Tutorial: Using EVRPGen for Generating Electric Vehicle Routing Problem Instances

Mehmet Anil Akbay Christian Blum

March 2025

1 Introduction

Electric Vehicle Routing Problems (EVRPs) have become increasingly relevant due to the rise of electric vehicles in logistics. Traditional vehicle routing problem datasets often lack realistic road network details, such as junctions and road types. To address this limitation, we introduce EVRPGen, a web-based application for generating realistic problem instances based on OpenStreetMap (OSM) data.

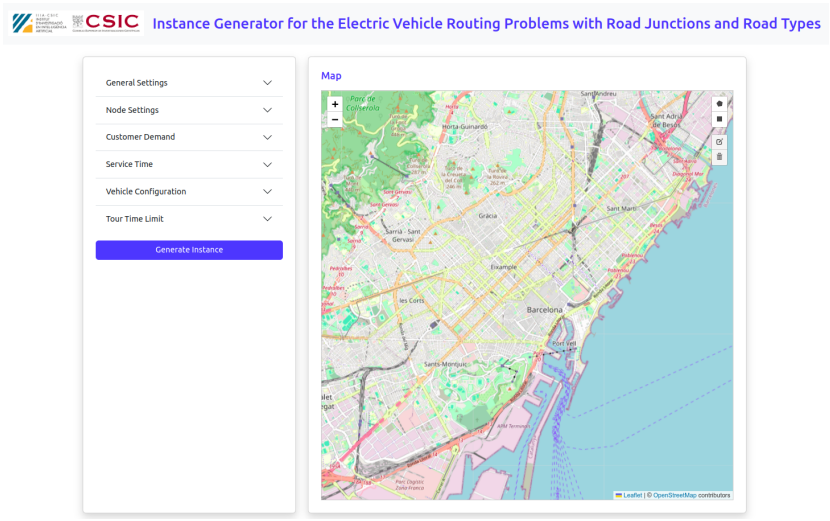

Figure 1: Vehicle Configuration Panel

This tutorial provides a step-by-step guide on how to use EVRPGen to define a geographic area, configure problem parameters, and generate instances that can be used for research and experimentation.

2 System Overview

EVRPGen consists of:

- A React-based frontend for user interaction.
- A Flask-based backend for instance generation.
- A RESTful API that connects the frontend and backend.
- Integration with OpenStreetMap (OSM) for extracting real-world road network data.

3 Getting Started

To use the EVRPGen web application, follow these steps:

3.1 Step 1: Accessing the Web Application

Visit the following link to access the web application:

`https://github.com/manilakbay/InstanceGenerator`

Ensure that you have a modern web browser (Google Chrome, Mozilla Firefox, or Microsoft Edge) for the best experience.

3.2 Step 2: Selecting the Area

Users can define the area for instance generation using either:

- **Address Input:** Enter a city or location (e.g., `Barcelona, Spain`).
- **Geographic Coordinates:** Manually enter a bounding box or polygon.

This selection will determine the road network data used in the instance.

3.3 Step 3: Configuring Instance Parameters

The configuration panel allows users to define several key parameters:

1. **Number of Nodes:** Specify the number of depots, customers, and charging stations.
2. **Demand and Service Time:** Choose between default, constant, or random values.
3. **Vehicle Settings:** Define battery capacity, load capacity, and charging rates.
4. **Road Network Settings:** Select whether to simplify road connections.

Figures **??** to 1 illustrate these settings.

3.4 Step 4: Generating an Instance

After configuring all parameters, click the **Generate Instance** button. The application will process the request and generate an EVRP instance.

Once the instance is ready, the following outputs will be available:

- A downloadable text file with the dataset.
- A summary of instance statistics.
- An interactive map visualization.

4 Running the Application Locally

To run the EVRPGen instance generator on your local machine, follow these steps:

4.1 Prerequisites

Ensure you have **Docker** installed and running:

- **Linux**: Ensure your user is in the `docker` group and that Docker is active.
- **Windows/macOS**: Install **Docker Desktop**.

4.2 Step 1: Build the Docker Image

Open a terminal in the repository's root directory (where the `Dockerfile` is located) and run:

```
docker build -t evrp-generator .
```

4.3 Step 2: Run the Docker Container

After building the image, start the application by running:

```
docker run -d -p 5000:5000 evrp-generator
```

4.4 Step 3: Access the Web Interface

Once running, open your browser and visit:

`http://localhost:5000`

4.5 Step 4: Downloading Outputs

Generated instances can be downloaded from the web interface or found in the container's directory `/app/created_datasets`.

4.6 Stopping the Container

To stop the running container, find its ID using:

```
docker ps
```

and stop it with:

```
docker stop <container-id-or-name>
```

5 Conclusion

This tutorial has provided a comprehensive guide on how to use EVRPGen for generating realistic Electric Vehicle Routing Problem instances. By leveraging OpenStreetMap data and customizable configurations, users can create benchmark datasets that better reflect real-world logistics challenges.

For further assistance or to contribute to the project, visit:

`https://github.com/manilakbay/InstanceGenerator`