and solution methods with minimal changes required.

Chapter 5

ACCOUNTING FOR UNCERTAINTY WITH REAL-WORLD BEB DATA

5.1 Introduction

5.1.1 Motivation

Some key inputs for the models introduced in Chapters 3 and 4 are widely recognized to be variable and unpredictable. Most significantly, passenger trip durations and energy demands are influenced by inherently random features including weather conditions, traffic, and passenger boardings and alightings. To test the models proposed in this dissertation under a wide range of realistic operating conditions, this section gathers and analyzes real-world data from King County Metro on both the on-time performance and energy consumption of BEBs. We analyze the data and develop statistical models that can be used to run simulations of system performance under dynamic conditions. These simulations are then used to evaluate the performance of the recharging scheduling model of Chapter 4 across a wide range of realistic scenarios and identify strategies to select model parameters that consistently yield good operations and minimal passenger delays.

5.1.2 Objectives

The guiding principle of this chapter is to understand how variation in BEB passenger trip durations and energy efficiency impacts overall system performance (in terms of passenger trip delays), especially when charging infrastructure and schedules are determined by the optimization models from Chapters 3 and 4. Because these optimization models are deterministic, they are based on only individual estimates of these key trip-level parameters without regard to the full distribution of possible values they could take. This chapter therefore

aims to quantify how much system performance differs under real-world parameter variations when decisions are made using a deterministic model. Repeatedly running the optimization models with different parameter setting strategies and testing simulated performance also reveals policies that yield good performance across a wide range of scenarios.

Several preliminary steps must be taken in order to reach this final objective. First, we must gather, clean, and analyze real-world data on schedule adherence and energy consumption to understand the key factors that influence these essential model parameters. The data can then be used to fit statistical models that identify the impacts of easily measurable trip parameters—such as elevation change and weather conditions—on trip durations and energy consumption. The statistical models provide insight into the factors driving these key parameters, but can also be used to generate predictions for simulations.

The objectives of this chapter can be summarized as follows:

- Analyze GTFS Realtime data to assess trends in bus on-time performance throughout the King County Metro network.

- Analyze ChargePoint data from Metro's current BEB fleet to understand trends in bus energy consumption.

- Fit statistical models of trip duration and energy consumption so that we can predict and simulate the values of these parameters for any trip in the Metro network.

- Incorporate these predictions into a system-wide simulation platform that tracks battery levels, charging patterns, and departure delays for all vehicles in order to quantify key performance metrics across a wide range of possible scenarios.

- Use the simulation platform to identify strategies for setting optimization parameters that yield solutions which are sufficiently robust to uncertainty.

5.2 Literature Review

Because predictions of BEB energy consumption have significant impacts on planning decisions including vehicle schedules and charging infrastructure, many studies have focused on estimating BEBs' energy usage in service. Before real-world data on BEB energy usage were widely available, research efforts typically used physics-based models to predict energy consumption based on simulated drive cycles, and this approach continues to be common. In recent years, several studies have also made use of data collected by BEBs currently in service to fit or validate models that predict energy usage under a variety of conditions.

The U.S. National Renewable Energy Laboratory has produced reports of real-world BEB energy consumption using onboard data loggers for a variety of bus operators including Foothill Transit [49] and Long Beach Transit [27] in California, Duluth Transit Authority in Minnesota [50], and the National Park Service [36]. Although these reports include only descriptive data analysis and do not include any statistical modeling to be used for further predictions, they provide some useful insights and baseline values.

In the Foothill Transit case, data collected from 2014 through 2020 identified average energy consumption of around 2 kWh/mi for a fleet of 35- and 40-foot Proterra buses. Energy consumption tended to be higher in summer months when temperatures were higher and more air conditioning load was required [49]. The results for nearby Long Beach Transit were similar, with their fleet of 10 BYD 40-foot buses averaging 1.94 kWh/mi over the course of 2019 [27]. The Duluth case covers much lower temperatures than those of the California agencies and shows more dramatic efficiency variations as a result. Average energy consumption of their 40-foot Proterra buses from 2019 through 2021 was 2.64 kWh/mi, but buses reported nearly double the energy consumption during the coldest months as during the warm summer months [50].

Other authors have combined real-world energy consumption data with statistical modeling and machine learning to quantify the impacts of key performance drivers like elevation gain and loss, weather conditions, and passenger load. Pamuła and Pamuła [91] used a deep

neural network to predict trip-level energy consumption for BEBs in Jaworzno, Poland. The variables used to predict energy consumption across each pair of stops in a trip included the distance to be traveled, travel time, elevation difference, and weather conditions. The neural network model was shown to slightly outperform a multiple linear regression model based on the same variables when making predictions on testing data. Abdelaty et al. [1] compared the performance of seven statistical and machine learning models in predicting energy consumption results from a simulation model. Among the models tested, multiple linear regression and support vector machine methods were deemed the most appropriate. The study also identified which variables were consistently identified as important for energy consumption prediction across all the different models and found that road gradient had the greatest impact. It should be noted, however, that many of the predictive variables used in Abdelaty et al. [1] study are not easy to reliably measure or predict in a real-world setting, such as HVAC load, driver aggressiveness, and drag coefficient.

Chen et al. [17] focused on high-resolution modeling of bus energy consumption, making predictions at 1 Hz frequency in order to support real-time control applications such as eco-driving. They trained long short-term memory (LSTM) and artificial neural network (ANN) models using real-world data from three 40-foot BYD buses operating in Chattanooga, Tennessee from 2019 to 2020. The authors reported mean absolute percentage error values of 6.3% and 4.2% for the ANN and LSTM models, respectively, on two weeks of testing data.

The LTI Bus Research and Testing Center at Penn State University, in partnership with the FTA, conducts performance testing on zero-emissions and conventional transit buses. This includes both models of BEB analyzed in this study: the New Flyer XE40 and XE60, with the most recent energy economy results documented in [108] and [107]. Each of these buses was tested on three different standardized drive cycles to simulate different driving conditions. All tests were performed with air conditioning off and with a weight corresponding to full seated passenger load with no standees [106]. The 40-foot XE40 reported energy consumption between 1.51 and 1.83 kWh/mi depending on the drive cycle, while the 60-foot XE60 reported energy consumption range from 2.39 to 4.46 kWh/mi.

It is important to note the differing vehicle and test characteristics between these two cases—the XE40 model was equipped with 466 kWh batteries and tested on a laboratory dynamometer, whereas the XE60 bus contained a 625 kWh battery and was tested on a road circuit because its weight exceeds the capacity of the dynamometer. As such, the results of these two tests are not directly comparable, but still provide a useful benchmark of the expected magnitude and variation of energy consumption rates.

5.3 Data Collection, Cleaning, and Processing

5.3.1 Data Sources

King County Metro ChargePoint Data

King County Metro granted access to their data dashboard used to track BEB operations and generate reports, provided by the electric vehicle infrastructure company ChargePoint. The ChargePoint interface allows for downloading high-resolution data on a wide variety of vehicle metrics; this work focuses simply on tracking the amount of energy consumed and distance driven by each bus. The ChargePoint software's Auto Exports feature was used to download all energy consumption and distance driven data for the month of March 2024 at a 10-second resolution.

King County Metro GTFS Realtime Data

Archived vehicle position and schedule adherence data for this chapter was provided by Dr. Zack Aemmer, who collected years of GTFS Realtime data for the King County Metro bus network using the OneBusAway API [3]. This archived data provided by Dr. Aemmer was essential for analyzing bus on-time performance as well as connecting the ViriCiti data to GTFS trip IDs and related data. As with the ChargePoint data, the data used for this analysis was restricted to March 2024. The data fields used in this chapter are documented in Table 5.1.

Attribute	Meaning
trip_id	Unique trip identifier
vehicle_id	Unique vehicle identifier
scheduleDeviation	Current estimated schedule deviation in seconds
locationtime	Timestamp

Table 5.1: Schema of archived GTFS Realtime data.

5.3.2 Data Cleaning and Processing

To obtain usable data for model fitting and simulation, it was necessary to clean and process the data from the Metro GTFS-RT feed and the ChargePoint data. There were two main challenges to handle:

1. The GTFS-RT data has some data quality problems, including inconsistent trip ID values and random outliers that need to be removed from the dataset. Section 5.3.2 describes how these problems were dealt with when cleaning and aggregating the data.

2. The ChargePoint data includes timestamps and vehicle IDs, but not GTFS trip IDs. To aggregate the ChargePoint data on a trip-by-trip basis and measure the impact of trip-specific features like route profiles, we needed to correctly assign a trip ID to each of these observations. Section 5.3.2 describes how the data were aggregated to the trip level for further analysis.

GTFS Realtime Data

Figure 5.1 shows an example of problematic raw data from GTFS-RT. This figure plots the schedule deviation for bus number 4712 over the course of its service on March 3, 2024. The y-axis values correspond to schedule deviation in seconds, while the color of each marker on the scatter plot corresponds to the trip ID reported in the GTFS-RT feed. Figure 5.1 highlights two recurring problems with this data source. The first is "run-on" trips, when the vehicles have actually completed a trip but continue reporting data for it alongside a

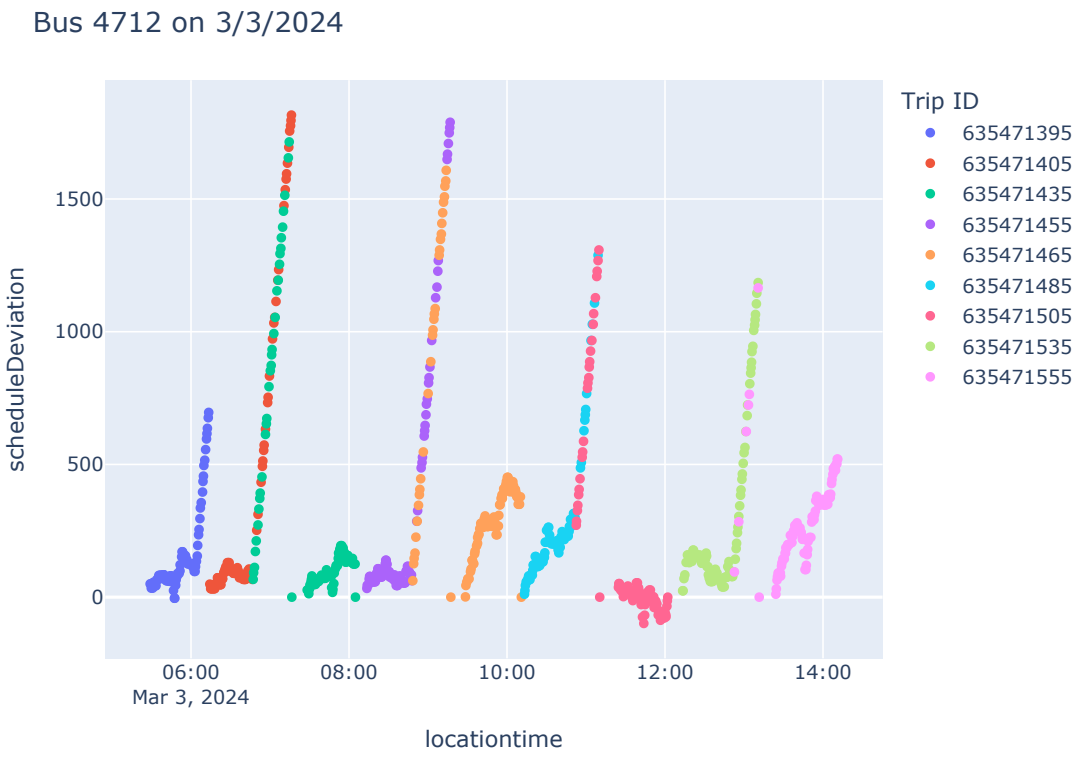


Figure 5.1: Realtime data before cleaning.

continually increasing `scheduleDeviation` value. There are five run-on trips in the figure. The second (and related) issue is inconsistent trip ID values, seen in most of the run-on trips in the dataset. Inconsistent trip IDs arise when the trip ID goes back and forth between different values over time, rather than consistently switching to another. We can see a clear example around 07:00 in Figure 5.1, when the trip ID value oscillates irregularly between 635471405 and 635471435. Both of these issues with the data make it difficult to identify the actual start and end times of each trip, which is critical for estimating trip durations and for connecting the ChargePoint data to GTFS trip IDs.

To reduce the impact of these data quality issues, we used the data cleaning procedure summarized in Figure 5.2. The first step is to detect and remove outliers for every trip in the dataset. This includes both sample time outliers and schedule deviation outliers, both

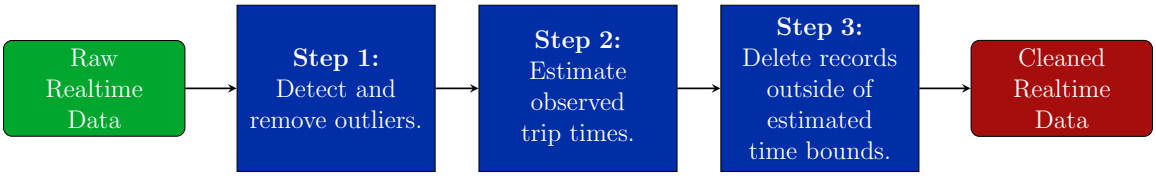


Figure 5.2: Overview of data cleaning process.

shown in Figure 5.3. Sample time outliers are observations recorded a long time before or after the majority of other observations for the same trip. Schedule deviation outliers are observations with an inconsistently large or small `scheduleDeviation` value compared to others for the same trip.

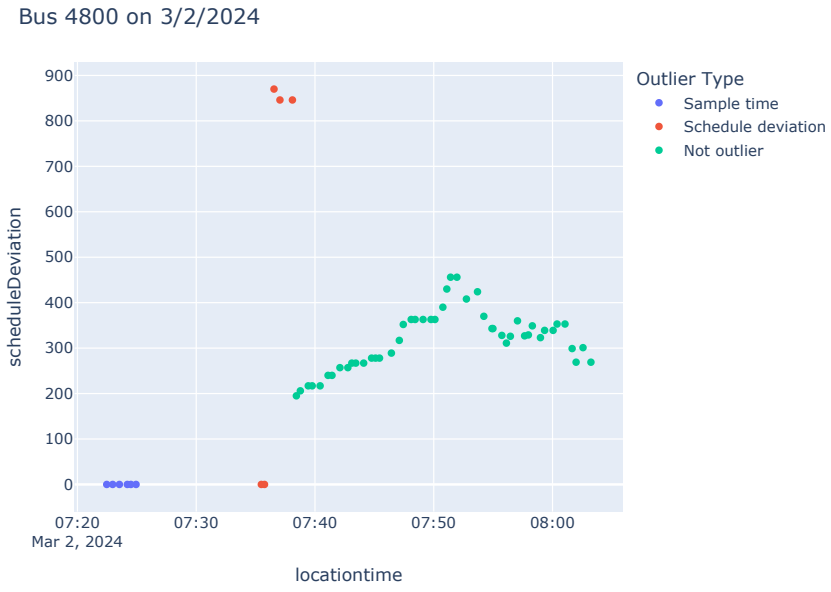


Figure 5.3: Example of outliers in realtime data.

Both types of outliers are identified and removed using the interquartile range (IQR) method. The IQR for a dataset is the range between the 25th and 75th percentile data points, or the first and third quartiles $Q1$ and $Q3$. That is, $IQR = Q3 - Q1$ In the IQR method, an

outlier is defined as any point with a value that is outside the range $[Q1-\alpha IQR, Q3+\alpha IQR]$, where α is an adjustment factor typically set to 1.5. If the data are normally distributed, then outliers will comprise about 1% of the data points when $\alpha = 1.5$. We use the typical $\alpha = 1.5$ factor for schedule deviation outliers, but decrease its value to $\alpha = 0.75$ for sample time outliers, since we expect the sample times to be spaced linearly throughout the trip duration (with a typical GTFS-RT frequency of about 2 samples per minute), not normally distributed. This method resulted in consistently good performance in identifying and removing outliers from the GTFS-RT data, as seen in Figure 5.3.

After outliers were removed from the data, Step 2 was to estimate the actual start and end times of all trips in the dataset. We estimated the start time based on the scheduled start time (identified from the static GTFS files) and the first "trusted" `scheduleDeviation` value. Likewise, the end time was estimated based on the scheduled end time and the last trusted `scheduleDeviation` value. The "trusted" points are the last observation corresponding to the first stop of a trip and the first observation corresponding to the last stop of a trip, identified based on the `nextStop` value. Using these trusted observations rather than the absolute first/last observation helps exclude the end of run-on trips as seen in Figure 5.1 and observations at the start of a trip before a bus has started moving.

To see how start and end times are inferred from the realtime data, suppose a particular trip is scheduled to start at 10:00 and end at 11:00. On a given day, its first trusted schedule deviation is -60 and the last trusted schedule deviation is 120. We set the actual start time to 9:59 and the actual end time to 11:02, for a total trip time of 63 minutes.

The third and final step of the data cleaning process is simply to exclude any data points outside of the trip start and end times estimated in Step 2. This ensures any suspicious observations associated with a given trip that weren't already excluded as outliers do not influence the final trip-level data. Figure 5.4 again plots bus 4712 on March 3rd, as seen in Figure 5.1, to show the results of the realtime data cleaning process. We see that run-on data at the end of trips has been removed and trip IDs are now consistent, as desired. Outliers have also been removed in terms of both timestamps and schedule deviation for each trip.

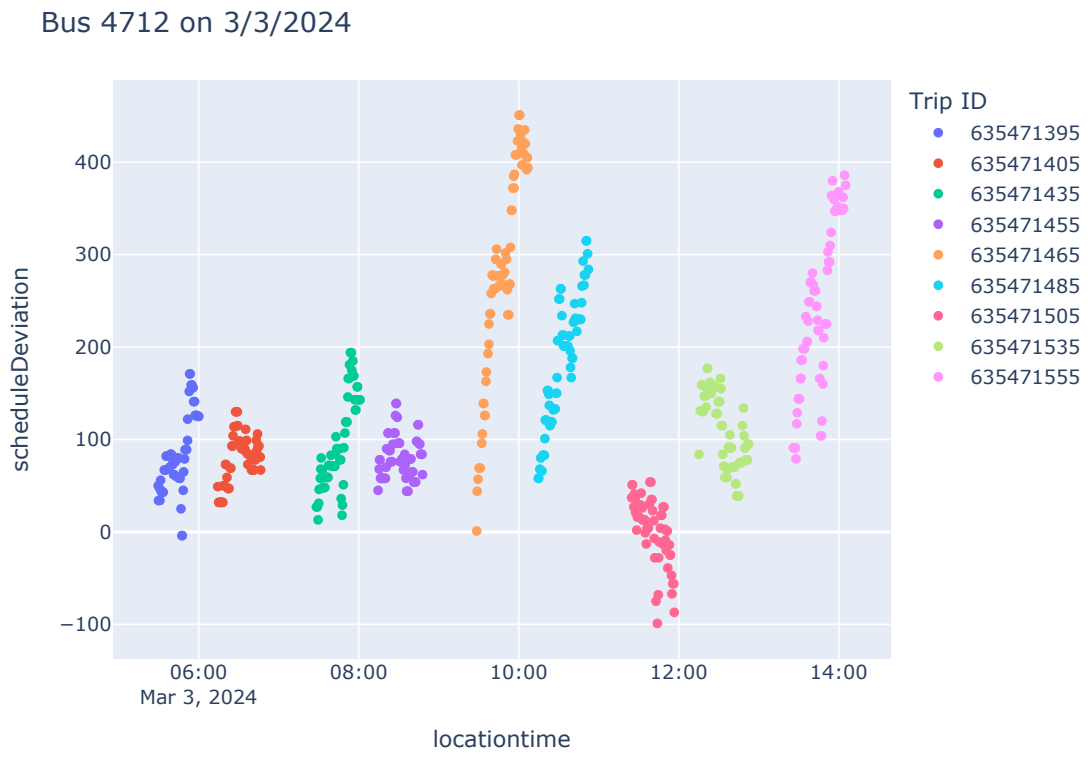


Figure 5.4: Realtime data after cleaning.

The final cleaned and processed dataset of all realtime data consists of 257,094 recorded trips made by 1191 distinct vehicles on 118 different routes. 3,129 of these trips are completed by BEBs recorded in the ChargePoint dataset. These BEB trips that provide both schedule deviation and energy consumption data are served by 30 vehicles and operate on 21 different routes.

ChargePoint Data

The cleaned and processed realtime data was used to aggregate the ChargePoint data at a trip level to determine average energy consumption in kWh/mi for all trips. Fortunately, the ChargePoint data did not have the same prevalence of issues as the GTFS-RT data, so a detailed cleaning procedure as in Section 5.3.2 was not needed.

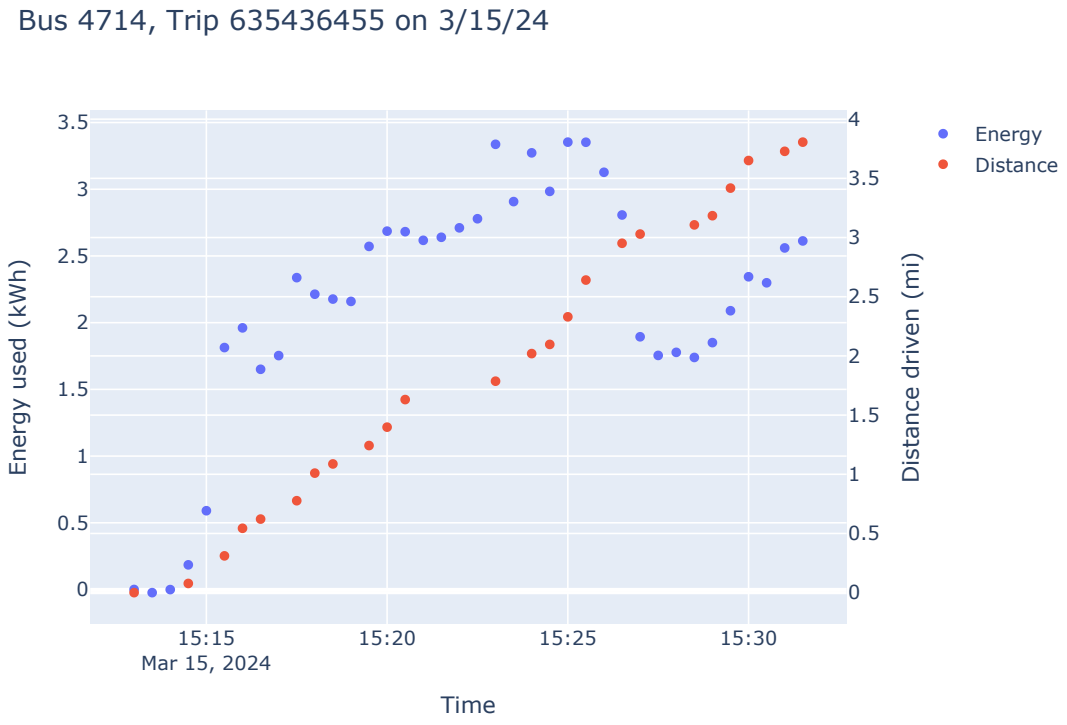


Figure 5.5: Example raw ChargePoint data for a single trip.

Figure 5.5 shows an example of the raw data for a single trip. The key metrics needed for each trip in our approach are the energy used and distance driven, which together yield the energy consumption rate in kWh/mi. We calculate these based on the difference between the first and last observation for which both energy and distance measurements are reported because sometimes one or both of these values may be missing. In the example trip shown in Figure 5.5, the total energy consumed is 2.6 kWh and the total distance driven is 3.8 miles, resulting in an energy consumption rate of 0.69 kWh/mi. Note that it is important to use the first and last points rather than minimum and maximum values, because energy consumption can decrease due to regenerative braking as seen in Figure 5.5.

5.4 Exploratory Data Analysis

This section documents the results of some exploratory data analysis intended to identify trends in the trip duration and energy consumption data.

5.4.1 Trip Duration

Figure 5.6 plots histograms of estimate actual trip duration (relative to scheduled duration) for the 20 routes served by BEBs for 3 trips or more (route 148, which was only served by BEBs for 2 trips, was excluded). Note that this data includes trips completed by all vehicles, not just BEBs, providing a large sample size for each route.

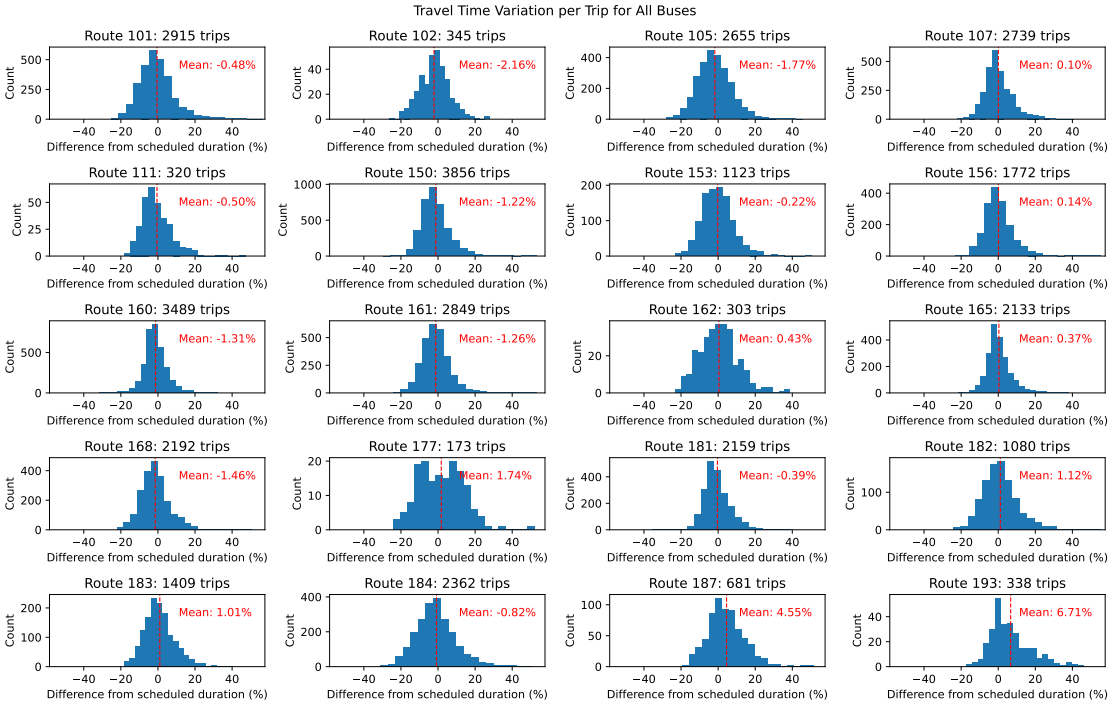


Figure 5.6: Histograms of trip duration for main routes served by BEBs.

Figure 5.6 shows that there is a significant variation in the trip duration distributions of different routes. Once the data is grouped by route, it tends to closely resemble a normal

distribution for most routes, though there are a few exceptions. In general, these exceptions are for routes that consistently experience major delays and have a distribution skewed to the right, such as Route 193, which frequently takes more than 20% greater than the scheduled time to complete its trips. Although the distributions are of similar shape, they show significant difference across the full spectrum of routes, with the mean time variation ranging from -11.38% for Route 989 (with a sample size of only 37 trips) to 10.76% for Route 11 (with a much larger sample size of 2,963 trips).

5.4.2 Energy Consumption

Figure 5.7 shows the distributions of energy consumption per trip broken down by route for each of the 20 routes traveled by BEBs included in the ChargePoint data, in the same format that Figure 5.6 showed trip duration trends. Figure 5.7 also splits these distributions according to the route direction, obtained from the GTFS `trips.txt` file. We can see that the energy consumption distributions vary considerably depending on the route and direction. For some routes such as 160, the distributions for the two different directions are not discernably different; for others such as Route 160, there is barely any overlap in the distributions for the two directions because there is a large amount of elevation change in the route. These results highlight the need to account for route- and direction-specific features when making energy consumption predictions.

Another key factor that determines energy consumption is the size and model of the bus. The data in the ChargePoint dataset are collected from a fleet that includes two different bus models: both 40-foot and 60-foot versions of the New Flyer Xcelsior battery-electric bus [59]. The mixed BEB fleet reflects the structure Metro's bus fleet as a whole; as of January 2023, the agency operated 60 35-foot buses, 581 40-foot buses, and 836 60-foot buses split across diesel, hybrid, battery-electric, and trolleybus models [57]. Among BEBs, the bus models can be identified based on the first two digits of their vehicle ID numbers in both the ChargePoint and GTFS Realtime data; buses that begin with *47* (e.g., *4701*) are 40-foot buses, while those that begin with *48* (e.g., *4802*) are 60-foot BEBs. The 60-foot buses, with

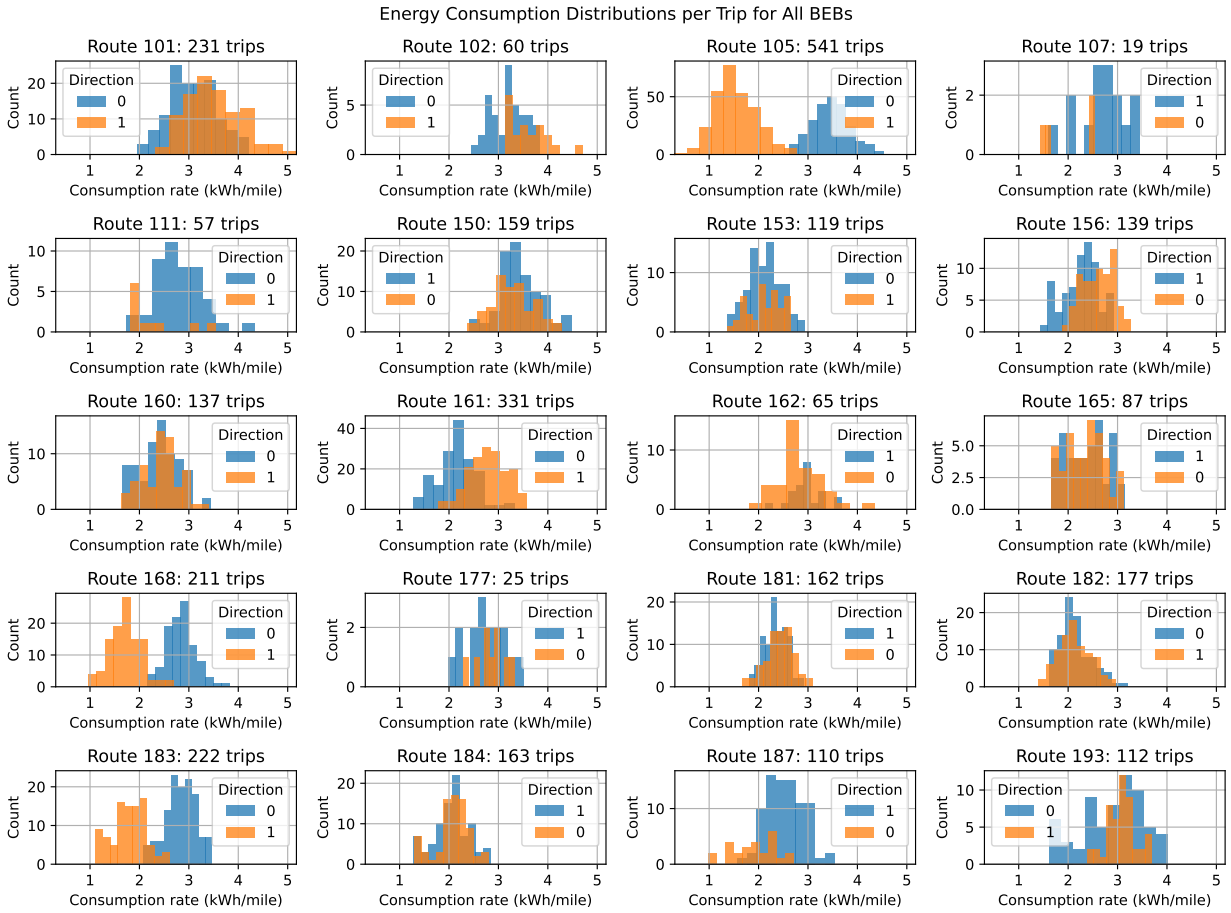


Figure 5.7: Histograms of energy consumption for all recorded BEB trips in March 2024.

their larger size and greater passenger capacity, can be expected to consume more energy. To show how energy consumption compares across the two vehicle types, Figure 5.8 shows overlaid histograms of average energy consumption for 40-foot and 60-foot buses, based on all trips in the dataset. Table 5.2 shares summary statistics for each of these distributions. We see that average energy consumption per trip varies widely across both bus types, but the 60-foot buses generally require more energy due to their greater mass and potentially increased HVAC loads.

To further investigate the elevation impact evident in Figure 5.7, elevation data was

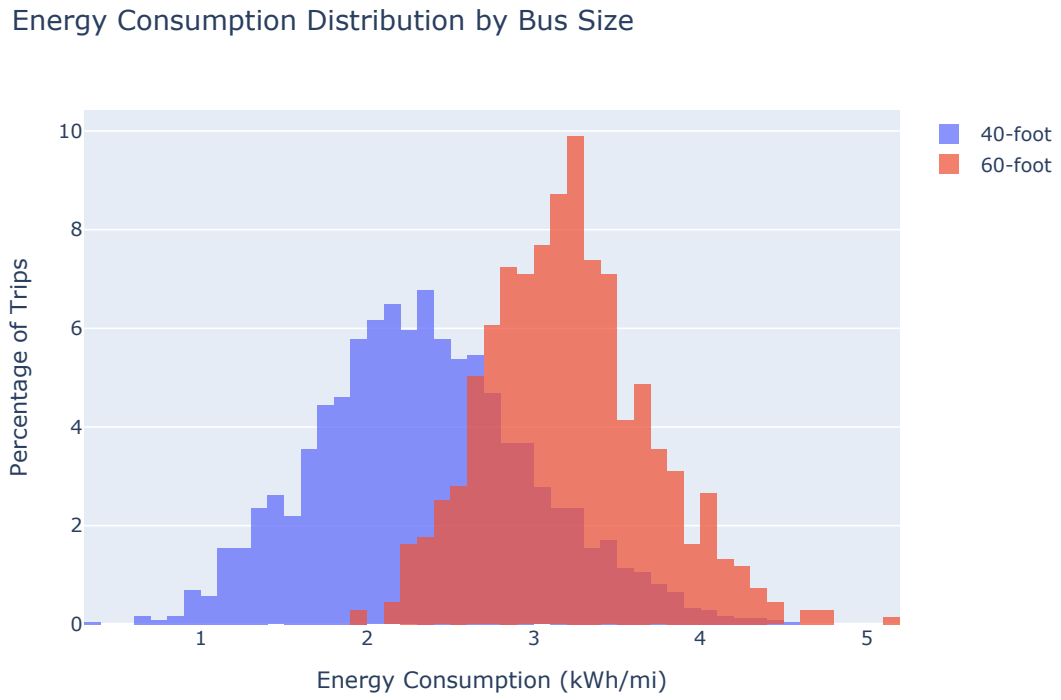


Figure 5.8: Histograms of energy consumption per trip for 40 and 60-foot bus models.

incorporated to calculate the total elevation gain and loss of each shape documented in Metro's static GTFS data, which includes full shape profiles for all trips as latitude/longitude coordinates, but no information on elevation. Elevation estimates for each shape were created based on digital elevation model (DEM) raster files of King County from the United States Geological Survey. The data used have a resolution of 1/3 arc-second, or approximately 10 meters. The `rasterio` Python package was used to sample the estimated elevation of all points contained in the GTFS shapes, then the elevation difference between successive points in every shape was used to determine total elevation gain and loss [114, 115].

Figures 5.9 and 5.10 document the impact that elevation change has on energy consumption. The x-axis of both figures is the average slope of the trip, defined as the net difference in elevation divided by the total trip distance calculated from the static GTFS `shapes.txt`

	40-foot buses	60-foot buses
Number of trips	2452	677
Minimum	0.31	1.96
25th percentile	1.91	2.86
50th percentile	2.31	3.19
75th percentile	2.75	3.49
Maximum	4.54	5.18

Table 5.2: Characteristics of energy consumption per trip by bus size.

file. In Figure 5.9, we can see that there is a clear linear trend of increasing energy consumption as trip slope increase. Despite this trend, there is still tremendous variability in energy consumption for trips with the same slope.

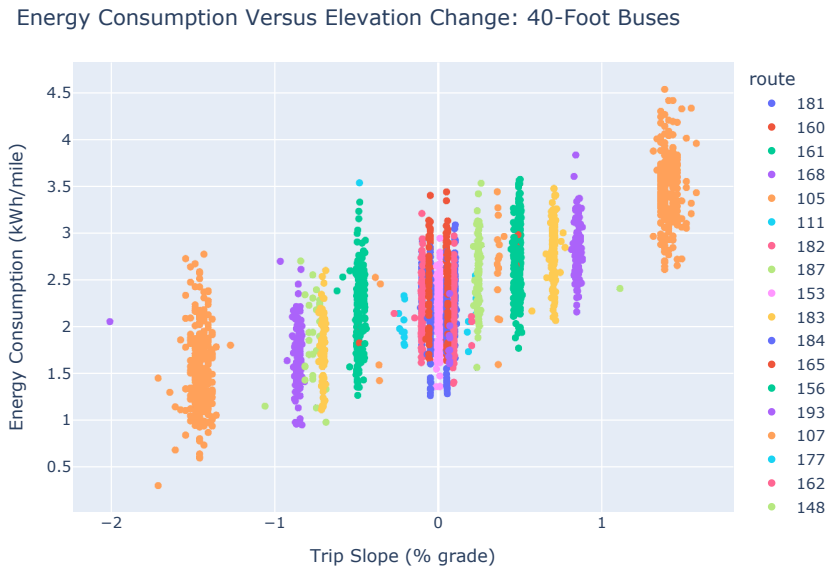


Figure 5.9: Impact of elevation of trip energy consumption for 40-foot buses.

The energy consumption trend is not as obvious for 60-foot buses in Figure 5.10 because these buses complete trips with a much narrower range of slopes. Nevertheless, we can see a slight upward trend in energy consumption as slope increases, consistent with physical

principles.

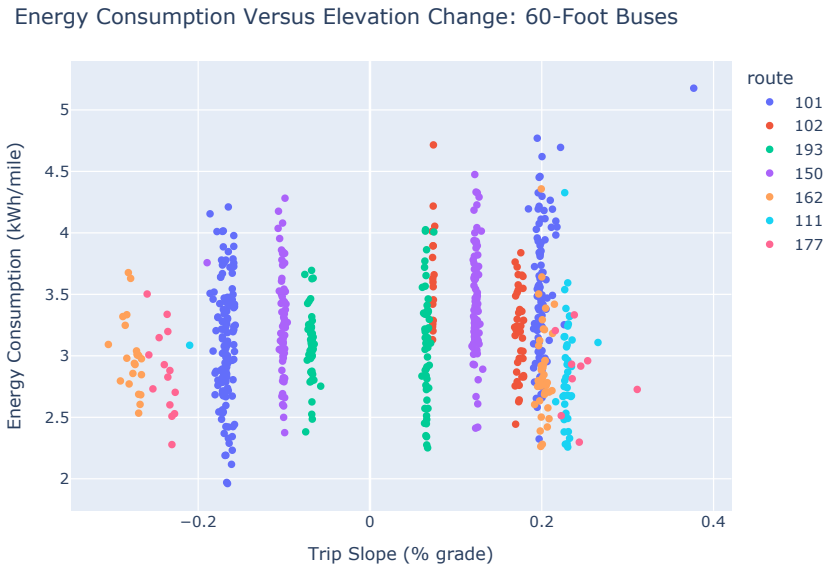


Figure 5.10: Impact of elevation of trip energy consumption for 60-foot buses.

Weather conditions are also recognized to have a significant impact on energy consumption because the batteries that supply the powertrain are typically used to support heating and cooling the vehicle interior as well. Energy consumption can generally be expected to increase under extreme temperatures, whether high or low. To see whether this expectation is realized in Metro's BEB data, Figure 5.11 plots average energy consumption versus outdoor temperature for both 40-foot and 60-foot buses. Average hourly temperature measurements are based on measurements at Seattle Boeing Field obtained from the National Centers for Environmental Information (NCEI, [86]). The temperature value reported on the x-axis corresponds to the temperature reported by NCEI at the time that the trip started.

Figure 5.11 shows a general downward trend in energy consumption as temperature increases for both bus sizes. Notably, both vehicle sizes show a slight increase in energy consumption towards the maximum temperatures observed (about 65 degrees Fahrenheit), though the exact trend is not clear because of the small sample size for these temperatures.

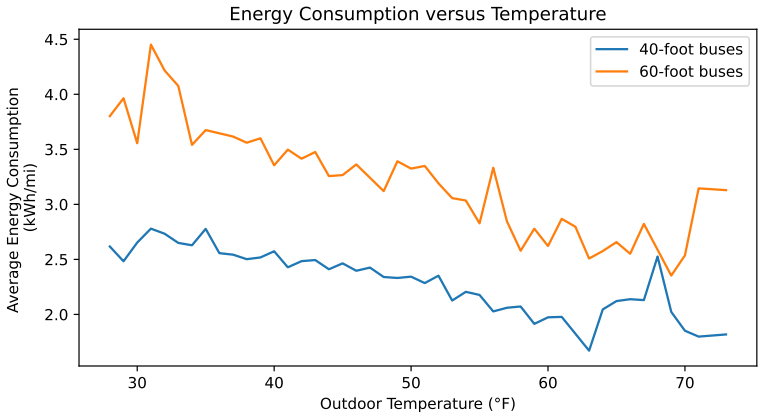


Figure 5.11: Impact of outdoor temperature on average energy consumption.

Only 55 trips made by 40-foot buses and 28 trips made by 60-foot buses occurred when the temperature was 65 degrees or greater.

Vehicle speed can also impact energy consumption. Low speeds may reflect heavy traffic conditions with lots of acceleration and deceleration, increasing energy consumption (although BEBs are equipped with regenerative braking, it is not 100% efficient). On the other hand, traveling closer to highway speeds (as some suburban routes do) may incur more energy usage due to greater aerodynamic drag. Figure 5.12 shows the impact of speed on average energy consumption. Here, the reported speed is based on the scheduled trip duration and distance as reported in GTFS (rounded to the nearest integer), not the estimated realtime duration, to make it more directly applicable for predicting energy consumption. Only speeds that applied to 20 or more trips in the dataset are shown in this figure. We see that the 40-foot buses tend to have significantly higher energy consumption on the slowest trips (14 mph or below) and the effect is less pronounced after that. For 60-foot buses, there is a less obvious pattern, but still a slight downward trend in energy use as speed increases.

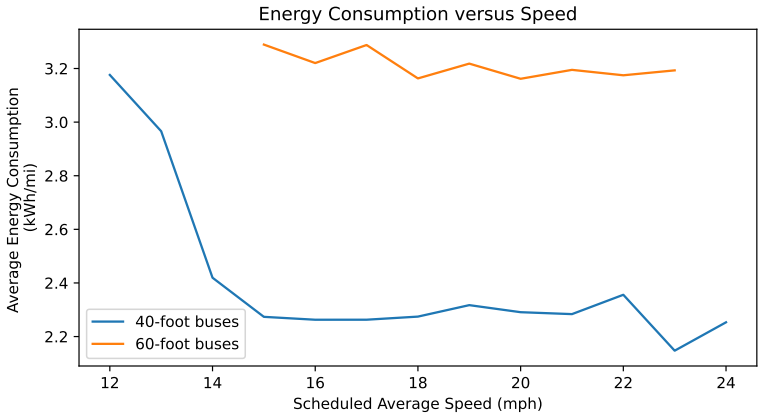


Figure 5.12: Impact of scheduled speed on energy consumption for 40- and 60-foot buses.

5.5 Statistical Modeling

Following the exploratory analysis of Section 5.4, the next step was to translate the identified trends in trip durations and energy consumption into predictive models that could be used for simulation. Section 5.5.1 describes the approach taken to simulating trip durations based on empirical distributions, while Section 5.5.2 describes a linear regression model used to predict and simulate energy consumption for BEB trips in King County.

5.5.1 Trip Durations

Trip durations are difficult to predict in advance, being mainly a product of passenger load (and the resulting boarding/alighting times), traffic conditions, and unpredictable events like vehicle breakdowns. Attempts to predict schedule adherence based on any obtainable data did not result in good performance. Instead, when modeling trip durations, we use the empirical distributions gathered for each route in the Metro network, as seen in Figure 5.6. We can use these distributions to extract specific values for use as optimization parameters, such as the mean or a particular quantile value for a given route's distribution.

5.5.2 Energy Consumption

We used linear regression to predict energy consumption, consistent with Abdelaty et al. [1]. To evaluate model performance on trips that were not represented in the data used to fit the model, we split the data into training and testing datasets. We used all data collected from March 1st through 27th as training data, with the remaining 4 days from March 28th-31st serving as testing data.

Approach

The linear regression model developed here was designed to predict energy consumption in kWh/mi per trip. The reason for selecting kWh/mi rather than kWh as the dependent variable is to better satisfy the homoscedasticity assumption of linear regression, which states that the variance of the residuals is the same regardless of the dependent variable value. In our case, we can expect that trips with large absolute energy consumption (e.g., 100 kWh) will have greater variance than those with small energy consumption (e.g., 10 kWh), so the assumption is better satisfied by the smaller range of kWh/mi values.

A variety of model compositions were tested by iteratively adding variables and evaluating performance. We tested a variety of variable encodings related to vehicle size, outdoor temperature, elevation gain and loss, and speed. The variable selections were informed by physical principles, data availability (especially in a predictive setting, i.e., on routes that haven't yet been served by BEBs), and the relevant literature as discussed in Section 5.2. Table 5.3 documents the predictive variables that were tested during the model selection process.

Model selection was carried out by testing various combinations of the variables in Table 5.3. In general, variables that were determined to be statistically significant at a 95% confidence level (i.e., $(P > |t|) \leq 0.05$) were retained and others were dropped. There were some exceptions to this rule; for example, if the calculated model effect was counterintuitive and it did not improve performance on the testing data, the variable was removed. This prin-

Table 5.3: Variables tested in regression model fitting process.

Parameter	Units	Notes
Elevation		
Elevation gain	ft./mi.	Obtained from [114, 115]
Elevation loss	ft./mi.	Obtained from [114, 115]
Trip average slope	%	Obtained from [114, 115]
Route Characteristics		
Stop density	stops/mi.	Calculated from `stop_times.txt`
Scheduled speed	mi./hr.	Calculated from `shapes.txt` and `stop_times.txt`
Realtime speed	mi./hr.	Calculated from realtime data and `shapes.txt`
Low speed flag	N/A	1 if scheduled speed is below 15 mph, 0 otherwise
Weather		
Temperature	°F	Obtained from NCEI [86]
Heating Degrees	°F	$\max\{0, 65 - \text{temperature}\}$
Cooling Degrees	°F	$\max\{0, \text{temperature} - 65\}$
Heating Degree Hours	°F * hours	Trip duration times heating degrees
Cooling Degree Hours	°F * hours	Trip duration times cooling degrees
Vehicle Characteristics		
60-Foot Bus	N/A	Binary dummy variable
Trip Time		
Peak/off-peak flag	N/A	Binary dummy variable
Weekday/weekend flag	N/A	Binary dummy variable

ciple was applied when testing peak period impacts—off-peak trips were consistently found to have lower energy consumption than peak-hour trips, the opposite of what was expected due to the greater passenger load and traffic during peak times. Because adding dummy variables to indicate peak status had this counterintuitive effect and the model performed better on the testing data without this variable, it was ultimately excluded.

In some cases, different variable encodings resulted in similar significant effects. In these cases, model performance was evaluated based on the reported Akaike Information Criterion (AIC), which captures the tradeoff between the number of model parameters and goodness of fit.

Results

The model was fitted using the `statsmodels` Python package. Table 5.4 summarizes the results of the Ordinary Least Squares procedure. The final model obtained a coefficient of variation (R^2) value of 0.77, indicating that the model explains 77% of the variance in the observed energy consumption data.

Table 5.4: Linear regression results.

	Coefficient	**std. err.**	**t**	**P> |t|**	**[0.025**	**0.975]**
Intercept	2.1384	0.057	37.789	0.000	2.027	2.249
60-Foot Dummy	0.6948	0.039	17.904	0.000	0.619	0.771
Heating Degrees (40')	0.0248	0.001	24.865	0.000	0.023	0.027
Heating Degrees (60')	0.0375	0.002	21.395	0.000	0.034	0.041
Ft. Gain per Mi.	0.0466	0.001	56.874	0.000	0.045	0.048
Ft. Loss per Mi.	-0.0351	0.001	-42.543	0.000	-0.037	-0.033
Scheduled MPH	-0.0232	0.003	-7.889	0.000	-0.029	-0.017

The final model achieves good predictive performance and is consistent with reasonable expectations of variable impacts. The best model performance was achieved when temperature impacts were expressed as heating degrees and a separate coefficient was used for 40-foot and 60-foot buses. Since more energy is required to heat the larger bus, it is not surprising to see that the coefficient for 60-foot buses is larger. Likewise, the dummy variable indicating that a trip is served by a 60-foot bus has a large positive coefficient, indicating that all else equal, a 60-foot bus typically uses about 0.7 kWh/mi more in service than a 40-foot bus.

Elevation impacts are also consistent with the physical context of our data. We see that elevation gain increases predicted energy consumption and elevation loss decreases it; the elevation gain coefficient also has a greater magnitude, so that hilly routes with no net elevation gain consume more energy than flat routes. Finally, trips consume less energy as scheduled average speed increases, consistent with the trend seen in Figure 5.12.

Table 5.5 summarize the model's accuracy on the training and testing datasets in terms

Table 5.5: Summary of linear regression model performance.

Dataset	Number of Trips	MAPE	RMSE
Training (March 1-27)	2702	11.69%	0.3423 kWh/mi (0.2699 kWh/km)
Testing (March 28-31	427	13.22%	0.3738 kWh/mi (0.2947 kWh/km)

of the the mean absolute percentage error (MAPE) and root mean squared error (RMSE). These values are reported separately for the training and testing dataset. The fitted model attained a MAPE value of 11.69% on the training dataset and 13.22% on the testing data. The RMSE values also compare favorably against the published literature; Abdelaty et al. [1] reported an R^2 value of 0.961 and an RMSE of 0.3163 kWh/km on training data for their multiple linear regression model. It should be noted that the simulation data in Abdelaty et al. [1] covers a broader range of energy consumption values than our observed data, so the RMSE results are not directly comparable.

Finally, Figure 5.13 plots the observed versus predicted values of energy consumption for both the training and testing datasets. On the training data, the distribution of errors appears to be consistent regardless of the predicted energy consumption, suggesting that errors are normally distributed with consistent variance as desired. The testing data results show no significant bias—the predicted values indicated by the green line fall evenly in the center of the actual values shown by the blue markers. There are, however, a few outlier trips with much greater energy consumption than predicted.

5.6 Testing Performance of Charging Scheduling under Uncertainty

The charging scheduling model described in Chapter 4 and accompanying 3S heuristic provide an approach to quickly generate good BEB recharging plans for a single day. However, the model is fully deterministic, treating the amount of energy consumed on each trip and the duration of each trip as fixed values that are known with certainty. Sections 5.4–5.6 showed that while energy consumption can be predicted with moderate accuracy, both energy and on-time performance have significant random components. This conflict presents two questions

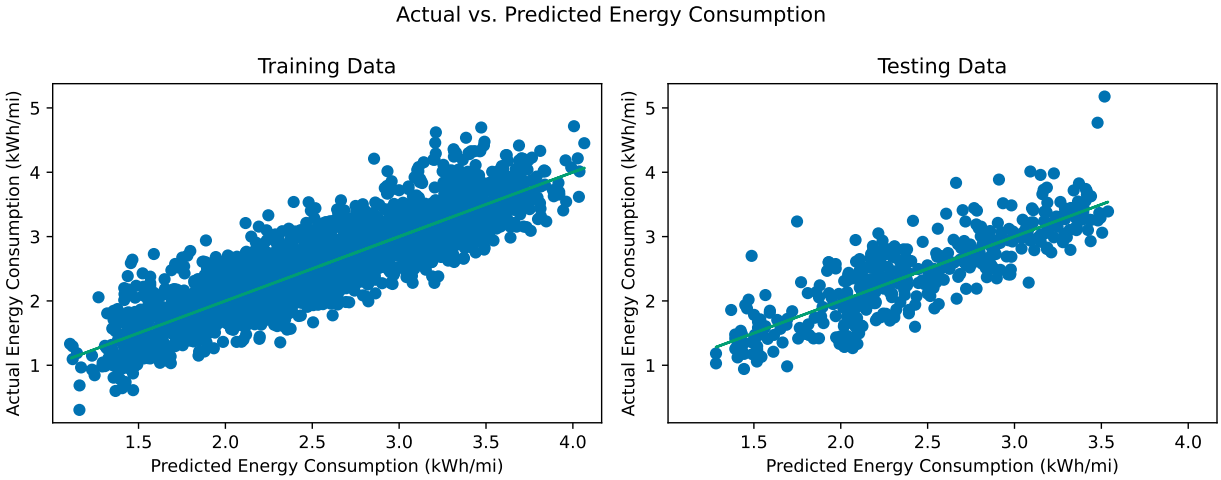


Figure 5.13: Actual versus predicted energy consumption for training and testing data.

critical to the real-world application of this work:

1. How should we set the trip duration and energy consumption parameters in the charge scheduling model of Chapter 4? Is it best to use average predicted values, or to use values that are either more optimistic or more conservative?

2. How does a recharging schedule based on these parameters perform when tested on real-world data?

This section aims to answer both questions by embedding the statistical models of trip duration and energy consumption within a discrete-event simulation platform that captures all daily activities of each BEB, including passenger trips, deadheading, and charging. Section 5.6.1 first describes how the simulation is performed, including the relevant assumptions. Section 5.6.2 defines the BEB system the model is tested on. Section 5.6.3 tests different strategies for setting optimization parameters and analyzes the simulated results to answer Question 1 above, while Section 5.6.4 presents model performance on the four days of testing data at the end of March 2024.

5.6.1 Simulation Platform

System performance under variations in schedule adherence in energy consumption is evaluated with a customized discrete-event simulation model originally proposed in McCabe [78]. For this application, the simulation model was extended to evaluate the results of both the charger location model of Chapter 3 and charging scheduling model of Chapter 4, and to generate random values of energy consumption and trip duration based on Section 5.5.

Figure 5.14 shows an overview of the data processing required to create a simulation run. The starting point is the complete set of trips and necessary data to be served by BEBs on a single day, as output by GTFS-BEB (Section 2.2). Because GTFS does not include buses' pull-out trips from the depot at the beginning of the day or pull-in trips back to the depot at the end of the day, a specified depot location is used to add estimated deadhead distances to the first and last trips in each block, as calculated with Openrouteservice [89]. Simulated durations are then added to each trip by sampling from the empirical distribution of the route served by each trip as described in Section 5.5. Adding durations, vehicle characteristics, and weather features to each trip allows us to make energy predictions using the linear regression model of Section 5.5. The final simulation inputs include charger characteristics (locations and power outputs) as well as a planned charging schedule as output by one of the optimization models.

The simulation logic is built on the following assumptions:

1. Buses will never start a trip early. If a bus arrives at the initial stop of a trip before the scheduled start time, it will wait until the scheduled start time to begin the trip.

2. The energy consumption rate for any deadhead trip is the same as that for the trip before it. Pull-out trips (the first trip of the day) consume energy at the same rate as simulated for the first passenger trip.

3. Buses are scheduled to operate between 15% and 95% state of charge. They will never charge beyond 95% state of charge and will only fall below 15% of charge if insufficient

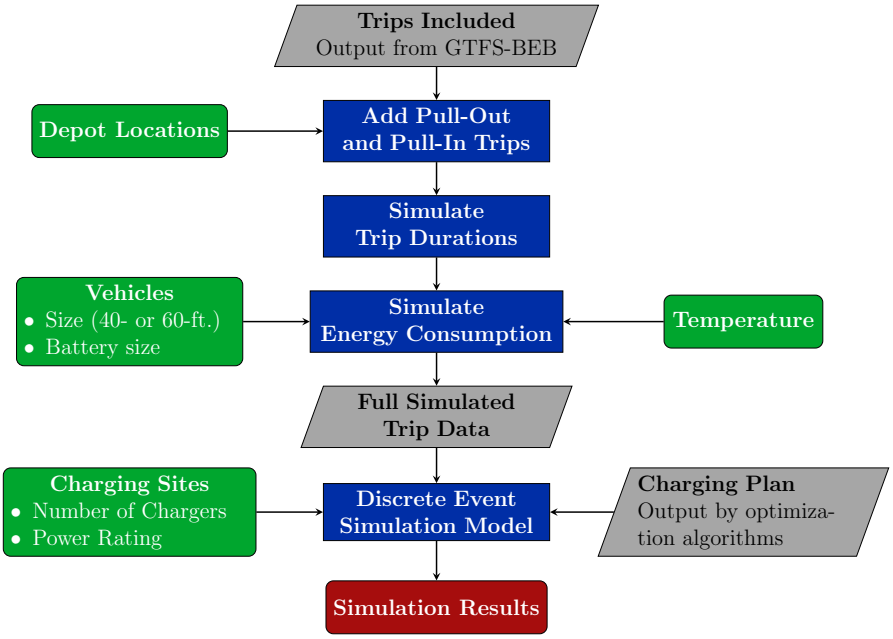


Figure 5.14: Data processing for discrete event simulation model.

charging was planned. If the battery level falls below 5% state of charge, the bus is removed from service.

The logic of the discrete event simulation model is shown in greater detail in Figure 5.15, which reflects the journey of a single bus. Each vehicle enters service by completing a pull-out deadhead trip to the start location of the first trip in its block, then follows the same process through all trips in its block. Each time a trip starts, the simulation records the delay (if any) at the start of the trip. Each time a trip ends, we first check if the bus has run out of battery (i.e., has less than 5% state of charge). If so, the bus is taken out of service and cannot complete its remaining trips. Otherwise, as long as the bus has more trips to complete, we check the charging plan to see if the bus is scheduled to use a charger. If charging is planned—or if the battery state of charge is below 15%, indicating that charging is necessary—the bus deadheads to the charger. If no charging is planned or needed, the bus deadheads to the start of its next trip.

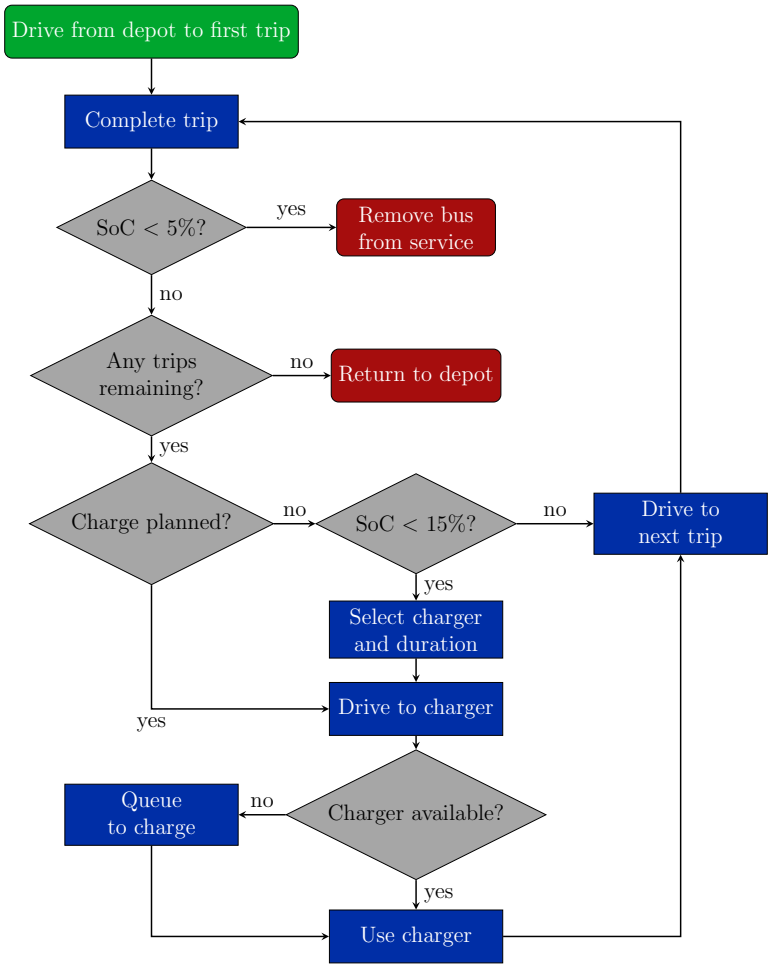


Figure 5.15: Overview of discrete event simulation process for a single block.

Each time a bus visits a charging site, we check if it is fully occupied by other buses that are already charging. If so, the bus is added to the end of a first-in, first-out queue to use the charger and waits until it becomes available. If not, it can begin charging immediately. In either case, the bus will start charging at the appropriate time, plug in for the scheduled amount of time, and then drive to the start of its next trip, restarting the loop. Note that while Figure 5.15 only depicts a single bus, the simulation simultaneously tracks the progress of all vehicles to ensure their interactions at charging sites are properly captured in queues.

When the simulation has terminated, the results are processed to calculate key metrics.

First, a delay penalty is applied to any trips that could not be completed because the bus intended to serve them had insufficient charge and was removed from service. Each trip that could not be completed is assessed a delay penalty of 60 minutes, or the maximum recorded delay among all trips if that exceeds 60 minutes. The total delay is then calculated as the sum of all individual trip delays, including these penalties.

The key metric we report for each simulation run is not the total delay but the "charging-induced delay," equal to the total delay of all trips served by BEBs minus the total amount of exogenous delay. The exogenous delay is defined as the amount of unavoidable delay due to schedule deviations. For example, suppose two consecutive trips are scheduled for 10:00 and 11:00, with five minutes of deadhead time required to drive between them. If the first trip takes 65 minutes to complete (ending at 11:05), the next trip cannot start until at least 11:10, so the exogenous delay is ten minutes. Exogenous delay is calculated by running the discrete event simulation with the energy consumption set to zero for all trips, so that charging is never necessary, and recording the total delay that results.

5.6.2 Test Network

The case study instance used for testing is based on Metro's planned BEB deployments in South King County. Opportunity charging sites are based on those reported on Metro's Zero Emissions Fleet website [60] and documented in Table 5.6. Based on Metro's current vehicles, both 40-foot and 60-foot have 525 kWh batteries, which are assumed to be operated between 15% and 95% state of charge. All chargers are assumed to have a power output of 450 kW.

BEBs in the case study serve 20 bus routes with terminals close to those sites, listed in Table 5.7. All buses are assumed to start and end the day at Metro's South Base; pull-out and pull-in deadhead trips were added accordingly based on driving distances from the Openrouteservice API [89]. To set the upper bound on charging time $\overline{t_i^l}$ for all trips and chargers, we identified the final stop coordinates of each trip and calculated the driving distance to each charger (again using Openrouteservice) and assumed buses were able to use

a charger if the distance to it was less than 1.5 miles (2.4 km).

Location Name	Number of Chargers	Latitude	Longitude
Burien Transit Center	1	47.4693256	-122.3403857
Kent Des Moines Link Station	1	47.389472	-122.294063
Kent Transit Center	2	47.3836401	-122.2346441
South Renton Park & Ride	1	47.4718414	-122.2147887
Federal Way Transit Center	1	47.3179917	-122.3056636

Table 5.6: Candidate charging sites used for testing.

Bus Size	Routes Served
40-foot	105, 106, 107, 153, 156, 160, 161, 165, 168, 182, 183, 187, 240
60-foot	101, 102, 131, 132, 150, 162, 177, 193

Table 5.7: Routes included in final case study and bus models that serve them.

The number of chargers at each site, as documented in Table 5.6, was set by with BEB-OCL. The energy parameters for BEB-OCL were set based on average values in the training data for each bus size: 2.31 kWh/mi for 40-foot buses and 3.19 kWh/mi for 60-foot buses. Block and schedule data were based on March 28th, lower and upper battery limits were based on 15% and 95% state of charge for a 525 kWh battery, and the remaining model parameters were identical to those used in Section 3.4.

Figure 5.16 plots the charger locations along with all BEB trips included for the first testing date of March 28th. The geographic distribution of trips is similar for the other three testing dates. The blue markers on the map show trip terminals where buses are located when they are available to visit an opportunity charger, while the red lines trace trip shapes.

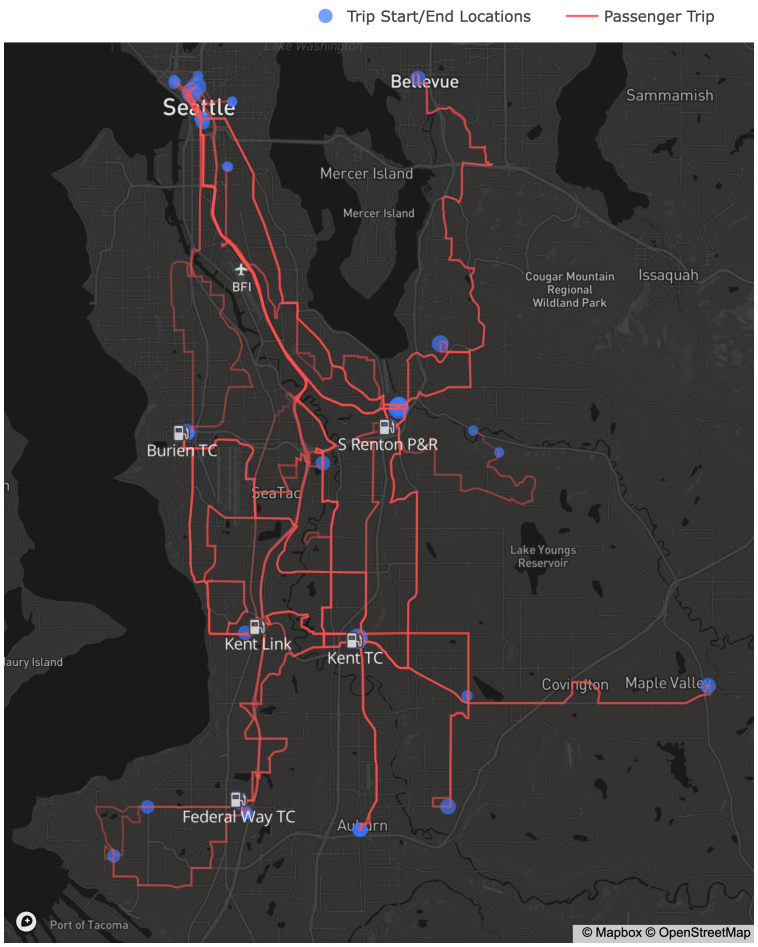


Figure 5.16: Map showing trips, terminals, and charger locations for the King County case.

5.6.3 Sensitivity Analysis on Scheduling Optimization Parameters

Methodology

This section returns to the first question proposed in Section 5.6: How should we set the trip duration and energy consumption parameters in the charge scheduling model of Chapter 4 to achieve good performance when faced with real-world uncertainty? Here, we test various strategies on a large number of simulated scenarios to evaluate performance with different strategies for setting the parameters δ (energy consumption of each trip) and τ (duration of

each trip) input to the charge scheduling model.

We should expect that choosing different parameter values will impact the charging schedule output by the 3S algorithm. If we use relatively low estimates of energy consumption, for example, the resulting charging schedule will include less total charging time than if we use higher estimates across all trips. Low estimates of energy consumption could result in insufficient charging that forces buses to make unplanned charges that are likely to generate delays. On the other hand, using large energy consumption estimates can produce a more robust solution in which buses consistently plan for enough charging, but taking such a conservative approach could generate delays as buses will spend more time at chargers. This section therefore presents an analysis of different strategies for setting these two parameters to provide recommendations about implementation in practice.

The different parameter setting methods are all based on selecting each parameter value based on its estimated distribution. Trip duration distributions are based on the empirical distribution of all trips completed on the same route in the training period from March 1-27; the distribution for trip energy consumption is based on the estimated delay from the linear regression model of Section 5.5.2. The predicted energy consumption for a given trip can be modeled as a normal distribution where the predicted value is the mean and the standard deviation of the hours from the regression model is the standard deviation.

In both cases, our modeling approach yields a statistical distribution for each model parameter, which is also used to generate simulation data. To set input parameters for the scheduling optimization, we sample specific quantiles from these distributions. For example, a quantile sampling value of 0.5 means that for a particular parameter, its value in the scheduling optimization is set to its 50th percentile (or median value of the distribution). Low quantile values can be seen as optimistic estimates and large values approaching 1 can be seen as conservative estimates reflecting a worst-case scenario in which all trips are heavily delayed and consume much more energy than expected.

Different quantile sampling values were tested by solving the charge scheduling model for the test network with all values considered and then simulating the resulting performance.

The scenarios tested are based on simulated trip duration and energy consumption values for all 30 days in April 2024. For each day, temperature data from NCEI [86] and GTFS schedule data were used to create problem instances and predictions of energy consumption and trip duration for every trip. The 3S heuristic algorithm (run for 30 iterations each time) was used to generate solutions to the charge scheduling model based on the calculated input parameters for each quantile, then the quality of these solutions was tested for 10 simulated scenarios on each day. The simulation data for each test instance was the same regardless of the quantile value, to ensure that the results documented for each strategy were directly comparable. The scheduling results based on each quantile value were therefore each tested on 300 identical scenarios. We test quantile values from 0.3 through 0.9 in increments of 0.1 for both energy consumption and trip duration.

Results

Results on the 300 simulated scenarios for each energy consumption sampling strategy are summarized as a box plot in Figure 5.17, which displays the full distribution of charging-induced delay in each case. Duration parameters were based on the 50th percentile value in all cases. The average delay (as indicated by the orange horizontal line within each box) is consistently low regardless of the quantile value, but the smaller quantiles yield less robust solutions: the maximum delay realized across all scenarios is much greater when running the optimization with 30th or 40th percentile estimates than for all other cases. The 50th percentile strategy yields the lowest average delay by a small margin, but larger quantile values produce more robust solutions that result in fewer high-delay scenarios. Larger quantile values start to have greater mean delays without a clear benefit to robustness.

Figure 5.18 highlights one factor determining these trends by plotting the number of unscheduled charges per trip, i.e., the number of times that a bus's state of charge drops below 15% and it must visit a charger. These unplanned charges are likely to introduce delays. We can see that, as expected, using a larger quantile value results in fewer unplanned charges because more charging is planned more based on a higher energy consumption estimate for

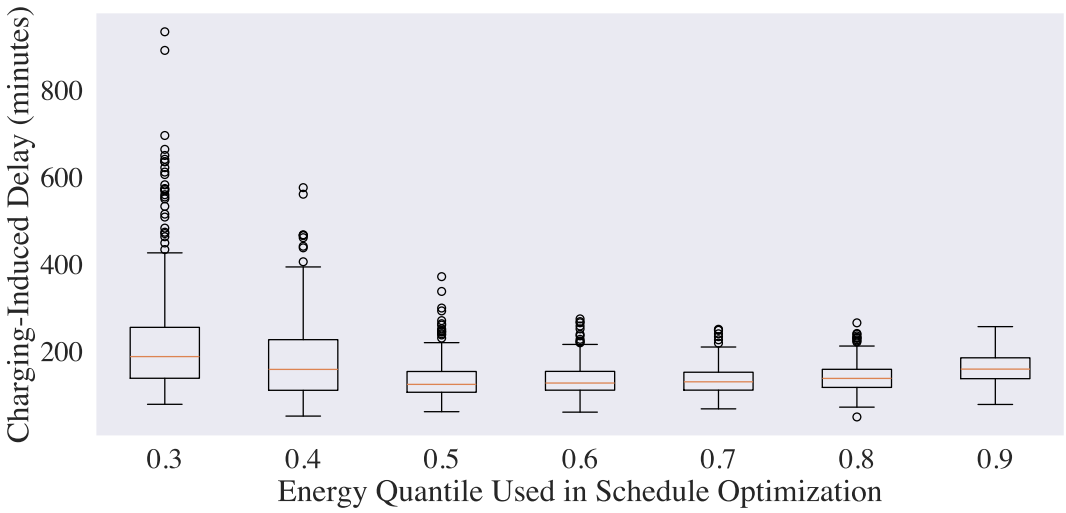


Figure 5.17: Impact of energy quantile used to set scheduling parameter on simulated delays with 450 kW chargers.

every trip.

Because all tested scenarios and parameter settings in Figures 5.17 and 5.18 result in relatively good performance, the same experiment was repeated with a lower charger power output of 150 kW in order to produce more challenging conditions where chargers are heavily utilized and delays are difficult to avoid. The results are summarized in Figure 5.19.

Figure 5.19 shows a slightly different trend than Figure 5.17, with system performance degrading noticeably for the largest quantile values. We see that when the most conservative predictions (i.e., 90th percentile) are used, performance is consistently worse than for moderate strategies like the 60th or 70th percentile. The trend shows that when chargers are heavily utilized and queuing is a major concern, it can be harmful to plan based on nearly worst-case predictions, because buses devote excessive time to charging, and instead a moderately conservative approach is best, with the 60th percentile showing the best overall performance by maintaining a low mean delay and minimizing the worst-case delay across the scenarios considered.

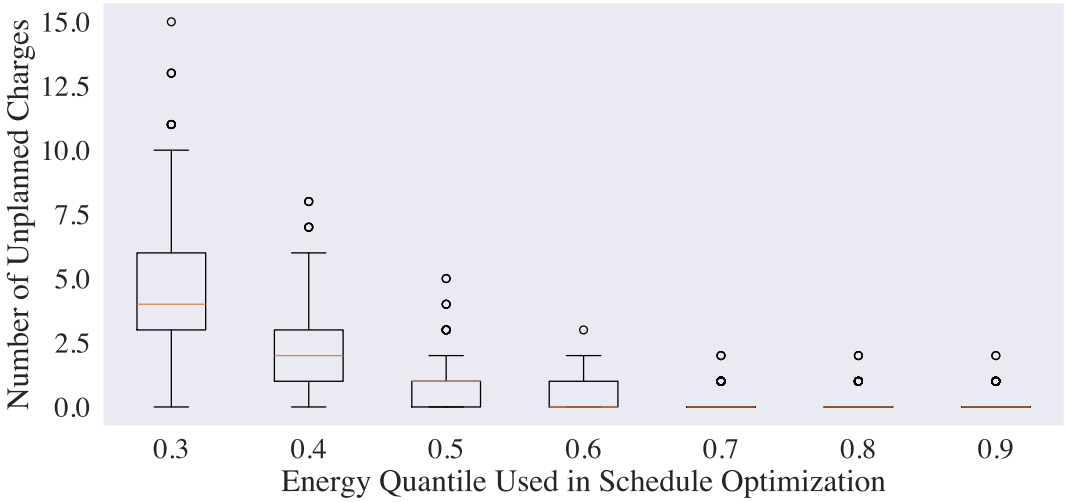


Figure 5.18: Impact of energy quantile used to set scheduling parameter on number of unplanned charges with 450 kW chargers.

These trends in performance highlight a key tradeoff when setting energy parameters for opportunity charging scheduling: using conservative estimates of energy consumption makes it very unlikely that buses will need to do any unplanned charging, but this also comes at a cost. When energy consumption does not consistently exceed what is predicted—which is usually the case—buses end up charging significantly more than is needed, which can introduce unnecessary delays.

Simulations were also run to test different strategies for setting trip duration parameters. Energy consumption parameters were based on the 50th percentile value in all cases. Figure 5.20 shows a box plot of the total charging-induced delay across all simulations versus the duration quantile value used for charge scheduling. These results show less variation in simulated performance when compared to the different energy quantiles tested. The distributions of charging-induced delay look nearly identical for all percentile sample strategies, though performance generally improves a bit as the quantile value increases. A likely explanation is that using conservative estimates of trip duration encourages 3S to only schedule charging

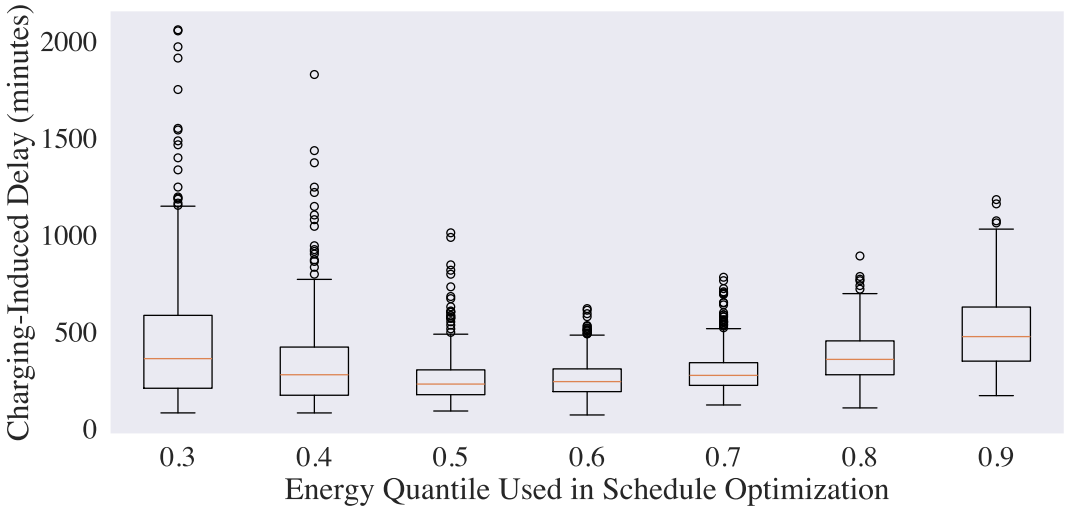


Figure 5.19: Impact of energy quantile used to set scheduling parameter on number of unplanned charges with 150 kW chargers.

when a large amount of layover time is available for charging. When smaller quantiles are used, charges are more likely to be scheduled during short layover periods, so that there is not as much time between trips for buses to recover from delays.

5.6.4 Performance on Testing Dataset

Section 5.6.3 explored the performance of different approaches to setting optimization parameters on performance across many simulated scenarios. This section assesses the performance on real-world data from the March 28-31 testing dataset, using the actual observed durations of all trips wherever possible and simulating trip durations to replace missing data when necessary. Based on the results of the quantile sampling strategies tested in Section 5.6.3, the charging schedules for the testing examples were generated based on 60th percentile energy consumption predictions and 90th percentile duration predictions for each trip.

The results for each day are summarized in Table 5.8, which records the number of trips and blocks in each instance, the scheduling results (the number of blocks requiring charging

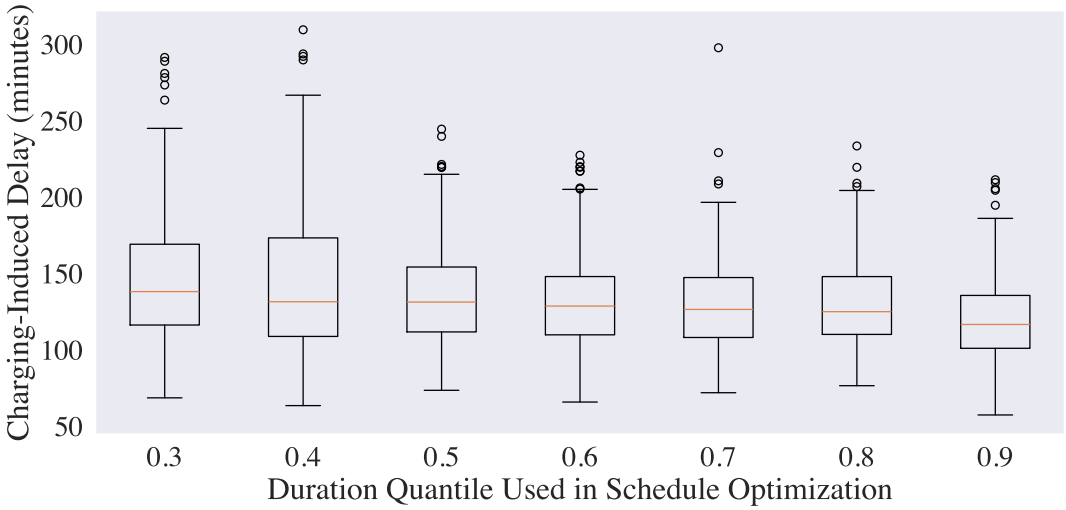


Figure 5.20: Impact of duration quantile used to set scheduling parameter on simulated delays.

and the amount of charging performed) as well as the mean simulated charging-induced delay across the 30 scenarios generated for each date. Note that for these cases, the duration is fixed across simulation runs since it is based on actual realtime durations, and only the energy consumption varies between runs. There is not much variance in the charging-induced delay across runs as a result—most scenarios yield the exact same delay, except for slight differences in scenarios that require unplanned charges—so we report only the mean value.

Table 5.8: Charge scheduling model results for testing data.

	March 28	**March 29**	**March 30**	**March 31**
Blocks Included	174	174	127	121
Trips Included	1285	1285	852	844
Number of Charging Blocks	45	37	25	28
Number of Charges Scheduled	77	63	35	46
Total Charging Time (min.)	806	685	419	539
Exogenous Delay (min.)	230	249	144	97
Charging-Induced Delay (min.)	84	28	27	27

Table 5.8 shows that the scheduling results from 3S perform well across all four test dates, with at most about five hours of total charging-induced delay spread across 45 different vehicles, an average of under 8 minutes of total charging-induced delay per block. In the broader context of transit service delays, this quantity is small. Table 5.8 also shows how the energy predictions from the regression model inform the 3S scheduling algorithm: although the same blocks and trips are active on March 28 and 29, the regression model predicts that energy consumption will be lower on the 29th and less total charging time will be needed. The result is that less charging-induced delay is present on the 29th than the 28th, even with a larger amount of exogenous delay introduced.

5.7 Discussion

The work included in this chapter spanned data collection and cleaning, statistical modeling, and simulation, with the goal of understanding how the charge scheduling model and 3S heuristic algorithm introduced in Chapter 4 perform when confronted with randomness. Analysis of the King County data showed that there were route-specific trends in trip durations; empirical distributions of on-time performance for each route were then used to simulate trip duration throughout the remaining analyses. A similar empirical approach would not be practical for trip energy consumption because these data were available for a much more limited set of trips; moreover, energy consumption showed a clear dependency on variables like elevation gain and outdoor temperature. These effects were better identified with a multiple linear regression model that was also used to generate realistic random values of energy consumption for all trips in the dataset based on vehicle size, trip elevation, planned speed, and temperature.

When these statistical models were used to simulate BEB system performance over a large number of testing scenarios, some clear trends emerged. We examined different approaches to setting charge scheduling optimization parameters based on quantiles of the estimated distributions and found that a slightly conservative approach was best for setting energy parameters, using the 60th percentile values. On the other hand, a more conservative strategy

using the 90th percentile was most effective for trip durations. There is a clear balance to be struck when setting the trip energy consumption value for scheduling, because conservative values generate more charging time so that buses maintained charged batteries more reliably, but planning too much charging can also generate delays that are unnecessary if actual energy consumption is less than the conservative predictions.

The sensitivity analysis on optimization parameters also underscores the value of modeling energy and trip durations as accurately as possible. While the models used in this chapter were able to capture real-world behavior reasonably well, if energy and trip time predictions can be made with greater certainty, performance can be expected to increase. Another takeaway from the sensitivity analysis is the value of building sufficient infrastructure, as BEB-OCL aims to. We saw that with the original 450 kW charger scenario, performance was generally good across a wide range of parameter quantiles. When charger power was reduced, making it more difficult to avoid delays, it became more important to set parameters properly to avoid poor performance. Essentially, providing plenty of chargers with a sufficiently high power output creates some cushion to get good performance under real-world variability even when that variability is not well reflected in the scheduling model parameters.

When the findings from the sensitivity analysis are applied to the four days of testing data, generating schedules based on 60th percentile predictions of energy consumption and 90th percentile predictions of durations, the resulting charge schedules perform well. Although some charging delay is present, the average amount of charging-induced delay per charging vehicle (across all trips) never exceeds 8 minutes. The calculated charging-induced delays ranging from 124 to 314 minutes also fall well within the typical range shown in the box plot of simulated performance in Figure 5.17, indicating some agreement between our simulated trip durations and the observed duration values used on the four days of testing data.

Finally, this chapter provides further emphasis on the value of Chapter 4's charging scheduling model and the 3S heuristic. Our flexible charge scheduling method that tracks but seeks to minimize delays lets us analyze difficult scenarios in which avoiding delays is not

possible. The sensitivity analysis showed that planning for these scenarios with conservative estimates of energy and trip duration can result in better overall performance, which would not be possible with the more common restrictive approach of constraining charging to keep trips on time. The speed with which the 3S method generates solutions—under 0.1 second per iteration in most of the test instances—was also an essential enabler of that sensitivity analysis.

Chapter 6

CONCLUSION

This dissertation presented several tools to plan for, analyze, and improve transit networks served by battery-electric buses. Chapter 2 described the GTFS-BEB data pipeline and ZEBRA web app that process General Transit Feed Specification data to identify the metrics most relevant to operating BEB systems, including the daily driving distances of all vehicle blocks. Chapter 3 described an optimization methodology to determine opportunity charging station locations and sizes to serve buses whose daily energy needs exceed their battery capacity. Chapter 4 presented an optimization approach to scheduling opportunity charging usage each day so that transit agencies can respond to daily changes in vehicles operating, trips served, energy consumption, and on-time performance. Chapter 5 analyzed real-world data from BEBs and conventional buses, which then served as the basis of a simulation platform to evaluate how the scheduling algorithm of Chapter 4 performed when faced with uncertainty in the key parameters that influence its solutions. The results indicated that despite the deterministic nature of the charging scheduling model, its solutions performed well across different scenarios as long as the parameters were set appropriately.

It should be noted that the modeling decisions and assumptions made throughout this work produce some limitations. First, batteries are assumed to gain charge linearly with respect to time whenever they use a charger. This should be a reasonable approximation when the battery state of charge is not too large or small, but may be a problematic assumption if buses are recharging nearly their entire state of charge, at which point charging behavior must change to protect battery health. In addition, the models throughout this work assume chargers are always used at their maximum rated power so that buses recharge as fast as possible, but some agencies may wish to adjust power output to limit their peak power and

resulting electricity bills.

One key limitation of Chapter 4's charging scheduling model is the assumption that deadhead distances to chargers are insignificant. While some deadheading can be accommodated by setting a deadhead distance threshold under which a charger is treated as accessible (as in Section 5.6), the model will still not perfectly account for this additional driving time and distance, which could lead to insufficient charging or unforeseen delays. On the other hand, not allowing longer deadhead trips to chargers may be overly restrictive as well. For example, the more flexible deadheading approach from Chapter 3 makes it possible for buses to use an opportunity charger located conveniently along the path of a pre-scheduled deadhead trip.

Finally, although the simulations of Chapter 5 provide some useful insights, the external validity of the linear regression model should be seen as questionable. Because all data collection was limited to March, few high temperatures were observed; likewise, the temperature rarely fell below freezing, so predictions based on weather conditions outside of the training data set should not be viewed as reliable. Additionally, because the training data were gathered from the small fleet of BEBs currently operating in King County, performance on routes that were not served by BEBs in the training dataset is not known. Likewise, energy consumption may be significantly different for other models of BEB besides the two New Flyer models included in this analysis.

This dissertation also generated several promising directions for future work. One particularly promising area is the application of the charging scheduling model and 3S heuristic to dynamic rescheduling. The sensitivity analysis on energy consumption highlighted the tradeoff between making sure enough charging is planned and avoiding unnecessary charges that introduce delays. By redesigning charging plans dynamically as the day progresses, it may be possible to achieve better balance. For example, if a bus consumes less energy than expected on its first few trips of the day, we might re-run the 3S algorithm and find a better solution with less total charging time than what was originally planned.

The analysis in Chapter 5 showed how the charging scheduling approach of Chapter 4 could perform well under real-world uncertainty, but the robustness of the BEB-OCL charger

location model from 3 was not studied in detail. Future work could use the simulation capabilities built for this dissertation to identify best practices for setting BEB-OCL parameters to ensure enough chargers are built to satisfy a wide range of operating conditions, without being overly conservative.

One more key direction for further research is better capturing interactions between BEB charging and the electric grid at both the planning and operational stages. At the long-term planning stage, directly accounting for electric grid capacity and reliability can help ensure that BEB charging infrastructure is economical and trustworthy. At the operational stage, recharging patterns have significant implications for an agency's total electricity bill, which typically depends on both the time energy is used and the maximum power load drawn over a billing period. Incorporating information about electricity costs into the charging scheduling model could help limit operations costs when buses have enough flexibility to charge at off-peak times or at less than maximum power.

To achieve their ambitious zero-emissions goals in the next one to two decades, American transit agencies will need to overcome a series of novel technical challenges in planning and managing their BEB networks. System-scale computational approaches like those proposed in this dissertation will be essential tools to help agencies manage costs, cope with uncertainty, and ultimately deliver reliable and even more sustainable transit service.

BIBLIOGRAPHY

[1] Hatem Abdelaty, Abdullah Al-Obaidi, Moataz Mohamed, and Hany E. Z. Farag. Machine learning prediction models for battery-electric bus energy consumption in transit. *Transportation Research Part D: Transport and Environment*, 96:102868, July 2021. ISSN 1361-9209. doi: 10.1016/j.trd.2021.102868. URL https://www.sciencedirect.com/science/article/pii/S1361920921001693.

[2] Ayman Abdelwahed, Pieter L. van den Berg, Tobias Brandt, John Collins, and Wolfgang Ketter. Evaluating and optimizing opportunity fast-charging schedules in transit battery electric bus networks. *Transportation Science*, 54(6):1601–1615, November 2020. ISSN 15265447. doi: 10.1287/trsc.2020.0982. Publisher: INFORMS Inst.for Operations Res.and the Management Sciences.

[3] Zackary Aemmer. *Navigating Widespread Urban Transit Dynamics With Standardized Data and Scalable Models*. Ph.D., University of Washington, United States – Washington, 2024. URL https://www.proquest.com/pqdtglobal/docview/3081471509/abstract/87B9522805964581PQ/1. ISBN: 9798383220528.

[4] Kun An. Battery electric bus infrastructure planning under demand uncertainty. *Transportation Research Part C: Emerging Technologies*, 111:572–587, 2020.

[5] Automotive World. Accuracy: UK's electric bus fleet set to be the largest in Europe by 2024, May 2021. URL https://www.automotiveworld.com/news-releases/accuracy-uks-electric-bus-fleet-set-to-be-the-largest-in-europe-by-2024/.

[6] Jacques F Benders. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4(1):238–252, 1962.

[7] U Narayan Bhat. *An Introduction to Queueing Theory: Modeling and Analysis in Applications*. Birkhäuser, 2015.

[8] BloombergNEF. Electric vehicle outlook 2021. Technical report, BloombergNEF, 2021.

[9] Kelly Blynn. Accelerating bus electrification: Enabling a sustainable transition to low carbon transportation systems. Master's thesis, Massachusetts Institute of Technology, 2018.

[10] Nico Brinkel, Marle Zijlstra, Ronald van Bezu, Tim van Twuijver, Ioannis Lampropoulos, and Wilfried van Sark. A comparative analysis of charging strategies for battery electric buses in wholesale electricity and ancillary services markets. *Transportation Research Part E: Logistics and Transportation Review*, 172:103085, April 2023. ISSN 1366-5545. doi: 10.1016/j.tre.2023.103085. URL `https://www.sciencedirect.com/science/article/pii/S136655452300073X`.

[11] Aaron Brooker, Jeffrey Gonder, Lijuan Wang, Eric Wood, Sean Lopp, and Laurie Ramroth. Fastsim: A model to estimate vehicle efficiency, cost and performance. Technical report, SAE Technical Paper, 2015.

[12] Claire Buysse. Zero-emission bus and truck market in the united states and canada: A 2021 update. Technical report, International Council on Clean Transportation, September 2022. `https://theicct.org/wp-content/uploads/2022/09/update-ze-truck-bus-market-us-can-sept22.pdf`.

[13] Michael L. Bynum, Gabriel A. Hackebeil, William E. Hart, Carl D. Laird, Bethany L. Nicholson, John D. Siirola, Jean-Paul Watson, and David L. Woodruff. *Pyomo–Optimization Modeling in Python*, volume 67. Springer Science & Business Media, third edition, 2021.

[14] CALSTART. Drive to Zero's Zero-emission Technology Inventory

(ZETI) Tool Version 8.3. https://globaldrivetozero.org/tools/zero-emission-technology-inventory/, 2024. Accessed: 2024-03-28.

[15] Avishai Ceder. *Public Transit Planning and Operation*. CRC Press, 2007.

[16] ChargePoint. Electrify your bus fleet with chargepoint® solutions. https://www.chargepoint.com/solutions/passenger-transportation-fleet, 2023. Accessed: 2023-01-10.

[17] Yuche Chen, Yunteng Zhang, and Ruixiao Sun. Data-driven estimation of energy consumption for electric bus under real-world driving conditions. *Transportation Research Part D: Transport and Environment*, 98:102969, September 2021. ISSN 1361-9209. doi: 10.1016/j.trd.2021.102969. URL https://www.sciencedirect.com/science/article/pii/S1361920921002674.

[18] Chicago Transit Authority. Charging forward: Cta bus electrification planning report. Technical report, Chicago Transit Authority, February 2022. https://www.transitchicago.com/assets/1/6/Charging_Forward_Report_2-10-22_(FINAL).pdf.

[19] Gianni Codato and Matteo Fischetti. Combinatorial benders' cuts for mixed-integer linear programming. *Operations Research*, 54(4):756–766, 2006.

[20] George Dantzig, Ray Fulkerson, and Selmer Johnson. Solution of a large-scale traveling-salesman problem. *Journal of the operations research society of America*, 2(4):393–410, 1954.

[21] Giovanni De Filippo, Vincenzo Marano, and Ramteen Sioshansi. Simulation of an electric transportation system at the Ohio State University. *Applied Energy*, 113:1686–1691, 2014.

[22] Guy Desaulniers and Mark D Hickman. Public transit. In Cynthia Barnhart and Gilbert Laporte, editors, *Handbooks in Operations Research and Management Science: Transportation*, chapter 2, pages 69–127. Elsevier, 2007.

[23] Nan Ding, Rajan Batta, and Changhyun Kwon. Conflict-free electric vehicle routing problem with capacitated charging stations and partial recharge. Technical report, University at Buffalo, SUNY, 2015. URL `http://www.acsu.buffalo.edu/~batta/Nan%20Ding.pdf`.

[24] Erica E Eggleton, Ryan Carlin, Harrison J Goldwyn, Lynna Truong, Danny Ilioiu, and Daniel T Schwartz. Route_dynamics–an open source python package for analyzing route-specific energy and power requirements that influence battery performance in electric bus fleets. *Available at SSRN 4158241*, 2022.

[25] Tolga Ercan and Omer Tatari. A hybrid life cycle assessment of public transportation buses with alternative fuel options. *The International Journal of Life Cycle Assessment*, 20(9):1213–1231, 2015.

[26] S. Esmaeilnejad, L. Kattan, and S. C. Wirasinghe. Optimal charging station locations and durations for a transit route with battery-electric buses: A two-stage stochastic programming approach with consideration of weather conditions. *Transportation Research Part C: Emerging Technologies*, 156:104327, November 2023. ISSN 0968-090X. doi: 10.1016/j.trc.2023.104327. URL `https://www.sciencedirect.com/science/article/pii/S0968090X23003169`.

[27] Leslie Eudy and Matthew Jeffers. Long Beach Transit Battery Electric Bus Evaluation (Final Report). Technical Report NREL/TP–5400-75582, 1665840, MainId:6400, National Renewable Energy Laboratory, September 2020. URL `https://www.osti.gov/servlets/purl/1665840/`.

[28] Federal Transit Administration. King County Department of Metro Transit: 2019

annual agency profile, 2019. https://www.transit.dot.gov/sites/fta.dot.gov/files/transit_agency_profile_doc/2019/00001.pdf.

[29] Federal Transit Administration. 2020 vehicles, 2020. URL https://www.transit.dot.gov/ntd/data-product/2020-vehicles.

[30] Federal Transit Administration. Bipartisan infrastructure law, Jan 2022. URL https://www.transit.dot.gov/BIL.

[31] Federal Transit Administration. Transit Bus Electrification Tool 1.0, 2022. URL https://www.transit.dot.gov/regulations-and-programs/environmental-programs/fta-transit-bus-electrification-tool.

[32] Federal Transit Administration. National transit database reporting changes and clarifications. *Federal Register*, 87(129), 2022.

[33] Aurélien Froger, Jorge E Mendoza, Ola Jabali, and Gilbert Laporte. A matheuristic for the electric vehicle routing problem with capacitated charging stations. Technical report, Centre interuniversitaire de recherche sur les reseaux d'entreprise, la logistique et le transport, 2017.

[34] Pranav Gairola and N. Nezamuddin. Optimization framework for integrated battery electric bus planning and charging scheduling. *Transportation Research Part D: Transport and Environment*, 118:103697, May 2023. ISSN 1361-9209. doi: 10.1016/j.trd.2023.103697. URL https://www.sciencedirect.com/science/article/pii/S1361920923000949.

[35] Zhiming Gao, Zhenhong Lin, Tim J LaClair, Changzheng Liu, Jan-Mou Li, Alicia K Birky, and Jacob Ward. Battery capacity and recharging needs for electric buses in city transit service. *Energy*, 122:588–600, 2017.

[36] Madeline Gilleran, Andrew Kotz, Leslie Eudy, and Kay Kelly. National Park Service Bus Electrification Study: 2020 Report. Technical Report NREL/TP–5400-78012,

1768279, MainId:31921, National Renewable Energy Laboratory, February 2021. URL https://www.osti.gov/servlets/purl/1768279/.

[37] GIRO. Electric buses. https://www.giro.ca/en-ca/our-solutions/segments/electric-buses/, 2023. Accessed: 2023-01-10.

[38] Google Maps Platform. Directions API Overview, 2022. https://developers.google.com/maps/documentation/directions/overview.

[39] Google Transit APIs. GTFS static overview, 2020. https://developers.google.com/transit/gtfs.

[40] William E Hart, Jean-Paul Watson, and David L Woodruff. Pyomo: Modeling and solving mathematical programs in Python. *Mathematical Programming Computation*, 3(3):219–260, 2011.

[41] Yi He, Zhaocai Liu, and Ziqi Song. Fast charging battery buses for the electrification of urban public transport—a feasibility study focusing on charging infrastructure and energy storage requirements. *Transportation Research Part D*, 117:103653, 2023.

[42] M John Hodgson. A flow-capturing location-allocation model. *Geographical Analysis*, 22(3):270–279, 1990.

[43] Jacob Holden, Nicholas Reinicke, and Jeffrey Cappellucci. Routee: A vehicle energy consumption prediction engine. *Society of Automotive Engineers Technical Paper Series*, 2(NREL/JA-5400-78089), 2020.

[44] John N Hooker. Planning and scheduling by logic-based benders decomposition. *Operations research*, 55(3):588–602, 2007.

[45] John N Hooker and Greger Ottosson. Logic-based benders decomposition. *Mathematical Programming*, 96(1):33–60, 2003.

[46] Mike Hynes, Alise Crippen, Kaila Lemons, and Emily Varnell. Zeroing in on zero-emission buses. Technical report, Calstart, February 2024. https://calstart.org/wp-content/uploads/2024/02/Zeroing-in-on-ZEBs-2024_Final-022324a.pdf.

[47] IEA. Global ev outlook 2023, 2023. URL https://www.iea.org/reports/global-ev-outlook-2023/trends-in-electric-heavy-duty-vehicles.

[48] IVU. Ivu.suite: e-mobility integrated. https://www.ivu.com/solutions/highlights/electric-mobility, 2023. Accessed: 2023-01-10.

[49] Matthew Jeffers and Leslie Eudy. Foothill Transit Battery Electric Bus Evaluation (Final Report). Technical Report NREL/TP-5400-80022, 1799885, MainId:41227, National Renewable Energy Laboratory, June 2021. URL https://www.osti.gov/servlets/purl/1799885/.

[50] Matthew Jeffers, Leslie Eudy, Erik Bigelow, Greg Olberding, and Amy Posner. Duluth Transit Authority Battery-Electric Bus Evaluation. Technical Report NREL/TP-5400-83038, 1887443, MainId:83811, National Renewable Energy Laboratory, September 2022. URL https://www.osti.gov/servlets/purl/1887443/.

[51] Caley Johnson, Erin Nobler, Leslie Eudy, and Matthew Jeffers. Financial Analysis of Battery Electric Transit Buses. Technical report, National Renewable Energy Laboratory, 2020.

[52] Jaeyoung Jung, Joseph Y.J. Chow, R. Jayakrishnan, and Ji Young Park. Stochastic dynamic itinerary interception refueling location problem with queue delay for electric taxi charging stations. *Transportation Research Part C: Emerging Technologies*, 40: 123–142, 2014. ISSN 0968090X. doi: 10.1016/j.trc.2014.01.008.

[53] Merve Keskin, Gilbert Laporte, and Bülent Çatay. Electric vehicle routing problem with time-dependent waiting times at recharging stations. *Computers and Operations Research*, 107:77–94, Jul 2019. ISSN 03050548. doi: 10.1016/j.cor.2019.02.014.

[54] Merve Keskin, Bülent Çatay, and Gilbert Laporte. A simulation-based heuristic for the electric vehicle routing problem with time windows and stochastic waiting times at recharging stations. *Computers and Operations Research*, 125, Jan 2021. ISSN 03050548. doi: 10.1016/j.cor.2020.105060.

[55] King County. Executive Constantine announces purchase of up to 120 battery-electric buses from New Flyer of America, Inc., Jan 2020. https://kingcounty.gov/elected/executive/constantine/news/release/2020/January/30-metro-battery-electric-bus-order.aspx.

[56] King County Metro. Transitioning to a zero-emissions bus fleet, Jun 2020. https://kingcounty.gov/depts/transportation/metro/programs-projects/innovation-technology/zero-emission-fleet.aspx.

[57] King County Metro. Transit development plan 2023-2028. https://cdn.kingcounty.gov/-/media/king-county/depts/metro/documents/about/data-and-reports/2023/transit-dev-plan-2023-2028.pdf?rev=ca51d181295f4c3491c28333e8fa6446&hash=13C6135E5C70241E8BBB478CAD80AA3E, 2023. Accessed: 2024-07-28.

[58] King County Metro. King County Metro GTFS feed, 2023. https://metro.kingcounty.gov/GTFS/.

[59] King County Metro. King County Metro electric bus summary. https://assets-global.website-files.com/65dd1596eb8f2dc1fec8c36d/65fa1434f284724e556fdc99_King%20County%20Metro%20BEBs%20Summary.pdf, 2024. Accessed: 2024-07-28.

[60] King County Metro. King County Metro zero-emissions fleet. https://www.kingcountymetrozefleet.com/, 2024. Accessed: 2024-07-28.

[61] King County Metro Transit. Battery-electric bus implementation report: Interim base and beyond. Technical report, King County Metro Transit, January 2020.

[62] King County Metro Transit. 2020 system evaluation. Technical report, King County Metro Transit, 2020.

[63] King County Metro Transit. Moving to a zero-emission bus fleet: Transition plan. Technical report, King County Metro Transit, May 2022. https://kingcounty.gov/~/media/depts/metro/accountability/reports/2022/zero-emission-bus-fleet-transition-plan-may-2022.

[64] Anton J Kleywegt, Alexander Shapiro, and Tito Homem-de Mello. The sample average approximation method for stochastic discrete optimization. *SIAM Journal on Optimization*, 12(2):479–502, 2002.

[65] Michael Kuby and Seow Lim. The flow-refueling location problem for alternative-fuel vehicles. *Socio-Economic Planning Sciences*, 39(2):125–145, 2005.

[66] Alexander Kunith, Roman Mendelevitch, and Dietmar Goehlich. Electrification of a city bus network—an optimization model for cost-effective placing of charging infrastructure and battery sizing of fast-charging electric bus systems. *International Journal of Sustainable Transportation*, 11(10):707–720, 2017.

[67] Rémi Lacombe, Nikolce Murgovski, Sébastien Gros, and Balázs Kulcsár. Integrated Charging Scheduling and Operational Control for an Electric Bus Network, September 2023. URL http://arxiv.org/abs/2309.00523. arXiv:2309.00523 [cs, eess].

[68] Lu Li, Hong K Lo, and Feng Xiao. Mixed bus fleet scheduling under range and refueling constraints. *Transportation Research Part C: Emerging Technologies*, 104:443–462, 2019.

[69] Yuping Lin, Kai Zhang, Zuo Jun Max Shen, Bin Ye, and Lixin Miao. Multistage large-scale charging station planning for electric buses considering transportation network

and power grid. *Transportation Research Part C: Emerging Technologies*, 107, 2019. ISSN 0968090X. doi: 10.1016/j.trc.2019.08.009.

[70] Kai Liu, Hong Gao, Yang Wang, Tao Feng, and Cheng Li. Robust charging strategies for electric bus fleets under energy consumption uncertainty. *Transportation Research Part D: Transport and Environment*, 104:103215, March 2022. ISSN 1361-9209. doi: 10.1016/j.trd.2022.103215. URL https://www.sciencedirect.com/science/article/pii/S1361920922000451.

[71] Tao Liu and Avishai Ceder. Battery-electric transit vehicle scheduling with optimal number of stationary chargers. *Transportation Research Part C: Emerging Technologies*, 114:118–139, 2020.

[72] Zhaocai Liu and Ziqi Song. Robust planning of dynamic wireless charging infrastructure for battery electric buses. *Transportation Research Part C: Emerging Technologies*, 83:77–103, 2017. ISSN 0968090X. doi: 10.1016/j.trc.2017.07.013. URL http://dx.doi.org/10.1016/j.trc.2017.07.013.

[73] Los Angeles County Metropolitan Transportation Authority. Final rollout plan. Technical report, Los Angeles County Metropolitan Transportation Authority, March 2021. https://ww2.arb.ca.gov/sites/default/files/2021-09/LAMetroRolloutPlanADA.pdf.

[74] Jônatas Augusto Manzolli, Joao Pedro F Trovao, and Carlos Henggeler Antunes. Electric bus coordinated charging strategy considering v2g and battery degradation. *Energy*, 254:124252, 2022.

[75] Massachusetts Bay Transportation Authority. Mbta approach to overcoming winter range challenges with battery electric buses. Technical report, Massachusetts Bay Transportation Authority, September 2021. https://cdn.mbta.com/sites/default/

files/2021-09/2021-09-30-overcoming-winter-range-with-bebs-accessible.pdf.

[76] Dan McCabe and Xuegang (Jeff) Ban. Optimal locations and sizes of layover charging stations for electric buses. *Transportation Research Part C: Emerging Technologies*, 152:104157, July 2023. ISSN 0968-090X. doi: 10.1016/j.trc.2023.104157. URL https://www.sciencedirect.com/science/article/pii/S0968090X23001468.

[77] Dan McCabe, Xuegang, Ban, and Balazs Kulcsar. Minimum-Delay Opportunity Charging Scheduling for Electric Buses, March 2024. URL http://arxiv.org/abs/2403.17527. arXiv:2403.17527 [math].

[78] Daniel McCabe. Selecting layover charging locations for electric buses: Mixed-integer linear programming models. Master's thesis, University of Washington, 2021.

[79] Jen McGraw, Peter Haas, Reid Ewing, and Sadegh Sabouri. An update on public transportation's impacts on greenhouse gas emissions. Technical report, Transportation Research Board and National Academies of Sciences, Engineering, and Medicine, 2021.

[80] Metropolitan Transportation Authority. 2023 zero-emission transition plan. Technical report, Metropolitan Transportation Authority, October 2023. https://new.mta.info/document/120411.

[81] MobilityData. GTFS: Making Public Transit Data Universally Accessible, 2024. URL https://gtfs.org/.

[82] MobilityData. Adding new fields to GTFS Realtime, 2024. URL https://gtfs.org/realtime/process/#revision-history.

[83] Saadiq Mohiuddin. Energy consumption modelling for electric bus schedule design. https://saadiqm.com/2022/08/28/transit-energy-modelling.html, 2022. Accessed: 2024-07-28.

[84] Alejandro Montoya, Christelle Guéret, Jorge E Mendoza, and Juan G Villegas. The electric vehicle routing problem with nonlinear charging function. *Transportation Research Part B: Methodological*, 103:87–110, 2017.

[85] Eammon Mulholland and Felipe Rodríguez. The rapid deployment of zero-emission buses in europe. Technical report, International Council on Clean Transportation, September 2022. https://theicct.org/wp-content/uploads/2022/09/zero-emission-buses-europe-sept22.pdf.

[86] National Centers for Environmental Information. Hourly/sub-hourly observational data, 2024. https://www.ncei.noaa.gov/maps/hourly/.

[87] New Flyer Infrastructure Solutions. Charger catalog. https://www.newflyer.com/site-content/uploads/2021/01/2021-Charger-Catalog.pdf, 2021. Accessed: 2021-06-05.

[88] Anders Nordelöf, Mia Romare, and Johan Tivander. Life cycle assessment of city buses powered by electricity, hydrogenated vegetable oil or diesel. *Transportation Research Part D: Transport and Environment*, 75:211–222, 2019.

[89] Openrouteservice. Openrouteservice, 2024. https://openrouteservice.org/.

[90] Optibus. Plan and optimize electric vehicle operations. https://www.optibus.com/solutions/optibus-for-electrical-vehicles/, 2023. Accessed: 2023-01-10.

[91] Teresa Pamuła and Wiesław Pamuła. Estimation of the Energy Consumption of Battery Electric Buses for Public Transport Networks Using Real-World Data and Deep Learning. *Energies*, 13(9):2340, January 2020. ISSN 1996-1073. doi: 10.3390/en13092340. URL https://www.mdpi.com/1996-1073/13/9/2340.

[92] Samuel Pelletier, Ola Jabali, Jorge E. Mendoza, and Gilbert Laporte. The electric bus fleet transition problem. *Transportation Research Part C: Emerging Technologies*, 109, 2019. ISSN 0968090X. doi: 10.1016/j.trc.2019.10.012.

[93] Shyam SG Perumal, Richard M Lusby, and Jesper Larsen. Electric bus planning & scheduling: A review of related problems and methodologies. *European Journal of Operational Research*, 2022.

[94] Samantha Pettigrew, Helmer Acevedo, and Oscar Delgado. Charging infrastructure for zero-emission buses — strategies in bogotá' d.c., colombia. Technical report, International Council on Clean Transportation, December 2023. `https://theicct.org/wp-content/uploads/2023/12/ID-53-%E2%80%93-Zero-emission-buses-in-Colombia_final.pdf`.

[95] Ragheb Rahmaniani, Teodor Gabriel Crainic, Michel Gendreau, and Walter Rei. The benders decomposition algorithm: A literature review. *European Journal of Operational Research*, 259(3):801–817, 2017.

[96] Matthias Rogge, Sebastian Wollny, and Dirk Uwe Sauer. Fast charging battery buses for the electrification of urban public transport—a feasibility study focusing on charging infrastructure and energy storage requirements. *Energies*, 8(5):4587–4606, 2015.

[97] Matthias Rogge, Evelien van der Hurk, Allan Larsen, and Dirk Uwe Sauer. Electric bus fleet size and mix problem with optimization of charging infrastructure. *Applied Energy*, 211, 2018. ISSN 03062619. doi: 10.1016/j.apenergy.2017.11.051.

[98] Matthew Roth. How Google and Portland's TriMet Set the Standard for Open Transit Data. *Streetsblog SF*, 2010. URL `https://sf.streetsblog.org/2010/01/05/how-google-and-portlands-trimet-set-the-standard-for-open-transit-data/`.

[99] Mariana Teixeira Sebastiani, Ricardo Lüders, and Keiko Veronica Ono Fonseca. Evaluating electric bus operation for a real-world brt public transportation using simulation optimization. *IEEE Transactions on Intelligent Transportation Systems*, 17(10):2777–2786, 2016.

[100] Chang Shen and Shiyue Mao. Zero-emission bus and truck market in china: A 2022 update. Technical report, International Council on Clean Transportation, December 2023. https://theicct.org/wp-content/uploads/2023/12/ID-57-%E2%80%93-ZETs-China_Final.pdf.

[101] Zuo-Jun Max Shen, Bo Feng, Chao Mao, and Lun Ran. Optimization models for electric vehicle service operations: A literature review. *Transportation Research Part B: Methodological*, 128:462–477, 2019.

[102] Kyle Stock. In the race to electrification, the humble bus is in the lead, Jun 2023. URL https://www.bloomberg.com/news/articles/2023-06-09/buses-are-going-electric-faster-than-passenger-cars.

[103] Timothy M Sweda, Irina S Dolinskaya, and Diego Klabjan. Adaptive routing and recharging policies for electric vehicles. *Transportation Science*, 51(4):1326–1348, 2017.

[104] Olaf Teichert, Fengqi Chang, Aybike Ongel, and Markus Lienkamp. Joint optimization of vehicle battery pack capacity and charging infrastructure for electrified public bus systems. *IEEE Transactions on Transportation Electrification*, 5(3):672–682, 2019.

[105] The Mobility House. Charging solutions for electric bus fleets. https://www.mobilityhouse.com/int_en/solutions/solutions-for-electric-bus-fleets, 2023. Accessed: 2023-01-10.

[106] The Thomas D. Larson Pennsylvania Transportation Institute. Altoona Bus Testing and Research Center Test Bus Procedure. Technical report, October 2017. URL https://www.altoonabustest.psu.edu/assets/docs/bus%20docs/updated%20procedures/Energy-Economy1.pdf.

[107] The Thomas D. Larson Pennsylvania Transportation Institute. Federal Transit Bus Test: New Flyer of America XE60. Technical Report LTI-BT-R2022-03-P,

September 2022. URL https://www.altoonabustest.psu.edu/bus-details.aspx?BN=2022-04.

[108] The Thomas D. Larson Pennsylvania Transportation Institute. Federal Transit Bus Test: New Flyer of America XE40. Technical Report LTI-BT-R2022-04, July 2023. URL https://www.altoonabustest.psu.edu/bus-details.aspx?BN=2022-04.

[109] The World Bank. Latin america can inspire electric buses adoption worldwide, Mar 2021. URL https://www.worldbank.org/en/news/feature/2021/03/23/uso-de-buses-electricos-marcha-sobre-ruedas-en-latinoamerica.

[110] Transitland. Feeds. https://www.transit.land/feeds, 2023. Accessed: 2023-01-23.

[111] Transportation Research Board and National Academies of Sciences, Engineering, and Medicine. *Battery Electric Buses—State of the Practice*. The National Academies Press, Washington, DC, 2018. doi: 10.17226/25061. URL https://www.nap.edu/catalog/25061/battery-electric-buses-state-of-the-practice.

[112] Transportation Research Board and National Academies of Sciences, Engineering, and Medicine. *Guidebook for Deploying Zero-Emission Transit Buses*. The National Academies Press, Washington, DC, 2021. doi: 10.17226/25842. URL https://www.nap.edu/catalog/25842/guidebook-for-deploying-zero-emission-transit-buses.

[113] Christopher Upchurch, Michael Kuby, and Seow Lim. A model for location of capacitated alternative-fuel stations. *Geographical Analysis*, 41(1):85–106, 2009.

[114] U.S. Geological Survey. 3D Elevation Program 1/3 Arc-Second Resolution Digital Elevation Model. https://www.sciencebase.gov/catalog/item/640824fdd34e76f5f75e4053, March 2023.

[115] U.S. Geological Survey. 3D Elevation Program 1/3 Arc-Second Resolution Digital Elevation Model. https://www.sciencebase.gov/catalog/item/640824fed34e76f5f75e4055, March 2023.

[116] Xiumin Wang, Chau Yuen, Naveed Ul Hassan, Ning An, and Weiwei Wu. Electric vehicle charging station placement for urban public bus systems. *IEEE Transactions on Intelligent Transportation Systems*, 18(1):128–139, 2016.

[117] Ying-Wei Wang and Chuah-Chih Lin. Locating road-vehicle refueling stations. *Transportation Research Part E: Logistics and Transportation Review*, 45(5):821–829, 2009.

[118] Ran Wei, Xiaoyue Liu, Yi Ou, and S Kiavash Fayyaz. Optimizing the spatio-temporal deployment of battery electric bus system. *Journal of Transport Geography*, 68:160–168, 2018.

[119] Maria Xylia, Sylvain Leduc, Piera Patrizio, Florian Kraxner, and Semida Silveira. Locating charging infrastructure for electric buses in Stockholm. *Transportation Research Part C: Emerging Technologies*, 78:183–200, 2017.

[120] Jie Yang, Jing Dong, and Liang Hu. A Data-Driven Optimization-Based Approach for Siting and Sizing of Electric Taxi Charging Stations. *Transportation Research Part C: Emerging Technologies*, 77(2):462–477, 2017. ISSN 0968090X. doi: 10.1016/j.trc.2017.02.014.

[121] Zemo Partnership. Summary of low and zero emission buses in service in the uk. https://www.zemo.org.uk/work-with-us/buses-coaches/low-emission-buses/areas-of-operation.htm, 2024. Accessed: 2024-03-27.

[122] Boya Zhou, Ye Wu, Bin Zhou, Renjie Wang, Wenwei Ke, Shaojun Zhang, and Jiming Hao. Real-world performance of battery electric buses and their life-cycle benefits with respect to energy consumption and carbon dioxide emissions. *Energy*, 96:603–613, 2016.