Publication Note: *This manuscript is a preprint version. The final published Version of Record is available at* [*https://doi.org/10.1007/s42421-026-00166-4*](https://doi.org/10.1007/s42421-026-00166-4)

An AI-reinforced traffic digital twin for testing emergency vehicle interventions

Farnoosh Namdarpour, Fan Zuo, Di Sha, Hannah Bonestroo, Jingqin Gao, Joseph Y. J. Chow, and Kaan Ozbay

C2SMART Center, New York University Tandon School of Engineering, Brooklyn, NY, USA

August 3, 2026

Abstract

Emergency response time is a vital indicator of public safety effectiveness in dense urban settings like New York City, where even small delays can significantly affect survival rates. To support data-driven decision-making and enhance emergency response strategies, this project developed a Traffic Digital Twin (TDT): a high-fidelity, simulation- and AI-enhanced framework that replicates Emergency Medical Vehicle (EMV) operations under real-world, transient conditions. Developed in collaboration with the New York City Fire Department (FDNY), the TDT integrated diverse data sources and employed advanced calibration methods to accurately model EMV behavior and surrounding driver responses, as detailed in a separate paper. This paper focuses on a complementary AI model, EMVAID, based on a Traffic Simulation-Informed Explainable Boosting Machine (TSI-EBM), designed to interpret key factors influencing EMV speeds. As an illustration of TDT decision support, it is then applied to optimize ambulance Cross Street Locations (CSLs), achieving up to 14.5% reductions in expected travel time during morning traffic while accounting for the likelihood of a congested CSL being unavailable. The study also demonstrates that proxy traffic features, such as 5-minute average speeds, can effectively support real-time EMV operations, underscoring the TDT's potential as a scalable, interpretable decision-support tool for emergency response planning.

1 Introduction

Response time is an important indicator of a city's emergency management effectiveness, as faster responses directly contribute to better outcomes, including saving lives and minimizing property loss (de Larrea et al., 2021). For instance, Berdowski et al. (2010) found that the survival rate from sudden cardiac arrest decreases by 7%–10% for every minute of delay, with survival becoming unlikely after 8 minutes. Since a significant portion of emergency response time is the travel time of Emergency Medical Vehicles (EMV) to the incident site, reducing this time is essential for improving a city's emergency response capabilities. However, this task has become increasingly challenging over the past decade as overpopulation and urbanization continue to exacerbate road congestion. Records (NYC, 2023) indicate that the average EMV response time in NYC across all types of EMVs increased from 2015 to 2023. Specifically, average medical emergency response times handled by the New York City Fire Department (FDNY) rose from 10.39 minutes to 13.68 minutes, with the majority of this 31.7% increase attributed to extended travel times in traffic (from 4.43 minutes to 6.08 minutes).

Given these increases, reducing EMV travel times remains an important priority for public safety agencies. However, testing the effectiveness of potential interventions in real-world settings can be both resource-intensive and challenging, particularly when agencies are operating with limited capacity and need to ensure that any changes do not compromise safety or performance. For example, EMV travel times with sirens on are not identical to passenger vehicle travel times that follow traffic control rules. The EMVs can pass by red lights but also depend on downstream vehicles pulling over or switching lanes to allow them to pass. As such, data from commonly available tools such as Google Maps do not provide consistent forecasts of EMV travel times along different paths.

Thus, the ability to test interventions virtually through a Traffic Digital Twin (TDT) could provide invaluable insights into reducing response times by properly capturing the nuances of travel by EMVs and passenger vehicles, as illustrated in Figure 1. However, EMV traffic simulation is not a trivial implementation. Whereas conventional traffic simulations most commonly model a *typical* time period to evaluate steady state performance, EMV-based simulations must reproduce conditions corresponding to specific incidents in transient conditions rather than typical conditions. This requirement complicates the use of simulation because unlike typical traffic simulation studies, one cannot simply run a simulation long enough to draw out the steady-state performance measures. Instead, it becomes necessary to develop a new way to use traffic simulations to obtain a simulated sample distribution by simulating multiple incidents.

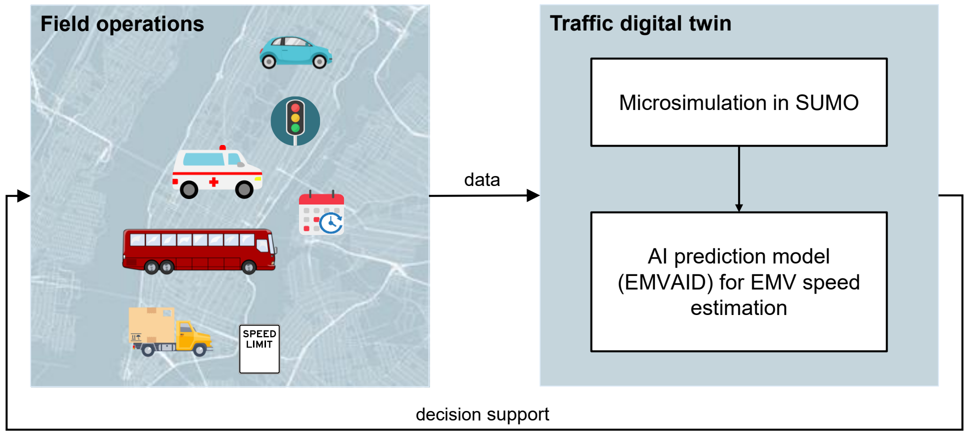


Figure 1: Overall TDT flowchart.

Furthermore, while open-source platforms such as Simulation of Urban Mobility (SUMO) are capable of modeling specific EMV behaviors, such as sublane movements, queue jumping (Su et al., 2020), and the use of sirens to alert surrounding drivers, developing an accurate EMV traffic simulation remains particularly challenging because it also requires proper behavioral calibration of non-EMV driver responses to EMVs and capturing transient traffic conditions during a dispatch. Specifically, the behavior of non-EMV drivers is difficult to simulate realistically because non-EMV drivers may not readily move aside, respond slowly, or become blocked by other vehicles. This challenge is further exacerbated by limited availability of observational data of these interactions, thus requiring proxies of these thoroughly time-dependent traffic data (e.g. surrounding 5-minute traffic speeds as opposed to observed speeds of downstream vehicles along an EMV's dispatch trajectory). Even in cases where such data were collected, it would not be possible to collect counterfactual data for alternative routes that an EMV might have taken, which would limit the use for planning purposes. The research question here is "whether such proxy data can adequately serve in explaining the differences in EMV travel speeds from passenger vehicle travel speeds under normal operating conditions".

Traffic simulation provides a means to synthesize features that are not observable, such as road occupancies and queues that form during an EMV's dispatch. This provides simulation-informed features that can potentially explain how the EMV travel speeds differ. This is tested in the development of a machine learning model to predict EMV travel speeds in the study area using both observed data and simulation-informed data. While the inputs of the model would then be dependent on a local area traffic simulation, it argues for the collection of such features if generalizing the model to a whole city where simulation would not be possible.

In other words, a successfully trained model using 5-minute passenger vehicle travel speeds and occupancies on the road suggests that their real-time collection for emergency response agencies can indeed help inform the dispatch decisions including selection of vehicles to assign to an incident and the routing of those vehicles. It helps answer the question of whether such features can adequately serve as proxy features for supporting real-time dispatch decisions in a digital twin environment.

Though developing an accurate EMV traffic simulation is challenging, agencies such as FDNY stand to benefit significantly from such innovative tools. Given the risks and costs associated with real-world intervention testing, developing an accurate Traffic Digital Twin (TDT) offers a promising and practical solution for evaluating strategies to reduce EMV travel times. The ultimate goal of this research is the creation of a citywide, real-time TDT that dynamically updates using AI methods. The initial phase of the work has focused on building foundational components through a smaller-scale EMV response simulation within New York's M6 dispatch area in West Harlem. This initial model serves not only as an essential building block toward achieving a comprehensive TDT but can itself be used by emergency response agencies for intervention testing within the smaller region, along with its related outputs which include a traffic simulation-informed explainable boosting machine (TSI-EBM) AI model, which we call EMVAID.

In Zuo et al. (2026), we focus on the development and calibration of the baseline TDT, and modeling EMV movement and behavioral responses of surrounding vehicles to generate proxy traffic features during dispatches. Zuo et al. (2026) answers how conventional traffic simulation tools can be used in support of transient emergency response scenarios. The current paper focuses on the development of an AI-driven traffic state prediction model for EMVs using the TDT, called EMVAID. It assesses the influence of the built environment on EMV dispatch performance and demonstrates the application of the TDT and EMVAID through a use case of supporting the agency's decision making in resource allocation. EMVAID is designed not only to replicate simulation outputs, but to serve as a scalable substitute for the simulation itself, with the potential to generate link-level EMV speed estimates in real time using only inputs available citywide, without requiring a locally calibrated microsimulation to be running continuously. In summary, this research answers the following questions:

- Can proxy traffic variables provide sufficient information to substitute the lack of real-time observations of downstream passenger vehicle behaviors as key components of emergency response digital twins?
- Can simulated proxy variables serve as important features for machine learning models

to predict EMV travel speeds?

- How can TDTs and AI models provide decision support for emergency response?

The rest of the paper is organized as follows. Section 2 presents an overview of the proposed framework. Section 3 explains the AI model development and the obtained results. Section 4 outlines a case study of cross street location optimization that demonstrates one of the potential applications of the developed framework. Finally, Section 5 concludes the study.

2 Overview of methodology framework

The overall methodology is shown in Figure 2. As the first step, a baseline microscopic simulation was developed using the open-source Simulation of Urban MObility (SUMO) (Lopez et al., 2018) for the study area located in upper Manhattan covering the West Harlem and Morningside Heights neighborhoods of New York City shown in Figure 3, functioning as a TDT for FDNY's M6 dispatch area. The model enriches the network with operational details (bus lanes, transit stops, parking zones, bike lanes) to better reflect street capacity and conflicts. Background traffic and control were calibrated with Simultaneous Perturbation Stochastic Approximation (SPSA) (Spall, 1988, 1999; Sha et al., 2023), then EMV-specific behaviors were tuned via a hybrid Evolutionary Algorithm–Reinforcement Learning (EA–RL) scheme discussed in Zuo et al. (2026). In this two-stage design, we first establish realistic background traffic and infrastructure conditions, then make EMVs and surrounding drivers behave plausibly around siren-activated events.

For background calibration, the baseline SUMO model was matched to 8:00–10:00 AM traffic counts at 63 intersections, third-party travel-time and speed data as shown in Figure 2, and targeted field videos used to review queues and operational conditions. The SPSA calibration reduced the combined background error from 23.98% to 16.54% for the best replication across six stochastic seeds. EMV behavior was then calibrated using FDNY AVL trajectories from 987 life-threatening trips in 2023; 300 trips with the richest trajectory coverage were selected, including 200 for calibration and 100 for validation. Although overall city-wide averages vary, a focused comparison using AirSage vehicle data of 460 spatiotemporally matched observations across both local and highway segments within the M6 region showed that EMVs can exceed general traffic by an average of 8 mph during travel to the scene and 17 mph during hospital transport for life-threatening incidents. Capturing these siren-induced dynamics is therefore essential for credible prediction and interpretation.

The evolutionary algorithm plus reinforcement learning (EA–RL) stage then targeted EMV

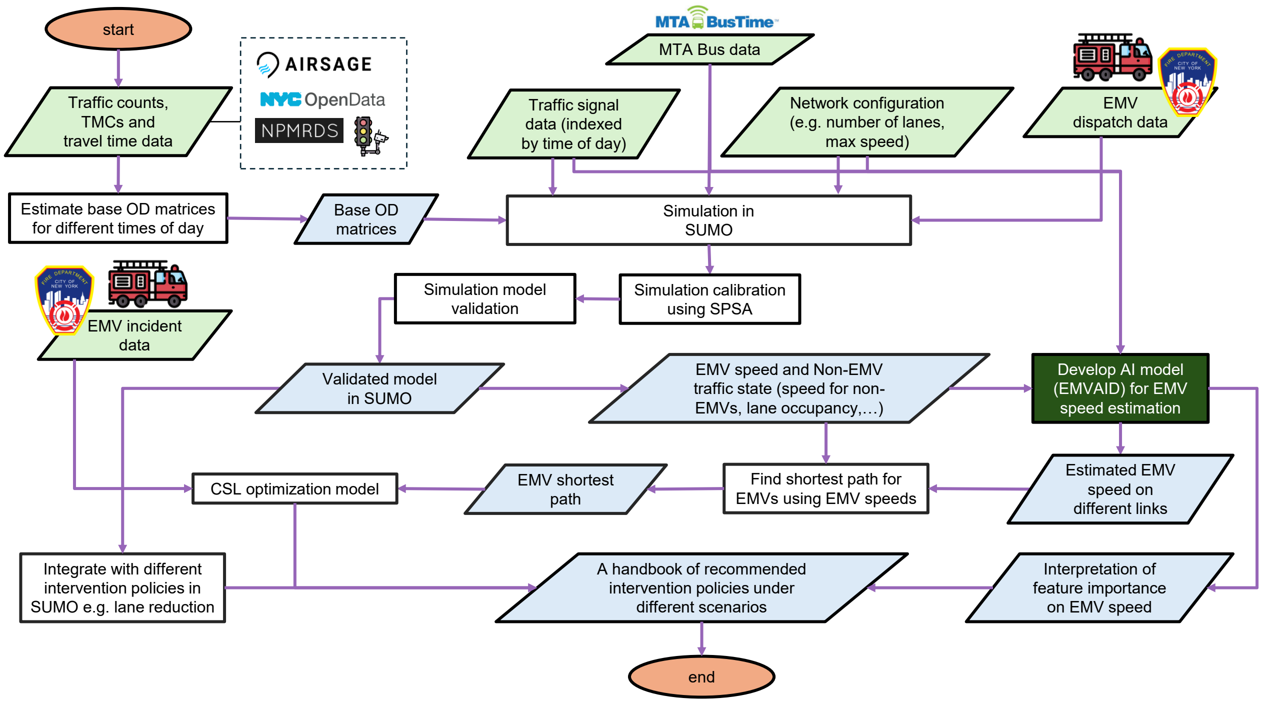


Figure 2: TDT and EMVAID methodology framework.

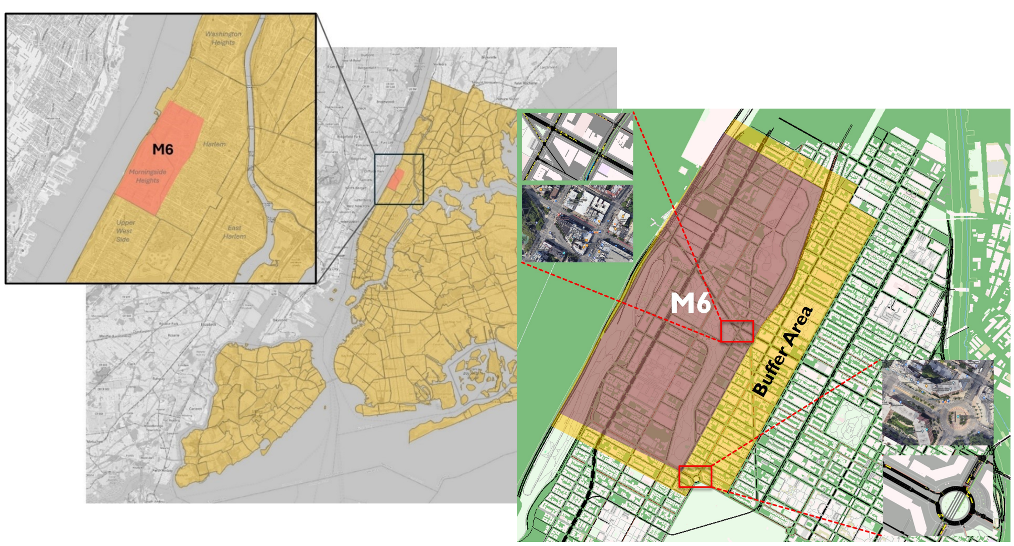


Figure 3: Study area and the SUMO simulated network details.

maneuvering and driver response, including sublane lateral moves, queue-jumping, and yielding propensities, which were tuned through parameters such as reaction distance, reaction rate according to distance, longitudinal/lateral minimum gaps, speed variation, and longitudinal/lateral acceleration behavior. Distribution-level convergence (not just point errors) was emphasized, using KL divergence (Csiszár, 1975), variance differences, and MAE over full trip-time distributions. On the 100-trip validation set, EA-RL achieved 36.8 ± 6.5 s MAE, 0.038 KL divergence, and 5.3×10^5 variance difference, improving on SPSA alone $(49.3 \pm 9.2s, 0.125, 9.1 \times 10^5)$ and EA only $(44.6 \pm 8.4s, 0.103, 6.4 \times 10^5)$. (for more details see Zuo et al. (2026)).

These validations support the use of the simulation as the basis for EMVAID, while also defining its limits. The main residual errors occur for trips affected by geometric coding issues, rare expressway or mid-trip operational changes, and non-recurrent disruptions such as closures or construction that are not represented in the baseline simulation. In addition, certain special EMV maneuvers, such as driving against traffic, remain difficult to model within current SUMO rules.

The calibrated simulation then supplies the AI stage with both static and dynamic features at link-level resolution for creating an interpretable artificial intelligence (AI) model, specifically a TSI-EBM model called EMVAID. Static attributes such as number of lanes, bike lane presence, and signal timing are readily available from existing data sources. The primary contribution lies in the dynamic feature layer, which includes background speeds, occupancies, queues, and delays aggregated in 5-minute windows over the two-hour analysis period, which are otherwise difficult or impossible to observe comprehensively during real dispatches. In plain terms, these features operationalize "how crowded and how constrained" each link is expected to be during an EMV's movement, including the reaction of the network to the EMV.

We argue that an AI model for EMV operations benefits materially from being simulation-informed. In short, a calibrated microscopic model provides us with dynamic, unobservable traffic states (e.g., link occupancies, queues, transient yielding) within the actual dispatch context; these states become features that improve both prediction and interpretation compared to using only readily available static or proxy data. Put simply, the simulation acts like a sensor for variables the field cannot observe in real time, yet that strongly shape how EMVs move through congested urban grids.

The resulting EMVAID model, described in Section 3, is then used to interpret how specific built environment features influence EMV speeds. The use of the TDT for decision support (per Figure 1) is demonstrated through using the outputs for predicting travel times in a Cross Street Location (CSL) optimization experiment, as described in Section 4.

3 AI model development

3.1 Background

EMVs behave differently from passenger vehicles in traffic; they can gain right of way through signalized intersections, their siren technologies can inform downstream vehicles to part for them to pass, etc. As such, typical navigation tools for predicting travel times like Google Maps may not be helpful. Unfortunately, real-time traffic features such as passenger vehicle speeds and road occupancies or queues are typically not collected during dispatches either. Even if they were, alternative route features would not be known without using newer technologies to monitor a city (Chow, 2016; Barmpounakis and Geroliminis, 2020; Yang et al., 2021; Sha et al., 2023).

Alternatively, steady state traffic variables during the time-frame corresponding to an EMV dispatch may serve as second-best proxies of actual traffic conditions, short of having access to the real observations. Some of these data are readily available, e.g. passenger vehicle speeds can be estimated over time using various sources (Jenelius and Koutsopoulos, 2013; Zhan et al., 2013; Guo et al., 2019). Other features like vehicle occupancies or congestion levels can also be estimated from such data (e.g. (Meng et al., 2017)) or by combining the travel speed data with traffic simulations to rely on the outputs for synthesized features as we have done.

Prior work in this area, such as the EMVLight model from Su et al. (2023), uses reinforcement learning as an optimization tool for route guidance and traffic control, not for prediction using real data. Emergency response data is often limited to vehicle-specific information collected during dispatches, without detailed insight into surrounding traffic conditions such as downstream congestion or intersection queues, that can significantly impact emergency vehicle speeds. To help fill this gap, we propose an interpretable artificial intelligence (AI) model called EMVAID that is informed by a calibrated traffic simulation to supplement dispatch data with synthesized background traffic data. The TSI-EBM model for predicting emergency vehicle speeds can be used to interpret the built environment features that contribute to the speed prediction and feed into various decision support tools.

The explainable boosting machine (EBM) is a tree-based, cyclic gradient boosting generalized additive model, developed to be as accurate as state-of-the-art black box models while

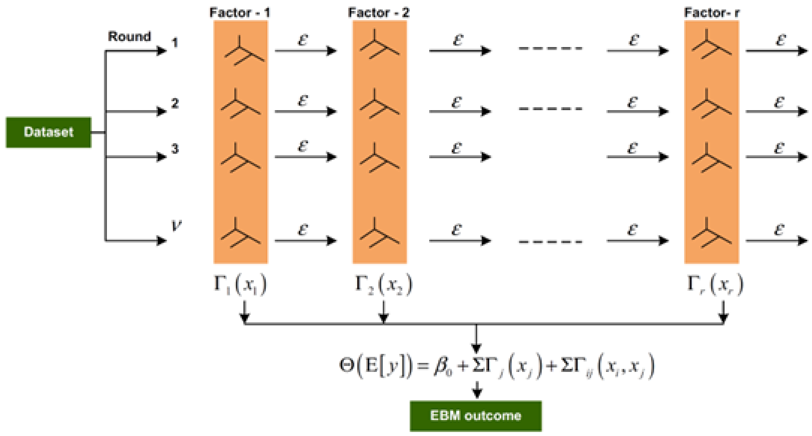


Figure 4: Illustration of EBM structure (Khattak et al., 2023).

remaining interpretable (Lou et al., 2013). The structure of the EBM is highlighted in Figure 4 (from Khattak et al. (2023)), where x_j is the jth feature, y is the target variable, θ is the link function that adapts the generalized additive model (GAM) for regression, β_0 is the intercept term, and Γ_j is the attribute function. Training in EBM is performed repeatedly, with small trees built sequentially in each round. Each tree uses only one feature, and the residual (ϵ) is updated afterward. Then, all the trees for each feature are summarized into a graph, providing an intuitive visualization of how the model works. In this model, selected terms of interacting pairs of features are detected and added to standard GAMs, making the model more powerful than GAMs, but still intelligible as the two-dimensional interactions can still be rendered as heatmaps. Lou et al. (2013) developed an efficient FAST method for ranking all possible pairs of features as candidates for inclusion into the model. To maintain the additivity of individual terms, EBMs incur a higher training cost, which can make them slower to train compared to similar models. However, they are among the fastest and most compact models at runtime.

We propose a TSI-EBM where certain features are obtained as an output of calibrated simulation of observed dispatch data to reflect steady state traffic conditions corresponding to those dispatch times. The output is a prediction of the EMV travel speed for a given edge in the arterial network. With EMVAID, we can further quantify and rank the different impacts that the features have on EMV response times. The model can also be used to predict travel times along each edge to output scenario-based travel times.

3.2 Data preprocessing and descriptive analysis of features

Edge-based average EMV speeds estimated in the SUMO model were used as input data for the TSI-EBM, along with other extracted edge-level static and dynamic features explained

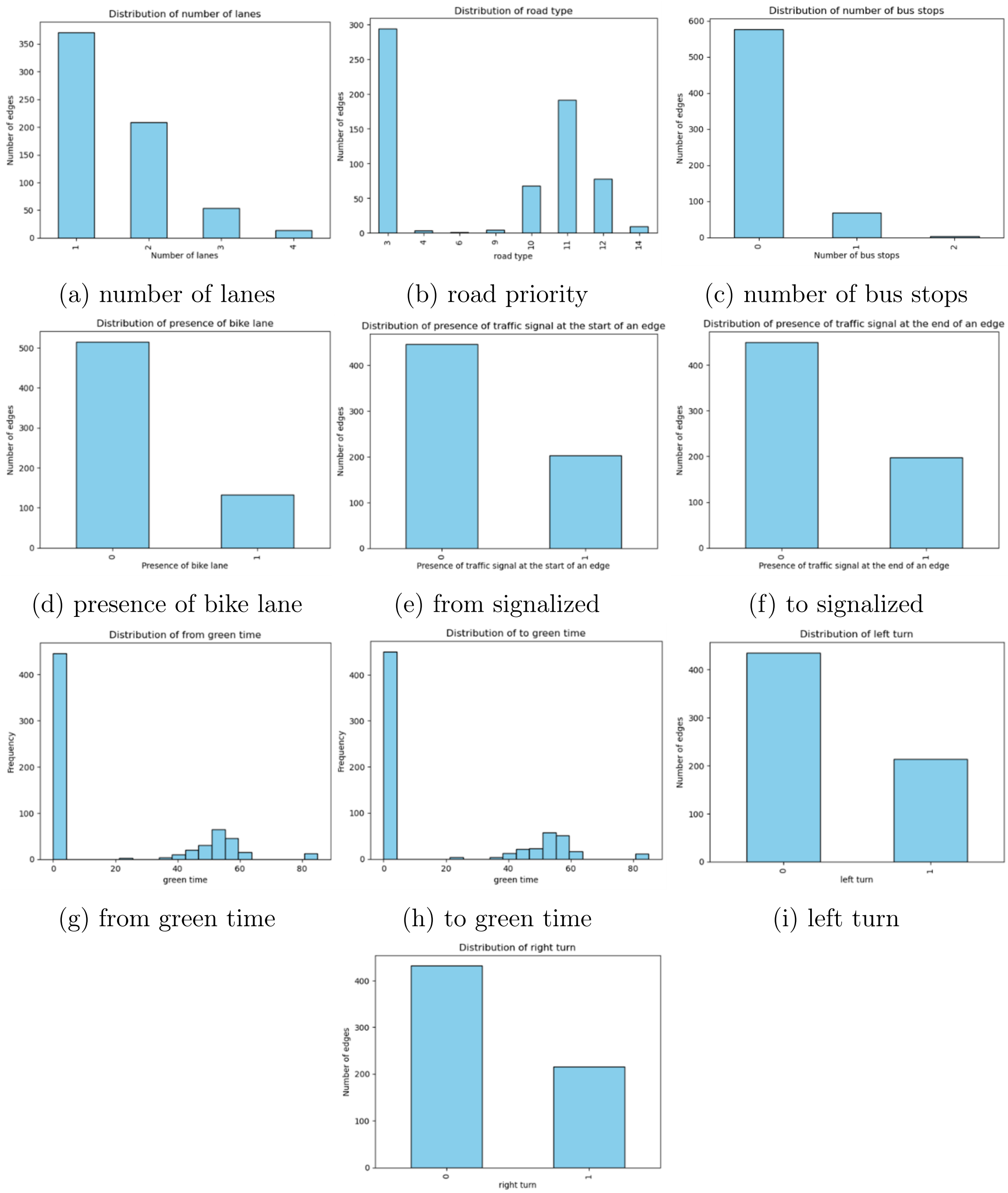
Table 1: Static features

Feature	Description
Number of lanes	
Road priority	A value that defines right-of-way priority at junctions, starting at 1, where higher values indicate roads with higher priority
Number of bus stops on the edge	
Presence of bike lane on the edge	All bike lanes are unprotected in the study area
From signalized	1 if there is a signalized intersection at the start of an edge, 0 otherwise
To signalized	1 if there is a signalized intersection at the end of an edge, 0 otherwise
From green time	Green time in seconds if there is a signalized intersection at the start of an edge, 0 otherwise
To green time	Green time in seconds if there is a signalized intersection at the end of an edge, 0 otherwise
Right turn	1 if right turn is allowed at the end of an edge, 0 otherwise
Left turn	1 if left turn is allowed at the end of an edge, 0 otherwise

below. Edges shorter than 10 meters were excluded for model training as dynamic data is not captured accurately if an edge is traveled in less than a second in SUMO with the settings used for the simulation. These micro-segments are artifacts of the SUMO network and generally represent short connector segments at or near intersections where vehicles pass through in under one simulation timestep, resulting in unreliable dynamic outputs. This exclusion does not bias the representation of intersection-heavy segments, as the relevant traffic dynamics are captured on the longer upstream approach edges retained in the dataset. Features including queue length, to signalized, and from/to green time on these edges ensure that intersection-level effects remain fully represented in the model. There are 629 edges within the study area with dynamic information after excluding short edges. Static features are presented in Table 1, and their distributions are shown in Figure 5.

Traffic signal cycle length was not used as a feature since the cycle length for all traffic signals within the study area is 90 seconds. Similarly, bike lane width was also excluded as the width of the bike lane is similar for all bike lanes in the simulated study area. The speed limit is 25 mph for the considered edges in the study area. Therefore, it is not used as a feature. Dynamic features are presented in Table 2, and their distributions are shown in Figure 6. To mitigate the effect of the inherent stochasticity of microsimulation, the SUMO model is run for multiple times with 5 different random seeds.

The dynamic features are extracted from the edge-level outputs of the model, which are aggregated for 5-minute time intervals across the two-hour simulation period (the SUMO simulation models traffic conditions between 8 AM and 10 AM). The 5-minute window was selected to align with the resolution of commercially available real world traffic data sources, which represent the most realistic inputs available to emergency response agencies at a city-wide scale. Furthermore, the FDNY average emergency response travel time in 2023 was 6.08 minutes. Because the 5 minute window is just under this average, it effectively captures the prevailing traffic conditions for the duration of a typical response trip without aggregating over too long of a period. While finer temporal resolutions are extractable from SUMO, sub minute aggregations introduce greater noise in microsimulation outputs and would require



(a) number of lanes (b) road priority (c) number of bus stops

(d) presence of bike lane (e) from signalized (f) to signalized

(g) from green time (h) to green time (i) left turn

(j) right turn

Figure 5: Distribution of static features.

Table 2: Dynamic features

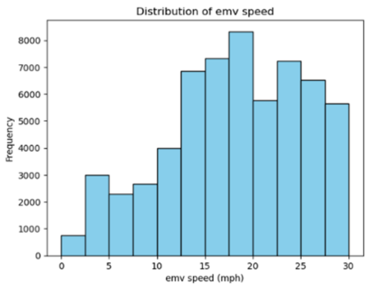
Feature	Description
Background traffic speed	The space mean speed on the edge within the reported interval.
Occupancy	Average 5-minute occupancy of the edge within the reported interval (in %). A value of 100 would indicate vehicles standing bumper to bumper on the whole edge.
Density	Average vehicle density of the edge within the reported interval.
Queue length	The length of queue measured from the junction until the final vehicle in line at each timestep. Three statistics are reported for each reported interval, i.e., average, max value and 75th percentile.
Queue time	The total waiting time of vehicles due to a queue at each timestep. Three statistics are reported for each reported interval, i.e., average, max value and 75th percentile.
Simulation time	Starting time of the measured 5-minute interval (in seconds). Various discretizations were tested for this specific feature, including 15-minute, 30-minute, and 1-hour intervals for model training.

data feeds not currently available for network wide deployment. Intersection level delays, which operate at shorter timescales, are partially addressed through the inclusion of signal timing attributes, specifically green time and signalization status, as static features in the model. There are in total 60,384 instances of edges with dynamic information from SUMO after excluding the short edges.

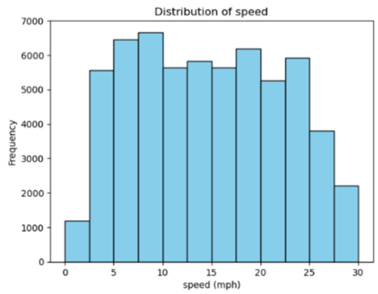
Simulated rather than directly observed link-level speeds were used as the training target because many individual links have insufficient observed trips to produce reliable average speeds for the specific time period and background traffic conditions being modeled. Specifically, only 69.6% of links within the M6 region have an AVL observation from severity level 1-3 incidents for weekdays between 8-10 AM, with a median of only four observations per link, leaving gaps in spatial coverage that the calibrated SUMO model resolves by generating a consistent, spatially complete speed estimate for every link in the network. This approach is valid to the extent that simulated speeds credibly represent real-world EMV behavior. As described in Section 2, the SUMO model was calibrated directly against observed FDNY AVL trajectories with distribution-level convergence evaluated via KL divergence, variance differences, and MAE over full trip-time distributions. This design also serves as a proof-of-concept for a citywide deployment in which the EMVAID, once trained, could generate real-time link-level speed estimates using only available inputs.

3.3 Model training and results

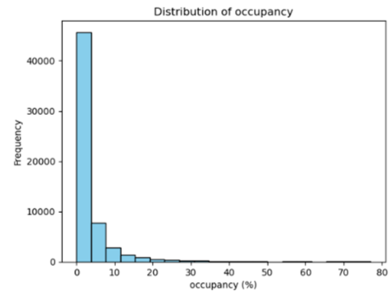
InterpretML library in Python (Nori et al., 2019) was used for developing the EBM model. All experiments were conducted on a laptop with an Intel Core i7-1260P CPU (12 cores), 16 GB RAM, and no dedicated GPU, running Windows 11. The performance of the EBM



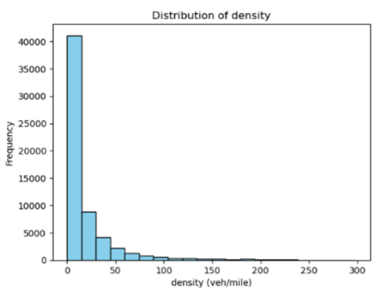
(a) EMV speed



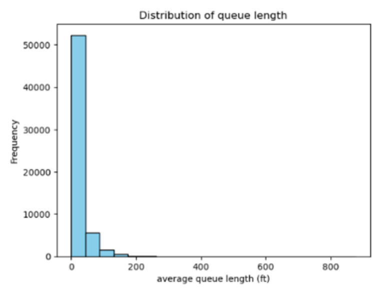
(b) background traffic speed



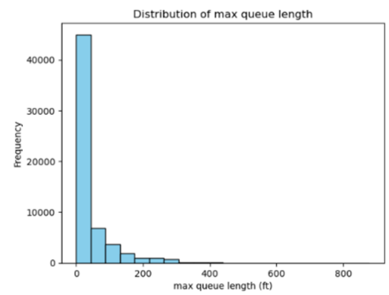
(c) occupancy



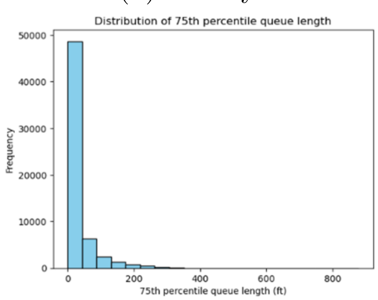
(d) density



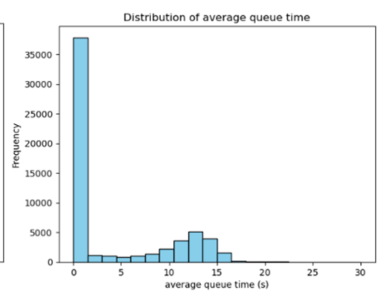
(e) average queue length



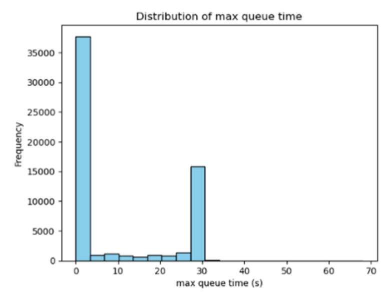
(f) max queue length



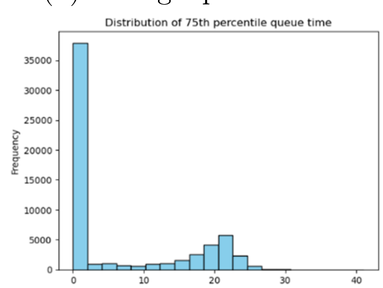
(g) 75th perc. queue length



(h) average queue time



(i) max queue time



(j) 75th perc. queue time

Figure 6: Distribution of target variable and dynamic features.

model is compared to three other models including decision tree, random forest, and XGBoost. Additionally, to assess the value of simulation-informed features and to address the operational feasibility of the framework, two baselines were compared. EBM - baseline (i) uses only realistically obtainable proxy features including 5-minute background traffic speeds and static network attributes. This represents a model that could be deployed using only commercially available data sources. EBM - baseline (ii), similar to the rest of the models, adds simulation-informed dynamic features, such as occupancy, which capture the more localized congestion that is triggered by an EMV during a dispatch.

To avoid potential information leakage due to repeated observations across stochastic realizations, we adopt a leave-one-seed-out cross-validation approach, where all samples generated from a given stochastic seed are held out during testing. Observations are grouped by seed because each seed represents a distinct stochastic realization of the system, and observations within a seed are inherently correlated. For each fold, model hyperparameters (listed in Table 3) are tuned via Bayesian optimization over 50 iterations, combined with grouped cross-validation applied exclusively to the training seeds. The objective is to minimize mean squared error, while ensuring that no observations from the same seed appear in both the training and validation splits during tuning. The final model in each fold is then evaluated on the held-out seed. This procedure ensures that no correlated samples from the same realization appear in both training and test sets. We report the mean and standard deviation of model performance across all folds in Table 4. The selected performance metrics include root mean squared error (RMSE), mean absolute percentage error (MAPE) and R^2.

Bayesian Optimization is a strategy for efficiently tuning hyperparameters of machine learning models, especially when evaluations are expensive or noisy (Jones et al., 1998; Frazier, 2018). It works by building a surrogate model, typically a Gaussian Process, that approximates the relationship between hyperparameters and model performance. An acquisition function then uses this surrogate to suggest the next set of hyperparameters to evaluate, balancing exploration of new regions and exploitation of known good areas. After each evaluation, the surrogate model is updated with the new result, and the process repeats.

Since some of the features are correlated, different combinations of features were tested to ensure interpretability. The selected feature combination used to train all models, except EBM - baseline (i), includes background traffic speed, occupancy, presence of bike lane, number of lanes, road type, number of bus stops, simulation time (15-minute intervals), left turn, right turn, from signalized, to signalized, from green time, and to green time.

The test results in Table 4 show that adding simulation-informed features in EBM – baseline

Table 3: Hyperparameter tuning using Bayesian optimization

EBM	
Hyperparameter	Range
Max_leaves	{2,3}
Smoothing_rounds	[350 ,1000]
Learning_rate	[0.02, 0.2]
Decision tree	
Hyperparameter	Range
Max_depth	[5,30]
Min_samples_split	[2,20]
Min_samples_leaf	[1,10]
Max_features	{1.0, sqrt, log2, None}
Min_impurity_decrease	[0.0,0.5]
Ccp_alpha	[0.0,0.1]
Random forest	
Hyperparameter	Range
N_estimators	[100,800]
Max_depth	[10,50]
Min_samples_split	[2,10]
Min_samples_leaf	[1,4]
Max_features	{sqrt, log2, None}
Bootstrap	{true, false}
XGBoost	
Hyperparameter	Range
N_estimators	[100,500]
Max_depth	[3,7]
Learning_rate	[0.01, 0.3]
Subsample	[0.5,1.0]
Gamma	[0.0, 0.3]
Reg_alpha	[0.01,100]
Reg_lambda	[0.01,100]

Table 4: Comparison of the performance of different models

Metric	EBM - baseline (i)	EBM - baseline (ii)	Decision tree	Random forest	XGBoost
RMSE (mph)	5.30 ± 0.27	5.17 ± 0.30	5.30 ± 0.21	5.01 ± 0.29	5.06 ± 0.31
MAPE (%)	37.04 ± 3.84	34.52 ± 4.39	36.53 ± 3.54	33.64 ± 4.25	33.94 ± 4.18
R^2	0.43 ± 0.05	0.46 ± 0.05	0.43 ± 0.03	0.49 ± 0.05	0.48 ± 0.05
Train time (sec)	11110 ± 2695	9976 ± 2453	157 ± 6	2386 ± 336	222 ± 5
Test time (sec)	0.0081 ± 0.0021	0.0110 ± 0.0013	0.0037 ± 0.0002	1.7306 ± 0.5938	0.0339 ± 0.0026

Table 5: Robustness analysis under grouped validation by link

Metric	EBM - baseline (ii)	Decision tree	Random forest	XGBoost
RMSE (mph)	5.08 ± 0.08	5.13 ± 0.07	5.02 ± 0.07	5.02 ± 0.02
MAPE (%)	34.07 ± 2.37	34.43 ± 2.51	34.18 ± 2.90	33.75 ± 2.76
R^2	0.48 ± 0.03	0.47 ± 0.03	0.49 ± 0.02	0.49 ± 0.02

(ii) compared to EBM – baseline (i) yields a modest improvement in predictive performance, with RMSE decreasing from 5.30 to 5.17 mph and MAPE from 37.02% to 34.52%. While the magnitude of improvement is relatively small, it is consistent across all presented performance metrics, suggesting a stable but incremental benefit from incorporating simulation-informed features. While random forest and XGBoost offer the best performance, EBM - baseline (ii) outperforms decision tree and comes close to these top-performing models. Importantly, EBM's slightly lower prediction accuracy is offset by its considerable advantage in interpretability. The root mean squared error for this model remains around 5 mph. The training and testing times for all models are also reported in the table. The EBM requires substantially longer training time but maintains fast inference speed comparable to decision trees, whereas models such as random forest, while achieving strong predictive performance, incur higher computational cost during testing.

To further evaluate the sensitivity of model performance to potential spatial correlations, we conduct an additional robustness analysis using grouped cross-validation by link. In this setup, all observations associated with a given road segment are assigned exclusively to either the training or testing set within each fold. This validation strategy evaluates the ability of the models to generalize to unseen links and provides a stricter partitioning scheme than the primary leave-one-seed-out evaluation. To isolate the effect of the grouping strategy, the same hyperparameter configuration obtained from the primary grouped tuning procedure is retained for all models during the robustness analysis.

Table 5 presents the robustness analysis results under grouped validation by link. Overall, the qualitative conclusions remain consistent with the primary leave-one-seed-out evaluation. Ensemble-based models continue to achieve the strongest predictive performance, while the Explainable Boosting Machine (EBM) maintains improved performance relative to the decision tree model.

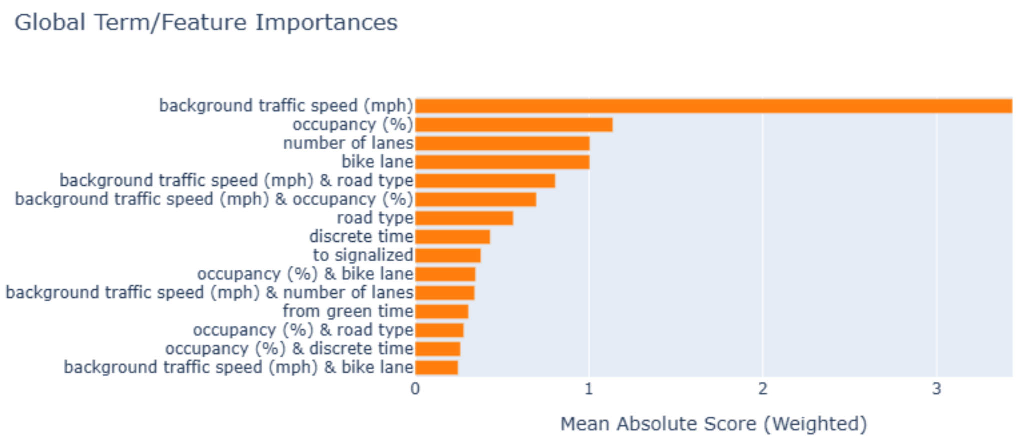


Figure 7: Global ranking of feature importance in EMVAID.

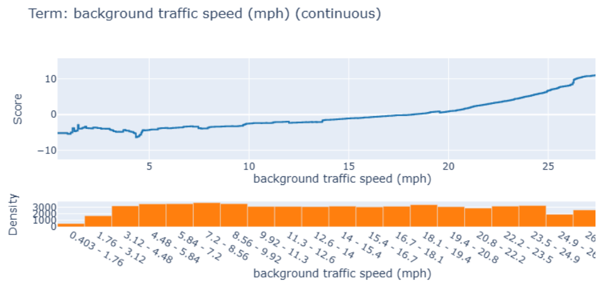
It is also noteworthy that grouped validation by link yields performance comparable to, and in some cases slightly better than, leave-one-seed-out validation for certain models. This behavior likely reflects the relatively localized spatial extent of the study area and the limited spatial variability across neighboring links, whereas variability across stochastic simulation seeds introduces greater differences in traffic conditions.

The additive structure of the EBM model enables global ranking and visualization of both individual feature contributions and pairwise interactions. It allows for the examination of how different feature combinations contribute to edge EMV speed prediction. For the final EBM–baseline (ii) model used for interpretation, hereafter referred to as the EBM model, hyperparameters are re-tuned on the full dataset using grouped cross-validation by seed.

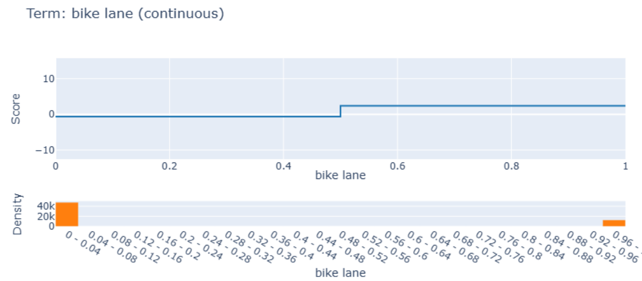
Figure 7 shows the global ranking of feature importances (top 15), including individual features and their pairwise interactions. It suggests that the background traffic speed has the most importance in predicting EMV speed. This result answers two of our research questions: indeed, steady state traffic metrics combined with simulation can result in synthetic data as proxies of real-time observed traffic speeds to inform on the prediction of EMV travel speeds.

Edge occupancy is about 1/3 of the importance, followed by number of lanes and presence of bike lane, respectively. It is worth noting that many bike lanes in the study area are conventional in design (e.g., without physical separation from vehicle lanes), which may offer additional usable space for emergency vehicles when needed. The graphical visualization of the contribution of the top four important features in EMV speed is shown in Figure 8.

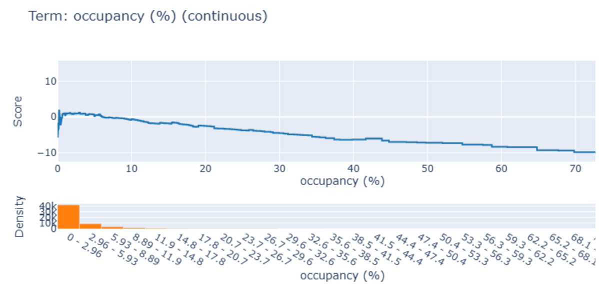
To quantify uncertainty in the learned feature effects, we employed a nonparametric bootstrap



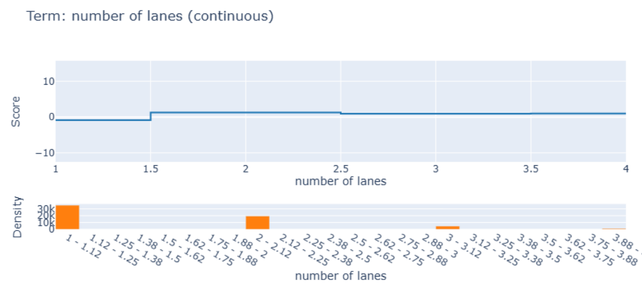
(a) background traffic speed



(b) bike lane



(c) occupancy (%)

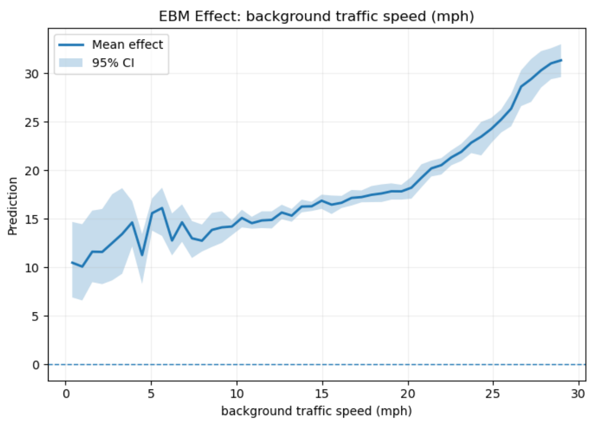


(d) number of lanes

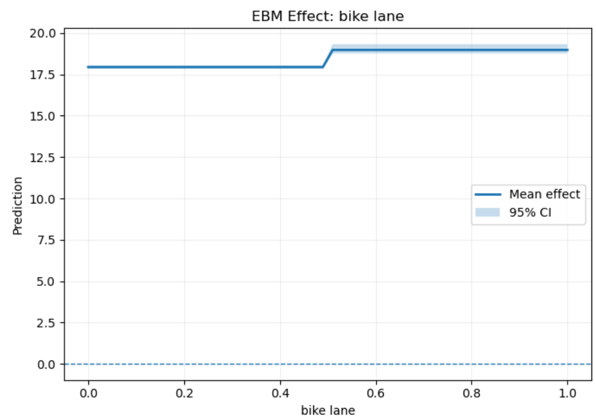
Figure 8: EMVAID shape functions for (a) background traffic speed; (b) bike lane; (c) occupancy; (d) number of lanes.

procedure, repeatedly refitting the EBM on resampled datasets. Each bootstrap sample was generated by sampling with replacement from the original dataset, preserving the original sample size. Feature effects were evaluated as partial dependence functions on a fixed grid, and pointwise 95% confidence intervals were computed from the empirical distribution of bootstrap estimates. The partial dependence plots with bootstrap confidence intervals shown in Figure 9, largely confirm the trends observed in the EBM shape functions, indicating that the learned relationships are robust to sampling variability.

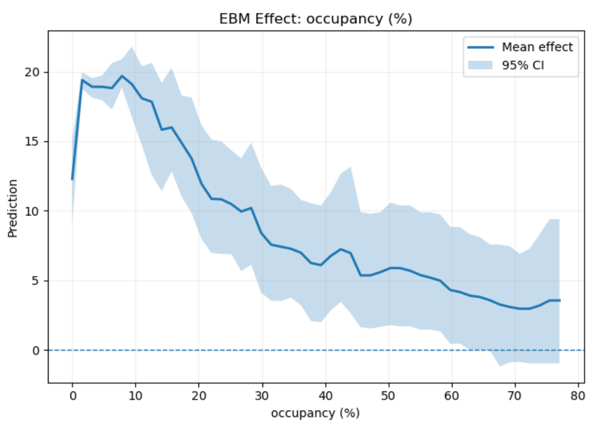
As shown in Figure 8(a), there is a nonlinear increasing trend in contribution of background traffic speed in EMV speed, where higher passenger traffic speeds are associated with increased EMV speeds. The nonlinearity suggests that simple linear models may not fully capture this relationship. Additionally, the model indicates a negative contribution of passenger vehicle speeds to EMV speed when traffic speeds fall below approximately 16–17 mph. This suggests that under highly congested conditions, nearby traffic may act more as a constraint on EMV movement, potentially requiring passenger vehicles to pull over and stop before EMVs can progress efficiently. While this pattern is consistent with operational intuition regarding transitions between congested and free-flow conditions, the observed transition region may be sensitive to the composition of the training data. The confidence intervals in



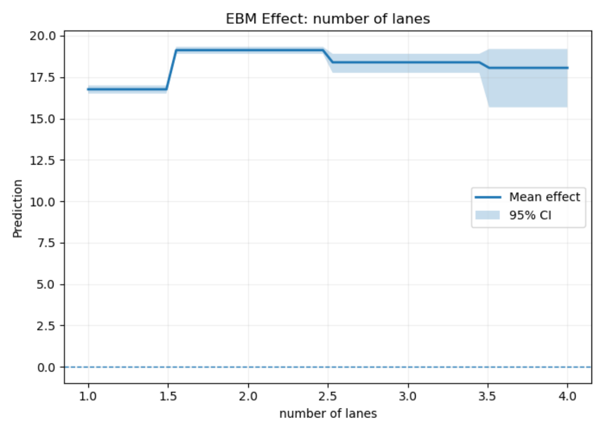
(a) background traffic speed



(b) bike lane



(c) occupancy (%)



(d) number of lanes

Figure 9: Partial dependence function for (a) background traffic speed; (b) bike lane; (c) occupancy; (d) number of lanes, evaluated on a fixed grid. Shaded regions denote pointwise 95% confidence intervals computed from bootstrap resampling.

Figure 9(a) suggest that the model is most reliable at medium and high speeds.

Figure 8(b) indicates a positive association between the presence of non-parking protected bike lanes and EMV speed. This relationship is consistent with the hypothesis that bike lanes provide additional maneuvering space, though it may also reflect correlations with other network design features, such as road type or link hierarchy, where higher-priority roads may already have greater available width. In Figure 8(c), a downward trend in the impact on EMV speed is observed as edge occupancy increases, suggesting that higher occupancy levels negatively impact EMV travel. The concentration of observations in the low range of the feature aligns with the narrower confidence intervals shown in Figure 9(c), indicating that the model is most reliable in these regions and less reliable in higher-value regions.

The effect of the number of lanes, illustrated in Figure 8(d), appears to be more subtle. A more noticeable improvement in EMV speed occurs when the number of vehicle lanes increases from one to two, likely because this allows passenger vehicles to create space for EMV passage. Beyond two lanes, additional lanes yield minimal benefit. However, as shown in Figures 5(a) and 9(c), links with three or four lanes are underrepresented in the dataset, which limits the reliability of the model at higher lane counts.

The model provides insights for operational policies in selection and routing of EMVs. In general, a shortest path search using conventional routing tools like Google Maps could provide helpful candidates for routing a vehicle. However, routes that go through single lane streets or lanes that have historically low speeds at those times of day should be avoided. Non-protected bike lanes offer extra space for maneuvering or allowing downstream vehicles to pull over.

Counterfactual scenarios can also be evaluated using this model. For example, if agencies wish to propose changing bike, bus, or parking lanes into additional lanes or vice versa, EMVAID can be used to update the predicted travel times for candidate routes.

In addition to offering interpretability, this model can be leveraged to estimate EMV travel times across different edges within a network, provided that relevant background traffic data is available. Since developing microsimulation for large-scale networks can be both time-consuming and resource-intensive, this model presents a more efficient alternative. It can be applied to similar regions where the necessary data exists, thereby reducing or even eliminating the need to build complex simulations. For example, these insights are transferable to a larger city environment without simulations by using time-dependent passenger vehicle speed data and inferring road occupancies for those road segments by time of day. This makes the model particularly valuable for large-scale planning in emergency response scenarios.

4 TDT decision support illustration: CSL optimization

To complete the TDT feedback loop, we illustrate the use of the TDT and AI model in providing inputs for an example use case: ambulance location planning. EMVs typically begin their tours from designated cross street locations (CSLs), with each vehicle assigned to a specific CSL. Generally, EMVs are expected to return to their CSL once they have completed responding to an incident. However, there might be cases when they are redirected to new incidents along the way. The location and configuration of CSLs can influence response times, making it valuable to identify an optimized CSL setup that helps reduce delays and improve overall operational efficiency.

The problem of choosing p facilities among a number of candidate locations is regarded as a facility location problem in transportation and logistics, called a p-median problem (see Owen and Daskin (1998)) in which the average time traveled by EMVs from CSLs to incident locations is minimized. One of the required inputs for solving this problem is the access time from a resource based in the facility to the incident location (noting that an assigned EMV may start its trip somewhere away from a CSL closest to the incident, and may not even be based at that CSL). As such, the access time reflects an expectation of random travel times from those assignments that gravitate around the located CSL even if they don't always start from there.

Travel time data from the CSL to the incident location can be used as a proxy for measuring this access time. However, EMV travel time can differ significantly from passenger travel times. This difference can lead to a different shortest path from an origin to a destination for an EMV compared to a passenger vehicle. However, due to data limitations for EMV travel times on different edges of the network, passenger travel times are more generally used for such models. One of the promising applications of the SUMO simulation is the capability of extracting EMV travel time for different edges on the network. Using this data, it is possible to get a more accurate solution for finding the optimal set of CSLs for EMVs.

Another required input for this model is demand, i.e. number of incidents at different locations within the area. Input data preparation, including the set of candidate facilities, demand analysis, and the origin-destination (OD) travel times are presented in the next section. In the base p-median problem, it is assumed that EMVs are always available at CSLs to respond to incidents. However, as mentioned above, EMVs might be responding to other incidents when a new request comes in. Therefore, they may not always be available at CSLs. Daskin's (Daskin, 1983) maximum expected covering location problem (MEXCLP) accounts for facility unavailability, or its p-median extension (Snyder and Daskin, 2005). Similarly, Marianov and ReVelle (1996) consider server-level co-location with queueing constraints incorporated in a piecewise linear fashion to obtain a tractable mixed integer program, MALP, and its corresponding p-median extension (Marianov and Serra, 2002). In Section 4.2, a reliability approach is presented for the p-median problem based on Snyder and Daskin (2005) which addresses this limitation by considering the expected unavailability of units at each CSL, which is calibrated to observations of percentage of responses coming from units within M6.

4.1 Data preparation

4.1.1 Candidate CSLs

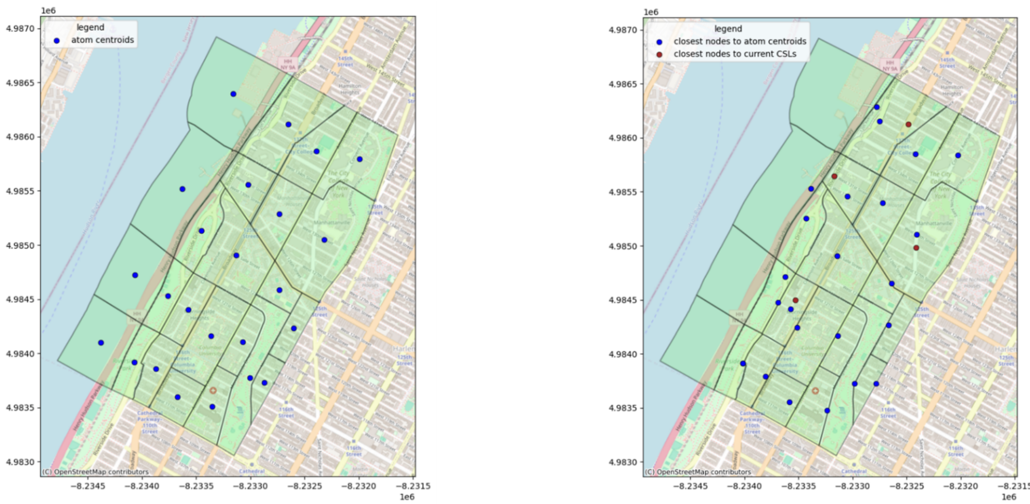
FDNY divides the city into small geographic areas known as *atoms*, which help inform recommendations for the nearest available units to respond to calls and guide units transporting patients to nearby hospitals. There are 24 atoms within M6 which are shown in Figure 10. The candidate CSLs are chosen according to atoms, i.e. each atom centroid is assumed to be a candidate CSL. Additionally, the existing four CSLs within M6 are added to the candidate set. Since the SUMO road network is used for navigation, each of these locations are assigned to the closest node on the network using Euclidian distance, leading to 27 candidate locations in total. Figure 10 shows atom centroids, current CSLs used by FDNY, and the final set of candidate CSLs assigned to the closest network node. The SUMO road network is shown in Figure 11(a), which includes 537 nodes and 976 edges. The largest strongly connected component of the network is used for all the analysis conducted in this section shown in Figure 11(b). This network has 510 nodes and 945 edges. The strongly connected component ensures the possibility of going from each node on the network to all the other nodes.

4.1.2 Demand analysis

The incident data within the M6 dispatch area in year 2023 was used as the demand input. The following steps were taken to preprocess the incident data:

1. Incident records missing incident datetime were excluded.

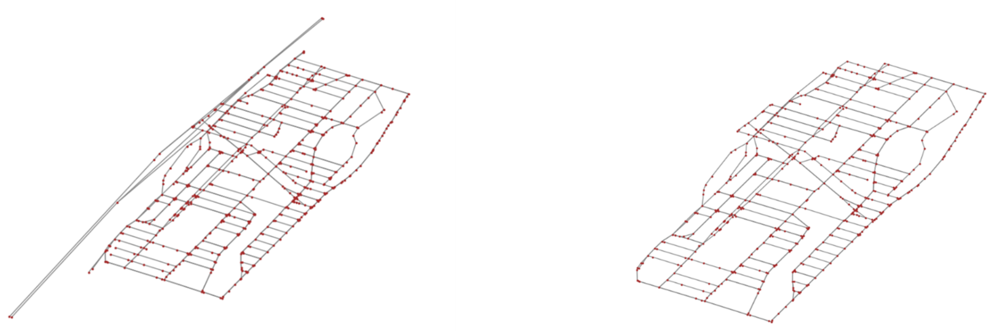
2. Different severity levels are assigned to each incident. Segment 1-3 represent life-threatening incidents for which units respond in emergency mode using both siren and light. Segment 4-8 are non-life-threatening incidents. Each call is assigned an initial severity level code and a final severity level code after the unit has responded to the call. Life-threatening incidents are the primary focus of this analysis. Therefore, only incidents with an initial or final severity level between 1 to 3 were included for the



(a)

(b)

Figure 10: Atoms within M6 (a) Atom centroids; (b) candidate CSLs.



(a) M6 network

(b) largest strongly connected component

Figure 11: SUMO network.

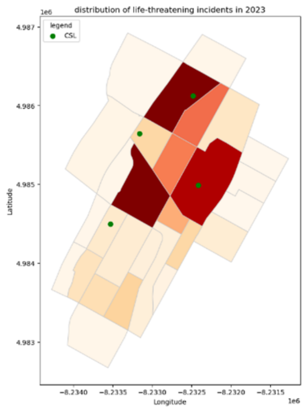


Figure 12: Spatial distribution of life-threatening incidents within M6 in 2023.

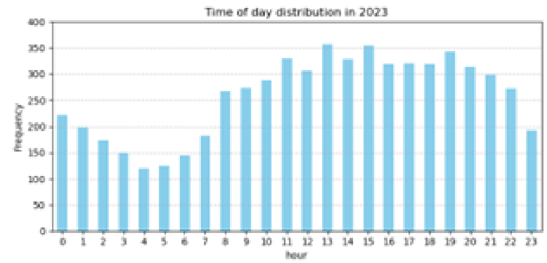


Figure 13: Time of day distribution of life-threatening incidents within M6 in 2023.

analysis.

3. Additionally, only incidents with an incident dispatch area designated as M6 were included.

There are 6200 life-threatening incidents within M6 in 2023 after implementing the data preprocessing steps. Figure 12 shows the spatial distribution of life-threatening incidents across different atoms within M6. Time-of-day incident distribution in Figure 13 reveals that demand peak typically starts around noon and continues into the evening. However, since the SUMO simulation models traffic conditions between 8 AM and 10 AM, we focus on the same period and use the corresponding demand, which includes 541 incidents.

Similar to the preparation of candidate CSLs, all incidents are assigned to the closest network node using Euclidean distance. Distribution of demand in 2023 from 8–10 AM across the network is shown in Figure 14 where larger nodes indicate a higher demand.

4.1.3 OD travel times

Another required input for the CSL optimization model is the travel time from each candidate CSL to all network nodes. As mentioned above, there are 27 candidate CSLs and 510 network nodes shown in Figure 10(b) and Figure 11(b), respectively. The Dijkstra algorithm (Schrijver, 2012) was used to find the shortest path on the network from each origin to each destination. As mentioned earlier, travel times were extracted from the SUMO simulation, which models traffic conditions between 8 AM and 10 AM. The average EMV travel time on each edge during this period was used as the edge's cost in the shortest path calculation. Average intersection travel times, often missing from default SUMO outputs, are also incorporated for improving the accuracy of Origin-Destination (OD) travel times. Due to the lack

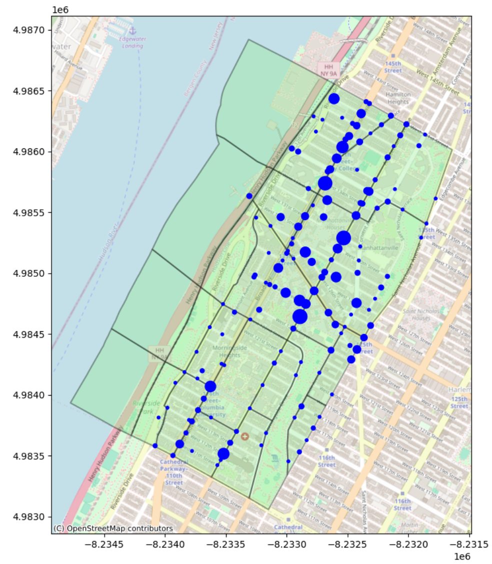


Figure 14: Demand distribution within M6 during 8 AM to 10 AM.

of intersection-specific observations, constant average delays were applied across the network for EMV and passenger vehicle movements.

4.2 Reliability-based P-median approach

4.2.1 Background

In the standard p-median approach, it is assumed that units are always available at CSLs, which may not always be the case in practice. In this section, we adopt the reliability-based p-median approach proposed by Snyder and Daskin (2005), which accounts for random facility failures, i.e. the congestion effect or unavailability of EMVs at CSLs due to being busy serving other incidents. We adapt the formulation to suit the specifics of the current problem. The objective of this approach is to minimize the expected failure cost, considering that facilities may fail with a given probability and that multiple facilities can fail simultaneously. It also allows certain facilities to be designated as not subject to failure.

Assuming incidents (demand) occur on network nodes and the candidate CSLs also lie on network nodes, we can formulate the problem as an integer program for set of nodes I with demand h_i, set of nodes J as candidate CSLs which is a subset of I, and shortest distances between candidate CSLs and network nodes determined a priori as d_{ji}, and facility budget of P. To incorporate facility failures, let F denote the set of facilities that are subject to failure and NF denote the set of facilities that are not subject to failure, such that $NF \cup F = J$. Each facility in F has a failure probability q, representing the long-term proportion of time the facility is unavailable for responding. X_j is the decision variable for location selection, which is equal to 1 if a facility is opened at node j, and 0 otherwise. The formulation assigns each demand node to a primary facility that serves under normal conditions, along with a sequence of backup facilities to serve the demand if the primary facility fails. Since multiple failures can occur simultaneously, each demand node must have a first backup facility in case the primary fails, a second backup if the first backup also fails, and so on. Therefore, a third dimension is added to the assignment decision variables (Y_{jir}). Y_{jir} is equal to 1 if demand node i is assigned to facility j as a level-r assignment. When $r = 0$, the assignment is considered primary; otherwise, it is a backup assignment. Each demand node i has a level-r assignment for every $r = 0, \ldots, P-1$, unless the demand node is assigned to a level-s facility that is not subject to failure where $s < r$ (Snyder and Daskin, 2005). The problem formulation is presented in Eqs. 1-8.

$$\min Z = \sum_{i \in I} h_i \left(\sum_{j \in NF} \sum_{r=0}^{P-1} d_{ji} q^r Y_{jir} + \sum_{j \in F} \sum_{r=0}^{P-1} d_{ji} q^r (1-q) Y_{jir} \right) \tag{1}$$

S.t.

$$\sum_{j \in J} Y_{jir} + \sum_{j \in NF} \sum_{s=0}^{r-1} Y_{jir} = 1, \forall i \in I, r = 0, \ldots, P-1 \tag{2}$$

$$Y_{jir} \leq X_j, \forall i \in I, \forall j \in J, r = 0, \ldots, P-1 \tag{3}$$

$$\sum_{j} X_j = P \tag{4}$$

$$\sum_{r=0}^{P-1} Y_{jir} \leq 1, \forall i \in I, \forall j \in J \tag{5}$$

$$X_u = 1 \tag{6}$$

$$X_j \in \{0, 1\}, \forall j \in J \tag{7}$$

$$Y_{jir} \in \{0, 1\}, \forall i \in I, \forall j \in J, r = 0, \ldots, P-1 \tag{8}$$

The objective function in Eq. 1 minimizes the expected failure cost. In this formulation, each demand node i is assigned to its level-r facility j when the r closer facilities have failed (an event with probability q^r). Facility j must remain functional to serve node i. The probability of j being functional is $1 - q$ if j is a facility subject to failure ($j \in F$) and 1 if it is not subject to failure ($j \in NF$). By definition of level-r assignment, all r facilities closer than j are subject to failure (Snyder and Daskin, 2005). This layered assignment structure ensures that service continuity is maintained even under multiple simultaneous facility failures, thereby enhancing the reliability of the overall system.

Eq. 2 ensures that for each demand node i and each level r, the node is either assigned to a level-r facility or it is assigned to a level-s NF facility where $s < r$. Eq. 3 ensures that demand is assigned to a facility at any level only if a facility is opened at that location. Eq. 4 is the budget constraint which requires P facilities to be opened. Eq. 5 makes sure that a demand node is assigned to a certain facility at most at one level.

There is a cost θ associated with each demand node, representing the cost incurred when the demand is served by a facility outside the region under study. Although the objective is to identify the best set of CSLs within M6 to minimize EMV travel times for incidents occurring inside M6, there are cases where demand must be served by units outside the region. This may occur either when all CSLs within M6 are unavailable or when a CSL outside M6 is closer to the incident location, making it more cost-effective to assign the demand to an external CSL. To account for these cases, an emergency facility $u (u \in NF)$ is defined, which is not subject to failure, and is forced to be open through the constraint in Eq. 6. This emergency facility is intended as a reduced-form representation of all resources located outside the study area. Given this abstraction, θ is modeled as a constant travel-time penalty; i.e., we assume a constant cost θ from this facility to all demand nodes ($d_{ui} = \theta, \forall i \in I$). Eqs. 7 and 8 are the binary constraints for the decision variables.

For the current case study, I set consists of 510 nodes, J includes 27 candidate locations, and P (facility budget) is set to 5, consisting of 4 internal facilities subject to failure and 1 external emergency facility not subject to failure. EMV travel time is used as edge cost to prepare d_{ji} (shortest distances between candidate CSLs and network nodes determined a priori). The problem is solved using GLPK solver (GNU Linear Programming Kit) (Makhorin, 2025) in Pyomo (Bynum et al., 2021; Hart et al., 2017). Two different scenarios are defined for the analysis:

- **Scenario A**: Best CSLs and demand assignments are found using the reliability-based p-median approach.

- **Base scenario**: The current CSLs employed by FDNY are set as the selected CSLs. Demand is assigned to the CSLs using the reliability-based p-median model.

4.2.2 Results

The probability of failure at a facility (q) and the cost of traveling from emergency facility u to each demand node (θ) are two parameters that need to be tuned. Analyzing the AVL data shared by FDNY showed that among life-threatening incidents within M6 (either initial or final severity level < 4) during 8–10 AM, about 41% were responded by M6 units. The expected demand assigned to each selected CSL subject to failure, and to the emergency CSL not subject to failure are found using Eqs. 9 and 10, respectively. The expected demand assignment rate for each selected facility is calculated by dividing the expected demand assigned to that facility by the total expected demand (i.e., the sum of the expected demand assigned to all selected CSLs).

$$\sum_{i \in I} h_i \left[\sum_{r=0}^{P-1} q^r (1-q) Y_{jir} \right] \tag{9}$$

$$\sum_{i \in I} h_i \left[\sum_{r=0}^{P-1} q^r Y_{jir} \right] \tag{10}$$

Due to the additional level-r dimension in the assignment variables, the problem takes significantly longer to solve compared to an original P-median problem because of the binary constraints on the decision variables. In the original P-median problem, it is sufficient to define the binary constraint only for the facility location variables (X_j), while allowing the assignment variables to be continuous and non-negative rather than binary. Due to the presence of the constraint defined in Eq. 2, this relaxation is no longer sufficient in the reliability-based p-median approach.

To improve computational efficiency, the binary constraint can instead be applied only to the NF assignment variables, while allowing the F assignment variables to remain continuous and non-negative. Thus, Eq. 8 can be replaced by Eqs. 11 and 12, leading to the same solution while reducing the computation time. Additionally, a maximum runtime of 1 hour or an optimality gap threshold of 1% was set to prevent excessively long runtimes.

$$Y_{uir} \in \{0, 1\}, \forall i \in I, r = 0, \dots, P-1 \tag{11}$$

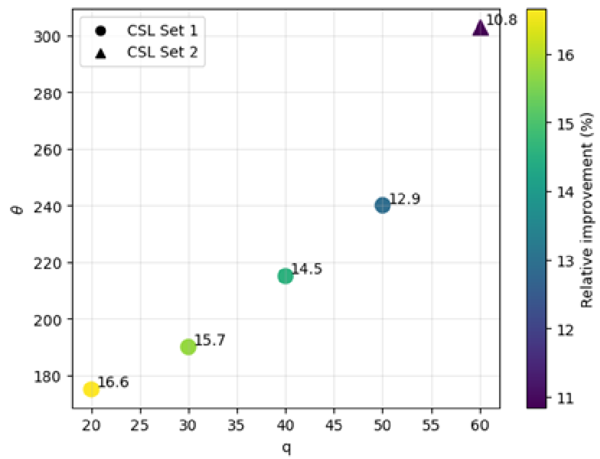


Figure 15: Calibration of q and θ

$$Y_{jir} \geq 0, \forall i \in I, \forall j \in J \setminus \{u\}, r = 0, \ldots, P-1 \tag{12}$$

The values of θ and q were selected as scenario parameters such that the base scenario reproduces the observed 41% demand assignment to M6 units reported in the FDNY data. The base scenario was used as the reference scenario because the CSLs are compatible with the observed FDNY data. Multiple combinations of q and θ can reproduce the observed 41% share of responses assigned to M6 units. This trade-off in the (q, θ) space is shown in Figure 15, where an increase in the failure probability q is accompanied by a higher calibrated value of θ. Intuitively, as failures become more likely, the system implicitly tolerates a higher external service cost, since relying on external resources becomes more probable in expectation.

We also evaluate the optimal CSL selection (Scenario A) for each calibrated (q, θ) pair and compute the relative improvement in the objective function compared to the base scenario. The results show a clear downward trend: as the failure probability increases from 20% to 60%, the relative improvement decreases from 16.6% to 10.8%. This suggests that higher uncertainty (larger q) reduces the marginal benefit of optimization, likely because increased failures diminish the effectiveness of strategically placed CSLs.

Additionally, the structure of the optimal solution remains stable over a substantial portion of the parameter space. Specifically, the selected CSL set remains unchanged for q values between 20% and 50% (CSL Set 1 in Figure 15), despite the corresponding changes in θ. This indicates a degree of robustness in the solution with respect to moderate parameter variation. However, when q reaches 60%, the optimal CSL configuration changes (CSL Set 2 in Figure 15), signaling a threshold beyond which the system behavior, and consequently

Table 6: Comparison of performance metrics in different scenarios

Scenario	Performance metric (seconds)
Base Scenario	98,035
Scenario A	83,789

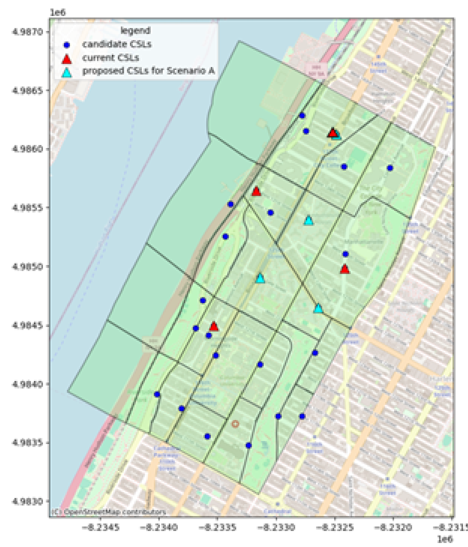


Figure 16: Proposed CSLs using reliability-based p-median approach.

the optimal design, shifts more substantially.

The remaining analysis in this section focuses on one representative scenario parameter setting, where $q = 40\%$ and $\theta = 215$, selected from the feasible (q, θ) combinations that reproduce the observed 41% share of responses assigned to M6 unit. To reiterate, the 40% failure probability means that when an incident occurs, the units based out of the CSL closest to it are 40% likely to be unavailable due to being busy or being located farther away from another unit that might be closer. In this way, the selected parameters provide a simplified representation of the underlying operational dynamics. This selected parameter setting used in the optimized Scenario A led to a 55% expected demand assignment to M6 units.

After solving the problem for each scenario, the value of the objective function is reported as the performance metric in Table 6, which represents the expected travel time for EMVs. The solution for base scenario is optimal, and the optimality gap of the obtained solution for Scenario A is 0.2%. Under the selected (q, θ) parameter setting, the improvement in the performance metric from the base scenario to Scenario A is 14.5%. Notably, this improvement is achieved alongside an increase in the expected demand served by M6 units, rising from 41% to 55%. Figure 16 illustrates the selected CSLs.

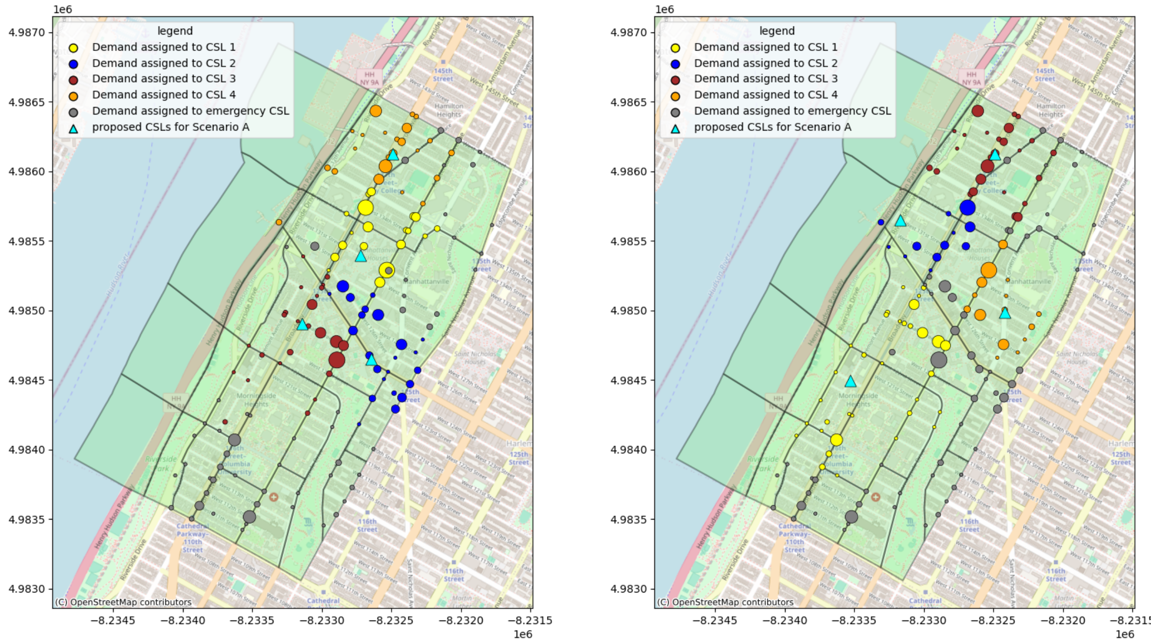
Level-0 distribution of demand across different CSLs under each scenario is shown in Figure

Table 7: Distribution of demand across selected CSLs at different levels

Base scenario					
CSL ID	Level-0	Level-1	Level-2	Level-3	Level-4
10	97	0	0	0	0
20	62	20	0	0	0
22	108	35	0	0	0
24	87	0	0	0	0
28 (emergency CSL)	187	299	55	0	0
Scenario A					
CSL ID	Level-0	Level-1	Level-2	Level-3	Level-4
8	126	34	0	0	0
14	102	93	0	0	0
19	98	47	1	0	0
22	90	32	0	0	0
28 (emergency CSL)	125	210	205	1	0

17, and demand assignments at different levels are shown in Table 7. The size of the demand nodes in Figure 17 is proportional to the amount of demand at each node, and nodes with the same color are assigned to the same CSL. In Scenario A, the CSLs are clustered in the upper and central regions, where most of the demand occurs, while demand nodes in the lower region are largely assigned to the emergency CSL. This emphasizes the importance of ensuring that an external CSL is available to respond to demand in this area within the assumed travel time. The primary demand assignment in Scenario A is distributed fairly evenly across the selected CSLs, with assignment ratios ranging from 17% to 23%. It is also worth noting that, as shown in Table 7, once a demand node is assigned to the NF emergency CSL, it is not assigned to any other CSLs at subsequent levels.

The CSL optimization presented here is intended as a proof-of-concept demonstration of how the TDT and EMVAID framework can support this class of planning problem, rather than as a direct operational recommendation for FDNY. Several underlying assumptions and limitations are associated with this model, as discussed below. It is assumed that all EMVs begin their trips from CSLs to respond to incidents, whereas in reality, they may be at the hospital or en route to their CSL when assigned to a new incident. As noted by FDNY partners, ambulances are rarely stationary at CSLs in practice and are instead dispatched dynamically based on availability and proximity. Some other simplifying assumptions include aggregating demand at network nodes and selecting candidate CSLs based on atom centroids, without considering practical operational factors. As a result, the approximately 14.5% reduction in expected EMV travel time observed in scenarios A compared to the base scenario may represent an optimistic estimate.



(a) Scenario A

(b) Base scenario

Figure 17: Primary demand assignment to each selected CSL.

4.2.3 Robustness of CSL Optimization Results to EMVAID Prediction Uncertainty

The optimization results presented in the previous subsection were based on travel times derived directly from the SUMO simulation outputs. To evaluate the potential impact of AI prediction uncertainty on optimization outcomes, an additional robustness analysis was conducted using EMVAID-predicted link speeds. This analysis propagated EBM-predicted link speeds through the routing and facility location framework to assess the sensitivity of the resulting CSL recommendations to prediction errors. Therefore, the robustness analysis focuses on uncertainty in the AI-predicted link speeds; uncertainty associated with the average intersection-delay assumptions was not modeled.

Out-of-sample residuals obtained from the grouped cross-validation procedure were treated as an empirical representation of EBM prediction uncertainty. Residuals from all held-out simulation seeds were pooled to create a prediction error distribution. Fifty perturbed network realizations were generated from the five SUMO simulation seeds by randomly sampling from the residual distribution and adding the sampled errors to the corresponding EMVAID-predicted link speeds on individual network links. The resulting speeds were constrained to remain within physically feasible bounds using a minimum speed of 1 mph and a maximum speed of 30 mph before being converted to link travel times.

For each perturbed realization, shortest-path travel times were computed and used as inputs to the reliability-based p-median model. The optimization was then solved and the resulting CSL selections and improvement in objective values from the base scenario to Scenario A were compared across all perturbed realizations.

Across the 50 perturbed network realizations, Scenario A achieved an objective value that was 14.3% $\pm$ 1.5% lower than that of the base scenario. The improvement ranged from 9.6% to 17.3%, indicating that the relative performance of the two configurations remained consistent under prediction uncertainty. The analysis also shows that the CSL recommendations remained stable despite the introduced prediction uncertainty. Two of the four CSL locations subject to failure were selected in 100% of realizations, while the remaining locations were selected in 96% of realizations. These findings suggest that the optimization recommendations are relatively insensitive to the level of prediction uncertainty observed in the EMVAID model and support the potential use of AI-predicted travel times in future routing and facility location applications when simulation outputs are unavailable.

5 Conclusion

The TDT framework integrates a high-resolution microscopic traffic simulation model with real-world data and an AI-based predictive component. An AI model, EMVAID, based on a Traffic Simulation-Informed Explainable Boosting Machine (TSI-EBM), was designed to interpret key factors influencing EMV speeds. It also supported an applied decision support use case aimed at testing potential urban mobility interventions and impact analysis to changes in the built environment. The use case demonstration of the TDT in this study highlights its potential to be used as a tool to assist decision making and intervention evaluation without risking real-world disruptions. Input from FDNY experts played an important role in establishing face validity of the TDT outputs, offering practical insights grounded in operational experience that would be difficult to obtain otherwise.

The study demonstrates that proxy traffic features, such as 5-minute average vehicle speeds, can effectively substitute for directly observing downstream passenger vehicle features during EMV dispatches within the proposed modeling framework. This supports the potential value of segment-level real-time data and inference of related traffic features (e.g., occupancies) for traffic digital twin operations. The AI-driven model, EMVAID, further provided interpretable insights into associations between EMV performance and traffic and built environment factors, such as background traffic speed, bike lane presence, and roadway geometry. In particular, the EBM identified a transition region around 17 mph in the learned relationship between background traffic speed and EMV speed contribution, reflecting differing model behavior under congested and higher-flow traffic conditions. Use case analysis demonstrated the potential of the TDT for real-world applications. In the cross street location optimization, the proposed reliability-based p-median model estimated a 14.5% reduction in expected EMV travel times compared to current CSL configurations.

CSLs are an important starting point, but it should be noted that many ambulances are dispatched while away from these locations, and voluntary EMV responses—which do not follow CSL protocols—constitute a sizable share. Further research is needed to understand and model these operational dynamics more comprehensively within a dynamic simulation framework. Future research can also benefit from selecting candidate CSLs based on practical operational considerations.

Overall, this work establishes a replicable, scalable, and adaptable framework for integrating advanced simulation and AI tools into the strategic planning and day-to-day operations of emergency response systems. Looking forward, the TDT framework offers a promising foundation for real-time deployment by further integration of live feeds, optimization tools, and predictive analytics and extending simulation coverage citywide. Building on the achievements of the initial phase, the next stage of work will focus on extending the AI-based model to a citywide scale. Since critical simulation outputs such as EMV link speeds, background traffic conditions, and queue lengths are currently limited to the M6 dispatch area, future efforts will involve deriving these variables from alternative data sources. In parallel, the feature set used in EMVAID will be adapted to rely more heavily on citywide-available data. While features like the number of lanes, bike lanes, and bus stops are already accessible at scale, simulation-dependent variables such as background traffic speed and queue lengths will need to be replaced or approximated.

A limitation of this study is the absence of direct link- or corridor-level validation using observed EMV speed data. Due to the limited availability of high-resolution real-world EMV trajectory and speed observations, the proposed AI model was trained and evaluated primarily using SUMO-derived link-level speeds within a calibrated simulation environment. While trip-level validation against observed AVL data demonstrated that the SUMO simulation can reasonably reproduce aggregate travel behavior and overall response times, such validation does not fully confirm the accuracy of EMVAID predicted speeds at the individual link level. To partially address this limitation, a robustness analysis was conducted by propagating out-of-sample EMVAID prediction errors through the routing and CSL optimization framework, which showed stable facility location recommendations under the observed level of prediction uncertainty. However, uncertainty associated with the average intersection-delay assumptions was not explicitly modeled. Future work should incorporate observed link-level EMV trajectories and intersection-specific operational data to further validate and refine the proposed framework.

Additionally, this study does not include a sensitivity analysis of the SUMO 5-minute aggregation window. While this resolution aligns with the temporal granularity of operational traffic data and deployment constraints, the impact of finer temporal resolutions on EMV prediction accuracy and delay dynamics remains an open question for future work.

In summary, the initial phase of the project established the technical feasibility and local accuracy of an AI-enhanced traffic digital twin for emergency response. Future efforts will expand the framework's geographic scope and analytical capabilities, moving toward a city-wide, data-driven decision support tool for emergency medical services planning and urban policy evaluation.

Acknowledgements

The research was funded by the USDOT through the C2SMARTER University Transportation Center. The authors would like to thank our project collaborator and agency partner, the New York City Fire Department (FDNY), for their invaluable support and insights throughout the course of this research. We are also grateful to our industry partner, Airsage, for providing the connected vehicle wayfinding data that made this work possible. Additionally, we thank the New York City Department of Transportation (NYC DOT) for providing signal timing and for hire vehicle data to support the development of the Traffic Digital Twin. The collaboration of these partners has been essential in advancing our understanding of mobility solutions for emerging medical services in complex urban environments. We would also like to thank Yu Tang, Dachuan Zuo, Shuo Zhang, Tu Lan, Enola Ma, Zerun Liu, Zhenglei Ji, Haggai Davis, Hai Yang, Miguel Valle, Xiyuan Ren, Bingqing Liu, and Yanchao Li for their valuable assistance with field data collection. We are also grateful for Nurgul Zhumasheva and Shimao Ma's contribution to data cleaning, analysis and model development, whose work was supported by the Undergraduate Summer Research Program at New York University.

References

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

Berdowski, J., Berg, R. A., Tijssen, J. G., and Koster, R. W. (2010). Global incidences of out-of-hospital cardiac arrest and survival rates: systematic review of 67 prospective studies. *Resuscitation*, 81(11):1479–1487.

Bynum, M. L., Hackebeil, G. A., Hart, W. E., Laird, C. D., Nicholson, B. L., Siirola, J. D., Watson, J.-P., Woodruff, D. L., et al. (2021). *Pyomo-optimization modeling in python*, volume 67. Springer.

Chow, J. Y. J. (2016). Dynamic uav-based traffic monitoring under uncertainty as a stochastic arc-inventory routing policy. *International Journal of transportation science and technology*, 5(3):167–185.

Csiszár, I. (1975). I-divergence geometry of probability distributions and minimization problems. *The annals of probability*, pages 146–158.

Daskin, M. S. (1983). A maximum expected covering location model: formulation, properties and heuristic solution. *Transportation science*, 17(1):48–70.

de Larrea, E. L., Lam, H., Sanabria, E., Sethuraman, J., Mohammadi, S., Olivier, A., Smyth, A. W., Dolan, E. M., Johnson, N. E., Kepler, T. R., et al. (2021). Simulating new york city hospital load balancing during covid-19. In *2021 winter simulation conference (WSC)*, pages 1–12. IEEE.

Frazier, P. I. (2018). A tutorial on bayesian optimization. *arXiv preprint arXiv:1807.02811*.

Guo, F., Zhang, D., Dong, Y., and Guo, Z. (2019). Urban link travel speed dataset from a megacity road network. *Scientific data*, 6(1):61.

Hart, W. E., Laird, C. D., Watson, J.-P., Woodruff, D. L., Hackebeil, G. A., Nicholson, B. L., Siirola, J. D., et al. (2017). *Pyomo-optimization modeling in python*, volume 67. Springer.

Jenelius, E. and Koutsopoulos, H. N. (2013). Travel time estimation for urban road networks using low frequency probe vehicle data. *Transportation Research Part B: Methodological*, 53:64–81.

Jones, D. R., Schonlau, M., and Welch, W. J. (1998). Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, 13(4):455–492.

Khattak, A., Chan, P.-w., Chen, F., and Peng, H. (2023). Assessing wind field characteristics along the airport runway glide slope: An explainable boosting machine-assisted wind tunnel study. *Scientific Reports*, 13(1):10939.

Lopez, P. A., Behrisch, M., Bieker-Walz, L., Erdmann, J., Flötteröd, Y.-P., Hilbrich, R., Lücken, L., Rummel, J., Wagner, P., and Wießner, E. (2018). Microscopic traffic simulation using sumo. In *2018 21st international conference on intelligent transportation systems (ITSC)*, pages 2575–2582. Ieee.

Lou, Y., Caruana, R., Gehrke, J., and Hooker, G. (2013). Accurate intelligible models with pairwise interactions. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 623–631.

Makhorin, A. (2025). *GNU Linear Programming Kit*. Version 5.0, `https://www.gnu.org/software/glpk/`.

Marianov, V. and ReVelle, C. (1996). The queueing maximal availability location problem: a model for the siting of emergency vehicles. *European Journal of Operational Research*, 93(1):110–120.

Marianov, V. and Serra, D. (2002). Location–allocation of multiple-server service centers with constrained queues or waiting times. *Annals of Operations Research*, 111(1):35–50.

Meng, C., Yi, X., Su, L., Gao, J., and Zheng, Y. (2017). City-wide traffic volume inference with loop detector data and taxi trajectories. In *Proceedings of the 25th ACM SIGSPATIAL international conference on advances in geographic information systems*, pages 1–10.

Nori, H., Jenkins, S., Koch, P., and Caruana, R. (2019). Interpretml: A unified framework for machine learning interpretability. *arXiv preprint arXiv:1909.09223*.

NYC (2023). *End-to-End Response Time*. `https://www.nyc.gov/site/911reporting/reports/end-to-end-response-time.page`.

Owen, S. H. and Daskin, M. S. (1998). Strategic facility location: A review. *European journal of operational research*, 111(3):423–447.

Schrijver, A. (2012). On the history of the shortest path problem. *Documenta Mathematica*, 17(1):155–167.

Sha, D., Gao, J., Yang, D., Zuo, F., and Ozbay, K. (2023). Calibrating stochastic traffic simulation models for safety and operational measures based on vehicle conflict distributions obtained from aerial and traffic camera videos. *Accident Analysis & Prevention*, 179:106878.

Snyder, L. V. and Daskin, M. S. (2005). Reliability models for facility location: the expected failure cost case. *Transportation science*, 39(3):400–416.

Spall, J. C. (1988). A stochastic approximation algorithm for large-dimensional systems in the kiefer-wolfowitz setting. In *Proceedings of the 27th IEEE Conference on Decision and Control*, pages 1544–1548. IEEE.

Spall, J. C. (1999). Stochastic optimization and the simultaneous perturbation method. In *Proceedings of the 31st conference on Winter simulation: Simulation—a bridge to the future-Volume 1*, pages 101–109.

Su, H., Shi, K., Chow, J. Y. J., Jin, L., et al. (2020). Dynamic queue-jump lane for emergency vehicles under partially connected settings: A multi-agent deep reinforcement learning approach. *arXiv preprint arXiv:2003.01025*.

Su, H., Zhong, Y. D., Chow, J. Y. J., Dey, B., and Jin, L. (2023). Emvlight: A multi-agent reinforcement learning framework for an emergency vehicle decentralized routing and traffic signal control system. *Transportation Research Part C: Emerging Technologies*, 146:103955.

Yang, D., Ozbay, K., Xie, K., Yang, H., Zuo, F., and Sha, D. (2021). Proactive safety monitoring: A functional approach to detect safety-related anomalies using unmanned aerial vehicle video data. *Transportation research part C: emerging technologies*, 127:103130.

Zhan, X., Hasan, S., Ukkusuri, S. V., and Kamga, C. (2013). Urban link travel time estimation using large-scale taxi data with partial information. *Transportation Research Part C: Emerging Technologies*, 33:37–49.

Zuo, F., Gao, J., Sha, D., Chow, J. Y. J., and Ozbay, K. (2026). A hybrid evolutionary algorithm and reinforcement learning (ea-rl) approach for calibrating emergency medical vehicle behavior in microscopic simulation. In *The 105th Annual Meeting of Transportation Research Board*.