

Technical Memorandum

FDOT Master University Agreement BDV29-977-66

**Assessment of the Mobility Energy Productivity (MEP) Metric
for Transportation Applications in Florida
Deliverable 6 –Final Report**

Prepared For

Transit Office

Florida Department of Transportation

Prepared By

Xia Jin, Daniel Frolich, Md Nizamul Mojumder
Florida International University
10555 E Flagler St., EC 3603
Miami, FL 33174

Venu Garikapati
Team Leader, Transportation Modeling and Metrics
Center for Integrated Mobility Sciences
National Renewable Energy Laboratory (NREL)

Date

June 2023

DISCLAIMER

The opinions, findings, and conclusions expressed in this publication are those of the authors and not necessarily those of the State of Florida Department of Transportation or the U.S. Department of Transportation.

METRIC CONVERSION CHART

APPROXIMATE CONVERSIONS TO SI UNITS

SYMBOL	WHEN YOU KNOW	MULTIPLY BY	TO FIND	SYMBOL
LENGTH				
in	inches	25.4	millimeters	mm
ft	feet	0.305	meters	m
yd	yards	0.914	meters	m
mi	miles	1.61	kilometers	km
AREA				
in²	square inches	645.2	square millimeters	mm ²
ft²	square feet	0.093	square meters	m ²
yd²	square yards	0.836	square meters	m ²
ac	acres	0.405	hectares	ha
mi²	square miles	2.59	square kilometers	km ²
VOLUME				
fl oz	fluid ounces	29.57	milliliters	mL
gal	gallons	3.785	liters	L
ft³	cubic feet	0.028	cubic meters	m ³
yd³	cubic yards	0.765	cubic meters	m ³
NOTE: Volumes greater than 1000 L shall be shown in m ³				
MASS				
oz	ounces	28.35	grams	g
lb	pounds	0.454	kilograms	kg
T	short tons (2000 lb)	0.907	megagrams (or metric ton)	Mg (or t)
TEMPERATURE (exact degrees)				
°F	Fahrenheit	5 (F-32)/9 or (F-32)/1.8	Celsius	°C
ILLUMINATION				
fc	foot-candles	10.76	lux	lx
fl	foot-Lamberts	3.426	candela/m ²	cd/m ²
FORCE and PRESSURE or STRESS				
lbf	pound force	4.45	newton	N
lbf/in²	pound force per square inch	6.89	kilopascals	kPa

APPROXIMATE CONVERSIONS TO SI UNITS

SYMBOL	WHEN YOU KNOW	MULTIPLY BY	TO FIND	SYMBOL
LENGTH				
mm	millimeters	0.039	inches	in
m	meters	3.28	feet	ft
m	meters	1.09	yards	yd
km	kilometers	0.621	miles	mi
AREA				
mm²	square millimeters	0.0016	square inches	in ²
m²	square meters	10.764	square feet	ft ²
m²	square meters	1.195	square yards	yd ²
ha	hectares	2.47	acres	ac
km²	square kilometers	0.386	square miles	mi ²
VOLUME				
mL	milliliters	0.034	fluid ounces	fl oz
L	liters	0.264	gallons	gal
m³	cubic meters	35.314	cubic feet	ft ³
m³	cubic meters	1.307	cubic yards	yd ³
MASS				
g	grams	0.035	ounces	oz
kg	kilograms	2.202	pounds	lb
Mg (or t)	mega grams (or metric ton)	1.103	short tons (2000 lb)	T
TEMPERATURE (exact degrees)				
°C	Celsius	1.8C+32	Fahrenheit	°F
ILLUMINATION				
lx	lux	0.0929	foot-candles	fc
cd/m²	candela/m ²	0.2919	foot-Lamberts	fl
FORCE and PRESSURE or STRESS				
N	newton	0.225	pound force	lbf
kPa	kilopascals	0.145	pound force per square inch	lbf/in ²

*SI is the symbol for the International System of Units. Appropriate rounding should be made to comply with Section 4 of ASTM E380. (Revised March 2003)

TECHNICAL REPORT DOCUMENTATION PAGE

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle Assessment of the Mobility Energy Productivity (MEP) Metric for Transportation Applications in Florida		5. Report Date June 2023	
		6. Performing Organization Code	
7. Author(s) Xia Jin, Daniel Frolich, Md Nizamul Mojumder, and Venu Garikapati		8. Performing Organization Report No.	
9. Performing Organization Name and Address Florida International University 10555 W. Flagler Street Miami, Florida 33174		10. Work Unit No. (TRAIS)	
		11. Contract or Grant No. BDV29-977-66	
12. Sponsoring Agency Name and Address Florida Department of Transportation 605 Suwannee St., Tallahassee, FL 32301		13. Type of Report and Period Covered Final Report January 2022 – July 2023	
		14. Sponsoring Agency Code	
15. Supplementary Notes			
16. Abstract This report outlines the use and application of the Mobility Energy Productivity (MEP) tool in evaluating the efficiency of transportation systems. The MEP metric assesses the quality of mobility in transportation systems, considering accessibility, energy consumption, and productivity. It incorporates time, cost, and energy-related measurements to assess accessibility and mobility. The key inputs for these calculations include isochrones, land use and employment data, energy intensity and cost parameters, activity frequency, and population density. Two main scenarios are studied: bike infrastructure improvements and the introduction of electric transit projects in South Florida. The findings suggest that enhancements in bike infrastructure and the transition to electric buses can significantly improve MEP scores, indicating more productive use of energy in mobility services. Nonetheless, the results highlight potential areas of improvement in MEP calculations for future scenarios. The report underscores the utility of the MEP tool in analyzing the impact of changes in infrastructure and service patterns on transportation systems. Further investigations are needed to fully understand the capabilities and limitations of the MEP tool. Future research can explore incremental parameter changes, alternative scenarios, and additional data sources. Improving instructions, guidelines, and developing a user-friendly interface would enhance the tool's accessibility and usability for informed decision-making.			
17. Key Words Transportation System Efficiency, Bike Infrastructure Improvements, Electric Transit Projects, Scenario Analysis, Energy Efficiency in Mobility Services		18. Distribution Statement No restrictions	
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 77	22. Price

EXECUTIVE SUMMARY

The Mobility, Energy, Productivity (MEP) tool is a quantitative framework that assesses the quality of mobility in transportation systems, considering accessibility, energy consumption, and productivity. This executive summary provides an overview of the methodology, key findings, and implications derived from the research conducted in South Florida.

The MEP metric, developed by the National Renewable Energy Laboratory (NREL), serves as a comprehensive performance measure for transportation systems. It incorporates time, cost, and energy-related measurements to assess accessibility and mobility. Case studies have demonstrated the practical application of the MEP metric in evaluating various scenarios, such as fuel efficiency improvements and the introduction of shared automated vehicles (SAV), providing insights into their impact on mobility. Additionally, the MEP metric can be customized to accommodate individual preferences, and its extension, the Freight Mobility Energy Productivity (F-MEP) metric, enables the assessment of freight system efficiency. Continued research is necessary to refine and enhance the MEP tool, improving its accuracy and customization capabilities.

In our case study in South Florida, the MEP tool was used to evaluate bike and transit scenarios. The bike scenario analyzed the impact of bike infrastructure improvements on mobility and accessibility. Adding the existing bike lane network resulted in a significant 141.18% increase in MEP scores compared to the baseline. However, future bike infrastructure improvements for 2045 showed a modest 3.15% increase, suggesting limited regional impact. The transit scenario assessed the effects of transit system expansions and the adoption of electric buses on energy. Surprisingly, the results revealed that route changes and additions for the 2045 case decreased the MEP score by 8.96%. However, when electrification of the bus fleet was considered, there was a 15.45% increase in the MEP score, emphasizing the importance of electrifying transit. The MEP tool proved sensitive to localized effects and effective in capturing transit service impacts. Future investigations should explore incremental parameter changes, alternative scenarios, and additional data sources. Improving instructions, guidelines, and developing a user-friendly interface would enhance the tool's accessibility and usability for informed decision-making.

In conclusion, the MEP tool provides a valuable framework for assessing the complex interactions between mobility, energy consumption, and productivity in transportation systems. It facilitates evidence-based decision-making and empowers stakeholders to develop sustainable and efficient transportation infrastructure. Continued development and refinement of the MEP tool will contribute to improved transportation planning practices and the realization of long-term sustainability goals.

CONTENTS

DISCLAIMER	2
METRIC CONVERSION CHART	3
TECHNICAL REPORT DOCUMENTATION PAGE	5
EXECUTIVE SUMMARY	1
LIST OF FIGURES	4
LIST OF TABLES	6
1 INTRODUCTION	7
2 LITERATURE REVIEW	8
2.1 MEP Methodology	8
2.2 Applications of the MEP Methodology	10
2.2.1 Columbus, Ohio	10
2.2.2 Denver Metropolitan Area	12
2.2.3 Detroit and Washington D.C.	12
2.2.4 Development of MEP approach for the Freight Systems in the U.S.	14
2.3 Summary	16
3 DATA COLLECTION AND PREPARATION	17
3.1 Overview of Key Input Data	17
3.2 Isochrones	17
3.2.1 Highway Network	18
3.2.2 Transit Network	19
3.3 Land Use Data	20
3.4 Energy Intensity and Cost Inputs	22
3.5 Frequency of Activity Engagement	22
3.6 Employment	23
3.7 Population	23
3.8 Test Run Results	23
4 SCENARIO DESIGN	28
4.1 Bike Scenario	28

4.2	Transit Scenario	34
5	MODEL STEPS	37
5.1	Preprocessing Data	37
5.2	Installing Docker and MEP code	37
6	Scenario Results.....	39
6.1	Bike Scenario Results.....	39
6.2	Transit Scenario Results	45
7	CONCLUSIONS.....	49
8	References	50
	APPENDIX A PREPROCESSING DETAILS.....	52
1.1	Installing Software	52
1.2	Preparing TAZ.....	54
1.3	Preparing Roadway Network	54
1.4	Preparing Bike Network	55
1.5	Preparing Transit Network.....	57
	APPENDIX B MODEL RUN PROCESS.....	61

LIST OF FIGURES

Figure 1 Isochrones of 10-, 20-, 30-, and 40-min travel time by biking. (Hou et al., 2019).....	11
Figure 2 Impact of fuel efficiency on MEP scores. (Hou et al., 2019).....	11
Figure 3 R Shiny interface for the customization of the MEP metric.....	13
Figure 4 SERPM highway network.	19
Figure 5 Intersecting MAZs with isochrones.	22
Figure 6 Calculated MEP scores based on SERPM network and TomTom network.....	24
Figure 7 Comparison of MEP scores based on SERPM network and TomTom network.....	25
Figure 8 Correlation between SERPM 8 MAZ data and CoStar data.....	26
Figure 9 Comparison of MEP scores based on SERPM data and TomTom+CoStar data.	26
Figure 10 Comparison of MEP scores based on SERPM data and TomTom+CoStar data.	27
Figure 11 Map of bike lanes in Florida from FDOT roadway characteristics inventory.	29
Figure 12 Miami Beach planned bicycle projects map.....	30
Figure 13 Fort Lauderdale Complete Streets Bike Network Plan	31
Figure 14 West Palm Beach Bikeway Recommendations	32
Figure 15 Map of baseline and 2045 scenario updates.....	33
Figure 16 Overview of three major transit systems in South Florida.....	35
Figure 17 Case study locations for bike improvements.....	39
Figure 18 Comparison between SERPM network and bike improvements.	40
Figure 19 Bike MEP score comparison between baseline and future network	42
Figure 20 Difference in MEP score between baseline and future network SERPM LU	43
Figure 21 Correlation matrix for MEP score baseline and future scenario SERPM LU	43
Figure 22 Case study inspection on specific TAZs to investigate local effects on MEP score	44
Figure 23 2015 transit network and future extensions for 2045.....	45
Figure 24 Litmus test of single point.....	46
Figure 25 Transit MEP score map.....	47
Figure 26 Transit MEP scores for 2015 and 2045 scenarios.....	48
Figure 27 Anaconda Navigator Download	52
Figure 28 Anaconda Navigator Installation.....	52
Figure 29 Applications to install within Anaconda Navigator	53
Figure 30 Essential Packages Installation	53
Figure 31 Code to Prepare TAZ	54
Figure 32 Code to Prepare Road Network	55
Figure 33 SERPM Highway Network with bike facilities marked in green.....	56
Figure 34 Code to Prepare Bike Network.....	57
Figure 35 GTFS structure diagram	58
Figure 36 Code to build Trip Start.....	58

Figure 37 Code to Build Stop Times	59
Figure 38 Code to Build Trips	59
Figure 39 Code to Build Calendar	60
Figure 40 Code to build Stops	60

LIST OF TABLES

Table 1 Key Input Data and Potential Data Sources	10
Table 2 Summary of MEP Applications.....	16
Table 3 Summary of Key Input Data for MEP Calculations	17
Table 4 SERPM Highway Network Functional Class.....	18
Table 5 GTFS Files Required for MEP Calculation.....	20
Table 6 SERPM 8 MAZ Employment Data by Industry	21
Table 7 Energy Intensity and Cost Values.....	22
Table 8 Activity Engagement Frequencies Used in MEP calculations	23
Table 9 Transit Operating Expenses for Miami Dade County Municipalities	34
Table 10 Energy Intensity and Cost by Mode	36
Table 11 Bike MEP Scores	41
Table 12 Transit Overall MEP score	48

1 INTRODUCTION

Focusing on integrated transportation and land-use development is a promising strategy for advancing environmental sustainability, economic development, and inclusive development. If done well, integrated spatial development, particularly the linkage of multimodal transportation investments and urban development can create positive and meaningful outcomes. A high-quality and efficient multimodal system that links urban and suburban centers with rural communities creates an environment that improves livability and quality of life. As our population continues to grow and age, there needs to be a conscious decision on how to plan for future transportation systems that are efficient and environmentally sensible.

In Florida, relatively new modes and transportation alternatives, such as micro-transit, express buses on managed lanes, as well as the purchase and use of alternative fuel vehicles present new opportunities. Therefore, a sophisticated methodology that can not only consider improving mobility and access but also help reduce carbon emissions needs to be in place. This study explores the use of the mobility energy productivity (MEP) metric as an evaluation tool. The metric could be used for assessing the impacts of various alternatives in improving the existing multimodal service or for using different modes and energy options to address current and future challenges in a cost-effective manner. The MEP metric can potentially assist with the planning and design of a more efficient multimodal system that can help improve mobility, accessibility, and environmental issues.

This project assesses the MEP metric tool to examine potential transportation applications in Florida. It provides insights on the elements used by the model, maturity of software development, system capabilities, data needs, ease of use, and potential applications. Findings from this research will help identify the possibility of implementation of the MEP metric model as a tool for planning, assessing, and improving development of transportation projects in Florida.

2 LITERATURE REVIEW

With the aim of providing a holistic quantitative framework to evaluate the quality of mobility provided by the transportation system, a team at the National Renewable Energy Laboratory (NREL) has created a performance measure called the MEP metric (*Hou et al., 2019*). This metric assesses the capability of a transportation system to link people to various opportunities. The MEP metric provides an opportunity to gauge accessibility at a particular location with existing network configurations and examine how different technological options (such as Electric Vehicles (EVs)) or infrastructure projects (such as implementing bus rapid transit (BRT) systems) could affect the location's mobility. An area with a high MEP score provides multiple travel modes to a variety of activities while minimizing time, cost, and energy utilization.

The development of the MEP metric could lead to unique strategies for addressing current and future challenges, such as traffic congestion, environmental pollution, and economic disparity. This review presents summaries of relevant documents of the MEP metric and is divided into two sections. The first section discusses the methodology for calculating the MEP metric. The second section summarizes case studies that have applied or extended the MEP metric for various locations.

2.1 MEP Methodology

This section is primarily based on *Hou et al., 2019* and other publications by the same authors. Various properties were considered for the development of the MEP metric. These properties include being applicable to all modes, being spatially scalable, having time-, cost-, and energy-related modal measurements, and being comparable across locations and for different planning approaches and technologies.

To calculate the MEP metric, the opportunities (work, school, meals, social, shopping, and medical) reachable in a specific travel time threshold through various modes are estimated using the following equation:

$$O_{nmt} = \sum_k O_{nkmt} * \frac{A^*}{A_k} * \frac{f_k}{\sum_k f_k} \quad (1)$$

- O_{nmt} is the total opportunities available by mode m in travel time t from the n th pixel.
- O_{nkmt} is the total opportunities of activity k available by mode m in travel time t from the n th pixel. Land-use and employment data are needed for calculating the total opportunities for each activity type.
- A^* is the total benchmark opportunities among multiple cities in the U.S.
- A_k is the total opportunities of activity k among multiple cities in the U.S.
- $(\frac{A^*}{A_k})$ is the relative spatial equivalency and is applied to translate the opportunities to a comparable benchmark opportunity. For each activity type (including work, meals, social

and recreational, shopping and errands, medical/dental services, and school/daycare/religious activities), the spatial equivalency factor was calculated using statistics from multiple cities in the U.S., default values are provided in the MEP tool, and can be customized by the users. This factor should remain constant for a given city when applying the MEP metric, but can vary from city to city.

- f_k is the frequency of accessing activity k . The MEP tool uses trip frequency to measure access frequency to activities.
- $(\frac{f_k}{\sum_k f_k})$ is the ratio of activity engagement to proportion the opportunities by the relative frequency of access to the activities. The data were derived from the 2017 National Household Travel Survey (NHTS) data. Default values for each activity type are provided in the MEP tool.

The MEP metric also applies modal weighting factors to account for the convenience (travel time), sustainability (energy efficiency), and affordability (cost) in accessing various activities. The modal weighting factors can be calculated as follows:

$$W_{nmt} = \beta e_m + \gamma t + \delta c_m \quad (2)$$

- W_{nmt} is the modal weighting factor for opportunities available by mode m in travel time t from the n th pixel.
- e_m is the intensity of energy (kilowatt-hours per passenger-mile) for mode m . For cars and transit, the energy intensity was obtained from Transportation Energy Data Book. For biking and walking, the energy intensity was assumed to be zero. Default values are provided in the MEP tool and customizable by the users.
- c_m is the cost (\$ per passenger-mile) for mode m . The transit and driving costs were obtained from Federal Transit Administration (FTA) and American Automobile Association (AAA), respectively. For biking and walking, energy intensity and travel cost were assumed to be zero. Default values are provided in the MEP tool and customizable by the users based on local conditions.
- t is the travel time, and
- β (set to be -0.5), γ (set to be -0.08), and δ (set to be -0.5) are weighting factors. The γ was determined based on a study by Owen and Levinson (2014), and β and δ were selected based on the judgment of the research team. These values are customizable by the users based on local conditions.

Then, the MEP metric is computed by weighting the cumulative opportunities utilizing a negative exponential function applied to the modal weighting factor. The following equation calculates the metric for location n :

$$MEP_n = \sum_m \sum_t (O_{nmt} - O_{nm(t-10)}) \cdot e^{W_{nmt}} \quad (3)$$

2.2 Applications of the MEP Methodology

2.2.1 Columbus, Ohio

Hou et al. (2019) illustrated the implementation of the MEP metric for Columbus, Ohio. Key input data required, and potential data sources are summarized in Table 1 below.

Table 1 Key Input Data and Potential Data Sources

Key Input Data	Potential Data Sources
Isochrones by mode	Travel demand models; the General Transit Feed Specification (GTFS); Mapzen, an open mapping platform (https://mapzen.com); OpenStreetMap (https://www.openstreetmap.org), and Transitland (https://transit.land)
Land use and employment data (to reflect cumulative opportunities)	Local planning organizations such as MPOs, cities; third-party data such as CoStarTM (http://www.costar.com), Google Places (https://developers.google.com/places/web-service/intro), or FourSquare (https://foursquare.com); Census Longitudinal Household Employer Dynamics (LEHD) data (https://lehd.ces.census.gov/data/)
Energy intensity and monetary cost for each mode	Transportation Energy Data Book (Oak Ridge National Laboratory); Federal Transit Administration (FTA, 2015); American Automobile Association (AAA)
Frequency of activity engagement	Trip frequency from the 2017 NHTS data for each activity type (work, meals, social and recreational, shopping and errands, medical/dental services, and school/daycare/religious activities)
Population Density (to aggregate MEP metric from individual pixels to larger geographies)	Census American Community Survey (ACS) (U.S. Census)

An example of isochrones is demonstrated in Figure 1. It shows isochrones of 10, 20, 30, and 40-min travel time by biking in Columbus, Ohio.

The Columbus study calculated MEP metrics for driving mode, all modes except driving, and all modes combined. The results indicated that driving provided access to more opportunities than walking, biking, and transit together. Moreover, the MEP scores were better in urban core areas

where energy-efficient modes could provide easy access to products, services, and destinations, compared with suburban or rural areas.

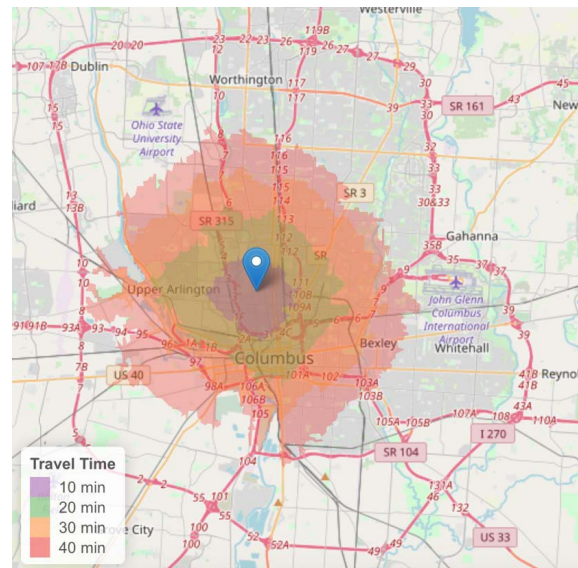
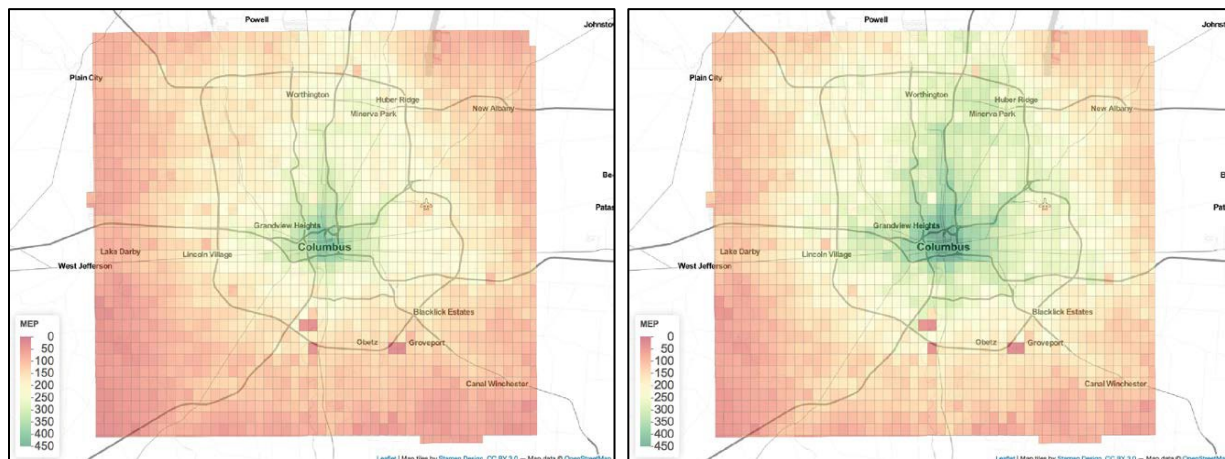


Figure 1 Isochrones of 10-, 20-, 30-, and 40-min travel time by biking. (Hou et al., 2019)

The study tested two scenarios to evaluate the impact of modal measurements on the MEP metric. The first scenario increased automobile fuel efficiency by 300% and the second scenario decreased car travel time by 7 minutes. The findings illustrated that increasing fuel efficiency increased the MEP score in all areas away from the downtown area, while reducing travel time improved mobility across the entire study area. Figure 2 below demonstrates visuals of the MEP output, showing MEP comparison between the baseline scenario and the fuel efficiency scenario.



(a) Baseline scenario.

(b) Fuel efficiency scenario.

Figure 2 Impact of fuel efficiency on MEP scores. (Hou et al., 2019)

2.2.2 Denver Metropolitan Area

Garikapati et al. (2019) applied the MEP metric to assess mobility in the Denver metropolitan area. The study focused on mobility to specific opportunities, including jobs, grocery stores, restaurants, recreation facilities, and medical services. Similar to the Columbus study, MEP metrics were calculated for different modal combinations, including all modes combined, driving, and all modes except driving. The results illustrated that cars enable access to more opportunities than other modes. Furthermore, the MEP metric for all modes except driving (transit, walk, and bike together) indicated a higher level of access in downtown areas than suburban areas.

The study also conducted three scenarios to evaluate the power and the functionality of the MEP metric. The first scenario focused on improving the fuel efficiency of vehicles by increasing effective MPG by 200%. As a result, the total MEP metric increased by 25%. The second scenario implemented shared automated vehicles (SAV) in a confined area in downtown Denver. Similar to the previous scenario, the overall MEP score increased, however, the improvement occurred only in the confined area. The last scenario added a Transportation Networking Company (TNC) mode in the network. For TNC trips, travel times were modified to reflect waiting time, the costs were increased to consider the fare, and energy intensity was changed to account for deadheading (i.e., vehicles without passengers). The results showed an increase in the MEP score in the study area after the inclusion of the TNC mode in the network.

2.2.3 Detroit and Washington, D.C.

Nag et al. (2020) extended the MEP method with more dynamic approach and customization. In this approach, people can tailor the metric based on their unique travel attributes (e.g., travel modes, the pattern of activities, and time-of-day inclinations). This approach allows the individual to choose and measure the relative importance of accessing various activities (work, schools, meals, social, shopping, and medical opportunities), and they can modify the energy intensity and travel cost of each mode based on their preferences.

A prototype was developed in R Shiny that enables individuals to select among the available modes (driving, walking, walking, transit, and TNC), choose the time they are willing to travel, and pick the activities they are interested in. Figure 3 illustrates the user interface in R Shiny for the customization of the MEP metric.

The app also allows the user to choose the modal preference for the selected modes, but the preferences must sum to 100%. Regarding mode preferences, the MEP score at each location for a specific mode was multiplied by the modal preference for the mode (p_m) (a unitless number between 0 and 1) and the number of selected modes. The customizable MEP metric for location n and mode m can be defined as below:

$$MEP_n = \sum_m [MEP_{nm} \times p_m \times \sum_m 1] \quad (4)$$

MEP Customization

MEP Customization

Choose City:
 Detroit

Choose Modes:
☒ drive
☒ tnc
☒ bike
☐ walk
☐ transit

Choose Drive Time Threshold:
 40

Choose TNC Time Threshold:
 40

Choose Bike Time Threshold:
 40

Choose Activities:
☒ sch_day_rel
☒ medical
☒ shop_err
☒ recreation
☒ meals
☒ jobs

Choose Time of Day:
 am_rush

Figure 3 R Shiny interface for the customization of the MEP metric.

The authors considered several hypothetical values for the modal preference to examine their impacts on the MEP score.

The proposed method was applied for two urban areas: Detroit and Washington D.C. Three scenarios were conducted for these locations. The first (baseline) scenario considered travel time up to 40 minutes and allowed travel by all modes (equal mode preferences were considered) to all activities. The next scenario, labeled as the “Car-Averse” scenario, allowed travel time up to 40 minutes to all activities and considered all modes except driving. In terms of modal preferences, a 70% preference was assigned to TNC, and a 10% preference was assigned to biking, walking, and transit, respectively. The third scenario, named the “nonmotorized” scenario, allows the individual to travel for up to 20 minutes by bike or walk, only for meals and recreational activities. A 75% and 25% modal preference were assigned to biking and walking, respectively.

The results showed that the drive mode had a substantial contribution to the MEP score as the second and third scenarios had a significantly lower MEP score than the first scenario. Moreover, comparing the base and Car-Averse scenarios, it showed that TNC was not equal to the driving mode (in terms of MEP metric) for any location, which can be attributed to higher fares, waiting

times, and energy intensity. The MEP score was the lowest for the nonmotorized scenario due to the additional removal of transit and TNC modes.

The study also conducted a series of modified scenarios for the Detroit urban area. The baseline scenario is similar to the previously defined baseline scenario. For the second scenario, unlike the Car-Averse scenario, an equal modal preference was assigned to TNC, transit, walk, and bike modes. Results showed that the MEP score decreased significantly compared to the Car-Averse scenario. The third scenario had similar assumptions to the second scenario except that the TNC fleet is electrified with higher fuel efficiency, which resulted to a 63% increase of the MEP score. The last scenario was built on the third scenario but the cost of TNC was reduced to assume the use of automated vehicles for its fleet. The findings illustrated that the MEP score increased by 54% in the fourth scenario compared to the third scenario. The last scenario indicates the significant impact of cost and fuel efficiency of travel modes on the mobility of a location.

2.2.4 Development of MEP Approach for the Freight Systems in the U.S.

Jeong et al. (2020) extended the MEP metric to freight and developed the Freight Mobility Energy Productivity (F-MEP) metric to evaluate the efficiency of existing and future freight systems (including air, water, rail, and truck) in the U.S., considering the associated costs (time, energy, and monetary cost). Because freight movement differs from passenger movement (e.g., wider geographical extent, movement within- and between- cities, and heterogeneity due to different product types), some additional assumptions were considered when developing the F-MEP metric. The assumptions include the ability to be fine-tuned to an area or product type, the capability to evaluate performance measures for different modes for existing and future freight systems, and the potential to assess the effect of emerging freight technologies.

Jeong considered a shipper's point of view for developing the metric. The F-MEP metric was defined as “the sum of the mobility benefit for shipping from a location o to any other location d that has freight attractions, weighted by the friction factors from o to d that are associated with energy, cost, travel time, and ease of shipping goods.” The F-MEP metric was computed as below:

$$FEMP_o = \sum_m \sum_n \sum_{d \neq o} B_{nd}(X) \cdot f_{n,od}^m(Y) \quad (5)$$

- n is commodity or business type,
- m is transportation mode,
- $B_{nd}(X)$ is the mobility benefit of commodity n at destination d , and it can be defined by various types of freight delivery opportunities (X). The commodity demand (tonnage value) at destination d is employed as a single variable for defining the mobility benefit. The commodity demand is derived from the Freight Analysis Framework (FAF), and
- $f_{n,od}^m(Y)$ is the impedance function defined by multiple cost factors (Y)

The impedance function was defined as follows:

$$f_{n,od}^m(Y) = \exp(\beta e_m + \delta c_m) \cdot r_{nm}^{lod} \cdot s_{om} \quad (6)$$

- e_m is the intensity of energy (kilowatt-hours per ton-mile) for mode m . For rail and water modes, the energy intensity information was obtained from the Transportation Energy Data Book. For truck mode, the total energy use for trucks was divided by the total ton-miles obtained from the Bureau of Transportation Statistics. The energy intensity of air mode was estimated by using data provided by Transportation Energy Futures.
- c_m is the cost (\$ per ton-mile) for mode m . The cost values were calculated based on data from U.S. business logistics costs as well as the total ton-miles for each mode.
- β and δ are weighting factors, and they are assumed to be -0.5,
- r_{nm}^{lod} is the weight fraction of commodity n moving in distance l (between origin o and destination d) by mode m . The weight fractions were determined utilizing commodity tonnage by distance for each mode, which was available in the Freight Analysis Framework, and
- s_{om} is the ease of commodity shipment.

The ease of commodity shipment is calculated by considering the number of intermodal facilities for each mode as a determining factor. s_{om} is defined as follows:

$$s_{om} = \begin{cases} 1 & \text{if } m = \text{truck} \\ \frac{F_{om}}{\max(F_{dm}, \forall d)} & \text{Otherwise} \end{cases} \quad (7)$$

- F_{om} is the number of intermodal facilities for the mode m at origin o .

F-MEP scores showed that Chicago and Salt Lake City had the highest and lowest scores, respectively. Generally, Mid-West and Mid-Atlantic regions had a higher F-MEP score than other areas, which could be related to the shorter distances from these regions to other areas, their higher accessibility to other transportation modes, and their proximity to the large freight markets of the Northeast. On the other hand, F-MEP scores showed significant heterogeneity among states, which could be originated from variations in commodity type, infrastructure richness, and shipment distance.

The authors calculated the correlation between F-MEP scores and the zones' employment and gross domestic product (GDP) to validate their results. The main assumption was that higher employment and GDP were associated with a higher F-MEP score. The correlation between these two factors and F-MEP was positive and significant, corroborating the findings. Furthermore, for scenario analysis, they considered two hypothetical situations: (1) electrification of powertrains with range limits (for distances less than 500 miles) and (2) electrification of powertrains without range limits (replacing all conventional trucks with electric trucks). The results indicated that zones with a high freight demand could significantly benefit from the electrification of trucks, and this was especially true in the second scenario.

Table 2 below summaries the case studies that have applied the MEP metric to demonstrate the types of scenarios and major findings.

Table 2 Summary of MEP Applications

Location	Application	Scenario Design	Major Findings
Columbus , Ohio	driving vs. Walking, biking and transit vs. all modes	Fuel efficiency travel time reduction	Better fuel efficiency and travel time reduction led to higher MEP scores
Denver Metro Area	driving vs. Walking, biking and transit vs. all modes	Fuel efficiency SAV; TNC mode	Higher MEP metrics as a result of higher fuel efficiency, and the introduction of SAVs and TNC modes
Detroit, IL; Washingt on D.C.	All modes vs. walking, biking and transit vs. walking and biking	Mode availability and modal preferences	Driving had substantial contribution to the MEP metric, TNC was not equivalent to driving, TNC with automation increased MEP by 54%
U.S.	Freight-MEP	Electrification of powertrains with and without range limit	Higher employment and GDP were associated with higher F-MEP scores; electrification improves F-MEP scores especially for areas with high freight demand

2.3 Summary

In summary, the MEP metric provides a comprehensive framework to quantify the value or quality of mobility based on the fundamental concept of accessibility. It measures the amount of opportunities accessible within certain time threshold (e.g., 10 minutes) weighted by modal factors (considering travel time, travel cost and energy efficiency). The cumulative opportunities are normalized to a benchmark opportunity to be comparable, and also weighted by importance of the opportunity (measured by trip frequency to that activity).

The MEP metric has high scalability in multiple aspects. The metric can be computed at small geographic scale (such as parcel, Census BG, or Census tract) and summed to a city or larger area weighted by population. It can also calculate mode-specific metrics and by opportunity type if it is desirable to focus on a specific mode or activity type (i.e., work, shopping, etc.)

In terms of applications, this tool can operate built on mapping platforms or be considered as a post processor of travel demand models, which are essential to estimate the network impacts of alternative projects and scenarios. The MEP metric is capable of testing various scenarios of vehicle technologies, modal options, land use and infrastructure changes.

3 DATA COLLECTION AND PREPARATION

3.1 Overview of Key Input Data

The previous report has summarized the required input data for MEP calculations and is included here again for convenience. The rest of this section discusses the data preparation for these key input data.

Table 3 Summary of Key Input Data for MEP Calculations

Key Input Data	Potential Data Sources
Isochrones by mode	Travel demand models; the General Transit Feed Specification (GTFS); Mapzen, an open mapping platform (https://mapzen.com); OpenStreetMap (https://www.openstreetmap.org), and Transitland (https://transit.land)
Land use and employment data (to reflect cumulative opportunities)	Parcel data from Local planning organizations such as MPOs, cities; third-party data such as CoStarTM (http://www.costar.com), Google Places (https://developers.google.com/places/web-service/intro), or FourSquare (https://foursquare.com); Census Longitudinal Household Employer Dynamics (LEHD) data (https://lehd.ces.census.gov/data/)
Energy intensity and monetary cost for each mode	Transportation Energy Data Book (Oak Ridge National Laboratory); Federal Transit Administration (FTA, 2015); American Automobile Association (AAA)
Frequency of activity engagement	Trip frequency from the 2017 NHTS data for each activity type (work, meals, social and recreational, shopping and errands, medical/dental services, and school/daycare/religious activities)
Population Density (to aggregate MEP metric from individual pixels to larger geographies)	Census American Community Survey (ACS) (U.S. Census)

3.2 Isochrones

An isochrone specifies the area accessible from an origin location within certain time threshold. Isochrones are needed to identify the number of opportunities one can reach within specified travel time thresholds by a mode or combination of multiple modes. Commonly considered modes in the MEP tool include drive, transit, bike and walk. Drive, bike and walk isochrones are built using the highway network. Other modes, such as high occupancy toll (HOT), or dedicated automated vehicle (AV) lanes can be specified as long as the corresponding facilities are provided.

3.2.1 Highway Network

The highway network used by most travel demand models can be directly used to calculate drive, walk, and bike isochrones. The key attributes needed include:

- Unique link ID
- Network link geometry, including the direction of travel (such as shape fields)
- Link length
- Link classification
- Estimated speed (from the loaded network as an output of the traffic assignment)

If the travel demand model performs assignment by time-period, then the isochrones can be created for specific time periods, which is also the preferred approach. It should also be noted that the MEP tool creates isochrones from each origin location, therefore requires network information in both directions. Bidirectional links need to be represented as two separate links. The highway network file for SERPM 8 has many bidirectional links. When exporting the network file (.net) into shapefiles in Cube, it automatically converts bidirectional links into separate links for each direction. Figure 4 below presents the SERPM highway network. Table 4 below shows the link classification.

Table 4 SERPM Highway Network Functional Class

Type	Description	Count
1	Freeway	887
2	Uninterrupted Roadways	194
4	Arterials	14,804
5	Centroid Connectors	16,262
6	Collectors	10,881
7	Ramps	2,965
8	HOV Lanes	335
9	Toll Roads	1,125
Total		46,566

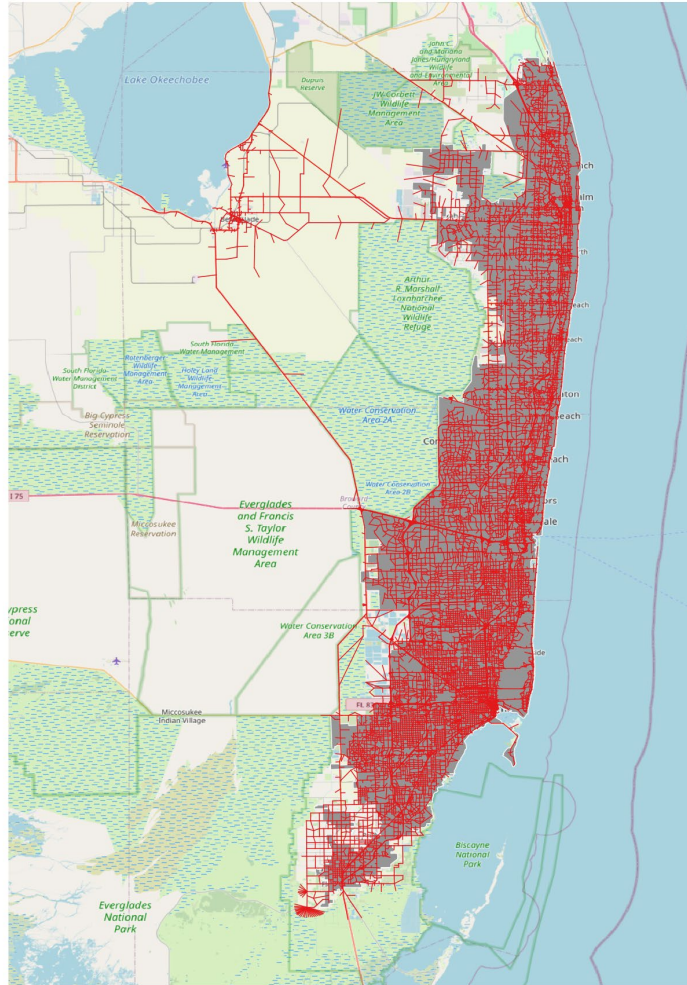


Figure 4 SERPM highway network.

3.2.2 Transit Network

The MEP tool requires transit network information in GTFS format. It is possible to convert transit network file in Cube (.net and .lin files) or TransCAD format into GTFS format. However, by nature, the transit network used by travel demand models are for planning purposes and does not have the operational level of details needed in GTFS format. For this reason, although transit network files are readily available for future scenario years from SERPM, those files are not usable by the MEP tool.

For the purpose of this task, base year (2015) GTFS files were downloaded from transit agencies' website. For future scenarios, the most feasible way is to modify the based year GTFS files on project basis. The research team will continue to explore the best approach and procedure to prepare transit network information for MEP.

The full list of files that comprise a GTFS feed for a transit line are listed here¹. The data files and attributes that are 'required' for a MEP calculation are listed in **Table 5**:

Table 5 GTFS Files Required for MEP Calculation

Filename	Defines
agency.txt	Transit agencies with service represented in this dataset.
stops.txt	Stops where vehicles pick up or drop off riders. Also defines stations and station entrances.
routes.txt	Transit routes. A route is a group of trips that are displayed to riders as a single service.
trips.txt	Trips for each route. A trip is a sequence of two or more stops that occur during a specific time period.
stop_times.txt	Times that a vehicle arrives at and departs from stops for each trip.

3.3 Land Use Data

MEP needs land use data to calculate the number of opportunities available within the isochrones. The MEP tool allows the specification of opportunities by category, including work, school, meals, social, shopping, and medical, so that specific MEP scores can be produced for specific type of opportunities or activities.

The ideal data source for the land use information would be parcel map data which would identify the activity type (from the 5 different categories noted above) pertaining to each building (or parcel). Although base year parcel data are available from the Department of Revenue or similar entities, obtaining or preparing data for future scenario years at parcel level would not be feasible. For this consideration, we recommend the use of micro analysis zones (MAZ) data provided by SERPM 8. This provides population and employment data for each MAZ for both base year (2015) and future year (2045).

NAICS data available from the MAZ file is categorized into the six difference MEP categories as shown in Table 6.

¹ <https://developers.google.com/transit/gtfs/reference>

Table 6 SERPM 8 MAZ Employment Data by Industry

MAZ Attribute	Industry	Entries	NAICS code	MEP Category
emp_ag	Agriculture, Forestry, Fishing, and Hunting	468	11	Jobs
emp_const_non_bldg_prod	Construction	6587	23	Jobs
emp_utilities_prod	Utilities	191	22	Jobs
emp_mfg_prod	Manufacturing	2416	31-33	Jobs
emp_whsle_whs	Wholesale Trade and Warehousing	3836	42,493	Jobs, Shopping
emp_trans	Transportation	3219	48-49	Jobs
emp_retail	Retail Activity	6477	44-45, 53	Jobs, Shopping
emp_prof_bus_svcs	Professional and Business Services	7740	51-55	Jobs
emp_pvt_ed_post_k12_oth	Education Post-Secondary	19	61	Jobs, School
emp_health	Health Services	3596	62	Jobs, Health
emp_personal_svcs_office	Personal Services	5870	62,81	Jobs
emp_amusement	Arts, Entertainment, and Recreation/Amusement Services	2132	71	Jobs, Recreation
emp_hotel	Accommodation/Hotels and Motels	817	72	Jobs, Meals
emp_restaurant_bar	Food Services/Restaurants and Bars	3373	72	Jobs, Meals
emp_state_local_gov_entity	Public Administration/Federal, State, and Local Government	1165	92	Jobs
emp_public_ed	Education Elementary and Secondary K-12	2722	61	Jobs

To calculate the number of opportunities within each isochrone, the MAZ boundaries are intersected with the isochrones (Figure 5). The employment within each MAZ is proportionally allocated to the isochrones based on the ratio of intersecting areas to the MAZ area.

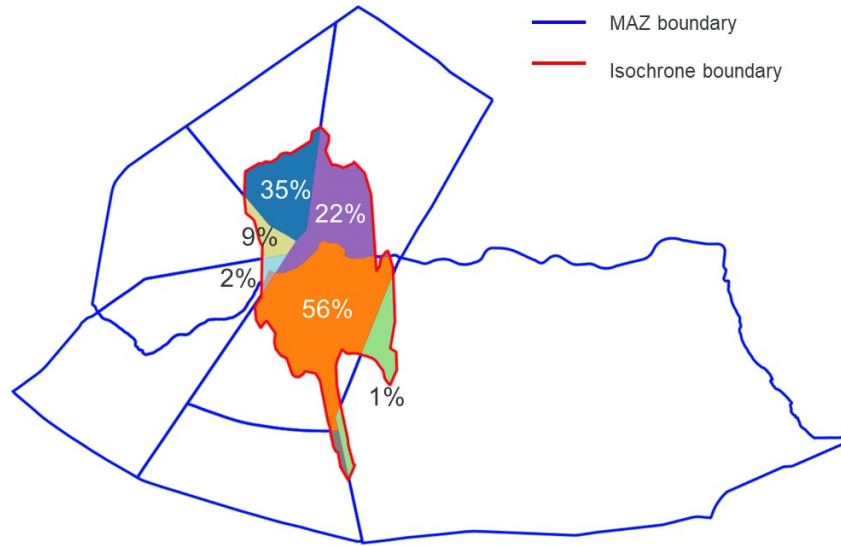


Figure 5 Intersecting MAZs with isochrones.

3.4 Energy Intensity and Cost Inputs

For the purpose of this test run, the default values provided in the MEP tool are used and listed in **Table 7** below (AAA, 2018; ALG 2016; Davis et al. 2017; FTA 2016).

Table 7 Energy Intensity and Cost Values

Mode	Energy intensity (kWh/passenger-mile)	Capital and operational cost (dollar/passenger-mile)
Driving	0.90	0.48
Transit	0.65	0.85
Bike	0	0
Walk	0	0
Transportation Network Company	1.8	1.54
Paratransit	4.13	2.25

The weighting factors were also set as the default values: β (-0.5), γ (-0.08), and δ (-0.5). The γ was determined based on a study by Owen and Levinson (2014), and β and δ were selected based on Hou et al. (2019). These values are customizable by the users based on local conditions.

3.5 Frequency of Activity Engagement

The 2017 National Household Travel Survey (NHTS) data were used to calculate the frequency of activity engagement for each opportunity type. Table 8 below lists the values used.

Table 8 Activity Engagement Frequencies Used in MEP calculations

Category	Frequency (%)
Jobs	30
Meals	12
Shopping/Errands	35
Hospital	3
School/Rel/Daycare	6
Social/Recreational	15

3.6 Employment

In a MEP calculation, access to employment is quantified separately from access to various types of activities. While the base MEP calculation uses employment data from the Longitudinal Employer-Household Dynamics², employment from the MAZ files is used for SERPM specific MEP runs.

3.7 Population

Total population (at the sq. km pixel level) is used for aggregating pixel level MEP scores to the city level. The base MEP calculation uses data from Census³. For SERPM specific MEP calculations, population data from MAZ data files is apportioned to the sq.km pixels.

3.8 Test Run Results

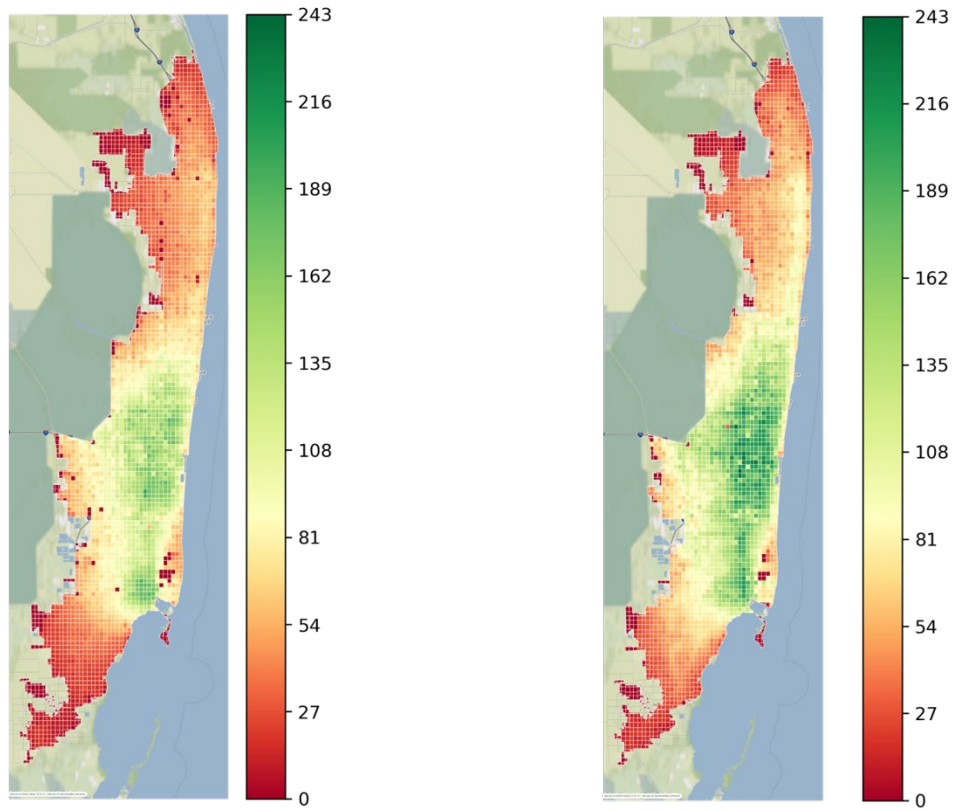
To verify the validity of the data collected for the SERPM area, the NREL team performed test runs using the data described from the previous sections and compared the results with the base run using TomTom network and CoStar⁴ land use data. CoStar data provides information on the primary activity pertaining to a majority of the buildings in the United States. The activity types available in the CoStar database are aggregated to the five activity categories noted in Table 8.

Figure 6 presents the calculated MEP scores for drive, walk and bike using the SERPM network and the TomTom network. Figure 7 shows correlation of the MEP scores by mode.

² <https://lehd.ces.census.gov/>

³ <https://www.census.gov/topics/population.html>

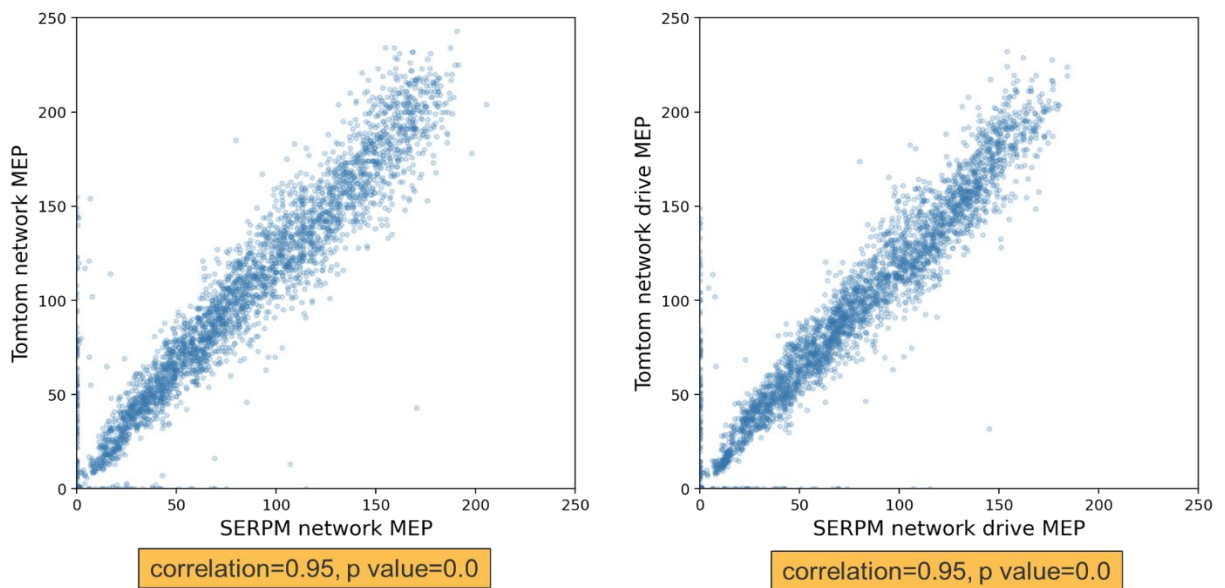
⁴ <http://www.costar.com>.



a) SERPM network MEP 103.01

b) TomTom network MEP 122.35

Figure 6 Calculated MEP scores based on SERPM network and TomTom network.



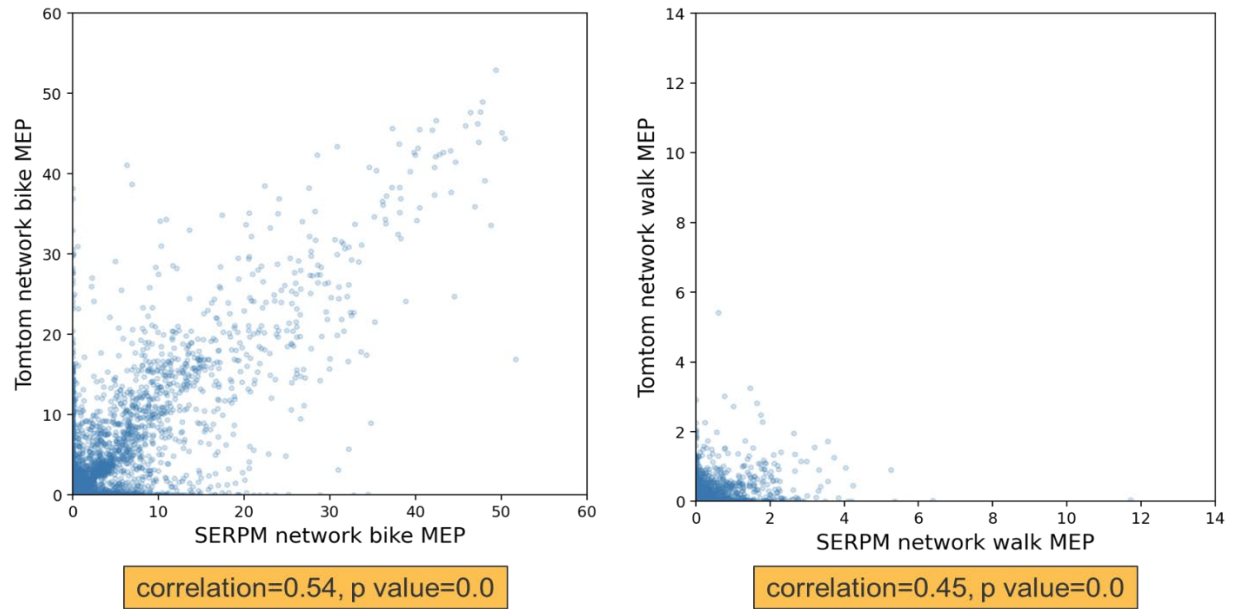


Figure 7 Comparison of MEP scores based on SERPM network and TomTom network.

The figures show that the MEP scores based on the SERPM network align closely with those based on the TomTom network, especially for drive and bike modes. The walk scores based on the SERPM network does not correlate well with those based on the TomTom network, which probably can be attributed to the sparser walk links in the SERPM network. Bike network is created from the master SERPM network by down selecting FRCs 2, 5, and 6, while walk network is created by selecting FRCs 5 and 6.

Figure 8 shows correlations between SERPM 8 MAZ data and CoStar data. It should be noted that the MAZ data represent number of employment by category while the CoStar data represent number of destinations. Although the magnitudes differ, Figure 7 shows that the patterns correlate very well.

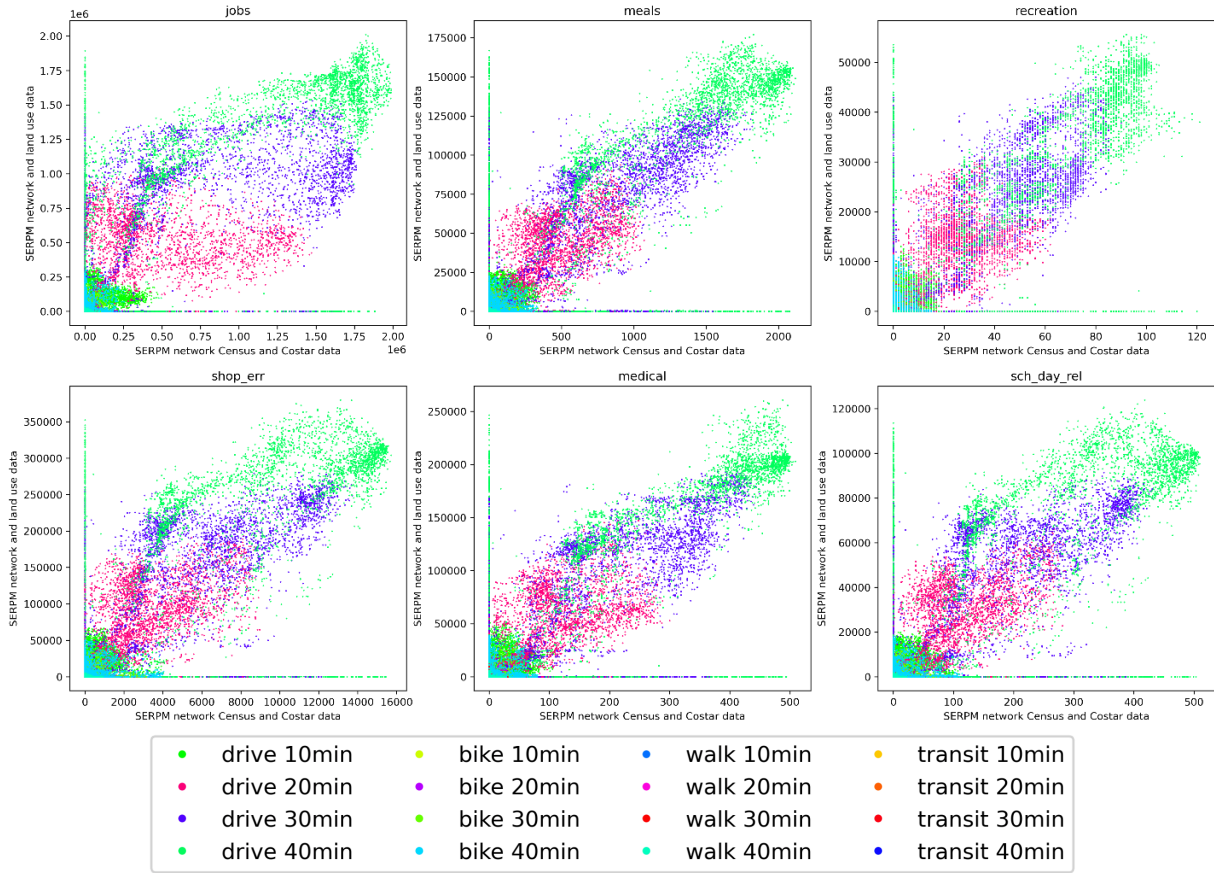
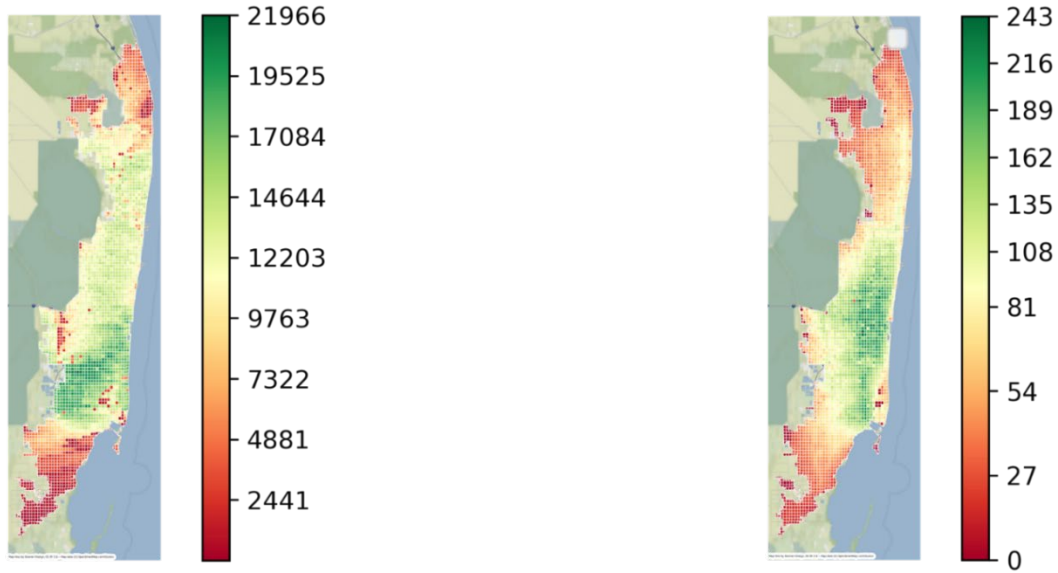


Figure 8 Correlation between SERPM 8 MAZ data and CoStar data.



a) SERPM network MEP 11983

b) TomTom network MEP 122.35

Figure 9 Comparison of MEP scores based on SERPM data and TomTom+CoStar data.

Figure 9 shows the MEP map based on SERPM 8 network and MAZ data in comparison to the MEP map based on TomTom network and CoStar data. Again, although the absolute scores come in different magnitudes, the patterns align reasonably well.

Figure 10 shows correlation of the MEP scores by mode for SERPM data and TomTom+CoStar data. From the final MEP calculation comparisons final calculation (one with all standard MEP data sources, and the other with all SERPM data sources) it can be noticed that the correlation (or the lack of it) is compounded from differences in network, as well as land use data sources. The correlations are particularly concerning for walk, and bike modes as they have sparser networks to begin with.

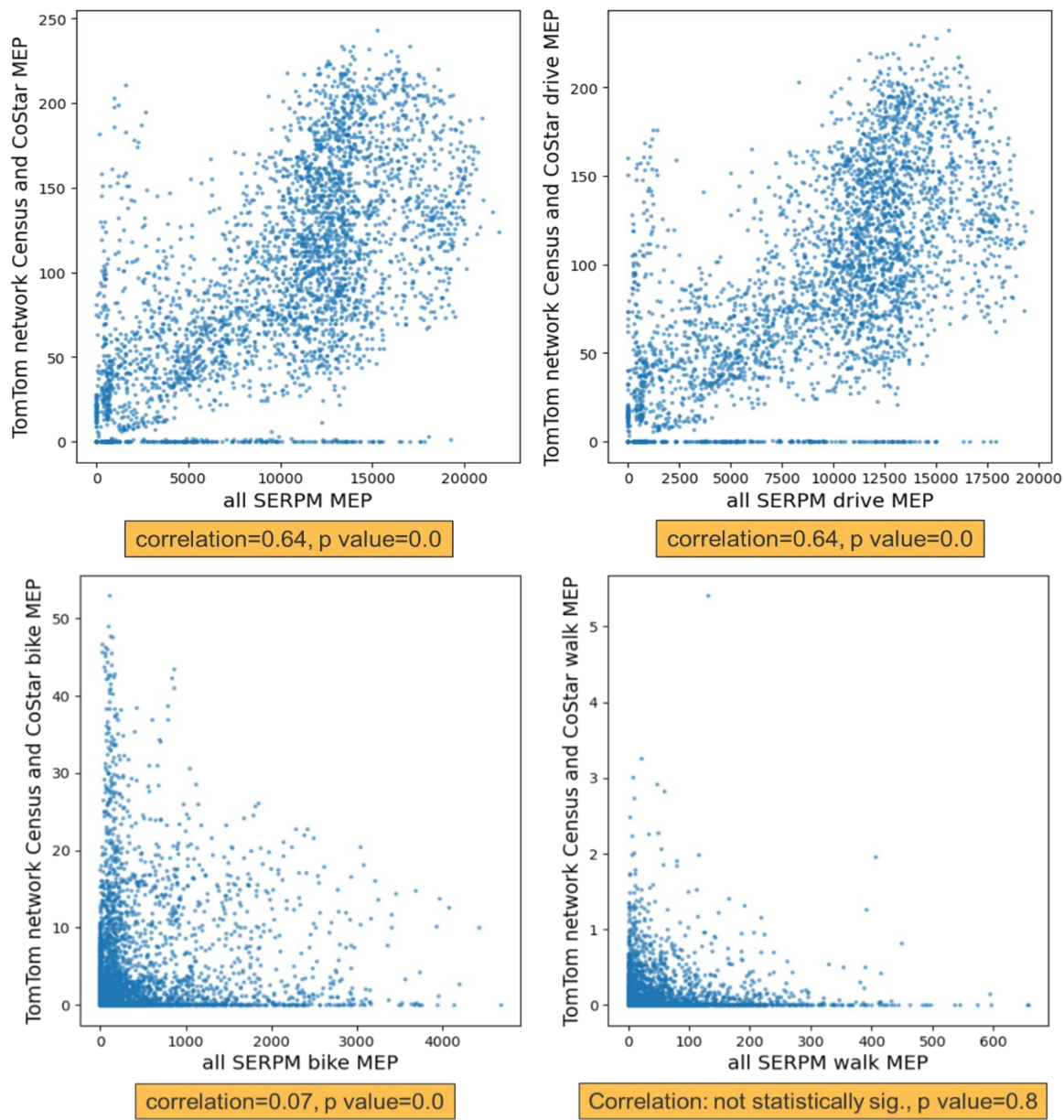


Figure 10 Comparison of MEP scores based on SERPM data and TomTom+CoStar data.

4 SCENARIO DESIGN

Two scenarios were created for the purpose of testing the MEP process. The first scenario focuses on improvements of bike facilities in South Florida. The second scenario investigates the effects of proposed transit projects with electric fleet. The goal with these two scenarios was to test different features within the MEP process. The bike scenario allows us to evaluate mobility enhancement given increased access through new infrastructure. The transit scenario evaluates the energy efficiency of electric buses. These two scenarios highlight the potential of the MEP tool to be used to analyze the effects of changes on the transportation system both in infrastructure and in service patterns when it comes to transit.

4.1 Bike Scenario

Three cases were created for the bike scenario to evaluate the enhancements in MEP scores, including:

1. **Base scenario:** 2015 base year model run, assuming all facilities with a functional class of 4 or lower are bikeable with a speed of 12 mph.
2. **2015 bike lane scenario:** 2015 base year condition, with an 18-mph bike speed for roadways identified as bicycle facilities based on the Florida Department of Transportation (FDOT) bicycle facility GIS map (FDOT, 2018). The higher bike speed is expected to result in a higher bike MEP score, simulating the improved access, safety and use of these bicycle facilities.
3. **2045 bike improvement scenario:** 2045 future year condition, with three bike improvement projects located in the City of Miami Beach, the City of Fort Lauderdale, and the City of West Palm Beach, respectively. This scenario is expected to show the impacts of bike facility improvements at project level.

The base scenario took the original SERPM 8 condition as is. The original network and speed output were used to compute the MEP scores.

The 2015 bike lane scenario entailed the establishment of a robust representation of the existing bike infrastructure within the South Florida region. For this purpose, the FDOT bike lane map from the FDOT's open data hub (FDOT, 2018), as depicted in Figure 11, was utilized. Due to the lack of clear distinctions between various types of bike infrastructure, including bike lanes, protected bike lanes, off-street paths, and sharrows, etc., all forms of bike infrastructure were included in this scenario and treated equally. These bike lanes were identified by overlaying the bike lane map with the SERPM 8 highway network. It assumes that a higher bike speed (18 mph) is achievable on these identified roadway segments, compared to the non-bike lanes (12 mph).

The 2045 bike improvement scenario intends to demonstrate the potential impacts of bike facility improvements, on top of the 2015 bike lane scenario. Compiling a complete future bike infrastructure plan for the entire South Florida region was not feasible. A further look into the

county level regional plans also showed that these plans lacked sufficient information at network level for inclusion in the scenario analysis. It is decided to focus on municipality level, which are the entities that are more focused on implementing bike infrastructure.

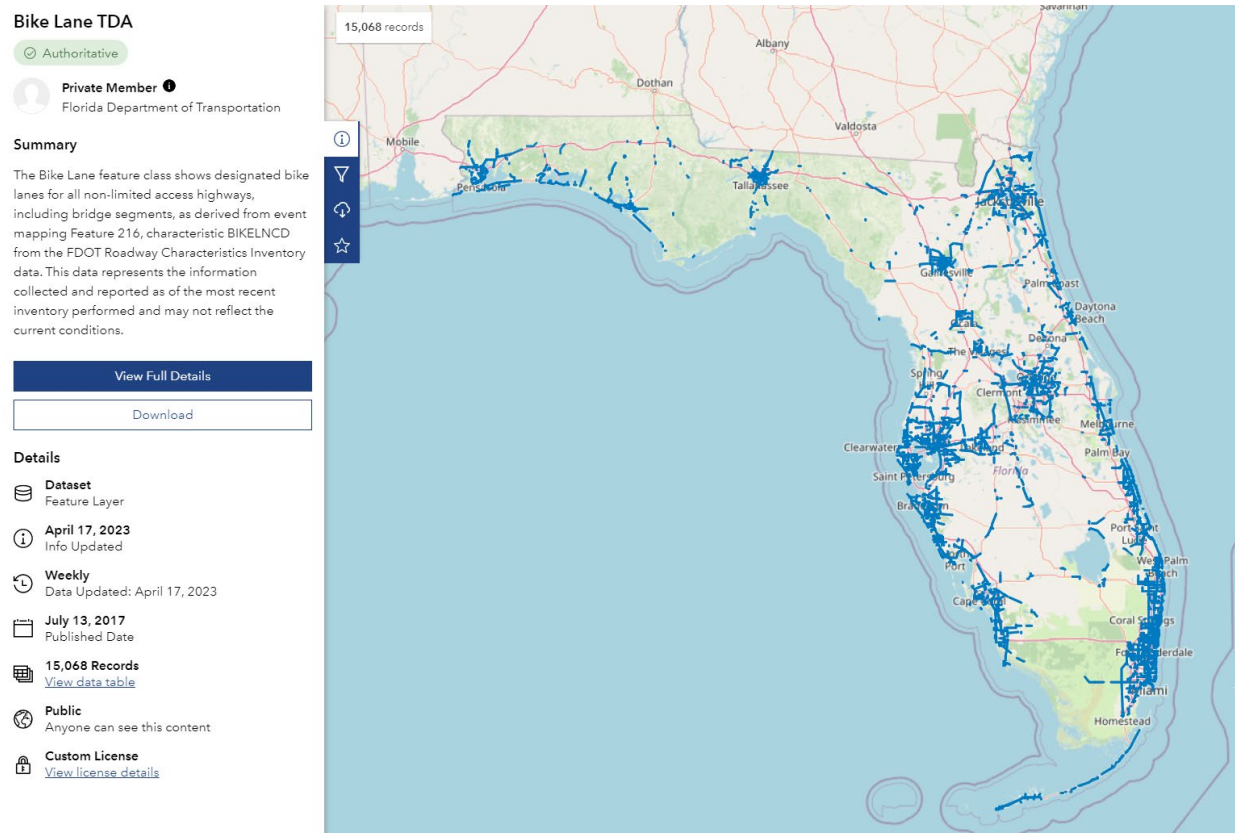


Figure 11 Map of bike lanes in Florida from FDOT roadway characteristics inventory.

Three municipalities (i.e., the City of Miami Beach, the City of Fort Lauderdale, and the City of West Palm Beach) were identified for this study. These municipalities were selected based on their potential to improve bike access and transportation, as disused later. They also represented diverse geographic locations within South Florida, with each municipality representing one of the three counties in the region.

City of Miami Beach

The City of Miami Beach exhibited a proactive response and furnished a future bike infrastructure map in the form of a shapefile, facilitating its integration into our scenario design. Notably, the bike projects within the city were systematically categorized based on infrastructure quality, encompassing distinct types such as Protected bike lanes, green painted bike lanes, off-street shared use paths, and neighborhood greenways, as illustrated in Figure 12. However, in maintaining fairness within the analysis of the future scenario, all these bike infrastructure improvements had to be treated equally, as the baseline scenario lacked comparable levels of detail.

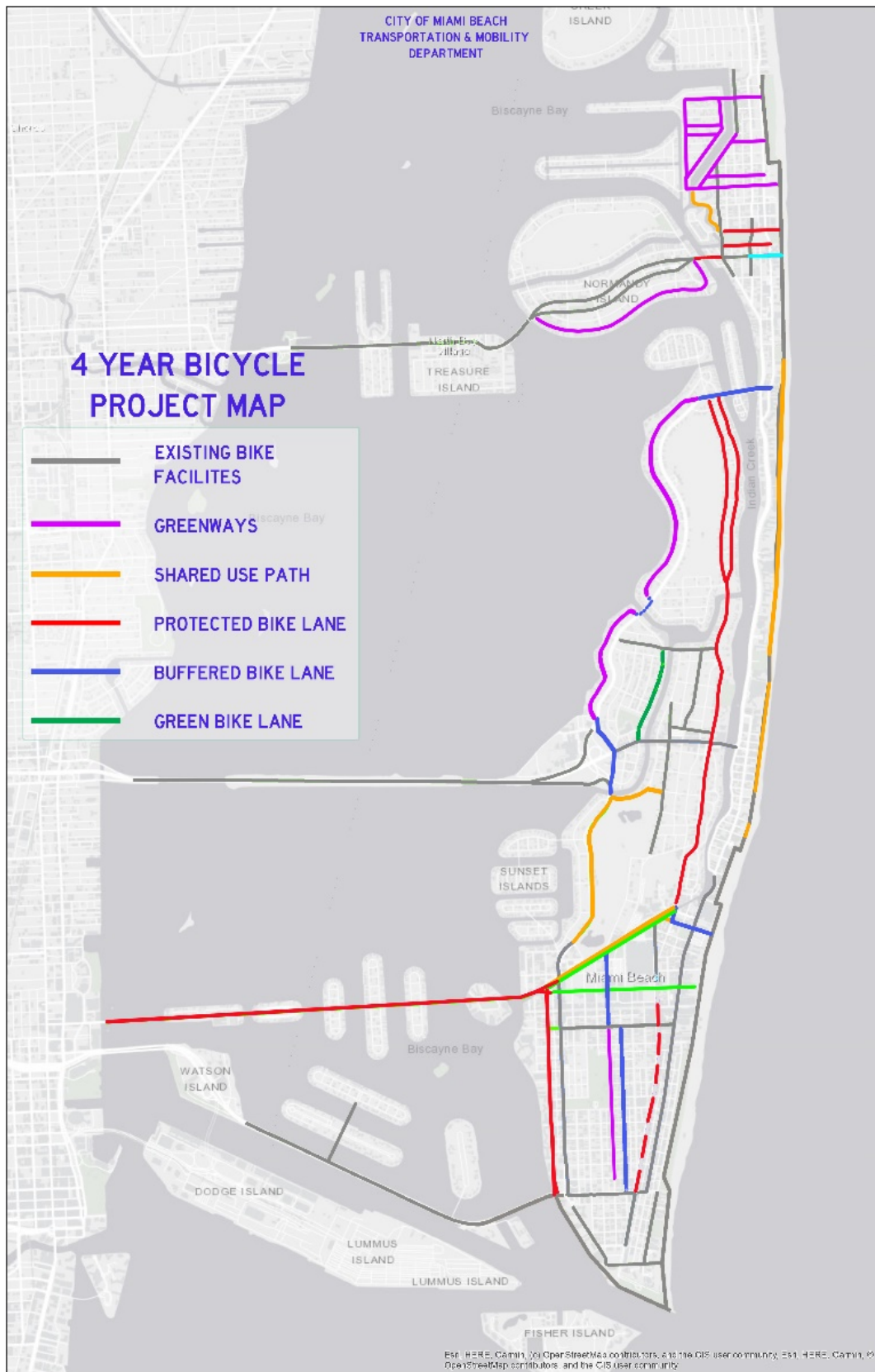


Figure 12 Miami Beach planned bicycle projects map.

City of Fort Lauderdale

The city's website contained comprehensive documentation regarding the forthcoming bike improvements planned for the area. Regrettably, the information was not available in a GIS file format but was presented solely as a PDF map, as depicted in Figure 13. Consequently, the bike lane links had to be manually chosen and adjusted on the SERPM network using the information provided in the PDF map.

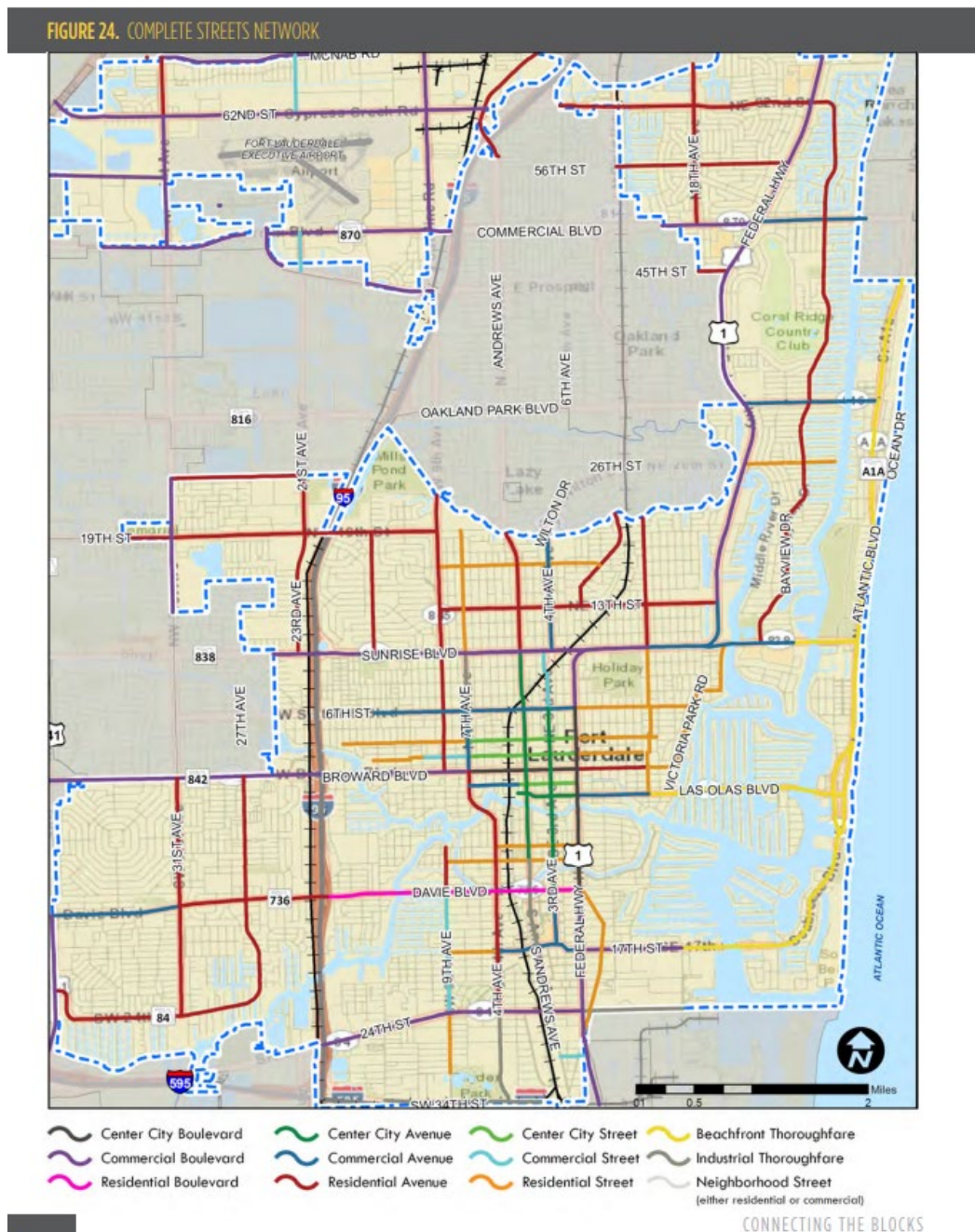


Figure 13 Fort Lauderdale Complete Streets Bike Network Plan

City of West Palm Beach

Similarly, the only accessible data were in the form of a PDF map for the City of West Palm Beach, as presented in Figure 14. A similar process to that undertaken for Fort Lauderdale was done to extract the relevant links within the SERPM network that were slated for upgrades to enhance bike facilities. All proposed multi-use trails, conventional bike lanes, bike boulevards and separated bike lanes were incorporated as future improvements.

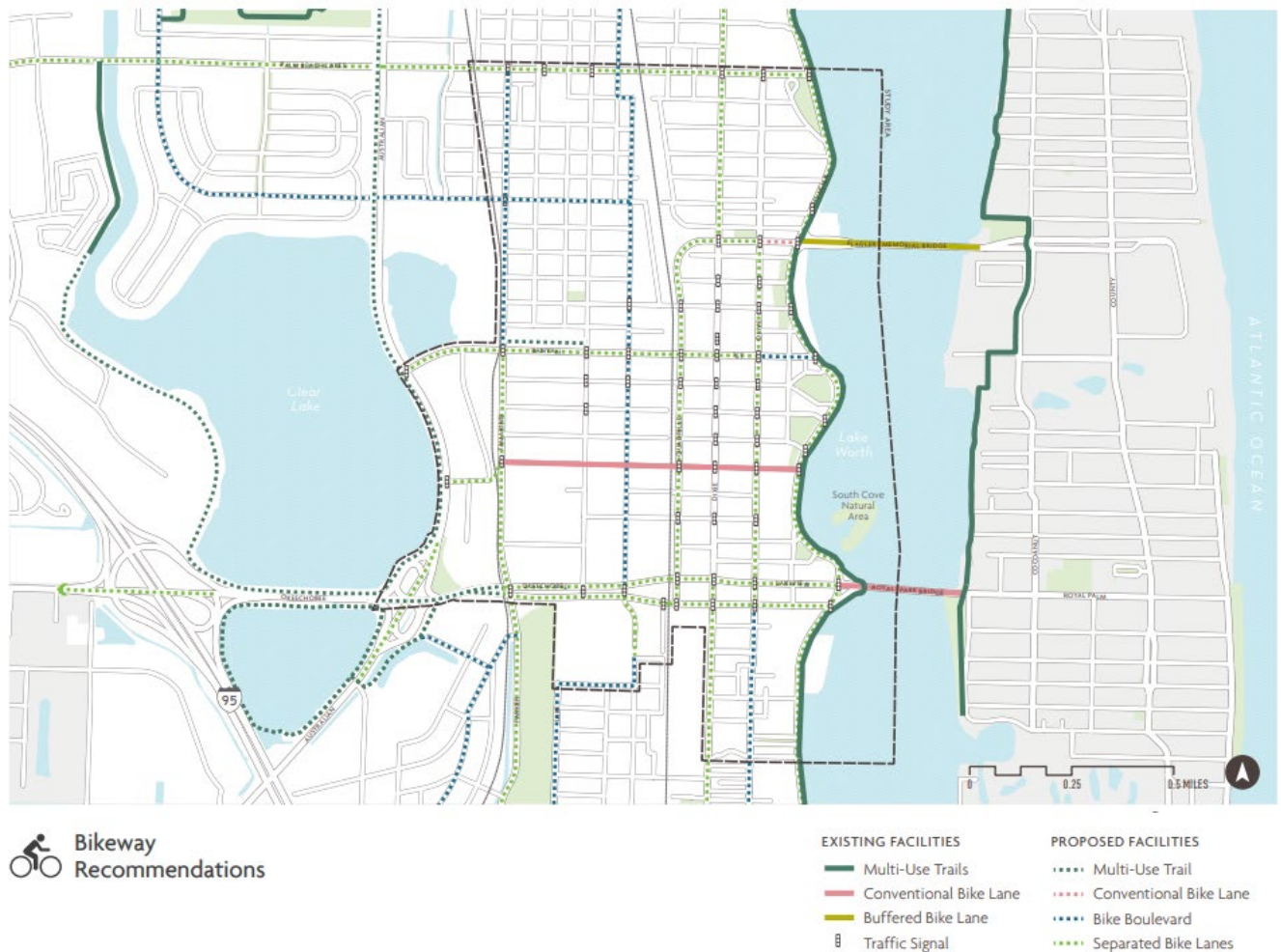


Figure 14 West Palm Beach Bikeway Recommendations

Figure 15 illustrates the 2015 base year bike lanes and the three locations with bike improvement projects in the year 2045. The improvements are localized due to constraints in getting information for the region, but this provides a great opportunity to see the potential of the MEP tool to analyze the effects of changes that are not system wide.

Base and 2045 Bikeable Links Visualization

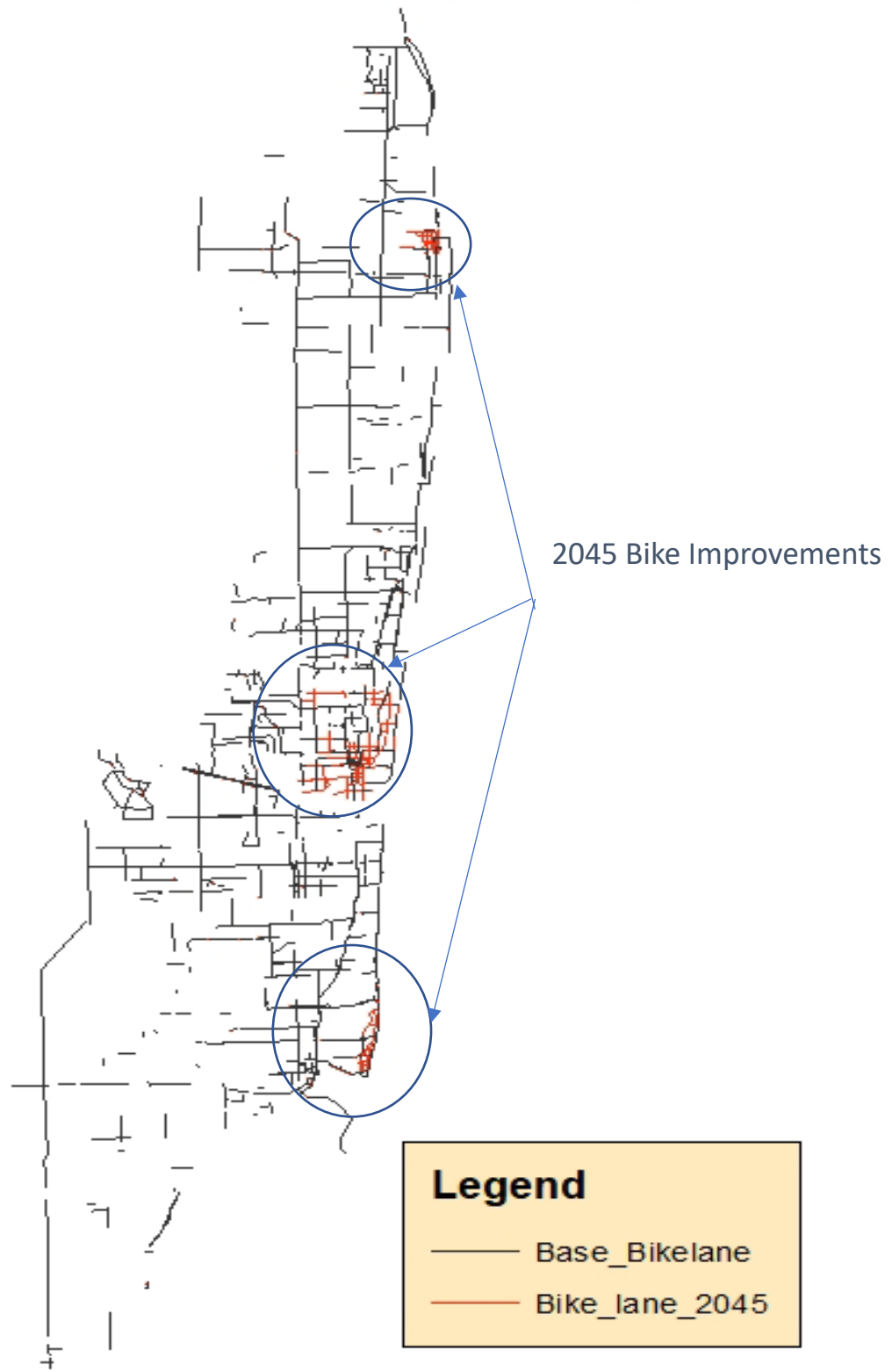


Figure 15 Map of baseline and 2045 scenario updates

4.2 Transit Scenario

Two cases were created for the transit scenario:

1. **2015 base scenario:** 2015 base year transit service condition, with default energy intensity and cost parameters.
2. **2045 future scenario:** 2045 future year transit service, assuming 100% adoption of electric buses with updated energy intensity and cost parameters.

Two major efforts were involved in this scenario design. The first effort focused on identifying reasonable energy intensity and cost parameters for electric bus fleet operations in the South Florida region. Information on energy consumption and costs for existing transit services were compiled based on available data from a Miami-Dade County TPO study that looked into the operating expenses for each of the municipal transit operators (Miami-Dade Metropolitan Planning Organization, 2016), and the tables from the FDOT transit information and performance handbook on the three major county level transit operators, namely Miami-Dade Transit, Broward Transit, and Palm Beach County Transportation Agency (FDOT, 2020).

Table 9 shows the costs for the municipal operators in South Florida including the operating expense per boarding and per service mile. Figure 16 presents the information for the county transit operators, which shows similar costs of about \$1 per passenger mile.

Table 9 Transit Operating Expenses for Miami Dade County Municipalities

Local Jurisdiction	Operating Expense per Service Mile	Operating Expense per Service Hour	Operating Expense per Boarding	Operating Expense per Resident
Bal Harbour	\$ 3.78	\$ 32.25	\$ 6.27	\$ 48.87
Bay Harbor Islands	\$ 4.70	\$ 40.76	\$ 3.73	\$ 17.44
Coral Gables	\$ 10.52	\$ 92.01	\$ 1.12	\$ 27.93
Cutler Bay	\$ 6.15	\$ 94.09	\$ 6.65	\$ 6.69
Doral	\$ 2.33	\$ 35.72	\$ 3.01	\$ 21.46
Hialeah	\$ 3.68	\$ 36.91	\$ 2.49	\$ 5.30
Homestead	\$ 9.79	\$ 70.18	\$ 3.31	\$ 6.87
Miami	\$ 5.07	\$ 34.02	\$ 1.27	\$ 11.46
Miami Beach	\$ 6.05	\$ 69.49	\$ 1.71	\$ 22.09
Miami Gardens	\$ 6.10	\$ 72.28	\$ 17.35	\$ 4.21
Miami Lakes	\$ 2.46	\$ 38.60	\$ 8.54	\$ 5.98
Miami Shores Village	\$ 4.06	\$ 51.00	\$ 5.50	\$ 10.78
Miami Springs	\$ 3.92	\$ 47.08	\$ 0.85	\$ 9.75
North Miami	\$ 6.06	\$ 48.63	\$ 1.23	\$ 9.25
Palmetto Bay	\$ 6.39	\$ 61.70	\$ 20.96	\$ 16.32
Pinecrest	\$ 8.14	\$ 55.95	\$ 7.02	\$ 9.57
Sunny Isles Beach	\$ 10.66	\$ 76.58	\$ 4.40	\$ 36.32
Surfside	\$ 6.83	\$ 63.81	N/A	\$ 31.34
Summary				
Min	\$ 2.33	\$ 32.25	\$ 0.85	\$ 4.21
Max	\$ 10.66	\$ 94.09	\$ 20.96	\$ 48.87
Mean	\$ 5.93	\$ 56.72	\$ 5.30	\$ 16.76
Percentiles				
25 th Percentile	\$ 3.89	\$ 38.17	\$ 1.26	\$ 6.83
75 th Percentile	\$ 7.15	\$ 70.70	\$ 6.83	\$ 23.55

Miami-Dade Transit (MDT)

Ms. Alice N. Bravo, P.E., Director
Overtown Transit Village
701 NW 1st Court, Suite 1700
Miami, FL 33136
(786) 469-5406
www.miamidade.gov/transit



Miami-Dade Transit is the largest transit agency in Florida, operating throughout Miami-Dade County, southern Broward County, and northern Monroe County. MDT is governed by Miami-Dade County, and is responsible for the management, construction, and operation of its services and facilities. Passengers have access to Metrobus, Metrorail (a 22-station heavy rail system), and Metromover (an automated people-mover system that serves the downtown Miami, Brickell and Omni areas). Vanpool services are also provided, and paratransit is available through MDT's Special Transportation Service. The data shown below represent all of MDT's fixed-route services and area vanpool services for 2018 and 2019.

	2018	2019
GENERAL INFORMATION	Service Area Population	2,496,435
	Service Area Population Density	8,158.3
	Operating Revenue	\$99,812,094
SERVICE SUPPLIED	Total Annual Revenue Miles	39,227,263
	Total Annual Revenue Hours	2,642,812
	Total Revenue Vehicles	1,205
	Peak Vehicles	1,021
	Route Miles	2,344.7
SERVICE USAGE	Annual Passenger Trips	80,197,205
	Annual Passenger Miles	489,407,036
	Average Trip Length	6.1
QUALITY OF SERVICE	Resident Access to Transit	92.01%
	Weekday Span of Service (hours)	24.0
COST EFFICIENCY	Operating Expense per Revenue Mile	\$12.69
	Operating Expense per Revenue Hour	\$188.29
	Operating Revenue per Operating Expense	20.06%
	Passenger Trips per Employee FTE	25.777
COST EFFECTIVENESS	Operating Expense per Passenger Trip	\$6.20
	Operating Expense per Passenger Mile	\$1.02
	Operating Expense per Capita	\$199.33
	Farebox Recovery Ratio	15.95%
	Average Fare	\$0.99
SERVICE EFFECTIVENESS	Passenger Trips per Revenue Mile	2.04
	Passenger Trips per Revenue Hour	30.35
	Passenger Trips per Capita	32.12
	Revenue Miles Between Safety Incidents	137,158
	Revenue Miles Between Failures	2,549

Broward County Transit (BCT)

Mr. Timothy Garling, Director
One N. University Drive, Suite 3100-A
Plantation, FL 33324
(954) 357-8300
www.broward.org/BCT



Broward County Transit is an agency of Broward County government responsible for operating a fixed-route motorbus service with several routes connecting with Palm Tran (in Palm Beach County), Miami-Dade Transit, and Tri-Rail (commuter rail service). In addition to fixed-route bus service, BCT also partners with many municipalities within the county to provide Community Bus services, including Fort Lauderdale's Sun Trolley, and offers contracted paratransit service. BCT and Broward Community Bus data for the years 2018 and 2019 are provided below.

	2018	2019
GENERAL INFORMATION	Service Area Population	1,935,878
	Service Area Population Density	4,721.7
	Operating Revenue	\$124,441,902
SERVICE SUPPLIED	Total Annual Revenue Miles	17,031,074
	Total Annual Revenue Hours	1,330,648
	Total Revenue Vehicles	436
	Peak Vehicles	355
	Route Miles	1,248.8
SERVICE USAGE	Annual Passenger Trips	29,934,418
	Annual Passenger Miles	137,942,555
	Average Trip Length	4.9
QUALITY OF SERVICE	Resident Access to Transit	94.86%
	Weekday Span of Service (hours)	20.2
COST EFFICIENCY	Operating Expense per Revenue Mile	\$7.31
	Operating Expense per Revenue Hour	\$93.52
	Operating Revenue per Operating Expense	33.89%
	Passenger Trips per Employee FTE	25.829
COST EFFECTIVENESS	Operating Expense per Passenger Trip	\$4.16
	Operating Expense per Passenger Mile	\$0.90
	Operating Expense per Capita	\$64.28
	Farebox Recovery Ratio	24.04%
	Average Fare	\$1.00
SERVICE EFFECTIVENESS	Passenger Trips per Revenue Mile	1.76
	Passenger Trips per Revenue Hour	22.50
	Passenger Trips per Capita	15.46
	Revenue Miles Between Safety Incidents	152,063
	Revenue Miles Between Failures	2,808

Palm Beach County Transportation Agency (Palm Tran)

Mr. Clinton B. Forbes, Executive Director
3201 Electronics Way
West Palm Beach, FL 33407
(561) 841-4200
www.palmtran.org



Palm Tran is a not-for-profit corporation owned by Palm Beach County that is governed by the seven-member Board of County Commissioners. The corporation provides transit services throughout Palm Beach County. Palm Tran operates fixed-route motorbus services and feeder bus services to the Tri-Rail system. In addition, Palm Tran offers route deviation and demand-response services via Palm Tran CONNECTION. 2018 and 2019 motorbus data for Palm Tran are provided below.

	2018	2019
GENERAL INFORMATION	Service Area Population	1,268,782
	Service Area Population Density	3,476.1
	Operating Revenue	\$61,568,830
SERVICE SUPPLIED	Total Annual Revenue Miles	7,310,660
	Total Annual Revenue Hours	454,023
	Total Revenue Vehicles	166
	Peak Vehicles	130
	Route Miles	1,110.0
SERVICE USAGE	Annual Passenger Trips	9,113,767
	Annual Passenger Miles	51,698,218
	Average Trip Length	5.7
QUALITY OF SERVICE	Resident Access to Transit	63.23%
	Weekday Span of Service (hours)	17.8
COST EFFICIENCY	Operating Expense per Revenue Mile	\$8.42
	Operating Expense per Revenue Hour	\$135.61
	Operating Revenue per Operating Expense	16.24%
	Passenger Trips per Employee FTE	18.695
COST EFFECTIVENESS	Operating Expense per Passenger Trip	\$6.76
	Operating Expense per Passenger Mile	\$1.19
	Operating Expense per Capita	\$48.53
	Farebox Recovery Ratio	14.34%
	Average Fare	\$0.97
SERVICE EFFECTIVENESS	Passenger Trips per Revenue Mile	1.25
	Passenger Trips per Revenue Hour	20.07
	Passenger Trips per Capita	7.18
	Revenue Miles Between Safety Incidents	128,257
	Revenue Miles Between Failures	3,045

Figure 16 Overview of three major transit systems in South Florida.

The default cost value in the MEP tool was \$0.85 per passenger-mile as shown in Table 10, slightly lower than the existing cost values in the three counties. Considering inflation, we decided to use the default values for the base year scenario.

For the 2045 scenario, all three operators had well established Bus Fleet Electrification plans (Broward County Transit, 2019; Southeast Florida Regional Climate Change Compact, 2023; T - Federal Transit Administration, 2022), and all three counties are expected to have 100% electric bus fleets by 2045, therefore, the reduction in the energy intensity and operation cost resulting from the electrification of the bus fleets need to be accounted for. To determine the values for the 2045 scenario, a literature review was conducted. A study found that electric buses get about 13.3 mpgde (miles per gallon diesel equivalent) on average, while diesel busses get about 5.1 mpgd (Eudy & Jeffers, 2018). This would make electric buses $13.3/5.1 = 2.61$ times more efficient than diesel buses. Given the default energy intensity of 0.65 kWh/passenger mile, this would give electric buses an energy efficiency of $0.65/2.61 = 0.25$ kWh/passenger mile. In terms of cost efficiency, another NREL study found a cost reduction of 8.3% over the course of a 12-year period with a net present value of 780,000 when evaluating a 4-bus fleet (Johnson et al., 2020). Accordingly, we proposed a value of \$0.78 per passenger-mile for the 2045 scenario, as shown in Table 10

Table 10 Energy Intensity and Cost by Mode

	2015 Baseline (Using Default NREL values)		2045 Future Scenario	
Mode	Energy intensity (kWh/passenger-mile)	Capital and operational cost (dollar/passenger-mile)	Energy intensity (kWh/passenger-mile)	Capital and operational cost (dollar/passenger-mile)
Driving	0.90	0.48	0.90	0.48
Transit	0.65	0.85	0.25	0.78
Bike	0	0	0	0
Walk	0	0	0	0
Transportation Network Company	1.8	1.54	1.8	1.54
Paratransit	4.13	2.25	4.13	2.25

The second major effort in this scenario involves converting the regional transit network to GTFS format, which is required by the MEP process. Since the transit network used in regional models generally lacks operational level details, especially for future years, direct conversion from regional network to GTFS format is not possible. For the purpose of this study, the NREL team developed a procedure to perform the conversion, which is shown in the appendix.

5 MODEL STEPS

Running the MEP process involves two main phases. The first phase compiles and reformats the data into the proper formats required for the MEP code to run. The second phase runs the MEP code through a docker container. This section lists the main steps.

5.1 Preprocessing Data

The MEP code requires the input data to be formatted in a specific way. The details for the preprocessing are described in Appendix A. The main steps include:

1. Install necessary software:
 - a. Anaconda navigator
 - b. Python
 - c. Jupyter Notebook
 - d. Spyder
 - e. Install necessary Python Packages to be able to run the prepackaging code
2. Prepare TAZs (transportation analysis zones)
3. Prepare roadway network
4. Prepare Bike Network
5. Prepare Transit Network

The objective of each of these steps is to transform the inputs into the format required by the MEP code. For link networks such as the roadway and bike network, the output should be in .gpkg format (Geopackage). For the transit network, the data should be in GTFS format. For the TAZ the data from csv and shp files should also be combined into a geopackage.

5.2 Installing Docker and MEP Code

The details for the installation and running the MEP code are described in Appendix B. The main steps include:

1. Setting up Docker:
 - a. Install Docker Desktop if not already installed.
 - b. Start Docker Desktop to initialize the Docker daemon.
2. Unzip the provided file:
 - a. Extract the UrbanMEP.zip file to a directory on the computer.
 - b. The extracted files include configuration files, data processing scripts, database scripts, Docker-related files, and R dependencies script.
3. Build Docker image:
 - a. Open a terminal and navigate to the UrbanMEP directory.

- b. Run a Docker build command to build the MEP image using the provided Dockerfile.
- 4. Drive, Bike, and Walk Data:
 - a. Update relevant parameters in the configuration file for drive, bike, and walk modes.
 - b. Provide a formatted network file for the drive, bike, and/or walk MEP calculation.
- 5. Transit Data:
 - a. Create a folder for GTFS files.
 - b. Update relevant parameters in the configuration file for transit mode.
 - c. Download or create GTFS data for the desired city/scenario and place it in the GTFS folder.
 - d. Download the state-level OSM data and put it in the data folder.
- 6. Docker Container and Database Configuration:
 - a. Run the Docker container with recommended arguments to set up shared volumes, internal PostgreSQL database, and interact with configuration parameters and data.
- 7. Run MEP code
 - a. Create Isochrones for bike/walk/drive data
 - b. Run opportunity counts computation
 - c. Compute MEP results

6 Scenario Results

6.1 Bike Scenario Results

This section presents the results from the three scenarios: the base 2015 scenario with no bike lanes, the base 2015 bike lane scenario, and the 2045 future scenario with select bike improvements from three municipalities. Figure 17 shows the additional bike improvements between the second and the third scenarios. Figure 18 shows the bike networks in the three scenarios, respectively. As discussed earlier, the locations selected for bike improvement were municipalities that have a high density of employment opportunities where an improvement to the cycling network could greatly improve access for the cycling mode and therefore show a significant improvement in the MEP score. One municipality was selected from each county in South Florida to have a broad geographic understanding of the impacts.



Figure 17 Case study locations for bike improvements.

- For links that are **marked as bikeable** in the hybrid bike network, a bike **speed of 18mph** is assumed.
- For links that are **not marked as bikeable** but belong to **functional class ≥ 5** , a **speed of 12 mph** is assumed.

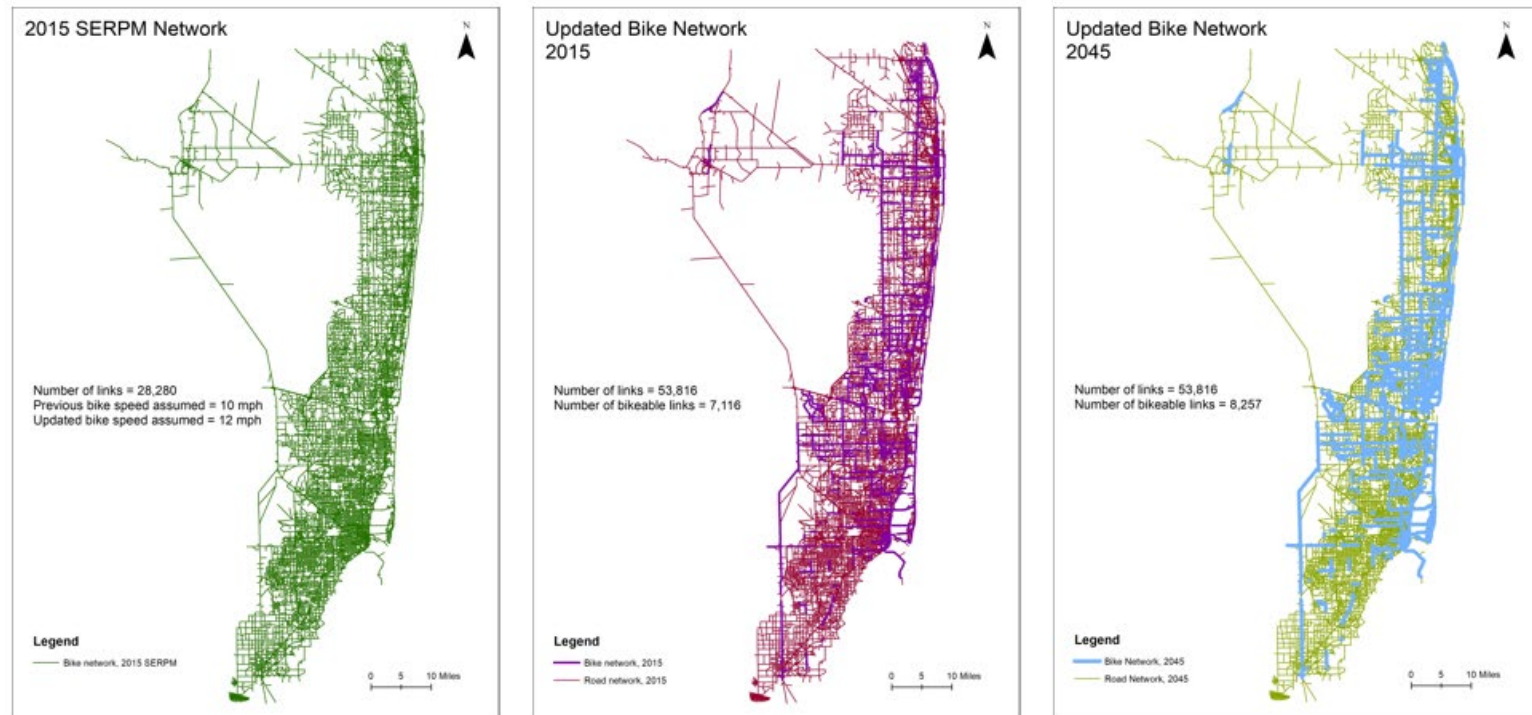


Figure 18 Comparison between SERPM network and bike improvements.

Table 11 presents the bike MEP scores for the three scenarios. A significant improvement in MEP scores was found between the baseline without any bicycle facilities and the updated 2015 bike network where the FDOT current bike network is added. However, the 2045 scenario showed only a small increase (3.15%) in the MEP score. This is reasonable given that bike improvement projects at local level would have limited impacts on the regional level MEP score.

Table 11 Bike MEP Scores

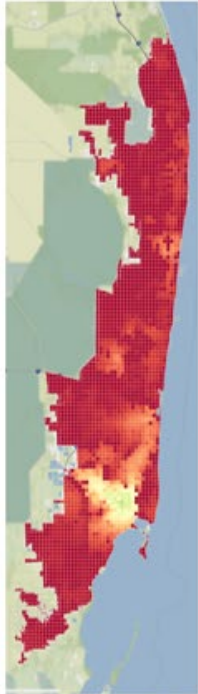
Scenarios	Bike MEP Score	% Improvement Compared to 2015 Baseline	% Improvement Compared to 2015 Bike Lane
2015 Baseline	1241.84	-	-
2015 Bike Lane Scenario	2995.11	141.18%	-
2045 Bike Improvement Scenario	3089.36	148.77%	3.15%

Figure 19 shows the MEP maps for the three scenarios. As expected, most areas have low bike MEP scores (shown in red) except where bike facilities are present, especially in high density areas. To better visualize the differences between the scenarios, Figure 20 presents the maps showing the differences in MEP scores between the 2015 baseline scenario and the 2015 bike lane scenario (on the left) and between the 2015 bike lane scenario and the 2045 bike improvement scenario (on the right). As expected, the map on the left shows overall MEP improvements across the entire area and a concentrated change in the downtown area. However, the map on the right indicates a slight decrease in MEP scores across the board in addition to the MEP improvements in the locations with bike improvement projects. Further investigation is needed to identify potential causes of the across-the-board changes in the MEP scores between the 2045 scenario and the 2015 bike lane scenario.

Figure 21 shows the correlation plots of the bike MEP scores between the 2015 baseline scenario and the 2015 bike lane scenario (on the left) and between the 2015 bike lane scenario and the 2045 bike improvement scenario (on the right). A strong correlation would indicate little change between the two scenarios. As the figure shows, there is a low correlation in the graph on the left, which indicates significant changes between the 2015 baseline and the 2015 bike lane scenarios. The exceptionally high correlation in the graph on the right (correlation score is 0.99) indicates minimal changes upon incorporating the local bike network improvements, as expected.

Although the bike improvement projects showed minimal impacts at regional level, a further analysis of the impacts at local scale showed significant improvements in the MEP scores, as shown in Figure 22. Specifically, there is a 25% rise in the northern corridor, a 60% surge in the central corridor, and a 20% increase in the southern corridor. This stands in contrast to the overall increase of 3%. This shows that improvements in the bike network have a localized effect rather than being uniformly distributed across the entire network.

'Previous' 2015 Bike Network with Updated Speed of 12mph



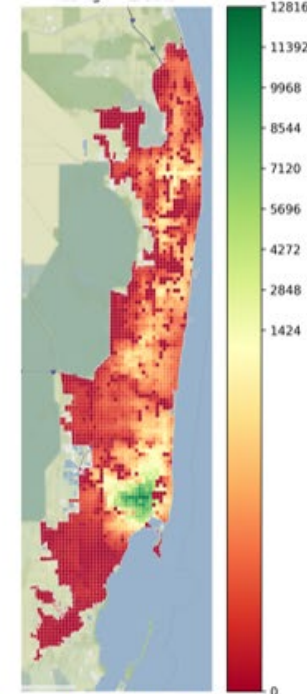
Overall MEP Scores

'Updated' 2015 Bike network: For links marked as Bikeable, a **speed of 18 mph** is assumed, and for non-bikeable network which have $FC \geq 5$, a **speed of 12 mph** is assumed



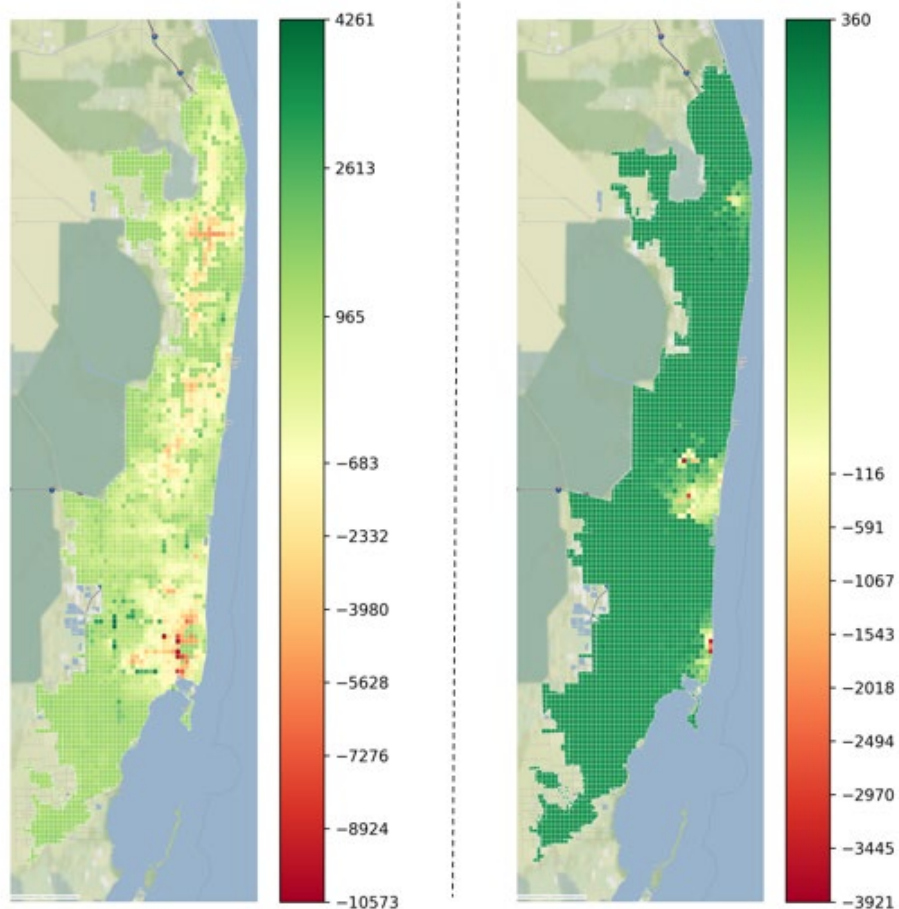
Overall MEP Scores

Updated Bike network with addition of bikeable lanes in three locations



Overall MEP Scores

Figure 19 Bike MEP score comparison between baseline and future network



Difference in MEP Score = Previous 2015 Baseline
- Updated Baseline

Difference in MEP Score = Updated Baseline -
2045

Figure 20 Difference in MEP score between baseline and future network SERPM LU

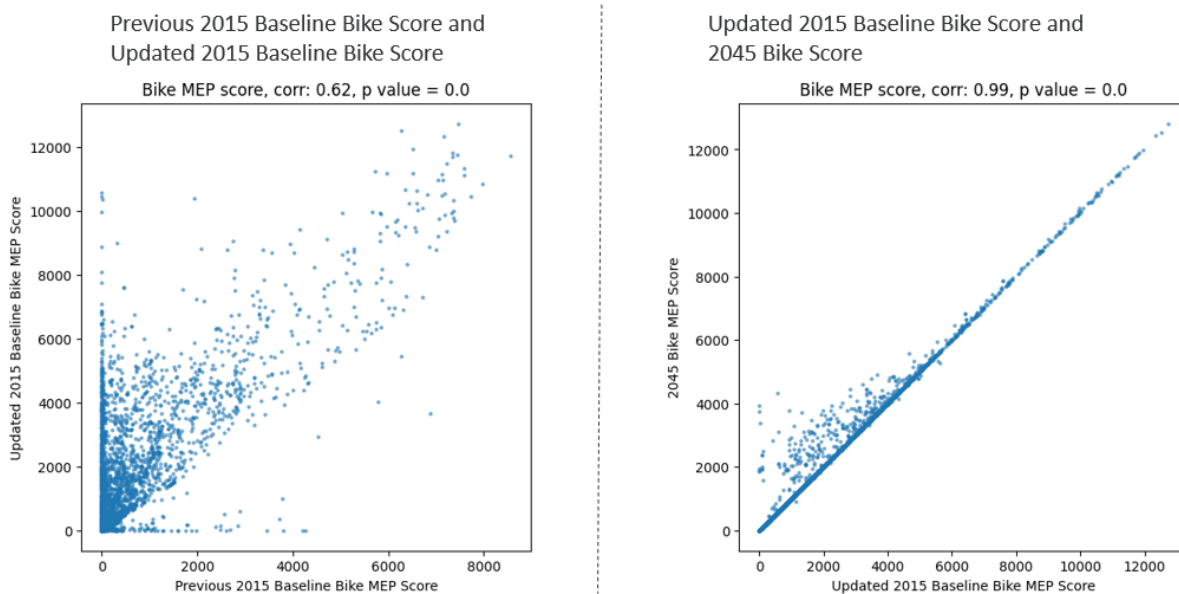
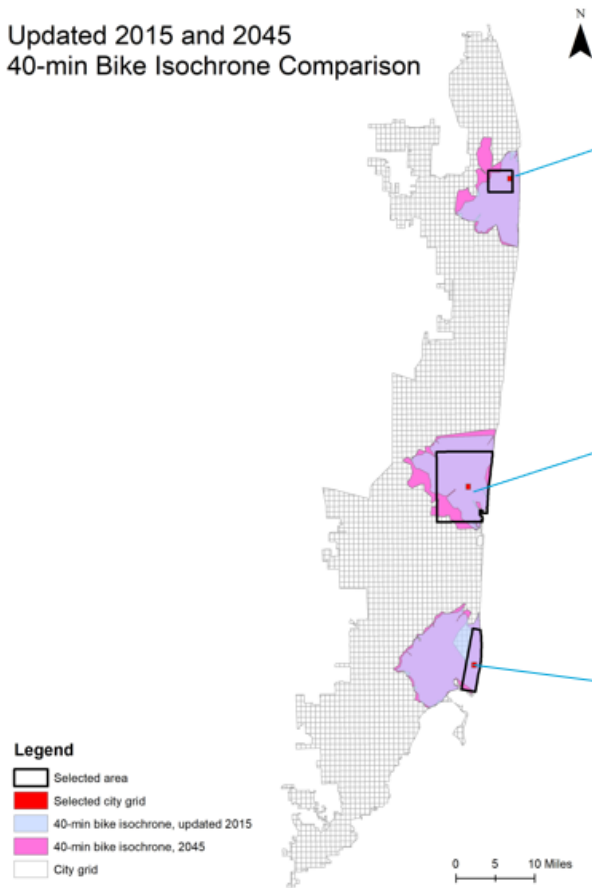


Figure 21 Correlation matrix for MEP score baseline and future scenario SERPM LU

Updated 2015 and 2045
40-min Bike Isochrone Comparison



ID = 470	Bike MEP Score	% Improvement Compared to 'Previous' Baseline	% Improvement Compared to 'Updated' Baseline
2015 'Previous' Baseline	1208.77	-	-
2015 'Updated' Baseline	3017.13	149.6%	-
2045 Scenario	3786.01	213.2%	25.5%

ID = 1664	Bike MEP Score	% Improvement Compared to 'Previous' Baseline	% Improvement Compared to 'Updated' Baseline
2015 'Previous' Baseline	971.77	-	-
2015 'Updated' Baseline	2203.77	126.8%	-
2045 Scenario	3507.72	261.0%	59.2%

ID = 2658	Bike MEP Score	% Improvement Compared to 'Previous' Baseline	% Improvement Compared to 'Updated' Baseline
2015 'Previous' Baseline	910.78	-	-
2015 'Updated' Baseline	4656.77	411.3%	-
2045 Scenario	5576.60	512.3%	19.8%

Figure 22 Case study inspection on specific TAZs to investigate local effects on MEP score

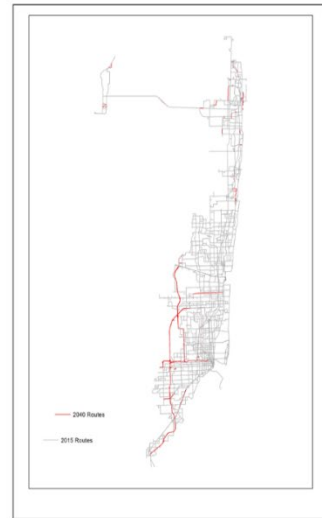
6.2 Transit Scenario Results

Figure 23 shows a comparison of the 2015 transit network and the 2045 transit network with future and proposed extensions. The major expansions are with express services along major freeways in Miami-Dade County.

In 2015, there were about 525 links in total, encompassing 7216.52 miles

In 2045, there are 575 links in total, encompassing 7977.36 miles; 10.54% increase in transit lane miles compared to 2015

Proposed Extension in 2045



In 2015, there were 24,019 stops in total

In 2045, there were 24,180 stops in total

Additional Stops in 2045

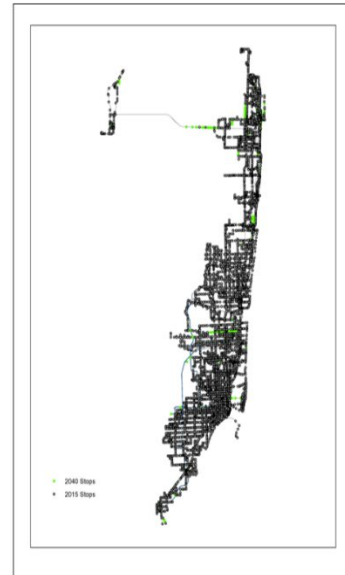
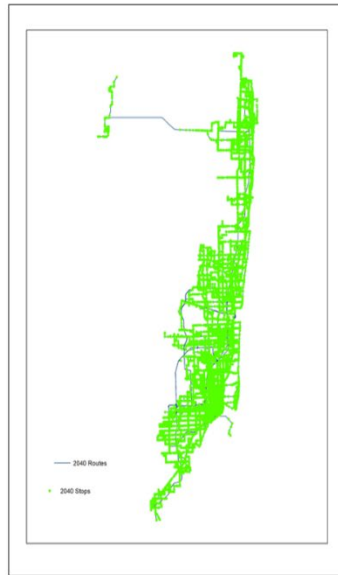


Figure 23 2015 transit network and future extensions for 2045

To understand how isochrones form around transit services, a litmus test was conducted, as shown in Figure 24. In this small test a single square on the MEP score grid, the 10, 20, 30, and 40-minute isochrones were visualized. The maps show how the shape of the isochrone for transit is composed of circular blobs surrounding the nearest transit stops indicating how far passengers may walk to/from the stops. This is different from the bike isochrone which is a contiguous shape.

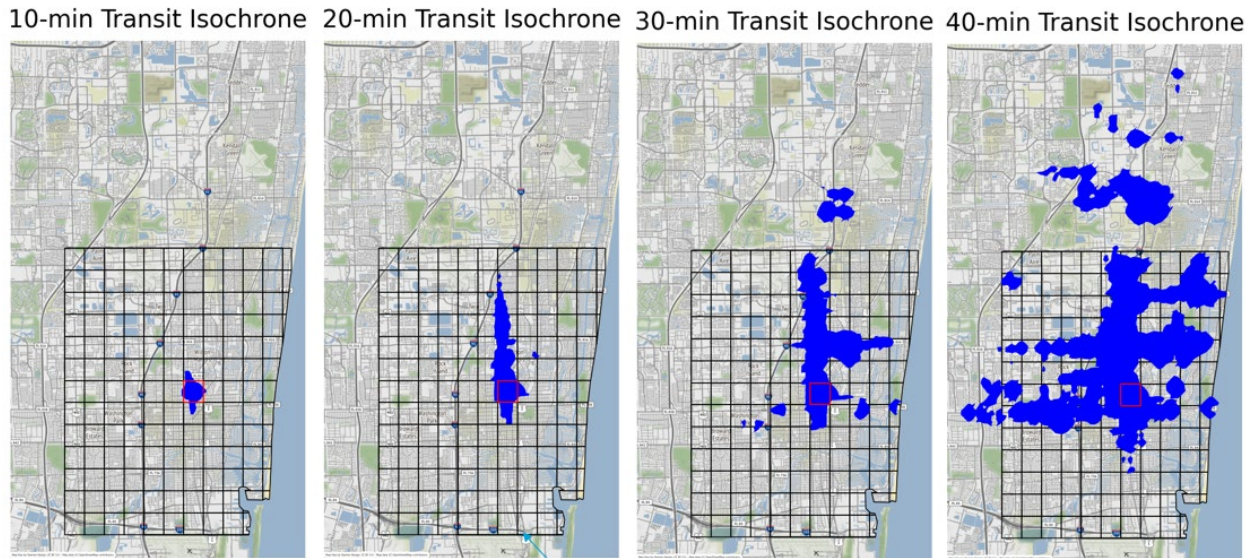


Figure 24 Litmus test of single point

Figure 25 presents the MEP map for the entire study area. The map shows that most of South Florida has very poor transit access resulting in a very low MEP score except for a few selected spots that are well served by transit such as the downtown core areas, and along the Tri-rail corridor. The MEP transit map appears to have significant noise as well, or spots with MEP scores don't entirely make sense based on the existing transit service in the area. This should be further investigated if the MEP tool is to be used.

Figure 26 presents the MEP maps for the 2015 base year scenario and 2045 future year scenarios. Substantial changes can be observed, particularly in areas with higher MEP scores. However, when considering the entire map, there is a similarity between the two scenarios, with downtown cores in Miami, Broward, and Palm Beach County displaying higher transit scores, while suburban areas exhibit lower scores. When comparing the 2045 baseline scenario with the 2045 scenario incorporating energy intensity and operational costs, minimal differences are observed in locations with higher MEP scores. However, there is a variation in the intensity of the MEP scores, as anticipated. It should be noted that energy intensity and operational costs does not impact the spatial pattern of MEP scores; instead, it amplifies the MEP score by accounting for energy and cost savings.

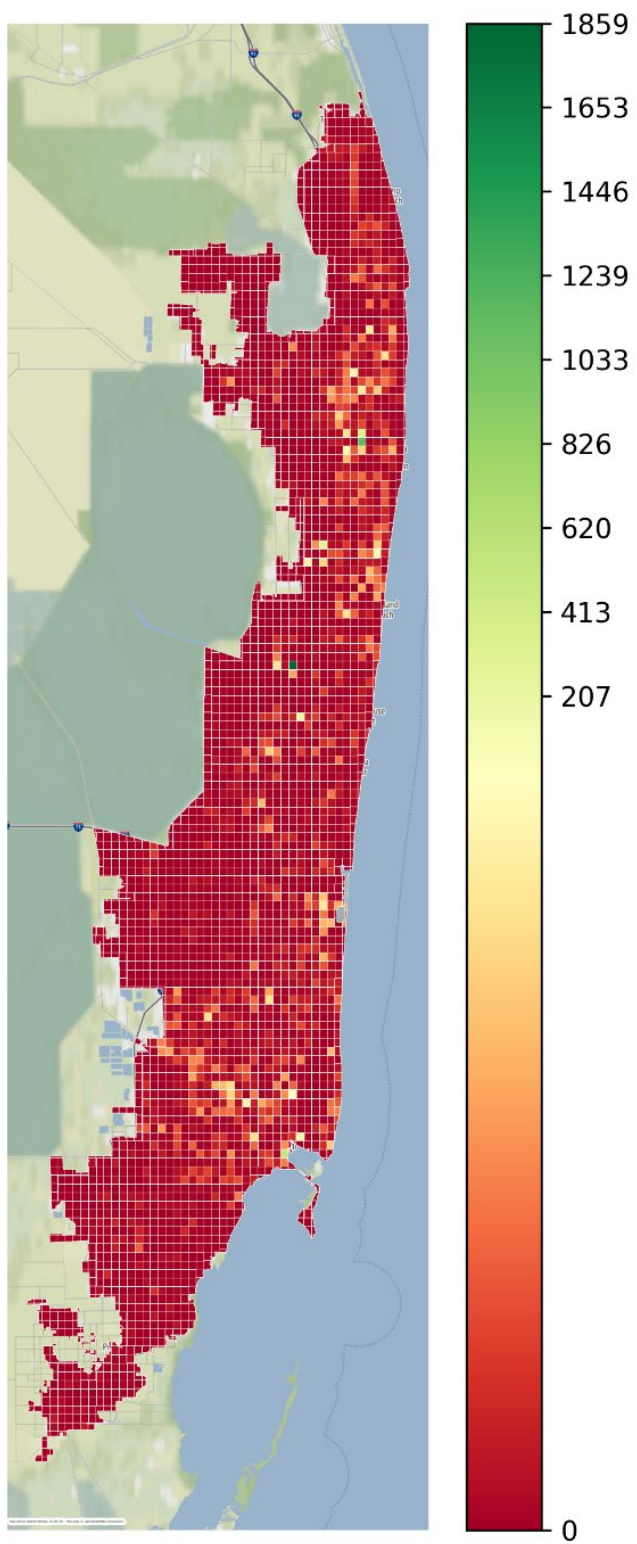


Figure 25 Transit MEP score map

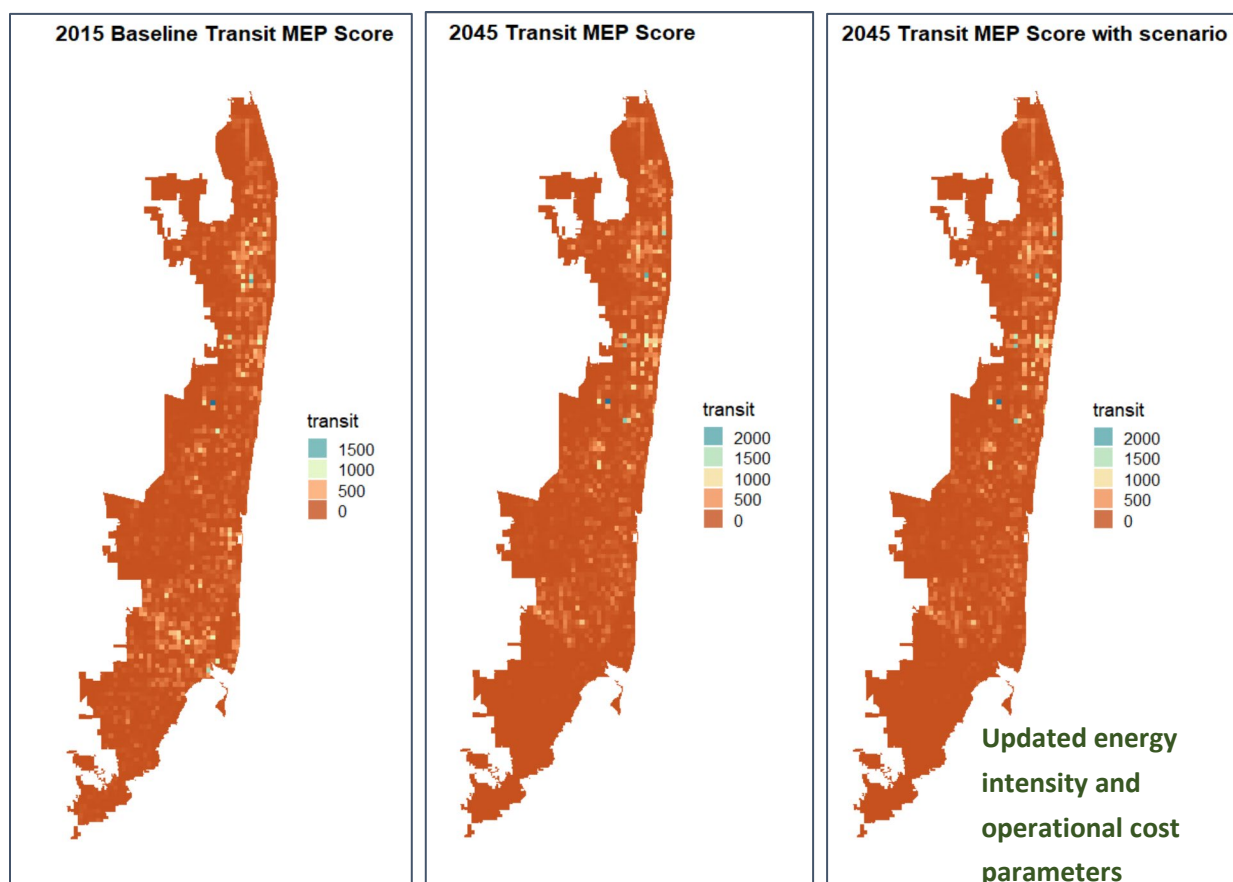


Figure 26 Transit MEP scores for 2015 and 2045 scenarios.

Table 12 shows the MEP scores for the scenarios. The 2045 scenario showed a 2.52% increase in the MEP score compared to the 2015 base scenario. This may indicate some flaws in either the type of expansion in the transit system for 2045 or a flaw in how the MEP score was calculated. The 2045 scenario with adjusted energy intensity and operational cost showed significant improvement in the MEP score.

Table 12 Transit Overall MEP score

Scenarios	Transit MEP Score	% Change Compared to 2015 Baseline	% Change Compared to 2045 Baseline Scenario
2015 Baseline	119.37	-	-
2045 Scenario Baseline	122.28	2.52	-
2045 Scenario with updated 'energy intensity and operation cost'	155.06	29.89	26.81

7 CONCLUSIONS

This project intends to evaluate the MEP process in terms of the feasibility of the tool and soundness of the results. NREL provided instructions on the data preparation process and the installation process. The instructions are generally helpful with some unforeseen issues that require localized troubleshooting.

Two groups of scenarios were designed to test the capabilities of the MEP process. The bike scenario showed reasonable results and illustrated the impacts of bike network improvements at regional level as well as local scale. This demonstrates the sensitivity of the MEP tool to capture localized effects. Analysis of the transit scenario also showed that the MEP tool can reasonably reflect the mobility and accessibility produced by transit services. Parameter changes on energy intensity and operation cost also showed expected impacts on the MEP scores. However, both scenarios showed some unexpected results, which require further investigation. Unfortunately, due to the limited time available, we were not able to dive further.

Further analysis is needed to fully understand all steps involved in the MEP calculation process and fully explore the capabilities of the tool, which would require incremental or isolated changes of certain inputs as well as some trial-and-error analysis, for example, to identify whether the resulting changes are due to nuances in the input data or due to some variations introduced in the MEP process.

More complex scenario design could also be beneficial to fully understand the capabilities of the tool. For example, categorize transit projects by group and compare the differences in the resulting MEP improvements, or compare alternative designs of service improvements in terms of their impacts on the MEP score.

The research team was not able to fully test the MEP tool partially due to the obstacles encountered when preparing and installing the tool, and the time spent on troubleshooting the issues. Hopefully, better instructions and guidelines could be prepared for future applications. A holistic tool environment with user interface instead of requiring the running of a sequence of codes would also be a tremendous improvement to facilitate the use of this tool.

8 References

- Automotive Leasing Guide (ALG). 2016. *The Road to 2030: Vehicle Production and Sales in the Autonomous Era*. Santa Monica, CA: ALG.
- American Automobile Association (AAA). 2018. *Your Driving Costs: How Much are You Really Paying to Drive (2018 Edition)*. Heathrow, FL: AAA Association Communication.
- American Automobile Association. 2022 *Your Driving Costs*.
<https://newsroom.aaa.com/auto/your-driving-costs/>.
- Brogan, James J., Aeppli, Andreas E., Beagan, Daniel F., Brown, Austin, Fischer, Michael J., Grenzeback, Lance R., McKenzie, Elaine, Vimmerstedt, Laura, Vyas, Anant D., & Witzke, Erika. Transportation Energy Futures Series: Freight Transportation Modal Shares: Scenarios for a Low-Carbon Future. *United States*. <https://doi.org/10.2172/1072829>
- Broward County Transit. (2019). *Electric Bus Program Implementation*. Plantation, FL: *Broward County Transit*
- Bureau of Transportation Statistics. (2019) U.S. Ton-Miles of Freight.
<https://www.bts.gov/content/us-ton-miles-freight>.
- Davis, Stacy C., Susan E. Williams, and Robert G. Boundy. 2017. *Transportation Energy Data Book: Edition 36*. Oak Ridge, TN: Oak Ridge National Laboratory. ORNL/TM-2017/513.
- Eudy, Leslie and Matthew Jeffers. 2018. *Zero-Emission Bus Evaluation Results: County Connection Battery Electric Buses*. Golden, CO: National Renewable Energy Laboratory. NREL/TP-5400-72864. <https://www.nrel.gov/docs/fy19osti/72864.pdf>
- FDOT. (2018, October 5). *FDOT Bike Routes | FDOT Bike Routes | FDOT Open Data Hub*. FDOT. <https://gis-fdot.opendata.arcgis.com/datasets/fdot::fdot-bike-routes/explore?location=27.635697%2C-83.741283%2C7.77>
- FDOT. (2020). *FLORIDA TRANSIT INFORMATION and PERFORMANCE HANDBOOK*. https://fdotwww.blob.core.windows.net/sitefinity/docs/default-source/transit/documents/2020-florida-transit-information-and-performance-handbook.pdf?sfvrsn=a2d91c1a_2
- Federal Transit Administration Office of Budget and Policy. 2016. *National Transit Summary & Trends*. Washington, D.C.: Federal Transit Administration.
- Federal Transit Administration, D. O. (2022). *PalmTran Zero Emission Transition Plan*. <https://www.transit.dot.gov/sites/fta.dot.gov/files/2023-02/PalmTran-Zero-Emission-Transition-Plan.pdf>

- Garikapati, V., Young, S., & Hou, Y. (2019). Measuring Fundamental Improvements in Sustainable Urban Mobility: The Mobility-Energy Productivity Metric. In *International Conference on Transportation and Development 2019: Innovation and Sustainability in Smart Mobility and Smart Cities* (pp. 111-121). Reston, VA: American Society of Civil Engineers.
- Hou, Y., Garikapati, V., Nag, A., Young, S. E., & Grushka, T. (2019). Novel and practical method to quantify the quality of mobility: Mobility energy productivity metric. *Transportation Research Record*, 2673(10), 141-152.
- Jeong, K., Garikapati, V., Hou, Y., Birky, A., & Walkowicz, K. (2020). Comprehensive Approach to Measure the Mobility Energy Productivity of Freight Transport. *Transportation Research Record*, 2674(7), 29-43.
- Johnson, Caley, Erin Nobler, Leslie Eudy, and Matthew Jeffers. 2020. Financial Analysis of Battery Electric Transit Buses. Golden, CO: National Renewable Energy Laboratory. NREL/TP-5400-74832. <https://www.nrel.gov/docs/fy20osti/74832.pdf>
- Kimley Horne and Associates, (2016). MUNICIPAL TRANSIT PROGRAMS. *Miami-Dade Metropolitan Planning organization*
<https://miamidadetpo.org/library/studies/guidelines-for-municipal-transit-programs-2016-03.pdf>
- Nag, A., Garikapati, V. M., Hou, Y., Young, S., & Schroeder, A. (2020). A Customizable Metric to Provide a Comprehensive Picture of the Mobility Potential of a Location. In *International Conference on Transportation and Development 2020* (pp. 251-261). Reston, VA: American Society of Civil Engineers.
- Owen, A., and D. M. Levinson (2014). Access Across America: Transit 2014. Center for Transportation Studies, University of Minnesota, Minneapolis.
- Penske (2017). CSCMP's Annual State of Logistics Report. Logistics in Transition: New Drivers at the Wheel. *AT Kearney*, Chicago, IL
- Southeast Florida Regional Climate Change Compact. (2023). Miami-Dade County Commits to Fleet Electrification by 2030 and Establishes a Fleet Fuel Efficiency Purchase Policy - Southeast Florida Regional Climate Compact.
<https://southeastfloridacclimatecompact.org/news/miami-dade-county-commits-to-fleet-electrification-by-2030-and-establishes-a-fleet-fuel-efficiency-purchase-policy/>
- U.S. Census 2016 American Community Survey. Retrieved from:
<https://www.census.gov/programs-surveys/acs/>
- US Census Bureau, C. for E. S. (2010, January 1). *US Census Bureau Center for Economic Studies Publications and reports page*. Data. <https://lehd.ces.census.gov/data/>
- U.S. Department of Transportation (2012). Freight Analysis Framework, Version 4.5.

APPENDIX A PREPROCESSING DETAILS

A.1.1 Installing Software

The first step is to install Anaconda. This is a distribution of python that is well suited for data management and processing.

Go to <https://www.anaconda.com/> and download the installer

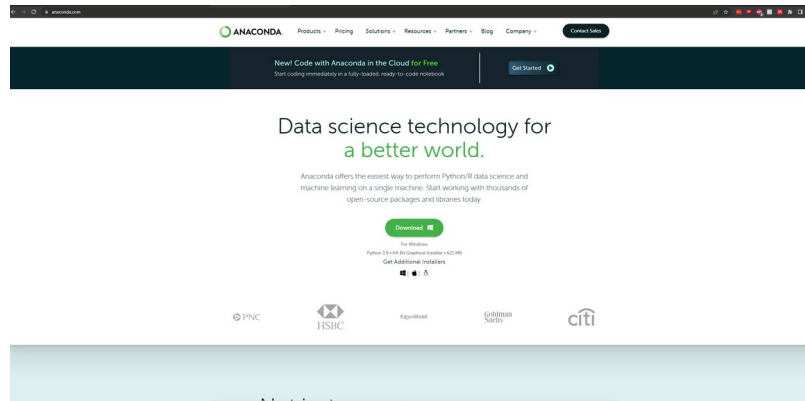


Figure 27 Anaconda Navigator Download

Install using all the Anaconda defaults

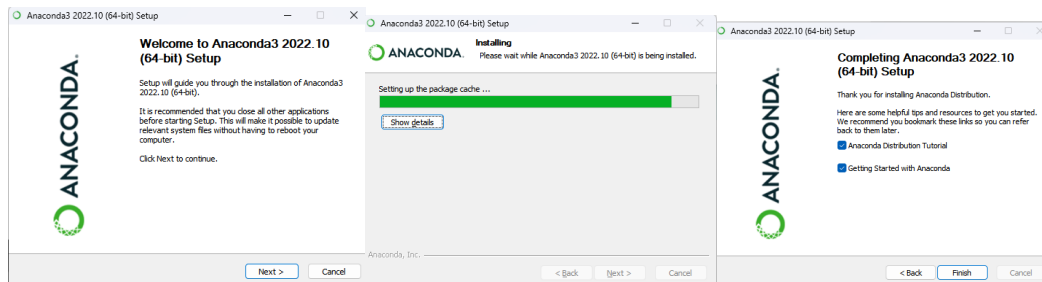


Figure 28 Anaconda Navigator Installation

Now open Anaconda Navigator. The three applications that will be used are the CMD prompt, Jupyter notebook and Spyder.

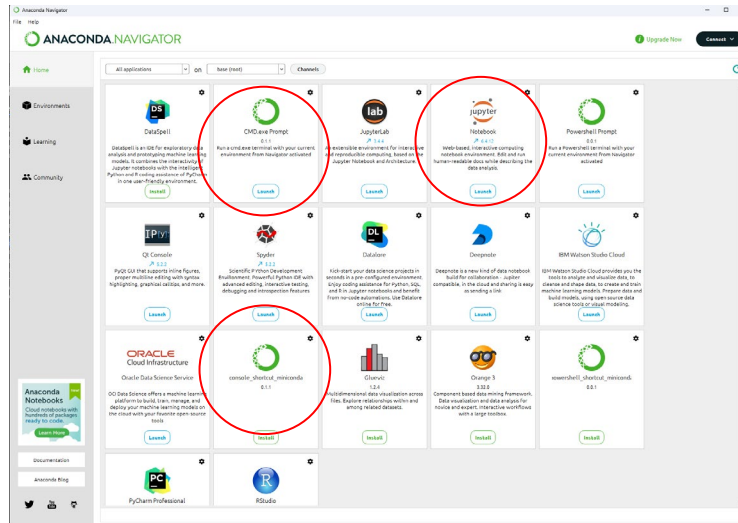


Figure 29 Applications to install within Anaconda Navigator

The next step is to install the necessary python packages to run the preprocessing code

Open the CMD prompt

Type each these commands one at a time to install each of the packages, when prompted to press y to proceed

conda install geopandas

conda install contextily

conda install psycopg2

conda install descartes

```

C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.22021.1205]
(c) Microsoft Corporation. All rights reserved.

(base) C:\Users\dafronconda>conda install geopandas

The following packages will be downloaded:

package | build | size
-----|-----|-----
descartes-1.1.0 | pyhd3eb1b0_4 | 9 KB
Total: 9 KB

The following NEW packages will be INSTALLED:

descartes pkgs/main/noarch::descartes-1.1.0-pyhd3eb1b0_4

Proceed [y]/n? y

Downloading and Extracting Packages

Preparing transaction: done
Verifying transaction: done
Executing transaction: done

(base) C:\Users\dafronconda>

```

Figure 30 Essential Packages Installation

Once the packages are installed everything is ready to run the preprocessing code to convert all the inputs into the correct format.

A.1.2 Preparing TAZ

To prepare the TAZ file, two data files are merged using a common attribute and generate new columns by aggregating and summing different employment data categories. The resulting dataframe is then filtered and saved to a geopackage file. Specifically, the code reads in a CSV file and a shapefile using the pandas and geopandas libraries, respectively. It then merges them based on a shared attribute and creates new columns to summarize employment data for each transportation analysis zone (TAZ). The new columns are then summed to create additional columns representing various employment categories. The resulting dataframe is then filtered to include only valid polygons and selected columns before being saved to a geopackage file, which will be used as input for the MEP code.

```
In [14]: taz = pd.read_csv('../data/FIU Data/maz_data_2015_v20210615.csv')
tazshp = gpd.read_file('../data/FIU Data/SERP8MAZ_TAZ/SERP8MAZ_NAD83_170502/SERP8MAZ_NAD83_170502.shp')
taz = tazshp.merge(taz, left_on='OBJECTID', right_on='mgna', how='left').fillna(0)

In [15]: taz['11'] = taz['emp_ag']
taz['21'] = 0
taz['22'] = taz['emp_utilities_prod']
taz['23'] = taz['emp_const_non_bldg_prod']
taz['31'] = taz['emp_mfg_prod']
taz['32'] = 0
taz['33'] = 0
taz['42'] = taz['emp_whsle_whs']
taz['44'] = taz['emp_retail']
taz['45'] = 0
taz['48'] = taz['emp_trans']
taz['49'] = 0
taz['52'] = 0
taz['53'] = 0
taz['54'] = 0
taz['55'] = taz['emp_prof_bus_svcs']
taz['56'] = 0
taz['61'] = taz['emp_pvt_ed_post_k12_oth'] + taz['emp_public_ed']
taz['62'] = taz['emp_health']
taz['71'] = taz['emp_amusement']
taz['72'] = taz['emp_hotel'] + taz['emp_restaurant_bar']
taz['81'] = taz['emp_personal_svcs_office']
taz['92'] = taz['emp_state_local_gov_ent']

In [16]: taz['jobs_tot'] = taz['11'] + taz['21'] + taz['22'] + taz['23'] + taz['31'] + taz['32'] + taz['33'] + \
    taz['42'] + taz['44'] + taz['45'] + taz['48'] + taz['49'] + taz['52'] + taz['53'] + \
    taz['54'] + taz['55'] + taz['56'] + taz['61'] + taz['62'] + taz['71'] + taz['72'] + \
    taz['81'] + taz['92']

taz['meals_tot'] = taz['72']
taz['recreation_tot'] = taz['71']
taz['shop_err_tot'] = taz['42'] + taz['44'] + taz['45']
taz['medical_tot'] = taz['62']
taz['sch_day_rel_tot'] = taz['61']
taz['area'] = taz.geometry.area

In [17]: taz = taz[['OBJECTID', 'MAZ', 'geometry', 'TAZ', 'POP', '11', '21', '22', '23', '31', '32', '33', '42',
    '44', '45', '48', '49', '52', '53', '54', '55', '56', '61', '62', '71', '72', '81', '92', 'jobs_tot',
    'meals_tot', 'recreation_tot', 'shop_err_tot', 'medical_tot', 'sch_day_rel_tot', 'area']]
taz = taz.loc[taz.geometry.type == 'Polygon']
taz = taz.loc[taz.geometry.is_valid == True]
taz.to_file('../data/FIU Data/processedMAZ.gpkg', driver='GPKG')
```

Figure 31 Code to Prepare TAZ

A.1.3 Preparing Roadway Network

To process the roadway network, it must be converted from a shp file to a gpkg and some small modifications must be made. The Python code reads in a street network file for Miami, Florida, splits the data into one-way and two-way streets, fills in missing values for some columns, calculates the average speed for each combination of posted speed limit and

functional class, and fills in missing speed values. The resulting dataframe is returned as the output of the code. This dataframe is then exported as a gpkg and is used as the roadway input in the MEP code.

```
In [41]: fiu_raw = gpd.read_file('../data/FIU DATA/2015 SERPM 8 Loaded Links Nodes/2015 SERPM 8 Loaded Links.shp')

In [23]: # fix Length
fiu_raw = fiu_raw.to_crs('epsg:32167')
fiu_raw.LENGTH = fiu_raw.geometry.length/5280

fiu_raw = fiu_raw.dropna(subset=['ID', 'ANODE', 'BNODE', 'LENGTH']).shape
fiu_raw.to_file('../data/miami_fl/FDOT Network/fiuNetworkFixedLength.gpkg', driver='GPKG')

In [24]: gdf_oneway = gpd.read_file('../data/miami_fl/FDOT Network/fiuNetworkFixedLength.gpkg')
gdf_twoway = gdf_oneway.loc[gdf_oneway['AB_TWOWAY']==1]
gdf_oneway = gdf_oneway[['ID', 'ANODE', 'BNODE', 'LENGTH', 'AB_FTG', 'AB_POSTEDS', 'AB_FFSPD', 'AB_AM_CON1', 'geometry']]

gdf_twoway.loc[gdf_twoway.BA_AM_CON1.isna(), 'BA_AM_CON1'] = gdf_twoway.loc[gdf_twoway.BA_AM_CON1.isna(), 'AB_AM_CON1']
gdf_twoway.loc[gdf_twoway.BA_FTG.isna(), 'BA_FTG'] = gdf_twoway.loc[gdf_twoway.BA_FTG.isna(), 'AB_FTG']
gdf_twoway.loc[gdf_twoway.BA_POSTEDS.isna(), 'BA_POSTEDS'] = gdf_twoway.loc[gdf_twoway.BA_POSTEDS.isna(), 'AB_POSTEDS']

gdf_twoway = gdf_twoway[['ID', 'BNODE', 'ANODE', 'LENGTH', 'BA_FTG', 'BA_POSTEDS', 'BA_FFSPD', 'BA_AM_CON1', 'geometry']]
gdf_twoway.columns = ['ID', 'ANODE', 'BNODE', 'LENGTH', 'AB_FTG', 'AB_POSTEDS', 'AB_FFSPD', 'AB_AM_CON1', 'geometry']
gdf_twoway.geometry = gdf_twoway.geometry.apply(lambda x: LineString(x.boundary[::1]))
gdf = pd.concat([gdf_oneway, gdf_twoway])
gdf = gdf.sort_values(by=['ANODE', 'BNODE']).reset_index()
del gdf['index']
gdf['ID'] = gdf.index

avgSpeed = gdf.dropna(subset=['AB_AM_CON1']).groupby(['AB_POSTEDS', 'AB_FTG']).mean().reset_index()

gdf.loc[gdf.AB_AM_CON1.isna(),].drop(columns=['AB_AM_CON1']).merge(avgSpeed[['AB_POSTEDS', 'AB_FTG', 'AB_AM_CON1']],\
on=['AB_POSTEDS', 'AB_FTG'], how='left')

gdf = pd.concat([gdf.loc[gdf.AB_AM_CON1.notna()],\
gdf.loc[gdf.AB_AM_CON1.isna(),].drop(columns=['AB_AM_CON1']).\
merge(avgSpeed[['AB_POSTEDS', 'AB_FTG', 'AB_AM_CON1']],\
on=['AB_POSTEDS', 'AB_FTG'], how='left')])

# if still have missing values, replace it by the speed limit
gdf.loc[gdf.AB_AM_CON1.isna(), 'AB_AM_CON1'] = gdf.loc[gdf.AB_AM_CON1.isna(), 'AB_POSTEDS']

gdf = gdf.dropna()
gdf.to_file('../data/miami_fl/FDOT Network/fiuNetworkFixedLengthTwoway.gpkg', driver='GPKG')
```

Figure 32 Code to Prepare Road Network

A.1.4 Preparing Bike Network

The bike network is very similar to the highway network. A simple .shp file can be used to distinguish different kinds of bike facilities to improve the bike isochrones.

The key attributes needed include:

- Unique link ID
- Network link geometry, including the direction of travel (such as shape fields)
- Link length
- Estimated speed (represents increased access from bike facilities)

Ideally, the bikeway network should be based on the highway network (to create the most accurate comparisons). Spatial selection of the highway links using the bike network will ensure that the bike network links are identical to the highway links.

The bike network is layered on top of the highway network to determine the bike isochrones. Limited access facilities are not traversable on bicycle, so these should be excluded from the bike mode.

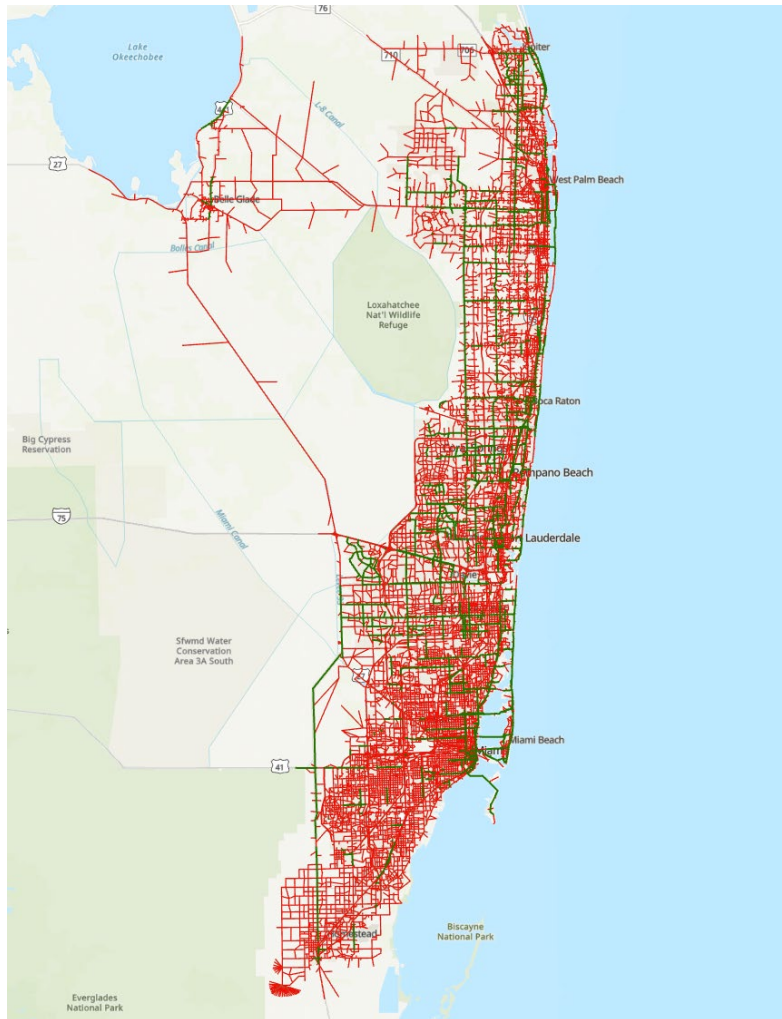


Figure 33 SERPM Highway Network with bike facilities marked in green.

To prepare the bikeway network, we need to extract the appropriate columns from the shp file including those identifying the links, the length and whether the link was a bikeway or not, the posted speed, and the functional classification. Then we need to establish the bike speed based on the functional classification and whether the link is categorized as a bikeway. After this is done, the table is exported as a geopackage which is the input required by the MEP code.

```

In [6]: fiu = fiu_raw[['A', 'B', 'LENGTH', 'Bikeway', 'FTG', 'POSTEDSPEED', 'geometry']]
fiu.head(2)

Out[6]:
   A  B  LENGTH  Bikeway  FTG  POSTEDSPEED  geometry
0  1  12075  0.278137    0.0    5.0         12.5  LINESTRING (1931756.565 9789854.611, 1931770.5...
1  2  12059  0.458943    0.0    5.0         12.5  LINESTRING (1926746.275 9791323.110, 1924486.0...

In [7]: fiu.loc[fiu['Bikeway'] == 1.0, 'bikeSpeed'] = 18
fiu.loc[(fiu['Bikeway'] == 0.0) & (fiu['FTG'] >= 5), 'bikeSpeed'] = 12
fiu.loc[(fiu['Bikeway'] == 0.0) & (fiu['FTG'] < 5), 'bikeSpeed'] = 0

In [8]: fiu.head(2)

Out[8]:
   A  B  LENGTH  Bikeway  FTG  POSTEDSPEED  geometry  bikeSpeed
0  1  12075  0.278137    0.0    5.0         12.5  LINESTRING (1931756.565 9789854.611, 1931770.5...    12.0
1  2  12059  0.458943    0.0    5.0         12.5  LINESTRING (1926746.275 9791323.110, 1924486.0...    12.0

In [10]: print('Speed = 18 mph, ', 'num=', fiu[fiu['bikeSpeed'] == 18].shape[0], ', account for', round((fiu[fiu['bikeSpeed'] == 18].shape[0] / fiu.shape[0]) * 100, 2), '% of total segments')
print('Speed = 12 mph, ', 'num=', fiu[fiu['bikeSpeed'] == 12].shape[0], ', account for', round((fiu[fiu['bikeSpeed'] == 12].shape[0] / fiu.shape[0]) * 100, 2), '% of total segments')
print('Speed = 0 mph, ', 'num=', fiu[fiu['bikeSpeed'] == 0].shape[0], ', account for', round((fiu[fiu['bikeSpeed'] == 0].shape[0] / fiu.shape[0]) * 100, 2), '% of total segments')

Speed = 18 mph, num= 7116 , account for 13.22 % of total segments
Speed = 12 mph, num= 32875 , account for 61.09 % of total segments
Speed = 0 mph, num= 13825 , account for 25.69 % of total segments

```

Figure 34 Code to Prepare Bike Network

A.1.5 Preparing Transit Network

The goal in this case is to convert the output from the SERPM travel demand model into GTFS which is what the MEP code needs. The typical output of a travel demand model (TDM) can be divided into three main components: stops/nodes, links, and routes. The stops/nodes component provides information about the location and sequence of nodes in the network, as well as whether a given node is a stop. The links component contains information about which nodes form a link, as well as the speed and distance between nodes. Finally, the routes component includes information on the route name and headways, which are the time intervals between successive vehicles on a given route.

Typical TDM Model Output:

- Stops/nodes - Contains location, sequence, and whether a node is a stop
- Links - Contains information on which nodes form a link, speed and distance between nodes
- Routes - Contains information on route name and headways

- GTFS is comprised of linked files

- Stops
- Stop_times
- Trips
- Routes
- Calendar
- Agency

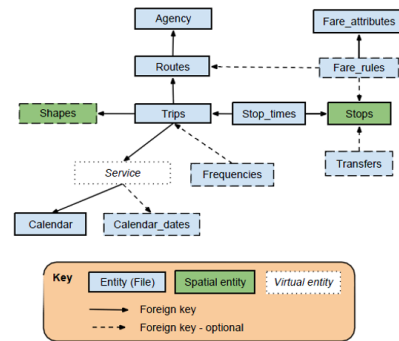


Figure 35 GTFS structure diagram

A python script to build this GTFS file from the travel demand model outputs was created to aid agencies who may not have GTFS for all of their transit routes use the MEP code. The steps to do so are:

1. Build Trip Start
2. Build link travel times
3. Use Trip start and link travel times to build stop times
4. Build trips
5. Build calendar
6. Convert transit nodes to transit stops

Build Trip Start

This process involves building a dictionary that includes information on all the times that trips for each route will start. It operates in a loop that is executed for each individual route. There are certain assumptions made during the execution of this process. For instance, if the data does not contain information on the start times of a service, it is assumed that the service will begin at 5 am. Additionally, it is assumed that rush hour will end at 9 am and then begin again at 5 pm, ending at 7 pm. Finally, the last service of the day is assumed to take place at midnight.

```
for route in routelist:
    peakheadway1 = dotroutes[dotroutes['LINEID2'] == route].iloc[0]['HEADWAY_1']
    offpeakheadway1 = dotroutes[dotroutes['LINEID2'] == route].iloc[0]['HEADWAY_2']
    tripstart1 = '05:00:00'
    tripend = '24:00:00'
    starttimes = tripstarts(peakheadway1, offpeakheadway1, tripstart1, '09:00:00',
                           '17:00:00', '19:00:00', tripend)
    starttimesdict[route]=starttimes
```

Figure 36 Code to build Trip Start

Build Link Travel Times

This process involves the creation of a data frame that captures the travel time between stops in a transportation network. It is executed only when the available data lacks information on transit stop to transit stop linkages. In such cases, certain assumptions are made during the execution of this process. Specifically, if information on transit link travel times is not available, the travel time between stops is calculated based on the road speeds and distances. By default, the process assumes that the bus travels at the same speed as traffic on the road.

(Insert image of code)

Use Trips start and link travel time to build stop time

This process involves constructing a data frame that provides information on the travel time between stops in a transportation network. It is only necessary to run this process when the available data does not already contain information on transit stop to transit stop linkages. During the execution of this process, certain assumptions are made. For instance, if transit link travel times are not available, it is assumed that the travel time between stops can be calculated using road speeds and distances. In this case, the default assumption is that the bus will travel at the same speed as traffic on the road.

```
stoptimesdf = buildstoptimesfromlinks(dirlist=['NB'], segments=linkmerge,
                                     segmentdirection='DIRECTION', segmentsequence='SEQNO',
                                     starttimedict = starttimesdict, fromstop='A', tostop='B',
                                     traveltime='tt1', routefield = 'LINEID2')
```

Figure 37 Code to Build Stop Times

Build trips

This process involves the creation of the trips data frame, where each trip must have a unique name. To generate a unique trip ID, the tool utilizes the route name and the start time of trips on that route. It is assumed that the process only builds trips for weekday service. The inputs required for this process include a dictionary of start times, a list of directions, and a service ID.

```
In [222]: tripsdf
Out[222]:
```

		trip_id	...	service_id
index			...	
0	18000.0	tripNBInter_County_Bus_PT1	...	1
1	19200.0	tripNBInter_County_Bus_PT1	...	1
2	20400.0	tripNBInter_County_Bus_PT1	...	1
3	21600.0	tripNBInter_County_Bus_PT1	...	1
4	22800.0	tripNBInter_County_Bus_PT1	...	1
...			...	
5227	81600.0	tripNBShuttle_PT4	...	1
5228	82800.0	tripNBShuttle_PT4	...	1
5229	84000.0	tripNBShuttle_PT4	...	1
5230	85200.0	tripNBShuttle_PT4	...	1
5231	86400.0	tripNBShuttle_PT4	...	1

```
[5232 rows x 4 columns]
```

Figure 38 Code to Build Trips

Build Calendar

This script is designed to generate the calendar data frame, which identifies the days on which each service is operating. To do this, the script leverages a dictionary of service and days as inputs. The dictionary provides information on the days of the week that each service is running. It should be noted that this script assumes that the service being created is only for weekdays, as this is the most common scenario for public transportation systems.

```
servicedict = {}
servicedict[1] = (['monday', 'tuesday', 'wednesday', 'thursday', 'friday'],
                 20220101, 20221231)
servicedict[3] = (['friday', 'saturday', 'sunday'],
                 20220101, 20221231)
calendar = buildcalendar(servicedict)
```

Figure 39 Code to Build Calendar

BuildStops

To transform transit nodes into the stops data frame, the inputs required are the latitude and longitude coordinates for each stop, as well as a stop ID field, and the nodes data frame. Prior to running the transformation process, it is necessary to ensure that the latitude and longitude coordinates are in the WGS84 projection format. Once the input data is prepared, the transformation process can be executed to convert the transit nodes into the stops data frame. This process is an important step in the analysis and optimization of transportation systems, as it allows for the accurate representation of stops and their locations, which is essential for planning and decision-making in the field of transportation.

```
dotnodemerge = pd.merge(dotnodes, dotnode, left_on = 'NODES', right_on='N')
geometry = [Point(xy) for xy in zip(dotnodemerge.X, dotnodemerge.Y)]

crs = ('EPSG:2235')

nodegdf = gpd.GeoDataFrame(dotnodemerge, crs=crs, geometry=geometry)

nodegdf = nodegdf.to_crs('EPSG:4269')

nodegdf['x1'] = nodegdf.geometry.x
nodegdf['y1'] = nodegdf.geometry.y

stopsdf = buildstops(nodegdf, 'NODES', 'y1', 'x1')
```

Figure 40 Code to build Stops

After all steps are done the final output should be a GTFS file that has been populated based on the travel demand model. The MEP code will use this GTFS output to generate the isochrones that will determine the MEP transit score.

APPENDIX B MODEL RUN PROCESS

Christopher Hoehne

Joshua Hoshiko

Robert Fitzgerald

Shivam Sharda

Venu M. Garikapati

National Renewable Energy Laboratory

UrbanMEP (config for Docker)

These instructions detail how to compute MEP with a provided Docker image. This is an updated version of the UrbanMEP repository and workflow that migrates all parameters and credentials required to execute MEP runs into a single configuration file that works within a Docker container. This approach no longer enables batching/looping multiple MEP runs and requires conducting one run at a time (per the Docker container). For you host computer (the computer where you install and run Docker), it is recommended to have at least 8 Gb of RAM. A workstation with higher available RAM and cores will improve the computation time.

B.1. Setup Docker

This step is recommended '*first*' as the first time you build an image of the provided container it could take an hour or more. Data preparation steps may also be conducted while the Docker image is being built for the first time but note if this is being done on the same computer, data preparation scripts may run slower than normal.

B.1.1 Install [Docker Desktop](#) (if you have not already)

B.1.2 Startup Docker Desktop

This will initialize the Docker daemon so it can be used in a terminal.

B.1.3. Unzip the provided file *UrbanMEP.zip* file to a directory on the computer

From this directory, we will run the Docker container and thus MEP. From this point on, all file paths will be relative to the location you choose to locate the /UrbanMEP/ folder on your computer or workstation. Inside this .zip, in addition to the READMEs, you should find:

- UrbanMEP/ data/ config.yml: this is the configuration file where you will set all parameters for MEP runs.
- UrbanMEP/ data/: the default location to read/write data used in MEP calculations (network, land use data, etc).
- UrbanMEP/ data/ [data processing scripts]: **SCRIPTS FOR PROCESSING DATA**

- UrbanMEP/data/db_connect.R: this script demonstrates how to connect and read/write data to the database created by our Docker container.
- UrbanMEP/setup-db.sql: this file specifies how to create a database needed for MEP runs.
- UrbanMEP/mep.dockerfile: this is the dockerfile that specifies how Docker should build a container image.
- UrbanMEP/install_R_dependencies.R: this is the R dependencies script that specifies what R packages, Docker should install when building the container.

B.1.4. Build Docker image

Navigate to your UrbanMEP directory in a terminal and run a Docker build command to build the image of the MEP codebase:

```
$ docker build -t <image name> -f mep.dockerfile .
```

where <image name> is the name of the image you would like to use (we've used mep-test for instance). This will build the image that will be used to run our container later. *This process will likely take up to an hour or more because the R dependencies take a while to install*, but once you build the docker image, Docker will cache steps for use later and will speed up future building if you need to delete and rebuild the image; unless the image is corrupted, it is not recommended to rebuild the image.

B.2. Data Preparation

This section overviews the data requirements for a MEP run and the steps to prepare the data. This MEP workflow uses a PostgreSQL database to host data and utilizes the pgRouting software on the database to run shortest path calculations used in MEP. The database is automatically set up and comes with one schema that is pre-loaded with processed data. In the section [Docker Container and Database Configuration](#), the details of the database and how to read/write data are described. If you are unfamiliar with the data formatting and MEP run process in general, it might be helpful to advance to [Docker Container and Database Configuration](#) and set up the container (and database within) for the first time. That way you can first see how data is formatted and stored for MEP runs.

B.2.1. Data Requirements

To run a MEP scenario, you need to obtain and prepare a series of datasets. We will need to determine the analysis boundary, prepare land use data, prepare employment data, prepare population data, and prepare network files (drive, bike, and walk, all of modes will have a separate network, but they can be derived from the same master network if it has enough representation of link types).

- an **analysis boundary** in shapefile format
- **land use point data** (or converted into point data) and classified into five MEP categories

- **employment (jobs) point data** (or converted into point data)
- **population point data** (or converted into point data)
- [for Drive, Bike, and/or Walk]: a **road network file** that describes all links in a transportation system that can be utilized by the mode(s)
- [for Transit]: **GTFS file(s)** describing the transit service of the region and an **OSM .pbk** **for the state(s) of interest** (download [here](#)).

These datasets need to be formatted in a specific way for the MEP code to work properly. You can check expected data formats, for processed datasets by querying the various tables after you set up the Docker container and it's database. See the data/db_connect.R script as an example to query and upload data to/from the database.

B.2.1.1. Configuration assumptions

There are other assumptions in the configuration file (UrbanMEP/data/config.yml) that you need to update, and some that you may choose to update or leave as defaults.

- **scenario**: a unique name (string) for a MEP run. This must meet the requirements of a [valid Postgres Schema name](#), such as using underscores in place of spaces, and beginning with a letter (e.g., my_test_scenario123). Each MEP scenario can have one mode of each type (one Drive, Bike, Walk, and Transit). You should create a different scenario for each set of changing assumptions in the network, land use, employment, or transit schedules.
- **modes**: a vector of the travel mode names you would like to run for this scenario, such as ["drive","bike","walk","transit"]
- **degrees**: resolution of grid cells for MEP in degrees lat/lon. recommended to keep at 0.01 which is approximately 1 kilometer. Smaller values increase compute time.
- **db_***: database credentials. Keep "mep" for db_name and db_user. We recommend choosing your own password for db_pass which you will also need to specify when initializing the Docker container for the first time. You may need to update the port if your host computer has PostgreSQL or another service utilizing the default port of 5432. See the next section for more details on this.
- **mode_energy**: the energy use by mode in kWh/PMT. **This vector must match the order and length of modes** (e.g., if drive is first in modes, the drive energy use must be first in mode_energy).
- **mode_cost**: the cost by mode in USD/PMT. **This vector must match the order and length of modes** (e.g., if drive is first in modes, the drive cost must be first in mode_cost).
- **city_bound_path**: the file path for a boundary shapefile to be used for MEP analysis.
- **activities**: the activity categories considered in MEP. It is recommended to keep this as is.

- **N_star**: the baseline total opportunities for reference category (meals by default) in the opportunity space. If wanting to localize the opportunity normalization, pick a reference category and sum all available opportunities of that type and use here.
- **N_j**: the total opportunities for all categories. **This vector must match the order and length of activities**. If wanting to localize the opportunity normalization, sum all available opportunities by type and use here (this includes the reference category).
- **freq**: a vector for the frequency of engagements by opportunity type. **This vector must match the order and length of activities**. By default, this is derived from NHTS (e.g., 3.1 means 3.1% of trips were by said activity type). Exclude trips that do not include the types of included MEP activities. This vector does not need to sum to 100.
- **otp_***: these are parameters for OpenTripPlanner settings used to calculate transit MEP. These are recommended to be left as is.

If you are interested in changing the activity engagement frequencies (**freq**) or land use normalization parameters (**N_star**, **N_j**) based on localized travel and land use data, it may be helpful to review the [MEP methodology paper](#).

B.2.2. Drive, Bike, and Walk Data

B.2.2.1. Update other Drive, Bike, and Walk parameters

Update the relevant parameters for Drive, Bike, and Walk modes UrbanMEP/ data/config.yml. Usually, we assume the bike and walk cost and energy use is zero, but you can update this to a nonzero assumption if desired.

- **modes**: a vector of the travel mode names you would like to run for this scenario, such as ["drive", "bike", "walk", "transit"]
- **mode_energy**: the energy use by mode in kWh/PMT. **This vector must match the order and length of modes** (e.g., if drive is first in modes, the drive energy use must be first in mode_energy).
- **mode_cost**: the cost by mode in USD/PMT. **This vector must match the order and length of modes** (e.g., if drive is first in modes, the drive cost must be first in mode_cost).

B.2.2.2. Network file(s)

For a Drive, Bike, and/or Walk MEP calculation, you must provide a formatted network file. It must have all links by mode representing their actual directionality. In other words, some networks will have a data column that specifies a roadway link is bidirectional, but this link only has a unidirectional spatial representation. The MEP routing process (using pgRouting) uses the node order (the two ends of links) to determine the direction of travel. You can find the node order by seeing which node appears first when viewing the data, or by importing the network into geospatial software like QGIS or ArcGIS and using directional arrows. If you find that there are issues with the actual versus desired directionality, you need to fix this before using in MEP otherwise routing may incorrectly assume directions. For links with one physical

spatial representation but bidirectionality desired, you can duplicate the links and reverse the duplicated links' directions (such as with ST_Reverse).

B.2.3. Transit data

B.2.3.1. Create UrbanMEP/gtfs/ folder

This folder is where the processed GTFS.zip file will go in addition to the processed osm.pbf file. Only these two files should be in this folder for a transit run. If there are multiple files of these types or other files, it may corrupt the process. If you need to store files for other Transit runs, you can create subfolders in UrbanMEP/data or anywhere outside of UrbanMEP and copy them in when ready.

B.2.3.2. Update other transit parameters

Update the relevant parameters for Drive, Bike, and Walk modes UrbanMEP/data/config.yml. Usually we assume the bike and walk cost and energy use is zero, but you can update this to a non-zero assumption if desired.

B.2.3.3. Download or create GTFS data

Download or process GTFS for your city/scenario from available online data sources or via other software. - [openmobilitydata](#) - [transitland](#) - MEP only works with GTFS Schedule datasets, not [GTFS Realtime](#) - Google search "gtfs" as last resort, sometimes you can find one at the municipality website - For more information on GTFS visit [gtfs.org](#).

Once you've created or located a GTFS file, *make sure it is zipped such that all .txt files are not in any subdirectories* (e.g., gtfs.zip/*.txt NOT gtfs.zip/gtfs/*.txt) and place it in the GTFS .zip in the UrbanMEP/gtfs folder with no other files. The name of the file does not matter. If you are manually zipping the collection of required .txt files, it is recommended to select all files then right-click over one and select add to .zip. If you select the folder and zip it, it will create an extra subfolder layer and OpenTripPlanner will not correctly find the files. Place

Inspect your GTFS .zip archive and view its "calendar.txt" file. Select a valid query date from the ranges of dates provided by the GTFS dataset and update gtfs_query_date in data/config.yml. - Look for columns "start_date", "end_date", pick a weekday in between a listed date range for any row.

B.2.3.3. Download OSM data

Download the state-level osm.pbf file(s) and put it in UrbanMEP/data. - [Geofabrik](#) is likely the best source for this. - Downloads from geofabrik might not work on a VPN - Make sure you are covering the entire city area. Some cross state boundaries, you will osm.pbf files for all states.

Once the Docker container is running, the osmosis tool will take the raw osm.pbf data you provided and trim it to a smaller version subset (not the whole state) based on your scenario's boundary (UrbanMEP/mep_transit_isochrones/generate_bbox_for_city.R). This is done to more easily read into OpenTripPlanner which uses the OSM street network for connections and first/last mile portions of transit trips.

B.3. Docker Container and Database Configuration

After the container image is built, you've prepared the necessary datasets, and you've finalized the parameters in `UrbanMEP/data/config.yml`, we can run the Docker container. Note there many different ways to run a container, but we will provide a recommendation of arguments to setup the MEP container to a) utilize shared volumes to interact with configuration parameters and share data with the docker that wasn't provided; and b) set up an internal PostgreSQL database to store input data and output results.

The entry point is Bash, so you will be running the terminal "in" the container. You should be able to use Linux commands as you normally would and interact with the files, check dependency versions, etc. The MEP container is built with some R and Python scripts and software including Java x64, Osmosis, and OpenTripPlanner to conduct a MEP run.

When you run the container for the first time, we will link certain volumes (folders) on the host (the computer running Docker). We will also initialize the database that is used to store prepared inputs for MEP and resulting outputs. This database will persist as long as you do not delete the container and perform a few simple steps. However, the database is only accessible *while the container is running*. Thus, to read/write data to/from the database (including results), you must restart the container. This is a simple process that is overviewed in this section.

B.3.1. Start the container (first time)

Open a terminal and navigate to UrbanMEP and run:

```
$ docker run -it --name=<container name> -v %CD%/setup-db.sql:/docker-entrypoint-initdb.d/setup-db.sql -v %CD%/data:/../data -v %CD%/gtfs:/../gtfs -v mep-data:/var/lib/postgresql -e POSTGRES_PASS=<your password> -e  
PASSWORD_AUTHENTICATION=md5 -p <host port>:5432 -e  
SCRIPTS_LOCKFILE_DIR=/var/lib/postgresql <image name>
```

such that:

- `<container name>` is your chosen name, e.g., `mep-fdot`
- `<image name>` is the name you chosen when building the image, e.g., `mep-test`
- the password in `data/config.yml`. By default it is `password` but we recommend changing this.
- `-it` is for interactively running the container
- `-v %CD%/setup-db.sql:/docker-entrypoint-initdb.d/setup-db.sql` links the SQL setup script to run in the docker. `%CD%` should use your relative path where UrbanMEP is located. You should not need to change this.
- `-v %CD%/data:/../data` and `-v %CD%/gtfs:/../gtfs` link your folders for general data and GTFS data. You should have already create UrbanMEP/gtfs and put your GTFS data must already be put in this separate folder before running the container or Docker

won't recognize the files if you drop them in after linking this volume when starting the container.

- `-v mep-data:/var/lib/postgresql` links a shared volume called "mep-data" where the PostgreSQL database can persist.
- `-e PASSWORD_AUTHENTICATION=md5` changes the password authentication type. This may not be required but we recommend it in case.
- `-p <host port>:5432` specifies to enable connections from the host (your computer) to the container. By default, we use 5432 for <host port>. However, if you already have PostgreSQL installed on the host (or another software that uses this port, you may need to choose another port. If you are unsure, see the next section [Changing your port](#) before for more details before running your container. You should leave at 5432 as this is specified automatically when the database is created.
- `-e SCRIPTS_LOCKFILE_DIR=/var/lib/postgresql` this creates a lockfile that should lock the database when the container is not active to avoid corruption of the database.

B.3.2. Changing your port

Note that this is only necessary if you have PostgreSQL installed on your host computer and have not modified the services port (which defaults to 5432), or if another service is using/blocking port 5432 from accepting connections. Check, and if so, please note:

- you can change your computers PostgreSQL port to something other than 5432 to avoid changing the port that the MEP Docker container will be forwarded across so you can access the Docker's database without modifying any parameters in the MEP Docker workflow. However, this may be more difficult than the other option of using another port;
- you can create a copy of `config.yml` (e.g., `config-ex.yml`) and change the copied file's `db_port` parameter an open port (5433 is a good option). You'll then need to update any scripts that are connecting to the Docker database from the host computer such as `UrbanMEP/data/db-connect.R`; in this script, you need to update the following line to link to this new config file (near the top of the script): `Sys.setenv(R_CONFIG_FILE = "config/config-ex.yml")`. This will tell your host computer to use the proper port to connect to the Docker database if we also update the `-p` command during docker run to `-p <my new port>:5432` (previous step).

B.3.2. Initialize the PostgreSQL database:

To initialize the PostgreSQL database where we store prepped data and results, run:

```
$ ./docker-entrypoint.sh
```

This will take a few minutes, and after it stops spitting out 'linestring' and other text, and say's "it's listening", leave the terminal open. The database is now up and running.

B.3.3. Using the container and database

After the database is initialized (no more text is spitting out and it says its 'listening'), open a **separate** terminal window, navigate to UrbanMEP, and run:

```
$ docker exec -it <container name> bash
```

This opens another interactive Bash entry into the container where we can start to execute other steps inside the Docker.

B.3.4. Stopping/starting the container and database

Any time you are finished or need to pause working after you've already run the container for the first time, you can start or stop the container by going to the Docker Desktop app and hitting the 'start' (triangle) or 'stop' (square) icon to the right of your container under the Containers tab. Or you can do this from the command line with:

```
$ docker start <container name>
```

```
$ docker stop <container name>
```

It will say your container is running in the Docker Desktop app, or you can check with `$ docker ps` which lists all running containers.

Just like how we connect to the container after initializing the database, we can reconnect to it interactively using:

```
$ docker exec -it <container name> bash
```

If you are restarting the container, we must also reboot up the database. The lockfile should prevent any issues with data corruption if you start and stop the container normally (we've not yet run into issues using this workflow). To do this, simply rerun the `docker-entrypoint.sh` Bash script:

```
$ ./docker-entrypoint.sh
```

Docker will recognize the database is already present and boot it up. This should not take very long. You will then need to again open a new window to interactively connect to the container with `$ docker exec -it <container name> bash`.

B.3.5. Updating data for the container

When initially starting the container, we specified some linked volumes, specifically `UrbanMEP/data` and `UrbanMEP/gtfs`. Because these are linked between the host and the container, the container should automatically see that you've updated any files or data within these folders. You can check that this is happening by using the `cat` command before and after a change to print out the file text:

```
$ cat ../data/config.yml
```

If for some reason you need to update a file in the container that is not in one of the shared volumes, you can simply drop the updated file in the shared volume (folder) on your host

computer, then from the Docker container copy the updated file from the shared location to the desired internal location:

```
$ cp ../data/my.file my.file
```

This can be used to test updates or tweaks to some of the scripts. But note that this is unstable as the dependencies in the container are locked and you must rely on what is already installed and built.

B.4. Add Prepped Data to Shared Docker Database

Once we've set up the container, it's database, and have prepared the necessary data, we begin to upload the data to the database and execute runs.

For testing purposes, we've created the schema `miami_fl_fiu_docker_0222` on the database that is pre-loaded with the proper formatted data. This is the default scenario specified in `config.yml`. When first testing your MEP Docker image, we recommend skipping this section and advancing to [Run MEP](#) for at least one mode to verify that the image and container are correctly set up.

B.4.1. Test connecting to the Docker database

Use the provided script `UrbanMEP/data/db_connect.R` to connect to the database once the container and the database are running. Inside this script are a number of examples of reading/writing data. You are not required to use R or a version of this script to read/write data, but you may.

B.4.2. Initiating a schema

If this is a new scenario (scenario is unique), we need to create a schema for the new scenario on the database.

```
$ Rscript mep2.R create_city_schema
```

The scenario will be created on the database under the a schema with the same ``scenario`` name. Subsequent tables will be added under this schema (e.g., ``my_city.bike_road``).

B.4.3. Creating a MEP grid

We create a MEP grid based on a provided boundary file. Many other steps require a city grid so we need to do this as soon as we've created a new schema. Put the boundary file in `UrbanMEP` data and specify the path and name in the variable `city_bound_path` in `config.yml`. Then, run the upload command:

```
$ Rscript city_grid_local
```

You can check this worked by using `data/db_connect.R`.

B.4.4. Uploading network data

Upload or import a processed network to database. If using a customly created or transformed network (e.g., for OpenStreetMap or TransCAD/DOT processed networks), use the provided

custom network processing and upload script. Network files are used for Drive, Bike, and Walk modes only.

B.4.4.1 Custom networks

[FDOT SPECIFIC INSTRUCTIONS WILL DIFFER, TBD]

B.4.4.2 TomTom networks

If you have a TomTom subscription and API, we need to query the network and upload it before running MEP for drive, bike and/or walk modes. Make sure you've entered the API key and the correct time zone for your location in config.yml (tomtom_key and time zone).

Query TomTom network data and city_grid table for you scenario:

```
$ Rscript mep2.R tomtom_query
```

Download the TomTom network data once it is available (this may take a few hours or a day depending on the license limitations):

```
$ Rscript mep2.R tomtom_download
```

Upload the TomTom network after the query has completed.

```
$ Rscript mep2.R upload_network
```

B.4.5. Uploading land use, employment, and population data

[FDOT SPECIFIC INSTRUCTIONS WILL DIFFER, TBD]

If you are using custom data and processing scripts to generate land use, population and/or employment data, use the instructions provided with those scripts to upload the data to the corresponding schema tables on the database.

If you are not using custom datasets, you can import data individually from the data (if it is already uploaded) with:

```
$ Rscript mep2.R import_landuse
```

```
$ Rscript mep2.R import_pop
```

```
$ Rscript mep2.R import_jobs
```

- Land use data must be uploaded to the database for the import_landuse function to work (specified in config.yml as costar_schema_name.costar_table_name). By default, NREL uses licensed data from CoStar (hence the legacy schema/table names), but the data can be sourced elsewhere but must follow the same format to query properly. We do not provide this data, and it must be provided by the user and formatted properly.
- Population data requires the user to obtain a Census API token which is specified in census_api_token.
- Employment (jobs) data requires land use data to be uploaded first as counties are sourced from land use data. The data must be located under the schema

name census_lodes with tables for each state using the two letter acronym, and the tables' data in the LODES format. We do not currently provide this data.

B.5. Run MEP

B.5.1. Running Drive, Bike, and Walk

This step involves running a variety of R and Python scripts that pull from the parameters set in config.yml.

B.5.1.2. Run isochrone and opportunity counts computations

To run isochrone computations, all required data must be uploaded to the database under your desired schema including a network for each mode you desire. You can run multiple modes at once by specifying so in the modes argument in config.yml (e.g., ["drive", "bike", "walk"]):

```
$ Rscript mep2.R create_iso
```

To run opportunity counts *for point data*, all required data must be uploaded to the database under your desired schema including a network for each mode you desire. You can run multiple modes at once by specifying so in the modes argument in config.yml (e.g., ["drive", "bike", "walk"]):

```
$ Rscript mep2.R count_opps
```

NOTE: *If you are not using point land use data (e.g., TAZ shapes or similar polygon data), the process to count opportunities is different.*

You can also run the isochrone and opportunity counts steps together with

```
$ Rscript mep2.R run
```

But this again requires the land use data to be in the point format.

B.5.2. Running Transit

Running Transit MEP is a slightly different process with some differing run steps.

B.5.2.1. Create a bounding box for city

This is to feed a bounding box to osmosis in order to crop the OSM network to our area of interest. NOTE: this assumes you've already have a city_bound in a scenario. If you do not, stop, and do that first.

```
$ Rscript generate_bbox_for_city.R
```

This produces "xmin", "xmax", "ymin", "ymax" inputs to osmosis queries, below.

B.5.2.2. Create a cropped .pbk file from bounding box using osmosis.

If the study area includes multiple states, you must first *combine them* with multiple state .pbk files Kansas City, MO. If you are only using one state, then skip the first command:

```
$ osmosis --rb ../data/<file1>.osm.pbf --rb ../data/<file2>.osm.pbf --merge --wb  
../data/merged.osm.pbf
```

Filter query:

```
$ osmosis --read-pbf ../data/<state or merged>.osm.pbf --bounding-box top=<ymin>  
left=<xmin> bottom=<ymin> right=<xmax> --write-pbf ../gtfs/<city name>.osm.pbf
```

B.5.2.3. Boot up an OTP server with the directory as an argument.

```
$ java -jar -Xmx2G /path/to/otp-1.4.0-shaded.jar --build /path/to/files --inMemory --  
port 8082
```

- wait until console reads "INFO (GrizzlyServer.java:153) Grizzly server running."
- LEAVE RUNNING! Open another terminal to connect to the container: `$ docker exec -it <container name> bash`
- NOTE: -Xmx2G allocates 2 Gb of memory in Java for the OTP server. This should usually be enough but can be increased if needed. Note that your Docker must also have more memory allocated than you specify here via it's host computer.

B.5.2.4. Run the Transit isochrone and opportunity count computations

```
$ mep_transit_isochrones/compute_isochrones.R
```

If this completes successfully, the isochrone tables will be added to the proper schema on the database.

After you finish running the transit isochrones workflow for your scenario, we just need to count the opportunities intersecting transit isochrones:

```
$ Rscript mep2.R count_opps
```

B.6. Compute MEP results

This is the final step to calculate MEP. You can do this on the Docker container which stores the results under the schema.table of `mep_no_tnc.[scenario]_mep`:

```
$ Rscript mep2.R compute
```

Alternatively, you can use the R script `data/compute_mep_local.R`. This might be advantageous if you wish to calculate multiple scenario alternatives that hold the opportunity counts constant but change factors such as the energy, cost, activities frequencies, weights (or anything non-speed / non-network). You can even use the script `data/db_connect.R` to query opportunity tables (after they've been computed) and save on your local (host) computer, and then use the `compute_mep_local.R` to retrieve those locally. This is helpful if you don't want to persist or restart the container every time you want to update or alter a scenario calculation.

B.7. Visualize MEP Results

[INSTRUCTIONS TBD]