POLITECNICO DI TORINO

Corso di Laurea in Physics of Complex Systems

Tesi di Laurea Magistrale

Inverse Problem in Legionella Outbreaks

From direct modeling to inference

Relatore
Prof. Alessandro Pelizzola (PoliTO)

Correlatori:
Dr. José Javier Ramasco (IFISC)
Dr. Riccardo Gallotti (IFISC)

Samuel Salini
matricola: 241954

Anno accademico 2017 – 2018

Summary

Legionnaires' (or Legionella) Disease (LD) is a type of pneumonia that people catch by inhaling small droplets of water suspended in the air containing the Legionella bacterium. There is no evidence of person-to-person transmission. Outbreaks occur from purpose-built water systems where temperatures are warm enough to encourage growth of the bacteria, e.g. in cooling towers and evaporative condensers. Thirty different outbreaks were officially registered in the world between 1976 and 2017 [1]. In total 3178 people were affected by the LD, 236 of whom died. The fatality rate ranged between 0.8% and 75% depending on the outbreak. A LD outbreak happened in Palmanova, a touristic neighborhood part of the municipality of Calvià, Spain [2], between September and October 2017. The people affected were 27: one local worker and 26 tourists. One tourist died. After a long investigation, the main source was identified to be a whirlpool spa (also known as Jacuzzi) on the rooftop of a hotel: the droplets fell down on the surrounding streets and people inhaled them while walking [3].

When an LD outbreak occurs, the health institutions followed the standard epidemiological protocol that consists in asking the infected people about their displacement and then checking one by one the potential sources of the disease. The protocol requires time and money. In addition, it usually forces a temporary closure of the buildings where the water systems are. In the Palmanova case most of them were hotels, obliging the hosts to find another accommodation for their guests and to not accept any new tourist. The consequence is a big economic trouble.

The IFISC[1] institute in Palma de Mallorca was contacted by the Spanish Health Institutions to collaborate and try to find a new approach to the problem. For this reason we build a computational (agent based) model capable of placing the source of the disease into a city, simulating people moving through the network of roads and getting infected when staying close to the source, and computing the epidemic outbreak informations. We repeat the simulation for many realizations and for different positions of the source. This comparison of the obtained data with the real ones allows us to infer the best parameters of the model and finally the source position. This will be obtained by creating a probability heat-map of the region, telling the more probable locations of the source with the help of different coefficients. The model does not intend to be an alternative to the epidemiological protocol, but an extra instrument in the toolbox of the institutions to fasten the process of finding the source and obtain both health and economic benefits for everyone.

[1] *Institute for Cross-Disciplinary Physics and Complex Systems*, the institute where thhis thesis has been developed.

Contents

I Introduction **6**

1 General overview 7
- 1.1 The problem . . . 7
- 1.2 State of the art . . . 7
- 1.3 The aim of the project . . . 8

II Real and collected data **9**

2 Real data for realistic modeling 10
- 2.1 The online survey . . . 10
  - 2.1.1 The questions . . . 11
  - 2.1.2 The collected data . . . 11
  - 2.1.3 Numbers and limitations . . . 11
- 2.2 Other data . . . 11

III The model **15**

3 Building the model 16
- 3.1 A computational overview . . . 16
- 3.2 The structure . . . 17
  - 3.2.1 Early steps . . . 17
  - 3.2.2 Building the agenda . . . 18
  - 3.2.3 Modifying the graph and preparing the system . . . 19
  - 3.2.4 Let the agents walk: infection probabilities . . . 19
  - 3.2.5 Let the agents walk: shortest path . . . 23
- 3.3 The inverse problem . . . 23
  - 3.3.1 χ^2_{size} . . . 25
  - 3.3.2 χ^2_{space} . . . 25
  - 3.3.3 χ^2_{time} . . . 26
  - 3.3.4 $\mathbb{P}_{size}$ . . . 27
  - 3.3.5 Jaccard index . . . 27

IV Results **28**

4 Results 29
4.1 The parameters' choice 29
4.2 Comparing the methods 30
4.3 Analyzing χ^2_{space} and the Jaccard index J 31

V Conclusions & Comments **37**

5 Conclusions 38
5.1 Developed work 38
5.2 Possible future work 38

Part I

Introduction

Chapter 1

General overview

1.1 The problem

In 1976, an outbreak of severe pneumonia among the participants of the American Legion Convention in Philadelphia led to the description of Legionnaires' disease [4]. The disease was found to be caused by the bacterium *Legionella pneumophila* (*Legionella* after the legionnaires who were infected at the convention; *pneumophila* meaning "lung-loving"), belonging to the family Legionellaceae. The generic term "legionellosis" is now used to describe these bacterial infections, which can range from a febrile illness (Pontiac fever) to a rapid and potentially fatal pneumonia (Legionnaires' disease). Legionellosis emerged because of human alteration of the environment, since *Legionella* species are found in aquatic environments, and proliferate in warm water and warm, damp places, such as cooling towers.

When an LD outbreak occurs, the health institutions usually follow an epidemiological protocol [5] that consists in: asking the infected people questions about their mobility (in which hotel, restaurants, museum, beaches, etc. did they spend their time, which roads did they use to move, etc.); from this informations, identifying the water systems that could possibly contain the source of the disease; analyzing sample of these waters in a laboratory. If lucky, the source is in one of the identified water systems and the last step consist in disinfecting it. If not lucky, the last step would probably be extended to all the identified water systems. The protocol can be divided into epidemiological, environmental, laboratory and geospatial investigations and all the process requires time and money.

1.2 State of the art

Papers have been published to understand the different aspects of the LD from an epidemiological point of view: how much the Legionella bacteria are diffused in a geographical region [6] or how their propagation is stimulated in the possible water sources [7]. Hopefully some effort have been done to make *predictions*, on the possible insurgence of outbreaks and also on their diffusion: the involvement of an entire community (and so of a wide geographical area) has pushed towards the study of the effect of the wind on the propagation of the bacteria and thus the aerosol dispersion was modeled [8]; in [9] a probabilistic system for

predicting the risk of Legionella bacteria presence in evaporative installations, exploiting remote information relating to the quality of the water, is discussed; even the exposure to LD while having a shower has been modeled [10].

Another part of the community interested in the LD spreading has focused on the application of *geographic information systems* (GIS) [1] and spatial data to model and predict ongoing outbreaks. The aim is to better understand the spatial relationships between infection cases and the environment where they happen. In [11] four types of spatial data have been identified as being potentially useful to an outbreak response: case data (*i.e.* locations visited in incubation period including their home); potential sources in the locality (i.e. a registry of cooling tower locations and field investigation of other sources); information about the broader demography of the population (*i.e.* how many people live in the administrative regions identified or a control group to compare to cases) and finally meteorological data (*i.e.* wind speed and direction if dispersion modeling is being performed). For what concerns the techniques to use, two broad families of statistical analysis were identified: one uses case data to infer zones for further/higher priority field analysis; the other focuses on known potential sources and checks whether the pattern of infection of cases is consistent with a release emanating from there. The model I here describe falls in between the two categories and in the next section I explain why.

1.3 The aim of the project

The model simulates people (agents) walking on a city road network which could pass near the source of the disease and so they may be infected. To do so it exploits different kind of GIS data: the road network gives the spatial description of the possible paths, while mobile phone tracks allow a realistic description of human displacement. None of them is directly related to infected people and that is why the model does not fall completely in the first category described above. The results of the simulations are infection curves (*i.e.* number of infected agents over time) to be then compared with some real infection curve, checking spatial,outbreak-size and time correlations through the computation of selected coefficients. It is an *inverse problem*: with the help of maximum likelihood techniques we want to infer the most probable location of the disease's source and show the result with colored heat-maps superimposed to the city map. And that is why the model falls also in the second category of statistical techniques described above. All the evaluations will be done onto the town of San Secondo di Pinerolo, a small town of 3000 inhabitants in the countryside nearby Torino, where I am from.

[1] As established in [11]: "*a GIS can be described as the integration of software and hardware for the digital capture, management, analysis and visualization of geographically referenced data. The majority of health data are inherently spatial and have a location, be it an address or a broader administrative unit. GIS enable interpretation of this information spatially, looking for patterns, trends and relationships that might exist between disease (or other occurrences), demography, environment, space and time*".

Part II

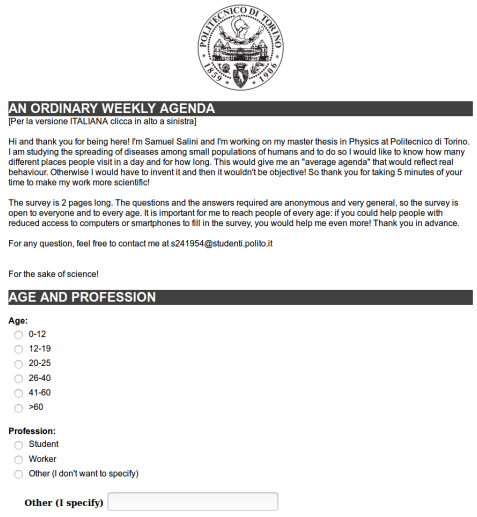
Real and collected data

Chapter 2

Real data for realistic modeling

An important aspect of the model is that it is *data-driven*. For this reason we need real data to understand the typical number of displacements a person undergoes in its daily life, the typical distances of these displacements (and so the typical travel times) and also the most common trajectories a person goes along when moving. All these informations does not have to be strictly related to people who got really infected. The idea is to model the entire population of a city (or a sample of it) commuting on a road network. The infection aspect is then introduced with some parameters. The kind of data that can be used is then really broad.

2.1 The online survey


AN ORDINARY WEEKLY AGENDA

[Per la versione ITALIANA clicca in alto a sinistra]

Hi and thank you for being here! I'm Samuel Salini and I'm working on my master thesis in Physics at Politecnico di Torino. I am studying the spreading of diseases among small populations of humans and to do so I would like to know how many different places people visit in a day and for how long. This would give me an "average agenda" that would reflect real behaviour. Otherwise I would have to invent it and then it wouldn't be objective! So thank you for taking 5 minutes of your time to make my work more scientific!

The survey is 2 pages long. The questions and the answers required are anonymous and very general, so the survey is open to everyone and to every age. It is important for me to reach people of every age: if you could help people with reduced access to computers or smartphones to fill in the survey, you would help me even more! Thank you in advance.

For any question, feel free to contact me at s241954@studenti.polito.it

For the sake of science!

AGE AND PROFESSION

Age:
- 0-12
- 12-19
- 20-25
- 26-40
- 41-60
- >60

Profession:
- Student
- Worker
- Other (I don't want to specify)

Other (I specify)

Figure 2.1: Screenshot of the survey.

The survey is an idea I had to collect real data about the "daily life" quantities described above. Without these data, either the informations will be invented (but then the results will lack a real and physical meaning), or they will be extracted from some probability distribution, as introduced in section 3.2.2. The survey can surely be improved and has to be intended as a different and potentially interesting starting point for further studies. It can still be visualized and compiled following the link in [12].

The idea did not come completely by chance: I was inspired by the way epidemiologists collect data of infected people, that is... using surveys. It is important to underline the big difference between the two methods, though: epidemiologists work *a posteriori*, asking to people already infected and ill where they used to spend their previous days and so on (an example can be found in [13]); in this thesis, the survey is done *a priori*, in the sense that no one is infected and there is

no infection and anyone can answer the questions. The aim is to collect as many answers as possible to build a probability distribution function of the number of stays of an agent and another probability distribution function of the stay-times (i.e. Δt). In both cases the final step is to infer the source position of the Legionnaire's disease.

2.1.1 The questions

The questions were thought to be in the meantime as useful for the research and as privacy-respectful as possible for the people compiling it. For example, the city of residency is not a necessary information, so the question was avoided. On the contrary, Legionella disease varies its strength with the age of the person, so the age was asked. For more details on all the questions, see table 2.1.

2.1.2 The collected data

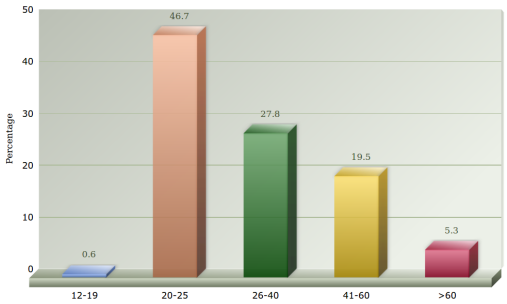
Here I simply report the histograms of the answers received by people that compiled the survey. The original plan was to analyze those histograms and obtain some probability distributions to be used in creating the agendas. In the end not much of the collected data is really useful: only the answer related to the number of placed visited in a day was used (see 2.2, question 5): almost all the answerers of the survey focus in a number of activities per day that ranges between one and five.

2.1.3 Numbers and limitations

169 people filled in the survey. A major problem comes from the high percentage of answerers being young people: under 40s cover more than 70% of the answers while the most probable target of the legionella disease are elder people.

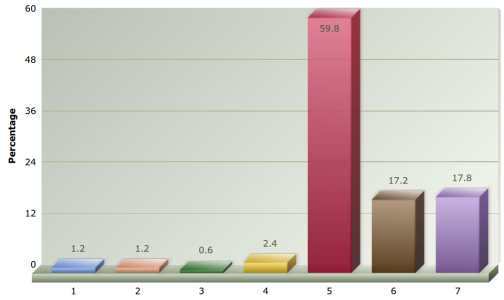
2.2 Other data

In order to obtain precise results it would be better to know the typical human trajectories in a given city, *e.g.* which roads are more often used. This would allow to know the typical distances walked by a person between two stay-points and thus it would be possible to build more precise agendas. The data we are talking about consist of mobile phones geo-localized tracks (from calls and app usage), which are really high-resolution, as well as census data, which instead is low-resolution but easy to obtain and without privacy issues. The usage of these data represent one of the next steps of this thesis: as usual we need to start with a basic model which can be later improved with more details about the real world.

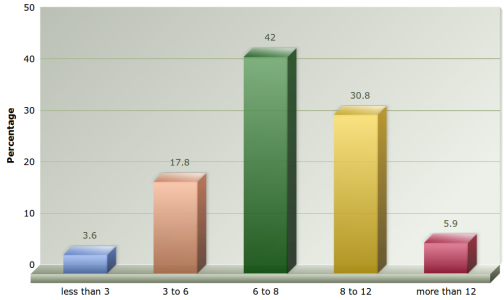


(a) Question 1

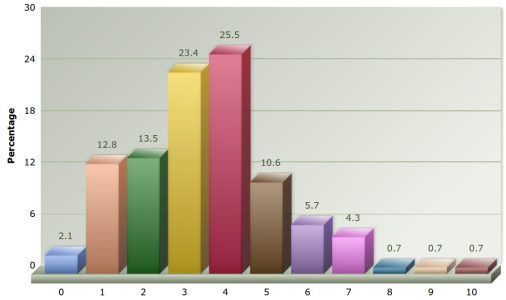
(b) Question 2



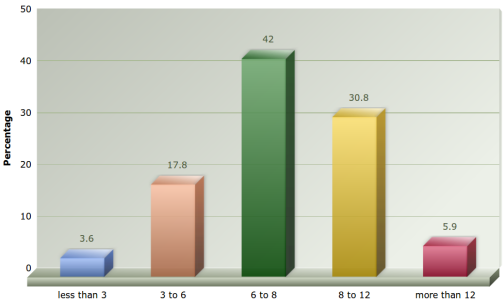
(c) Question 3



(d) Question 4

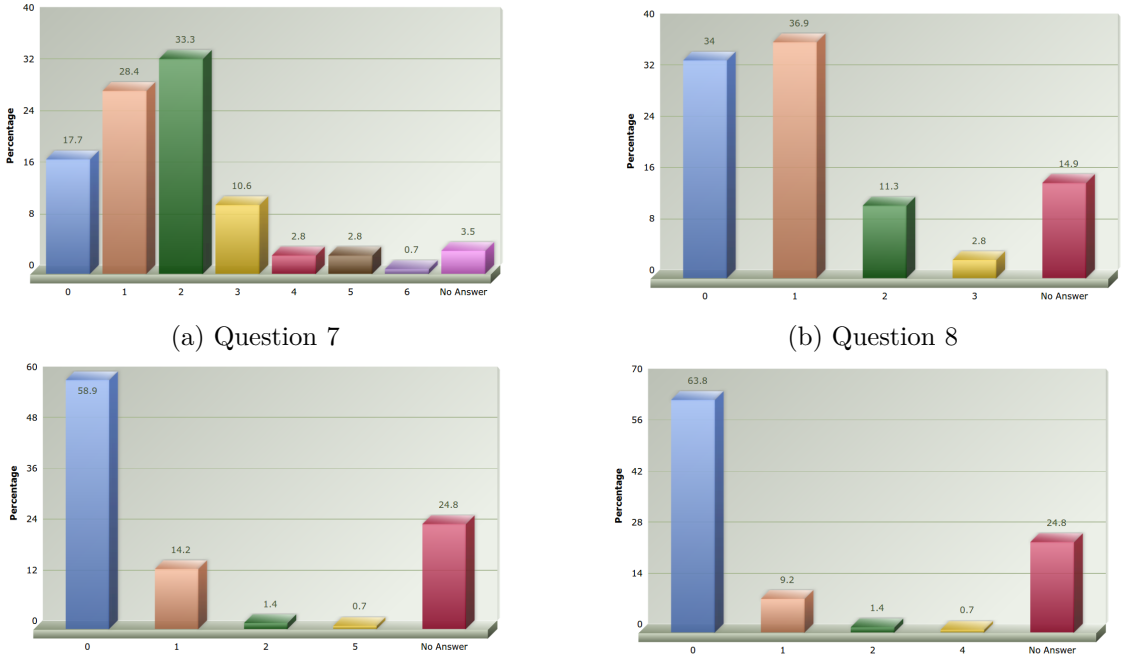


(e) Question 5



(f) Question 6

Figure 2.2: Histograms of received answers for questions from 1 to 6 of the survey. They are expressed in percentage.



(a) Question 7

(b) Question 8

(c) Question 9

(d) Question 10

Figure 2.3: Histograms of received answers for questions from 7 to 10 of the survey. They are expressed in percentage. In (b), (c) and (d) the column *No answer* is equal to the column *0*.

#	Question	Possible answers or answers given									
1	age	12-19	20-25	26-40	41-60	>60					
2	Profession	Student	Worker	Other*							
3	How many days per week do you spend doing your profession?	1	2	3	4	5	6	7			
4	How many hours per day do you spend doing your profession?	<3	3 to 6	6 to 8	8 to 12	>12					
5	How many activities did you select?**	0	1	2	3	4	5	6	7	8	9 10
6	Number of activities that last less than 1 hour per day	0	1	2	3	4	5	6			
7	Number of activities that last 1 to 2 hours per day	0	1	2	3	4	5	6^			
8	Number of activities that last 2 to 3 hours per day	0	1	2	3	4	5	6			
9	Number of activites that last 3 to 4 hours per day	0	1	2	3	4	5	6			
10	Number of activities that last more than 4 hours per day	0	1	2	3	4	5	6			

Table 2.1: Questions asked in the survey. *It was possible for people the specify what they do or not, to respect the privacy.
**In order to understand this and all the successive questions, here is reported the text preceding the question in the survey: *Think of all the other activities you do during your weekly routine. Restrict the selection to those that last more than 30 minutes, that you don't do neither at home nor in your study/work/other-place and that are "localized" in space. Some examples are sport activities (in a park, a football pitch, in a gym are ok, while running, riding a bike or sailing are not of interest), having lunch in the park, going to the supermarket, restaurant, cinema, hairdresser (only if you go there every week).....Now that you have a list, keep a maximum of 10, those that are more important to you. They can be less than 10 of course. How many of them last less than an hour PER DAY? How many take 1 to 2 hours per day? And 2 to 3 per day? And 3 to 4 per day? And more than 4 hours per day?.*
^6 is the maximum answer given by people in the survey.

Part III

The model

Chapter 3

Building the model

3.1 A computational overview

As already said, the model is data-driven and based on real GIS data of road networks. The latter are downloaded from the free and open source Open Street Maps (OSM, [14]) database. The data collected are really complete and include, as examples, the name of the roads, their length, if they are highways or pedestrian streets.... Since the aim of this work is to simulate agents *walking*, only pedestrian roads are used. This is possible through the use of a new python library, called OSMNX [15]. The name is the combination of *OSM* + *NX*, since the library was built exactly to deal with GIS data and treat them as network objects, with the support of the well known NetworkX library (NX, [16]).

The model works as follows: first it prepares the network of roads on which the agents will move, creating and simplifying a spatial graph. Then it places the source of the illness somewhere on the town and associate to it the characteristic lengths r_0 and r_{action}, making possible to label as *dangerous* all the nodes and roads inside the circle of radius r_{action} (both the lengths will be discussed later). Afterwards the steps related to one agent are executed: the model creates an *agenda* consisting of consecutive *stay-points* together with an amount of time each (the *stay-time*), where the agent spends its day. Now the stay-points are added to the graph, linked to the existing roads and labeled as dangerous, using th same procedure as for the road nodes and edges. Lastly the simulation part begins: the agent start to "live" and walk from the *home* stay-point to the successive ones according to its agenda and following the shortest paths connecting them. Of course one constraint is to make the agent go back home at the end of the day. If it crosses the dangerous area, the model starts computing the probability of infection and, if necessary, labels the agent as *infected*. In this case the simulation for that agent ends. The infection simulation is repeated for many runs, in order to obtain good statistics. The outcome of these computations allows to plot infection curves (i.e. histograms of the number of infected agent with respect to time) used to tune the parameters of the model, and then nice heatmap graphs (used to solve the inverse problem). What is interesting then is to move the source of the disease in different places for the simulations, and see if the number of infected of the simulation reflects the dimension of the real outbreak. In this work the real outbreak data is not taken from the real world, but it is made with one direct simulation: no iterations are used, just one run and. So we will move the source position

in the simulation and then compare the results with the direct simulation's data.

All the process can be divided into three blocks that differs one from the others for the number of times they are called in one simulation:

- ***una-tantum***: it consists of all the computations to be done only once in order to prepare the system to host the agents' displacements.

- ***una-simulatio***: it is the block to be run once for every simulation (*i.e.* every time the source of the disease is moved) and characterizes the nodes and edges of the network which are affected by the disease.

- ***simulatio***: it is the only iterated block and makes the simulation happen: agents move onto the network according to their agendas and record time and place of infection.

Routine	What it does
AGENDA	it creates the list of stays-points and time of stays for an agent in a day.
MODIFICATIONS	given the agenda, it modifies the network of streets so as to include the stays and reduce the number of nodes.
DANGER	given the position of a source and its radius of action, it identifies the edges and nodes affected by the disease.
DIJKSTRA	computes shortest-paths with the bidirectional implementation.
INFECTION	it computes probabilities of infection if an agent goes along dangerous paths.

Table 3.1: Short description of the routines used by the model. Each of them has a role in one of the three code blocks.

When the simulation are completed, the results are analyzed and treated to solve the inverse problem.

3.2 The structure

3.2.1 Early steps

In order to introduce how the routines reported in table 3.1 work, an explanation of the very first steps executed to prepare the system is needed. The result of downloading OSM data of a city road's network via the OSMNX functions is a graph, which includes edges/roads and many nodes, that can be either intersections or interstitial nodes [1]. A curve is made

[1] An interstitial node is defined as a node having degree equal to 2 (when a simple road) or 4 (when the road has two directions and to each direction is associated a different edge of the network, so two

of a lot of interstitial nodes that are useless since we are interested in computing shortest paths in the end. Another OSMNX function helps us reducing the complexity of the graph by "eliminating" the interstitial nodes (the nodes are removed, but their coordinates are kept so as to still have a correct spatial description of the roads, i.e. their curvature and length). The complexity of the graph will be again enhanced when adding the agenda's stay-points as nodes of the graph. All this paragraph is depicted by the three images in figure 3.1.

(a) (b) (c)

Figure 3.1: Variation of the complexity of the graph applied to the small village of San Secondo di Pinerolo (ITALY): (a) shows how the downloaded network appears, with a lot of interstitial nodes along the roads; (b) instead shows how easily the same graph can be described when using only the intersection nodes, while keeping the geometry of the roads; finally (c) is an example of how the complexity of the graph rises up when the *home* stay-points (first stay-point of one agent's agenda) are added all at the same time. Here 700 agents' homes are shown. So the number of added nodes is 700 plus eventually some interstitial nodes, if they correspond to the nearest node of some home point P.

3.2.2 Building the agenda

The purpose here is to create a list of *stay-points* [2] and *stay-times* Δt for each agent. How can one decide the stay-points for an agent? Much freedom is left here: one could extract them according to some distribution of the distances between stay-points built on real human mobility data (see [17]), or from some distribution that arise from online surveys (see section 2.1), or directly interpolating real human trajectories, or yet one could simply extract them uniformly from the region of the town, for an easy starting point. Another aspect concerning this routine is the freedom left on the $\Delta t's$, on the number of stay-points

going *in* and other two going *out* of the node). Roughly speaking, you can imagine it as a node staying in the middle of just one road, useful only to define the shape of the road and that is not an intersection of different roads

[2] We define a stay-point as a point on the map where the agent spends a certain amount of time Δt. When modifying the network it will become a node identified by a negative label, while road nodes have a positive label called *osmid* inherited by the official OSM metadata.

per agent and also on the number of days an agenda should cover. Of course the three "parameters" are related to each other and can be extracted from probability distributions (again built from real data in different ways described above) or arbitrarily chosen.

The model here uses a combination of these possibilities: the agenda is about a one day routine for each agent; the number of stay-points is extracted from a Poissonian distribution of mean 3.5 (as suggested in [18]); since we lack the informations about the average traveled distance by one person walking, the position of the stay-points are chosen randomly inside a circle of radius $d = 1000\ m$ centered in the first stay-point, *i.e.* one agent's home (its position is really random though). Random because we do not have much choice, while the distance limit d is used to avoid having really long distances to walk from one stay-point to another; also the maximum number of stay-points possible is limited, this time to be between one (staying only at home) and 5. This information comes from the histograms of section 2.1; the stay-times are eight hours for the home-stay and random for the other stay-points.

3.2.3 Modifying the graph and preparing the system

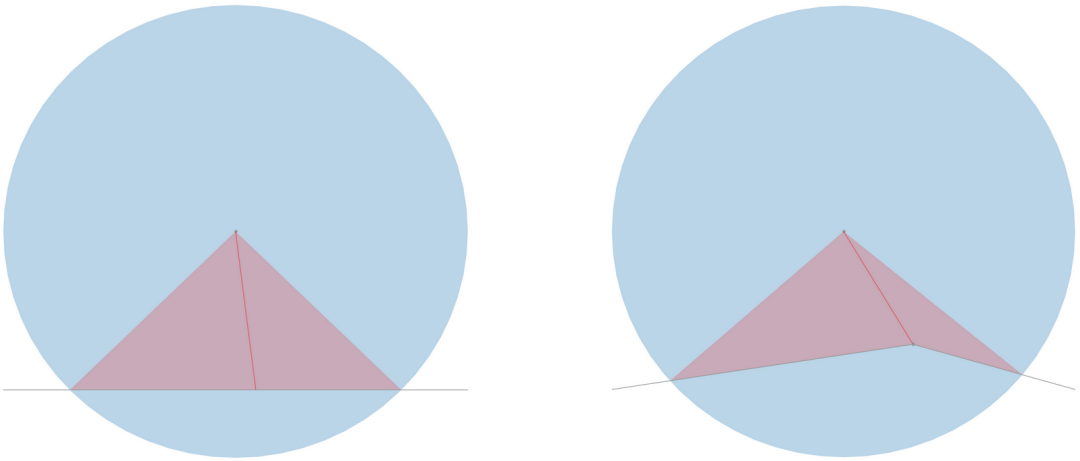
In order to add the stay-point P into the network of roads to be used in the simulations, the model finds the nearest node to P in the simplified network and creates a link between them. If the nearest node corresponds to an interstitial node of the original network, it splits the edge geometry (*i.e.* the spatial curve of the road) of the simplified network into 2, putting the nearest node in between. A possible outcome is shown in figure 3.1c.

3.2.4 Let the agents walk: infection probabilities

The Legionnaire's disease spreads through bacteria that live in warm waters or in areas with high levels of humidity and temperature. Micro-droplets of water bring the bacteria and people could inhale them, having a chance of getting infected. In principle the infection probability depends on a lot of parameters such as the temperature of the environment, the temperature of the water, the concentration of bacteria, the wind, the size of the droplets, etc. In practice, we cannot take all of them into account to model the phenomenon. So we do an approximation: we consider the water droplets as random-walkers with a life time τ due to the fact that the droplets slowly evaporate until they disappear [3]. We need to know the spatial probability distribution $P_t(r)$ of finding one droplet that was generated a time t ago at a distance r from the source. Then, in order to obtain a spatial density distribution of droplets to be used to define the infection rate β, we need to integrate over all times t between 0 and τ. The starting point of the analysis is the assumption that water droplets undergo an isotropic Brownian diffusion process limited in time, so the first distribution is the Rayleigh function

$$P_t(r) = \frac{r}{\sigma^2(t)}\, e^{-r^2\,/\,2\sigma^2(t)}, \tag{3.1}$$

[3] One can consider the droplet as a 2D random walker with an absorbing wall that appears after a certain time τ in the exact position of the random walker at that time, regardless of its distance from the center.



(a) (b)

Figure 3.2: Depicting the two possible situations an agent can encounter while crossing the infectious region (here the blue circle). It is clear that the distance r from the source location to the agent's position on the gray path varies along the path itself, *i.e.* the red segment varies its length in time (the red segment is the distance r between the source position, centered in the blue circle, and the position of the agent crossing the dangerous area while moving on the thin, gray path. The red shape must be seen as a simultaneous representation of all the possible values r can take from the moment the agent enters the dangerous area to the moment it exits it). (a): the agent crosses the region without stopping. The path can be divided in many very small segments δx, all of the same length, along which r can be considered as constant and so a unique value of β is used. The resulting probability is given by the equation 3.6; (b) the agent goes along the first edge, stops for a time Δt in the stay-point inside the circle and then goes away along the edge in the other direction. In this case the probability of infection is given by the equation 3.9.

where the variance is

$$\sigma^2(t) \doteq 4\,D\,t. \tag{3.2}$$

Here D is the diffusion coefficient. What we want now is the probability of finding a droplet at distance r in any moment of its life. This is given by the time integral of the Rayleigh function:

$$\begin{aligned} \rho(r) &= \int_0^\tau dt\, \frac{r}{4\,D\,t}\, e^{-r^2\,/\,8\,D\,t} \\ &= \frac{2\,r}{{r_0}^2}\,\Gamma\left(0\,,\,\frac{r^2}{{r_0}^2}\right), \end{aligned} \tag{3.3}$$

where

$${r_0}^2 \doteq 4\,D\,\tau, \quad \Gamma(\alpha\,,\,z) \doteq \int_z^\infty y^{\alpha-1}\,e^{-y}\,dy. \tag{3.4}$$

Here r_0 is introduced since we do not want to deal directly with D, while the second equation is the definition of *incomplete Gamma function*.

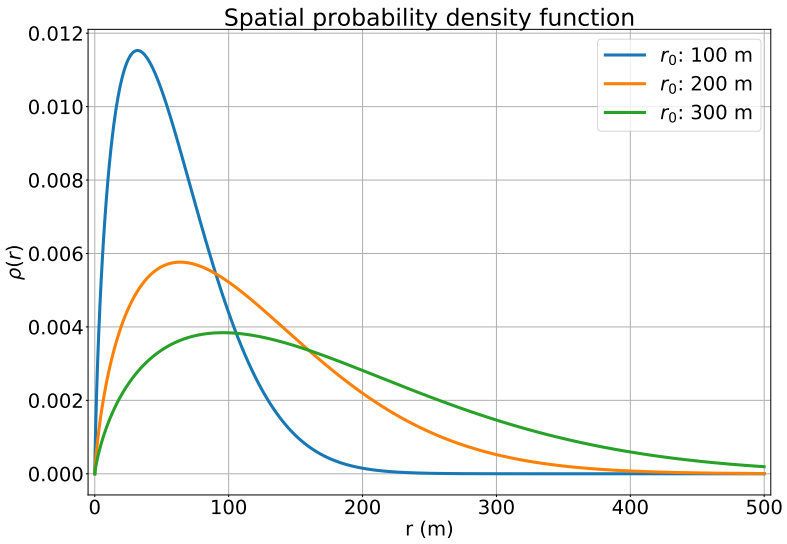


Figure 3.3: Graphical view of $\rho(r)$, derived in equation 3.3, with three different values of r_0. It is 0 at $r = 0$, has a maximum at $r \simeq \frac{r_0}{3}$ and then decreases to 0 quite fast: $\rho(r = 3\,r_0) \sim 10^{-3}$, making the infection rate become really small.

After all these computations it is clear why it makes sense to put the LD source on the map and associate to it a circular region of action. While "living" on the road network an agent could pass through this circle and while being inside it could be infected. There are two possible situations for which this happens and we need to mathematically treat them separately. Figure 3.2 shows them graphically. In one case the agent goes through the circle of action while going from one stay-point to the next one, so without stopping inside. Given an infection rate

$$\beta(r) \approx \beta_0\,\rho(r), \tag{3.5}$$

where β_0 is an effective parameter including the influence of parameters (temperature, etc.) on the infectivity, the probability of infection in a unit of time dt is $\beta\,dt$. The model wants to be as realistic as possible, so we cannot deal with infinitesimal quantities. What we can do is to take the intersection between the area of action of the LD and the edge of the agent's path, split it into arbitrary small segments all of length δx, so the time needed to cross them is always δt small enough, and the distance to the source along the segment can be considered constant along all the segment, then compute the probability of infection as one minus the probability of not being infected during the walk. In other words:

$$\begin{aligned} P_1 &\doteq P(agent\ infected\ in\ intersection) \\ &= 1 - P(agent\ not\ infected\ in\ all\ intersection) \\ &= 1 - \prod_{i=1}^{n} P(agent\ not\ infected\ in\ segment\ i) \\ &\doteq 1 - \prod_{i=1}^{n} q_i. \end{aligned} \tag{3.6}$$

But we know q_i. It is

$$\begin{aligned} q_i &= 1 - \beta_i \, \delta t \\ &\approx e^{-\beta_i \cdot \delta t}, \end{aligned} \tag{3.7}$$

where the exponential approximation is possible since we know we can make δt really small and β_i is constant along the segment i. Thus going back to equation 3.6, the product becomes a sum in the exponent and so

$$P_1 = 1 - exp\left(-\delta t \sum_{i=1}^{n} \beta_i\right), \qquad \beta_i = \beta_0 \, \rho(r_i). \tag{3.8}$$

In the other case one stay point is inside the region of action of the LD, so for sure to reach it one part of the incoming edge is also inside the circle and the same is true for the outgoing edge, which will be crossed when moving towards the next stay-point. There are always one incoming and one outgoing edges since we force the agents to go back home, meaning that the complete path of a day is a closed loop. For this situation the probability of infection changes, but the reasoning is similar:

$$\begin{aligned} P_2 &\doteq P(agent\ infected\ in\ intersections\ or\ node) \\ &= 1 - P(agent\ not\ infected\ in\ intersections\ nor\ node) \\ &= 1 - \Big[P(not\ inf.\ in\ incoming\ edge)\,\mathrm{x} \\ &\qquad \mathrm{x}\, P(not\ inf.\ in\ outgoing\ edge)\,\mathrm{x} \\ &\qquad \mathrm{x}\, P(not\ inf.\ in\ node)\Big] \\ &= 1 - exp\left(-\delta t \sum_{i=1}^{n} \beta_i\right) exp\left(-\delta t \sum_{i=1}^{m} \beta_i\right) exp\left(-\Delta t \, \beta_{node}\right) \\ &= 1 - exp\left(-\Delta t \, \beta_{node} - \delta t \sum_{i=1}^{n+m} \beta_i\right). \end{aligned} \tag{3.9}$$

The index i goes up to n for the incoming edge and up to m for the outgoing edge. Since the δt is the same along all the path and since the two exponentials are multiplied, we can directly sum the corresponding exponents together; the last exponential comes from the node stay. Since the agent is not moving here, β_{node} is a constant and we can see the stay-time Δt as a sum of many δt-contributions. Now it is clear that the process can be iterated to all possible configurations of stay-points inside the circle and edges going inside and outside. What is important is to sum the resulting exponents of consecutive objects, so as to obtain one single probability value.

We can make another approximation due to a technical detail: since the infection rate defined in equation 3.5 decreases quite fast in r (see figure 3.3), we can say that people moving through edges and nodes far from the source can not be affected by the disease, because the probability of infection is almost zero. Indeed it is not necessary to associate a probability to *all* the edges and nodes of the graph. For this reason, when putting the source of the disease somewhere onto the map we associate to it the *radius of action* r_{action},

a "technical" length bigger than r_0 introduced to facilitate the simulation part of the work: if an edge/node falls into the circle of radius equal to r_{action}, it will be labeled as *dangerous* and the probability of infection will be computed. A graphical explanation is given by figure 3.4.

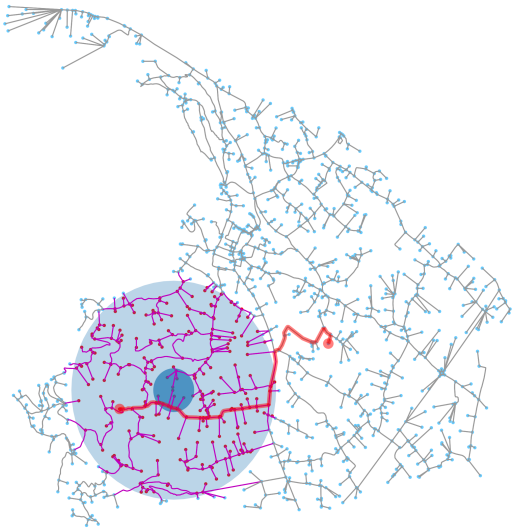


Figure 3.4: The effect of the *danger routine* and a (red) shortest path between two nodes. The dark blue circle is centered in the source of the disease and has a radius equal to r_0; the light blue circle has a radius equal to r_{action} and all the edges and nodes falling into it are magenta, meaning they are labeled as *dangerous*. Obviously the *radius of action* is larger than r_0.

3.2.5 Let the agents walk: shortest path

The Dijkstra algorithm is used to compute shortest paths between couples of nodes (called *origin* and *target*) in the weighted graph (weighted with the edges' lengths). An example is shown in figure 3.4. It is known to be the most efficient algorithm for this purpose and to make it even more efficient, we use the *bidirectional* implementation: in practice bidirectional Dijkstra is much more than twice as fast as ordinary Dijkstra. The latter expands nodes in a sphere-like manner from the origin. The radius of this sphere will eventually be the length of the shortest path. Bidirectional Dijkstra instead will expand nodes from both the origin and the target, making two spheres of half this radius. Volume of the first sphere is πr^2 while the others are $2\pi(\frac{r}{2})^2$, making up half the volume. This may not be true for a general graph, but for sure it is correct when dealing with road networks, as shown in [19].

3.3 The inverse problem

The results of the simulations are condensed in lists, one for each run (given a source position, and a set of parameters). Each of these lists contains the information regarding the agent's health and their "life time": if an agent got infected, he will be labeled as $+1$ and the time associated will be the time he spent moving around before getting the Legionella disease; on the contrary, the label will be -1 if it managed to "live" the entire simulation time without getting infected. The same is done for the direct simulation (*i.e.*

what corresponds to the real outbreak). The data are used first to tune the parameter values, so to find their best values among many trials. Then we try to solve the inverse problem by comparing the simulation data with the direct data. We do this using five different approaches here below described. The results' section shows how good they are in finding the real source position.

All the approaches compare some quantity related to 1000 runs of the simulation for a given source position, and return a value from that comparison. This is done for many different source positions. In fact, we divide the map with a squared grid and place one source in the centre of each square that overlaps with a part of the graph (see figure 3.5). The external area is not taken into account for the simulations. We chose the number of squares based on two informations. First of all the computational time: the smaller the size, the higher the number of squares necessary to cover the graph, the higher the time needed to compute everything; second, the resolution: it does not make sense to use too small squares, since they would contain few homes and the comparison would result in nothing really clear. Furthermore, we mentioned the possibility to use geo-localized data to improve the model, data from mobile phones' GPS. These data are provided by private corporations and for privacy reasons they give them in an aggregate form, *i.e.* they provide average values on a grid like ours, to avoid the possibility of recognizing the single person from his mobile phone's data. Their grid's squares are normally of length around 500 meters. So summing up we used a grid with squares of length 398 meters. Given the map of San Secondo, this results in 156 squares to cover the image, only 105 of which overlap the graph. From now on we talk about squares and source positions treating them as the same thing. In order to obtain realistic results, we can not forget that even if we divide the map into discrete and contiguous squares, the real space is continuous. We keep this information using a radius of action r_{action} equal to the diagonal of the small squares, long enough to partially overlap the surrounding squares. In this way agents living near the square edges will be taken into account also by the neighboring squares computations. We have to remember why this problem is interesting: it is related to an epidemiological problem in the real world. Our aim is to stay as close to the real situation as possible, so the information we want to use for our analysis must be limited, although from a numerical point of view we know many different aspects of the outbreaks: from the simulation outcomes we are able to retrieve the exact moment and the exact place our agents get infected. This does not happen in the real world: the only certain information is were people live, *i.e.* the home-stays. For this reason we try to solve the inverse problem using the same piece of information. So all the plots now on will represent only the home-stays of the agents, without really caring about their movements and other stay-points, and the same applies to the computations.

Three of the following methods are called *Chi-squared* χ^2 simply to indicate that they compute differences between two sets of values and the goal is to find the where is the minimum. The other instead of the differences they try to compute how similar the two sets are, in two different ways. Here what is interesting is the maximum over the outcomes.

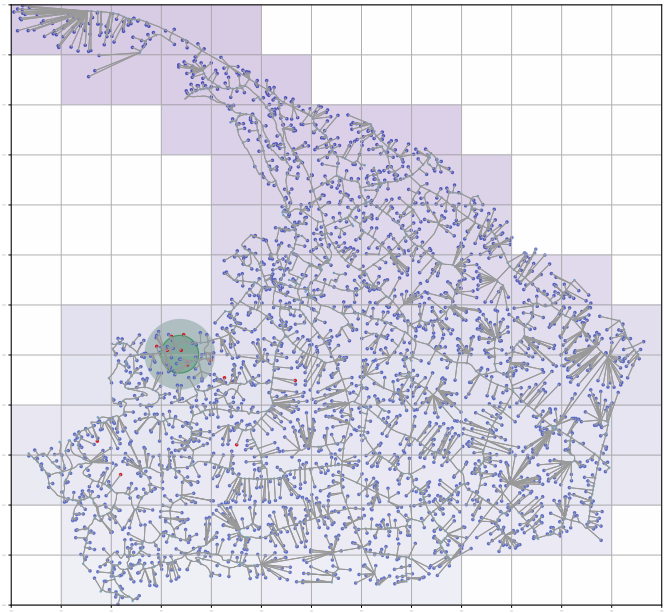


Figure 3.5: Graphical view of the grid used on the graph of San Secondo to compute the different values used in the inverse problem. The colored squares overlap with the map, so they are taken into account for the simulations. The blue nodes represent the home-stays of the 3000 agents living in the town. Some of the are red, indicating that in that simulation the agents living there got infected by the LD that is centered in the green circle.

3.3.1 χ^2_{size}

This is a first, easy attempt to compare the simulation with the real epidemic situation. It is based on the size s of the outbreak, defined as the number of infected in one simulation. For a given source/square at position $\vec{x}$, we compute

$$\chi^2_{size}(\vec{x}) = \frac{1}{N} \sum_{i=1}^{N} \left(s_{ref} - s_i(\vec{x})\right)^2 , \tag{3.10}$$

where N is the number of repeated runs, s_{ref} is the size of the outbreak happened in the real/direct case (ref stands for *reference*), while $s_i(\vec{x})$ is the outbreak's size in the i^{th} iteration, given the position $\vec{x}$. Each square has a correspondent $\chi^2(\vec{x})$ value, thus a heatmap can be drawn. We do not expect it to be really precise, since usually only the size of an outbreak is not enough information for finding the spatial location of the real source.

3.3.2 χ^2_{space}

One natural evolution of the above χ^2-coefficient consists of doing the same thing, but with a higher resolution. We compute a similar coefficient but at the the single squares level: given a position of the source $\vec{x}$ we sum the differences in outbreak-size of the single

squares as

$$\begin{aligned}\chi^2_{space}(\vec{x}) &= \frac{1}{N\,N_{sq}} \sum_{i=1}^{N} \chi^2_i(\vec{x}) \\ &= \frac{1}{N\,N_{sq}} \sum_{i=1}^{N} \sum_{j=1}^{N_{sq}} \left(s_{ref,j} - s_{i,j}(\vec{x})\right)^2 ,\end{aligned} \tag{3.11}$$

where again N is the number of runs, N_{sq} is the total number of squares covering the map[4] and so the index j on the two s indicates we are looking at the size of the outbreak in one particular square. We expect this one to be a good coefficient, since it takes into account the spatial information of the outbreak size. It is reasonable that the dimension of the outbreak is related to the spatial position of the houses.

3.3.3 χ^2_{time}

Also time-evolution similarity could be an important indicator to find the real source position. We want to compare the time evolution of the outbreaks in reality and simulations, so we compute the cumulative outbreak-size curves and compute a χ^2 on their points. Given a time interval Δt, we build the outbreak's histogram, with bins' values $n_1, n_2, ..., n_k$ (number of infected in consecutive time intervals of Δt) and sum them one with another to build the cumulative curve:

$$\begin{aligned} y_1 &= n_1, \\ y_2 &= y_1 + n_2, \\ &\cdot \\ &\cdot \\ y_k &= y_{k-1} + n_k. \end{aligned}$$

Now we compute the coefficient for a given square in $\vec{x}$ as

$$\begin{aligned}\chi^2_{time}(\vec{x}) &= \frac{1}{N\,|\vec{y}|} \sum_{i=1}^{N} \chi^2_{t,i}(\vec{x}) \\ &= \frac{1}{N\,|\vec{y}|} \sum_{i=1}^{N} \sum_{j=1}^{|\vec{y}|} \left(y_{ref,j}(\vec{x}) - y_{i,j}(\vec{x})\right)^2 ,\end{aligned} \tag{3.12}$$

where $|\vec{y}|$ is the length of the vector $|\vec{y}|$, that is the number of points of the cumulative curve. This is the most unknown coefficient *a priori*, since we do not know what will be the results. With $\chi^2_t(\vec{x})$ we finish the χ^2-like coefficients. Remember that for these what we will be interested in is the minimum value, *i.e.* the square with the minimum value.

[4] One could argue that covering a bigger region and thus using more squares would increase the denominator without having any affect on the outbreak, since the effect of the source is kept confined by r_0. We could use the number of squares containing at least one infected agent's home to be more correct. This number would change only varying the square's dimension, but in this case also the simulation outcomes would be different. In this thesis there is no difference since the region covered and the square's size are fixed.

3.3.4 $\mathbb{P}_{size}$

Another simple indicator is the probability of having an outbreak size, given its source position, computed as the fraction of runs with an outbreak size s equal to s_{ref} with a tolerance $0 \leq \alpha \geq 1$. So the counted runs are those with

$$s_i(\vec{x}) \in s_{ref} \left[1-\alpha, 1+\alpha\right].$$

So we call $\mathbb{P}$ the number of counted runs over the total number of runs:

$$\mathbb{P}_{size}(\vec{x}) = \frac{\sum_{i=1}^{N} \delta^* \left(s_{ref}\,,\, s_i(\vec{x})\right)}{N}, \tag{3.13}$$

where the $*$ symbol indicates that the Kronecker's delta δ^* is equal to one every time the above condition is true and not only when $s_i(\vec{x})$ is strictly equal to s_{ref}. As for $\chi^2_{size}(\vec{x})$, we do not expect this coefficient to be really good, since one outbreak size can be obtained by having the source in different places too easily.

3.3.5 Jaccard index

The last method uses the Jaccard index [20] $J(\vec{x})$ to measure how similar the simulation and the real outcomes are. The coefficient, named after his developer the botanist Paul Jaccard, is also known as *Intersection over Union coefficient* and measures how close two finite sample sets are. It is defined as the size of the intersection divided by the size of the union of the sample sets:

$$J\left(A\,,\,B\right) = \frac{|A \cap B|}{|A \cup B|}.$$

In the LD model, for a given source position $\vec{x}$, one sample set is defined as the ensemble of squares where at least one infected agent live. So again it is a way of taking into account the spatial correlations between cases and source position. What we compare is then the list of squares with at least one infected agent's home resulting from one run with the same list resulting from the real outbreak. Call $A = \left\{S_{ref,j}\right\}_{\,j\,\in\,\{1,\ldots,N_{sq}\}}$ and $B = \left\{S_j(\vec{x})\right\}_{\,j\,\in\,\{1,\ldots,N_{sq}\}}$ and obtain

$$J(\vec{x}) = \frac{|\left\{S_{ref,j}\right\} \cap \left\{S_j(\vec{x})\right\}|}{|\left\{S_{ref,j}\right\} \cup \left\{S_j(\vec{x})\right\}|}. \tag{3.14}$$

The last two coefficients defined will be interesting when looking at their maxima, among the values computed for each source position.

Part IV

Results

Chapter 4

Results

4.1 The parameters' choice

Before running the simulations it is necessary to set all the parameters of the model. For some of them it is easy to decide their values, for others it is not. In particular, the easy ones are:

- the walking speed $v = 5\,km/h$;
- the maximum number of days to be simulated, fixed at 14 days;
- the length of the segment in which we split the roads when computing the probabilities of infection $\delta x = 1\,m$.

These values will never change throughout the different simulations. Now to r_0 and β_0: these are the "biological" parameters of the system, they are the ones strictly linked to how to disease spread and work. They must be tuned in a heuristic way, through trials and errors. In a first attempt we combined the values (r_0 expressed in m and β_0 in $infected/s$, from now on not explicitly indicated anymore)

$$r_0 = [\,10,\,100,\,1000\,], \quad \beta_0 = [\,10^{-1},\,10^{-2},\,10^{-3},\,10^{-4},\,10^{-8}\,].$$

From this we selected $r_0 = 100$ and $\beta_0 = 10^{-3}$ and then did a second attempt to refine these values. By simulating with the values

$$r_0 = [\,150,\,200\,], \quad \beta_0 = [\,5 \cdot 10^{-3},\,10^{-3},\,10^{-4}\,],$$

we ended up choosing $r_0 = 150$ with $\beta_0 = [\,10^{-3},\,10^{-4}\,]$. Now the reasoning behind this decision. For every combination of these two parameters we did 100 independent runs, counted the number of infected in each of them, averaged the results over the 100 runs and made histograms of the average number of infected per day. The plots and the total average numbers of infected are shown in 4.2 and they explain almost everything about our decision, so I report the explanation here instead of using the image caption, as I usually do. The legends report also the value N, that is the average total number of infected (*i.e.* the sum of the columns of one histogram). Since the average is not an integer but at the

same time it does not make sense to report a fractional number of infected, we rounded the results. That is why there is a $<$ symbol. Figures (a), (b), (c), (d) and (e) shows the results with r_0 fixed and β_0 varying. As expected $r_0 = 10$ is not enough to see something interesting: N is always between 0 and 1, indicating that neither one of the agents is getting infected. Scaling r_0 by one order of magnitude already shows some decreasing behaviour: a big part of the agents living (or spending much time) close to the source get infected almost at the first time they inhale the bacteria. Then, of course, the number can only decrease in time, since the number of potential agents has been reduced. This behaviour is present for any other value of r_0 used, so what becomes interesting now is N: knowing that in real outbreaks this value stay more or less between 10 and 100, the parameters values in (c) seem to be the ones to be selected.
Figure (f) instead has $\beta_0 = 10^{-4}$ fixed and the radius r_0 as the variable parameter. It is interesting since the low value of β_0 shows how the number of infected is almost constant along the two weeks simulated, regardless of r_0. The explanation is that not all the agents living (or working) close to the source get infected at the first contact they have with the bacteria. It is hard to get infected and only sometimes it happens. This well represents the situation where the disease is present on the territory but is not so strong to affect all the people close to it, thus it can not be found immediately.
In figure 4.1 we report the outcome of the single simulation, representing the real case, for $r_0 = 150$ fixed and for $beta_0$ taking the two selected values. Interestingly when $beta_0$ decreases, it becomes really hard to understand where the source is located looking only at the houses of the infected agents (red nodes): they are not well localized inside the dangerous region depicted in green.

4.2 Comparing the methods

We now fix $r_0 = 150$, $\beta_0 = 10^{-4}$, the real source to be the one of figure 4.1 and compare the heat-maps resulting from the application of the five different methods previously described. Everything can be seen in figure 4.3. A unique set of colors is associated to each of the methods: red for χ^2_{size}, gray for $\mathbb{P}_{size}$, purple for χ^2_{time}, green for χ^2_{space} and blue for $J(\vec{x})$.

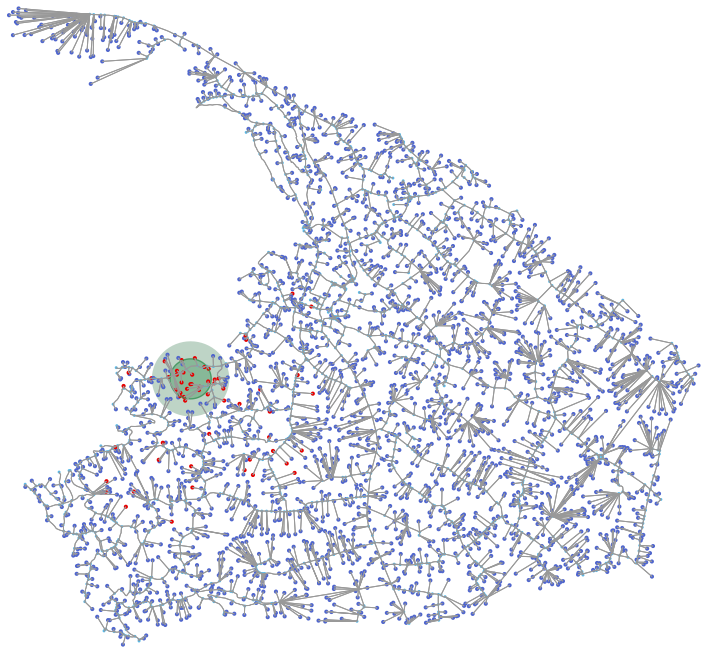
The coefficients χ^2_{size} and $\mathbb{P}_{size}$ (figures 4.3 (a) and (b)) are clearly useless: the heatmaps are flat and no information can be extracted. We can say that knowing only the outbreak size is not sufficient to know a probable location of the real source. Same conclusion for χ^2_{time} (figure **??**): even though some color gradient appears, almost all the map represents a minimum in the χ^2-value. The reported plot is the result of the computation using a $\Delta t = 1\,h$, but changing this value to $6\,h$ or even $1\,day$ does not change the outcome. Finally some good results come out as we take into account the outbreak sizes at the single-square level: both χ^2_{space} and the Jaccard index methods were able to identify the real source location. Of course we are not talking about the precise position, but the squares containing it (in the depicted case the real source acts in a region between two contiguous squares). The χ^2_{space} method looks promising and the output is a quite broadened distribution over the map: the minimum is evident, but then many shades of green cover the rest of the map, showing a smooth transition between the squares. On the contrary the Jaccard index shows a quite definite peak in the region of the real source position: J is flat on almost all the map and jumps to a high value close to the inferred position. We repeated the

computation for many different real source positions and the results were always the same (the plots are not reported because they are really identical, especially for χ^2_{size} and $\mathbb{P}_{size}$): χ^2_{space} and J being really precise while χ^2_{size}, $\mathbb{P}_{size}$ and χ^2_{time} being almost useless. For this reason we stop analyzing the bad coefficients. We go on with two good ones.

4.3 Analyzing χ^2_{space} and the Jaccard index J

Figure 4.4 resumes the heatmaps for χ^2_{space} using two different real source positions and two β_0 values. Figure 4.5 shows the equivalent results for the Jaccard index J. The two coefficients show the same behaviour, so the following analysis is valid for both. Varying β_0, we observe that increasing the value of β_0 the color distribution smoothens around the extremum (whether it is the maximum for J or the minimum for χ^2_{space}). In other words, the colors are more uniform around the extremum, or the peak is less high relative to its surroundings. This is due to the fact that as β_0 increases, the probability of getting infected increases as well and so does the number of infected (and thus the outbreak size), as seen in figures 4.2. All the agents living near the source have associated a higher probability of infection, and for a given outbreak size s, the number of combinations of agents infected, giving rise to that s, is greater. This is true for both the source positions.

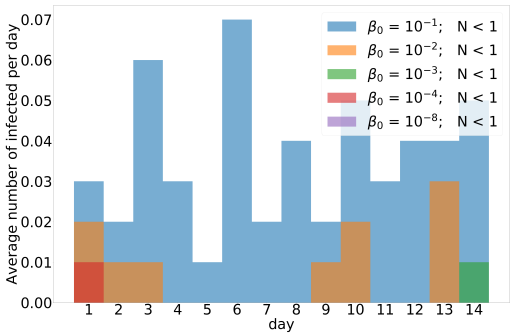
Comparing the two heatmaps for a given source, it is evident that the relative color-difference between contiguous squares looks unaltered even when changing β_0. The fluctuations are really small. It is the result of averaging 1000 runs outcomes.



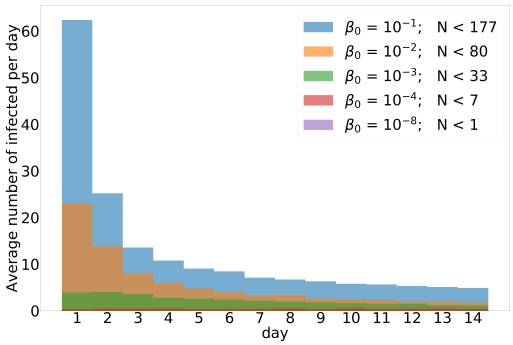
(a) $r_0 = 150\,,\ \beta_0 = 10^{-3}$

(b) $r_0 = 150\,,\ \beta_0 = 10^{-4}$

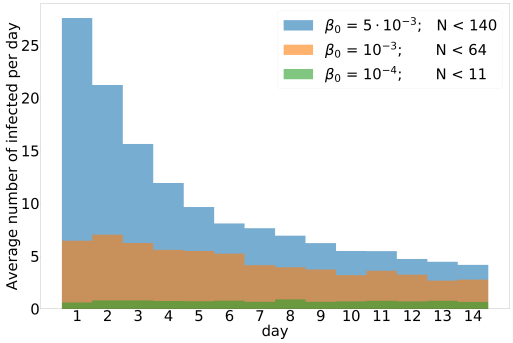
Figure 4.1: The single simulations representing the real outbreaks event for our analysis.



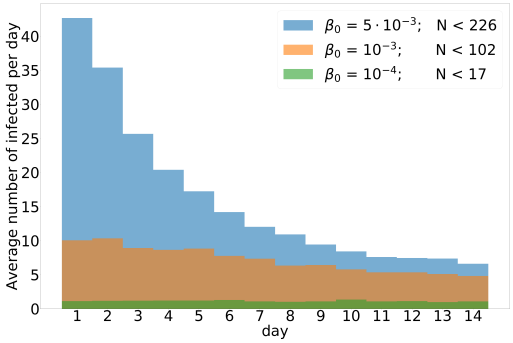
(a) $r_0 = 10$



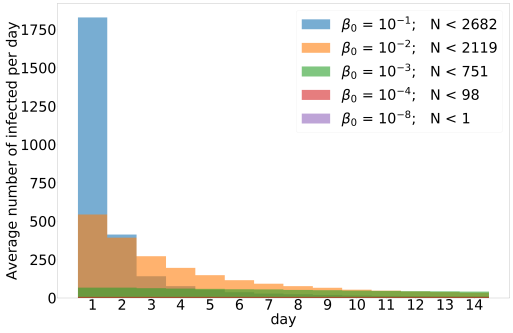
(b) $r_0 = 100$



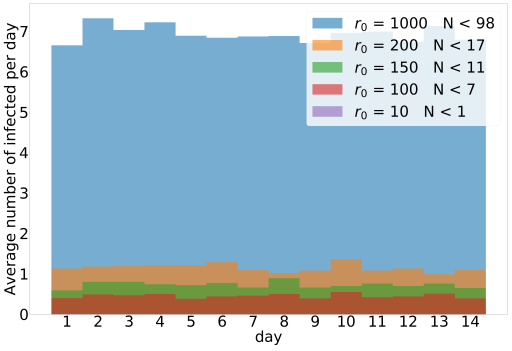
(c) $r_0 = 150$



(d) $r_0 = 200$

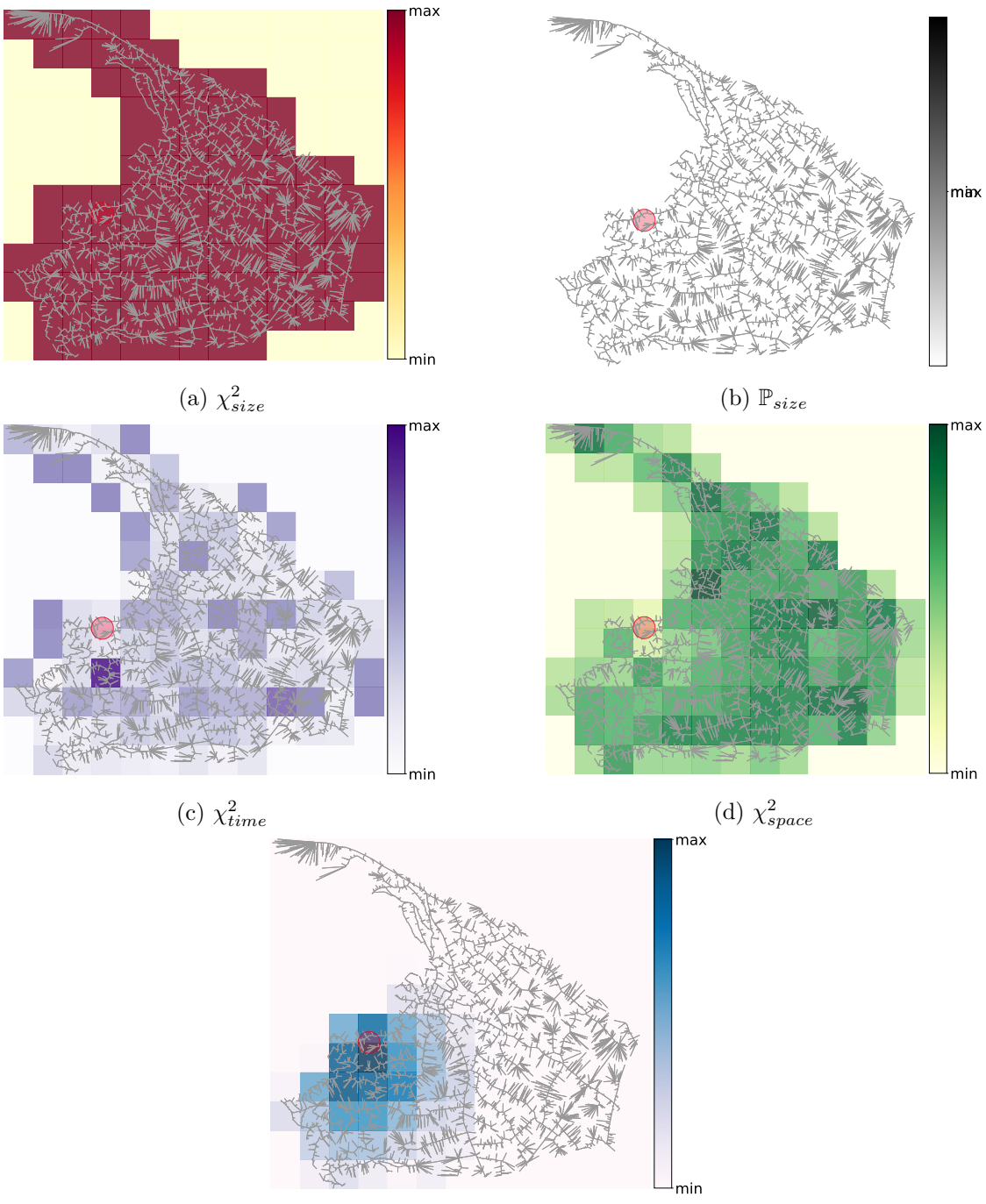


(e) $r_0 = 1000$



(f) $\beta_0 = 10^{-4}$

Figure 4.2: Histograms showing the average number of infected per day obtained in simulations using different values of r_0 and β_0. For the details see section 4.1.

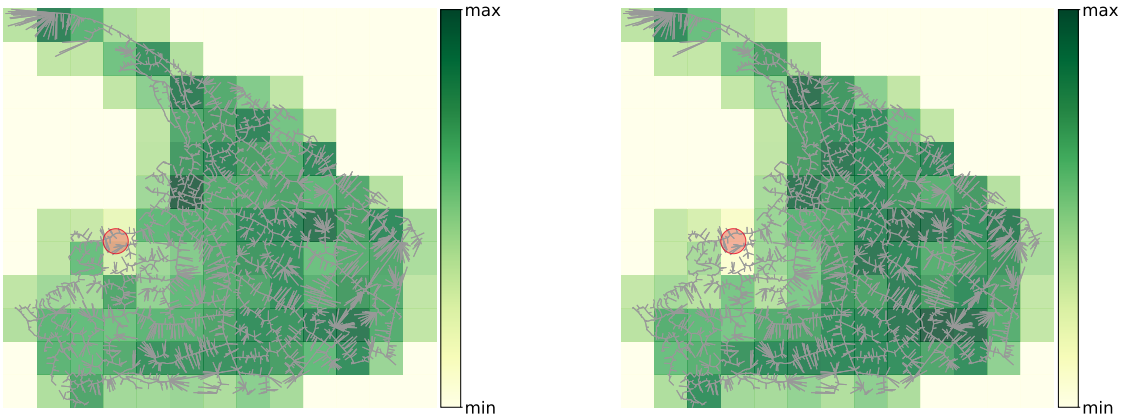


(a) χ^2_{size} (b) $\mathbb{P}_{size}$

(c) χ^2_{time} (d) χ^2_{space}

(e) J

Figure 4.3: Comparing the five different inverse problem approaches, for $r_0 = 150$, $\beta_0 = 10^{-4}$. The red circle indicates the real position of the source that the methods should be able to infer. (a), (c) and (d) represent χ^2 values, so the inferred source position is located in the minimum, *i.e.* the clearest square; for (b) and (e) the situation is reversed and the inferred position is in the maxim, or darkest square.



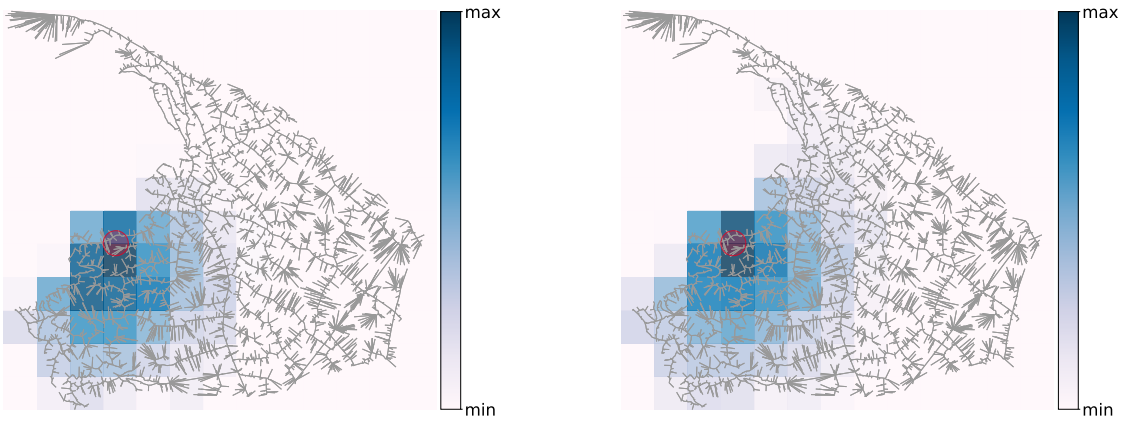
(a) source 0, $\beta_0 = 10^{-4}$

(b) source 0, $\beta_0 = 10^{-3}$

(c) source 1, $\beta_0 = 10^{-4}$

(d) source 1, $\beta_0 = 10^{-3}$

Figure 4.4: Comparing χ^2_{space} outputs for two different real source positions (one per row) and two different values of β_0 used in the simulations (one per column).



(a) source 0, $\beta_0 = 10^{-4}$

(b) source 0, $\beta_0 = 10^{-3}$

(c) source 1, $\beta_0 = 10^{-4}$

(d) source 1, $\beta_0 = 10^{-3}$

Figure 4.5: Comparing Jaccard index J outputs for two different real source positions (one per row) and two different values of β_0 used in the simulations (one per column).

Part V

Conclusions & Comments

Chapter 5

Conclusions

5.1 Developed work

We developed a model able to emulate people walking on a real road map, following some agenda (defined as consecutive stay-points on the map creating a loop, so that the first and last points coincide and represent the agent's home). It positions the source of the Legionella disease on the map and tunes it with its infection parameters, such as r_0 and β_0, deeply described in this thesis. The closer to the source an agent walks, the higher the probability of getting infected. After one simulation we are able to compute the outbreak size (defined as the number of infected people) and its time evolution. Having one simulation run as the reference(or real) data, we iteratively used the model placing the source in many different places: each place is the centre of a square that is part of the grid covering all the map. We finally tested different coefficients whose aim is to compute how similar the simulations are to the real case. Some of them failed, others succeeded, confirming that the model works: comparing the outbreak size at a single square (local) level using the χ^2_{space} and Jaccard J coefficients, the model is able to infer the constrained region in which the real source is placed.

5.2 Possible future work

We started the project with almost no references, so much time was spent to find to good direction to follow. Now that the foundations are solid and confirmed, a lot of possibilities are available for further developments. A first thing to do is to try and push the system to its limits. As for any physical model, it is interesting and useful to understand what are the precise ranges of parameters under which the model is working. This also helps to find out when the model is reasonable and when it is not. Varying the size of the squares is also a valuable development. Another important study to do is a quantitative measure of how good the coefficients used are, together with a quantitative comparison between them. Developing other metrics to infer the source position is another branch of research and can lead to even better performances of the model. Then of course, as anticipated in the first chapters of the thesis, using different real data, such as GPS tracks of people's movements and census data will make the model more data-driven and can improve it really fast. This

unfortunately requires some agreements between universities and private corporations and in our case it was not possible.

Bibliography

[1] W. Foundation, *List of Legionnaire's disease outbreaks*, Jun. 18. [Online]. Available: https://en.wikipedia.org/wiki/List_of_Legionnaires%27_disease_outbreaks.

[2] ultimahora.es, *Investigan un brote de legionela en Palmanova con 19 afectados y 1 fallecido*, Oct. 2017. [Online]. Available: https://ultimahora.es/noticias/part-forana/2017/10/18/300579/investigan-brote-legionella-palmanova-afectados-fallecido.html.

[3] diariodemallorca.es, *Un jacuzzi de exteriores, foco del brote de legionela de Palmanova*, Dec. 2017. [Online]. Available: https://www.diariodemallorca.es/mallorca/2017/12/19/jacuzzi-exteriores-foco-brote-legionela/1273059.html.

[4] W. H. Organization, *Legionella and the prevention of legionellosis.*

[5] P. S. W. et al., «Epidemiological investigation of a Legionnaires' disease outbreak in Christchurch, New Zealand: the value of spatial methods for practical public health.», *Epidemiology and Infection*, vol. 141, no. 4, pp. 789–99, Apr. 2013.

[6] R. P. e. a. A. Magnet, «Vectorial role of Acanthamoeba in Legionella propagation in water for human use», *Science of The Total Environment*, vol. 505, pp. 889–95, Feb. 2015.

[7] M. e. a. Yamamoto Hiroyuki; Sugiura, «Factors stimulating propagation of legionellae in cooling tower water», *APPLIED AND ENVIRONMENTAL MICROBIOLOGY*, vol. 58, pp. 1394–97, Apr. 1992.

[8] T. M. Nhu Nguyen, D. Ilef, S. Jarraud, L. Rouil, C. Campese, D. Che, S. Haeghebaert, F. Ganiayre, F. Marcel, J. Etienne, and J.-C. Desenclos, «A Community-Wide Outbreak of Legionnaires Disease Linked to Industrial Cooling Towers—How Far Can Contaminated Aerosols Spread?», *The Journal of Infectious Diseases*, vol. 193, no. 1, pp. 102–111, 2006. eprint: /oup/backfile/content_public/journal/jid/193/1/10.1086/498575/2/193-1-102.pdf.

[9] A. A. e. a. Carmen Armero, «A probabilistic expert system for predicting the risk of Legionella in evaporative installations», *Expert Systems with Applications*, vol. 38, pp. 6637–43, Jun. 2011.

[10] M. E. Schoen and N. J. Ashbolt, «An in-premise model for Legionella exposure during showering events», *Water Research*, vol. 45, no. 18, pp. 5826–5836, 2011, ISSN: 0043-1354.

[11] H. I. M. e. a. Bull M., «The application of geographic information systems and spatial data during Legionnaires' disease outbreak responses», *Euro Surveill*, 2012.

[12] S. Salini, *An ordinary weekly agenda*, Apr. 2018. [Online]. Available: `http://www.surveygalaxy.com/surPublishes.asp?k=9ANCQ9X2B9AV`.

[13] P. White, F. Graham, D. Harte, M. Baker, C. Ambrose, and A. Humphrey, «Epidemiological investigation of a Legionnaires' disease outbreak in Christchurch, New Zealand: The value of spatial methods for practical public health», vol. 141, pp. 1–11, Jun. 2012.

[14] O. W. contributors, *Open Street Map*, Jul. 2014. [Online]. Available: `http://wiki.openstreetmap.org/w/index.php?title=Main_Page&oldid=1060762`.

[15] G. Boeing, «OSMnx: New Methods for Acquiring, Constructing, Analyzing, and Visualizing Complex Street Networks.», *Computers, Environment and Urban Systems*, no. 65, pp. 126–139, 2017.

[16] A. A. Hagberg, D. A. Schult, and P. J. Swart, «Exploring Network Structure, Dynamics, and Function using NetworkX», in *Proceedings of the 7th Python in Science Conference*, G. Varoquaux, T. Vaught, and J. Millman, Eds., Pasadena, CA USA, 2008, pp. 11–15.

[17] C. Song, T. Koren, P. Wang, and A.-L. Barabasi, «Modelling the scaling properties of human mobility», vol. 6, Oct. 2010.

[18] A. del Transport Metropolità, «Enquesta de Mobilitat en Dia Feiner», p. 2, 2016.

[19] S. Sawlani, «Explaining the Performance of Bidirectional Dijkstra and A* on Road Networks», Master's thesis, University of Denver, 2017.

[20] P.Jaccard, «Étude de la distribution florale dans une portion des Alpes et du Jura», *Bulletin de la Societe Vaudoise des Sciences Naturelles*, vol. 37, no. 142, pp. 547–579, Jan. 1901.