PolRoute-DS: a Crime Dataset for Optimization-based Police Patrol Routing

Bruno Cunha Sá[1], Gustavo Muller[1], Maicon Banni[1], Wagner Santos[1,2]
Marcos Lage[1], Isabel Rosseti[1], Yuri Frota[1], Daniel de Oliveira[1]

[1] Fluminense Federal University, Niterói, Rio de Janeiro, Brazil
{bcunha,gustavomuller,maiconbanni,wagnergs}@id.uff.br,
{mlage,rosseti,yuri,danielcmo}@ic.uff.br
[2] Polícia Militar do Estado do Rio de Janeiro

Abstract. It is a well-known fact that criminality is an open, yet important, issue in most urban centers worldwide. Especially in Brazil, creating solutions to reduce crime rates is a top priority. To reduce crime rates, many cities are adopting predictive policing techniques. Predictive policing techniques are highly based on extracting valuable knowledge from a massive dataset that contains information about times, locations, and types of past crimes. The extracted knowledge is then used to provide insights to police departments to define where the police must maintain its presence. These datasets may also be used for a critical predictive policing task: defining where police patrols should patrol. Such patrols are commonly defined to cover a series of crime hot spots (areas that present high criminality levels) and have some restrictions to be considered (number of available police officers, vehicles, *etc.*). Thus, defining the route for each police vehicle is a complex optimization problem, since in most cases, there are many hot spots and the existing resources are scarce, *i.e.*, the amount of vehicles and police available is much smaller than necessary. Unfortunately, high-quality crime rates data are not easy to obtain. Aiming to tackle this problem, this article proposes the PolRoute-DS dataset, a dataset designed to foster the development and evaluation of police routing approaches in large urban centers. The PolRoute-DS combines the spatial structure of the city of interest (in the context of this article, the city of São Paulo) represented as a connected and directed graph of street segments enriched with criminal data obtained from public sources. PolRoute-DS is available for public use under the *Creative Commons By Attribution 4.0 International* license (CSV and PostgreSQL versions) and can be downloaded at https://osf.io/mxrgu/.

Categories and Subject Descriptors: H.2.8 [**Database Management**]: Database Applications

Keywords: Crime Dataset, Predictive Policing, Police Routing

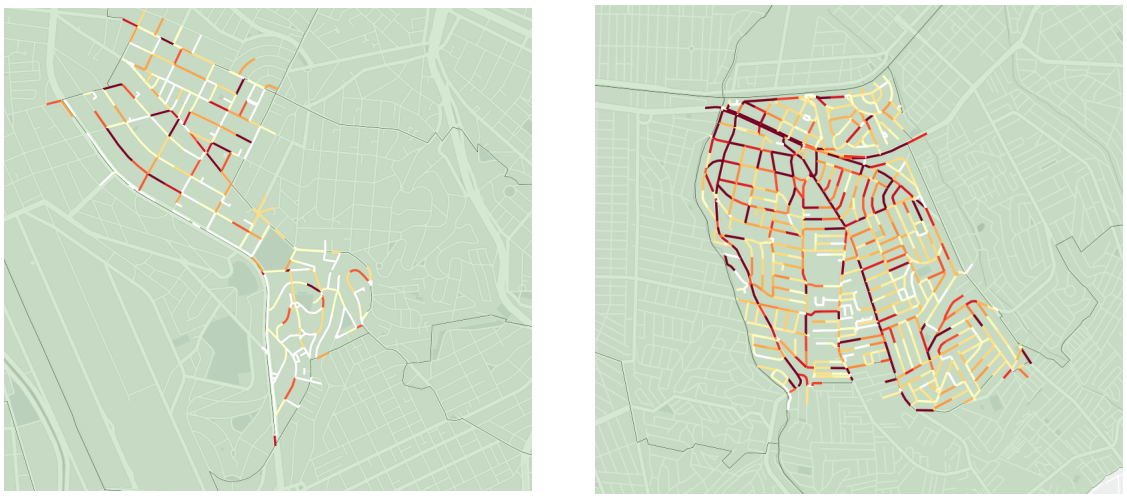
1. INTRODUCTION

The concept of Smart Cities has gained considerable attention over the last decade [Shapiro 2006]. Several initiatives have been proposed in the most varied domains of knowledge, *e.g.*, healthcare [Caban and Gotz 2015; Omar et al. 2021], transport system [Ota et al. 2017], manufacturing [Mylonas et al. 2021], trash management [Fabbri et al. 2020], urban planning [Miranda et al. 2020], *etc.* According to the *Cities in Motion Index* from the IESE Business School [Berrone and Ricart 2020], a smart city can be evaluated by the policies and solutions adopted in the areas of (i) Governance, (ii) Public Administration, (iii) Urban Planning, (iv) Technology, (v) Environment, (vi) International Connections, (vii) Social Support, (viii) Human Capital and (ix) Economy. In this article, we focus on the *Urban Planning* dimension, especially in the *Public Safety* area.

Reducing crime rates in most Brazilian cities is a historic challenge, especially considering large urban centers like São Paulo, Rio de Janeiro, Fortaleza, and Recife [Lourenço et al. 2018]. In recent years, impacted by the national economic crisis faced since 2015, the crime rate in these cities has

shown a fickle, yet rising, trajectory. For example, the total number of crime occurrences registered in the city of São Paulo[1] went from 152,481 in the first quarter of 2018 to 155,307 in the same period of 2019. In 2020, despite the social distancing policy imposed by the COVID-19 pandemic, the number of crimes reached 158,063 in the first quarter. Finally, the index dropped to 145,598 in the first three months of 2021 (period still impacted by the COVID-19 pandemic). In fact, due to the complexity of these large urban centers (*e.g.*, population and socioeconomic differences in the city's regions, the existence of areas with the informal occupation of difficult access, availability of streets, *etc.*), defining public policies capable of reducing crime rates is far from trivial.

One of the possible policies, which has been adopted in many cities and presents good results, is *predictive policing*. The idea of predictive policing is to make police officer's work evident to the population, whether through the presence of police officers at strategic points in the city or police patrols. Using this policing strategy, the police officers aim at maintaining public order, preventing the occurrence of crimes or infractions. In the context of predictive policing, one of the most complex tasks to be performed is the *definition of police patrol routes* as discussed by Saint-Guillain *et al.* [2021]. In general, the existing resources are scarce, *i.e.*, the amount of vehicles and police officers available is much smaller than necessary. Thus, the police officers must define *a priori* which specific routes the patrols will cover. This type of route definition ends up prioritizing the so-called *Hot Spots* [Reis et al. 2006], *i.e.*, areas of the city where the crime rate is high. To illustrate the occurrence of hot spots, let us take two regions of the city of São Paulo: Alto da Mooca and Itaim Paulista. According to the crime index of São Paulo[2], in 2017, Alto da Mooca was the region with the city's lowest crime rate, while Itaim Paulista is a region with a high crime rate. Fig. 1a presents a heat map over the streets to present hot spot areas. In Fig. 1a, most streets of Alto da Mooca do not show any crime occurrence for the selected period (from October to November 2017). On the other hand, in Fig. 1b, most of the streets of the Itaim Paulista region present high crime rates for the same period, thus characterizing several crime hot spots.



(a) Alto da Mooca Region (b) Itaim Paulista Region

Fig. 1: Crime Occurrences from October to November 2017 in Alto da Mooca and Itaim Paulista

Thus, considering that there may be multiple crime hot spots to cover and the scarcity of resources, the efficient definition of vehicle routes becomes a top priority. Although defining the routes (also called routing) of patrols can be performed manually, this process can become laborious and error-prone. The ideal scenario is to automatically define the routes based on historical crime data (also

[1] `http://www.ssp.sp.gov.br/Estatistica/Trimestralis.aspx`
[2] `https://infograficos.estadao.com.br/cidades/criminalidade-bairro-a-bairro/`

called data-driven policing [Dewinter et al. 2020]). There is a huge volume of crime data in urban regions available for analysis and use, either provided by government institutes such as the São Paulo Public Security Office (SSP-SP)[3] or the Institute of Public Security (ISP)[4] in Rio de Janeiro, or by sites that map crimes using crowdsourcing such as *Onde fui roubado*[5] and WikiCrimes[6].

The automatization of the patrol routing can be performed using different computational strategies, such as optimization techniques [Saint-Guillain et al. 2021]. Some authors of this article have been developing a solution to this problem using Metaheuristics[7] such as GRASP [Resende and Ribeiro 1997]. For this type of solution to be developed, a database with the number of crime occurrences organized by type (*e.g.*, car theft, femicide, murder, *etc.*) must be available, and the occurrences must be associated with city topology. Thus, in this article, we represent the city's streets as a graph $G^c = (V, E, Q)$. In this representation, the set $E = E^1 \cup E^2$ is composed of street segments, and these segments can be either one-way (E^1) or two-way (E^2). We reinforce that graph G^c is composed of street segments, *i.e.*, we split all city streets into segments of length between 150 and 200 meters. This strategy prevents long streets/avenues/roads from being associated with very high crime rates while the crimes may occur in a small part of the street. Furthermore, graph G^c is formed by the set of vertices V of order $n = |V|$ representing connections between two streets or junctions of segments of the same street. Fig. 2 presents an example of a graph generated for the region around the Assis Chateaubriand Art Museum of São Paulo (MASP) located on Paulista Avenue.

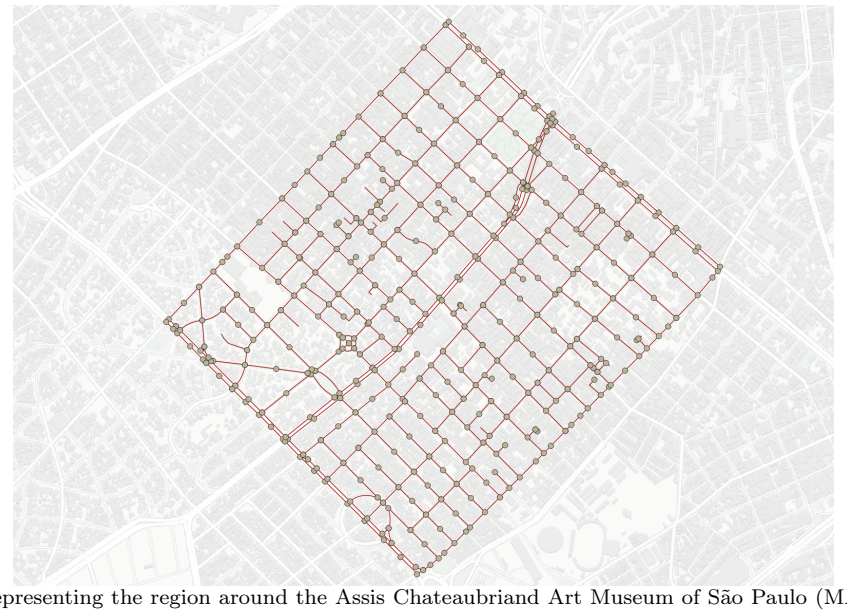


Fig. 2: Graph representing the region around the Assis Chateaubriand Art Museum of São Paulo (MASP) located on Avenida Paulista - Adapted from Sá *et al.* [2021].

To be able to generate the police routes using optimization techniques, three primary data types are required: the data points that represent the crime occurrences (*i.e.*, events); a geospatial layer depicting the configuration of streets; and a user-defined graph with each vertice presenting a unique identifier number (*e.g.*, 86179, 172652, 83350, *etc.*, as presented in Fig. 3). All these three types of data have to be combined in a single dataset. Thus, in this article, we propose the dataset entitled `PolRoute-DS`, created to enable the development and testing of approaches to generating police routes

[3] `http://www.ssp.sp.gov.br/`
[4] `http://www.isp.rj.gov.br/`
[5] `https://www.ondefuiroubado.com.br/`
[6] `http://www.wikicrimes.org/`
[7] The details of this optimization approach are beyond the scope of this article.

in urban centers. `PolRoute-DS` combines the spatial structure of the city of interest (*i.e.*, a connected graph of street segments) with criminal data obtained from public sources. The graph is represented in a relational database that follows the snowflake schema, traditionally used in the development of Data Warehouses [Inmon 1996]. This approach aims at allowing the representation of the total number of crime events of each type by segment and to calculate the aggregations in the temporal dimension. These operations are fundamental not only for vehicle routing algorithms but also for traditional data analysis applications. In the current version, `PolRoute-DS` only contains the data of the city of São Paulo.

Fig. 3: Graph that represents the region around the Assis Chateaubriand Art Museum of São Paulo (MASP) located on Avenida Paulista.

This article is an extension of a conference paper published in the Proceedings of the 2021 Dataset Showcase Workshop, held in conjunction with the 2021 Brazilian Symposium on Databases (SBBD) [Sá et al. 2021]. In this extended version, we provided more details about the dataset. We have also added a Background section and improved the related work section. The remainder of this article is organized as follows: In Section 2 we formalize the crime graph represented in `PolRoute-DS`. In Section 3, we discuss related work. In Section 4, we present the structure of `PolRoute-DS` and its generation process. In Section 4.3, we analyze some statistics of the dataset. In Section 4.4, we present a usage example of the dataset. In Section 4.6, we present download instructions and how to cite the dataset. Finally, Section 5 presents conclusion and future work.

2. BACKGROUND: CRIME GRAPH

A crime graph plays a vital role in the police patrol routing task. Based on the information provided by a crime graph, the optimization approach can define an optimal (or near-optimal) solution. Thus, in this section, we provide more details about the crime graph that is represented in `PolRoute-DS`. As aforementioned, let $G^c = (V, E, Q)$ be a graph, denoted as crime graph, with vertex set V of order $n = |V|$ representing street corners or street divisions and edge set $E = E^1 \cup E^2$ of size $m = |E|$ representing (segments of) streets, where E^1 and E^2 represent the set of one-way and two-way streets, respectively. Let us also define $Q = \{Q_1, Q_2, ..., Q_q\}$ as the partition of V into q disjoint components (zones), *i.e.*, $Q_1 \cup Q_2 \cup ... \cup Q_q = V$ and $Q_u \cap Q_o = \emptyset$, for every $u, o = 1...q$ with $u \neq o$.

We also denote by $Q[i]$ the index of the component of vertex $i \in V$: *i.e.*, $i \in Q_{Q[i]}$. In addition, for each edge $\{i, j\} \in E$, we define f_{ij} and l_{ij} as the crime factor and length of the edge, respectively. It is worth noticing that the crime factor f_{ij} may be calculated based on the number of occurrences c_x^{ij} of each type of crime $x = 1...d$ in edge $ij \in E$ (assuming that we have d types of crimes) and the weighted associated with each type α_x (the weight defines the importance of a specific type of crime in the analysis) in the dataset, *i.e.*, $f_{ij} = \sum_{x=1}^{d} c_x^{ij} \alpha_x$. In Fig. 4, we present an example of a crime graph with nine vertices and three zones (Q_1, Q_2, and Q_3) where one-way streets are represented as single lines, two-way streets are represented as dashed lines, and edge labels (f, l) represent the crime factor and the length of the edge, respectively.

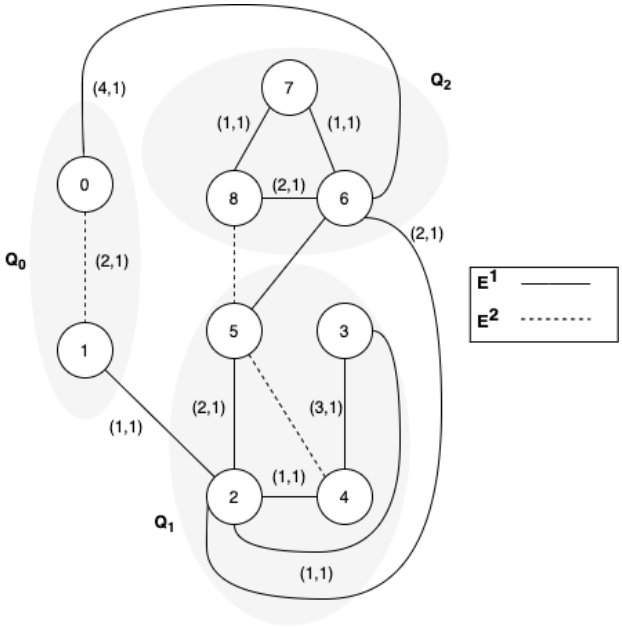


Fig. 4: An example of a crime graph

3. RELATED WORK

Many papers in the literature deal with the problem of police routing patrol and consume a crime dataset similar to PolRoute-DS. We can classify these papers into two categories: (i) the ones that aim at generating the police route and use a crime dataset, and (ii) the ones that propose a crime dataset to be used for other approaches. If we consider the first category, most of the proposed approaches are based on a graph representation of the city. In this section, we discuss some of these works that produce datasets that are somehow similar to PolRoute-DS. Chawathe [2007] models a network of streets using a graph-based representation, called the *street-graph*. In the proposed graph, the vertices represent intersections, and edges represent street segments. It is worth noticing that in the graph proposed by Chawathe [2007], the edge weights represent the benefit of patroling this segment. Although the concept of representing the benefit of patrolling a specific area of the graph is interesting, this dataset is not available for general use.

Chen *et al.* [2015] propose an approach for defining patrol routes using ant colony optimization technique. Although the authors also model the street as a graph, they consider just a subset of types of crimes (mobile phone and car theft) since the focus is on on-foot patrolling. The dataset used in this approach is not available for general use. Melo *et al.* [2005] use the concept of Society agent to simulate the environment and propose a police patrol route. Differently from the previous approaches, Melo *et al.* [2005] model the environment as a group of independent agents that represent criminals,

police officers, and commercial stores. These agents can "walk" through the map and then the routes are defined as a set of points that a police officer should go through during a determined period at a defined speed to reach the criminal agent.

The aforementioned approaches are based on static solutions, *i.e.*, they consume the crime dataset and generate a patrol route as an outcome. However, these routes are not adaptive through time, *i.e.*, routes are not modified in the case of a change in the crime occurrence pattern. This way, many approaches focus on the dynamic vehicle routing problem that considers changes in the environment to adjust the produced route at runtime. Bertsimas *et al.* [1991] propose an approach that implements the dynamic traveling repairman problem for the police routing scenario. In turn, Gendreau *et al.* [2006] analyze the maximal expected coverage relocation problem for emergency services (that also considers police vehicles) and also Patrascu *et al.* [2016] and Shen *et al.* [2018] focus on optimization strategies for the emergency vehicles, especially police vehicles and ambulances. All these approaches consume a crime graph to generate the route, but this graph is not available (it is hard-coded within the tools).

The second category of approaches groups the ones that aim at providing a general use crime dataset, *i.e.*, the dataset can be used by any routing or analytical approach. Most of these approaches are public, similarly to `PolRoute-DS`. Yoo [2019] discusses how to materialize, in a single database, spatial (*e.g.*, map of a city) and criminal occurrences that can be useful to build solutions to reduce criminal activity. Similarly to `PolRoute-DS`, the solution proposed by Yoo [2019] uses as a basis the concepts of Data Warehousing. The Crime Data-Warehouse, from the Royal Canadian Mounted Police[8], gathers into one Data Warehouse multiple datasets about crime events officially reported to law enforcement authorities in the city of Burnaby, Canada. This dataset has approximately 4.4 million events described by attributes such as date, time, location, and criminal information. Differently from `PolRoute-DS`, the approach proposed by Yoo [2019] does not represent the data as a graph.

Spadon *et al.* [2017] propose using complex networks to analyze the crime patterns in a city. The authors propose metrics to detect regions with a high crime rate in a city. To evaluate their proposal, the authors use a dataset called `G-FranC` [Scabora et al. 2019] which has a complex network structure combined with crime data that occurred in the city of San Francisco, in the United States. As in `PolRoute-DS`, `G-FranC` proposes the representation of city streets as a graph. However, `G-FranC` does not take into account if a street is one-way or two-way since its objective is to identify regions with a high crime rate and not to define routes for police patrolling.

4. PROPOSED DATASET: `POLROUTE-DS`

As aforementioned, `PolRoute-DS` combines the topology of the city with crime data, provided by external sources. In its current version, `PolRoute-DS` contains data imported from the open data portal of the state police of São Paulo. Following, we present the data acquisition and transformation process, the `PolRoute-DS` schema, dataset statistics, and some practical usage examples.

4.1 Data Acquisition and Transformation Process

To create `PolRoute-DS`, we follow the architecture presented in Fig. 5. The process starts (Step 1 in Fig. 5) by importing data from public data sources. The ETL (*Extract, Transform, Load*) component is responsible for downloading files or accessing public APIs to gather data. In the current version of `PolRoute-DS` it only contains crime data provided by the São Paulo State Police (SSP-SP)[9]. All data in the current version of `PolRoute-DS` was downloaded in August 2019.

[8] `https://www.rcmp-grc.gc.ca/en/evidence-and-reports-data-warehouse`
[9] `http://www.ssp.sp.gov.br/transparenciassp/Consulta.aspx`

The data obtained from the SSP-SP web portal are made available in XLS files organized by year and type of crime. In the context of this article, the following types of crime are considered: (i) Femicide, (ii) Mobile Phone Theft, (iii) Mobile Phone Robbery, (iv) Murder, (v) Robbery, (vi) Vehicle Theft, and (vii) Vehicle Robbery. Despite being available on the web portal, all data associated with (i) Death as a result of police intervention and (ii) Suspicious death are not used. The total amount of files obtained after download, the data volume, and the time window considered for each type of crime are presented in Table I.

Table I: Statistics of Downloaded Files

Crime Type	# of files	Total Size	Time Window
Femicide	52	1,2 MB	04/2015 - 07/2019
Mobile Phone Theft	115	1,3 GB	01/2010 - 07/2019
Mobile Phone Robbery	115	2,2 GB	01/2010 - 07/2019
Murder	198	126,6 MB	01/2003 - 07/2019
Robbery	196	7,6 MB	04/2003 - 07/2019
Vehicle Theft	199	1,6 GB	01/2003 - 07/2019
Vehicle Robbery	199	1,9 GB	01/2003 - 07/2019

All downloaded files present the same set of attributes. In this article, we consider only the attributes Date of Occurrence, Time of Occurrence, Period of Occurrence (*e.g.*, Morning, Afternoon, Night, Dawn), Street Name (Address), Number (Address), Latitude, Longitude, City Name, and State Name. In a pre-processing step, all downloaded files are combined into a single file containing all occurrences of all types of crimes.

Since all types of crime are combined in a single file, we have to add a new attribute to define the Crime Type (this information is extracted from the title of each downloaded file). In addition, empty fields (*i.e.*, ";;") are replaced by a string to identify a null value (*i.e.*, ";NULL;"). Once formatted, the file is uploaded to a PostgreSQL database whose schema resembles a crime Data Mart (which is presented in Subsection 4.2).

When all crime data are already loaded and available for querying in the Data Mart, the process of creating the city's street graph starts (Step 2 in Fig. 5). This step is performed with the support of QGIS. QGIS is a Geographic Information System (GIS) tool that allows for the visualization and analysis of georeferenced data in multiple layers. In QGIS, data can be stored as points, lines, or polygons.

We use QGIS to partition the streets into segments of size ranging from 150 to 200 meters (this variation in segment size depends on the length of the street since the division into segments of the same size is not possible in most scenarios). Then, all crime data are loaded into the software to be combined with the edges of the city street graph. In this process, tuples with latitude/longitude outside the city of São Paulo area are not considered (*e.g.*, there are some outliers in the raw data as crime occurrences located in Europe or in the Atlantic Ocean).

For each crime occurrence found in the Data Mart, a *Spatial Join* is executed in QGIS with the street graph, using the operation *Join Attributes by Nearest*. This operation requires two layers as input, one with crime data and one with street data. For each feature of the occurrences layer, the features closest to the street layer are identified. The output of the algorithm is composed of the geometries of the street layer associated with the attributes of the crime occurrence layer. Given an $e \in E$ street segment in the graph, only crimes that occurred within a distance of up to 0.001 decimal degrees (*i.e.*, approximately 100 meters) from e are associated with the street segment. Once the association is identified, the number of occurrences per segment is calculated.

The resulting graph (henceforth named as *crime graph*) is then stored in the database. Finally, in Step 3, the extractor component queries the database and generates CSV files containing the stored

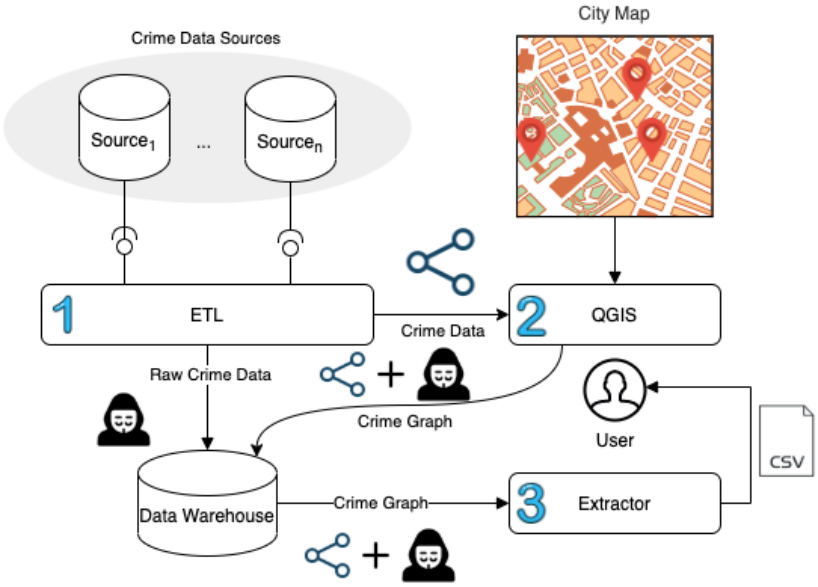


Fig. 5: Data Acquisition and Transformation of `PolRoute-DS` - Adapted from Sá *et al.* [2021]

data so that they can be easily imported into existing tools. These files can be used as input for routing approaches based on *hot spots* like the one our group has been developing.

It is worth mentioning that `PolRoute-DS` can be updated as soon as new data are available at the crime data sources. According to the principles of data warehousing, the loading process of Dimensions and Fact tables may be classified into two categories: (i) total and (ii) incremental. In the first type, the complete loading of data is performed every time new data are available. On the other hand, the incremental type only considers the new data, inserting them into the data warehouse. For the sake of simplicity, in the context of `PolRoute-DS` both dimension tables and fact tables are reloaded every time an update has to be performed. We are still working on a mechanism to perform an incremental update and to version the street graph (since the topology of the city may change over time).

4.2 Database Schema

After the ETL step, which includes loading crime data and mapping the crime occurrences to each graph segment, the data are loaded into a database modeled as a Data Mart that follows the snowflake schema [Inmon 1996]. This data mart is composed of *fact tables* and *dimension tables* to store the data. The fact tables contain the values of interest for the query (*e.g.*, the number of occurrences of a certain type of crime), while the dimensions qualify this quantitative data. A data mart follows one of the following schemas: (i) star schema and (ii) snowflake. The star schema is based on the design of a central fact table, while the dimensions are associated with the fact using foreign keys. In the snowflake schema, dimension tables are normalized with "one-to-many" relationships. We follow the snowflake schema to represent the hierarchy in the topology of the city (*e.g.*, districts, zone, *etc.*).

Fig. 6 presents the database schema of `PolRoute-DS`. The schema is composed of seven tables: (i) *Crime*, (ii) *Segment*, (iii) *Vertex*, (iv) *Time*, (v) *District*, (vi) *Zone*, and (vii) *Neighborhood*. The *Crime* table stores the number of occurrences of each type of crime associated with a certain segment of the map at a certain period. The table has nine attributes: *segment_id* (primary key and foreign key) that defines which segment the number of crimes are associated with, *time_id* (primary key and foreign key) that defines which date the crimes occurred, *total_femicide* which defines the total number of occurrences of femicide in the specific segment in a specific period of time, *total_murder* which represents the total number of homicides in the specific segment in a specific period of time,

total_ robbery which represents the total occurrences of robbery in the specific segment in a specific period of time, *total_ theft_ mobile* which represents the total number of cell phone thefts in the specific segment in a specific period of time, *total_ robbery_ mobile* which represents the total number of cell phone robberies, *total_ theft_ vehicle* which represents the total vehicle theft in the specific segment in a specific period of time, and *total_ robbery_ vehicle* which represents the total number of vehicle robbery in the specific segment in a specific period of time. It is worth noticing that the crime table can store data pre-aggregated for large time windows, to speed up the queries. In addition, the structure of this table has to be updated when a new type of crime is considered, *i.e.*, new attributes have to be added to the table.

The *Time* table stores the time periods with occurrences of crimes in multiple temporal resolutions (*e.g.*, hour, day, month, *etc*). It is composed of the attributes *time_ id* (primary key), *day* an integer that represents the day part of date, *month* an integer that represents the month part of a date, *year* an integer that represents the year part of a date, *day_ of_ week* (boolean that defines whether a given date is a business day or weekend) and the *period* (morning, afternoon, night or dawn). Thus, the *Time* table stores a specific date/time when a crime happened. The *Segment* table stores the segments that are part of the street graph. Segments are associated with the amount of criminal occurrences in the Crime table. The *Segment* table is composed of the attributes *id_ segment* (primary key), *vertex_ start_ id* and *vertex_ end_ id* (foreign keys) that represent the start and end vertices of the segment, *geometry* (QGIS's internal representation of the segment), *one_ way* which informs whether the street is one-way or two-way, and *length* which represents the size of a segment in meters.

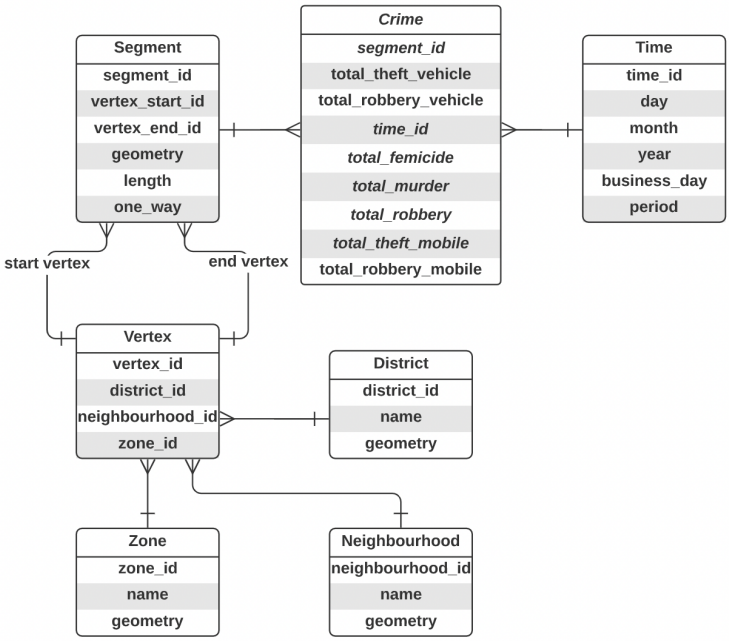


Fig. 6: PolRoute-DS database schema.

The *Zone* table represents areas that the user can delimit on the map to create a patrol route or to perform a hot spot analysis. The zone is not an official city organization, but defined by the user as needed. It is composed of *zone_ id* (primary key), *name* (string) and *geometry* (QGIS's internal representation of the zone). The *District* table defines the districts of a city. It is composed of the attributes *district_ id* (primary key), *name* (string) and *geometry* (QGIS's internal representation of the district). The *Neighborhood* table represents the official neighborhoods of a city. It is composed of

the attributes *neighbourhood_ id* (primary key), *name* (string) and *geometry* (QGIS's internal representation of the neighbourhood). Finally, the *Vertex* table represents the vertices of the street graph. It is composed of the attributes *vertice_ id* (primary key), *name* (string) and *geometry* (QGIS's internal representation of the vertex). In addition, it has foreign keys *district_ id*, *neighborhood_ id* and *zone_ id* that reference the neighbourhood, district and zone associated to a vertex, respectively.

4.3 Statistics of `PolRoute-DS`

In this subsection, we present some statistics of the current version of `PolRoute-DS`. Fig. 7 shows the number of occurrences *per* type of crime (×1,000). It is important to point out that not all occurrences of all types of crime that can be obtained from the SSP-SP web portal are considered in the construction of the `PolRoute-DS`. In addition, several occurrences do not present location information that allows for associating the crime occurrence with a segment of the graph. Thus, Fig. 7 shows the total number of occurrences in the downloaded data (in gray) and the total number of occurrences that are part of `PolRoute-DS` (*i.e.*, useful, in black).

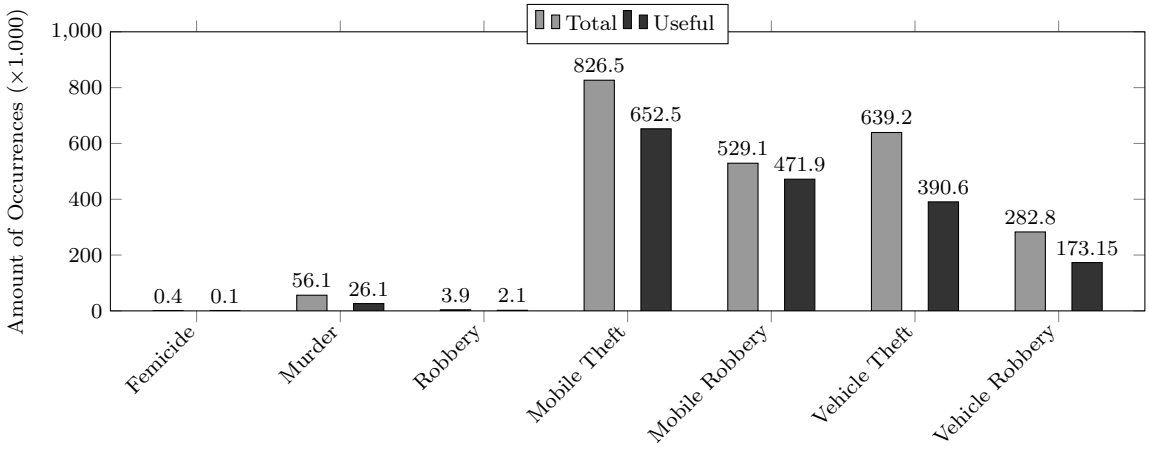


Fig. 7: Number of Crime Occurrences per Type of Crime (×1.000).

Besides the number of occurrences used in the construction of `PolRoute-DS`, we present in Table II the number of tuples in each table of the data model. It is important to emphasize that by following the dimensional modeling (snowflake schema), `PolRoute-DS` already has several tuples that represent aggregated data in time, *e.g.*, for a given segment, we already have the total number of crimes pre-aggregated by day, month, and year. This pre-aggregation eases both the query formulation (thus avoiding using aggregate functions) and the generation of the input for the routing heuristic that has been developed.

4.4 Case Study - Optimization-based Police Patrol Routing

In this section, we present a usage example to illustrate the potential of `PolRoute-DS`. The example is in the context of generating police patrol routes through metaheuristics, which consume data from `PolRoute-DS`. Let us take as an example the district of Sacomã, in the city of São Paulo. It is divided between the neighborhoods of Anchieta, Moinho Velho, and São João Clímaco. In Fig. 8(a), 2 zones are defined within the district. These zones define regions in which routes will be generated by the metaheuristic. There are two types of routes: intra-zone and inter-zone. Intra-zones routes are the ones that cover segments within a specific zone. On the other hand, inter-zone routes are the ones that contain segments from multiple zones.

Table II: Total Number of Tuples per Table

Table	Number of Tuples
Crime	7.267.353
Time	28.593
Zone	115
Segment	226.975
Vertex	180.598
District	97
Neighbourhood	323

Once the zones are defined in the application, the map with the criminal factor of each segment can be visualized (Fig. 8(b)). The segments in dark red are those with the highest values for the crime factor. In this example, we calculate the crime factor of a segment from vertex i to vertex j as follows (assuming that there are seven types of crimes as presented in Table II):

$$f_{ij} = \sum c_x^{ij} \alpha_x = c_f^{ij} \alpha_f + c_m^{ij} \alpha_m + c_r^{ij} \alpha_r + c_{mt}^{ij} \alpha_{mt} + c_{mr}^{ij} \alpha_{mr} + c_{ct}^{ij} \alpha_{vt} + c_{cr}^{ij} \alpha_{vr}$$

where f = femicide, m = murder, r = robbery, mt = mobile theft, mr = mobile robbery, vt = vehicle theft and vr = vehicle robbery. Thus, the user may define weights for each type of crime depending on the interest. Let us consider that the police department is interested in routes for avoiding mobile phone (main goal) and vehicle (secondary goal) theft. Then, the crime factor is calculated as:

$$f_{ij} = \sum c_{mt}^{ij} \times 10 + c_{vt}^{ij} \times 5$$

It is worth noticing that the weights in this example are empirically defined. The user can define different weights according to his/her needs. Also, the user can consider different periods, as specific zones can be more dangerous in the present date than they were in past periods. The user can also visualize which segments are considered in a specific zone (Fig. 8(c)). Finally, when the zones and associated segments are chosen (with their associated crime factors), the heuristic can define the best routes according to the crime graph and the restrictions imposed (number of available police vehicles, number of available police officers, *etc.*). In the example presented in Fig. 8, seven fixed police officers (blue pins in Fig. 8(d)) and eight police vehicles are defined to patrol the zone. We also define the maximum distance of 2.4 km for a patrol. In Fig. 8(d) one can visualize the routes produced by the metaheuristic. It is important to emphasize that PolRoute-DS can be used by any routing approach, not just the approach that has been developed by our group.

4.5 Current Limitations of PolRoute-DS

Although the data downloaded from the SSP-SP web portal covers a period from 2003 to 2019, we can consider as a limitation of the dataset the fact that the current version only considers data until 2019. However, it should be noted that the occurrences related to the years 2020 and 2021 can be loaded *a posteriori* as discussed in Subsection 4.1. Another limitation is that the current version of PolRoute-DS only considers seven types of crimes. The authors are aware that there are other types of crimes that could be considered, but this would require changes in the database schema as well as in the ETL process.

4.6 Download and Citation

The PolRoute-DS (PostgreSQL dump and CSV files) and the QGIS scripts can be downloaded at https://osf.io/mxrgu/. The dataset is available for community under the *Creative Commons By Attribution 4.0 International* license. The use of PolRoute-DS requires citation to the *dataset* DOI

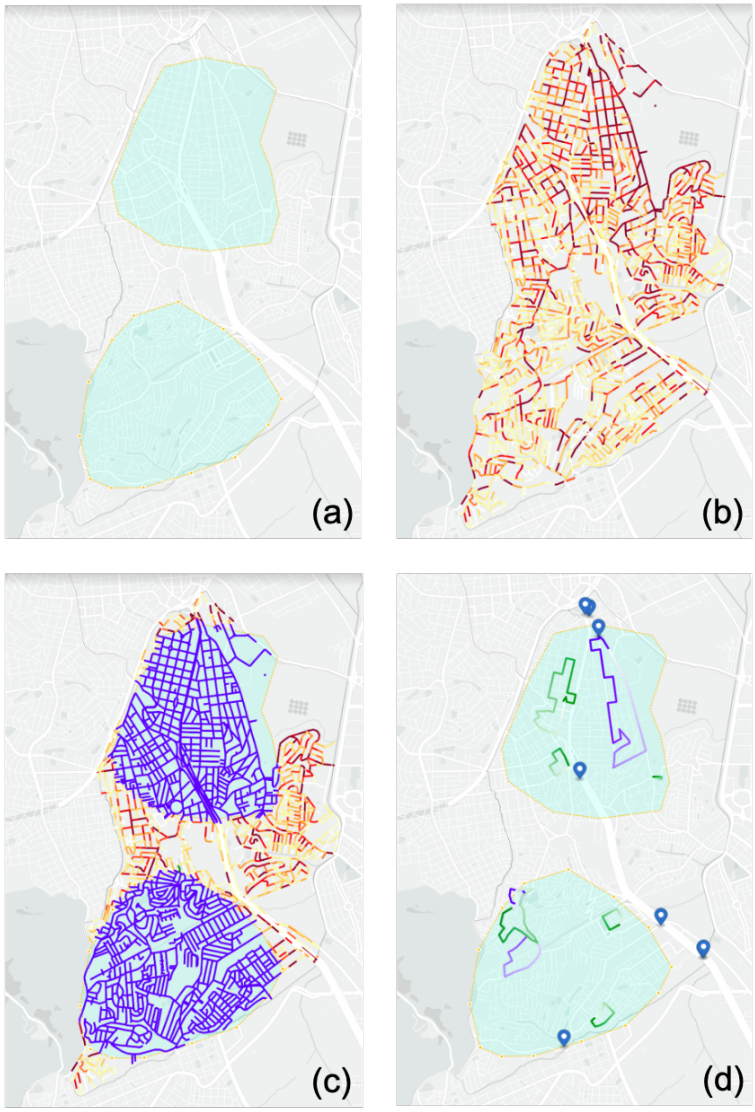


Fig. 8: (a) Zone Definition; (b) Visualization of Hot Spots (accoording to the crime factor associated to each segment); (c) Visualization of the segments in each zone; (d) Generated Police Patrol routes - Adapted from Sá *et al.* [2021]

`10.17605/OSF.IO/MXRGU` and to the present article. The routing application used in the case study which consumes `PolRoute-DS` can be accessed at `http://crimedashboard.herokuapp.com/`.

5. CONCLUSIONS

Reducing criminality in large urban centers is an open, yet important, issue. One of the most common approaches used to reduce criminality is patroling areas of the city where the incidence of crime events is high (also called hot spots). Defining the routes that police officers should follow to patrol is a complex process. The patrol route should consider both the topology of the city and the high incidence of crimes in specific areas. An efficient patrol route may prevent individuals from committing crimes and respond to incidents at runtime. However, to produce an efficient patrol route is far from trivial since we have to consider scarce police resources and large areas to cover.

To provide necessary data for generating police patrol routes, in this article, we present the

PolRoute-DS dataset, which combines the topology of the city of São Paulo (represented as a graph) with crime data, provided open data web portals (*e.g.*, SSP-SP). The generated crime graph can be used as input for routing approaches based on crime data, especially those based on hot spots. The contribution of this article is a dataset available for public use. PolRoute-DS can be obtained from https://osf.io/mxrgu/.

As future work, we plan to expand the Polroute-DS to consider new types of crimes, *e.g.*, death as a result of police intervention, suspicious death, illegal possession of weapons, illegal possession of narcotics, drug trafficking, murder of police officers, *etc.* In addition, we also plan to update the dataset with new crime occurrences after 2019. Besides including new types of crimes, we also plan to consider data from other cities (*e.g.*, Fortaleza and Rio de Janeiro) with regions dominated by criminal factions. These regions, although present a low crime rate, are very dangerous regions that need predictive action. We also plan to include rainfall and blocked streets data, so that this information can be used in route generation (*e.g.*, it makes no sense to propose a patrol route in a street that is currently blocked by a flood).

ACKNOWLEDGEMENTS

This work was funded by CNPq and FAPERJ. This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001.

REFERENCES

Berrone, P. and Ricart, J. Iese cities in motion index 2020. *IESE Business School University of Navarra: Barcelona, Spain*, 2020.

Bertsimas, D., van Ryzin, G., and Management, S. A stochastic and dynamic vehicle routing problem in the euclidean plane. *Operations Research* vol. 39, 02, 1991.

Caban, J. J. and Gotz, D. Visual analytics in healthcare–opportunities and research challenges, 2015.

Chawathe, S. S. Organizing hot-spot police patrol routes. In *2007 IEEE Intelligence and Security Informatics*. pp. 79–86, 2007.

Chen, H., Cheng, T., and Wise, S. Designing daily patrol routes for policing based on ant colony algorithm. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* vol. II-4/W2, pp. 103–109, 2015.

Dewinter, M., Vandeviver, C., Vander Beken, T., and Witlox, F. Analysing the police patrol routing problem: A review. *ISPRS International Journal of Geo-Information* 9 (3), 2020.

Fabbri, I., Lelli, G., and Nicolino, W. Puntonet: Innovative prototype of urban trash containers improving waste sorting and widening the services offered to the city. In *Proceedings of the 9th International Conference on Smart Cities and Green ICT Systems, SMARTGREENS 2020, Prague, Czech Republic, May 2-4, 2020*, C. Klein and M. Helfert (Eds.). SCITEPRESS, pp. 29–37, 2020.

Gendreau, M., Laporte, G., and Semet, F. The maximal expected coverage relocation problem for emergency vehicles. *J. Oper. Res. Soc.* 57 (1): 22–28, 2006.

Inmon, W. H. The data warehouse and data mining. *Commun. ACM* 39 (11): 49–50, 1996.

Lourenço, V., Mann, P., Guimaraes, A., Paes, A., and de Oliveira, D. Towards safer (smart) cities: Discovering urban crime patterns using logic-based relational machine learning. In *2018 International Joint Conference on Neural Networks, IJCNN 2018, Rio de Janeiro, Brazil, July 8-13, 2018*. IEEE, pp. 1–8, 2018.

Melo, A., Belchior, M., and Furtado, V. Analyzing police patrol routes by simulating the physical reorganization of agents. In *Multi-Agent-Based Simulation VI, International Workshop, MABS 2005, Utrecht, The Netherlands, July 25, 2005, Revised and Invited Papers*, J. S. Sichman and L. Antunes (Eds.). Lecture Notes in Computer Science, vol. 3891. Springer, pp. 99–114, 2005.

Miranda, F., Hosseini, M., Lage, M., Doraiswamy, H., Dove, G., and Silva, C. T. Urban mosaic: Visual exploration of streetscapes using large-scale image data. In *CHI '20: CHI Conference on Human Factors in Computing Systems, Honolulu, HI, USA, April 25-30, 2020*, R. Bernhaupt, F. F. Mueller, D. Verweij, J. Andres, J. McGrenere, A. Cockburn, I. Avellino, A. Goguey, P. Bjøn, S. Zhao, B. P. Samson, and R. Kocielnik (Eds.). ACM, pp. 1–15, 2020.

Mylonas, G., Kalogeras, A. P., Kalogeras, G., Anagnostopoulos, C., Alexakos, C., and Muñoz, L. Digital twins from smart manufacturing to smart cities: A survey. *IEEE Access* vol. 9, pp. 143222–143249, 2021.

Omar, A. A., Jamil, A. K., Khandakar, A., Uzzal, A. R., Bosri, R., Mansoor, N., and Rahman, M. S. A transparent and privacy-preserving healthcare platform with novel smart contract for smart cities. *IEEE Access* vol. 9, pp. 90738–90749, 2021.

Ota, M., Vo, H. T., Silva, C. T., and Freire, J. Stars: Simulating taxi ride sharing at scale. *IEEE Trans. Big Data* 3 (3): 349–361, 2017.

Patrascu, M., Constantinescu, V., and Ion, A. Controlling emergency vehicles in urban traffic with genetic algorithms, 2016.

Reis, D., Melo, A., Coelho, A. L. V., and Furtado, V. Towards optimal police patrol routes with genetic algorithms. In *Intelligence and Security Informatics, IEEE International Conference on Intelligence and Security Informatics, ISI 2006, San Diego, CA, USA, May 23-24, 2006, Proceedings*, S. Mehrotra, D. D. Zeng, H. Chen, B. M. Thuraisingham, and F. Wang (Eds.). Lecture Notes in Computer Science, vol. 3975. Springer, pp. 485–491, 2006.

Resende, M. G. C. and Ribeiro, C. C. A GRASP for graph planarization. *Networks* 29 (3): 173–189, 1997.

Saint-Guillain, M., Paquay, C., and Limbourg, S. Time-dependent stochastic vehicle routing problem with random requests: Application to online police patrol management in brussels. *Eur. J. Oper. Res.* 292 (3): 869–885, 2021.

Scabora, L. d. C., Spadon, G., Rodrigues, L. S., Cazzolato, M. T., Araújo, M. V. d. S., Sousa, E. P. M. d., Traina, A. J. M., Rodrigues Junior, J. F., and Traina Junior, C. G-franc: a dataset of criminal activities mapped as a complex network in a relational dbms. In *Brazilian Symposium on Databases - SBBD*. SBC, Fortaleza, 2019.

Shapiro, J. M. Smart Cities: Quality of Life, Productivity, and the Growth Effects of Human Capital. *The Review of Economics and Statistics* 88 (2): 324–335, 05, 2006.

Shen, Y., Lee, J., Jeong, H., Jeong, J., Lee, E., and Du, D. H. C. Saint+: Self-adaptive interactive navigation tool+ for emergency service delivery optimization. *IEEE Transactions on Intelligent Transportation Systems* 19 (4): 1038–1053, 2018.

Spadon, G., Scabora, L. C., Oliveira, P. H., Araujo, M. V. S., Machado, B. B., de Sousa, E. P. M., Jr., C. T., and Jr., J. F. R. Behavioral characterization of criminality spread in cities. In *International Conference on Computational Science, ICCS 2017, 12-14 June 2017, Zurich, Switzerland*, P. Koumoutsakos, M. Lees, V. V. Krzhizhanovskaya, J. J. Dongarra, and P. M. A. Sloot (Eds.). Procedia Computer Science, vol. 108. Elsevier, pp. 2537–2541, 2017.

Sá, B., Muller, G., Banni, M., Santos, W., Lage, M., Rosseti, I., Frota, Y., and de Oliveira, D. Polroute-ds: um dataset de dados criminais para geração de rotas de patrulhamento policial. In *Anais do III Dataset Showcase Workshop*. SBC, Porto Alegre, RS, Brasil, pp. 117–127, 2021.

Yoo, J. S. Crime data warehousing and crime pattern discovery. In *Proceedings of the Second International Conference on Data Science, E-Learning and Information Systems*. DATA '19. Association for Computing Machinery, New York, NY, USA, 2019.