

A Dataset for the Medical Support Vehicle Location–Allocation Problem

Miguel Medina-Perez ¹, Giovanni Guzmán ^{1,*}, Magdalena Saldana-Perez ¹, Adriana Lara ²
and Miguel Torres-Ruiz ¹

¹ Centro de Investigación en Computación (CIC), Instituto Politécnico Nacional (IPN), UPALM-Zacatenco, Mexico City 07320, Mexico; jmedinap2022@cic.ipn.mx (M.M.-P.); amsaldanap@ipn.mx (M.S.-P.); mtorresru@ipn.mx (M.T.-R.)

² Escuela Superior de Física y Matemáticas (ESFM), Instituto Politécnico Nacional (IPN), UPALM-Zacatenco, Mexico City 07320, Mexico; alaral@ipn.mx

* Correspondence: jguzmanl@ipn.mx; Tel.: +52-(55)-5729-6000 (ext. 56617)

Abstract

In mass-casualty incidents, emergency responders require access to accurate and timely information to support informed decision-making and ensure the efficient allocation of resources. This article presents a dataset derived from a case study conducted in Mexico City (CDMX) based on the earthquake of 19 September 2017. The dataset presents hypothetical scenarios involving multiple demand points and large numbers of victims, making it suitable for analysis using optimization techniques. It integrates voluntary collaborative geographic information, open government data sources, and historical records, and details the data collection, cleaning, and preprocessing stages. The accompanying Python 3 source code enables users to update the original data for consistent analysis and processing. Researchers can adapt this dataset to other cities with similar risk characteristics, such as Santiago (Chile), Los Angeles (USA), or Tokyo (Japan), and extend it to other types of catastrophic events, including floods, landslides, or epidemics, to support emergency response and resource allocation planning.



Academic Editor: Johannes Schobel

Received: 7 November 2025

Revised: 29 November 2025

Accepted: 5 December 2025

Published: 10 December 2025

Citation: Medina-Perez, M.; Guzmán, G.; Saldana-Perez, M.; Lara, A.; Torres-Ruiz, M. A Dataset for the Medical Support Vehicle Location–Allocation Problem. *Data* **2025**, *10*, 206. <https://doi.org/10.3390/data10120206>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Dataset: <https://doi.org/10.5281/zenodo.17845383> (accessed on 4 December 2025).

Dataset License: All publicly shared data and derived layers are provided under CC BY-NC 4.0. All accompanying scripts are licensed under GPL-3.0. Traffic information obtained from the TomTom Traffic API is not redistributed in this dataset. Users must retrieve these data themselves using their own API key, in accordance with TomTom’s licensing terms. Only the source code required to reproduce the traffic extraction process is provided.

Keywords: multi-objective optimization; mass casualty incidents; location–allocation problem; *p*-median problem; maximum coverage problem

1. Introduction

Anticipation and effective planning for an immediate response to mass-casualty situations are critical for improving survival outcomes. Such incidents represent significant challenges that complicate emergency operations, including adverse weather conditions, limited accessibility to affected areas, and the geographical dispersion of victims. In addition, several local socioeconomic factors, such as limited public awareness of prevention

strategies, insufficient resources for disaster response, and poor coordination between public and private security entities, can further aggravate these situations.

During large-scale emergencies, multiple focal points often require coordinated efforts to strategically allocate human and material resources across different geographical locations. Consequently, precise management tailored to each area's specific needs becomes essential. One of the most widely adopted risk-management frameworks worldwide that addresses these aspects is the Incident Command System (ICS) [1]. The ICS was designed to facilitate inter-agency coordination across jurisdictions, enabling an effective unified response. It establishes an organizational structure to manage incidents of varying scales and complexities.

The ICS outlines best practices to ensure that trained personnel can efficiently and safely address five key medical components. However, several factors can further complicate real-world scenarios, including the dynamic nature of disasters, uncertainty regarding the extent of the damage, and the spatial distribution of victims. A common situation in large-scale mass-casualty incidents (MCIs) is the occurrence of multiple demand points, where victims are dispersed across a broad area rather than concentrated in a single location. In such situations, the personnel in charge of the emergency must be strategically distributed. Examples of these situations include widespread forest fires, floods, earthquakes, and pandemics.

To address this challenge, a methodology for modeling disaster-response scenarios was proposed in previous studies [2,3]. Nowadays, diverse studies have addressed smart city applications for emergency and disaster management, with particular focus on routing optimization and infrastructure-aware systems. Some examples include route optimization for emergency vehicles [4–6], evacuation planning and guidance [4,7], resource allocation and dispatch [4,5] and infrastructure damage assessment [4,7] among others.

Despite the growing body of research in this area, current disaster-related datasets, such as global earthquake catalogs, building exposure databases, and damage assessments issued by emergency agencies, often lack the comprehensive, infrastructure-level, and traffic-aware information needed for effective operational location–allocation models. They usually do not include the following essential data:

- Coordinated multi-layer geospatial structure;
- Synthetic multi-demand scenarios derived from historical patterns;
- A harmonized transportation network in a projected CRS appropriate for routing;
- Congestion-aware travel-time weights;
- A reduced set of realistic candidate bases derived from road-network topology.

Although Mexican open-data portals (PDACDMX, ANR, INEGI) and global exposure databases (GED4ALL, EM-DAT, Desinventar) provide high-quality geospatial information, these resources are fragmented and cannot be directly combined to construct operational disaster-response instances. In particular, they do not integrate cadastral records, historical impact data, transportation network topology, and traffic-aware travel time information into a single, harmonized structure. The dataset presented in this paper fills this gap by providing a unified and fully reproducible pipeline for generating consistent mass-casualty disaster scenarios.

This approach employs multi-objective optimization, modeling the case study as a location–allocation problem to optimize the placement and distribution of resources in high-casualty incidents. The proposed case study focuses on a post-earthquake scenario in Mexico City, using geospatial data obtained from voluntary collaborative mapping, open-government data sources, and historical information from the earthquake of 19 September 2017. Indeed, the resources to be optimally located and allocated consist of a set of ambulances,¹ emphasizing medically relevant aspects that influence victim survival.

On the other hand, to solve the optimization problem, a metaheuristic approach was employed, coupled with a parameter-tuning technique to identify configurations with robust performance across multiple problem instances and hypothetical scenarios. These configurations serve as a reference for future analyses. This article presents an updated dataset (as of October 2025) and the accompanying scripts that enable the generation of these scenarios, allowing future research to perform comparative analyses using alternative methods or optimization strategies.

2. Data Description

This section provides a comprehensive description of all datasets used to construct A Dataset for the Medical Support Vehicle Location–Allocation Problem. It presents (i) the raw geospatial layers obtained from government repositories, (ii) their semantic content and spatial coverage, (iii) their role within the dataset, and (iv) a summary of metadata and reproducibility considerations. The layers documented in this manuscript constitute the raw data of the project, before any processing, transformation, or integration into the analytical pipeline described in Section 3. All layers are included in the `raw_data` directory of the dataset distribution.

2.1. Overview of the Dataset

The dataset integrates official open government geospatial layers, census information, land-registry data, building impact records from the 2017 Mexico City earthquake, and the OpenStreetMap (OSM) transportation network. These datasets collectively describe the following:

- The administrative and political structure of Mexico City;
- Seismic risk and emergency management divisions;
- Demographic and land-use distribution;
- Building-level information for population estimation;
- Historical damage patterns used to calibrate synthetic disaster scenarios.

All layers are clipped to the official boundary of Mexico City and maintain their original geometry, licensing constraints, and metadata.

2.2. Raw Data Sources

Table 1 summarizes the official datasets used in the study, including the name used in [3], availability, format, spatial reference system and geometry. All datasets were obtained between September 2024 and October 2025 from

- The Mexico City Open Data Portal (in Spanish Portal de Datos Abiertos de la Ciudad de México—PDACDMX);
- The National Risk Atlas (in Spanish Atlas Nacional de Riesgos—ANR);
- The National Institute of Statistics and Geography (in Spanish Instituto Nacional de Estadística y Geografía—INEGI);
- The Secretariat of Urban Development and Housing (in Spanish Secretaría de Desarrollo Urbano y Vivienda de la Ciudad de México—SEDUVI);
- The Mexico City Reconstruction Commission (in Spanish Comisión para la Reconstrucción de la Ciudad de México—CRCDMX).

For layers no longer publicly available, notes are provided regarding origin and reproducibility limitations.

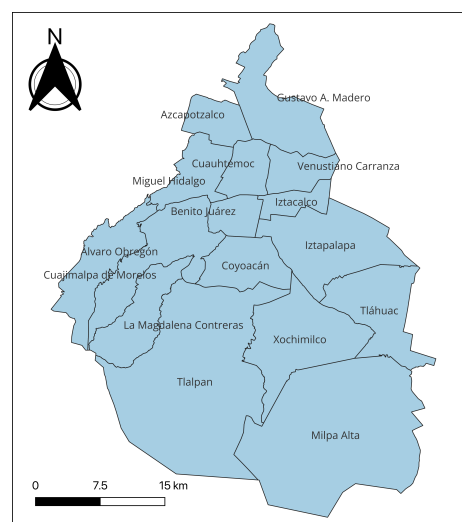
Table 1. Raw datasets obtained from official sources.

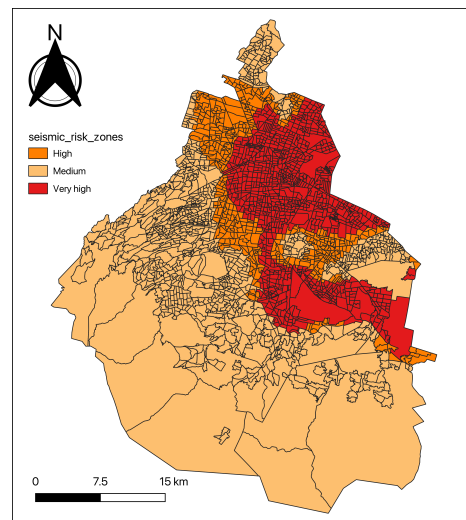
Layer Name	Alias	Online	Format	SRS	Geometry
territorial boundaries [8]	boundaries	yes	shapefile	EPSG:4326	polygon
mayorships [9]	—	yes	shapefile	EPSG:4326	polygon
seismic risk zones [10]	risk_zones	yes	shapefile	EPSG:4326	polygon
emergency zones [11]	regions	yes	PDF/Doc	—	polygon
city blocks [12]	blocks	yes	shapefile	EPSG:4326	polygon
land registry [13]	land_registry	yes	shapefile	EPSG:32614	polygon
hospitals [14]	—	yes	shapefile	EPSG:32614	point
gathering centers [15]	gathering_centers	yes	CSV	EPSG:4326	point
shelters [16]	refuges	yes	shapefile	EPSG:4326	point
buildings data [17]	seduvi	yes	shapefile	EPSG:32614	point
collapsed buildings	collapses	no	shapefile	EPSG:4326	point
damaged buildings	damages	no	shapefile	EPSG:4326	point

2.3. Administrative and Risk-Related Layers

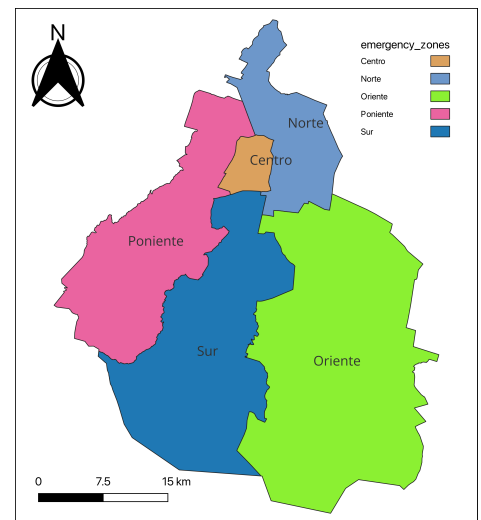
These datasets provide spatial context for political, emergency, and seismic risk zones in Mexico City, (Figure 1a–d displays these administrative and emergency layers):

- Territorial boundaries: Polygon defining the official extent of Mexico City. The polygon with attribute COV_ID = 9 is selected from the original dataset.
- Mayorships: Geographic boundaries of the sixteen districts that constitute the city.
- Seismic risk zones: Areas classified by AGEb, representing expected seismic amplification and hazard.²
- Emergency zones: Regions defined by the local emergency management agency to coordinate response operations.

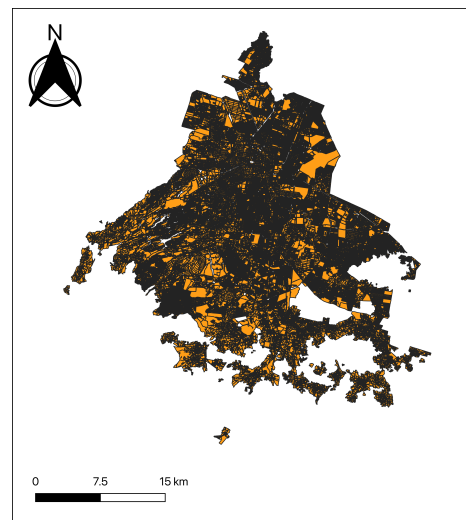
**(a)** Territorial boundaries**(b)** Mayorships**Figure 1.** Cont.



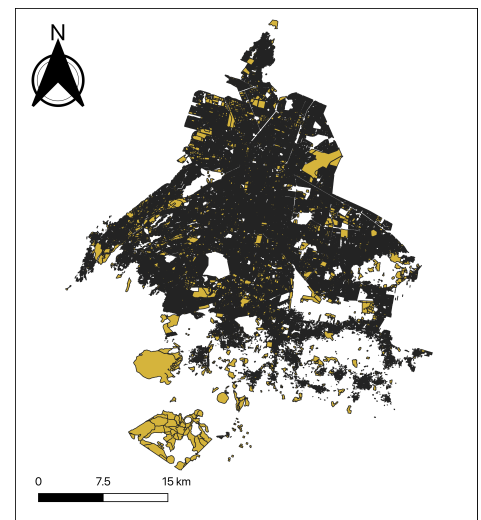
(c) Seismic risk zones



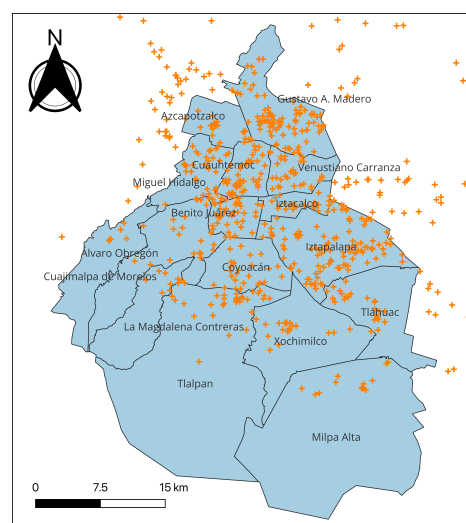
(d) Emergency zones



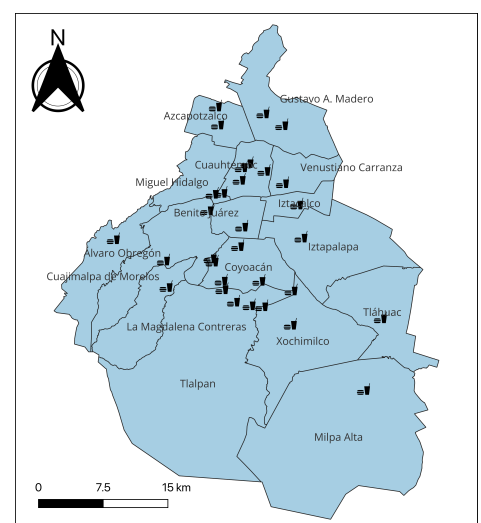
(e) City blocks



(f) Land registry



(g) Hospitals



(h) Gathering centers

Figure 1. Cont.

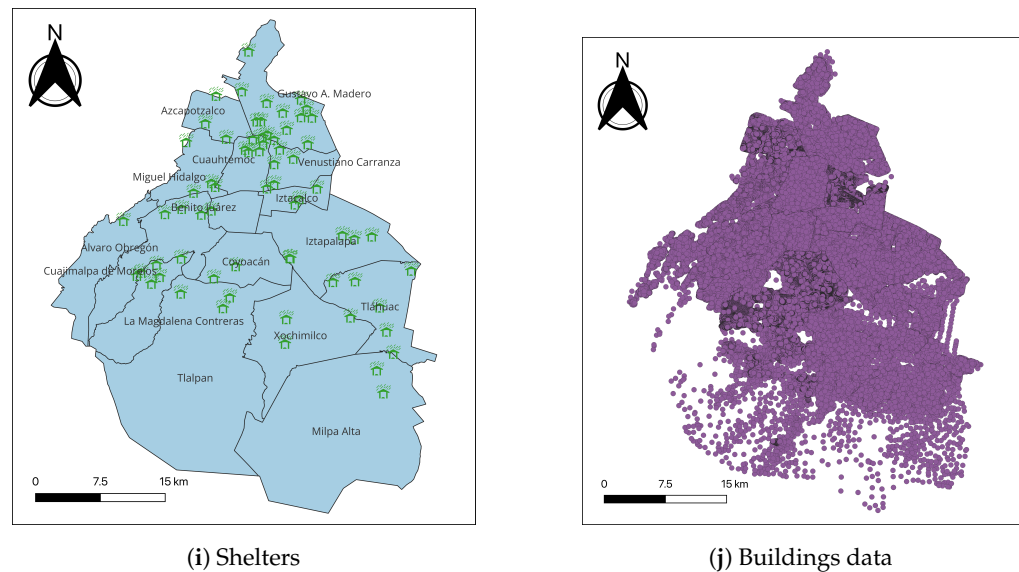


Figure 1. Raw data updated to October 2025.

2.4. Socioeconomic and Cadastral Layers

These layers describe population distribution, urban structure, and land-use information (Figure 1e–j shows these layers):

- City blocks: AGEB-level blocks including population and housing census data.
- Land registry: Cadastral parcels with use, lot size or parcel area, and identifiers.
- Buildings data: Centroids and attributes of registered buildings.
- Hospitals: Public and private hospitals operating within city boundaries.
- Gathering centers: Official collection and supply points for emergencies.
- Shelters: Temporary accommodations designated during disasters.

2.5. Earthquake Impact Layers

Two historical datasets provide the empirical basis for scenario calibration:

- Collapsed buildings: Locations where structures fully collapsed during the 2017 earthquake. This dataset is no longer available publicly; the version included corresponds to the official file released in late 2018.
- Damaged buildings: Buildings requiring structural intervention. The current public version differs from the original 2018 release, because this new dataset combines the following:
 - The 2017–2018 social and technical census;
 - Entries added between 2019 and 2023 after inspections and citizen reports.

An important methodological clarification concerns the damaged buildings dataset. The version in this repository corresponds to the original 2018 release, which contains only structures directly affected by the 19 September 2017, earthquake. More recent public versions available through PDACDMX include additional points documented between 2019 and 2023, many of which correspond to building damage unrelated to the 2017 seismic event. Preliminary spatial analyses conducted by the authors revealed that the extended dataset does not preserve the spatial concentration patterns characteristic of the 2017 earthquake impact and is, therefore, not suitable for calibrating disaster impact models or identifying heavily affected zones. For this reason, the 2018 version is retained to preserve the historical signal necessary for scenario calibration and spatial validation. On the other hand, Figure 2 illustrates both datasets.

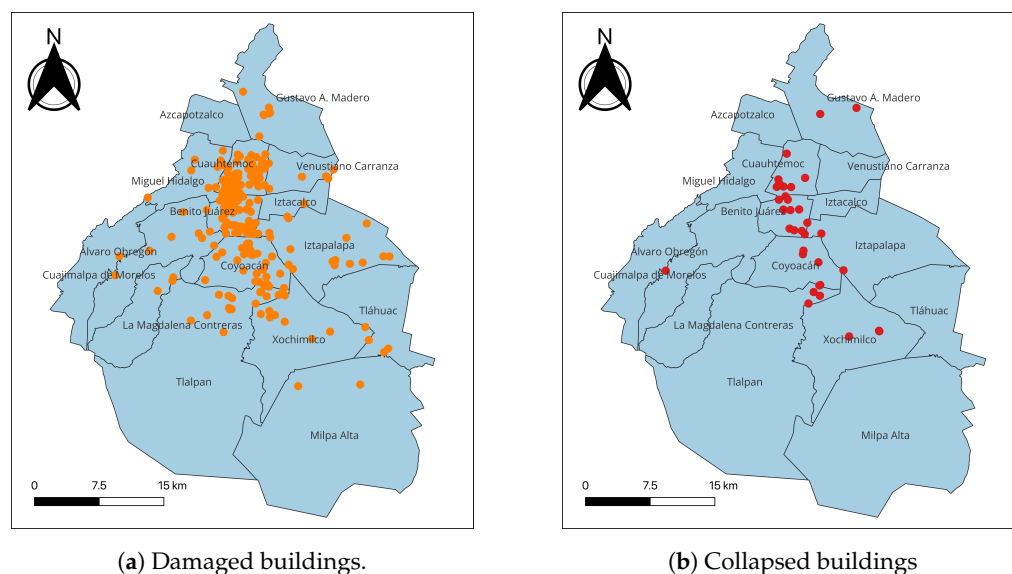


Figure 2. Information on buildings affected by the earthquake of 19 September 2017.

2.6. Data Availability, Provenance, and Limitations

Some datasets used in this work have been removed from public portals over the last two years, particularly the damaged buildings and collapsed buildings layers. Although the dataset included in this article preserves the most complete available records, users should be aware of the following reproducibility limitations:

- Collapsed buildings: original dataset from 2017 is no longer hosted by PDACDMX.
- Damaged buildings: the current public release differs from the 2017–2018 official census; the version included corresponds to the 2017–2018 census.
- Emergency zones: official shapefiles are not provided; polygons were digitized from the governmental document.

All other datasets listed in Table 1 remain publicly accessible as of October 2025.

2.7. ISO 19115 Metadata Summary

Each dataset is documented in accordance with a reduced version of the ISO 19115 standard [18]. Table 2 summarizes key metadata elements applicable to all raw layers.

Table 2. ISO 19115 metadata summary for the raw datasets.

Metadata Element	Description
Spatial reference system	EPSG:4326 or EPSG:32614 (as provided by the source)
Temporal reference	2017 (earthquake impact), 2019–2025 (other layers)
Geographic extent	Mexico City official boundary
Lineage	Official government sources, harmonized version included in dataset
Positional accuracy	As reported by PDACDMX, ANR, INEGI, SEDUVI, CRCDMX and OSM
Thematic accuracy	As defined by source agencies
Completeness	Full coverage of Mexico City; clipped versions provided in processed data
License	Original licensing from PDACDMX, ANR, INEGI, SEDUVI, CRCDMX and OSM

2.8. Role of Raw Data in the Dataset Pipeline

The raw layers described in this section conform to distinct roles in subsequent stages:

- Administrative layers define clipping and aggregation units;
- Cadastral and building layers support victim estimation;
- Seismic and emergency layers guide synthetic scenario calibration;
- Earthquake impact layers calibrate the spatial kernel density estimator used to generate hypothetical disaster scenarios.

Their transformation into coordinated, derived, and synthetic layers is described in Section 3.

3. Materials and Methods

As described in [3], the proposed methodology consists of eight stages, depicted in Figure 3, and outlined below:

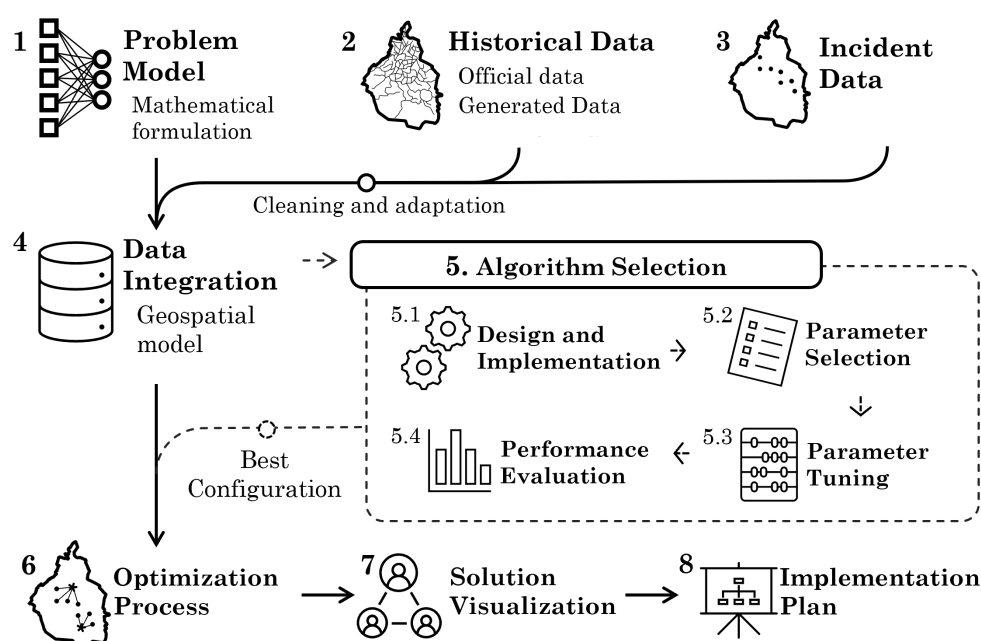


Figure 3. Proposed methodology. Adapted from [3].

1. Problem model: It is formulated mathematically as an assignment optimization problem.
2. Historical data. This stage involves acquiring or generating the information necessary to define the case study, as described in Section 2.
3. Incident data. At this stage, data specific to the incident are collected from the locations of multiple demand points (with confirmed or suspected victims), road blockages, and current hospital capacity.
4. Data integration. The outcome is a geospatial model that captures the full context of the incident. The technical details regarding geospatial harmonization and mobility graph updates thoroughly described in Sections 3.1 and 3.3.2, respectively.
5. Algorithm selection. Once the geospatial model and optimization problem have been defined, we selected an appropriate algorithm through the following sub-stages:

- 5.1. Design and implementation. An optimization algorithm is proposed and implemented to address the formulated problem.
- 5.2. Parameter selection. Each algorithm requires a specific set of parameters that must align with the geospatial model's information. Parameters may be categorical or numerical.
 - Categorical variables include internal procedures such as assignment methods, crossover or mutation operators, and performance metrics.
 - Numerical variables correspond to hyperparameters that can take discrete or continuous values, such as population size, crossover and mutation rates, and number of offspring. For numerical variables, it is essential to define a finite search space to ensure computational feasibility.
- 5.3. Parameter tuning. This step aims to determine the optimal algorithm configuration for solving a set of instances of the same problem.
- 5.4. Performance evaluation. The configurations obtained in the previous step are evaluated using performance criteria, including execution time, the best solution found, and average solution quality. Based on these criteria, the most suitable algorithm was selected for real-world implementation.
6. Optimization process. With the problem model and incident-specific data defined, the optimization algorithm that demonstrated the best performance is executed, producing a set of solutions for use in the subsequent stage.
7. Solution visualization. The resulting solutions are visualized to support decision-making by response teams, along with any additional information relevant to the emergency plan.
8. Implementation plan. The resulting solutions are visualized graphically to support decision-making by response teams, accompanied by any additional information relevant to the emergency plan.

The geospatial layers documented in Section 2 provide the historical and contextual information required for stages 2–4 of this methodology. The dataset published with this research work focuses on making these stages reproducible, by providing both the raw data (Section 2) and the full processing pipeline described in this section.

In particular, this section details the Jupyter Notebooks—executed in Google Colab using Python 3.12.12 on Ubuntu 22.04.4 LTS, within the managed Jupyter environment provided by Colab—developed for the study, clarifying how they implement the “Historical data”, “Incident data”, and “Data integration” stages (2–4), as well as the “Optimization process” and “Solution visualization” stages (6–7). Stage 5 (“Algorithm selection”) is not reproduced here, since the NSGA-II configuration used in this dataset corresponds to the algorithmic results already published in [3]. The notebooks operate on the raw layers stored in the `raw_data` directory and produce the derived, coordinated, and synthetic layers used to construct simulation scenarios involving multiple demand points and large numbers of victims (mass casualty incidents).

Figure 4 summarizes the end-to-end processing pipeline implemented in Python. The Block (A) corresponds to geospatial data harmonization and graph construction, the Block (B) to scenario and traffic generation, and finally, Block (C) to optimization and visualization. Each gray box in the figure corresponds to a Jupyter Notebook in the repository.

The following subsections describe each stage of this pipeline, from geospatial data harmonization to instance generation, optimization, and visualization.

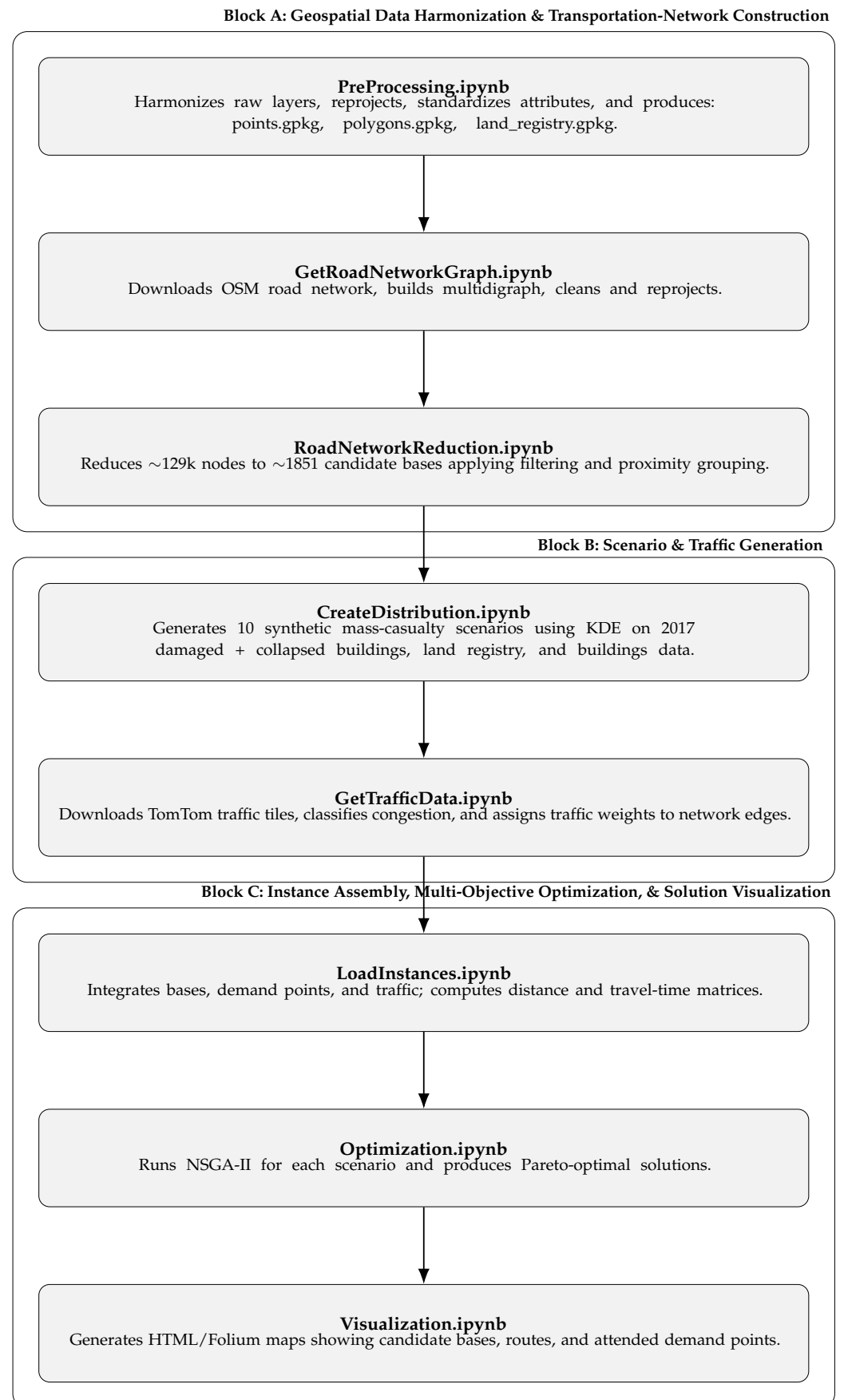


Figure 4. Processing pipeline implemented through Jupyter Notebooks for constructing the full set of coordinated, derived, and synthetic layers used to generate the proposed dataset.

3.1. Geospatial Data Harmonization and Transportation-Network Construction

Block (A) groups the first three Jupyter Notebooks of the pipeline, PreProcessing, GetRoadNetworkGraph, and RoadNetworkReduction. Together, these scripts transform the raw layers described in Section 2 into a harmonized geospatial database and a reduced transportation network suitable for scenario generation and optimization.

3.1.1. Geospatial Data Harmonization

The PreProcessing.ipynb notebook processes the contents of the raw_data folder. Each layer is standardized by harmonizing attribute structures and reprojecting all geometries to the projected coordinate system EPSG:6369 system, which is suitable for metric distance calculations in Mexico City. In addition, layer and attribute names are converted to lower-case to ensure a consistent naming convention. A spatial clipping operation is then applied using the territorial_boundaries layer as the reference extent. The resulting transformed layers are organized into three thematic GeoPackage files stored in the processed_data directory: polygons.gpkg, points.gpkg, and land_registry.gpkg.

3.1.2. Extraction of the Transportation Network

The harmonized territorial boundary is used as the spatial query extent for retrieving the road network from OpenStreetMap (OSM) through the OSMNX library. GetRoadNetworkGraph.ipynb notebook downloads the complete street network of Mexico City and constructs a NetworkX MultiDiGraph, where nodes represent intersections and edges correspond to directed road segments.

During this stage, several preprocessing operations are performed: (i) cleaning and standardization of OSM attributes, (ii) reprojection of all geometries from the native geographic CRS (EPSG:4326) to the projected CRS EPSG:6369, (iii) removal of topological inconsistencies, and (iv) reduction of non-essential attributes. Table 3 summarizes the edge attributes available in the unsimplified graph, including their definitions and completeness percentage.

Table 3. Attributes of the edges of the NetworkX graph provided by OSM.

Attribute	Specification	Type	Percentage Data
osmid	unique identifier	int	100
highway	road type	str	100
lanes	number of lanes	int	20.54
maxspeed	maximum speed	int	9.85
name	name	str	86.28
oneway	direction	bool	100
ref	alternative name	str	2.93
reversed	geometry direction	bool	100
length	length	float	100
geometry	geometry	geometry	100
merged_edges	merged edges	list	35.49
junction	intersection segment	str	0.32
width	road width	float	0.23
bridge	indicates whether it is a bridge	str	0.48
access	access type	str	5.75
tunnel	indicates whether it is a tunnel	bool	0.08
service	related service type	str	0

Because the raw OSM graph contains a large number of nodes and edges, the notebook applies attribute reduction to retain only fields relevant for routing and travel-time estimation. Attributes such as junction, width, bridge, access, tunnel, and service are removed, as they provide minimal analytical value for the emergency-mobility framework and contain very limited data coverage.³ The reversed attribute is also eliminated because the dataset explicitly stores edge directionality. Since incident-related mobility changes (e.g., road closures or congestion) mainly affect edge weights rather than topology, the road graph only needs to be downloaded once. If new infrastructure becomes available, the graph can be updated incrementally.

After simplification, the resulting transportation graph consists of 129,067 nodes and 304,771 directed edges. The graph is exported to the `graph_data` directory in both GeoPackage and GraphML formats. For subsequent notebooks, it is recommended to work with the nodes and edges tables separately: nodes include a unique identifier, whereas edges are indexed by their `osmid` attribute. Alternatively, edges may be referenced using triplets (u, v, k) , where u and v are node identifiers and k distinguishes multiple parallel edges.

3.1.3. Candidate-Base Reduction

The next step consists of identifying a manageable set of candidate locations for temporary ambulance deployment. If the full search space were considered, every road segment in the transportation network could serve as a potential base for the search. This would require evaluating 304,771 possible locations (one per edge), or 129,067 locations if only network intersections (nodes) were used. Both alternatives are computationally infeasible for generating large-scale scenarios and multi-objective optimization.

In contrast with common assumptions in the literature—where potential ambulance sites are known a priori and typically restricted to hospitals—Mexico City lacks an official dataset defining such facilities. In practice, emergency response teams frequently use existing roadway infrastructure (e.g., wide avenues, major intersections, or parking areas) as temporary staging areas.

To obtain a realistic and tractable set of candidate sites, the `RoadNetworkReduction.ipynb` notebook applies the four-step procedure introduced in [3]: (i) selection by characteristics, (ii) selection by capacity, (iii) elimination of redundancy, and (iv) proximity-based grouping. For this dataset, these steps are implemented in Python to reduce the original set of 129,067 network nodes to 1851 candidate base locations. Table 4 summarizes the effect of each reduction stage.

Table 4. Simplification of the transport graph.

Step	Result	Number of Nodes
Transport graph from Networkx	$\{V_t, E_t\}$	129,067
Selection by characteristics	$\{V_f, E_f\}$	15,640
Selection by capacity	$\{V_f, E_f\}$	8208
Elimination of redundancy	V_s	5743
Proximity-based grouping	V	1851

3.2. Scenario and Traffic Generation

Block (B) comprises the `CreateDistribution` and `GetTrafficData` Jupyter Notebooks. Together, these scripts generate the dynamic components of the dataset: (i) synthetic mass-casualty scenarios derived from historical earthquake effects, and (ii) traffic-aware edge weights obtained from TomTom's Traffic Flow API. The outputs of this block provide the scenario- and congestion-dependent inputs required for creating fully specified emergency-response instances.

3.2.1. Generation of Mass-Casualty Scenarios

The generation of hypothetical mass-casualty scenarios depends on spatial information derived from the 19 September 2017, Mexico City earthquake. The Mexico City Open Data Portal provides georeferenced locations of collapsed and damaged buildings for that event. Although these observations cannot be used to predict the precise locations of future damage, they offer a historical spatial pattern of observed impacts that supports the construction of reproducible synthetic instances for algorithmic testing.

A direct prediction of which buildings may collapse in a future earthquake is infeasible, given the number of contributing factors, among them soil type, construction date, building materials, regulatory history, and structural integrity. Moreover, approximately 70% of residences in Mexico City were built under unknown design conditions [19], making reliable forward modeling outside the scope of this dataset. For this reason, the dataset adopts a data-driven scenario approach rather than a predictive one.

To generate the geographic distribution of hypothetical demand points, the CreateDistribution.ipynb notebook computes a kernel density estimation (KDE) using the locations of buildings damaged in 2017. The KDE is implemented with the scikit-learn library using a Gaussian kernel. A one-dimensional grid search over 1000 bandwidth values (from 10^1 to 10^4 on a logarithmic scale) is performed. For each value, leave-one-out cross-validation is used to evaluate the cumulative log-likelihood of the model, and the bandwidth that maximizes this quantity is selected.

In addition to spatial location, each synthetic demand point must include an assessment of the number of victims. In real emergency operations, this estimate is initially reported by the first responder arriving at the scene [20]. Because the 2017 damaged-building dataset does not contain this information, the notebook implements the procedure described in [3] to identify viable candidate buildings and assign a victim estimation. This analysis integrates three information sources: KDE intensity values, land registry records, and SEDUVI building data.

The script produces ten hypothetical scenarios, stored in the point_instances.gpkg file within the processed_data directory. Each scenario consists of a set of synthetic demand points representing potential locations requiring emergency medical attention. Their spatial distributions are shown in Figures 5 and 6.

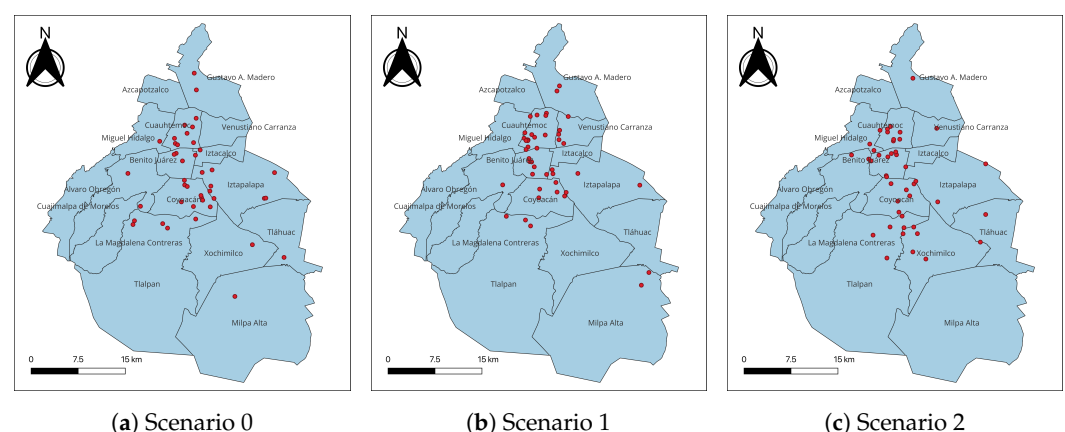


Figure 5. Cont.

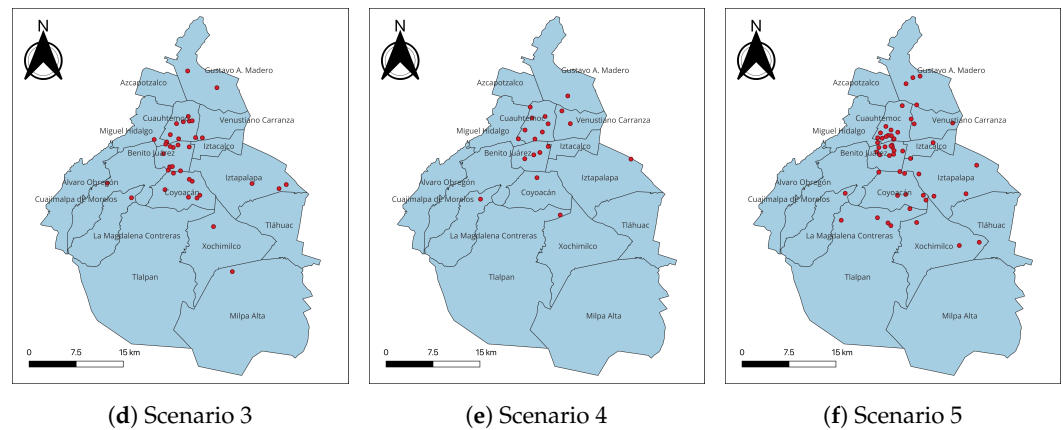


Figure 5. First six mass-casualty scenarios generated by the CreateDistribution.ipynb script.

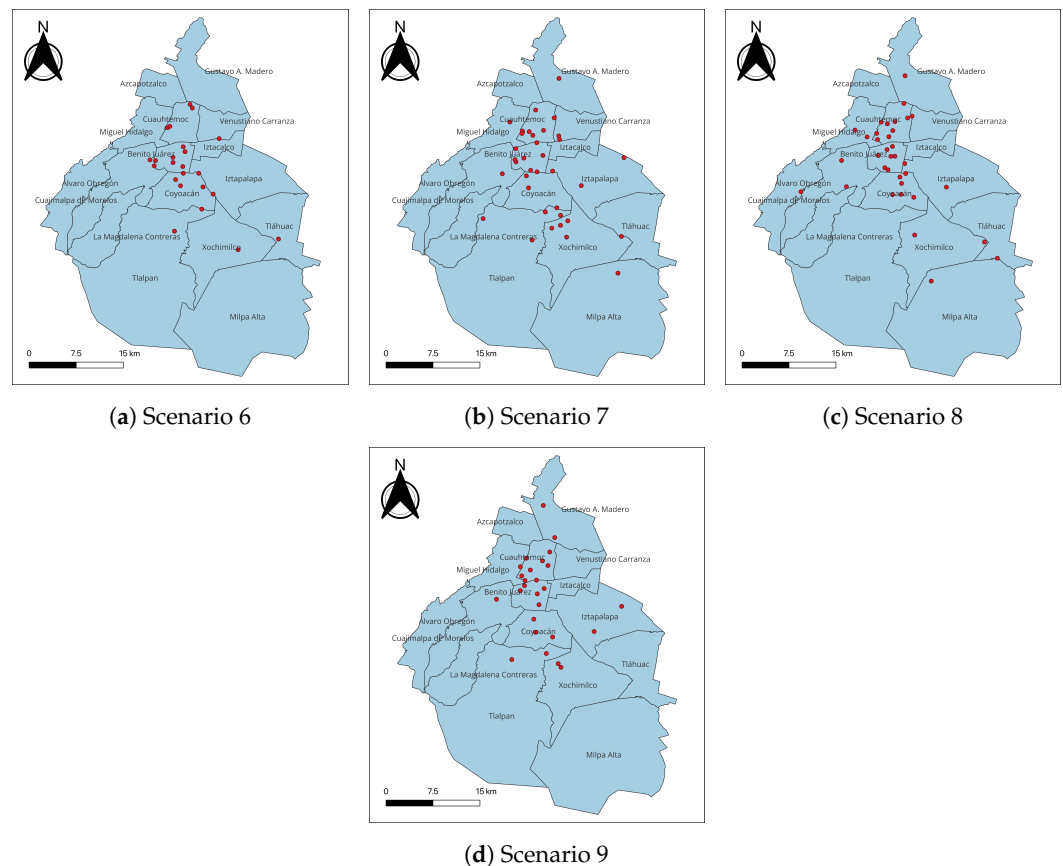


Figure 6. Last four mass-casualty scenarios generated by the CreateDistribution.ipynb script.

3.2.2. Traffic-Based Edge Weighting

Traffic conditions change rapidly after a mass-casualty incident and directly affect travel times across the transportation network. The GetTrafficData.ipynb notebook obtains up-to-date congestion information by querying TomTom's Traffic Flow API, which returns raster tiles in PNG format following an exponential tiling scheme [21]. TomTom provides global coverage, making this procedure transferable to other regions.⁴

The notebook first identifies the tiles intersecting the study area. Tile indices (x, y) are computed from the bounding box of Mexico City using coordinates in EPSG:4326 and the inverse Mercator projection. Given a selected zoom level z , these indices are combined to construct the API query URL:⁵

[https://\[baseURL\]/traffic/map/{versionNumber}/tile/flow/{style}/{z}/{x}/{y}.{format}?key=\[apiKey\]&thickness={thickness}&tileSize={tileSize}](https://[baseURL]/traffic/map/{versionNumber}/tile/flow/{style}/{z}/{x}/{y}.{format}?key=[apiKey]&thickness={thickness}&tileSize={tileSize})

Parameters:

- baseUrl: base URL of TomTom services (default: api.tomtom.com).
- versionNumber: API version (currently 4).
- style: tile rendering style (set to relative0).
- z: zoom level.
- x: x coordinate of the tile.
- y: y coordinate of the tile.
- format: image format (currently PNG only).
- apiKey: personal API access key.
- thickness: line width multiplier (integer value).
- tileSize: tile size (256 or 512).

Each request returns a single RGBA tile. The notebook iterates over all required tiles, applies a binary transparency mask, posterizes the RGB channels, and converts each pixel to a congestion class using TomTom's relative with 0 style. Only tiles intersecting the Mexico City polygon are retained (Figure 7). Experimental tests showed that a zoom level of $z = 16$ provides sufficient detail to capture all edges in the simplified transportation graph.

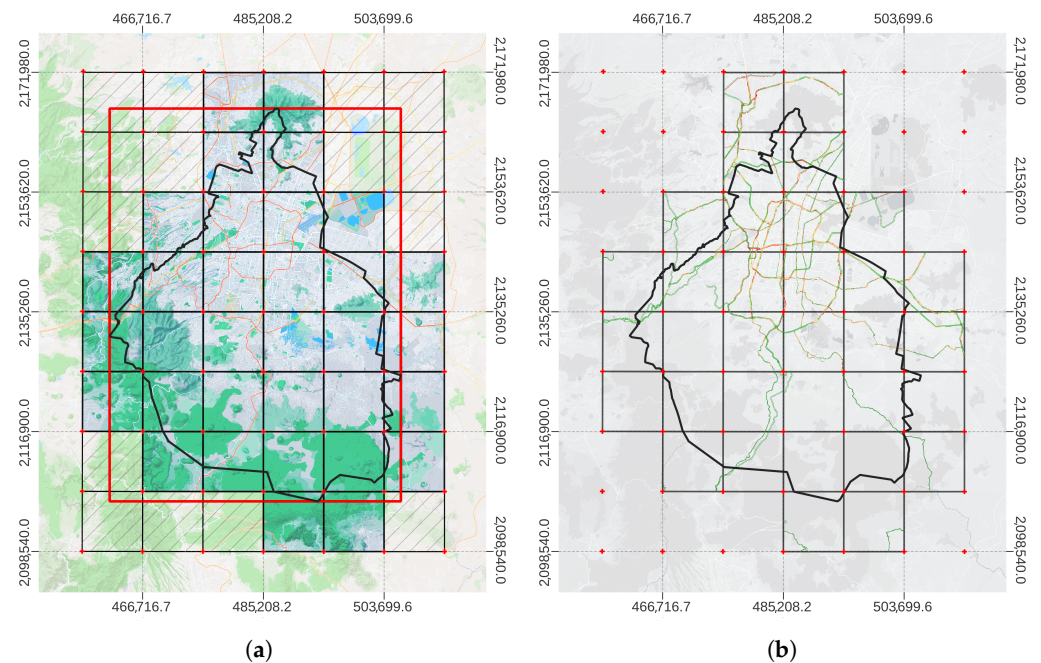


Figure 7. Traffic in Mexico City on 20 October 2023 at 6:58 p.m. obtained at different zoom levels. (a) Zoom level 12, with all tiles covering the bounding box of Mexico City (red rectangle). (b) Selection of tiles containing information on Mexico City roads. The color scheme corresponds to the default TomTom traffic rendering and is included only for illustration. Adapted from [3].

Finally, the notebook assigns a traffic class to every edge of the graph. For each edge (u, v, k) , a perpendicular sampling window is constructed, and the modal congestion value within that window is extracted. The resulting values are stored in a tabular file within the traffic_data directory and later used to compute travel-time modifiers during optimization. Algorithm 1 summarizes the complete procedure, where $T_{x,y,z}$ represents the geographic region covered by the TomTom tile with indices (x, y) at zoom level z .

Algorithm 1 Formal procedure for obtaining and processing traffic data from TomTom tiles**Require:** Study area polygon P , zoom level z , API key κ **Ensure:** Set of values $D(u, v, k)$ representing the traffic level per edge of the transportation graph

```

1: procedure GETTRAFFICDATA( $P, z, \kappa$ )
2:   Compute the bounding box  $BB(P)$  of the study area
3:   Determine indices  $(x_{\min}, y_{\min}), (x_{\max}, y_{\max})$  such that  $BB(P) \subseteq T_{x,y,z}$ 
4:   for all  $(x, y)$  in the range  $[x_{\min}, x_{\max}] \times [y_{\min}, y_{\max}]$  do
5:     Build the query URL:
6:      $url = header\_url + \{z, x, y\} + params(\kappa)$ 
7:     Retrieve the image  $I_{RGBA}(x, y)$ 
8:     Define a binary mask:
9:      $\mathcal{M}(x, y) = \begin{cases} 1, & \text{if } \alpha(x, y) \geq 250, \\ 0, & \text{otherwise.} \end{cases}$ 
10:    Apply the mask to the RGB channels:
11:     $I_{RGB}(x, y) = I_{RGBA}^{(RGB)}(x, y) \odot \mathcal{M}(x, y)$ 
12:    Posterize each color channel:
13:     $I_{RGB}^{(c)}(x, y) = \left\lfloor \frac{4 I_{RGB}^{(c)}(x, y)}{255} + \frac{1}{2} \right\rfloor \cdot \frac{255}{4}, \quad c \in \{R, G, B\}$ 
14:    Combine the channels to obtain a single-band image:
15:     $I_{mono}(x, y) = \mathbf{w}^\top I_{RGB}'(x, y)$ , where  $\mathbf{w} = (1, 5, 10)$ 
16:    if centroids  $d_i$  and tolerances  $\sigma_i$  are not defined then
17:      Select 30 samples per class to form  $\mathcal{S}_i = \{s_{i,1}, s_{i,2}, \dots, s_{i,30}\}$  for style relative0
18:      Compute  $d_i = \text{mean}(\mathcal{S}_i)$  and  $\sigma_i = \text{std}(\mathcal{S}_i)$ 
19:    end if
20:    for all  $(x, y) \in \Omega(I_{mono})$  with  $\mathcal{M}(x, y) = 1$  do
21:      where  $\Omega(I_{mono})$  denotes the spatial domain of the image
22:      Determine class:
23:       $c(x, y) = \arg \min_i |I_{mono}(x, y) - d_i|$ 
24:      if  $|I_{mono}(x, y) - d_{c(x,y)}| < \sigma_{c(x,y)}$  then
25:         $C(x, y) = c(x, y)$ 
26:      else
27:         $C(x, y) = 0$ 
28:      end if
29:    end for
30:    Save the single-band raster  $C(x, y)$ 
31:  end for
32:  for all edges  $(u, v, k)$  of the transportation graph do
33:    Compute the midpoint  $m_{(u,v,k)} = \frac{1}{2}(v + u)$ 
34:    Compute the perpendicular line  $\ell_{(u,v,k)}^\perp$  passing through  $m_{(u,v,k)}$ 
35:    Define the sampling window center  $c_{(u,v,k)}$  as the endpoint of  $\ell_{(u,v,k)}^\perp$  shifted to the right
36:    Define the sampling window  $\mathcal{W}_{(u,v,k)}$  centered at  $c_{(u,v,k)}$ 
37:    Compute the modal class value:
38:     $t(u, v, k) = \text{mode}\{C(x, y) \mid (x, y) \in \mathcal{W}_{(u,v,k)}\}$ 
39:    where  $\text{mode}(\cdot)$  returns the most frequent class value (ties resolved by choosing the smallest class)
40:    if  $t(u, v, k) = 0$  then
41:      Assign free-flow traffic  $t(u, v, k) = 1$ 
42:    end if
43:    Record the value  $D(u, v, k) = t(u, v, k)$ 
44:  end for
45:  return  $D$ 
46: end procedure

```

3.3. Instance Assembly, Multi-Objective Optimization, and Solution Visualization

Block (C) integrates the final three notebooks of the pipeline—LoadInstances, Optimization, and Visualization. Concurrently, these scripts combine the harmonized network, the reduced candidate-base set, the synthetic mass-casualty scenarios, and the traffic-derived edge weights to construct complete problem instances; solve them using a multi-objective evolutionary algorithm; and generate visual, map-based summaries of the resulting emergency response configurations.

3.3.1. Instance Construction and Cost-Matrix Generation

The LoadInstances.ipynb notebook assembles all elements required to define a complete emergency-response instance. Specifically, it loads (i) the simplified transport network generated in Block (A), (ii) the reduced set of candidate bases, (iii) the synthetic demand

scenarios generated in Block (B), and (iv) the traffic-derived congestion levels obtained from the GetTrafficData.ipynb script.

For each scenario, the notebook computes two cost matrices: a distance matrix and a travel-time matrix. These are obtained by running shortest-path searches on the simplified graph using Dijkstra’s algorithm, with edge weights defined as either geometric length or length adjusted by the traffic class. Both OSMNX (CPU-based) and cuGraph (GPU-accelerated) implementations are supported, allowing for efficient computation based on the available hardware.

The resulting instance files include (i) the set of candidate bases, (ii) the synthetic demand points with estimated casualties, and (iii) the corresponding distance and travel-time matrices. All instance outputs are stored in the instances_data directory, one file per scenario. Table 5 summarizes the structure of the generated instance files.

Table 5. Description of variables contained in each instance file.

Variable	Type	Description
len_sources	int	Total number of sources (<i>candidate bases</i>)
len_targets	int	Total number of targets (<i>demand points</i>)
len_ambulances	int	Total number of ambulances
max_cost	int	Maximum cost for a solution, length in meters of the route
sources	dict	List with the IDs of the candidate bases
targets	dict	List with the IDs of the demand points
source_i_to_node	dict	Equivalence between the position in the ST matrix (row <i>i</i>) and the ID (node attribute) of the candidate base
source_node_to_i	dict	Equivalence between the ID (node attribute) of the candidate base and the position in the ST matrix (row <i>i</i>)
target_i_to_num	dict	Equivalence between the position in the ST matrix (column <i>i</i>) and the ID (num attribute) of the demand point it represents
target_num_to_i	dict	Equivalence between the ID (num attribute) of the demand point represented by the position in the ST matrix (column <i>i</i>)
target_num_to_node	dict	Equivalence between the ID of the demand point and the two terminal vertices of the nearest edge
target_node_to_num	dict	Equivalence between the terminal vertices of the edge nearest to each demand point
capabilities	list	List containing the capacities of each candidate base on the sources field
demands	list	List containing the demand at each point in the targets field
matrix_ST	np.array	Cost matrix from each candidate base to each demand point
routes	list(cudf)	List storing the predecessor table for each demand point, which can be used to obtain the routes used to calculate the cost matrix

3.3.2. Multi-Objective Optimization of Emergency Response Plans

The Optimization.ipynb notebook solves each emergency-response instance using the NSGA-II evolutionary algorithm. For a given scenario, the script loads: (i) the synthetic demand points, (ii) the reduced set of candidate bases, and (iii) the distance and travel-time matrices generated in the previous stage. The optimization problem seeks a set of non-dominated solutions that minimize two objectives: the total travel distance and the total travel time required to serve all demand points. Two operational constraints are imposed: a maximum permitted route length per vehicle and a maximum number of bases that may be activated.

NSGA-II produces a Pareto front of alternative deployment plans. Each solution specifies which candidate base serves each demand point, together with its associated objective values. All optimized solutions for a scenario are stored in the solutions directory.

The structure of these solution files is summarized in Table 6. This representation facilitates downstream analysis, including Pareto-front visualization, decision-space exploration, and post-optimization filtering.

Table 6. Structure of the solution files generated by Optimization.ipynb.

Element	Description
Header	Normalization value used for comparing Pareto fronts across scenarios (inverse hyper volume)
Solution ID	Unique identifier corresponding to one non-dominated solution
Objective 1	Total distance required to serve all demand points
Objective 2	Total travel time computed with traffic-adjusted weights
Constraint 1	Maximum route length; encoded as a negative value representing feasibility
Constraint 2	Maximum number of active bases; also represented as a negative feasibility value
Assignment Vector	For each demand point i , the ID of the candidate base assigned to serve it

3.3.3. Visualization of Optimized Deployment Plans

The Visualization.ipynb notebook provides the final stage of the workflow, rendering the optimized solutions for each scenario. For a selected mass-casualty instance, the script loads the corresponding Pareto front and identifies the solution that activates the fewest candidate bases while still satisfying all operational constraints. This solution is then visualized on an interactive map generated with the Folium library, using OpenStreetMap as the base layer and Mapbox tiles for enhanced cartographic detail (a valid API key stored in mapbox_api_key.txt is required).

The resulting HTML map highlights the spatial configuration of the activated bases, the affected demand points, and the routes linking each demand point to its assigned base. This facilitates the visual inspection of deployment patterns, spatial coverage, and overall geographic behavior of the solution. Figure 8a shows the full-extent visualization for Scenario 0, including all activated bases and the associated demand points. On the other hand, Figure 8b provides a zoomed-in view of a subregion to illustrate local assignment patterns and routing structure. The map is produced as a mashup combining Mapbox basemap tiles with routes computed directly by the notebook.

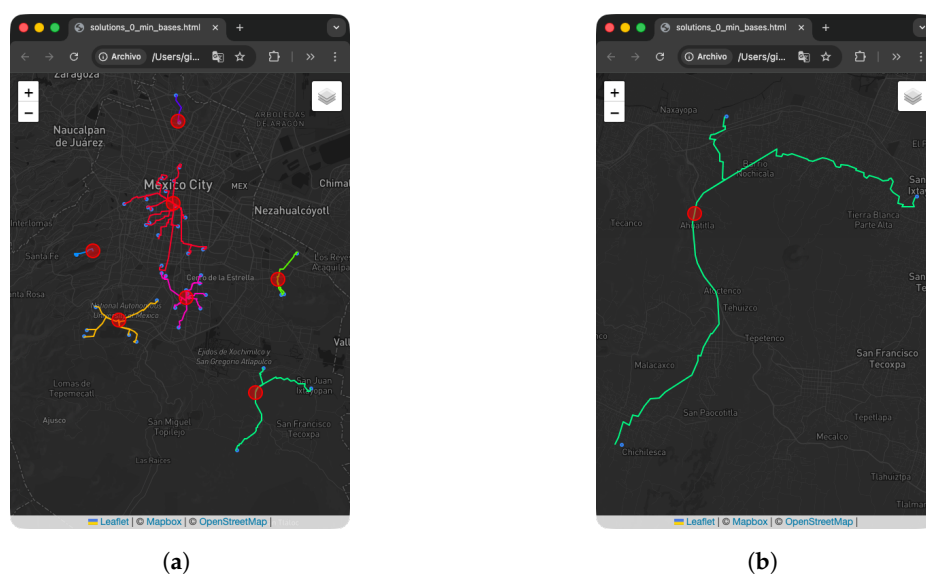


Figure 8. Visualization, over Mexico City's map, of solution with minimum number of enabled candidate bases for scenario 0. (a) Full extent of damaged buildings data. (b) Zoom-in of attended locations by some candidate base. Basemap: © Mapbox, © OpenStreetMap (and contributors).

4. Conclusions

The methodology proposed in this research enables a substantial reduction of the search space for candidate emergency post locations intended to serve multiple points of attention that may arise following water-related, seismic, geological, or other types of disasters, particularly when prior planning or predefined placement strategies are unavailable. Although the candidate bases are identified statistically, additional sites can be excluded by incorporating complementary incident data, such as real-time traffic information.

By obtaining (i) the set of roads that can function as ambulance candidate bases with their respective capacities, (ii) the demand points of care along with their estimated number of victims, and (iii) the transportation network graph, the ambulance location–allocation problem can be formulated as a bi-objective optimization model. This model aims to determine which bases should be activated (location) and which demand points each base should serve (allocation), subject to eight constraints that ensure solution consistency and feasibility, while considering the maximum response time and the maximum number of enabled candidate bases.

Using kernel density estimation (KDE) based on historical data of buildings damaged during the September 19, earthquake, hypothetical scenarios were generated. For each scenario—whether hypothetical or real—a specific instance of the problem was produced. The data-reading and preprocessing stages require approximately the same amount of time, whereas the computational cost of the cost-matrix calculation grows proportionally with the number of demand points.

The NSGA-II algorithm, implemented with the proposed parameter configuration, successfully identifies solutions to the formulated model, balancing the two defined objectives and the multiple imposed constraints. This pipeline can easily be adapted to other urban areas with similar seismic or metropolitan features, such as Santiago, Los Angeles, or Tokyo. This adaptability is achievable by TomTom’s extensive global traffic data coverage and the fully parameterized Jupyter notebooks available in the repository.

All data and scripts are made publicly available to ensure transparency and reproducibility. The repository hosted on Google Colab allows results to be reproduced without requiring local installation or configuration. At the same time, a mirrored dataset is also archived in Zenodo as an additional long-term access option.

Author Contributions: Conceptualization, M.M.-P. and G.G.; methodology, M.M.-P. and G.G.; software, M.M.-P. and M.S.-P.; validation, A.L. and M.T.-R.; formal analysis, A.L. and M.T.-R.; investigation, A.L. and M.S.-P.; resources, M.T.-R. and M.S.-P.; data curation, M.M.-P. and G.G.; writing—original draft preparation, G.G. and M.M.-P.; writing—review and editing, A.L. and M.T.-R.; visualization, M.M.-P. and M.S.-P.; supervision, G.G. and M.T.-R.; project administration, G.G. and M.S.-P.; funding acquisition, A.L. and M.S.-P. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially sponsored by the Instituto Politécnico Nacional under grants 20250037, 20250285, 20251107, 20251126, 20251128, and by the Secretaría de Ciencia, Humanidades, Tecnología e Innovación (SECIHTI) under grant 1183927.

Data Availability Statement: All data and scripts supporting the findings of this study are publicly available. The complete dataset, including raw layers, processed geospatial files, synthetic scenarios, and all Jupyter Notebooks, hosted in a public Google Drive repository: https://drive.google.com/drive/folders/1bVR9WXVuiDekKnxwmu4DeNld_X5iY_9c (accessed on 4 December 2025). For long-term preservation, version control, and citation, a mirrored archive of the dataset is available in Zenodo: <https://doi.org/10.5281/zenodo.17845383> (accessed on 4 December 2025). Both repositories contain identical versions of the dataset and source code, allowing full reproducibility of the results reported in this article.

Conflicts of Interest: The authors declare no conflicts of interest.

Notes

- ¹ Other resource types, such as search-and-rescue personnel or emergency medical staff, can also be analyzed.
- ² AGEB (Basic Geostatistical Area -in Spanish Área Geoestadística Básica-) is a geographic unit defined by INEGI for the statistical division of Mexican territory. These areas can be urban or rural and operate as the smallest unit for demographic and socioeconomic analysis.
- ³ In practice, only attributes should be removed; however, some OSMNX functions internally reference auxiliary fields that may also be discarded.
- ⁴ If TomTom services are temporarily unavailable, archived or historical traffic tiles may also be used.
- ⁵ The URL is a query template for the TomTom Traffic API and is not a navigable hyperlink. It contains placeholder parameters (e.g., {x}, {y}, {apiKey}) required to construct requests programmatically.

References

1. Campos, V. *PHTLS: Soporte Vital de Trauma Prehospitalario*; Jones & Bartlett Learning: Burlington, MA, USA, 2020.
2. Medina-Perez, M.; Legaria-Santiago, V.K.; Guzmán, G.; Saldana-Perez, M. Search Space Reduction in Road Networks for the Ambulance Location and Allocation Optimization Problems: A Real Case Study. In Proceedings of the Telematics and Computing, Puerto Vallarta, Mexico, 13–17 November 2023; Mata-Rivera, M.F., Zagal-Flores, R., Barria-Huidobro, C., Eds.; Springer: Cham, Switzerland, 2023; pp. 157–175. [\[CrossRef\]](#)
3. Medina-Perez, M.; Guzmán, G.; Saldana-Perez, M.; Legaria-Santiago, V.K. Medical Support Vehicle Location and Deployment at Mass Casualty Incidents. *Information* **2024**, *15*, 260. [\[CrossRef\]](#)
4. Nurwatik; Hong, J.H. A framework: Implementation of smart city concept towards evacuation route mapping in disaster management system. *IOP Conf. Ser. Earth Environ. Sci.* **2019**, *389*, 012043. [\[CrossRef\]](#)
5. Bhatti, F.; Shah, M.A.; Maple, C.; Islam, S.U. A Novel Internet of Things-Enabled Accident Detection and Reporting System for Smart City Environments. *Sensors* **2019**, *19*, 2071. [\[CrossRef\]](#)
6. Darwassh Hanawy Hussein, T.; Frikha, M.; Ahmed, S.; Rahebi, J. BA-CNN: Bat Algorithm-Based Convolutional Neural Network Algorithm for Ambulance Vehicle Routing in Smart Cities. *Mob. Inf. Syst.* **2022**, *2022*, 7339647. [\[CrossRef\]](#)
7. Veisi, O.; Du, D.; Moradi, M.A.; Guasselli, F.C.; Athanasoulis, S.; Syed, H.A.; Müller, C.; Stevens, G. Designing SafeMap Based on City Infrastructure and Empirical Approach: Modified A-Star Algorithm for Earthquake Navigation Application. In Proceedings of the 1st ACM SIGSPATIAL International Workshop on Advances in Urban-AI, Hamburg, Germany, 13 November 2023; ACM: New York, NY, USA, 2023; pp. 61–70. [\[CrossRef\]](#)
8. Comisión Nacional para el Conocimiento y Uso de la Biodiversidad. División Política Estatal 1:250,000. 2019. Available online: <http://geoportal.conabio.gob.mx/metadatos/doc/html/dest2019gw.html> (accessed on 28 October 2025).
9. Instituto Nacional de Estadística y Geografía (INEGI). Límite de Alcaldías (Áreas Geoestadísticas Municipales). 2023. Available online: <https://datos.cdmx.gob.mx/dataset/alcaldias> (accessed on 28 October 2025).
10. Secretaría de Gestión Integral de Riesgos y Protección Civil. Atlas de Riesgo Sísmico. 2021. Available online: <https://datos.cdmx.gob.mx/dataset/atlas-de-riesgo-sismico> (accessed on 28 October 2025).
11. Gaceta Oficial de la Ciudad de México. No. 685bis, Protocolo del Plan de Emergencia Sísmica. 2021. Available online: https://data.consejeria.cdmx.gob.mx/portal_old/uploads/gacetas/18652b057da05daee4cf7a0784093fff.pdf#page=5.25 (accessed on 28 October 2025).
12. Instituto Nacional de Estadística y Geografía (INEGI). Polígonos de Manzanas de la Ciudad de México. 2023. Available online: <https://datos.cdmx.gob.mx/dataset/poligonos-de-manzanas-de-la-ciudad-de-mexico> (accessed on 28 October 2025).
13. Agencia Digital de Innovación Pública “Sistema Abierto de Información Geográfica (SIGCDMX)”. Catastro de la Ciudad de México. 2021. Available online: https://sig.cdmx.gob.mx/datos/descarga#d_datos_cat (accessed on 28 October 2025).
14. Instituto de Planeación Democrática y Prospectiva. Hospitales públicos y Privados en Operación de la ZMVM. 2022. Available online: <https://datos.cdmx.gob.mx/dataset/hospitales-publicos-y-privados-en-operacion-de-la-zmvm> (accessed on 28 October 2025).
15. Secretaría de Gestión Integral de Riesgos y Protección Civil. Centros de Acopio. 2021. Available online: https://datos.cdmx.gob.mx/dataset/centros_acopio (accessed on 28 October 2025).
16. Secretaría de Gestión Integral de Riesgos y Protección Civil. Refugios Temporales. 2023. Available online: <https://datos.cdmx.gob.mx/dataset/refugios> (accessed on 28 October 2025).
17. Agencia Digital de Innovación Pública “Sistema Abierto de Información Geográfica (SIGCDMX). Datos de Uso de Suelo. 2022. Available online: https://sig.cdmx.gob.mx/datos/descarga#d_datos_seduvi (accessed on 28 October 2025).

18. International Organization for Standardization (ISO). *ISO 19115-1:2014—Geographic Information—Metadata—Part 1: Fundamentals*; ISO: Geneva, Switzerland, 2014. Available online: <https://www.iso.org/standard/53798.html> (accessed on 27 November 2025).
19. Comisión Nacional de Protección Civil y Centro Nacional de Prevención de Desastres. Guía para la Reducción del Riesgo Sísmico. Available online: http://www.atlasnacionalderiesgos.gob.mx/descargas/Gu_a_RRS-Final.pdf (accessed on 7 November 2025).
20. Secretaría de Gestión Integral de Riesgos y Protección Civil de la Ciudad de México and Urzúa, Myriam. Protocolo del Plan de Emergencia Sísmica. Available online: https://data.consejeria.cdmx.gob.mx/portal_old/uploads/gacetas/18652b057da05daee4cf7a0784093fff.pdf (accessed on 7 November 2025).
21. TomTom Developers. Zoom Levels and Tile Grid. 2023. Available online: <https://developer.tomtom.com/map-display-api/documentation/zoom-levels-and-tile-grid> (accessed on 23 August 2024).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.