AIMS Mathematics

http://www.aimspress.com/journal/Math

AIMS Mathematics, 8(5): 10196–10209.
DOI: 10.3934/math.2023516
Received: 11 September 2022
Revised: 18 February 2023
Accepted: 20 February 2023
Published: 27 February 2023

Research article

A space-time model for analyzing contagious people based on geolocation data using inverse graphs

Salvador Merino[1,*], **Juergen Doellner**[2], **Javier Martínez**[1], **Francisco Guzmán**[3], **Rafael Guzmán**[4] **and Juan de Dios Lara**[3]

[1] Department of Applied Mathematics, University of Malaga, 29071, Malaga (Spain)
[2] Hasso-Plattner-Institute, University of Potsdam, Germany
[3] Department of Electrical Engineering, University of Malaga, 29071, Malaga (Spain)
[4] Department of Design and Projects, University of Malaga, 29071, Malaga (Spain)

* **Correspondence:** Email: smerino@uma.es; Tel: +34951952725; Fax: +34952132766.

Abstract: Mobile devices provide us with an important source of data that capture spatial movements of individuals and allow us to derive general mobility patterns for a population over time. In this article, we present a mathematical foundation that allows us to harmonize mobile geolocation data using differential geometry and graph theory to identify spatial behavior patterns. In particular, we focus on models programmed using Computer Algebra Systems and based on a space-time model that allows for describing the patterns of contagion through spatial movement patterns. In addition, we show how the approach can be used to develop algorithms for finding ”patient zero” or, respectively, for identifying the selection of candidates that are most likely to be contagious. The approach can be applied by information systems to evaluate data on complex population movements, such as those captured by mobile geolocation data, in a way that analytically identifies, e.g., critical spatial areas, critical temporal segments, and potentially vulnerable individuals with respect to contact events.

Keywords: spatial epidemiology; disease transmission; mining contact information; human movement patters; spatial analysis; Covid-19; inverse graph
Mathematics Subject Classification: 53B05, 05C85

1. Introduction

In the mid-19th century, the English physician John Snow [9] refused to accept the theories of cholera infection prevailing at that time. Both the sharing of clothing and transmission through the air seemed to him to be insufficient explanations for what was happening in London, under an epidemic that was undermining the entire population of any social class. He established a famous practical example of spatial epidemiology, measuring the distances of each patient to the nearest water source.

Not only Covid-19 [11, 22], but many other infectious diseases [6] are forcing us, in general, to rethink new techniques for describing and discovering spatial epidemiology, that is, “the description and analysis of geographic variations in disease with respect to demographic, environmental, behavioral, socioeconomic, genetic, and infectious risk factors” [10]. To this end, we can draw on data that reflect human behavior in terms of spatial location and movement. In particular, geolocation data [24], which includes geographical location information associated with the hosting device (e.g., mobile phone), is key for modeling the spread and evolution of infectious diseases in space and time.

Using this kind of data, we cannot only track the movement patterns of individuals, but also derive general mobility patterns for the population over time. They give insight into, e.g., “the prediction of future moves, detecting modes of transport, mining trajectory patterns and recognizing location-based activities” [13]. That is, geolocation data is key data computational spatial empidemiologic models and simulations.

The idea that a microscopic coronavirus in the city of Wuhan has caused an economic crisis in Europe reminds us of Edward Lorenz’s famous speech at MIT, where he raised how the flapping of a butterfly in Brazil could cause a tornado in Texas. That is, the well-known butterfly effect and its repercussions on chaos theory.

In this work, data about spatial population movement such as geolocation data are used as key sources of insight in the way infectious diseases spread in space and time among individuals. To this end, this work proposes a mathematical foundation following an inverse methodology on the chaotic bases [18] since it raises the search of the starting attractor or “patient zero”. This procedure collects the ideas of reverse engineering, as a way of establishing the unknown initial parameters of a problem whose solutions and effects are known.

In particular, tracking and analyzing population movements are key to get insights about patterns of propagation of environmentally-transmitted pathogens. For example in the case of Covid-19, “[...] changing geography of migration, the diversification of jobs taken by migrants, the rapid growth of tourism and business trips, and the longer distance taken by people for family reunion are what make the spread of COVID-19 so differently from that of SARS” [19].

To this end, we introduce a graph-based approach to model population movements in space and time, including a health status of individuals that allows us to classify risks for individuals as well as spatial locations and temporal periods.

The main methods to be used will be differential geometry of curves adapted to the symptomatology of the disease and graph theory for its representation and monitoring [16]. In its inverse reasoning, we can establish algorithms that, e.g., facilitate finding a “patient zero”, classify possible originators of the contagion [1, 3, 12, 17, 20], or classify spatial locations with respect to the spread of pathogens.

2. Concept

As a key source of human mobility patterns, we consider mobile phone location data. The high penetration of mobile phones in the population allows a very representative survey of mobility dynamics [14].

The trajectories on which we base the monitoring of population movements are those recorded by the smartphones devices with GPS. We know that both smartphones and social networks [23] have the ability to store the paths followed by their owners, just as telephone companies can track that path followed by their customers [7, 8]. This information is collected both by these devices and by the telephone companies, and can be represented geospatially by the different navigation systems. As an example we can see the path followed by an smartphone and represented in Figure 1 using Google Maps software:

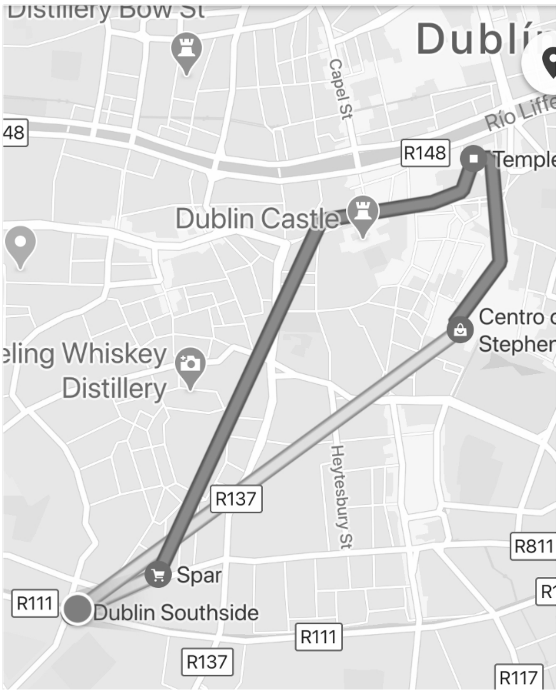

Figure 1. Geolocation register.

Definition 2.1. *For every patient P we distinguish three disjointed and sequential time intervals (as can be seen in the medical graphs in Figure 2) that we will note by:*

1. *H: where the person is "healthy". In the timeline it will be distinguished by a simple line and during this interval [5] (before infection) the value of the state variable will be* 1.
2. *C: where the person is "contagious" (incubation period). This interval or latency period will be estimated, given the patient's symptoms, based on the maximum and minimum time it could take to become "ill". It will be represented by a double line and its state variable will be* 2*. Currently this period is from 0 to 7 days aproximately.*
3. *I: where the patient is "ill". It will be represented by a triple line, together with the previous one completes the incubation time [21] and its state variable will be* 3.

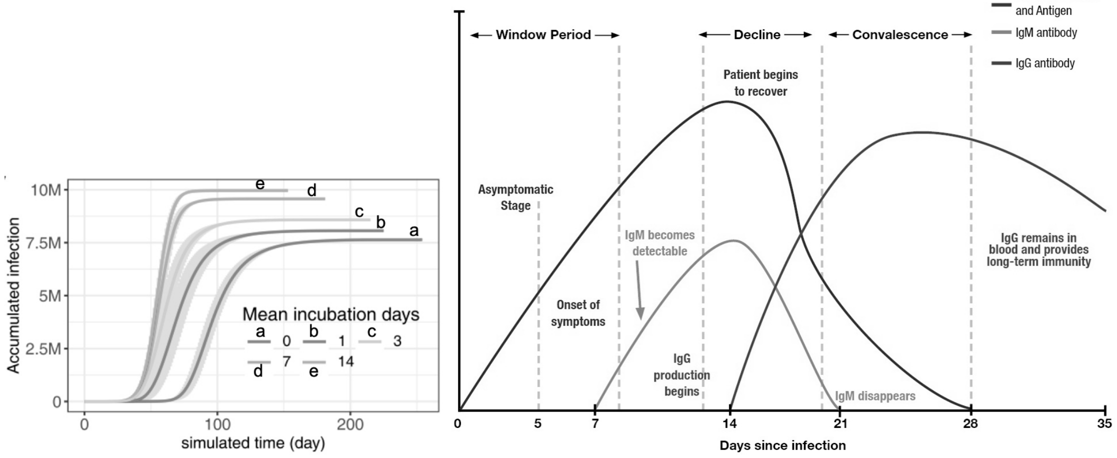

Figure 2. Incubation Time.

Definition 2.2. *To establish the path we use the notation of differential geometry by using Cartesian coordinates, incorporating the state variable over time $\rho(t) \in \{1, 2, 3\}$ remaining:*

$$\vec{\alpha}(t) = (x(t), y(t), \rho(t)) \text{ with } t \in \{H \cup C \cup I\}$$

and whose graph is represented in Figure 3.

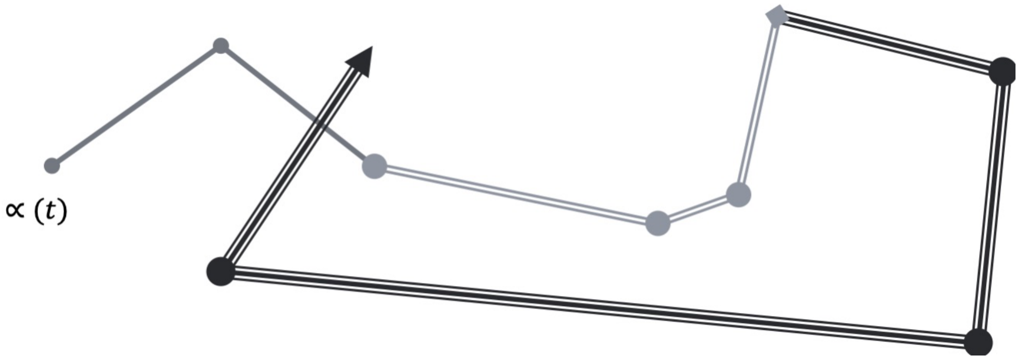

Figure 3. Patient graph.

Definition 2.3. *We will say that a contagion has occurred if, given two patients P_i and P_j it is fulfilled:*

- *the paths cross in a sufficiently close environment, i.e. there is a maximum transmission distance M such that:*

  $$\text{for an instant } t \text{ it is fulfilled that } \sqrt{(x_i(t)-x_j(t))^2+(y_i(t)-y_j(t))^2} \leq M$$

  To simplify the calculation, without loss of generality, we will consider that this distance M is null, or what is the same, they are in the same place. With this we have:

  $$\text{for an instant } t \text{ it is fulfilled that } (x_i(t), y_i(t)) = (x_j(t), y_j(t))$$

- *one patient's status must be "contagious" and the other patient's status must be "ill", i.e., $t \in C$ for P_i and $t \in I$ for P_j or vice versa. This is simplified as $\rho_i(t) \cdot \rho_j(t) = 6$ (product of a state that is "contagious" by another "ill").*

We will note the contagion between both patients as $P_i \sim P_j$.

Notation 2.4. *The point of infection of each patient P will be identified with the symbol $\otimes$ and its coordinates will be $\vec{\otimes} = (x, y, n, t)$ where:*

- *x and y are the cartesian coordinates of the place of infection*
- *n is the variable where the number of infections produced in this place is accumulated.*
- *t is the variable that indicates the furthest moment in which the contagion occurred in this place.*

It can be seen in the contagion graph of Figure 4 that a different situation occurs at each confluence $\otimes_i$ since in

$\otimes_1$ Both patients are healthy
$\otimes_2$ The second patient P_2 has been infected by the first P_1
$\otimes_3$ The first patient P_1 is still "healthy" after meeting the second P_2 who is ill. They may not have coincided in time at this intersection and the first one was not in a "contagious" state.
$\otimes_4$ Here both patients are already "ill"

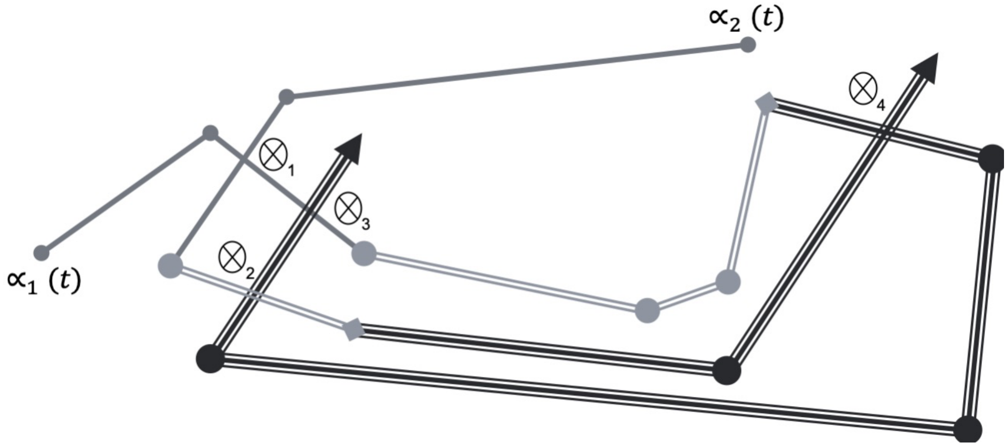

Figure 4. Contagious points graph.

3. Modelization

Procedure 3.1 (Intersections analysis)**.** *Given two different paths, $\vec{\alpha}_i$ and $\vec{\alpha}_j$, these can be subdivided into segments, being considered therefore as polygonal *. Hence, the analysis of intersections is restricted to the calculation of common points between two segments of different paths [16]. Let us consider the segments between points:*

*A polygonal sufficiently close to the curve is considered. Its existence is assured by Schoenflies theorem for polygonal Jordan curves. This concept also appears in the formulation of the length of a parametric curve, making use of the approximations by sufficiently small segments and its use in the Riemann integral.

$$S_p^i = \overline{\left(x_p^i(t_1), y_p^i(t_1)\right)\left(x_{p+1}^i(t_2), y_{p+1}^i(t_2)\right)} \in \vec{\alpha}_i$$

$$S_q^j = \overline{\left(x_q^j(t_3), y_q^j(t_3)\right)\left(x_{q+1}^j(t_4), y_{q+1}^j(t_4)\right)} \in \vec{\alpha}_j$$

1. *We check for the existence of temporal (one-dimensional) intersection, that is, time concurrence:*

$$(t_1, t_2) \cap (t_3, t_4) \neq \emptyset \quad \Leftrightarrow \quad t_1 \leq t_4 \wedge t_3 \leq t_2$$

2. *We check for geometric (two-dimensional) intersection:*

   (a) *It can be done by solving the system, as long as it is well-conditioned:*

$$\begin{cases} x_p^i(t_1) + \lambda\left(x_{p+1}^i(t_2) - x_p^i(t_1)\right) = x_q^j(t_3) + \mu\left(x_{q+1}^j(t_4) - x_q^j(t_3)\right) \\ y_p^i(t_1) + \lambda\left(y_{p+1}^i(t_2) - y_p^i(t_1)\right) = y_q^j(t_3) + \mu\left(y_{q+1}^j(t_4) - y_q^j(t_3)\right) \end{cases}$$

If there are $0 \leq \lambda \leq 1$ *and* $0 \leq \mu \leq 1$ *that fulfill both equations or are collinear, that is:*

$$\begin{vmatrix} x_{p+1}^i(t_2) - x_p^i(t_1) & x_{q+1}^j(t_4) - x_q^j(t_3) \\ y_{p+1}^i(t_2) - y_p^i(t_1) & y_{q+1}^j(t_4) - y_q^j(t_3) \end{vmatrix} = 0$$

where contagion could occur if they are coincidental:

$$\frac{x_{p+1}^i(t_2) - x_p^i(t_1)}{y_{p+1}^i(t_2) - y_p^i(t_1)} = \frac{x_{q+1}^j(t_4) - x_q^j(t_3)}{y_{q+1}^j(t_4) - y_q^j(t_3)} = \frac{x_p^i(t_1) - x_q^j(t_3)}{y_p^i(t_1) - y_q^j(t_3)}$$

In any other situation, as there are no points in common between the segments, the minimum distances between each point and the opposite segment are calculated. Finally, it is checked whether the minimum of them is below the maximum transmission distance, so that there is a possibility of contagion (as Figure 5):

$$\min\{d_1, d_2, d_3, d_4\} \leq M$$

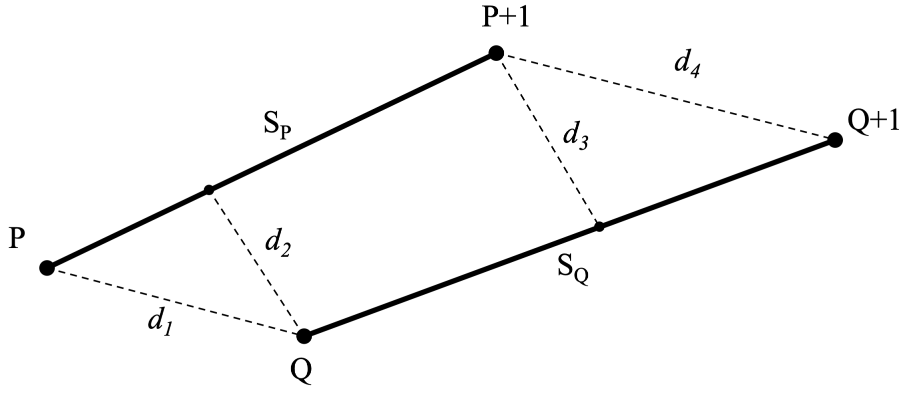

Figure 5. Distances between points and opposite segments.

In this case the execution time for n segments would be

$$\Theta(n^2)$$

(b) *Or using Bentley's & Ottmann algorithm [4], whose execution time for n segments with s points of intersection is*

$$\Theta(n\log(n) + s\log(n))$$

Procedure 3.2 (Establishing Contagion)**.** *The big data algorithm must fulfill the following phases:*

1. *Analyze the intersections of patient trajectories by pairs ($\vec{\alpha}_i$ and $\vec{\alpha}_j$) where contagion may occur [15]:*

$$\text{find } t \text{ such that } (x_i(t), y_i(t)) = (x_j(t), y_j(t)) \text{ with } \rho_i(t)\cdot\rho_j(t) = 6$$

2. *In each analysis, establish the possible point of contagion (if any) and increase both its contagion index and its most remote time variable in which a contagion occurred*

$$\text{For } \vec{\otimes} = (x, y, n, t), \text{ is executed } n = n + 1$$

3. *Patients are classified according to the sum of the indexes of all the points of contagion contained in their trajectory. In this way, we will order the patients from the one that has caused the most infections to the one that has caused the least.*
4. *Patients are classified according to the lowest value of t contained in their points of contagion. With this we can establish patient zero.*

With this procedure, we calculate the distribution of contagion points of known patients $\vec{\otimes}_k$, and the final number of contagions in each one of them $\vec{\otimes}_k$, which gives us an image of the city as shown in Figure 6:

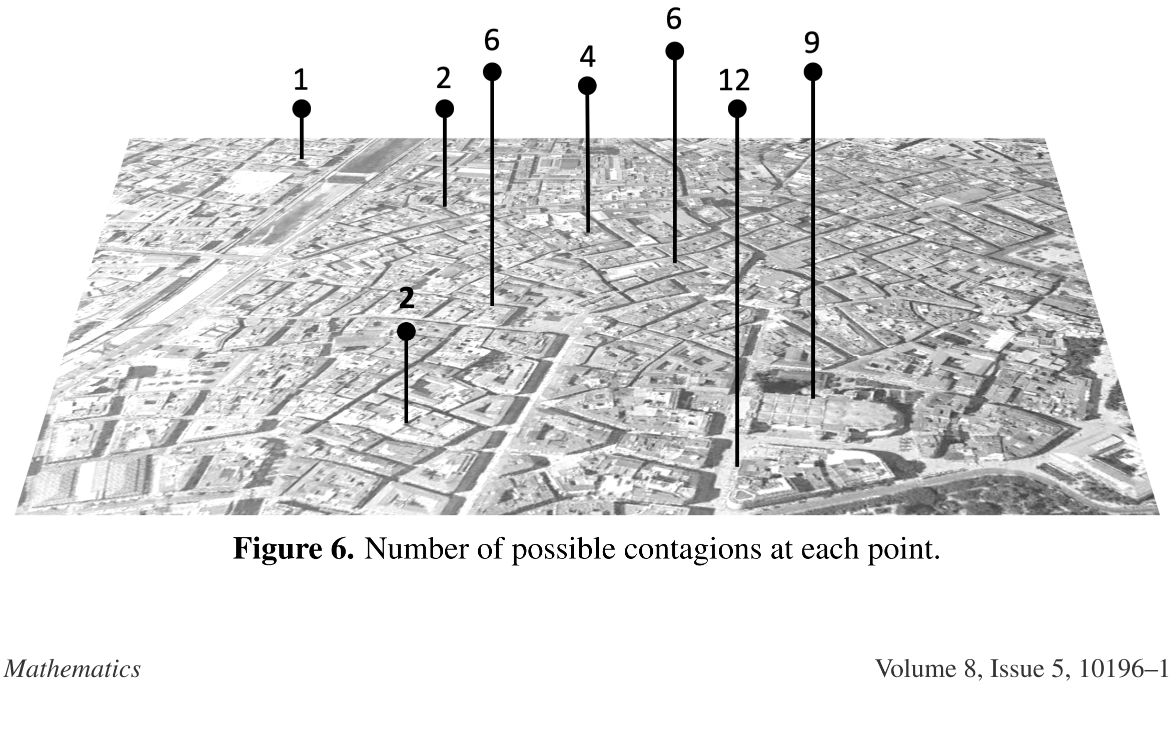

Figure 6. Number of possible contagions at each point.

4. Inverse graph

Once the possible cut-off points between known patient routes have been established, we ask ourselves whether there are any undetected patients who can be deduced from the possible points of contagion. This question involves a probabilistic analysis of what the possible path of the unknown patient would look like.

Let us suppose that there is an unknown patient that has been causing an uncontrolled contagion, which we will call P_x.

1. If the contagion has occurred at an intersection $\vec{\otimes}_{ij}$ of two "contagious" patients P_i and P_j, that is, $\rho_i(t) \cdot \rho_j(t) = 4$, it could be that:

   (a) one of them was already really ill: This forces a recalculation of the C and I intervals for both. In this case, if the one who was already ill had been P_i, he must have been previously infected by a stranger, and therefore there was a cross with P_x. Similarly for P_j.

   (b) neither of them infected the other: In this case there is a coincidence of the path of both with the one of P_x at the point of infection.

   In both cases we assign a new contagion in the point $\vec{\otimes}_{ij}$, although in the first assumption we also know that there is some point in the path of P_i or P_j in the state C where it must have crossed with that of P_x in the state I. Therefore, if we call $t(\vec{\otimes}_{ij}, C_i^0)$ and $t(\vec{\otimes}_{ij}, C_j^0)$ the time from the crossing point to the place where P_i and P_j met P_x respectively, which are C_i^0 and C_j^0, then the crossing probabilities (CP) with P_x of each interval are set as:

   $$PC(x,i) = \frac{t(\vec{\otimes}_{ij}, C_i^0)}{[C_i]}$$

   $$PC(x,j) = \frac{t(\vec{\otimes}_{ij}, C_j^0)}{[C_j]}$$

   where $[C_i]$ and $[C_j]$ are the times at which P_i and P_j have been contagious.
2. If the infection has occurred in the C range of any patient P_i, the path of P_x matches that of P_i with $\rho_i(t) = 2$ and $\rho_x(t) = 3$. In this case it is known that there is some point in the path of P_i in the state C where it should have crossed with that of P_x in the state I. Here the probability of the whole C interval is equal to 1.

The zone graph, in these cases, is represented by Figure 7

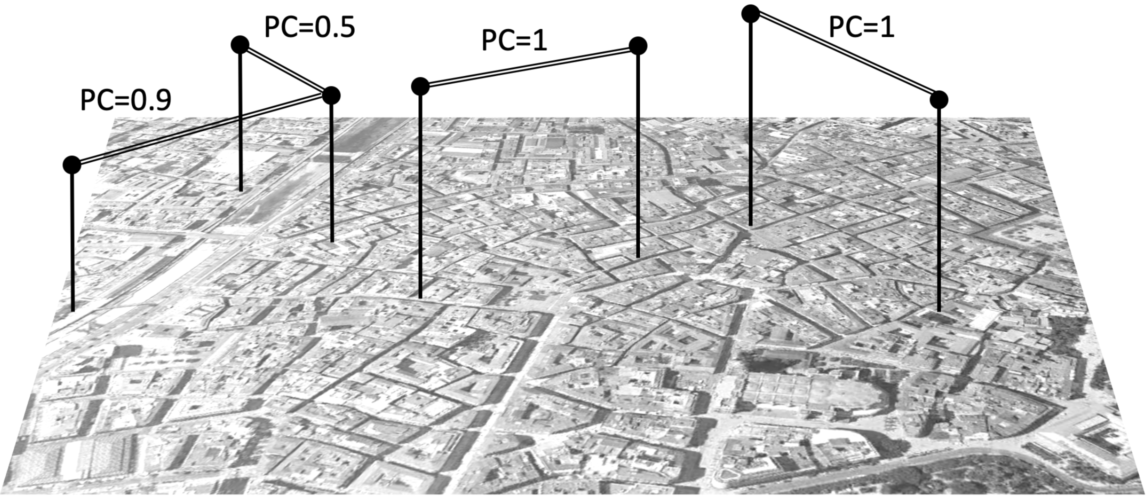

Figure 7. Intervals with probability of contagion.

Procedure 4.1 (Inverse graphs by big data)**.** *The big data algorithm will then be based on finding, of all existing paths of citizens that go through this area, those that achieve a higher sum of probabilities at the intersections of the N intervals of known contagion [12].*

1. *To do this, the route followed by each citizen's smartphone P_x is analysed and its intersections with the probable paths, assuming that this citizen was permanently ill:*

$$\vec{\alpha}_x(t) = (x(t), y(t), 3)$$

2. *This would cause a contagion at each intersection and would add probabilities to its final value.*

$$\sum_{i=1}^{N} PC(x,i)\ con\ P_x \sim P_i$$

3. *Those candidates with the highest score will be studied to see if they are carriers or not. A new set of patients will be deducted from the patients prior to the existing ones. Using the procedure recursively, we will reach patient zero.*

5. Analysis of algorithm

After the theoretical development of the Inverse Graph method, the general geolocation algorithm has been programmed using a Computer Algebra System (CAS) as MatLab [†]. The developed algorithm has a quadratic execution time, i.e. $\Theta(n^2)$ and its pseudo-code [‡] would be as follows:

```
begin   {Read kml compress files from Google Maps timeline for each
        user $i$. Build matrix $M_i$ of coordinates and its convex hull.
        Analyzes only disjunct concentrations of convex hulls (Figure 8)}
```

[†] Algorithms and functions programmed in MatLab version R2020b and tested on a MacBook with processor 1.3 GHz Intel Core m7

[‡] All that appears between braces are comments

```
    $M = []$; for i:1:users do $M = [M; M_i]$ end for
    Times=intervals(M);     {Build matrix time interval for all users}
    I=sort(Times)     {Sort the intervals as a function of time}
    point=[]; inter=size(I,1);
    for j=2:inter
        for i=1:j-1     {Check the space-time match}
            if fit(Times(i),Times(j))
                point=cuts(I,Times,M);     {Found intersection points}
            end if
        end for
    end for
end
```

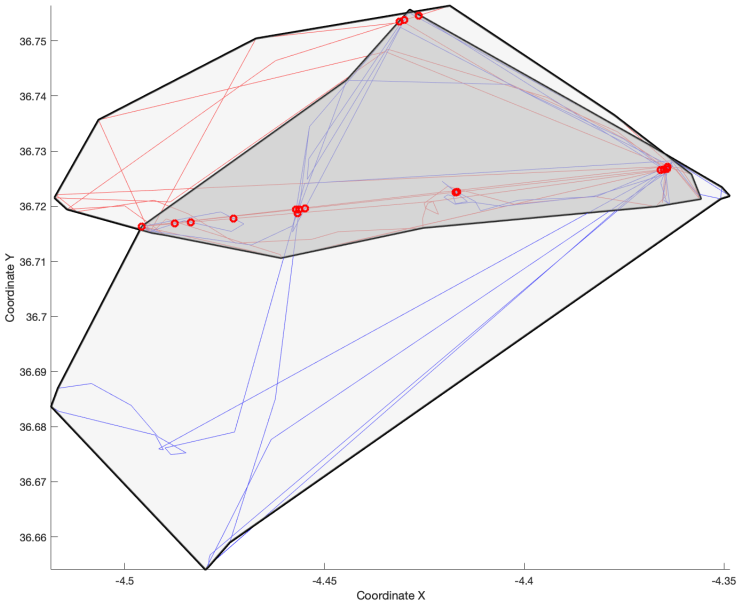

Figure 8. Convex hulls non-disjoint and spatial-temporal intersections.

Next Table 1 and Figures below (Figures 9 and 10) show execution times (in seconds) and the number of contact points that occurred between different people, based on user graph intersections for the last twelve days, with an average of 388 vertexes at their polygonal paths for each user

Table 1. Execution time and intersection points found.

Users	Time	Points
10	0.29	837
20	0.41	1755
30	0.72	2565
40	0.94	3595
50	1.20	4954
100	3.60	10746
150	7.68	15397

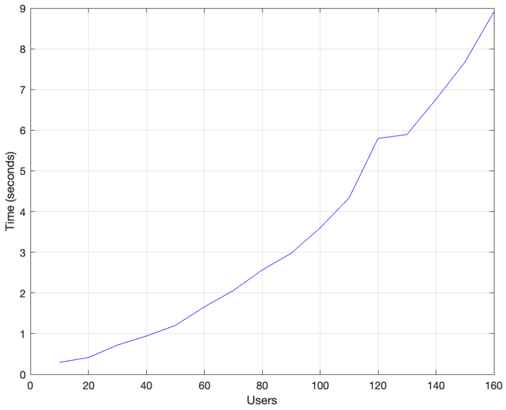

Figure 9. Execution time.

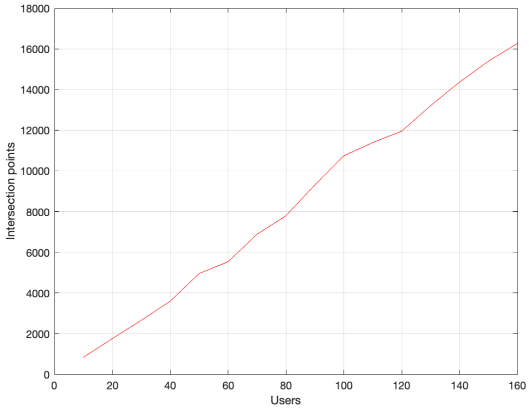

Figure 10. Intersection points.

6. Conclusions

The capacity of Big Data to solve the problem will depend on the reduction that is made on the population to be analyzed for the search of the patient zero [20].

This reduction is conditioned by the need for temporal and geometric coincidence of the routes with those of the probable intervals. Therefore, it is a priority to establish the condition of temporal coincidence, which will considerably reduce the set of candidates (Figure 11), and then analyze the intersections in the plane.

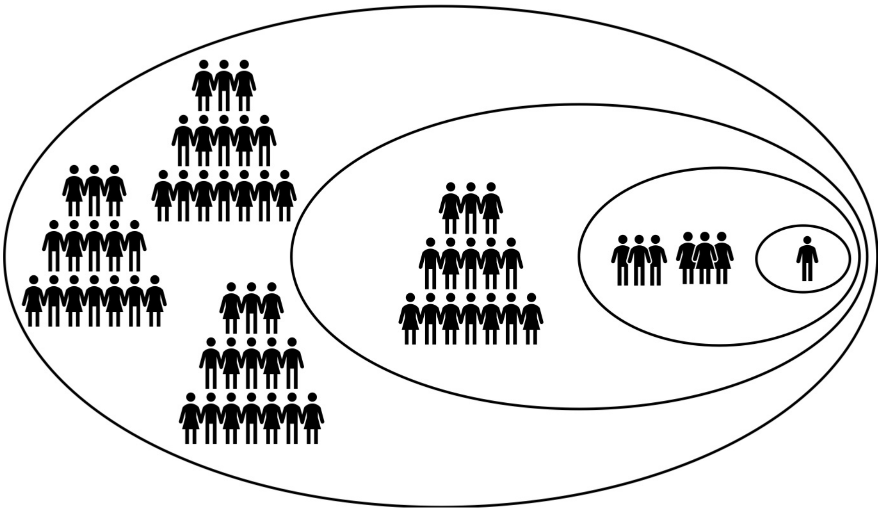

Figure 11. Progressive reduction.

7. Future work

7.1. Matrix representation

Each element a_{ij} of a dispersed matrix of contagions $A = (a_{ij})$ would represent with values 0 or 1 the possibility that the patient P_i has infected the patient P_j, with $i, j \in \{1, \ldots, N\}$

With this representation the rows i, whose sum of terms is greater $\sum_{j=1}^{N} a_{ij}$, will correspond to the index of the most dangerous patients. The number of infections of the most dangerous patient will be indicated by

$$\|A\|_\infty = \max_i \sum_{j=1}^{N} a_{ij}$$

7.2. Markov chains

The application of Markov Chains on arrays, whose elements represent the mesh of an area, will allow us to see to which position each individual moves in a unit of time and to which state of health he can evolve [3, 17].

7.3. Artificial Intelligence

Neural networks can provide a trained system for the pre-selection of possible candidates for asymptomatic or unknown patients [1].

Ackowledgements

We thank the support of this paper from University of Málaga and CBUA (funding for open access charge: Universidad de Málaga / CBUA) and we thank also the anonymous reviewers whose suggestions helped improve and clarify this manuscript.. This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Conflict of interest

The authors declare that there are no conflicts of interest.

References

1. B. Alsolami, R. Mehmood, A. Albeshri, *Hybrid Statistical and Machine Learning Methods for Road Traffic Prediction: A Review and Tutorial*, Smart Infrastructure and Applications. EAI/Springer Innovations in Communication and Computing. Springer. 2020. https://doi.org/10.1007/978-3-030-13705-2_5
2. W. B. Arthur, W. Polak, The evolution of technology within a simple computer model, *Complexity,* **11** (2006), 23–31. https://doi.org/10.1002/cplx.20130
3. V. Ayumi, I. Nurhaida, Prediction using Markov for determining location of human mobility, *J. Inf. Sci. Technol.,* **4** (2020), 2550–5114. https://innove.org/ijist/index.php/ijist/article/view/141
4. J. Bentley, T. Ottmann, Algoritms for reporting and counting geometric intersections, *IEEE T. Comput.,* **C-28** (1979), 643–647. https://doi.org/10.1109/TC.1979.1675432
5. M. R. Benzigar, R. Bhattacharjee, M. Baharfar, G. Liu, Current methods for diagnosis of human coronaviruses: Pros and cons, *Anal. Bioanal. Chem.,* (2020), 1618–2650. https://doi.org/10.1007/s00216-020-03046-0
6. W. V. Bortel, D. Petric, A. I. Justicia, W. Wint, M. Krit, J. Marian, et al., Assessment of the probability of entry of Rift Valley fever virus into the EU through active or passive movement of vectors, *EFSA Supporting Publications,* **17** (2020), 1801–1824. https://doi.org/10.2903/sp.efsa.2020.EN-1801
7. A. A. Brincat, F. Pacifici, S. Martinaglia, F. Mazzola, The Internet of Things for Intelligent Transportation Systems in Real Smart Cities Scenarios, *IEEE 5th World Forum on Internet of Things (WF-IoT),* Limerick, Ireland, 2019, 128–132. https://doi.org/10.1109/WF-IoT.2019.8767247
8. A. A. Brincat, F. Pacifici, F. Mazzola, IoT as a Service for Smart Cities and Nations, *Internet Things Magazine IEEE,* **2** (2019), 28–31. https://doi.org/10.1109/IOTM.2019.1900014
9. J. Cerda, G. Valdivia, John Snow, the cholera epidemic and the foundation of modern epidemiology, *Rev. Chil. Infect.,* **24** (2007), 331–334. https://doi.org/10.4067/s0716-10182007000400014
10. P. Elliott, D. Wartenberg, Spatial epidemiology: Current approaches and future challenges, *Environ. Health. Persp.,* **112** (2004), 998–1006. https://doi:10.1289/ehp.6735
11. C. Huang, Y. Wang, X. Li, L. Ren, J. Zhao, Y. Hu, et al., Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China, *Lancet,* **395** (2020), 497–506. https://doi.org/10.1016/S0140-6736(20)30183-5

12. J. G. Lee, M. Kang, Geospatial big data: Challenges and opportunities, *Big Data Res.,* **2** (2015), 74–81. https://doi.org/10.1016/j.bdr.2015.01.003

13. M. Lin, W. J. Hsu, Mining GPS data for mobility patterns: A survey, *Pervasive Mob. Comput.,* **12** (2014), 1–16. https://doi.org/10.1016/j.pmcj.2013.06.005

14. Y. Lin, N. Lin, Z. Zhao, Mining daily activity chains from Large-Scale mobile phone location data, *Cities,* **109** (2021), 74–81. https://doi.org/10.1016/j.cities.2020.103013

15. R. Minetto, N. Volpato, J. Stolfi, R. Gregori, M. da Silva, An optimal algorithm for 3D triangle mesh slicing, *Computer-Aided Design.,* **92** (2017), 1–10. https://doi.org/10.1016/j.cad.2017.07.001

16. N. Neumann, F. Phillipson, Finding the Intersection Points of Networks, *17th International Conference on Innovations for Community Services, Comm. Com. Inf. Sc.*, **717** (2017), 104–118. https://doi.org/10.1007/978-3-319-60447-3_8

17. S. Salimi, Z. Liu, A. Hammad, Occupancy prediction model for open-plan offices using real-time location system and inhomogeneous Markov chain, *Build Environ.,* **152** (2019), 1–16. https://doi.org/10.1016/j.buildenv.2019.01.052

18. E. M. Shahverdiev, S. Sivaprakasam, K. A. Shore, Inverse anticipating chaos synchronization, *Phys. Rev. E. Stat. Nonlin. Soft Matter. Phys.,* **66** (2002), 172–176. https://doi.org/10.1103/physreve.66.017204

19. Q. Shi, D. Dorling, G. Cao, T. Liu, Changes in population movement make COVID-19 spread differently from SARS, *Soc. Sci. Med.,* **255** (2020), 113036. https://doi.org/10.1016/j.socscimed.2020.113036

20. S. Suma, R. Mehmood, A. Albeshri, Automatic event detection in smart cities using big data analytics, *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST,* **224** (2018), 111–122. http://dx.doi.org/10.1007/978-3-319-94180-6_13

21. X. Yang, T. Xu, P. Jia, H. Xia, L. Guo, K. Ye, Transportation, Germs, Culture: A Dynamic Graph Model of 2019-nCoV Spread, *Preprints,* 2020. http://dx.doi.org/10.20944/preprints202002.0063.v1

22. P. Zhou, X. Yang, X. Wang, B. Hu, L. Zhang, W. Zhang, et al., A pneumonia outbreak associated with a new coronavirus of probable bat origin, *Nature,* **579** (2020), 270–273. https://doi.org/10.1038/s41586-020-2012-7

23. P. Zola, P. Cortez, M. Carpita, Twitter user geolocation using web country noun searches, *Decis. Support Syst.,* **120** (2019), 50–59. https://doi.org/10.1016/j.dss.2019.03.006

24. M. Caceres, R. Grant, Geolocation API, *W3C Recommendation*, 2022. `https://www.w3.org/TR/geolocation/`